ITRIA 2015 | COMPUTATIONAL METHODS IN DATA ANALYSIS

ITRIA 2015

COMPUTATIONAL METHODS IN DATA ANALYSIS

INSTITUTE OF COMPUTER SCIENCE
POLISH ACADEMY OF SCIENCES

INFORMATION TECHNOLOGIES: RESEARCH
AND THEIR INTERDISCIPLINARY APPLICATIONS

COMPUTATIONAL METHODS IN DATA ANALYSIS

Warsaw, 22-24 October 2015
ITRIA 2015

INSTITUTE OF COMPUTER SCIENCE
POLISH ACADEMY OF SCIENCES

Warsaw, 2015

Publication issued as a part of the project:
"Information technologies: research and their interdisciplinary applications",
Objective 4.1 of Human Capital Operational Programme.
Agreement number UDA-POKL.04.01.01-00-051/10-01.

Publication is co-financed by European Union from resources of European Social Fund.

Project leader: Institute of Computer Science, Polish Academy of Sciences

Project partners: System Research Institute, Polish Academy of Sciences, Nałęcz Institute of Biocybernetics and Biomedical Engineering, Polish Academy of Sciences

Reviewers: Michał Baczyński
Piotr Bogorodzki
Michał Dąbrowski
Przemysław Klęsk
Henryk Komorowski
Jacek Koronacki
Marcin Korzeń
Mikołaj Morzy
Zbigniew Nahorski
Paweł Teisseyre
Ewa Szczurek
Rafał Weron

Layout: Grzegorz Murzynowski
Cover design: Waldemar Słonina

Publication is distributed free of charge

ISBN 978-83-63159-22-1

Table of Contents

Clustering and Aggregation of Informetric Data Sets

Anna Cena[1, 2] and Marek Gągolewski[1, 3]

[1] Systems Research Institute, Polish Academy of Sciences
ul. Newelska 6, 01-447 Warsaw, Poland
[2] Warsaw University of Technology, Faculty of Mathematics and Information Science,
ul. Koszykowa 75, 00-662 Warsaw, Poland

Abstract. This paper presents recent developments on clustering algorithms designed to deal with numeric strings, i.e., non-increasingly ordered numeric vectors of possible varying lengths. Such objects can be found in real-world data sets in the field of informetrics. Investigation carried out in this paper focuses on partitional clustering algorithms. The genetic approach proposed in this paper as well as the K-means algorithm introduced previously are investigated from, both, machine learning and aggregation theory perspective. Also, a projection of original data into a space of fixed number of indices is considered.

1 Introduction

Informetrics is an active field of research, which mostly deals with measurable aspects of information processes. One of the main informetric tasks is the so called Producers Assessment Problem (PAP) in which we would like to evaluate a set of producers of information resources according to, both, the quantity of information they output and its quality. PAP is often identified with bibliometrics, where a scientist is a producer and scientific articles he/she published are products. Moreover, the quality of each paper is often measured by the number of citation it received. However, application of PAP exceeds beyond that. Let us consider for example on-line social networking services, like "Facebook", "Twitter" or "Stack Exchange". Each active user is a producer of new information items that are assessed by the members of the on-line community (cf., e.g., "Like", "Share", "Follow", "UpVote", or "DownVote" buttons).

The nature of informetric data may be situated "somewhere between" multidimensional real data and the character string domain. On the one hand, observations are real numbers but, on the other, their number is not established a priori. Informetric data sets consisting of non-increasingly ordered vectors of unequal lengths are examples of numeric strings [1]. Most often, such vectors

are analyzed by the means of aggregation theory [3], for example with various bibliometric indexes [4,5]. Recently, the usage of unsupervised machine learning techniques in the field of informetrics is studied too.

In this paper we focus on various clustering algorithms that may be applied on informetric data in order to automatically discover diverse groups of producers. Such methods are crucial not only in identification and/or description of certain groups of producers (productive, high impact, low impact, etc. ones), but also may be used in automated informetric decision support systems.

One of the possible approaches to apply clustering techniques on vectors of nonconforming lengths is to reduce the data dimension by considering a fixed number of attributes or indicators, see, e.g., [6,7]. Please note that such an approach has, however, some limitations, like for instance arbitrariness in the choice of considered indexes and their quantity, unstable behavior of some bibliometric indexes when it comes to input data transformation, etc. On the other hand, in [8] a class of modified metrics was proposed so that they can be applied on vectors of nonconforming lengths. Owing to that, i.a., hierarchical clustering algorithms may be used in order to determine an input data set's partition consisting of sets of homogeneous producers. What is more, a K-means-like algorithm together with a more general c-means algorithm based on modified dissimilarity measure were proposed and studied in [2,9], respectively.

In this paper we extend the mentioned results. First of all, connections between the modified clustering techniques for informetric data and aggregation theory are investigated. The notion of cluster centers as an aggregated representation of all vectors from a given cluster was partially studied in [9]. Nevertheless, that study is far from being complete. Moreover, please note that the k-means algorithm is just a heuristic and therefore, especially for unbalanced data, may return results far from being optimal. Thus, in this paper a genetic algorithm designed for informetric data clustering is proposed and studied as well. In the second place, a comparative study of various informetric data sets (e.g. Stack Exchange data base, dependency network of R packages, Elsevier's Scopus citations base) including proposed procedures and mentioned above projection to fixed number of indexes approach is presented.

The paper is organized as follows: Sec. 2 reviews the recent results concerning clustering algorithms for informetric data and proposes a new genetic solution. Next, in Sec. 3 the empirical analysis is performed. Finally, Sec. 4 concludes the paper and gives a future research results.

2 Clustering

Clustering techniques are usually classified as either partitional or hierarchical [12]. The former class of algorithms directly divide data points into some predefined number of clusters, see, e.g., the k-means procedure, while the latter class of methods determines the whole hierarchy of possible data partitioning schemes, level by level, which may be cut at an arbitrary height. Both groups, however, require the definition of the measure that can be used to as-

sess the dissimilarity between observations. Classically, this is achieved with the notion of a metric, i.e., a function $\mathfrak{d} : X \times X \to [0, \infty)$ such that for any $\mathbf{x}, \mathbf{y}, \mathbf{z} \in X$ it holds: (a) $\mathfrak{d}$ is symmetric, (b) $\mathfrak{d}$ fulfills the triangle inequality, i.e. $\mathfrak{d}(\mathbf{x}, \mathbf{y}) \leq \mathfrak{d}(\mathbf{x}, \mathbf{z}) + \mathfrak{d}(\mathbf{z}, \mathbf{y})$, and (c) $\mathfrak{d}(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$. Consider for example the well known Minkowski distance given by $\mathfrak{d}(\mathbf{x}, \mathbf{y}) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p}$, for $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, $p \geq 1$. Please note that for $p = 1$, Minkowski distance is simply the Manhattan distance $\mathfrak{d}_{L_1} = \sum_{i=1}^{n} |x_i - y_i|$, and for $p = 2$ the Euclidean distance $\mathfrak{d}_{L_2} = \left(\sum_{i=1}^{n} |x_i - y_i|^2 \right)^{1/2}$. Let us note that if $\nu : X \times X \to [0, \infty)$ is a function such that fulfills (a), (b), and a relaxed version of (c), namely, $(\mathbf{x} = \mathbf{y}) \implies \nu(\mathbf{x}, \mathbf{y}) = 0$, then ν is called a pseudometric.

As it was stated above, the considered informetric data may be situated "somewhere between" multidimensional real data and the character string domain. On the one hand, observations are on the interval scale but, on the other, their quantity is not established a priori. Therefore, the distance function that can capture the similarity/dissimilarity between such objects shall be introduced.

Let $\mathcal{S} := \{(x_1, \ldots, x_n) \in \bigcup_{n \geq 1} \mathbb{R}^n : x_1 \geq x_2 \geq \cdots \geq x_n\}$ denote the space of non-increasingly ordered numeric vectors of arbitrary length and $\mathcal{X} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(l)}\}$, where $\mathbf{x}^{(i)} = \left(x_1^{(i)}, \ldots, x_{n_i}^{(i)}\right)$ for all $i = 1, \ldots, l$, its finite subset representing input data points. Moreover, let $\mathcal{S}_n = \{(x_1, \ldots, x_n) \in \mathbb{R}^n, x_1 \geq \cdots \geq x_n\}$ denote the subset of $\mathcal{S}$ consisting of vectors of length n and $\mathcal{S}_{\leq n} = \bigcup_{i=1}^{n} \mathcal{S}_i$, $n \in \mathbb{N}$ the subset of $\mathcal{S}$ with vectors of length not greater than n. It is clear to see, that $\mathcal{X} \subset \mathcal{S}_{\leq m}$, where $m = \max\{|\mathbf{x}|; \mathbf{x} \in \mathcal{X}\}$. The following theorem (see [8] for the proof) defines the class of metrics on $\mathcal{S}_{\leq m}$, for any $m \in \mathbb{N}$.

Theorem 1. *(Cena, Gągolewski, Mesiar [8]) Let $d_M : \mathcal{S}_{\leq m} \times \mathcal{S}_{\leq m} \to [0, \infty)$ be such that $d_M(\mathbf{x}, \mathbf{y}) = \mathfrak{d}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) + \nu(\mathbf{x}, \mathbf{y})$, where $\tilde{\mathbf{x}} = (x_1, x_2, \ldots, x_n, 0, \ldots, 0) \in \mathcal{S}_m$, $\mathfrak{d}$ is a metric on $\mathbb{R}^m$ and ν is a pseudo-metric on $\mathcal{S}_{\leq m}$. Then d_M is a metric on $\mathcal{S}_{\leq m}$ if and only if for all $\mathbf{x}, \mathbf{y}$ such that $\tilde{\mathbf{x}} = \tilde{\mathbf{y}}$ it holds $\nu(\mathbf{x}, \mathbf{y}) = 0 \implies n_x = n_y$, where $n_x = |\mathbf{x}|$ and $n_y = |\mathbf{y}|$, denote the length of $\mathbf{x}$ and $\mathbf{y}$, respectively.*

In practice, the pseudometric ν might be defined only in terms of vectors' lengths, e.g., for $\mathbf{x}, \mathbf{y} \in \mathcal{S}$, $\nu_{p,q}(\mathbf{x}, \mathbf{y}) = p|n_x^r - n_y^r|$.

Remark 1. It is easily seen that the d_M metric defined as $d_M(\mathbf{x}, \mathbf{y}) = \mathfrak{d}_{L_1}(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) + \nu_{1,1}(\mathbf{x}, \mathbf{y})$ can be decomposed as follows:

$$d_M(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{\min\{n_x, n_y\}} |x_i - y_i| + \sum_{i=\min\{n_x, n_y\}+1}^{n_x} |x_i| + \sum_{i=\min\{n_x, n_y\}+1}^{n_y} |y_i| + |n_x - n_y|,$$

with the convention $\sum_{i=u}^{v} \cdot = 0$ for $u > v$. Hence, d_M is in fact a sum of the distance between the first $\min\{n_x, n_y\}$ largest observations plus a norm of the remaining observations in the longer vector (which is the same as the distance to a vector $\mathbf{0}$ with zeros at each coordinate) plus some penalty for the difference

in vectors' lengths. This raises the association with the Levenshtein distance for strings [1,13], defined as minimal number of single character insertions, deletions and replacements needed to obtain one string form another and provides an appealing interpretation of the proposed solution.

Remark 2. It is easily seen that the dissimilarity measure $d_D : \mathcal{S}_{\leq m} \times \mathcal{S}_{\leq m} \to [0, \infty)$, i.e., function fulfilling conditions (a) and (c), but not (b) – triangle inequality, can be defined in the manner presented in Theorem 1, i.e., $d_D(\mathbf{x}, \mathbf{y}) = \mathfrak{d}'(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) + \nu(\mathbf{x}, \mathbf{y})$, were $\mathfrak{d}'$ is a dissimilarity measure on $\mathbb{R}^m$ and ν is pseudometric on $\mathcal{S}_{\leq m}$.

2.1 K-means-like algorithm

Classically, in the Euclidean space a partitional clustering task can be defined as follows. Given a set of observations $\mathcal{Y} = \{\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(l)}\}$, where each $\mathbf{y}^{(i)} \in \mathbb{R}^n$, we aim at partitioning the l observations into k nonempty pairwise disjoint sets $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$, $\bigcup_{i=1}^{k} C_i = \mathcal{Y}$, so that:

$$\mathcal{C} = \underset{\text{partition } \mathcal{C} \text{ of } \mathcal{Y}}{\arg\min} \sum_{i=1}^{k} \sum_{\mathbf{y} \in C_i} d_{L_2}^2 \left(\mathbf{y}, \boldsymbol{\mu}^{(i)} \right), \tag{1}$$

where $\boldsymbol{\mu}^{(i)}$ is the centroid of all the vectors in C_i, $\mu_j^{(i)} = \sum_{\mathbf{y} \in C_i} y_j / |C_i|$, and $d_{L_2}^2(\mathbf{y}, \boldsymbol{\mu}) = \sum_{j=1}^{n} (y_j - \mu_j)^2$ is the squared Euclidean distance.

As the problem stated in Eq. (1) is known to be NP-complete [14], the following heuristic – K-means algorithm, see [15], may be used. For the initial set of cluster centers, do what follows until convergence occurs:

1. Assign each point in $\mathcal{Y}$ to the cluster with the nearest center,
2. Recalculate cluster centers by computing the means $\boldsymbol{\mu}^{(1)}, \ldots, \boldsymbol{\mu}^{(k)}$ of all the points assigned to particular clusters.

$d_{D;p,q}$-centroid In the informetric settings, we may consider the dissimilarity measure based on squared Euclidean distance, i.e.,

$$d_{D;p,q}(\mathbf{x}, \mathbf{y}) = d_{L_2}^2(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) + p|n_x^r - n_y^r|.$$

Therefore Eq. (1) can be redefined as

$$\mathcal{C} = \underset{\text{partition } \mathcal{C} \text{ of } \mathcal{Y}}{\arg\min} \sum_{i=1}^{k} \sum_{\mathbf{y} \in C_i} d_{D;p,q} \left(\mathbf{y}, \boldsymbol{\mu}^{(i)} \right), \tag{2}$$

where $\boldsymbol{\mu}^{(i)} \in \mathcal{S}$ is a centroid of C_i. Thus, in order to derive a K-means like procedure for informetric data, first we have to provide a method for computing a $d_{D;p,q}$-centroid of a set of vectors $\mathcal{X} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(l)}\} \subseteq \mathcal{S}$, i.e.,

$$\boldsymbol{\mu} = \underset{\boldsymbol{\mu} \in \mathcal{S}}{\arg\min} \sum_{i=1}^{l} d_{D;p,q}(\mathbf{x}^{(i)}, \boldsymbol{\mu}) := \underset{\boldsymbol{\mu} \in \mathcal{S}}{\arg\min} F(\boldsymbol{\mu}). \tag{3}$$

In [2] it was shown that the length of $\boldsymbol{\mu}$ which is a minimizer of F cannot be greater than m and, hence, the task defined via Eq. (3) can be decomposed as follows. For $n = 1, \ldots, m$ determine:

$$\boldsymbol{\mu}^{(n)} = \operatorname*{arg\,min}_{\boldsymbol{\mu} \in \mathcal{S}_n} F(\boldsymbol{\mu}) \tag{4}$$

and then compute:

$$\boldsymbol{\mu} = \operatorname*{arg\,min}_{n=1,\ldots,m} F(\boldsymbol{\mu}^{(n)}). \tag{5}$$

Please note that the problem defined by Eq. (4), is in fact a constrained optimization problem, as our search space consists of vectors that are sorted non-increasingly: we have that $\mu_1^{(n)} \geq \mu_2^{(n)} \geq \cdots \geq \mu_n^{(n)}$.

To present the solution to Eq. (4) given in [2], let us recall the notion of a contiguous partition of an index set $[n] := \{1, 2, \ldots, n\}$, that is a set of nonempty, disjoint sets of consecutive elements in $[n]$. In other words, $\mathcal{P} \subseteq 2^{[n]}$ is a contiguous partition of $[n]$ if $\bigcup_{P \in \mathcal{P}} = [n]$, $P \cap P' = \emptyset$, $|P| > 0$, $\{i, j\} \in P$ with $i \leq j$ implies that $i + 1, i + 2, \ldots, j - 1 \in P$ for all $P \neq P'$. The whole class of such contiguous partitions will from now on be denoted as $\mathcal{CP}([n])$. It might be shown that $|\mathcal{CP}([n])| = 2^{n-1}$. For example, we have:

$$\mathcal{CP}([3]) = \left\{ \begin{array}{c} \big\{\{1\},\{2\},\{3\}\big\}, \\ \big\{\{1,2\},\{3\}\big\}, \\ \big\{\{1\},\{2,3\}\big\}, \\ \big\{\{1,2,3\}\big\} \end{array} \right\}.$$

Given $\mathcal{P} \in \mathcal{CP}([n])$ and $i \in [n]$, let $P_{\{i\}}$ stand for an element in $\mathcal{P}$ such that $i \in P_{\{i\}}$. Moreover, let $P^{(i)}$ be the ith ordered element in $\mathcal{P}$, i.e., such that for $1 \leq i < j \leq |\mathcal{P}|$ it holds $\max P^{(i)} < \min P^{(j)}$. Assuming that $\tilde{x}_i = \sum_{\mathbf{x}:|\mathbf{x}| \geq i} x_i$ we have what follows (see [2] for the proof).

Theorem 2. *(Cena, Gągolewski [2]) Fix $n \in [m]$ and let $\mathcal{P} \in \mathcal{CP}([n])$. Define $\mathbf{y} \in \mathbb{R}^n$ as:*

$$y_i = \frac{1}{l|P_i|} \sum_{j \in P_i} \tilde{x}_j \quad \text{for } i = 1, \ldots, n.$$

If $y_1 \geq y_2 \geq \cdots \geq y_n$ and for all $i \in [n]$ with $i \in (P_{\{i\}} \setminus \{\max P_{\{i\}}\})$ we have

$$\frac{i - \min P_{\{i\}} + 1}{|P_{\{i\}}|} \sum_{j \in P_{\{i\}}} \tilde{x}_j - \sum_{j \in P_{\{i\}}, j \leq i} \tilde{x}_j > 0,$$

then $\mathbf{y}$ is a solution to Eq. (4).

Theorem 2 induces a simple algorithm to determine $\mu^{(n)} \in \mathcal{S}_n$. One may consider every possible contiguous partition of $[n]$ and then verify if the conditions listed in the theorem are met.

Example 1. Let us consider a simple exemplary set consisting of three vectors of the form:

$$\mathcal{X} = \left\{ \begin{array}{l} (\quad 7, \quad 5, \quad 1, 0, 0\ ), \\ (\quad 1, -2, -5 \qquad\ \ ), \\ (-12 \qquad\qquad\quad\ \ ) \end{array} \right\}.$$

It is easily seen that the maximum of vectors' lengths is equal to $m = 5$. Based on the procedure provided by Theorem 2, let us derive the $d_{D;1,1}$-centroid of length $n = 3$. The vector $\tilde{\mathbf{x}}$ is of the form $(-4, 3, -4)$. As it was stated above, in order to do so, we have to consider all contiguous partitions of a set $[n]$:

(i) $\mathcal{P} = \{\{1\}, \{2\}, \{3\}\}$; Here the candidate solution is of the form $\mathbf{y} = (-1\frac{1}{3}, 1, -1\frac{1}{3})$ and it is clear to see that this solution does not fulfill required ordering.
(ii) $\mathcal{P} = \{\{1, 2\}, \{3\}\}$; Thus, $\mathbf{y} = (-\frac{1}{6}, -\frac{1}{6}, -1\frac{1}{3})$. Since the vector $\mathbf{y}$ is non-increasingly ordered, we have to check the optimality conditions: $\frac{1}{2}(-4 + 3) + 4 = 3.5 > 0$.
(iii) $\mathcal{P} = \{\{1\}, \{2, 3\}\}$; Therefore $\mathbf{y} = (-1\frac{1}{3}, -\frac{1}{6}, -\frac{1}{6})$. Since, the ordering is not preserved, this candidate solution is rejected.
(iv) $\mathcal{P} = \{\{1, 2, 3\}\}$; Here $\mathbf{y} = (-\frac{5}{9}, -\frac{5}{9}, -\frac{5}{9})$ with conditions $\frac{1}{2}(-4 + 3) + 4 = 3.5 > 0$ and $\frac{1}{3}(-4 + 3 - 4) + (-4 + 3) = -2\frac{2}{3} < 0$. As the second condition is not fulfilled, this candidate solution has to be rejected.

Therefore, the solution is equal to $\mathbf{y} = (-\frac{1}{6}, -\frac{1}{6}, -1\frac{1}{3})$

Such a routine, being of course mathematically correct, is unfortunately practically unusable. Therefore, to solve Eq. (4), the algorithm which runs in $O(n^2)$ time, see the Algorithm 1, was proposed and the following theorem holds (see [2] for more details and the proof):

Theorem 3. *(Cena, Gągolewski [2]) Fix n and let $\mathcal{X} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(l)}\}$. If $\mathbf{y}$ is the result of applying Algorithm 1, then $\mathbf{y} = \arg\min_{\mathbf{y} \in \mathcal{S}_n} F(\mathbf{y})$.*

Example 2. Let us again focus on a data set discussed in Example 1. Here is the output of Algorithm 1 for each $n = 1, 2, \ldots, 5$. The optimal solution is obtained for $n = 3$.

```
xtilde=    | -1.333  1.000 -1.333  0.000  0.000
---------- | ----------------------------------------
n  dist    |  y1     y2     y3     y4     y5
1 249.67   | -1.333
2 253.83   | -0.167 -0.167
3 *247.50* | -0.167 -0.167 -1.333
4 251.17   | -0.167 -0.167 -0.667 -0.667
5 253.06   | -0.167 -0.167 -0.444 -0.444 -0.444
```

Moreover, the generalization of Theorem 2 and Theorem 3 to a fuzzy clustering was derived and discussed in [9].

Data: A set of l vectors $\mathcal{X} \subset \mathcal{S}(\mathbb{I})$ and $n \in \mathbb{N}$.
Result: $\boldsymbol{\mu}^{(n)} = \arg\min_{\boldsymbol{\mu} \in \mathcal{S}_n} F(\boldsymbol{\mu})$.
Let $\tilde{\mathbf{x}}$ be such that $\tilde{x}_i = \sum_{\mathbf{x}:|\mathbf{x}|\geq i} x_i$, $i \in [n]$;
Let $\mathcal{P} = \emptyset$;
Let $\mathbf{y} \in \mathbb{R}^n$;
for $k = 1, 2, \ldots, n$ **do**
  $y_k = \tilde{x}_k / l$;
  Let $\mathcal{P} := \mathcal{P} \cup \big\{\{k\}\big\}$;
  while $|\mathcal{P}| > 1$ *and* $y_{\min P^{(|\mathcal{P}|)}} > y_{\max P^{(|\mathcal{P}|-1)}}$ **do**
    $\mathcal{P} := \Big(\big(\mathcal{P} \setminus \{P^{(|\mathcal{P}|)}\}\big) \setminus \{P^{(|\mathcal{P}|-1)}\}\Big) \cup \{P^{(|\mathcal{P}|)} \cup P^{(|\mathcal{P}|-1)}\}$;
    for $i \in P^{(|\mathcal{P}|)}$ **do**
      Set $y_i := \frac{1}{l|P^{(|\mathcal{P}|)}|} \sum_{j \in P^{(|\mathcal{P}|)}} \tilde{x}_j$;
    end
  end
end
return $\mathbf{y}$;

Algorithm 1: An algorithm to solve Eq. (4).

Properties of the $d_{D;p,q}$-centroid In the context of aggregation theory, the procedure of determining the $d_{D;p,q}$-centroid is a fusion function in sense of [1,10,11], which combines a set of objects into an representative object of the same type.

Definition 1. *Let $\mathcal{F} : \mathcal{S}^l \to \mathcal{S}$ be a fusion function such that*

$$\mathcal{F}(\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(l)}) = \arg\min_{\boldsymbol{\mu} \in \mathcal{S}} \sum_{i=1}^{l} d_{D;p,q}(\mathbf{x}^{(i)}, \boldsymbol{\mu}). \tag{6}$$

Please note that $\mathcal{F}$ may be computed via Algorithm 1. Let us now consider basic properties proposed in [1] suitable for such functions.

Remark 3. (Cena, Gągolewski [9]) Unfortunately, the function $\mathcal{F}$ defined by Eq. (6) is not a $\trianglelefteq$-monotonic fusion function, where $\trianglelefteq$ is the partial ordering:

$$\mathbf{x} \in \mathcal{S}_n \trianglelefteq \mathbf{y} \in \mathcal{S}_m \Leftrightarrow n \leq m \text{ and } x_i \leq y_i \text{ for all } i \in [n].$$

Let us consider for example

$$\mathcal{X} = \{\mathbf{x}^{(1)} = (10, 2, 1, 0, 0), \mathbf{x}^{(2)} = (-11), \mathbf{x}^{(3)} = (-5, -6, -10)\}$$

and

$$\mathcal{Y} = \{\mathbf{y}^{(1)} = (10, 2, 1, 0, 0), \mathbf{y}^{(2)} = (10, -100), \mathbf{y}^{(3)} = (-5, -6, -10)\}.$$

It is clear to see that for each $i = 1, 2, 3$ we have $\mathbf{x}^{(i)} \trianglelefteq \mathbf{y}^{(i)}$. However, for the corresponding centroids we have $(-1.67, -1.67, -3) \not\trianglelefteq (5, -34.67)$.

Proposition 1. *The function $\mathcal{F}$ defined by Eq.* (6) *is:*

(i) *idempotent, i.e.* $\mathcal{F}(\mathbf{x},\mathbf{x},\ldots,\mathbf{x})=\mathbf{x}$,
(ii) *symmetric, i.e.,* $\mathcal{F}(\mathbf{x}^{(1)},\mathbf{x}^{(2)},\ldots,\mathbf{x}^{(l)})=\mathcal{F}(\mathbf{x}^{(\sigma(1))},\mathbf{x}^{(\sigma(2))},\ldots,\mathbf{x}^{(\sigma(l))})$, *where* σ *is a permutation of set* $\mathcal{X}$,
(iii) *componentwise internal on common indices, i.e.,* $\mu_i\in[\bigwedge_{j=1}^{l}x_i^{(j)},\bigvee_{j=1}^{l}x_i^{(j)}]$, $i=1,\ldots,\min_{j=1,\ldots,l}|\mathbf{x}^{(j)}|$,
(iv) *global internal, i.e.,* $\mu_{n_\mu}\geq u$ *and* $\mu_1\leq v$, *where* $\boldsymbol{\mu}=\mathcal{F}(\mathcal{X})$ *and* $u=\min_{i=1,\ldots,l}\{x_{n_i}^{(i)}\}$, $v=\max_{i=1,\ldots,l}\{x_1^{(i)}\}$
(v) *length internal, i.e.,* $n_\mu\geq n_{\min}$ *and* $n_\mu\leq n_{\max}$, *where* $\boldsymbol{\mu}=\mathcal{F}(\mathcal{X})$, $n_\mu=|\boldsymbol{\mu}|$ *and* $n_{min}=\min_{i=1,\ldots,l}\{|\mathbf{x}^{(i)}|\}$, $n_{max}=\max_{i=1,\ldots,l}\{|\mathbf{x}^{(i)}|\}$.

Proof. (i) Trivial.
(ii) Trivial.
(iii) Is induced by the properties of averaging fusion functions.
(iv) Assume this does not hold, i.e., $(\exists i)\mu_i>v$ for some i. This implies that $\frac{1}{l|P_{\{i\}}|}\sum_{j\in P_i}\tilde{x}_j>v$. On the other hand, please note that $v<\frac{1}{l|P_{\{i\}}|}\sum_{j\in P_i}\tilde{x}_j\leq x'$, where $x'=\max_{j\in P_i}\{x_j^{(1)},\ldots,x_j^{(l)}\}$. Therefore, $v\neq\max_{\mathbf{x}\in\mathcal{X},i=1,\ldots,n_x}\{x_i\}$, and the proof is complete.
(v) In Lemma 1 [2] it was shown that $n_\mu\leq n_{\max}$. Let us now consider $n_\mu\geq n_{\min}$. Let $\boldsymbol{\mu}=\mathcal{F}(\mathcal{X})=\operatorname{arg\,min}F(\boldsymbol{\mu})$ and $|\boldsymbol{\mu}|=n_\mu<n_{\min}$. Then, $F(\boldsymbol{\mu})=\sum_{i=1}^{l}\left(\sum_{j=1}^{n_\mu}(x_j^{(i)}-\mu_j)^2+\sum_{j=n_\mu+1}^{n_{x_i}}x_j^2\right)+\sum_{i=1}^{l}n_{x_i}-ln_\mu$. Let us now consider vector $\boldsymbol{\mu}'=(\boldsymbol{\mu},(n_{\min}-n_\mu)*0)$. Now, $|\boldsymbol{\mu}'|=n_{\min}$ and $F(\boldsymbol{\mu}')=\sum_{i=1}^{l}\left(\sum_{j=1}^{n_\mu}(x_j^{(i)}-\mu'_j)^2+\sum_{j=n_\mu+1}^{n_{x_i}}x_j^2\right)+\sum_{i=1}^{l}n_{x_i}-ln_{\min}$. Therefore, $F(\boldsymbol{\mu})-F(\boldsymbol{\mu}')=l(n_{\min}-n_\mu)>0$ and $\boldsymbol{\mu}\neq\operatorname{arg\,min}F(\boldsymbol{\mu})=\mathcal{F}(\mathcal{X})$.

2.2 Genetic algorithm

Another approach to solve the optimization task defined by Eq. (2) is via genetic algorithms [16,17,18], i.e., search heuristics that mimic the process of natural selection. Basically, the genetic algorithm requires a population of candidate solutions, called individuals, to an optimization problem and an objective function associated with each individual that represents the quality of the result. During the computations, population is evolved toward better solutions. In general a genetic algorithm consists of operators that are used to produce a new population of candidate solutions (so called *offsprings*), i.e., *selection* (operator that chooses which solutions are used for crossover), *crossover* (the process of combining selected individuals to obtain a new solution) and *mutation* (a random perturbation of a solution candidate), see Algorithm 2. Investigation carried out in this paper focuses on algorithms to solve clustering problems for which the number of clusters K is known or set up a priori.

Representation of the individuals. When it comes to a clustering task, individuals in the genetic algorithm can be expressed via binary, integer, and real representation [17]. In our investigation we focus on the real encoding, i.e, clusters centers. Classically, in n dimensional space, if individual i encodes K clusters

Data: The initial population P.
evaluation(P);
repeat
  $P :=$ crossover(selection(P));
  mutation(P):
  evaluation(P);
until *termination condition met*;

Algorithm 2: A genetic algorithm.

then its length is Kn, where the first n positions represent the n coordinates of the first cluster prototype, the next n positions represent the coordinates of the second cluster prototype, and so on. However, in our setting the individuals are of possibly different lengths. Therefore, each individual is represented by a set of K vectors encoding K clusters, i.e., the i-th individual is of the form $\{\boldsymbol{\mu}_i^{(1)}, \ldots, \boldsymbol{\mu}_i^{(K)}\}$, for $i = 1, \ldots, N$, where N is the size of population. The initial population is produced as a random sample of given data points.

Selection. Typically, selection is based on the value of the objective function of the solutions. In the proposed algorithm the selection simply chooses M of all the individuals that gives the smallest value of the objective function.

Crossover. In the proposed algorithm two individuals (parents) are selected form the population. The offspring is created as a concatenation of the randomly chosen sub-sequences of each one of them, see Algorithm 3.

Data: Population of clusters centers of size M.
Result: Population P' of size N.
Let $P = \{(\boldsymbol{\mu}_1^{(1)}, \ldots, \boldsymbol{\mu}_1^{(K)}), \ldots, (\boldsymbol{\mu}_M^{(1)}, \ldots, \boldsymbol{\mu}_M^{(K)})\}$;
$P' := \{\}$;
for $i = 1, \ldots, N$ **do**
  $j_1 := \mathcal{U}\{1, \ldots, M\}$; *(random observations)*
  $j_2 := \mathcal{U}\{1, \ldots, M\}$;
  for $j = 1, \ldots, K$ **do**
    $k := (\mathcal{U}\{1, \ldots, K\}, \mathcal{U}\{1, \ldots, K\})$;
    $n1 := \mathcal{U}\{1, \ldots, |\boldsymbol{\mu}_{j_1}^{k_1}|\}$;
    $n2 := \mathcal{U}\{1, \ldots, |\boldsymbol{\mu}_{j_2}^{k_2}|\}$;
    $n3 := |\boldsymbol{\mu}_{j_2}^{k_2}|$;
    $\tilde{\boldsymbol{\mu}}_i^j :=$ sort_decreasing$((\boldsymbol{\mu}_{j_1}^{k_1}[1 : n_1], \boldsymbol{\mu}_{j_1}^{k_2}[n_2 : n_3]))$;
    (we have $\mathbf{x}[i : j] = (x_i, x_{i+1}, \ldots, x_j), i \leq j$*)*
  end
  $P' := P' \cup (\tilde{\boldsymbol{\mu}}_i^{(1)}, \ldots, \tilde{\boldsymbol{\mu}}_i^{(K)})$;
end

Algorithm 3: Crossover procedure for genetic approach for informetric data sets.

Mutation. In our setting the mutation stage consists of three, randomly chosen actions: remove one element of an individual, add one element to an individual and perturb individual with noise generated from a Gaussian distribution, repeated several times (see Algorithm 4).

Data: The population of clusters centers P of size N.
The number of mutations to make LM.
Result: Population P'.
Let $P = \{(\boldsymbol{\mu}_1^{(1)}, \ldots, \boldsymbol{\mu}_1^{(K)}), \ldots, (\boldsymbol{\mu}_N^{(1)}, \ldots, \boldsymbol{\mu}_N^{(K)})\}$;
for $i = 1, \ldots, LM$ **do**
  $u := \mathcal{U}(0, 1)$;
  $j_1 := \mathcal{U}\{1, \ldots, N\}$;
  if $u < 0.33$ **then**
    $j_2 := \mathcal{U}\{1 \ldots, K\}$;
    $\mathbf{z} := \boldsymbol{\mu}_{j_1}^{(j_2)}$;
    $\boldsymbol{\mu}_{j_1}^{(j_2)} := \mathbf{z}[-\mathcal{U}\{1, \ldots, |z|\}]$;
  else
    if $u < 0.67$ **then**
      $j_2 := \mathcal{U}\{1 \ldots, K\}$;
      $\mathbf{z} := \boldsymbol{\mu}_{j_1}^{(j_2)}$;
      $y := \mathcal{U}(0, \mathsf{MAX})$;
      $\boldsymbol{\mu}_{j_1}^{(j_2)} := \mathsf{sort\_decreasing}((\mathbf{z}, y))$;
    else
      $j_2 := \mathcal{U}\{1 \ldots, K\}$;
      $\mathbf{z} := \boldsymbol{\mu}_{j_1}^{(j_2)}$;
      $\boldsymbol{\mu}_{j_1}^{(j_2)} := \mathsf{sort\_decreasing}(\mathbf{z} + \mathcal{N}(|\mathbf{z}|, 0, \sigma))$;
    end
  end
end

Algorithm 4: Mutation procedure for genetic approach for informetric data sets.

Additionally, during the computations, if the convergence is slow, the algorithm is automatically restarted.

Remark 4. Please note that a function $\mathcal{R} : \mathcal{S}^2 \to \mathcal{S}$ given by the procedure of combining two parents into one offspring in the crossover operator (inner loop in Algorithm 3) is also a fusion function. Moreover, it is easily seen that $\mathcal{R}$ is not symmetric, componentwise internal on common indices and length internal. It fulfills, however, global internality.

3 Empirical Analysis

In this section, a comparative analysis of the proposed approach and the projection to a fixed space of indexes is performed. We consider the following sources

of data: Stack Exchange data base, dependency network of R packages, and Elsevier's Scopus citations base.

Stack Exchange data base. Stack Exchange is a network of question-answer sites each devoted to a specific topic, e.g., mathematics, physics, philosophy, etc. In each of such, sites users ask questions concerning a given topic and also give answers to other users questions. Such post (answer, question) is evaluated by the whole community by means of DownVotes (-1) and UpVotes ($+1$). Therefore, each user can be described by the vectors of such evaluations (possibly negative). Moreover, the length of such vectors may vary from user to user. In the investigation carried out here we focused on the users of the Physics Stack Exchange (*physics* data set). In the evaluation process we consider only answers to the questions given by users. The data were collected on the September 15, 2015 and consist of 6470 vectors corresponding to users who answered at least one question. Please note that in this particular data set, about 64% of all vectors are of length 1, and among them about 33% are equal to 0.

R packages dependency network. In R each user may create a package and make it publicly available. Each new software item is built by reusing the functionality provided by packages that are already available. Such dependencies may be viewed as citations and the total number of such citations is overall assessment of the package quality (importance). Because of that, the system of R packages may be perceived as a structure of interrelated items that depend on each other. The collected data consists of information considered 4356 packages (see [19] for the description of the data set). Please note that there are 2928 packages (i.e. 67.2%) which are not cited at all. Moreover, we found 997 items (i.e. 22.43%) that do not cite any other package. Please note that in this particular data set, about 41% of all vectors are of length 1, and among them about 66% are equal to 0.

Elsevier's Scopus citations base. The *scopus* data set consists of 16282 citations vectors gathered from Elsevier's Scopus (see [20] for the description of the data set). For the sake of the clarity of the results presented in this paper, a subset of 3500 randomly chosen authors (*scopus*) has been selected. However, please note that the structure of the data set remains the same. Table 1 presents the sample statistics (minimum, maximum, quantiles and arithmetic mean) of basic characteristics of vectors in each set.

Moreover, please note that about 78% of all vectors in Elsevier's data base are of length 1, and among them about 32% are equal to 0. For the random sample *scopus* the proportions are similar (about 78% of length 1 and among them about 31% equals to 0).

K-means-like algorithm vs projection approach. The aim of the analysis presented in this section is to determine the relationship between partitioning schemes obtained with k-means algorithm on the projection original data into space of fixed number of aggregation indexes and the clustering obtained with modified K-means-like algorithm applied on the raw data points.

Table 1. The comparision of the Elsevier's Scopus data base and random sample *scopus*.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
	Vetors' lengths					
scopus	1	1	1	1.696	1	95
Elsevier's Scopus	1	1	1	1.667	1	129
	Vectors' maximal element					
scopus	0	0	3	9.746	10	791
Elsevier's Scopus	0	0	3	9.098	9	836
	Vetors' sum of elements					
scopus	0	0	3	14.94	11	1610
Elsevier's Scopus	0	0	3	13.53	11	2396

We perform computations with sequences of parameters p and q ranging from 1 to 10 and 0.5 to 2, respectively. For the projection methods we consider the following sets of indexes:

(A) N (vectors' length), Min (vectors' minimal value), Q_1 (first quartile of vectors' elements), NP2 (the number of vectors' elements that are ≤ 0), and NP (the number of vectors' elements greater of 0)
(B) Max (vectors' maximal value), Sum (sum of all vectors' elements), Q_2, Q_3 (second and third quartile of vectors' elements), and arithmetic mean (Mean)
(C) indexes form set (A), (B) and additionally h-index, g-index and w-index when possible, i.e., for data sets with only non-negative values (*rpkg*, *scopus*).

It is clear to see that the set (A) focuses on vectors' lengths, while set (B) on the values of their elements.

Note, however, that such a comparison cannot be performed directly. Not only each algorithm optimizes a differently defined objective functions, but also is situated in different spaces. Therefore, to assess the differences between projection and K-means-like procedure we choose to use Rand Index, i.e., $A/(A+D)$, where A denotes the number of all pairs of data points assigned by both partitions into the same cluster or into different clusters (both partitionings agree for all pairs A) and D denotes the number of all pairs assigned differently by both partitions (the partitions disagree for all pairs D), compare [21]. Moreover, please note that the Rand Index has zero expected value in the case of a random partition, and it is bounded above by 1 in the case of perfect agreement between two partitions.

Table 2 present the maximal Rand Index value for each data set between all considered combinations of parameters. The smallest agreement between the projection and K-means-like approaches usually obtain for scenario (A) then (B) and (C), especially while partitioning into fewer groups. This seems reasonable since this set of indexes does not include specific measure of quality. However,

the obtain results indicate that with appropriate choice of parameters p and q, the K-means-like algorithm produce clustering quite similar (about 80%) to the one obtained with the projection approach (especially for (B) and (C) set of indexes).

Table 2. The maximal value of the Rand Index between clustering obtained via K-means-like algorithm with $d_{D;p,q}$ dissimilarity measure with various parameters p and q and via the k-means algorithm on projection to a space of indexes.

	physics			*rpkg*			*scopus*		
K	(A)	(B)	(C)	(A)	(B)	(C)	(A)	(B)	(C)
3	0.65	0.71	0.44	0.76	0.75	0.80	0.54	0.91	0.84
6	0.72	0.82	0.83	0.74	0.94	0.95	0.68	0.74	0.78
9	0.24	0.86	0.92	0.78	0.92	0.93	0.70	0.94	0.85

K-means-like vs genetic approach. Let us now consider the comparison between the K-means-like procedure and the proposed genetic algorithm. Firstly, both procedures were applied on random samples of each considered data sets. The Table 3 presents the percentage of all cases in which genetic algorithm performed better in case of partitioning into 3, 6, and 9 clusters. In most cases, between 84% for *physics* data set and 60% for *scopus* data set, the K-means-like algorithm performed better in case of detecting three clusters. However, for the six and nine clusters, genetic approach gives better results (between 68% for *physics* and 100% for *rpkg*).

Table 3. The percentage of cases when GA performed better than K-means-like procedure, i.e., the returned value of the objective function was smaller. The mean difference between GA and KMA is also presented in brackets, where the first value concerns cases when GA performed better and the second one when KMA performed better.

Set \ K	3	6	9
scopus	0.40 (3227.9 | 599.3)	0.96 (5673.1 | 946.3)	0.84 (6116.2 | 526.4)
rpkg	0.36 (8829.2 | 36154.9)	0.80 (93120.7 | 5158.3)	1.00 (99950.7 | $\emptyset$)
physics	0.16 (51851 | 1410.6)	0.68 (2837.7 | 662.3)	0.84 (2829.6 | 790.2)

Let us now focus on exemplary results obtained via K-means-like and genetic approaches. Table 4 presents the value of the objective function for clustering obtain via K-mean-like (denoted as KMA) procedure and genetic approach (denoted as GA) for *physics*, *scopus* and *rpkg* data sets, and the Rand Index calculated between those two partitionings. In each case the GA performed better with agreement about 60% (with exception for *rpkg* data set with about 92% of

agreement). Table 5 presents cluster sizes for both algorithms. Note that even though results are similar (one large cluster and few smaller), it seems that the genetic approach tends to create smaller groups, then KMA, e.g., cluster of one element in *physics* data set. This is also reflected in Table 6 with the total inner dissimilarity in each cluster.

Table 4. Values of the objective function for clustering obtained via K-means-like (denoted as KMA) procedure and genetic approach (denoted as GA), and the Rand Index (denoted as RI).

	physics	*scopus*	*rpkg*
p, q	7, 1	2, 1	1, 1
KMA	610702.5	567252.7	546948.6
GA	585719.2	503527.2	646174.7
RI	0.594171	0.6192729	0.9212828

Table 5. Sizes of clusters obtained via KMA and GA.

Cluster no.	1	2	3	4	5	6
	physics					
KMA	5081	665	538	7	159	20
GA	5757	1	474	164	67	7
	scopus					
KMA	2581	678	188	37	8	8
GA	2932	487	4	67	4	6
	rpkg					
KMA	1240	94	54	27	5	8
GA	1259	98	58	5	3	5

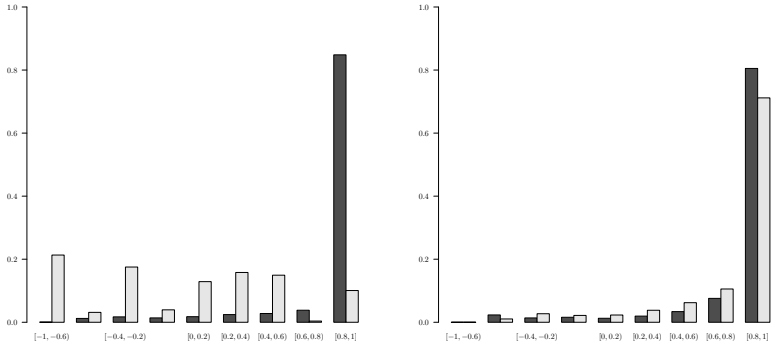
Let us consider the Silhouette information proposed in [22]. The Silhouette information for each observation $\mathbf{x}^{(i)}$ is defined as:

$$s(i) = \frac{(b(i) - a(i))}{\max(a(i), b(i))},$$

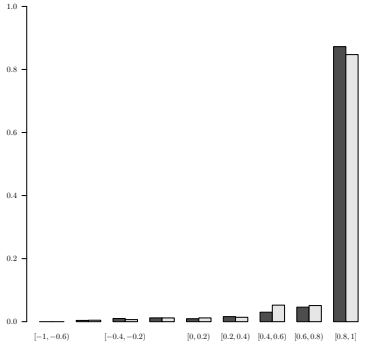
where $a(i) = \frac{1}{|C_k|} \sum_{\mathbf{y} \in C_k} d_D(\mathbf{y}, \mathbf{x}^{(i)})$ if $\mathbf{x}^{(i)} \in C_k$, and $b(i) := \min_{j=1,\ldots,K; j \neq k} d(i, C_j)$, where $d(i, C_j) = \frac{1}{|C_j|} \sum_{\mathbf{y} \in C_j} d_D(\mathbf{y}, \mathbf{x}^{(i)})$ for all C_j such that $j \neq k$. Please note that $a(i)$ is the average dissimilarity between $\mathbf{x}^{(i)}$ and all other points of the cluster to which i belongs (if i is the only observation in its cluster, $s(i) = 0$ without further calculations) and $d(i, c_j)$ is an average dissimilarity of $\mathbf{x}^{(i)}$ to all observations of C_j that $\mathbf{x}^{(i)}$ does not belong to. Moreover, $b(i)$ can be seen as the dissimilarity between i and its "neighbor" cluster, i.e., the nearest one to

which it does not belong. Observations with a large $s(i)$ (almost 1) are very well clustered, a small $s(i)$ (around 0) means that the observation lies between two clusters, and observations with a negative $s(i)$ are probably placed in the wrong cluster.

Fig. 1 depicts the percentage of observations with $s(i)$ within $[-1, -0.6)$, $[-0.6, -0.4)$, $[-0.4, -0.2)$, $[-0.2, 0)$, $[0, 0.2)$, $[0.2, 0.4)$, $[0.4, 0.6)$, $[0.6, 0.8)$, $[0.8, 1]$. Please note that in cases of the *physics* data sets the results for the genetic algorithm are better than for K-means-like algorithm, while for the *scopus* and *rpkg* are about the same.



(a) *physics* (b) *scopus*



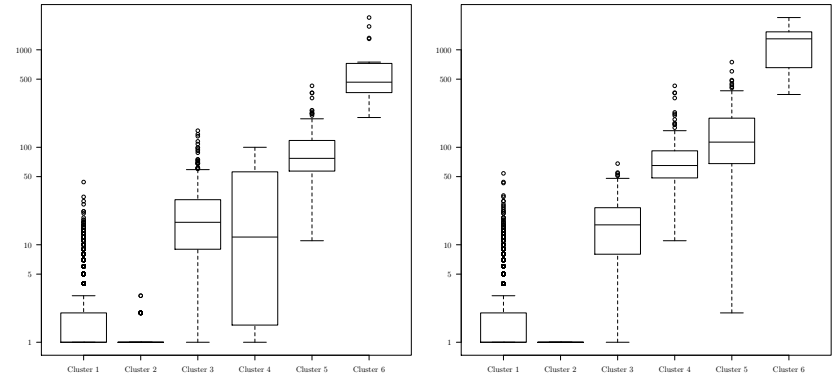
(c) *rpkg*

Fig. 1. The bar plots of the Silhouette width for KMA-partitioning and GA-partitioning, depicted in light and dark gray respectively.

Table 6. Total inner clusters dissimilarity for clustering obtain via the K-means-like (denoted as KMA) procedure and the genetic approach (denoted as GA).

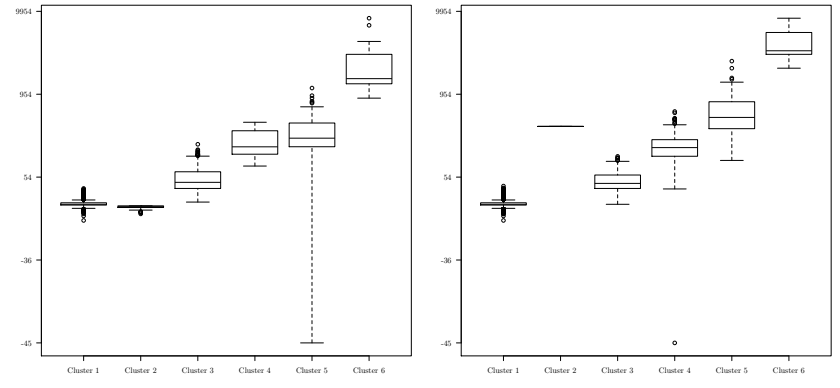
Cluster no.	1	2	3	4	5	6
	physics					
KMA	77455.62	1415.78	133645.88	57756.86	165782.01	174646.30
GA	91168.00	4764.41	123587.82	91364.88	198037.46	76796.58
	scopus					
KMA	20533.37	44994.50	107017.54	74370.41	262758.88	57578.00
GA	56029.00	135444.09	71226.79	151434.95	45038.46	44353.86
	rpkg					
KMA	28827.21	34890.90	84307.37	69966.89	140740.00	287442.38
GA	41313.94	74210.71	170839.21	81404.04	24833.33	154347.40

Fig. 2 depicts exemplary box-and-whisker plots of vectors' basic sample characteristics in each cluster obtained with, both, KMA and GA algorithms for *physics* data set. Please note the logarithmic scale on Y axis. On the other hand, Figs. 3 and 4, present step plots of vectors in each cluster *physics* for KMA and GA partitioning, respectively, (depicted in gray color) with their centroids and centers (depicted in black). The centers obtained with the GA are more differential according to, both, lengths (1, 8, 18, 71, 106, 1222) and total sum of elements (1, 324.14, 31.93, 192.3, 449.27, 4236.88), then K-means-like derived centroids (1, 1, 18, 14, 79, 489) and (1.92, -1.98, 47.77, 198.86, 285.02, 1997.75), respectively.



(a) Vectors' lengths (KMA-partitioning).

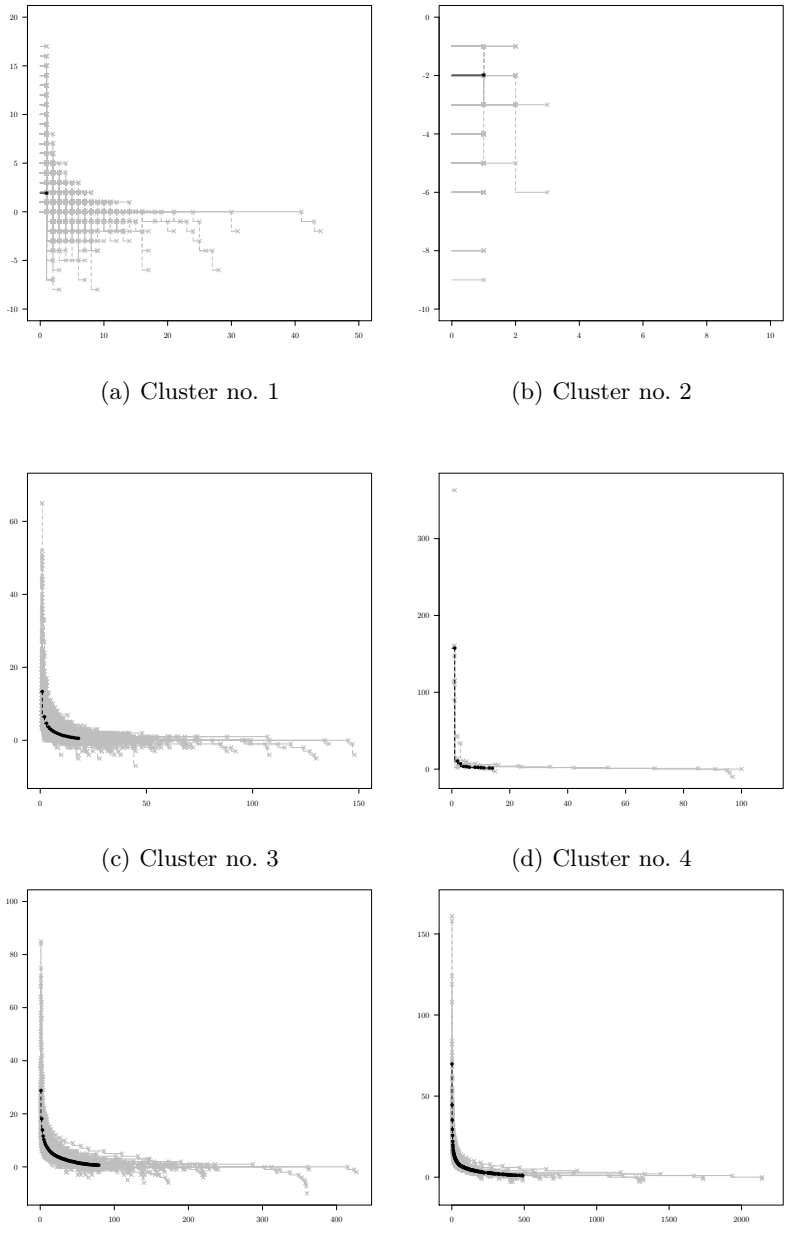
(b) Vectors' lengths (GA-partitioning).



(c) The total sum of vectors' elements (KMA–partitioning).

(d) The total sum of vectors' elements (GA–partitioning)

Fig. 2. The box-and-whisker plots of vectors' basic sample characteristics – *physics* data set.



(a) Cluster no. 1

(b) Cluster no. 2

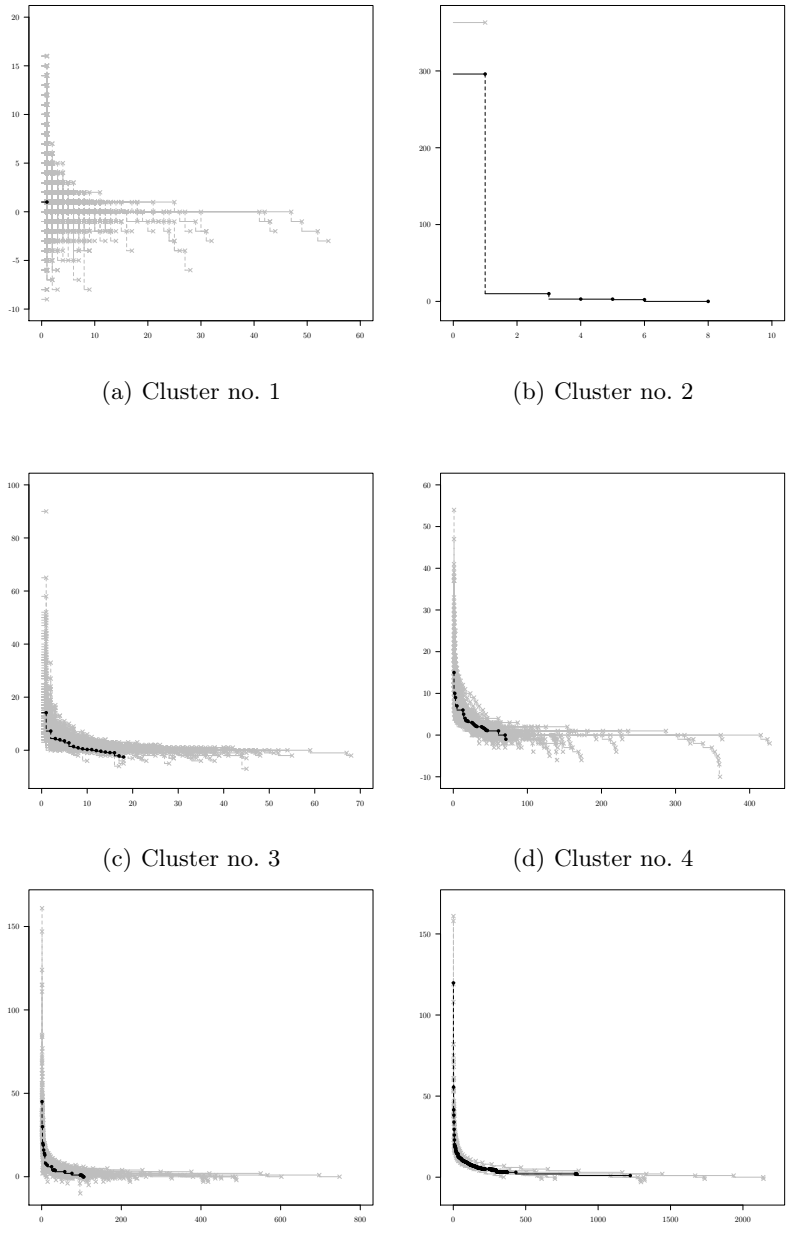
(c) Cluster no. 3

(d) Cluster no. 4

(e) Cluster no. 5

(f) Cluster no. 6

Fig. 3. Step plots of vectors in each cluster *physics* for KMA partitioning (depicted in grey color) and their centroids (depicted in black).



(a) Cluster no. 1

(b) Cluster no. 2

(c) Cluster no. 3

(d) Cluster no. 4

(e) Cluster no. 5

(f) Cluster no. 6

Fig. 4. Step plots of vectors in each cluster *physics* for GA partitioning (depicted in grey color) and their centroids (depicted in black).

4 Conclusions

This contribution presents the recent developments on clustering algorithms design to deal with so numeric strings. The genetic approach for such setting in proposed and compare to K-means-like algorithm. Both, the data mining and aggregation perspective is taken into account while reviewing obtained results. Moreover, reduction of the data dimension by considering a fixed number of attributes or indicators in order to apply clustering techniques on vectors of nonconforming lengths is also investigated. It turns out that agreement between this approach and algorithms applied directly on data points depends on parameters of the dissimilarity function. Moreover, with appropriate choice of parameters K-means-like procedure can mimic the results of projection approach. Therefore, some interesting directions worth of deeper investigation arise. First of all, the other penalty terms in dissimilarity measures should be considered and tested so the procedure may be better calibrated to suit the nature of an input data set we analyze. Also, calibration of the parameters of a dissimilarity measure, so that produced values are close as much as possible to the one obtain with an arbitrary set of indexes may be considered.

Acknowledgments

This study was partially supported by the National Science Center, Poland, research project 2014/13/D/HS4/01700.

Anna Cena would like to acknowledge the support by the European Union from resources of the European Social Fund, Project PO KL "Information technologies: Research and their interdisciplinary applications", agreement UDA-POKL.04.01.01-00-051/10-00 via the Interdisciplinary PhD Studies Program.

References

1. Gągolewski, M.: Data fusion: Theory, methods and applications, Institute of Computer Science, Polish Academy of Sciences, Warsaw (2015)
2. Cena, A., Gągolewski, M.: *A K-means-like algorithm for informetric data clustering*, In Alonso, J., Bustince, H., Reformat, M., eds.: Proc. IFSA/Eusflat 2015, Atlantic Press, 536–543, 2015
3. Grabisch, M., Marichal, J.L., Mesiar, R., Pap, E.: *Aggregation functions,* Cambridge University Press (2009)
4. Hirsch, J.E.: *An index to quantify individual's scientific research output*, Proceedings of the National Academy of Sciences, 102, 46, 16569–16572, 2005
5. Egghe, L.: *An improvement of the h-index: the g-index*, ISSI Newsletter, 2, 1, 8–9, 2006
6. Ortega, J.L., López-Romero, E., Fernández, I.: *Multivariate approach to classify research institutes according to their outputs: The case of the CSIC's institutes*, Journal of Informetrics, 5, 323–332, 2011
7. Cheng, Y., Liu, N.C.: *A first approach to the classification of the top 500 world universities by their disciplinary characteristics using scientometrics*, Scientometrics, 68, 1, 135–150, 2006

8. Cena, A., Gągolewski, M., Mesiar, R.: *Problems and challenges of information resources producers' clustering*, Journal of Informetrics, 9, 2, 273–284, 2015
9. Cena, A., Gągolewski, M.: *Aggregation and soft clustering of informetric data*, In Baczynski, M., De Baets, B., Mesiar, R., eds.: Proc. 8th International Summer School on Aggregation Operators (AGOP 2015), Katowice, Poland, University of Silesia, 79–84, 2015
10. Bustince, H., Fernandez, J., Kolesárová, A., Mesiar, R.: *Fusion functions and directional monotonicity*, Communications in Computer and Information Science, 444, 262–268, 2014
11. Bustince, H., Fernandez, J., Kolesárová, A., Mesiar, R.: *Directional monotonicity of fusion functions*, European Journal of Operational Research, 244, 1, 300–308, 2015
12. Xu, R., Wunsch II, D.C.: Clustering, Wiley-IEEE Press (2009)
13. Levenshtein, V.I.: *Binary codes capable of correcting deletions, insertions, or reversals*, Soviet Physics Doklady, 10, 8, 707–710, 1966
14. M.R. Garey, D.S. Johnson, H.W.: *The complexity of the generalized Lloyd-Max problem*, IEEE Transactions on Information Theory, IT-28, 2, 255–256, 1982
15. MacQueen, J.B.: *Some methods for classification and analysis of multivariate observations*, In: Proc. Fifth Berkeley Symp. on Math. Statist. and Prob.,1, Berkeley, University of California Press, 281–297, 1967
16. Maulik, U., Bandyopadhyay, S.: *Genetic algorithm-based clustering technique*, Pattern Recognition, 33, 1455–1465, 2000
17. Hruschka, E., Campello, R., Freitas, A., de Carvalho, A.: *A survey of evolutionary algorithms for clustering*, Trans. Sys. Man Cyber Part C, 39, 2, 133–155, 2009
18. Nanda, S.J., Panda, G.: *A survey on nature inspired metaheuristic algorithms for partional clustering*, Swarm and Evolutionary Computation, 16, 1–18, 2014
19. Cena A., Gągolewski M., Tartanus B., Żogała–Siudem B.: *Current State of R Packages*, SRI PAS Research Report RB/2/2013 (2013)
20. Gągolewski, M.: *Bibliometric impact assessment with R and the CITAN package*, Journal of Informetrics, 5, 4, 678–692, 2011
21. Lawrence, H., Phipps, A.: *Comparing partitions*, Journal of Classification, 2, 193–218, 1985
22. Rousseeuw, P.J.: *Silhouettes: A graphical aid to the interpretation and validation of cluster analysis*, Journal of Computational and Applied Mathematics, 20, 53–65, 1987

Review on Sensitivity Analysis in Biochemical Models

Agata Charzyńska

Institute of Computer Science, Polish Academy of Sciences
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland

Abstract. In this work we focus on sensitivity analysis of biochemical models. The research includes both formal modelling of signalling and metabolic pathways as well as developing theoretical methods for model assessment. The aim of this study is to better understand a natural phenomenon in quantitative an qualitative manner by mathematical modelling. To validate any model it is crucial to determine which factors are most influential for a modelled system behaviour. Part of my study is to develop a new sensitivity analysis method based on mutual information that provides an efficient identification of parameters and group of parameters that are crucial for a modelled system, providing additionally information about interactions between parameters in accordance to the model output.

In the first section of this paper we briefly present the motivation behind formal modelling and its necessity in any experimental design.

The second section contains review of classical sensitivity analysis methods based on literature and in the second part of this section we recall two recently invented SA methods: Stochastic Noise Decomposition (SND) and Sensitivity Analysis (SA) based on Mutual Information (MI). We tested and implemented the SND method in direct cooperation with authors of this method (cf. Komorowski et al., 2013) the results of our work were presented in application note - StochDecomp Matlab package (Jetka et al., 2014). The second method SA based on MI was deeply studied and developed by us with the application to continuous random variables. We introduce a novel correction to the classical k-nn entropy estimator to reduce the bias of estimation in finite sample size for highly dimensional data.

The third section is devoted to a brief summary of biochemical models of our interest. Some models were adopted from literature and used as a test example for application of theoretical SA methods e.g. p53-Mdm2 negative feedback loop model and other models were fully developed and implemented by us e.g. sphingolipid metabolism model (Wronowska et al., 2015). To all presented in this section models we applied several SA methods.

1 Motivation

Mathematical modelling of biological phenomena described e.g. by dynamical systems, complements experimental technologies used to identify and comprehend a role of system components. The process in which a model is formulated and refined helps to articulate hypotheses and thereby supports the design of experiments to validate these hypotheses and the model itself. Once the model is validated it is used to speculate about mechanisms underlying cell functions.

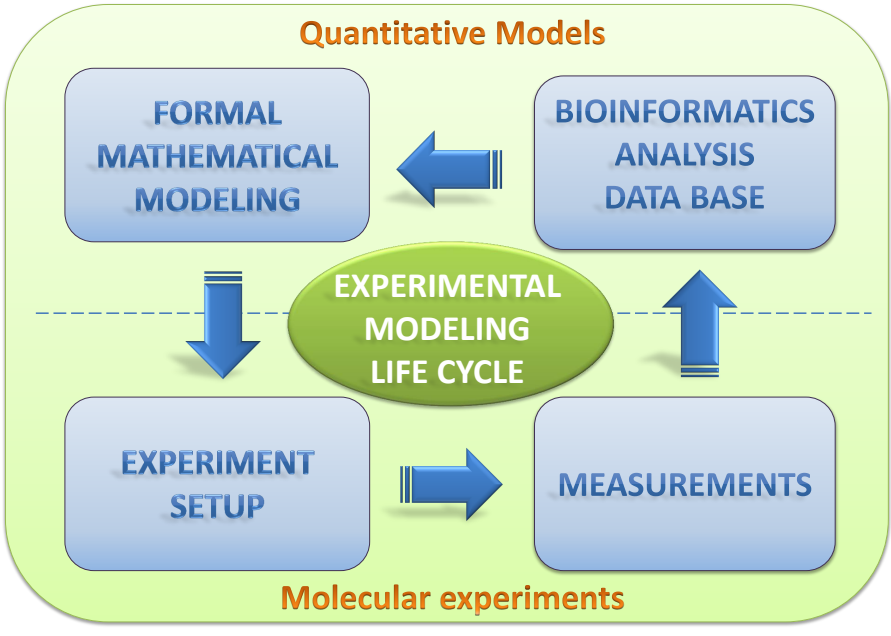


Fig. 1. Scheme of experimental modelling life cycle

The standard approach to understand dynamics of biological system is to observe the behaviour of as many as possible system components. An important element of the model design is analysis and identification of most informative and responsive to perturbations model elements (sensitivity analysis) to reveal the spectrum of available dynamical regimes (validated by model checking). Verification of the model design together with parameters estimates is carried out experimentally by comparing model predictions with experimental results for stimulation profiles. Any necessary corrections in model structure and parameter estimates must be made. The experimental modelling life cycle scheme is depicted in Fig. 1.

The construction and analysis of mechanistic models of biological systems is a part of recently established, highly interdisciplinary fields of systems and computational biology. Computational modelling of signal transduction integrates available knowledge about pathway regulation, and the general chemical and physical principles with experimental data from different biotechnology platforms. Such approach constitutes a powerful solution for formalizing and extending traditional molecular and cellular biology.

2 Sensitivity Analysis

Mathematical modelling of biological phenomena can be carried out in a deterministic, stochastic or hybrid manner. The first approach is based on the Ordinary Differential Equations (ODEs), while the second one is based on the stochastic processes or stochastic differential equations (SDEs) theory. Both types of models are usually based on some simplifying assumptions i.e. that the temperature of a chemical environment is constant, and that the diffusion process occurs immediately, which ensures an even distribution of a substance over a limited volume. Deterministic models describe changes in mean concentrations of reagents (species) over time, and they do not include the effect of fluctuations which occur in reality. This means that for given initial conditions, a deterministic model will always provide the same results. While stochastic models describe the evolution of the probability distribution of all possible system states with respect to time. Both types of modelling requires proper verification and analysis.

A biochemical model described by ODEs can be expressed in the matrix form:

$$\frac{d\mathbf{S}(t)}{dt} = M\mathbf{v}(\mathbf{S}(t)),$$

where the system state is represented by the time dependent state vector $\mathbf{S}(t)$ of species concentration, M denotes the stoichiometry matrix and $\mathbf{v}(\mathbf{S}(t))$ denotes a vector of reaction fluxes (in simplest standard modelling according to Mass Action Law (MAL) or Michaelis Menten (MM) kinetics possibly including inhibition rates).

The most popular approach to describe discrete stochastic model of biochemical pathway is Chemical Master Equation (Chapman-Kolmogorov equation of Markov chain modelling the evolution of the system):

$$\frac{pP(\mathbf{x},t)}{dt} = \sum_j a_j(\mathbf{x}-\mathbf{m}_j)P(\mathbf{x}-\mathbf{m}_j,t) - \sum_j a_j(\mathbf{x})P(\mathbf{x},t),$$

where the system state is denoted by the vector $\mathbf{X}(t) \in \mathbb{N}^N$ of numbers of molecules each row for one of N reacting species, $\mathbf{m}_j$ denotes the j-th column of stoichiometry matrix $M = (\mathbf{m}_1, \ldots, \mathbf{m}_R)$ and $P(\mathbf{x},t)$ denotes the time- and state-dependent distribution of system being in state $\mathbf{X}(t) = \mathbf{x}$ and finally $a_j(\mathbf{X}(t))$ denotes the propensity function associated with the j-th reaction (Charzyńska et al., 2012).

2.1 Classification of SA Methods

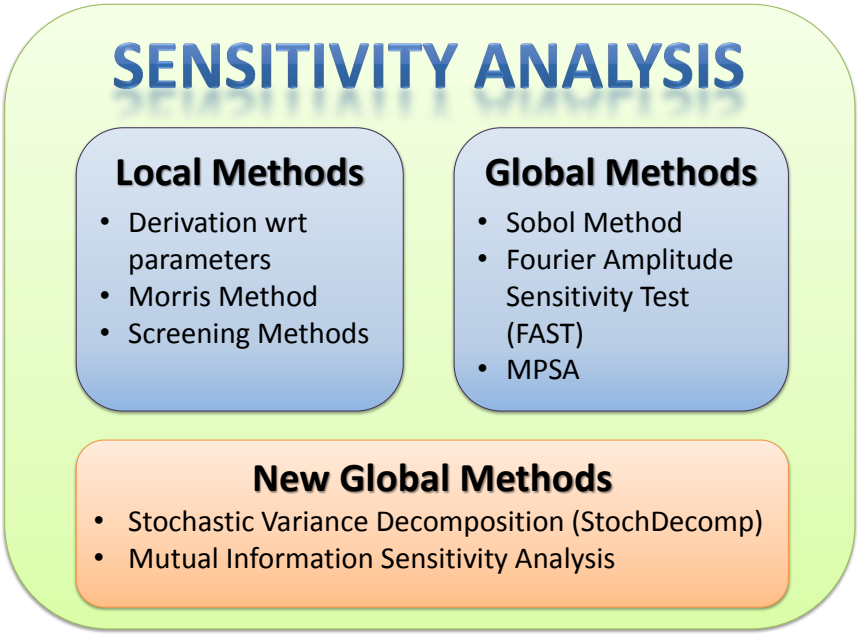


Fig. 2. Local and Global Sensitivity Analysis Methods

Sensitivity analysis is used to determine dependencies between input parameters and the results of the model. One can chose as input parameters for example initial concentrations of modelled species or reaction rates. Result of the biochemical model is most commonly defined as the density of species as a function of time. SA is very useful in mathematical modelling, as it describes dependencies between different elements of the model, it is also applicable to empirical experiments planning and enables verification of theoretical model results together with numerical and empirical results. SA also enables recognition of model's conceptual and implementational omissions.

Sensitivity analysis investigates the relations between uncertain parameters of a model, and a property of the observable outcome, which represents some prototypic features of the modelled system (Saltelli et al., 2008) . SA has been used in various parametrization tasks for models of biological systems, such as finding essential and insignificant parameters for the prioritization (Yue et al., 2008) , identifying parameters interactions or or parameters clustering (Mahdavi et al., 2007).

Classically, sensitivity of the model to the parameters is determined by the partial derivation of the outcome variables with respect to parameters. SA methods based on such quantities are called local (LSA), as the derivative is taken at a fixed point in the state space. Sensitivity indices are defined as partial derivatives of system states with respect to parameters integrated by time:

$$s_{n,i} = \int_0^T \left| \frac{\partial S_n(t)}{\partial \theta_i} \right|_{\theta=\theta_0} dt$$

where S_n are different species concentrations, θ is the vector of parameters and θ_0 is some fixed point in parameters space. One of disadvantage of this method is the high dependence of sensitivity indices to arbitrary choice of time horizon T that can influence the SA results.

Moreover, these methods belong to the class of one-factor-at-time (OAT) methods, because the net effect of a parameter to the model outcome is taken while assuming that all other factors are fixed. However, most of the biochemical reactions networks yield models of a non-linear nature and for these models, OAT methods can be of limited use if not outright misleading (Saltelli et al., 2005) . Possible solution is to ingestive of the influence of simultaneous changes in parameters values by assessing higher order partial derivatives (Mahdavi et al., 2007), where the order depends on the non-linearity level of the model. Nevertheless, it is still a local method, highly dependent on the given values of parameters.

On the other hand, there are so-called global sensitivity analysis (GSA) methods, that simultaneously examine a whole range of input parameters values. Exemplary implementations of the GSA indices are the model-free, global sensitivity measures such as the variance decomposition (Saltelli et al., 2008), or the parameters space mapping method of Monte Carlo filtering (MCF) such as the multi-parameter sensitivity analysis (MPSA) (Hornberger and Spear, 1981).

In between, there are screening techniques which approximate the GSA indices. Screening techniques, such as the weighted average of local sensitivities (Bentele et al., 2004) or the elementary effects of Morris (1991), are global in the sense that they scan a whole range of parameters values, but they use local OAT methods for each analysed set of parameter values.

For a sake of clarity, if not explicitly stated otherwise, we will use a term local method meaning the local and OAT method, as well as a term global method meaning GSA method (Global and simultaneous).

Finally, there are SA methods tailored specifically to the stochastic models (Gunawan et al., 2005). These methods recognize that the response is in form of distribution rather than a single value corresponding, for instance, to the mean value. Consequently, for systems where a parameter disruption does not significantly influence the mean but significantly influences the distribution itself, the model-free SA indices can incorrectly indicate a lack of sensitivity of the model (cf. Degasperi and Gilmore, 2008).

To extend the range of available global sensitivity analysis methods we recall here new approaches: method based on information theoretic measure and

stochastic noise decomposition. Both methods can be applied to dynamical systems whether formulated in deterministic or stochastic manner. Each method has its specificity: noise decomposition method allows to track how the stochastic noise distribute within the biochemical system in division into single reactions noise compartments, whereas mutual information method based on entropy estimation provides sensitivity indices and interaction indices for any group of parameters that represent model input.

2.2 Stochastic Noise Decomposition Method

The question which molecular species or parts of a network contribute most to the variability of a system or are responsible for most of the information loss has attracted much attention in recent years. Stochasticity is an indispensable aspect of biochemical processes not only but especially at the cellular level. Studies on how the noise enters and propagates in biochemical systems provides a non-trivial insights into the origins of noise in a model. Numerous studies focus on analysis of noise in signalling networks in detail and decomposition of the noise into contributions attributable to fluctuations in species concentration.

Recently developed StochDecomp (Jetka et al., 2014) is a flexible and widely applicable noise decomposition tool that allows to calculate contributions of individual reactions to the total variability of a system output. The method allows to quantify how the noise enters and propagates in biochemical systems. It is based on recently developed method (Komorowski et al., 2013) that allows to analyse how the structure of biochemical networks gives rise to noise in its outputs. In principle, this allows to efficiently calculate the contribution each reaction makes to the variability in all concentrations for any network, which can be modelled within the Linear Noise Approximation (LNA) framework. LNA is one of the possible simplification of the Chemical Master Equation , with the system dynamic modelled as Poisson process:

$$\mathbf{X}(t) = \mathbf{X}(0) + \sum_{j=1}^{R} \mathbf{m}_j N_j \left(\int_0^t f_j(\mathbf{X}(\tau), \tau) d\tau \right)$$

where $N_j(\mathbf{X}(t), t)$ denotes Poisson process dependent on time and a system state $\mathbf{X}(t)$, corresponding to occurrence of j-th reaction. The probability that j-th reaction occur during the time interval $[t; t + dt)$ equals $f_j(x, t)dt$, where the $f_j(x, t)$ is called the transition rate.

It is more efficient to transit from discrete to continuous process, as accurate discrete models describe the exact evolution of probability distribution of the system state counted in molecules number. Discrete biochemical models are computationally not efficient, as simulations require significant resources. Consequently by use of deterministic approximation:

$$\Phi(t) = \Phi(0) + \sum_{j=1}^{R} m_j \int_0^t f_j(\Phi(s), s) ds$$

where $\Phi(t)$ is the mean system state being the solution of the ODEs, one can describe the system state evolution by dividing it into deterministic and stochastic part:

$$x(t) = \xi(t) + \Phi(t)$$

where $\Phi(t)$ is the deterministic part and $\xi(t)$ is the Weiner process describing stochastic noise of a system state (Komorowski et al. 2009). The next step of stochastic noise decomposition is to divided noise linearly into noise steaming from separate reactions. The total variance:

$$\Sigma(t) = \langle (x(t) - \langle x(t) \rangle)(x(t) - \langle x(t) \rangle)^T \rangle$$

is described by the differential equation

$$\frac{d\Sigma}{dt} = A(t)\Sigma + \Sigma A(t)^T + D(t), \tag{1}$$

where

$$\{A(\Phi, t)\}_{ik} = \sum_{j=1}^{r} m_{ij} \frac{\partial f_j(\Phi, t)}{\partial \Phi_k}$$

and $D(t)$ denotes diffusion matrix. The fact, that the variance can be represented as the sum of individual contributions,

$$\Sigma(t) = \Sigma^{(1)}(t) + \ \ldots \ + \Sigma^{(r)}(t). \tag{2}$$

results directly from the decomposition of the diffusion matrix $D(t) = \sum_{j=1}^{r} D^{(j)}(t)$ and the linearity of the equation for $\Sigma(t)$.

The Stochastic Noise Decomposition method is based on the LNA, which is assumed to provide a reasonable representation of analysed systems even in case of priori deterministic formulation. Origins of variability can be therefore assigned to individual reactions and arbitrarily defined network components.

Contrary to most available methods Stochastic Noise Decomposition is tailored for biochemical dynamical systems and provides an insight in time evolution of noise decomposition into reaction network. The tool is computationally effective even for vast biochemical models (compare Fig. 11) and can successfully provide a required information, see Fig. 3.

2.3 Sensitivity Analysis Based on Mutual Information

Another recently developed method for sensitivity analysis of multi-variables system has been oryginaly proposed by Lüdtke (Lüdtke et al. 2008). One of the biggest advantage of this method is its applicability to investigation of a model sensitivity to groups of parameters, and not only to single parameters, so it is not OAT method. Moreover this method provides an insight into interactions between parameters.

However, the approach proposed by Lüdtke is based on discrete variables and discrete entropy estimator. Consequently it requires computationally inefficient

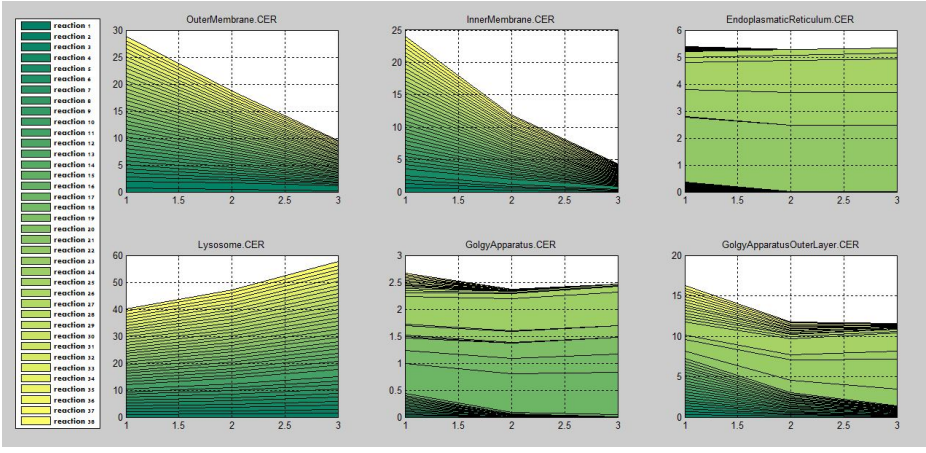


Fig. 3. Stochastic Noise Decomposition into single reactions vs. time for the Ceramide Metabolism Model - Scheme in Fig. 11

variable discretization procedure, that is highly biased and inefficient in high-dimensional space (i.e. in many parameters case). Having this concern in mind and due to fact that biochemical models deal with continuous measurements we propose to amend the method to continuous variables case.

The fundamental concept for sensitivity indices is mutual information Eq. (3) between random variables defined by parameters and random variables being model output. Let us denote by variable $X \sim g(x)$ model parameters and by variable $Y \sim f(y)$ model output, then the mutual information between this continuous random variables is defined by

$$I(X;Y) := \int_X \int_Y \log \frac{h(x,y)}{g(x)f(y)} h(x,y)dydx \tag{3}$$
$$= \mathbb{E}\left[\log \frac{h(x,y)}{g(x)f(y)}\right] = H(Y) + H(X) - H(X,Y),$$

where $g(x)$ and $f(y)$ denotes probabilities densities functions and $h(x,y)$ is the joint probability density function of joint random variable (X,Y).

Measurements of MI is based on entropy estimation. In our approach to SA based on MI we use differential entropy Eq. (4), so there is no need of variables discretization.

$$H(X) := \int_X -\log g(x)dx = \mathbb{E}\left[-\log g(x)\right] \tag{4}$$

As a starting point for differential entropy estimation we used the k-th nearest neighbour entropy estimator. In order to achieve more reliable results we introduced more efficient k-nn differential entropy estimator for multivariate random variables. We have noticed the biased behaviour of the k-nn entropy estimator

Eq. in higher dimension and propose bias correction, which yields more accurate k-nn entropy estimates especially in higher dimensions. Our improved k-nn entropy estimator explore the idea of correcting the density function evaluation near to the boundary of random variable support.

Definition 1 *Assume that X_i are the parameters of the model and Y is the model output, then* ***sensitivity indices*** *are defined as:*

$$I(X_i;Y) = H(Y) + H(X_i) - H(X_i,Y) = H(Y) - H(Y|X_i).$$

Analogously, ***sensitivity indices for pairs of parameters*** *are defined as:*

$$I(X_i,X_j;Y) = H(Y) + H(X_i,X_j) - H(Y,X_i,X_j) = H(Y) - H(Y|X_i,X_j).$$

The sensitivity indices reflect the impact of parameters on the model output, in other words this definition indicates correlations between parameters and the output. Definition 1 can be extended for any subset of parameters.

The group sensitivity index for a pair of parameters may have high value indicating the significant influence of these parameters to the model output, while two sensitivity indices for these two single parameters may in the same time have low value. We interpret such case as opposite -negative interaction between this pair of parameters, compare Fig. 4.

Definition 2 *Let X_i denote parameters of a model and Y denote model output, then* ***interactions indices within pair of parameters*** *are defined by:*

$$I(X_i;X_j;Y) = \mathbb{E}_{X_i,X_j,Y}\left[-\log\frac{p(x_i)p(x_j)p(y)p(x_i,x_j,y)}{p(x_i,x_j)p(x_i,y)p(x_j,y)}\right]$$
$$= H(X_i) + H(X_j) + H(Y) - H(X_i,Y) - H(X_j,Y) - H(X_i,X_j) + H(X_i,X_j,Y)$$
$$= I(X_i;Y) + I(X_j;Y) - I(X_i,X_j;Y).$$

3 Biochemical Models

Within our research we concentrate on investigation and development of formal sensitivity analysis methods and also we implement and test the methods on various dynamical biochemical models. The complexity of a model depends on the number of variables and parameters and the kinetics defined in ODEs or SDEs, compare Fig. 5. In order to better understand and capture model features we test different standard and novel approaches.

3.1 Ligand-induced receptor model

In paper (Charzyńska et al., 2012) we focus on recently available methods of sensitivity analysis for dynamic biochemical models, such as local sensitivity analysis based on derivatives with regards to single parameters, and global sensitivity

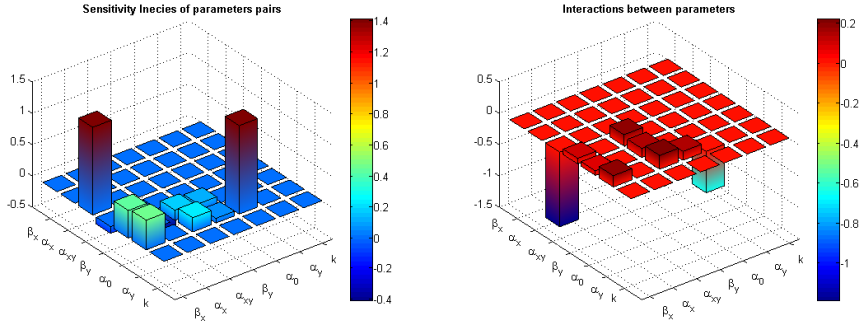


Fig. 4. Sensitivity analysis based on MI for the p53-Mdm2 negative feedback loop model - scheme of the model in Fig. 8. (Submitted to Entropy)

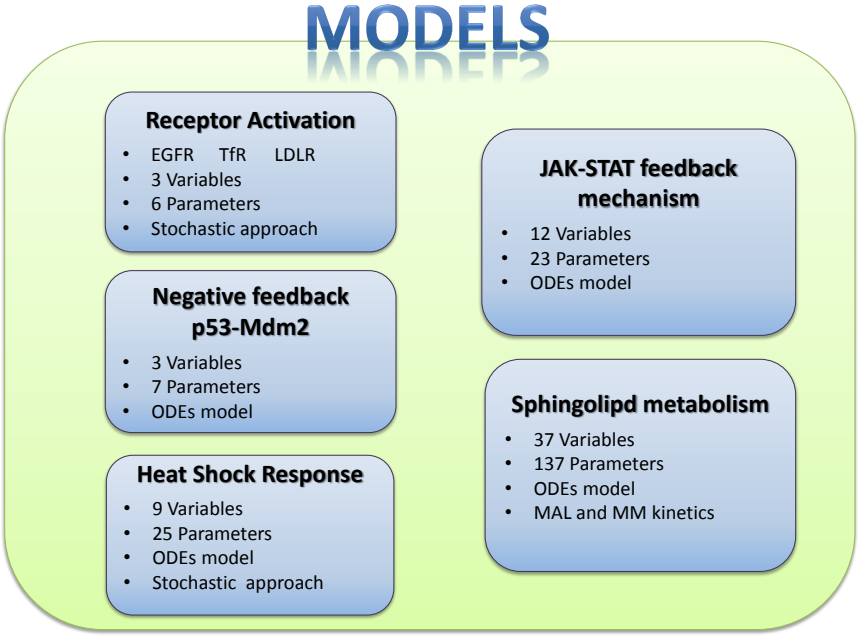


Fig. 5. Examined biochemical models

analysis methods i.e. variance decomposition, Fourier Amplitude Sensitivity Test (FAST), screening methods or stochastic methods. As an illustrative example of the presented ideas we consider the mathematical model of ligand-induced receptor system (Shankaran et al. 2007), see Fig. 6.

We transformed the classical deterministic version into a stochastic model. For both approaches, appropriate SA were applied. The model reflects a system of cell surface receptors in a single cell and describes the time evolution of three different species: ligands in the inter-cellular space, free receptors on a cell membrane and ligand-receptor complexes, see Fig. 6. The set of the model parameters contain also the volume of the inter-cellular space that falls for a single cell V and the level of receptors concentration in the steady state R_T. We investigate four types of receptors:

- epidermal growth factor receptor, (EGFR), which stimulates cell division and plays an important role in the process of tumour formation,
- transferrin receptor (TfR), responsible for the transport of iron into cells,
- low-density lipoprotein receptor (LDLR), transporting cholesterol into cells,
- vitellogenin receptor (VtgR), which mediates the uptake of vitellogenin (Vtg) in oocyte development.

The results for sensitivity analysis based on Morris method were presented in Fig. 7. In three of four analysed receptors types the crucial parameters were k_{off} and k_{on} corresponding respectively to rate of complexes disintegration and complexes binding, as well R_T corresponding to concentration of receptor in stationary state.

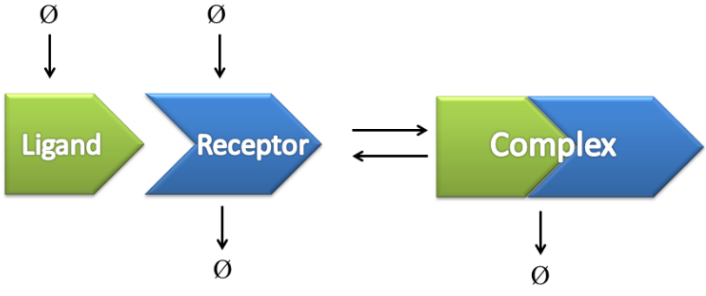


Fig. 6. Ligand-induced Receptor Activation Model

3.2 Negative feedback model of p53-Mdm2

To validate new approach of the global sensitivity analysis based on mutual information measure we tested the method on a well known and widely studied

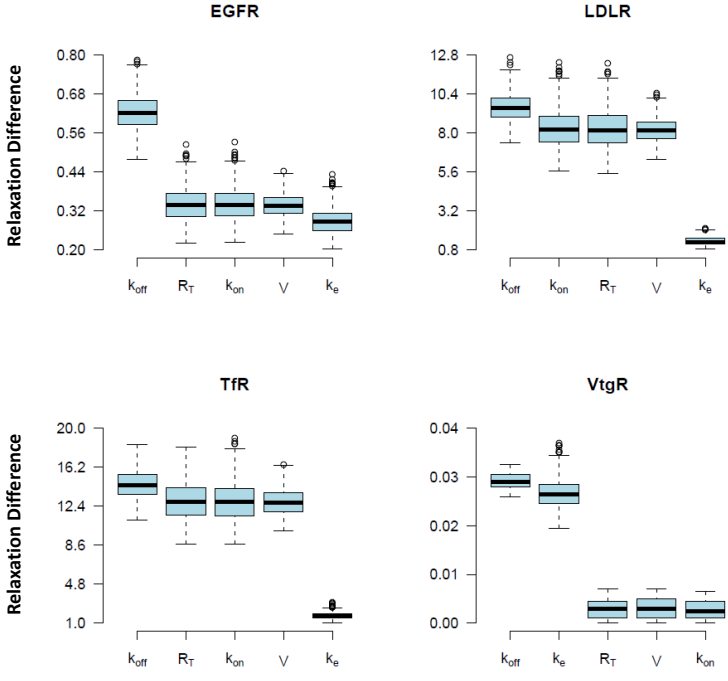


Fig. 7. SA based on Morris method for Ligand-induced Receptor Activation Model (Figure previously published in Biotechnologia by Charzyńska at al., 2012)

example of negative feedback loop of p53 protein and Mdm2 ligase (Zatorsky et al. 2006) for the model scheme see Fig. 8.

Tumour suppressor p53 protein also known as TP53 transcription protein 53 is a transcription factor determining the fate of a cell in case of DNA damage; p53 indirectly, via activation of transcription of the p21 gene encoding, can block cell cycle to repair DNA or activate a process of programmed cell death called apoptosis. The main regulator of the concentration of p53 protein is ligase Mdm2 / Hdm2 (double minute 2 mouse / human double minute 2), which through ubiquitination leads to degradation of p53 in the proteasome. In more than half of the cases of human cancers p53 is inactivated or absent, which allows the mutated tumor cells to replicate and determines their immortality. Consequently, this protein is under investigation due to its property to lead to self-destruction of cancer cells, which could be successfully used as therapy in many types of cancer.

By use of SA method based on MI we were able to capture the negative interactions between parameters β_x and α_{xy} corresponding respectively to p53 inflow and Mdm2 negative loop, for the results of SA see Fig. 4.

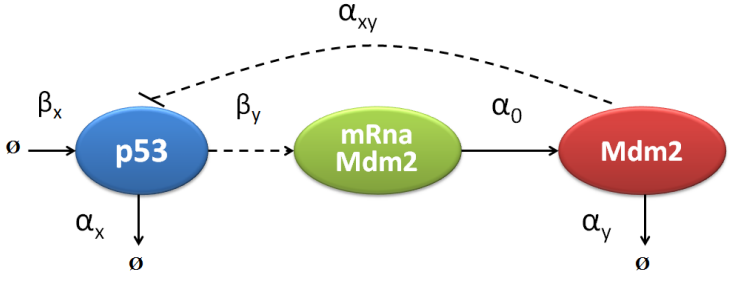


Fig. 8. Negative Feedback p53-Mdm2 Model

3.3 Heat shock response model

One of the most important questions in cell biology is how cells cope with rapid changes in their environment. The range of molecular responses includes a dramatic change in gene expression pattern and higher synthesis of so-called heat shock (or stress) proteins (HSPs). Induction of HSPs increases cell survival under stress conditions (Morimoto 1993). To test hypothesis about heat shock treatment we implemented and verify a mathematical model of heat shock protein synthesis induced by an external temperature stimulus (see Fig. 9), both in deterministic and stochastic meaner. The deterministic model consists of a system of nine non-linear ordinary differential equations describing the temporal evolution of the key variables involved in the regulation of HSP synthesis.

Computational simulations of the model were carried out for different external temperature stimuli. Stochastic version of the model was implemented by use of Chemical Master Equation. To validate the stochastic model output we used the indices of variance to mean ratio for all modelled species. The greatest variance to mean ratio both in homoeostasis and in heat shock case was scored by HSP indicating highest sensitivity to stimuli of rapid substrate concentration raise.

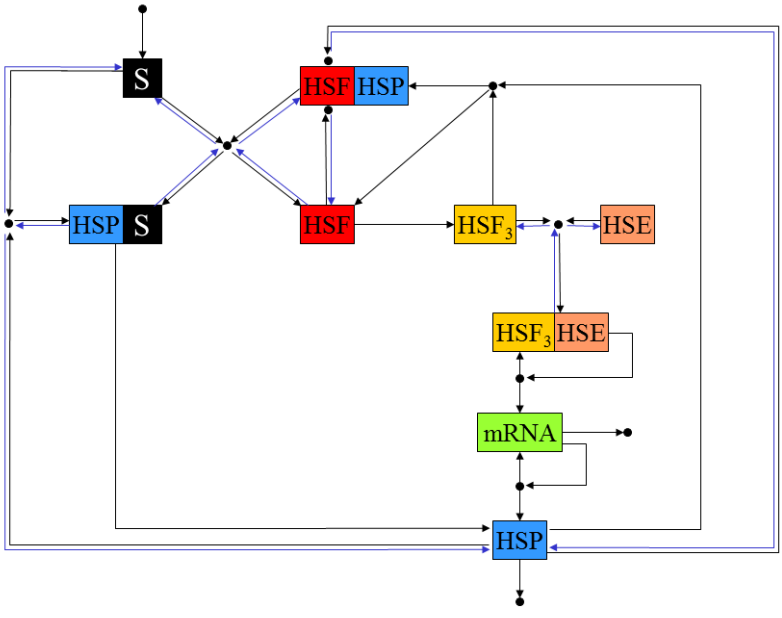


Fig. 9. Heat Shock Response Model

3.4 JAK-STAT feedback model

The paper (Gambin et al., 2013) is a review of computational models of JAK1/2-STAT1 signalling pathway. Despite conceptually simple mechanism of *JAK-STAT* signalling pathway it has highly complex behaviour. This model describes a control mechanism and factors influencing kinetics of the JAK-STAT pathway with increased IFN-γ activity. The model is relatively complex as it captures all essential elements in the JAK1/2-STAT1 signaling. Scheme of the model is depicted in Fig. 10. The model can be informally divided into three modules: receptor module, transcription factor module (the STAT life-cycle) and post-translational feedback module.

Computational modelling is a tool to investigate complex molecular signalling pathways and formalize the description of the dynamics of the system. Computational models integrate experimental data with formal description of a modelled system and consequently they allow to test new hypotheses about interactions between modelled species. In paper (Gambin et al., 2013) we compared three different approaches to the modelling of JAK1/2-STAT1 phenomenon. The sensitivity analysis was useful not only to find the crucial parameters of any analysed JAK-STAT models, but we used it also as a tool to compare different modelling approaches.

steaming

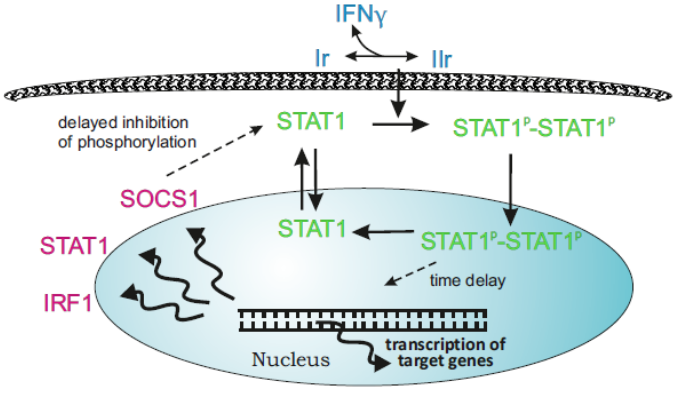


Fig. 10. JAK-STAT Feedback Model

3.5 Sphingolipid metabolism model

In paper (Wronowska et al., 2015) we propose the first comprehensive computational model of sphingolipid metabolism in human tissue. Contrary to the previous attempts, we use a model that reflects cell compartmentalization thereby highlighting the differences among individual organelles, see Fig. 11.

It has been proven that a significant role in the cell apoptosis pathway can be played the ceramides - bioactive lipids, members of sfingolipid family. The exact role of ceramides in signals transduction within nerve cells is still not fully explained. Our motivation was to formally describe an empirically observed correlation between ceramides concentration and cell viability response in human neuroblastoma SH-SY-5Y. Ceramides in low concentrations increase cell viability and may stimulate proliferation but in high concentrations ceramides induce cell apoptosis. One of the hypothesis which may explain the pro-survival role of ceramides in low concentrations is connection with sphingosine 1-phosphate

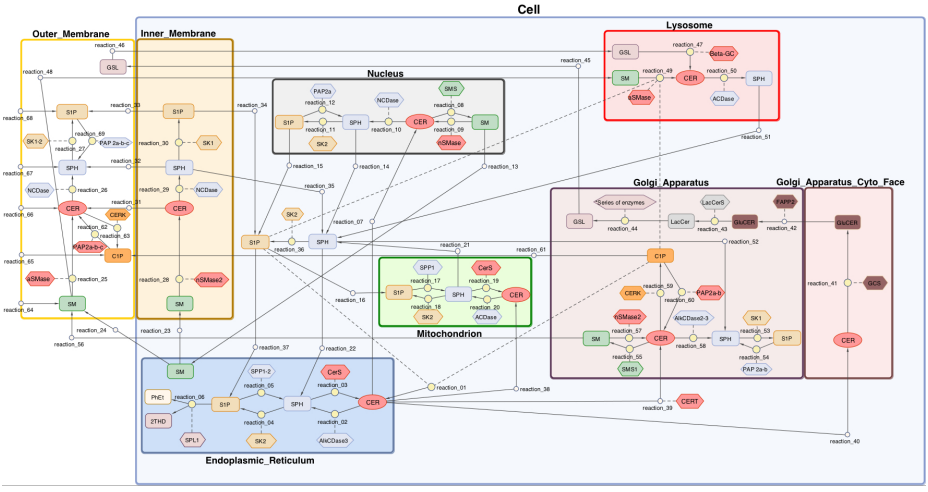


Fig. 11. Sphingolipid Metabolism Model (Figure previously published in BMC Systems Biology by Wronowska et al., 2015)

(S1P) synthesis involving sphingosine ceramide kinase activity. We aim to examine molecular mechanism of cell death evoked by ceramides within nerve cells for both pathological states: cancer and neurodegeneration.

The model that we had build was validated using recently proposed methods of model analysis, allowing to detect the most sensitive and experimentally non-identifiable parameters and determine the main sources of model variance. Moreover, we demonstrate the usefulness of our model in the study of molecular processes underlying Alzheimer's disease, which are associated with sphingolipid metabolism. This model allows to study sphingolipid metabolism disorders that have been observed in various pathological conditions such as cancer and neurodegeneration.

We performed local sensitivity analysis for this ODE model, but unfortunately due to the model complexity the method was of limited use and must have been complemented by the other SA method. The reason for limited applicability of LSA was due to its sensitiveness to the arbitrary choice of time horizon. In case of sphingolipid model we found useful the stochastic noise decomposition method based on LNA described in Section 2.2, for the results see Fig. 3. The StochDecomp method allowed to detect parameters of highest variance components and it complemented the LSA method.

4 Summary

There are plenty recently available SA methods that were developed over decades. We briefly recalled most popular SA methods in Section 2.1. Nevertheless each method has some limitations in its applicability to model assessment. Our in-

terest in development of new SA methods resulted from the need for efficient analytical tools to assess computational models that deeply explore natural complex phenomena. Section 3 contain description of some larger models examples of transcriptional signalling and metabolic pathways that were under our investigation. For this models classical methods such as local SA was of limited use duet to its applicability only to one at the time factor. Consequently it was not enough to perform simple LSA to understand all model dependencies. In case of larger networks (cf. model of sphingolipids metabolism Fig. 11) to understand the complex relations within modelled species and parameters we prefer to investigate all parameters at once, as they may interact one with another and they can have common impact to the model. In case of sphingolipid metabolism model we found useful to compare the results of LSA with the StochDecop output that let us identify the species with the greatest variance component resulting from different reactions.

We also focused on development of a new method based on MI in lieu of its discrete equivalent. This method seems to be promising as it can be applied to any subset of parameters and can provide the information about interactions within parameters groups with respect to the model output. We used this method to compare with the LSA results of the p53-Mdm2 negative feedback loop model. Contrary to LSA that can only provide the sensitivities of a single species to a single parameter separately, SA based on MI provide us with the information of sensitivity of global output to any subset o parameters and moreover with the information of parameters interactions.

To conclude there are many SA methods with different applications. LSA is most popular for biochemical dynamical models but it is a OAT method. The alternative GSA methods are usually computationally inefficient especially for stochastic version of biochemical models. The StochDecomp tool is a solution that by LNA can provide the variance decomposition steaming from separate reactions and can be easily applied to any biochemical model. The SA method based on MI can by applied to samples of data from continuous random variables and can provide the sensitivity indexing for any subset of parameters, consequently we found it useful for biochemical models.

Acknowledgements

The paper is co-founded by the European Union from resources of the European Social Fund. Project PO KL "Information technologies: Research and their interdisciplinary applications", Agreement UDA-POKL.04.01.01-00-051/10-00.

References

1. Andrews S.S. et al., *Detailed simulations of cell biology with smoldyn 2.1*, PLoS Comput. Biol., 6, e1000705, 2010
2. Bentele M., Lavrik I., Ulrich M., Stösser S., Heermann D.W., Kalthoff H., Krammer P.H., Eils R, *Mathematical modeling reveals threshold mechanism in CD95-induced apoptosis*, J. Cell Biol. 166: 839-851, 2004

3. Butcher J.C. , *Numerical Methods for Ordinary Differential Equations*, Wiley, 2003
4. Campolongo F., Saltelli A., Tarantola S. , *Sensitivity analysis as an ingredient of modelin*, Statistic. Sci. 15: 377-395, 2000
5. Charzyńska A., Nałęcz A., Rybiński M., Gambin A., *Sensitivity Analysis of Mathematical Models of Signaling Pathways*, BioTechnologia 93(3), 291–308, 2012.
6. Charzyńska A., Gambin A., *Improvement of k-NN entropy estimator with applications in systems biology*, Entropy - submited.
7. Cukier R.I., Levine H.B., Shuler K.E., *Nonlinear sensitivity analysis of multiparameter model systems*, J. Comput. Phys. 26: 1-42, 1978
8. Degasperi A., Gilmore S., *Sensitivity analysis of stochastic models of bistable biochemical reactions*, Lect. Notes Comput. Sci. 5016: 1-20, 2008
9. Dorval A., *Probability distributions of the logarithm of inter-spike intervals yield accurate entropy estimates from small datasets*, Journal of Neuroscience Methods 173, 129-139, 2008
10. Gambin A., Charzyńska A., Ellert-Miklaszewska A., Rybiński M., *Computational models of JAK1/2-STAT1 signaling*, JAK-STAT 2:3, e24672, Landes Bioscience, 2013.
11. Gillespie D.T. , *A rigorous derivation of the chemical master equation*, Physica A: Stat. Mechan. Applic. 188: 404-425, 1992
12. Gillespie D.T. , *The chemical Langevin equation*, J. Chem. Phys. 113: 297-306, 2000
13. Goutsias J., *Classical versus stochastic kinetics modeling of biochemical reaction systems*, Biophys. J. 92: 2350-2365, 2007
14. Gunawan R., Cao Y., Petzold L., Doyle F.J., *Sensitivity analysis of discrete stochastic systems*, Biophys. J. 88: 2530-2540, 2005
15. Hornberger G.M., Spear R.C., *An approach to the preliminary analysis of environmental systems*, J. Environ. Manag. 12: 7-18, 1981
16. Jetka T., Charzyńska A., Gambin A., Stumpf M.P.H., Komorowski M., *StochDecomp - Matlab package for noise decomposition in stochastic biochemical systems*, Bioinformatics 30(1), 137–138, 2014.
17. Kampen N.V., *Stochastic Processes in Physics and Chemistry*, North Holland, Third Edition, 2007
18. Kazachenko L.F., Leonenko, N.N., *Sample Estimate of the Entropy of a Random Vector*, Probl. Peredachi Inf. 23(2), 9–16 1987.
19. Komorowski M. et al., *Decomposing noise in biochemical signalling systems highlights the role of protein degradation*, Biophys. J., 104, 1783-1793, 2013
20. Komorowski M. et al., *Bayesian inference of biochemical kinetic parameters using the linear noise approximation*, BMC Bioinformatics, 10, 343, 2009
21. Kraskov A., Stögbauer H., Grassberger P., *Estimating mutual information*, Phys. Rev. E 69, 066138, 2004; Erratum Phys. Rev. E 83, 019903, 2011
22. Lüdke N., Panzeri S., Brown M., Broomhead D.S., Knowles J., Montemurro M.A., Kell D.B., *Information-theoretic sensitivity analysis: a general method for credit assignment in complex networks*, J. R. Soc. Interface 5(19), 223–235, 2008.
23. Lazo A., Rathie, P., *On the entropy of continuous probability distributions. Information Theory*, IEEE Transactions on 24, 120-122, 1978
24. Ma J., Sun Z., *Mutual Information is Copula Entropy* , http://arxiv.org/abs/0808.0845v1, 2008
25. Mahdavi A., Davey R.E., Bhola P., Yin T., Zandstra P.W., *Sensitivity analysis of intracellular signaling pathway kinetics predicts targets for stem cell fate control*, PLoS Comput. Biol. 3: e130, 2007

26. Morimoto R.I., *Cells in stress: transcriptional activation of heat shock genes*, Science 259, 1409-1410, 1993
27. Morris M.D., *Factorial sampling plans for preliminary computational experiments*, Technometr. 33: 161-174, 1991
28. Pérez-Cruz F., *Estimation of Information Theoretic Measures for Continuous Random Variables*, In: Advances in Neural Information Processing Systems 21 (NIPS), Vancouver, 2008
29. Ramaswamy R. et al., *Discreteness-induced concentration inversion in mesoscopic chemical systems*, Nat. Commun., 3, 779, 2012
30. Rateitschak K., Karger A., Fitzner B., Lange F., Wolkenhauer O., Jaster R., *Mathematical modelling of interferon-gamma signalling in pancreatic stellate cells reflects and predicts the dynamics of STAT1 pathway activity*, Cell Signal, 22:97-105, 2010
31. Raykar V.C., *Probability Density Function Estimation by different Methods*, ENEE 739Q SPRING 2002, course assignment 1 report, 1-8, 2002
32. Saltelli A., Ratto M., Andres T., Campolongo F., Cariboni J., Gatelli D., Saisana M., Tarantola S., *Global Sensitivity Analysis*, The Primer, Wiley-Interscience, 2008
33. Saltelli A., Ratto M., Tarantola S., Campolongo F., *Sensitivity analysis for chemical models*, Chem. Rev. 105: 2811-2828, 2005
34. Saltelli A., Tarantola S., Chan K.P.S., *A quantitative model-independent method for global sensitivity analysis of model output*, Technometrics 41: 39-56, 1999
35. Shankaran H. Resat H., Wiley H.S., *Cell surface receptors for signal transduction and ligand transport: a design principles study*, PLoS Comput. Biol. 3: 0986-0999, 2007
36. Shankaran H., Wiley H.S., Resat H., *Modeling the effects of HER/ErbB1-3 co-expression on receptor dimerization and biological response*, Biophys. J. 90: 3993-4009, 2006
37. Sjöberg P., Lötstedt P., Elf J., *Fokker-Planck approximation of the master equation in molecular biology*, Comput. Visual. Sci. 12: 37-50, 2009
38. Sobol I.M., Kucherenko S., *Global sensitivity indices for nonlinear mathematical models*, Review. Wilmott 1: 2-7, 2005
39. Sobol I.M., *Sensitivity analysis for nonlinear mathematical models*, Mathem. Model. Comput. Exper. 1: 407-414, 1993
40. Sricharan K., Raich R. III A.O.H., *Boundary Compensated k-NN Graphs*, Proc. IEEE International Workshop on Machine Learning for Signal Processing (MLSP), 2010
41. Sricharan K., Raich R., Hero A, *Optimized intrinsic dimension estimation*, Proc. IEEE Intl. Conf on Acoustics, Speech and Signal Processing (ICASSP), 5418-5421, 2010
42. Swameye I. et al., *Identification of nucleocytoplasmic cycling as a remote sensor in cellular signaling by databased modeling*, Proc. Natl Acad. Sci. USA, 100, 1028, 2003
43. Szymańska Z., Zylicz M., *Mathematical modeling of heat shock protein synthesis in response to temperature change*, Journal of Theoretical Biology 259(3), 562-569, 2009
44. Thomas P. et al., *Intrinsic noise analyzer: a software package for the exploration of stochastic biochemical kinetics using the system size expansion*, PLoS One, 7, e38518, 2012
45. Turanyi T. , *Sensitivity analysis of complex kinetic systems. tools and applications*, J. Mathem. Chem. 5: 203-248, 1990
46. van Kampen N.G., *Stochastic Processes in Physics and Chemistry*, North-Holland Personal Library, Elsevier Science, 2007

47. Vanlier J. et al., *An integrated strategy for prediction uncertainty analysis*, Bioinformatics, 28, 1130-1135, 2012
48. Wallace E. et al., *Linear noise approximation is valid over limited times for any chemical system that is sufficiently large*, IET Syst. Biol., 6, 102-115, 2012
49. Westerhoff H.V., Palsson B.O., *The evolution of molecular biology into systems biology*, Nature Biotechnology, 22, 1249-1252, 2004
50. Wolkenhauer O., Ullah M., Kolch W., Cho K.H., *Modeling and simulation of intracellular dynamics: Choosing an appropriate framework*, IEEE Trans. Nanobiosci. 3: 200-207, 2004
51. Wronowska w., Charzyńska A., Nienałtowski K., Gambin A., *Computational modeling of sphingolipid metabolism*, BMC Systems Biology 9:47, 2015
52. Yamada S., Shiono S., Joo A., Yoshimura A., *Control mechanism of JAK/STAT signal transduction pathway*, FEBS Lett 534:190-6, 2003
53. Yue H., Brown M., He F., Jia J., Kell D., *Sensitivity analysis and robust experimental design of a signal transduction pathway system*, Intern. J. Chem. Kinet. 40: 730-741, 2008
54. Zatorsky N.G., Rosenfeld N., Itzkovitz S., Milo R., Sigal A., Dekel E., Yarnitzky T., Liron Y., Polak P., Lahav G., Alon U, *Oscillations and variability in the p53 system*, Molecular Systems Biology 2: 2006.0033, 2006

Approximate Bayesian Computation Methods in the Localization of Atmospheric Contamination Sources in an Urban Area

Piotr Kopka[1,2], Anna Wawrzyńczak[2] and Mieczyslaw Borysiewicz[2]

[1] Institute of Computer Science, Polish Academy of Sciences
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
[2] National Centre for Nuclear Research,
Otwock, Poland

Abstract. Sudden releases of harmful material into a densely-populated area pose a significant risk to human health. The apparent problem of determining the source of an emission in urban and industrialized areas from the limited information provided by a set of released substance concentration measurements is an ill-posed inverse problem. When the only information available is a set of measurements of the concentration of released substances in urban and industrial areas it is difficult to determine the source of emission. However, the problem can be solved when there is additional information available together with the appropriate tools. A convenient choice is the Bayesian probability framework, which provides a connection between model, observational and additional information about the source. The Bayesian approach was applied in this study to find the posterior probability density function of the contamination source parameters (location and strength) given a set of concentration measurements. The posterior distribution of the source parameters was sampled using an Approximate Bayesian Computation (ABC) algorithm. The stochastic source determination method was validated against the real data set acquired in a highly disturbed flow field in an urban environment. The datasets used to validate the proposed methodology include the dispersion of contaminant plumes in a full-scale field experiment performed within the project "Dispersion of Air Pollutants and their Penetration into the Local Environment in London (DAPPLE)". It demonstrates the use of the proposed approach for the event reconstruction problem in a highly urbanized environment.

1 Introduction

In emergency response management it is important to know the extent of the area that might become contaminated following the release of dangerous material in cities and the subsequent movement of polluted air. The lack of pertinent experimental information means there is a gap in the understanding of

short range dispersion behavior in highly urbanized areas. Given a gas source and wind field, we can apply an appropriate atmospheric dispersion model to calculate the expected gas concentration for any location. Conversely, given concentration measurements and knowledge of the arrangement of buildings, wind field and other atmospheric air parameters, identifying the actual location of the release source and its parameters is difficult. This problem has no unique solution and can be analyzed using probabilistic frameworks. In the framework of Bayesian approach, all quantities included in the mathematical model are modeled as random variables with joint probability distributions. This randomness can be interpreted as lack of knowledge of parameter values, and is reflected in the uncertainty of the true values as expressed in terms of probability distributions. Bayesian methods reformulate the problem thusly: by comparing data, and efficient sampling of a group of simulations, to find a solution.

The problem of source term estimation has been studied in literature, based on both the deterministic and probabilistic approach. [1] implemented an algorithm based on integrating the adjoint of a linear dispersion model backward in time to solve a reconstruction problem. [2, 3] introduced dynamic Bayesian modeling, and the Markov Chain Monte Carlo (MCMC) sampling approaches to reconstruct a contaminant source for synthetic data. Source reconstruction in an urban environment using building-resolving simulations was studied in [4] and [5]. [4] used an adjoint representation of the source-receptor relationship. They used a Bayesian inference methodology in conjunction with MCMC sampling procedures. This approach was validated using data from water channel simulations and a field experiment (Joint Urban 2003) in Oklahoma City. In [5] the authors applied the methodology presented in [2] to the reconstruction of the flow around an isolated building and the flow during IOP3 (third intensive observation period) and IOP9 of the Joint Urban 2003 Oklahoma City experiment. In these experiments they found the source location $\sim$ 70m from the true location for IOP3 (within the domain $\sim 400m \times 400m$) while for the IOP9 model errors and other uncertainties limit the ability to pinpoint the source location.

Methods of approximate Bayesian computation (ABC) are especially useful for problems in which the likelihood function is analytically intractable or too expensive to compute. The original version of the approximate Bayesian computation with Sequential Monte Carlo (ABC SMC) algorithm was proposed in [6]. Applications of this algorithm have been presented in a variety of areas including population biology [7], genetics [8] and psychology [9]. Also, there has been an increased interest in extensions and improvements of this algorithm, as demonstrated in ([10], [11], [12], [13]). The more advanced form of the algorithm, which relies upon the new idea "Sequential Monte Carlo with Adaptive Weights", is shown in Algorithm 1 section 4 and was originally presented in [14].

Previously [15], we have tested the methodology by combining Bayesian inference with MCMC methods and applied these to the problem of dynamic, data-driven contaminant source localization, based on data from the the synthetic experiment. In [16] various modifications of the MCMC algorithm to estimate the probability distributions of searched parameters were examined. We

have shown the advantages of several algorithms. These algorithms use, in a variety of ways, the probability distributions of the source location parameters obtained based on available measurements. Once the new concentration data are received, the marginal probability distribution of the selected parameters is updated. We have also presented the application of the Sequential Monte Carlo (SMC) methods combined with the Bayesian inference to the problem of locating the atmospheric contamination source based on synthetic experiment data [17].

We propose algorithms to locate the source of contamination based on the data from the central London DAPPLE experiment that was performed in May and June 2007 (see section 2) [18]. We used the fast running QUIC-URB [19] model for computing mean flow fields around buildings and QUIC-PLUME [20] as the forward model to predict the concentrations at the sensor locations (section 3). As a sampling approach in the event reconstruction procedure we used the modern algorithm from the class of likelihood-free Bayesian methods [14] with some extension, described in section 4.

2 Dispersion Experiments in London - DAPPLE

The DAPPLE experiment took place in central London (see fig. 1). The two major roads in the vicinity are Marylebone Road, which runs from west to east, and Gloucester Place, which intersects perpendicularly with Marylebone Road near the Westminster City Council building (the red star in fig. 1) [18]. The mean building height in the study area is $21.6m$ (range 10 to $64m$). The experimental site was chosen so as to have a diameter of approximately $500m$ in order to cover the whole dispersion field. There are over 50 experiment sets of dispersion from point sources in the whole DAPPLE data, but to address the issue of source reconstruction we selected a time-resolved contamination experiment. A selected release was carried out on the fourth day, 28th June 2007, in which a sequence of ten samples was taken over a 30 minute sampling period at each of the 18 receptor positions. The sampling process included the collection of ten $150s$ samples at each of the 18 sites, each sample separated from the next by $30s$. The source locations (green X point) and monitoring sites (numbered yellow points) are shown on the map included in fig. 1. The total mass emitted from point-source release was $323mg$ of *perfluoromethyl-cyclohexane* ($PMCH, C7F14$), in accordance with experimental requirements. The other source locations Y and Z were chosen and fixed for the run of experiments conducted during each tracer day. This choice was based on analysis of the weather forecast on the preceding day and a reconstruction of these sources is not present in this publication. Two sets of long-term reference measurements were taken to generate the wind data sets: the rooftop Westminster City Council (WCC) ($18m$) and tower top ($190m$) winds. In order to not increase the height of the domain in the calculations only data from WCC has been taken into account. All aggregate information of the analyzed experiments and wind condition are shown in table 1.

Fig. 1: The map shows the DAPPLE area of central London and is centered at the focal intersection, that of Marylebone Road and Gloucester Place (at 51.5218N 0.1597W). The sampling receptors are numbered 1-18 (yellow dots). Three fixed-point tracer sources (green dots X,Y and Z); red star - Westminster City Council (WCC). The white rectangle shows the computational domain.

In fig. 1 the rectangle area was separated as a computing domain (white line). The positions of all the objects (sensors, source, buildings, wind direction, etc.) have been rotated by 17^o angle, in order to fix the main streets parallel to the edges of the domain. The $latitude - longitude$ geographic coordinate system was changed to the metric system with a reference point $(0, 0)$. This reference point denotes the lower left corner of the white rectangle, both for the convenience of creating a domain and the presentation of results. The domain after the transformation is presented in fig. 2a. All the information presented above (experiment setup - table 1 and the geometry of the domain fig. 2) have been introduced into the Quick-URB environment, which is described in the next section.

DAPPLE experiment summary	
Date and time	28 Jun 07 13:00
Number of tracers released experiment	3
Number of samples and sample duration experiment	10x3 mins
Number of sampling sites	18
Range of source-receptor separations(m)	22-437
Point-source release total mass (mg)	323
WCC Roof data summary	
Wind speed $\frac{m}{s}$	2.6
Wind direction	+19
Longitudinal turbulence u'/U_H	0.80
Lateral turbulence v'/U_H	0.59
Vertical turbulence w'/U_H	0.27

Table 1: DAPPLE and WCC summary [18]

3 Forward dispersion model - QUIC

The Quick Urban Industrial Complex (QUIC) Dispersion Modeling System is intended for applications where dispersion of air pollutants released near buildings must be computed very quickly [20]. The QUIC system, comprises a wind model - QUIC-URB, a dispersion model QUIC-PLUME, and a graphical user interface. The modelling strategy adopted in QUIC-URB was originally developed by Rockle [21] and uses a 3D mass-consistent wind model to combine properly resolved time-averaged wind fields around buildings [22]. The mass-consistent technique is based on a 3D complex terrain diagnostic wind model. The basic methodology involves first generating an initial wind field that includes various empirical parameterizations to account for the physics of flow around buildings. Next, this velocity field is forced to be divergence free, subject to the weak constraint that the variance of the difference between the initial velocity field and mass consistent final velocity field is minimized. The ability of the QUIC-URB model to produce proper wind fields around buildings is dependent on the empirical wind parameterizations. These parameterizations introduce rotation into the flow field and without these parameterizations the method is essentially a potential flow solver. QUIC-PLUME uses a stochastic Lagrangian random walk approach to estimate concentrations in a gridded domain. The model is designed to use averaged wind fields produced by the QUIC-URB system. Parcels, representing substances, are transported with a vector sum of mean winds from QUIC-URB plus turbulent fluctuating winds computed using the random walk equations. Turbulence parameters required in the random walk equations are estimated from vertical and horizontal gradients in the mean wind. A detailed description of the theory is described in [23]. Fig. 2b shows a 3D domain model of the part of London created in QUIC-GUI environment based on the extracted most important buildings from fig. 2a. On the other hand, figs. 2c and 2d present the output of subsystem QUIC-URB which is a wind flows map between

the buildings obtained from WCC measurements. QUIC-PLUME is a 'forward' model, that is run repeatedly for various parameter sets representing position and sources based on the Bayesian inference tool presented in section 4.

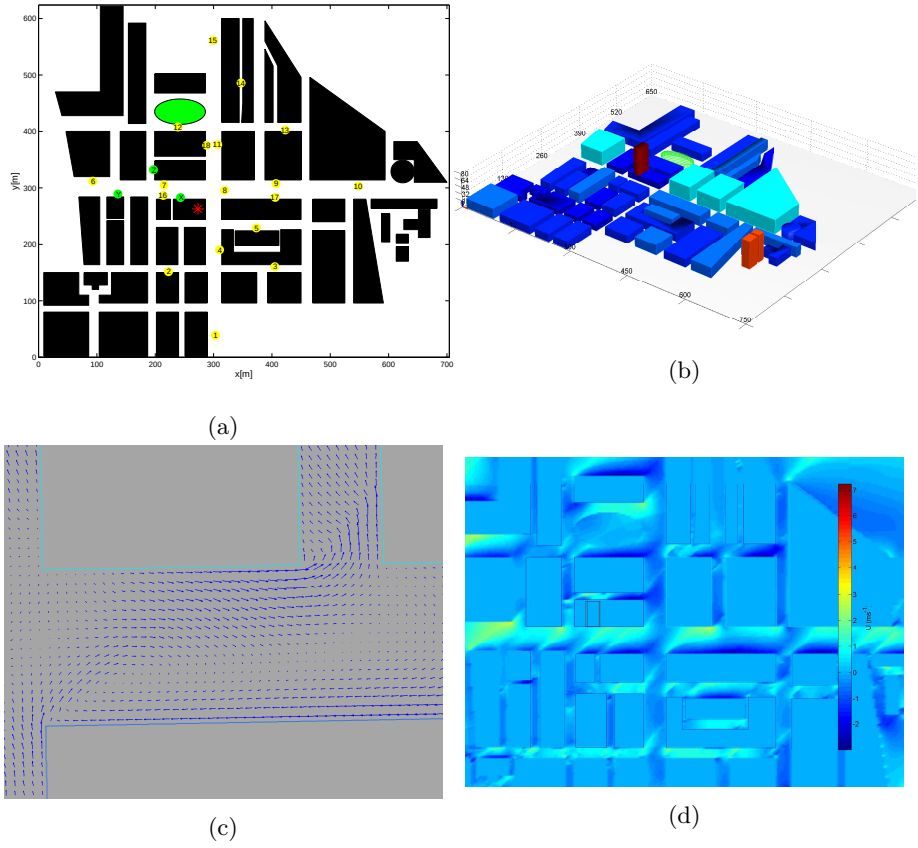


Fig. 2: a) The rotated DAPPLE area, with the selected buildings (black rectangles) and greenery (green ellipses), created using the map from fig. 1 ; sampling receptors are numbered 1-18 (yellow dots), three fixed-point tracer sources (green dots X,Y and Z); red star - Westminster City Council (WCC) b) 3D model of city buildings designed in QUIC-GUI base on the maps. c) map section presenting the wind vectors in the given points d) map presenting the strength of the wind between the buildings in experimental area

4 ABC Methodology

Let θ be a parameter vector, with the prior distribution $\pi(\theta)$. The goal of the Bayesian inference is to approximate the posterior distribution, $\pi(\theta|x) \propto$

$\pi(x|\theta)\pi(\theta)$, where $\pi(x|\theta)$ is the conditional distribution of θ given the data x. The main idea of Approximate Bayesian Computation (ABC) methods is to accept θ as an approximate posterior draw if its associate data x is close enough to the observed data x_{obs}. Accepted parameters are a sample from $\pi(\theta|\rho(x, x_{obs}) < \epsilon)$ where the $\rho(x, x_{obs})$ is the chosen measure of discrepancy, and ϵ is a threshold defining the "closeness margin". If ϵ is sufficiently small then the distribution $\pi(\theta|\rho(x, x_{obs}) < \epsilon)$ will be a good approximation for the posterior distribution $\pi(\theta|x)$. It is often difficult to define an adequate distance function $\rho(x, x_{obs})$ between the simulated and observed data, so in many cases it is replaced with a distance defined by summary statistics, $\rho(S(x), S(x_{obs}))$. However, as we are considering values of concentrations in specific places at a set of time points, we are able to compare those data directly without the use of summary statistics.

In ABC methods, Sequential Monte Carlo (SMC) is used in order to automatically, sequentially "clean" the posterior distribution used to generate proposals for further steps. In $ABCSMC$ methods, the set of samples with weights, called particles, sampled from the population with the prior distribution $\pi(\theta)$, are propagated through a sequence of intermediate posterior distributions $\pi(\theta|\rho(x, x_{obs}) < \epsilon_t)$, $t = 1, ..., T$, until it represents a sample from the target distribution $\pi(\theta|\rho(x, x_{obs}) < \epsilon_T)$. These methods aim to generate draws from $p(\theta|\rho(x, x_{obs}) < \epsilon_t)$, at each of a series of sequential steps t, where ϵ_t defines a series of thresholds. One of the most important issues in $ABCSMC$ is the defining of the particle weights formula correctly. In [14] the authors propose strategies called $ABCSMC$ with Adaptive Weights ($ABCSMCAW$). This method includes a new step where the weights are modified according to the respective values of x. Algorithm 1 shows the description of $ABCSMCAW$ presented in [14].

After initialization of the threshold schedule, first N samples are simulated based on the predefined a priori distribution $\pi(\theta)$ and the corresponding acceptance condition $\rho(x, x_{obs}) < \epsilon_1$. In time step $t = 2$ simple uniform weights are changed based on additional kernel $K_{x,t}(x_{obs}|x_i^{t-1})$ proposed in [14]. Samples, denoted by a tilde are drawn from the previous generation with probabilities v_j^{t-1}. Using perturbation kernel $K_{\theta,t}(\theta_i^t|\tilde{\theta}_i)$ new "fresh" samples θ_i^t are obtained, with the veracity of the condition $\rho(x, x_{obs}) < \epsilon_t$. The weights are calculated according to the formula in step (11); in step (12) the weights are normalized and the time step is increased - $t = t + 1$. The procedure is repeated until $t \leq T$. In the section 4.1 the details are discussed, along with the motivation for choosing specific components of the Algorithm 1 for the problem of stochastic event reconstruction. More information and also theoretical aspects can be found in [14].

4.1 Data and distance measure

In the problem of stochastic event reconstruction all observed data can be split into two types of information: 1) concentration data from the sensor network, and 2) background information. The background information consists of all of the data included in the dispersion model e.g. strength and direction of the wind,

Algorithm 0.1 ABC SMC AW

1. Initialize threshold schedule $\epsilon_1 > \epsilon_2 > ... > \epsilon_T$
2. Set $t = 1$

for $i = 1$ to N **do**
- 3. Simulate $\theta_i^t \sim \pi(\theta)$ and $x \sim \pi(x|\theta_i^t)$
- 4. Until $\rho(x, x_{obs}) < \epsilon_t$
- 5. Set $w_i^t = \frac{1}{N}$

end for

for $t = 2$ to T **do**
- 6. Compute new weights $v_i^{t-1} \propto w_i^{t-1} K_{x,t}(x_{obs}|x_i^{t-1})$ for $i = 1, ..., N$
- 7. Normalize weights v_i^{t-1} for $i = 1, ..., N$
- **for** $i = 1$ to N **do**
  - 8. Pick $\tilde{\theta}_i$ from the set $\{\theta_j^{t-1}\}_{1 \le j \le N}$ with probabilities $\{v_j^{t-1}\}_{1 \le j \le N}$
  - 9. Draw $\theta_i^t \sim K_{\theta,t}(\theta_i^t|\tilde{\theta}_i)$ and $x \sim \pi(x|\theta_i^t)$
  - 10. Until $\rho(x, x_{obs}) < \epsilon_t$
  - 11. Compute new weights as $w_i^t \propto \frac{\pi(\theta_i^t)}{\sum_j v_j^{t-1} K_{\theta,t}(\theta_i^t|\theta_j^{(t-1)})}$
  - 12. Normalize weights w_i^t for $i = 1, ..., N$
- **end for**

end for

temperature and so on. To compute the $\rho(x, x_{obs})$ value we use only data from the sensor network which measures gas concentration $\hat{C}_i^{Sj}$ where i corresponds to the time step and Sj is the sensor identifier. In this test case we have 18 sensors $(S1, S2, ..., S18)$, whose positions are given in fig. 1 and fig. 2a as yellow dots. We assume that the substance concentrations registered by the sensors arrive subsequently at time intervals, hereafter referred to as 'time steps'. It is important to know that for time step t only data $\hat{C}_1^{Sj}\hat{C}_2^{Sj}\dots\hat{C}_t^{Sj}$ are available and finally we have ten time steps $(t = 10)$. The reconstruction algorithm starts to search a source location (x, y) and release rate (q) just after the first 6 minutes $(t = 2)$. To get the predicted concentration a QUIC-PLUME forward model is running and it refers to the procedure $x \sim \pi(x|\theta_i^t)$ in Algorithm 1. To run a dispersion model and obtain data x we use source parameter vector θ_i^t and the information obtained from the QUIC-URB subsystem. The simulated data also have a form of concentration value C_i^{Sj} where Sj corresponds to the known locations of j sensor.

The choice of distance measure or summary statistics is a crucial step in ABC. Since distance measures are not sufficient in many cases, this choice involves a trade-off between loss of information and reduction of dimensionality. In those cases we chose to normalize approximation error between all the data obtained to the current time step t which is also called Fractional Bias (FB) [24] . The FB is used to indicate a bias towards underprediction or overprediction of concentration data by the model. Due to the data type for all sensors in time step t the $\rho(x^t, x_{obs}^t)$ measure is as follows:

$$\rho(x^t, x^t_{obs}) = \frac{1}{18}\sum_{j=1}^{18}(\frac{1}{t}\sum_{i=1}^{t}\frac{|C_i^{Sj} - \hat{C}_i^{Sj}|}{C_i^{Sj} + \hat{C}_i^{Sj}}), \tag{1}$$

under additional definition, that $\frac{|C_i^{Sj}-\hat{C}_i^{Sj}|}{C_i^{Sj}+\hat{C}_i^{Sj}} = 0$ when $C_i^{Sj} = 0$ and $\hat{C}_i^{Sj} = 0$. Given that the concentration $C_i^{Sj} \geq 0$, the value of $\rho(x^t, x^t_{obs})$ is always between 0 and 1 . Let us notice that $\rho(x^t, x^t_{obs}) = 0$ is the situation when our prediction is perfect. In the opposite case, when $\rho(x^t, x^t_{obs}) = 1$ the prediction is wrong. In finding source parameters one of the most important areas is the detection time window, when there is a measurement in the current sensor. The measure (1) supports this approach, because when we have non-zero concentration in some time steps but our model shows that there should be 0 concentration value, the penalty value for this step will be 1. The situation is the same, if the observed value is equal to 0 and the model shows a positive value of the concentration. On the other hand, if $C_i^{Sj} > 0$ and $\hat{C}_i^{Sj} > 0$ then the absolute difference also has an impact on the value of $\rho(x^t, x^t_{obs})$ measure. Finally, the contributions of all time steps are averaged for one sensor. Because $\rho(x^t, x^t_{obs}) \in < 0, 1 >$ one sensor cannot corrupt the overall $\rho(x^t, x^t_{obs})$ value. Also, each sensor has an equal contribution to the $\rho(x^t, x^t_{obs})$ measure, regardless of the level of concentration, which is of course smaller in sensors located further from the source.

4.2 Threshold schedule and weights

The most commonly used adaptive scheme for threshold choice is based on the quantile of the empirical distribution of the distances between the simulated data and observations from the previous population, (see [8], [13]). The method determines ϵ_t at the beginning of the t time-iteration by sorting the measure $\rho(x_i^{t-1}, x_{obs}^{t-1})_{1<i\leq N}$ and setting ϵ_t such that α_t percent of the simulated data $\rho(x_i^{t-1}, x_{obs}^{t-1})_{1<i\leq N}$ are below it, for some predetermined α_t. In [12] the authors show a new strategy based on an acceptance rate curve but also discuss a cumulative number of simulation versus different threshold schedules. In this, and many other cases, quantile-based methods seem to be an easy and appropriate solution of estimating ϵ_t. Based on our own preprocessing experience we set quantile $\alpha_2 = 0.7$ in the second time step, that subsequently decreases to $\alpha_{10} = 0.3$ for $t = 10$ [12]. The additional kernel $K_{x,t}(x_{obs}|x_i^{t-1})$, which is used in calculating the weights, depends on observed and simulated data. Since weights are normalized in step (7), in Algorithm 1 we can simply use the $\rho(x^t, x^t_{obs})$ measure as the proposed kernel. Due to the restriction $0 \leq \rho(x^t, x^t_{obs}) \leq 1$ we can define $K_{x,t}(x_{obs}|x_i^{t-1}) \equiv 1 - \rho(x_i^{t-1}, x_{obs})$, because the greater weight should correspond to a better solution.

4.3 Transition kernel

We chose transition kernel $K_{\theta,t}(\cdot|\cdot)$ to be a Gaussian kernel. Unfortunately in this type of inverse problems the parameters are often highly correlated and

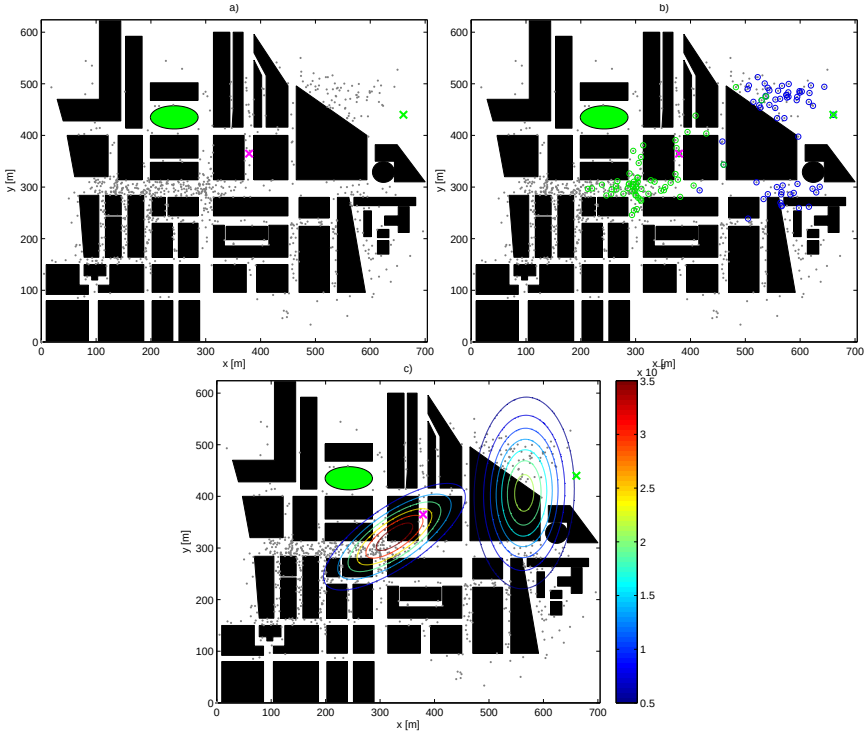


Fig. 3: *a*) all samples - grey point, selected samples - green and magenta crosses; *b*) M-nearest samples to selected green point are marked by blue circles, for magenta green circles; *c*) multivariate normal perturbation kernels evaluated from a set of $M - neighbors$ samples for two selected points.

multimodality is very common. Especially when the (x, y) domain contains a lot of prohibited regions, like buildings. Samples may tend to split in a disjointed group by filling out different street canyons. In such cases it is interesting to consider the use of a local mean and covariance matrix. Instead of computing the covariance matrix based on all the samples from $(t-1)$, a better idea is to use only limited information about the local correlation. In [11] one of the proposed methods is to use the multivariate normal kernel based on the M neighbours. Application of that procedure is presented below:

After the procedure of drawing a new sample from local multivariate normal perturbation kernel, if a new sample is accepted, new weights are also computed using this empirical kernel. The authors in [11] pay attention to the disadvantages of choosing this perturbation kernel. First, the parameter M typically has to be fixed before any of the information about the posterior are known (too small a value of M may lead to a lack of exploration of parameter space, while too large

Algorithm 0.2 ABC SMC AW Step 8

8. Pick $\tilde{\theta}_i$ from the set $\{\theta_j^{t-1}\}_{1\leq j\leq N}$ with the probabilities $\{v_j^{t-1}\}_{1\leq j\leq N}$.
8.1 Select M-nearest samples to $\tilde{\theta}_i$ from the set $\{\theta_j^{t-1}\}$ by using Nearest Neighbors Algorithm.
8.2 Compute the empirical covariance $\sum_{\tilde{\theta}_i,M}^t$ and mean $\bar{\theta}$ from the M nearest neighbours samples of $\tilde{\theta}_i$.
8.4 Set local perturbation kernel $K_{\theta,t}(\theta_i^t|\tilde{\theta}_i) as N(\bar{\theta}, \sum_{\tilde{\theta}_i,M}^t)$.

would offer little or no advantage compared to the standard multivariate normal kernel). In our case the number of samples allocated to one time step is $N = 1000$ samples for each time step. Based on pre-processed experiments we determined the number of neighbors $M = 70$. This kind of procedure may seem to be computationally expensive. However, in experiments the $M - NearestNeighbors$ multivariate normal perturbation kernel minimizes the number of samples needed to be generated, which in the case of stochastic event reconstruction problems is highly preferred. Furthermore, the computation time of running the forward model is much longer than the start-up procedure for finding the nearest neighbors and computing covariance estimation. It is worth mentioning that the choice of the correct determination of the $NearestNeighborsAlgorithms$ is important and depends on the problem.

In the experiment presented in this publication we use classical *M-Nearest Neighbors* algorithm with Mahalanobis distance due to the differences between the various dimensions of the parameters. Results of an experiment using this procedure are presented in fig. 3. This experiment refers to the source location (x, y) but the samples are three-dimensional vectors. We can see that the set of possible solutions is spread among the buildings. Sub-optimal solutions are related to two cases, where the first involves possible sources located in the center of the domain, as contrasted with the north-east location. In fig. 3 *a*) the selected sample is illustrated by a green and magenta cross surrounded by all the samples - i.e. grey points. In fig. 3 *b*) the M nearest samples are marked by blue circles relative to green points and green circles relative to magenta samples. Finally, in c) the subplot shows empirical multivariate normal kernel evaluated from the set of $M - neighbors$ samples for two sets of samples. The shapes of kernel correspond to the correlation between x and y parameters and also support only a single candidate solution. It is worth noting that the locations inside buildings are permitted although the launch dispersion model for these sites is impossible. Consequently, if the drawn sample in step 3) $\theta_i^t \sim \pi(\theta)$ and step 9) $\theta_i^t \sim K_{\theta,t}(\theta_i^t|\tilde{\theta}_i)$ in Algorithm 1 does not satisfy the assumptions then there is a re-drawing of the θ_i^t sample. The next section presents the results for the stochastic parameters reconstruction for the setup described above and the experimental data presented in section 2.

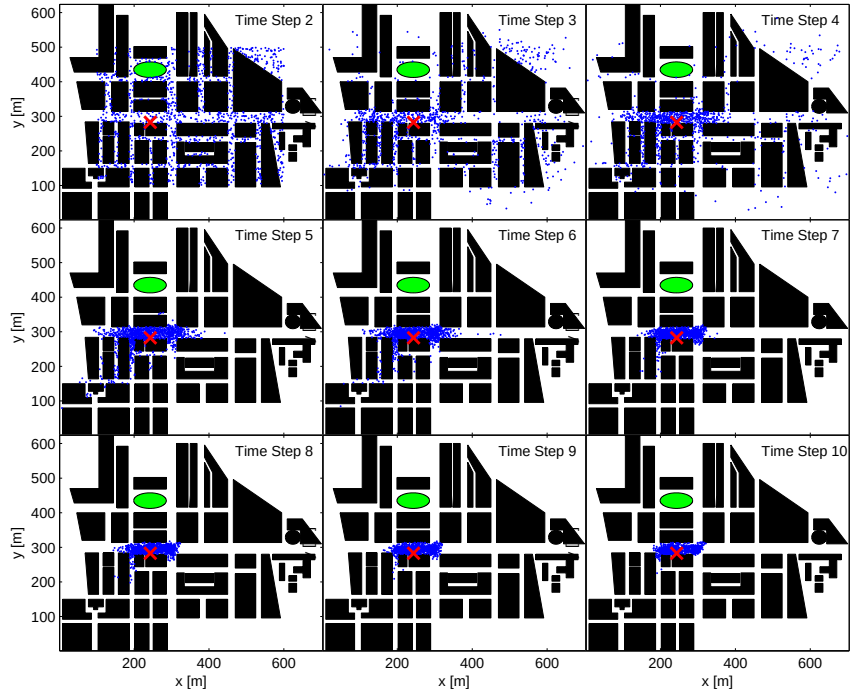


Fig. 4: A scatter plot of all samples generated in the subsequent time steps $t = 2, 3, ..., 10$ in (x, y) space of source location. The red cross marks the true source position

5 Results of DAPPLE reconstruction experiment

Fig. 4 shows the locations of the buildings in the DAPPLE London area, together with all the samples generated in subsequent time steps $t = 2, 3, .., 10$ which are decomposed directly to $6, 9, .., 30$ experimental minutes. As we can see, samples after the $4th$ time step converge from all possible (x, y) space to the vicinity of the actual source location. Using these samples, we construct the marginal probability distributions for the source location and release rate, as shown in fig. 5 for all time intervals. As time goes on, the mass of probability distribution is concentrating in the vicinity of the proper values of x and y. This looks quite different for emissions amounts, where posterior distribution for the parameter q looks like a bimodal distribution. This is better shown in figure fig. 6 where all samples are included.

After limiting the (x, y) domain to the area surrounding the real source, we can see that the distribution is divided into two areas, which suggests two different solutions of the problem. One location is closer to the main intersection

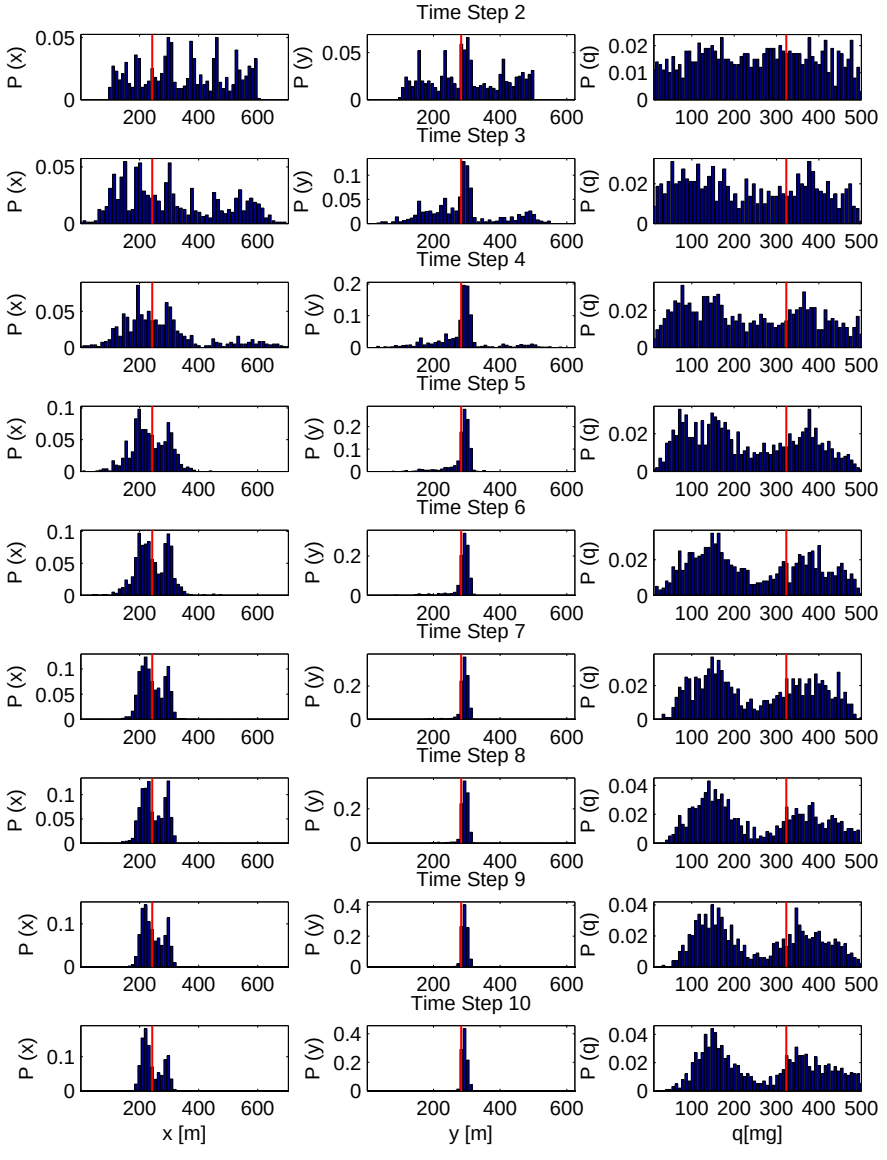


Fig. 5: Evolution of the marginal posteriori probability distribution for x, y and q parameters for time steps $t = 2, 3, .., 10$. The red vertical line represents target value of parameters.

and the second is around the true source. These results can be also noticed in the marginal distributions for x where we note two picks of probabilities. The results from all time steps are summarized in a so-called trellis plot presented in fig. 6, where the parameters reconstruction was started after 6 minutes. A color pattern reflected in fig. 6 was used to show empirical 2D probability distribution of all parameters combinations. The colored contour lines are enveloping higher probability at the joint posterior distributions. The diagonal plots are marginal empirical posterior distributions of the forward model parameters. The real parameter values from the field experiment are highlighted with vertical red lines in diagonal plots and black cross markers on the other subplot, which are successfully captured by the high posterior probability region. The correct position obtained after the transformation of the relative domain is $x = 243m$, $y = 282m$ and $q = 323mg$, where the most probable parameters values are $P(x = 223.0 \pm 7.6m) = 0.0632$, $P(y = 291.4 \pm 6.7m) = 0.1990$ and $P(q = 144.9 \pm 5.3mg) = 0.0218$. To accurately analyze the results for release rate parameter in fig. 7 a) we split the samples into two groups supported by two separate probability masses. After this assumption, two different groups of samples are presented in fig. 7 b). One can see that the green samples corresponding to $q < 250mg$ are distributed closer to the center, while the blue points are closer to the true source (red cross) and the corresponding estimates of $q = 323mg$ group closer to the real value (see fig. 7 a) red vertical line). Fig. 7 c) shows two histograms of weights $1 - \rho(x^{10}, x^{10}_{obs})$ for the green and blue points. As we can see, more points from the blue subset have higher weights (better model fit). As it means that the points have higher probability to be drawn in the next step, we can conclude that with the extension of the reconstruction procedure the "green" solutions should be slowly converging to the other (blue solution) which is close to the true value of the source parameters.

6 Conclusion

A stochastic event reconstruction method for atmospheric contaminant dispersion in an urban environment has been presented. The method described in section 4 is based on Bayesian inference with the Approximate Bayesian Computation (ABC) tool with an extension. Fast-running QUIC-PLUME dispersion models have been adopted as the forward model in the Bayesian framework. The dispersion model has been uniquely enhanced by taking into account empirical wind turbulence between buildings obtained from the QUIC-URB tool. Additional attention was given to the formulation of the distance function to take into account concentration measurements provided in successive time steps that can be available from a sensor network. The event reconstruction method has been successfully validated against the real DAPPLE experiment. In particular, the modeling of a priori distribution based on the threshold schedule substantially improved the results. Also the transition kernel set treated as a local empirical distribution, conformable to the non-standard domain, had an impact on convergence. In the event reconstruction of the DAPPLE tracer experiment, up to three

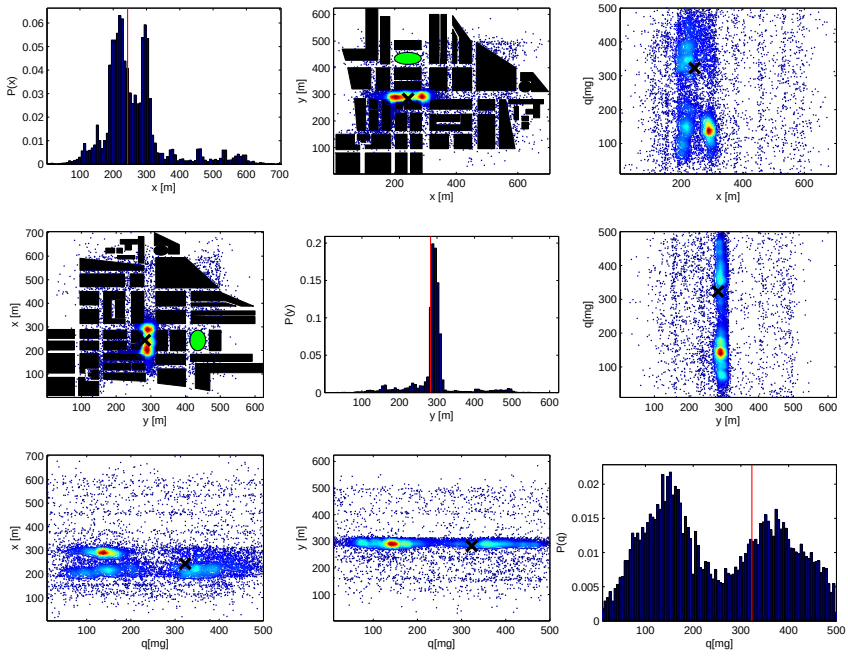


Fig. 6: Bivariate and marginal posterior distributions for all parameters $\theta \equiv (x, y, q)$. The plot is colored according to probability density, where the most probable regions are colored the deepest red (i.e., a heatmap). The vertical red lines in diagonal plots (black cross in bivariate) show the real value of each parameter. The distributions are built based on all the samples generated in the reconstruction procedure.

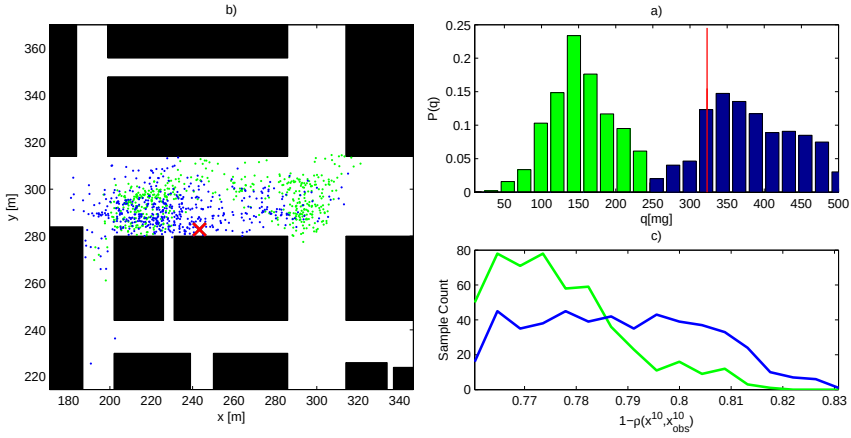


Fig. 7: *a*) Marginal posteriori distribution of q split into two sample sets *b*) Scatter plot of all samples in the (x, y) space - the sample colors correspond to the sample sets in a) *c*) The histogram of weights, which was obtained from the two groups of samples - green and blue.

parameters were estimated. From a practical point of view, release location and emission rates are of the greatest significance to the emergency responders. The present study has shown that the event reconstruction problem can be solved for the urban area without using the time-consuming Computational Fluid Mechanic model. Posterior probability distributions of model parameters were also used to build priori distribution when new concentration data became available. Although the ABC framework is general, a comprehensive operational event reconstruction tool needs to address various release scenarios. The present study focused on steady point source releases in a highly urbanized area. However, possible release scenarios may include moving sources. Furthermore, the scale of the event may range from local sites to areas of greater size. Future work will concentrate on adding new possible hazardous scenarios to the present stochastic event reconstruction tool, not necessarily the release of gases into the atmosphere.

7 Acknowledgments

The study is cofounded by the European Union from resources of the European Social Fund. Project PO KL "Information technologies: Research and their interdisciplinary applications", Agreement UDA-POKL.04.01.01-00-051/10-00.

References

1. Pudykiewicz, J.A.: Application of adjoint tracer transport equations for evaluating source parameters. Atmospheric environment **32**(17) (1998) 3039–3050

2. Johannesson, G., Hanley, B., Nitao, J.: Dynamic Bayesian models via Monte Carlo an introduction with examples. Lawrence Livermore National Laboratory, UCRL-TR-207173 (2004)
3. Johannesson, G., Dyer, K., Hanley, W., Kosovic, B., Larsen, S., Loosmore, G., Lundquist, J., Mirin, A.: Sequential Monte-Carlo based framework for dynamic data-driven event reconstruction for atmospheric release. In: Proc. of the Joint Statistical Meeting, Minneapolis, MN, American Statistical Association and Cosponsors. (2005) 73–80
4. Keats, A., Yee, E., Lien, F.S.: Bayesian inference for source determination with applications to a complex urban environment. Atmospheric environment **41**(3) (2007) 465–479
5. Chow, F.K., Kosovic, B., Chan, S.: Source inversion for contaminant plume dispersion in urban environments using building-resolving simulations. Journal of applied meteorology and climatology **47**(6) (2008) 1553–1572
6. Sisson, S.A., Fan, Y., Tanaka, M.M.: Sequential monte carlo without likelihoods. Proceedings of the National Academy of Sciences **104**(6) (2007) 1760–1765
7. Toni, T., Welch, D., Strelkowa, N., Ipsen, A., Stumpf, M.P.: Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. Journal of the Royal Society Interface **6**(31) (2009) 187–202
8. Beaumont, M.A., Zhang, W., Balding, D.J.: Approximate Bayesian computation in population genetics. Genetics **162**(4) (2002) 2025–2035
9. Turner, B.M., Van Zandt, T.: A tutorial on approximate Bayesian computation. Journal of Mathematical Psychology **56**(2) (2012) 69–85
10. Lenormand, M., Jabot, F., Deffuant, G.: Adaptive approximate Bayesian computation for complex models. Computational Statistics **28**(6) (2013) 2777–2796
11. Filippi, S., Barnes, C.P., Cornebise, J., Stumpf, M.P.: On optimality of kernels for approximate Bayesian computation using sequential Monte Carlo. Statistical applications in genetics and molecular biology **12**(1) (2013) 87–107
12. Silk, D., Filippi, S., Stumpf, M.P.: Optimizing threshold-schedules for approximate Bayesian computation sequential Monte Carlo samplers: applications to molecular systems. arXiv preprint arXiv:1210.3296 (2012)
13. Del Moral, P., Doucet, A., Jasra, A.: An adaptive sequential Monte Carlo method for approximate Bayesian computation. Statistics and Computing **22**(5) (2012) 1009–1020
14. Bonassi, F.V., West, M., et al.: Sequential Monte Carlo with Adaptive Weights for Approximate Bayesian Computation. Bayesian Analysis **10**(1) (2015) 171–187
15. Borysiewicz, M., Wawrzynczak, A., Kopka, P.: Stochastic algorithm for estimation of the model's unknown parameters via Bayesian inference. In: Computer Science and Information Systems (FedCSIS), 2012 Federated Conference on, IEEE (2012) 501–508
16. Borysiewicz, M., Wawrzynczak, A., Kopka, P.: Bayesian-based methods for the estimation of the unknown model's parameters in the case of the localization of the atmospheric contamination source. Foundations of Computing and Decision Sciences **37**(4) (2012) 253–270
17. Wawrzynczak, A., Kopka, P., Borysiewicz, M.: Sequential Monte Carlo in Bayesian Assessment of Contaminant Source Localization Based on the Sensors Concentration Measurements. In: Parallel Processing and Applied Mathematics. Springer (2014) 407–417
18. Wood, C.R., Barlow, J.F., Belcher, S.E., Dobre, A., Arnold, S.J., Balogun, A.A., Lingard, J.J., Smalley, R.J., Tate, J.E., Tomlin, A.S., et al.: Dispersion experi-

ments in central London: the 2007 DAPPLE project. Bulletin of the American Meteorological Society **90**(7) (2009) 955–969
19. Pardyjak, E.R., Brown, M.: QUIC-URB v1. 1 Theory and User's Guide. Los Alamos National Laboratory, Los Alamos, NM (2003)
20. Williams, M.D., Brown, M.J., Singh, B., Boswell, D.: QUIC-PLUME theory guide. Los Alamos National Laboratory (2004)
21. Röckle, R.: Bestimmung der Strömungsverhältnisse im Bereich komplexer Bebauungsstrukturen. na (1990)
22. Sherman, C.A.: A mass-consistent model for wind fields over complex terrain. Journal of applied meteorology **17**(3) (1978) 312–319
23. Williams, M.D., Brown, M., Boswell, D., Singh, B., Pardyjak, E.: Testing of the QUIC-PLUME model with wind-tunnel measurements for a high-rise building. In: 5th AMS Urban Env Conf, Vancouver, BC, Canada LA-UR-04-4296. (2004)
24. Cox, W.M.: Protocol for determining the best performing model. Technical report, Environmental Protection Agency, Research Triangle Park, NC (United States). Technical Support Div. (1992)

Modified Concentric Rings Trajectory (cCR) in Hyperpolarized ^{13}C Magnetic Resonance Spectroscopy Imaging

Kamil Lorenc[1,2], Christoffer Laustsen[3], Hans Stødkilde-Jørgensen[3], and Rolf F Schulte[4]

[1] Institute of Computer Science, Polish Academy of Sciences, ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
[2] Nałęcz Institute of Biocybernetics and Biomedical Engineering, Polish Academy of Sciences, ul. Ks. Trojdena 4, 02-109 Warsaw, Poland
[3] Aarhus University, MR Research Center, Nordre Ringgade 1, DK-8000 Aarhus C, Denmark
[4] GE Global Research, Freisinger Landstrasse 50, D-85748 Garching b. Muenchen, Germany

Abstract. *Hyperpolarization* can increase signal in NMR experiments by up to 4 orders of magnitude. Among available techniques, hyperpolarization dissolution DNP (*dissDNP*) is particularly interesting in the clinical setting as a non-radioactive method for provides novel diagnostic data.
This study is focused on the application of concentric rings sequence. This sequence is based on gradient train to simultaneous acquisition of spatial and spectral dimensions, to imaging hyperpolarized pyruvate and its spectral downstream metabolites alanine and lactate. The study aims to include non-localized spectroscopic information in the image reconstruction process.
A better robustness was found for our method than for the previously published alternative. With an improved spectral resolution via the inherent sampling of k-space zero, central concentric rings can be a valuable alternative in ^{13}C hyperpolarization studies.

1 Introduction

Nuclear magnetic resonance (*NMR*) is a phenomenon commonly used in medicine and chemistry to determine structure, magnetic properties and chemical composition of the tissue. Although NMR could provide in vivo spatially resolved information, its application is limited by inherent low signal to noise ratio (SNR).

As a remedy for low SNR a few methods for hyperpolarization were introduced. Hyperpolarization can increase the signal in NMR experiments by

up to 4 orders of magnitude. Among the available techniques dissolution DNP (*dissDNP*) is particularly interesting in the clinical setting potentially providing useful diagnostic data[1].

In hyperpolarization chosen compounds are prepared (hyperpolarized) in a polarizer and then administrated to study a particular organ of interest. Hyperpolarized [1-^{13}C] pyruvate was used in clinical trials [1] and applied in number of cancer models (e.g. [2]) or metabolic impairments (e.g. [3]). This boosted interest in the application of hyperpolarized pyruvate as a new generation of contrast agents in clinical trials. The introduction of *dissDNP* to pre-clinical and clinical studies requires fast and robust sequences maximizing the obtainable information.

For imaging of spatial concentration of pyruvate and its metabolites methods known from magnetic resonance spectroscopy imaging (*MRSI*) can be adopted. The signal in *MRSI* can be described as linear combination of finite number of metabolites

$$z(t) = \sum_{m=1}^{M} A_m * z_m(t) = \sum_{m=1}^{M} A_k * exp(-t/T_{2m} + i * 2\pi f_{0m} + i * \phi_{0m}) \qquad (1)$$

where: A_m - amplitude of k-th metabolite; T_{2m} - spin spin relaxation constant; f_{0m} - frequency of chemical shift; ϕ_{0m} - initial phase;

Equation 1 shows some important aspects of signal in magnetic resonance spectroscopy: the signal relaxes with a time constant T_2; the signal of k-th metabolite rotates around frame of reference with a frequency f_0 (i.e. chemical shift frequency). Frequency f_0 is a quantity which allow for distinguishing between chemical compounds. The amplitude of a signal is proportional to concentration of the metabolite. Relaxation process in general can be multiexponential not just monoexponatiol as presented in eq. 1

For compounds of interests in hyperpolarized magnetic resonance i.e. pyruvate, lactate, alanine, pyruvate hydrate chemical shift and frequency for B_0 3.0T as used in this study listed in table 1. Frequency can be changed by modulating with carrier frequency in the receive path.

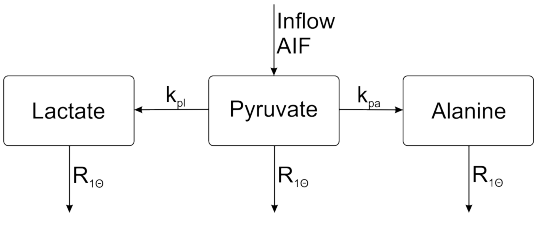


Fig. 1: Model used for simulation of signal evolution; (k_{pl}, k_{pa}) are apparent reaction rates; $R_{1\theta}$ are relaxation rates which includes $1/T_1$ relaxation rates and lost of signal due to excitation

Model of signal kinetic is presented in figure 1. After inflowing to organ of interest pyruvate is metabolized to lactate and alanine with a rate k_{pl} and k_{pa} respectively. For all metabolites pyruvate, lactate and alanine signal decay is observed. This loss is caused by T_1 relaxation and excitation.

Imaging of the metabolite spatial concentration require choosing k-space trajectory. An arbitrary 2d k-space trajectory signal acquired in magnetic resonance

scanner is described by equation 2 adopted from [4]. Equation 2 presents situation after slice selective excitation.

$$S(t) = \sum_{m=1}^{M} \int \int A_m(x,y) z_m(t) exp(i(xk_x(t) + yk_y(t))dxdy \quad (2)$$

where: $k(t) = (k_x(t), k_y(t))$ is an arbitrary k-space trajectory. When k(t) is chosen to be Cartesian trajectory $A_m(x,y)$ can be reconstructed by Fourier transformation. When k is an arbitrary non-cartesian trajectory $A_m(x,y)$ has to be reconstructed by non-uniform Fourier transformation (problem of type I non-uniform Fourier transformation [5]) or gridding algorithm.

	Pyruvate	Lactate	Alanine	Pyruvate Hydrate
Chemical shift [ppm]	170.6	183.2	176.5	179.0
Resonance frequency [Hz]	-200	192	-23	67

Table 1: Chemical shift for a compounds of interest in hyperpolarized magnetic resonance [6]. Chemical shift listed for compounds labelled on [1-^{13}C] position. Resonance frequency given for B_0 3.0T as used in this study.

Transformation of the coordinate system requires multiplication of the new coordinate system by a determinant of Jacobian of transformation. That can be also thought as filtering with density compensation function (dcf) where data sampled in a denser region of k-space has to be penalized/filtered.

Equation 2 identifies that except of spatial dimensions, a spectral dimension has to be encoded as well in acquisition. Further this simplified scenerio is only valid in the static case. In fact living organisms are dynamic and impairment of homeostasis often leads to severe diseases. This made a need for dynamic metabolic imaging study. In this case $A_m(x,y)$ is also function of t - $A_m(x,y,t)$.

In effect, the signal evolves in 5 dimensions - 3 spatial - 1 spectral and 1 temporal. A number of methods have been designed for image acquisition of the high-dimensional space in hyperpolarized MR. Some of them were recently reviewed [7]. The other method which allowed for fast spectroscopic acquisition are spatiotemporal encoding[8] or balanced steady state free precession (bSSFP)[9] and MAD-STEAM[10]. When off-resonant effects can severely distort images, sequences that that provide full spectral information are preferred [7], thereby limiting the acquisition options.

Probably the most commonly used sequence is free induction decay chemical shift imaging (*13C-FID-CSI*) which compresses time domain to one point. It relies on the sequential acquisition of *FID* from consequent k-space points which are determined by phase encoding gradients. If a whole acquisition of *13C-FID-CSI* could be reduced to one moment in time then the image is a linear function of true enzymatic conversion [11].

The methods commonly used in full spectrum acquisition utilize gradient echo trains, which rely on traversing through k-space in a periodic manner. This allows for simultaneous acquisition of spatial and spectral dimensions. Depending on the k-space trajectory, there are a few measurement options: echo-planar spectroscopic imaging (*EPSI*)[12], spiral-chemical shift imaging[13] and introduced most recently concentric rings[14].

This paper is organised as follow: section 2.1 contains numerical simulation for effects of using *13C-FID-CSI* sequence to study dynamic process. Section 2.2 suggest a fast method for acquisition in hyperpolarized magnetic resonance, and is then compared to other similar sequence.

The main innovation of this work is the application of concentric rings sequence for imaging hyperpolarized [1-^{13}C] pyruvate and its spectral downstream metabolites ([1-^{13}C] alanine and [1-^{13}C] lactate. The aim was to include information from the central k-space point (i.e. non-localized spectroscopy) in the reconstruction process. We refer to this version as central concentric rings (***cCR***), which are different to previously published non-central concentric rings (***ncCR***)[14].

2 Results

2.1 Simulation of signal dynamic in hyperpolarized 13C MRSI

13C-FID-CSI sequences with phase encoding schemes in fig. 3 was simulated for reconstruction matrix 16x16; repetition time (*TR*) 80 ms; flip angle θ 10°. True enzymatic conversion rates were defined as a Shepp Logan-like phantom (fig. 2). Lactate and pyruvate noise-free images were simulated by the one-way kinetic linear model with fixed relaxation rate $T_1 = 45\ s$ for all metabolites. Dirac's delta was chosen as the arterial input function (*AIF*). Using the same approach an *EPSI* sequence behaviour was simulated (*TR* 200 ms; $\theta = 10°$).

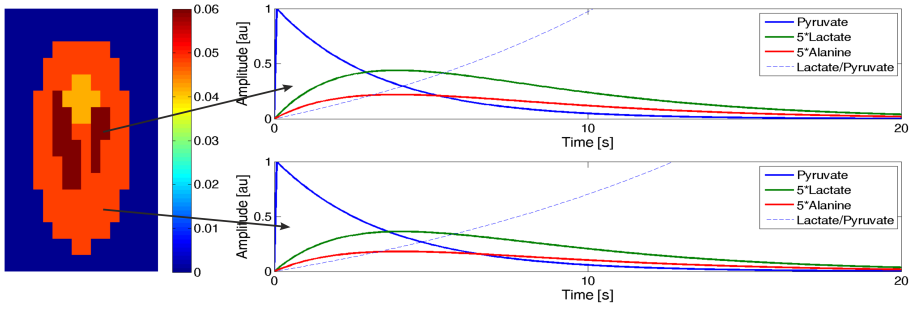


Fig. 2: Left: a numerical phantom used to simulate metabolic signals; k_{pl} values shown on color bar; Right: signal of pyruvate lactate alanine and lactate/pyruvate ratio for distinct areas of phantom.

Qualitative analysis of CSI sequences with different starting times with respect to bolus arrivals are presented in fig. 3. The upper row shows results for CSI sequential phase encoding and the bottom row shows results for in-out phase encoding scheme. A Shepp Logan shaped phantom (fig. 2) was used for simulation. Each voxel was represented as two values (k_{pl}, k_{pa}) which are apparent reaction rates of pyruvate to lactate and pyruvate to alanine. A voxel signal was simulated using model shown in figure 1. Then, the signal in a point in time was sampled in k-space. At the end of whole acquisition the signal was transformed back to the final image. Figures 3 and 4 show values of the lactate/pyruvate ratio, which typically reflects k_{pl} parameter and is a promising marker in cancer diagnostic. Figure 2 shows how the lactate/pyruvate ratio is changing during sampling of images.

The amount of blurring is substantially reduced by applying fast spectroscopic sequences, which can be clearly observed in figure 4.

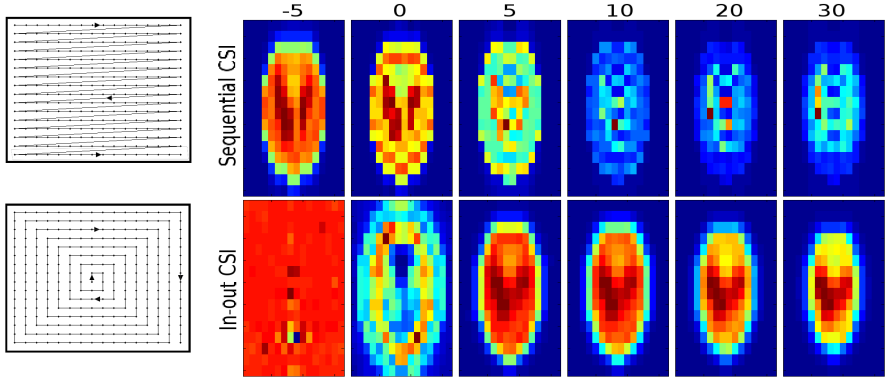


Fig. 3: Left: Phase encoding schemes for *13C-FID-CSI* typically used in hyperpolarized magnetic resonance imaging. Top row sequential order, bottom from origin to edge of k-space - in-out order. Right:Simulation of CSI images (top: sequential CSI; bottom: in-out CSI) with varied starting time with respect to bolus arrival (-5, 0, 5, 10, 20 and 30 seconds after bolus)

Our simulations show a need for applying fast schemes in hyperpolarized magnetic resonance.

2.2 Concentric rings trajectory

An additional halving of acquisition time can be obtained by using concentric k-space instead of Cartesian readout trajectories. Concentric ring trajectory have been shown to be a viable alternative to *EPSI* by *Jiang et al*[14]. However in this version of concentric rings central k-space point is omitted during acquisition. As mentioned in section 1 transformation of the coordinate system require including

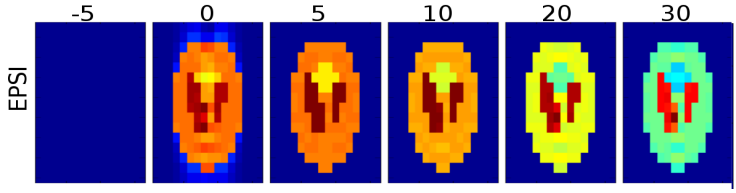


Fig. 4: Simulation of *EPSI* images with varied starting time with respect to bolus arrival (-5, 0, 5, 10, 20 and 30 seconds after bolus)

determinant of Jacobian matrix for the transformation. For a transformation to polar coordinate the determinanat: $|J| = r$. In result for a central k-space point where $r = 0$, data from central k-space point vanishes. However this is true only for continues case. For a discrete case central k-space point should be weighted with $\pi \cdot dr^2/4$ as shown is figure 5.

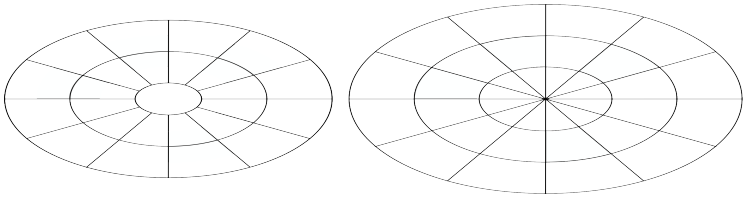


Fig. 5: Ilustration of area covered by k-space points for cCR (left) and ncCR (right). Area correspond to the value of density compensation function for respetive points. Central points in left image correspond to area πr^2 where r radius of smallest circle

K-space data was regridded to a Cartesian grid with the density compensation function (cf. [15]) as defined in eq.3. Then data was transform to an image domain with inverse Fourier transformation.

$$\begin{aligned} dcf(r,\phi) &= r \cdot dr \cdot d\phi \\ dcf(0,\phi) &= \pi \cdot dr^2/4 \end{aligned} \tag{3}$$

2.3 Numerical simulation of concentric rings trajectory

Image Reconstruction Toolbox[16] was used for efficiency comparison of the *cCR* and *ncCR* k-space signal. The simulation was performed on high resolution regulat Shepp Logan phantom. Normally distributed random noise was added to the complex signal in the k-space domain. Then, the reconstructed image was compared to Shepp Logan phantom with mutual information. The advantage of this metric is a clearly defined lowest (0) and highest values (entropy of reference image). Mutual information between images was computed after discretizing both to 256 bins.

Analysis for cCR and ncCR (fig. 6) with a presents of noise, shows a better cCR performance. Artifacts in the corner are originating from the point spread function (PSF) of cCR and $ncCR$ sequences[15].

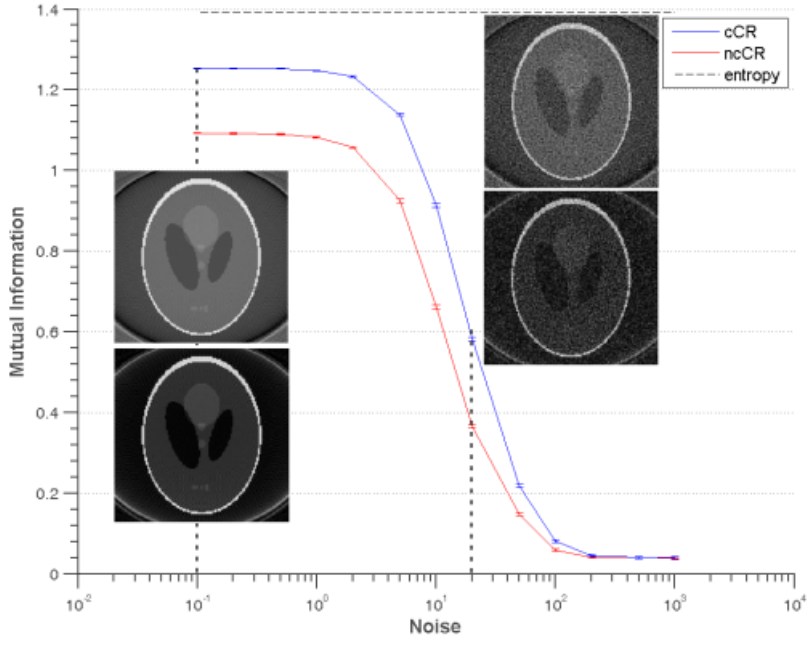


Fig. 6: Monte Carlo simulation of cCR and ncCR sequences. Mutual Information with Shepp Logan phantom vs noise level for central Concentric Rings (blue) and non-central Concentric Rings (red); Shepp Logan phantom entropy (dashed)

2.4 Phantom measurements with concentric rings

Further comparison of cCR and $ncCR$ sequences were performed by phantom measurements in thermal equilibrium condition. Sequences were designed for 500 Hz spectral bandwidth; 100 ms readout time. Data were phase corrected linearly in readout direction. Data was regridded to a Cartesian grid with density compensation function as defined in eq. 3. 15 repetitions of sequence were acquired.

The phantom consisted of a sodium acetate syringe and a bicarbonate ball. Thermal equilibrium phantom measurements (TR=2000 ms; $\theta = 90°$) was done on 3T clinical MR scanner (GE HDx, USA) and dual-tunned $^{1}H - ^{13}C$-volume coil[17].

sequence	mean ± sd [au]
cCR	**24.89 ± 1.05**
ncCR	29.59 ± 1.07

Table 2: Residuum between real data and fitted model for phantom measurements summed over whole image space. Criterion of type less-better. Difference between values statistically significant.

Protocols were compared by using t-test for the sum of residuals between data and damped sinusoidal model [18] as defined in equation 1. Nonlinear least square method (trust region reflective algorithm [19], [20]); was used for fit model parameter (A_i, f_0, T_2, ϕ_0) to the data. Initial guess was chosen as $f_0 = \int\int |A(x,y)|dxdy$; $1/T_2 = 15\ Hz$; $A_m = max(|\rho_z(x,y)|)$.

Criterion is type less-better. Results for phantom measurements are summarised in table 2. *cCR* shows a statistically significant better performance over ncCR (p-value¡0.05).

2.5 *In vivo* measurements - proof of concept

Final validation of *cCR* as described in section 2.4 (TR=200 ms; $\theta = 12°$) was done by *in vivo* measurements in a healthy rat. One injection of 1,5 ml, 90 mM hyperpolarized pyruvate (polarizer SpinLab, GE, USA) was performed.

In vivo results show good quality of spectra in a single voxel (fig. 8). Figure 7 shows feasibility of dynamic studies with cCR sequences.

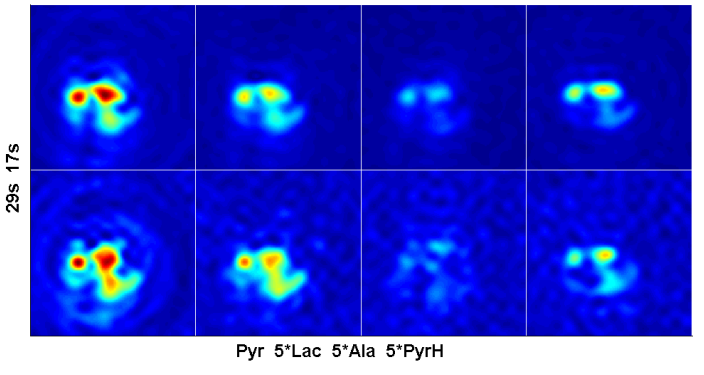


Fig. 7: Hyperpolarised pyruvate and its metabolites: lactate (Lac), alanine (Ala) and pyruvate hydrate (PyrH) 17s (top row) and 29 s (bottom row) after injection

3 Discussion/Conclusion

Different Cartesian acquisition schemes for hyperpolarised ^{13}C were analysed in noise free environment, the results elucidate benefits of applying fast acquisition

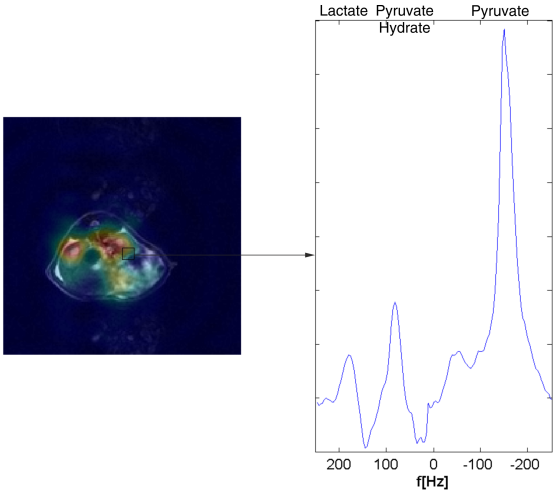


Fig. 8: Left: Pyruvate concentration overlayed on anatomical detailed image. Right: Single voxel spectrum with clearly resolved pyruvate, lactate and pyruvate hydrate peaks and spurious alanine peak

schemes for a ^{13}C DNP-MRSI study. However ^{13}C-CSI sequences are in use because of robustness in the presence of noise.

Assuming *AIF* is Dirac's delta exaggerate disadvantages of slow sequences. In a more realistic situation *AIF* is blurred in time which justify the application of slow sequences. Deep analyzes of effects of different *AIF* is far beyond of scope of this article.

Even faster schemes than *EPSI* i.e. concentric rings was analysed. cCR method was found more robust, with an improved spectral resolution via the inherent sampling of k-space zero in cCR compared to the ncCR method. Further feasibility of using cCR sequence for animal studies was shown. The combination of excessive oversampling of the k-space center and optimal sampling patterns ensures a robust and versatile method, without comprising the irrecoverable hyperpolarisation.

Acknowledgements

The paper is co-founded by the European Union from resources of the European Social Fund. Project PO KL "Information technologies: Research and their interdisciplinary applications", Agreement UDA-POKL.04.01.01-00-051/10-00.

References

1. Nelson, S.J., Kurhanewicz, J., Vigneron, D.B., Larson, P.E., Harzstark, A.L., Ferrone, M., van Criekinge, M., Chang, J.W., Bok, R., Park, I.: Metabolic Imaging of Patients with Prostate Cancer Using Hyperpolarized [1-13C]Pyruvate. **5**(198) (2013) 198ra108—-198ra108
2. Albers, M.J., Bok, R., Chen, a.P., Cunningham, C.H., Zierhut, M.L., Zhang, V.Y., Kohler, S.J., Tropp, J., Hurd, R.E., Yen, Y.F., Nelson, S.J., Vigneron, D.B., Kurhanewicz, J.: Hyperpolarized 13C Lactate, Pyruvate, and Alanine: Noninvasive Biomarkers for Prostate Cancer Detection and Grading. Cancer Research **68**(20) (2008) 8607–8615
3. Koellisch, U., Laustsen, C., Nø rlinger, T.S., Ø stergaard, J.A., Flyvbjerg, A., Gringeri, C.V., Menzel, M.I., Schulte, R.F., Haase, A., Stø dkilde Jø rgensen, H.: Investigation of metabolic changes in STZ-induced diabetic rats with hyperpolarized [1-13C]acetate. Physiological Reports **3**(8) (2015) e12474
4. Mansfield, P.: Spatial mapping of the chemical shift in NMR. Magnetic resonance in medicine : official journal of the Society of Magnetic Resonance in Medicine / Society of Magnetic Resonance in Medicine **1**(3) (1984) 370–386
5. Dutt, A., Rokhlin, V.: Fast fourier transforms for nonequispaced data. SIAM Journal on Scientific computing **14**(6) (1993) 1368–1393
6. Merritt, M.E., Harrison, C., Storey, C., Jeffrey, F.M., Sherry, A.D., Malloy, C.R.: Hyperpolarized 13c allows a direct measure of flux through a single enzyme-catalyzed step by nmr. Proceedings of the National Academy of Sciences **104**(50) (2007) 19773–19777
7. Durst, M., Koellisch, U., Frank, A., Rancan, G., Gringeri, C.V., Karas, V., Wiesinger, F., Menzel, M.I., Schwaiger, M., Haase, A., Schulte, R.F.: Comparison of acquisition schemes for hyperpolarised 13C imaging. NMR in Biomedicine (November 2014) (2015) n/a–n/a
8. Schmidt, R., Laustsen, C., Dumez, J.N., Kettunen, M.I., Serrao, E.M., Marco-Rius, I., Brindle, K.M., Ardenkjaer-Larsen, J.H., Frydman, L.: In vivo single-shot 13C spectroscopic imaging of hyperpolarized metabolites by spatiotemporal encoding. Journal of Magnetic Resonance **240** (2014) 8–15
9. Leupold, J., Må nsson, S., Stefan Petersson, J., Hennig, J., Wieben, O.: Fast multiecho balanced SSFP metabolite mapping of 1H and hyperpolarized 13C compounds. Magnetic Resonance Materials in Physics, Biology and Medicine **22**(4) (2009) 251–256
10. Larson, P.E.Z., Kerr, A.B., Swisher, C.L., Pauly, J.M., Vigneron, D.B.: A Rapid Method for Direct Detection of Metabolic Conversion and Magnetization Exchange with Application to Hyperpolarized Substrates. Journal of Magnetic Resonance **225**(12) (2012) 71–80
11. Durst, M., Koellisch, U., Gringeri, C., Janich, M.a., Rancan, G., Frank, A., Wiesinger, F., Menzel, M.I., Haase, A., Schulte, R.F.: Bolus tracking for improved metabolic imaging of hyperpolarised compounds. Journal of Magnetic Resonance **243** (2014) 40–46
12. Cunningham, C.H., Vigneron, D.B., Chen, A.P., Xu, D., Nelson, S.J., Hurd, R.E., Kelley, D.a., Pauly, J.M.: Design of flyback echo-planar readout gradients for magnetic resonance spectroscopic imaging. Magnetic Resonance in Medicine **54**(5) (2005) 1286–1289
13. Mayer, D., Yen, Y.F., Tropp, J., Pfefferbaum, A., Hurd, R.E., Spielman, D.M.: Application of Sub-Second Spiral Chemical Shift Imaging to Real-Time Multi-Slice

Metabolic Imaging of the Rat In Vivo after Injection of Hyperpolarized 13C1-Pyruvate. Magnetic Resonance in Medicine **62**(3) (2009) 557–564
14. Jiang, W.: Concentric Rings K-space Trajectory for Hyperpolarized C-13 MR Spectroscopic Imaging. (2014)
15. Lauzon, M.L., Rutt, B.K.: Effects of polar sampling in k-space. Magnetic resonance in medicine : official journal of the Society of Magnetic Resonance in Medicine / Society of Magnetic Resonance in Medicine **36**(6) (1996) 940–949
16. Fessler, J.A.: Image reconstruction toolbox. `http://www.eecs.umich.edu/fessler/code`
17. Derby, K., Tropp, J., Hawryszko, C.: Design and evaluation of a novel dual-tuned resonator for spectroscopic imaging. Journal of Magnetic Resonance (1969) **86**(3) (1990) 645–651
18. Yao, Y.X., Pandit, S.: Cramer-Rao lower bounds for a damped sinusoidal process. IEEE Transactions on Signal Processing **43**(4) (1995) 878–885
19. Coleman, T.F., Li, Y.: An interior trust region approach for nonlinear minimization subject to bounds. SIAM Journal on optimization **6**(2) (1996) 418–445
20. Coleman, T., Li, Y.: On the convergence of interior-reflective newton methods for nonlinear minimization subject to bounds. Mathematical Programming **67**(1-3) (1994) 189–224

Modeling Vague Preferences in Recommender Systems

Paweł P. Ładyżyński[1], and Przemysław Grzegorzewski[2,3]

[1] Institute of Computer Science, Polish Academy of Sciences,
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
[2] Systems Research Institute, Polish Academy of Sciences,
ul. Newelska 6, 01-447 Warsaw, Poland
[3] Warsaw University of Technology, Faculty of Mathematics and Information Science,
ul. Koszykowa 75, 00-662 Warsaw, Poland

Abstract.

Efficient analysis of consumer's preferences is a crucial problem in recommender systems. However, in practice we often have to deal with different types of vagueness of the data. Due to the large number of products in the databases, the knowledge of each user is usually incomplete and the ratings are often uncertain (i.e. the same ratings for different products). In this paper we discuss several IF-sets based methods, that are helpful in vague preferences modeling and use full available knowledge about user preferences, to support customers decisions in most appropriate way. We show how to improve algorithms applied in recommender systems using these methods.

We also show some IF-set based modifications of the probabilistic models applied in the instance-based label ranking algorithms, which improve their performance and make them applicable in content-based recommender systems.

Finally, we propose a novel methodology for graphical summarizing of possible recommendations that enables a user to choose such recommendation that fits best to his individual decision-making strategy, e.g. corresponding to his attitude to risk.

1 Introduction

The main goal of a recommender system is to generate meaningful recommendations for items or products that might be interesting for a user. Two basic architectures of recommender systems may be highlighted: content-based filtering (focused on the similarity of items determined by measuring the similarity in their properties) and collaborative filtering systems (focused on the similarity

of items determined by the similarity of the ratings of those items rated by the users). The problem of preference modeling is common in both types of recommender systems. These preference systems in real world applications are usually vague due to the large number of products rated by users with only partial knowledge about the whole set of analyzed items. In this paper we show, that IF-set based model can be a very useful tool while representing vague preferences and can be successfully applied in recommendation algorithms.

In this contribution we also discuss the choice of a recommendation strategy. The majority of methods proposed in the literature, in general, focus on providing the final recommendation to the user. The final recommendation strategy is assumed at the level of designing the recommender system, which means that it must be chosen to fit the individual decision making strategies of the users. This is rather a difficult problem. We can imagine a very common situation when two items are rated by significantly different number of users. The decision how to deal with aggregation of ratings for such item has to be made before providing a user with a final recommendation. Another point concerns the case of different degree of knowledge or experience connected with every user. One may ask, whether we should treat ratings given by the user who has experience with 100 products equally to the rating given by the user who knows only 2 items? We can use i.e. logarithmic weighting to differentiate the impact of more and less competent users, but if a product was rated only by inexperienced users, the overall rating of it, without any additional information, might still be misleading . Providing user with some aggregated value of the level of experience of people who rated this item seems to be a natural solution to this problem. There are also some contributions proposing trust-aware recommendations (see [1],[2]), but these approaches do not take into consideration the experience of the user with respect to his previous products history.

In this paper, we propose some new tools like entropy-based similarity or a graphical method for comparing recommendations, that may be considered interesting and turn out useful in solving mentioned problems. We show how to use this new idea in collaborative filtering, and how to apply it to build computationally efficient predictive models in content-based recommender systems.

2 Modeling preference systems

Let $\mathbb{X} = \{x_1, \ldots, x_n\}$ denote a set of objects (e.g. films, books or other goods). A user A is associated to a vector $R_A = (A_1, \ldots, A_n)$, where A_i describes a position of element x_i among all other elements in $\mathbb{X}$ in his preference system according to objects from $\mathbb{X}$.

If all elements $\{x_1, \ldots, x_n\}$ create the total order (complete information and no ties), then R_A is just a ranking. For example, if we get the following vector $R_A = (1, 3, 4, 2, 5, 6)$ then it means that in A's opinion the most favorite element in the set $\{x_1, \ldots, x_6\}$ is x_1, the next one is x_4, then we have x_2, x_3, x_5, and the worst is x_6.

However, in real databases used in recommender systems, due to the large number of products, users knowledge about all of them is limited and their rankings may be incomplete or some elements may be indistinguishable. Thus, in general, a vector R_A may be not a representation of the total order of n objects. Suppose, e.g., that we get $R_A = (1, 2, 3, \mathrm{NA}, \mathrm{NA}, 2)$. In this case x_1 is the most favorite element. Then observer A indicates two elements x_2 and x_6 but he cannot decide which one of these two objects is better. The next one is x_3 and finally, there are two non-classified elements x_4 and x_5, described by NA, (i.e. "not available"). Further on, we will reserve the word "ranking" only to vectors describing linearly ordered elements. A general case that allows also partial orders we will call a preference system.

There are several methods of dealing with preference systems with missing information and indistinguishable elements. One possibility is omit such data. Appropriate imputation method to transform a vague preference system into a ranking may also be used. The first method leads to loss of information about the amount of knowledge possessed by users, whereas the second may be criticized for unavoidable subjectivity. Thus, we propose to use the model admitting vague preference systems that was proposed by Grzegorzewski (see [3],[4]). This model deals with both well-ordered items, possible ties, missing ranks and non-comparable elements. The key idea in construction proposed in [3],[4] is to represent a vague preference system by the appropriate IF-set. Due to such kind of representation, we can take advantage of the broad apparatus of mathematical methods defined for IF-sets.

Let $\mathbb{U}$ denote a usual set, called the universe of discourse. An IF-set (Atanassov's intuitionistic fuzzy set, see [5]) is given by a set of ordered triples $\tilde{C} = \{(u, \mu_{\tilde{C}}(u), \nu_{\tilde{C}}(u)) : u \in \mathbb{U}\}$, where $\mu_{\tilde{C}}, \nu_{\tilde{C}} : \mathbb{U} \to [0, 1]$ stand for the membership and nonmembership functions, respectively. It is assumed that $0 \leq \mu_{\tilde{C}}(u) + \nu_{\tilde{C}}(u) \leq 1$ for each $u \in \mathbb{U}$.

Consider any finite set of objects $\mathbb{X} = \{x_1, \ldots, x_M\}$. Given any user A let us define two functions $w_x, b_x : \mathbb{X} \to \{0, 1, \ldots, M-1\}$ as follows: for each $x_i \in \mathbb{X}$ let $w_A(x_i)$ denote a number of elements in $\mathbb{X}$ surely worse than x_i, while $b_x(x_i)$ let denote a number of elements surely better than x_i, with respect to the preference related to user A. Next let

$$\mu_{\tilde{A}}(x_i) = \frac{w_A(x_i)}{M-1}, \quad \nu_{\tilde{A}}(x_i) = \frac{b_A(x_i)}{M-1}. \tag{1}$$

denote a membership and nonmembership function, respectively, of the IF-set $\tilde{A} = \{(x_i, \mu_{\tilde{A}}(x_i), \nu_{\tilde{A}}(x_i)) : x_i \in \mathbb{X}\}$ describing the preference system connected with user A.

Using above representation (1), the following vector corresponding to the preference system of one of users $R_1 = (1, 2, \mathrm{NA}, 2, \ 3, \mathrm{NA})$ can be represented in the form of an IF-set where the values for membership function are equal to $\mu_{\tilde{R}_1} = (0.6, 0.2, 0, 0.2, 0, 0)$ and the non-membership function $\nu_{\tilde{R}_1} = (0, 0.2, 0, 0.2, 0.6, 0)$ respectively for elements $x_1, \ldots, x_6$.

3 Preferences in Collaborative Filtering - Finding Similar Users

Measuring similarity between preferences is a crucial problem for collaborative filtering recommender systems. This task becomes significantly harder when preferences are incomplete or somehow vague.

In previous section we have shown the way of modeling preference systems by IF-sets. Since our main goal is to compare preference systems hence one may ask about methods for IF-sets comparison. This topic seems to be interesting not only in our context. Three general types of comparison measures were discussed in [6]: IF-distances, IF-dissimilarities and IF-divergences, and some relationships between them were also examined.

In [7], [8], we discussed the problem of choosing similarity measure between preference systems, with appropriate properties to be applied in collaborative filtering recommender systems. Below we mention the list of requirements that were proposed in [8]:

(C-1) A similarity measure between preference systems A and B takes its maximal value if and only if A and B are perfectly concordant rankings.
(C-2) A similarity measure between preference systems A and B takes its minimal value if and only if A and B are perfectly discordant rankings.
(C-3) A similarity measure between two preference systems A and B is larger than between C and D if and only if a correlation between A and B is stronger than between C and D.

After analyzing several types of similarity measures, we propose two similarity measures with desired properties proved in [8]:

$$S_E(R_1, R_2) = 1 - \sqrt{\frac{3(n-1)}{n(n+1)}} D_E(\tilde{R}_1, \tilde{R}_2) \tag{2}$$

and

$$S_H(R_1, R_2) = 1 - \frac{2(n-1)}{n^2} D_H(\tilde{R}_1, \tilde{R}_2). \tag{3}$$

where

$$D_E(\tilde{R}_1, \tilde{R}_2)^2 = \frac{1}{2} \sum_{i=1}^{n} ((\mu_{\tilde{R}_1}(x_i) - \mu_{\tilde{R}_2}(x_i))^2 + (\nu_{\tilde{R}_1}(x_i) - \nu_{\tilde{R}_2}(x_i))^2), \tag{4}$$

and

$$D_H(\tilde{R}_1, \tilde{R}_2) = \frac{1}{2} \sum_{i=1}^{n} (|\mu_{\tilde{R}_1}(x_i) - \mu_{\tilde{R}_2}(x_i)| + |\nu_{\tilde{R}_1}(x_i) - \nu_{\tilde{R}_1}(x_i)|). \tag{5}$$

Both (2) and (3) reach their maximal values if and only if R_1 and R_2 are perfectly discordant. However, although (2) reaches its minimal value if and only if R_1 and

R_2 are perfectly concordant, (3) reaches its minimal value not only for perfectly discordant preference systems.

According to property (C-3), we compared the behavior of measures (2) and (3) with the generalized Kendall's τ and Spearman's ρ defined in [4], [9]. The generalized Kendall correlation coefficient ([4]) is defined as follows

$$\tilde{\tau} = \frac{1}{2n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} [sgn(\mu_A(x_j) - \mu_A(x_i)) \cdot sgn(\mu_B(x_j) - \mu_B(x_i)) + sgn(\nu_A(x_j) - \nu_A(x_i)) \cdot sgn(\nu_B(x_j) - \nu_B(x_i))], \quad (6)$$

The generalized Spearman coefficient ([9]) is defined by

$$\tilde{r}_s(A, B) = 1 - \frac{3(n-1)}{n(n+1)} \sum_{i=1}^{n} [(\mu_A(x_i) - \mu_B(x_i))^2 + (\nu_A(x_i) - \nu_B(x_i))^2]. \quad (7)$$

In [8], we proved the following property of measures S_E and S_H:

Proposition 1. *Let $A, B, C, D \in \mathbb{IFS}(\mathbb{X})$ describe preference systems with respect to elements of $\mathbb{X} = \{x_1, \ldots, x_n\}$. If $\tilde{r}_s(A, B) \leq \tilde{r}_s(C, D)$, then $S_E(A, B) \leq S_E(C, D)$.*

and the following lemma for measure S_E:

Lemma 1. *Let $A, B, C, D \in \mathbb{IFS}(\mathbb{X})$ describe preference systems with respect to elements of $\mathbb{X} = \{x_1, \ldots, x_n\}$. If $\tilde{r}_s(A, B) < \tilde{r}_s(C, D)$, then $S_E(A, B) < S_E(C, D)$.*

From 1 and 1 we can observe that measures (2), (3) posses desired properties connected with (C-3).

As measures (2), (3) pretend to behave properly in recommender system environment, we can consider the further steps of creating recommendations. The simplest way of recommending a new item to a user A is to find another users (say $B_1, \ldots, B_k$) with preferences similar to A and to suggest A some resources highly preferred by $B_1, \ldots, B_k$ which are yet not known to A.

However, during our experiments, we noticed that the situation where several users have identical preference systems (also no additional products known by some of $B_1, \ldots, B_k$) is quite common. What is more, in [8], we proved the following property of measure S_H:

Proposition 2. *Let R_1 and R_2 denote two preference systems with respect to n objects from the set $\mathbb{Y} = \{x_1, \ldots, x_n\}$. Suppose, that at least one element of $\mathbb{Y}$ got no opinion according to both preference systems R_1 and R_2. Moreover, let R_2^* denote a preference system which is identical to R_2 up to one element $x_i^* \in \mathbb{Y}$ which is ranked according to R_2^* but not considered by R_1 and R_2. Then*

$$0 \leq S_H(R_1, R_2) - S_H(R_1, R_2^*) < \frac{2}{n}. \quad (8)$$

The simple deduction from proposition 2 is, that this measure does not promote users who know a lot about many different products not yet known by A. In fact, we can not prove similar property for measure S_E (it does not hold in general) however, in [8] we show the experimental evaluation based on 10 million randomly generated experiments (different parameters i.e. number of products, fraction of missing values were considered) to analyze the distribution of $(S_E(R_1, R_2) - S_E(R_1, R_2^*))$. The results of our experiment show that for measure S_E, the inequality from Proposition 2 does not hold for less than 2% cases (see fig. 1, 2, detailed description can be found in [8]).

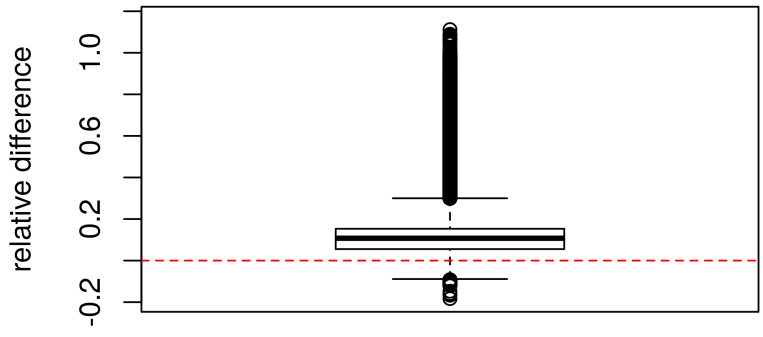


Fig. 1. A simulation for measure S_E. A boxplot for $\mathrm{Diff}(R_1, R_2, R_2^*)$ obtained for 10 millions of random pairs of preference systems, fraction of missing ranks: q=0.5.

Therefore, as we are interested in finding users not only similar to A but who also differ from A in a sense that they could provide an information on items not known yet by A, we had to modify these measures to make them possess the property of promoting those customers similar to a new user, who have a broad knowledge on the items not seen yet by this new user. The general idea is to modify the similarity measures by including some penalty connected with those users whose knowledge is not sufficient. We decided to take an advantage from the two main types of the entropies that appear in the IF-set environment. The first one is connected with the fuzziness of given IF-set, while the second with the hesitancy and the lack of knowledge connected with this IF-set (see i.e. [10], [11]). We use the following definition of two-tuple entropy (see [11]):

Definition 1. *Let* $E_F, E_{HLK} : \mathbb{IFS}(\mathbb{X}) \to [0,1]$ *denote two mappings. A pair* (E_F, E_{HLK}) *is said to be a two-tuple entropy if* E_F *and* E_{HLK} *satisfy the following conditions:*

(i) $E_F(A) = 0$ *if and only if* A *is crisp or* $\mu_A(x) = \nu_A(x) = 0$ *for every* $x \in A$,
(ii) $E_F(A) = 1$ *if and only if* $\mu_A(x) = \nu_A(x) = 0.5$ *for every* $x \in A$,

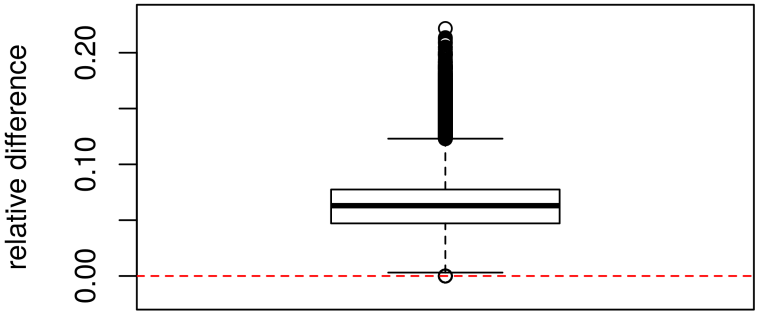


Fig. 2. A simulation for measure S_E. A boxplot for $\mathrm{Diff}(R_1, R_2, R_2^*)$ obtained for 10 millions of random pairs of preference systems, fraction of missing ranks: q=0.3.

(iii) $E_F(A) = E_F(A^C)$, *where* $A^C = \{(x, \nu_A(x), \mu_A(x)) : x \in \mathbb{X}\}$ *is the complement of* A,
(iv) $E_F(A) \leq E_F(B)$ *if* $\mu_A(x) \leq \mu_B(x) \leq 0.5$ *and* $\nu_A(x) \geq \nu_B(x) \geq 0.5$ *for* $\mu_B(x) \leq \nu_B(x)$ *or if* $\mu_A(x) \geq \mu_B(x) \geq 0.5$ *and* $\nu_A(x) \leq \nu_B(x) \leq 0.5$ *for* $\mu_B(x) \geq \nu_B(x)$,
(v) $E_{HLK}(A) = 0$ *if and only if* $A \in FS(\mathbb{X})$,
(vi) $E_{HLK}(A) = 1$ *if and only if* $\mu_A(x) = \nu_A(x) = 0$,
(vii) $E_{HLK}(A) = E_{HLK}(A^C)$,
(viii) $E_{HLK}(A) \geq E_{HLK}(B)$ *if* $\mu_A(x) + \nu_A(x) \leq \mu_B(x) + \nu_B(x)$ *for every* $x \in \mathbb{X}$.

It is seen that E_F is strictly related to fuzziness while E_{HLK} is connected with the hesitancy and the lack of knowledge. In [8], we propose to use the following expression:

$$E_{HLK}(A) = \frac{1}{n} \sum_{i=1}^{n} [1 - (\mu_A(x_i) + \nu_A(x_i))], \tag{9}$$

and we show how to decompose it into two parts where one is connected with lack of knowledge and would later be used as a penalty for users whose knowledge is not rich enough:

Proposition 3. *Let* $\tilde{R} \in \mathbb{IFS}(\mathbb{X})$ *describe a preference system* R *with respect to* n *objects from the set* $\mathbb{X} = \{x_1, \ldots, x_n\}$. *Suppose that* t *different ranks* $(1 < t \leq n)$ *were attributed to elements of* $\mathbb{X}$ *in such way that* k_i *denote the number of elements which obtained* i*-th rank according to* R. *Moreover, let* m *denote the number of objects in* $\mathbb{X}$ *not ranked according to* R, *where* $m + \sum_{i=1}^{t} k_i = n$. *Then*

$$E_{HLK}(\tilde{R}) = E_H(\tilde{R}) + E_{LK}(\tilde{R}), \tag{10}$$

where

$$E_H(\tilde{R}) = \frac{1}{n}\sum_{i=1}^{t}\frac{k_i(k_i-1)}{n-1}, \tag{11}$$

$$E_{LK}(\tilde{R}) = \frac{1}{n}\left[\frac{(n-m)m}{n-1}+m\right]. \tag{12}$$

In proposition 12, E_{LK} quantifies the lack of knowledge impact connected with unavailable opinions on evaluated objects. Thus we proposed in [8] the following form of modified S_E measure, including some penalty for not sufficient knowledge about many different products:

$$S_E^{pen}(R_1, R_2) = S_E(R_1, R_2)\cdot(1 - E_{LK}(\tilde{R}_2)). \tag{13}$$

Now, the proposed measure (13) promotes, in the process of finding users similar to A due to their preference systems, users with broad knowledge about many products. Such tool is quite satisfactory to be used in collaborative filtering. The next important step of creating recommendation is reasoning from multiple preference systems of several users, which are the most similar to the new user A due to measure S_E^{pen}. The question how to aggregate different preferences about product not yet known by A is not trivial. Formally, we can express the preferences of chosen users, say $B_1, \ldots, B_k$, represented by appropriate IF-sets $\tilde{B}_1, \ldots, \tilde{B}_k$, in the form of their membership and non-membership functions $(\mu_{\tilde{B}_j}(x_1), \mu_{\tilde{B}_j}(x_2), \ldots, \mu_{\tilde{B}_j}(x_n))$ and $(\nu_{\tilde{B}_j}(x_1), \nu_{\tilde{B}_j}(x_2), \ldots, \nu_{\tilde{B}_j}(x_n))$ respectively for $j = 1, \ldots, k$.

Further performance of the recommender system depends strongly on the choice of the aggregation method for these preferences and the final recommendation strategy. What is more, the strategy of decision making may be very specific for different group of users, which may affect the overall accuracy of the system. Exemplary "ready to use" algorithms of creating final recommendation were mentioned in [8]. However, we also proposed in [8] the new idea of a graphical tool that summarizes properties of possible recommendations and may be used in the form of interaction with users to let them choose the recommendation which best fits their characteristic of individual decision making strategy. It may also be helpful for the designers of fully automatic recommender systems to analyze different possible algorithms and choose the one, that is fitted to the specificity of the group of users they consider. The idea of that graph is to provide the user with a value of a special score function calculated for different items together with some information on the strength (or credibility) of the score. Exemplary results of summarizing two possible recommendations using proposed method can be seen in fig. 3.

On the vertical axis of Figure 3 we place the aggregated values of μ and ν functions in the form of interval valued fuzzy set, i.e. $[\mu^L_{agg}(x_i), \mu^R_{agg}(x_i)]$, where $\mu^L_{agg}(x_i) = \mu^A_{agg}(x_i)$ and $\mu^R_{agg}(x_i) = 1 - \nu^A_{agg}(x_i)$ (since IF-sets are isomorphic

with interval-valued fuzzy sets, see e.g., [12]), where

$$\mu_{agg}^{A,k}(x_i) = \frac{1}{\sum_{j=1}^{k} \mathbb{I}(\mu_{\tilde{B}_j}(x_i) \vee \nu_{\tilde{B}_j}(x_i) > 0)} \sum_{j=1}^{k} \mu_{\tilde{B}_j}(x_i), \tag{14}$$

$$\nu_{agg}^{A,k}(x_i) = \frac{1}{\sum_{j=1}^{k} \mathbb{I}(\mu_{\tilde{B}_j}(x_i) \vee \nu_{\tilde{B}_j}(x_i) > 0)} \sum_{j=1}^{k} \nu_{\tilde{B}_j}(x_i), \tag{15}$$

and $\mathbb{I}(\cdot)$ denotes the indicator.

This interval can be interpreted as the aggregated degree of membership to the set of highly preferred items. Due to the different level of knowledge of several users, the information they provide about products they already know may be more or less precise. The general interpretation is that the thinner is the interval, the more experienced users (with knowledge about many different products) rated the product and thus, the recommendation is more trustful. Quite a different aspect is the confidence of the recommendation. Even if the product was rated by experienced users, but only a small number of them, we can suspect that the recommendation does not have appropriate level of confidence. Thus, on the horizontal axis, we present the fraction of nearest neighbors that have any knowledge about product presented in the graph (in our example 0.4 NN know product x_1 and 0.8 of NN know product x_2). In fig. 3, we can observe that x_2 is more confident recommendation, which is also known by users with higher amount of knowledge than x_1 (thinner interval). It is worth noticing that the interval reduces to the single point if and only if this item is rated by all users "similar" to our customer, it is considered on the same position by all of them and they have full knowledge about all products in the database. On the other hand, product x_1, can still be potentially the best product in our data set ($\mu_{agg}^{R}(x_1)$=1 means that due to the users that know x_1, none of other products is better than it). Considering this example, the optimistic person with low aversion to risk would probably choose product x_1 (due to available knowledge), whereas users, who need more confident recommendations, would choose x_2.

In [8], we show the experimental results of evaluation of collaborative filtering recommender system based on similarity measure (13). We analyzed 3 possible different strategies of decision making and the simulated user interaction strategy with usage of proposed graphical tool. The results were promising, so in this contribution, we decided to apply this method in content based recommender system, and compare the results with one of well performing algorithms we proposed in [13].

4 Modeling Preferences in Content Based Recommender Systems

We will now focus on creating recommendations in a different situation, where some meta-data about users of a recommender systems are available. Let $\mathbb{U}$,

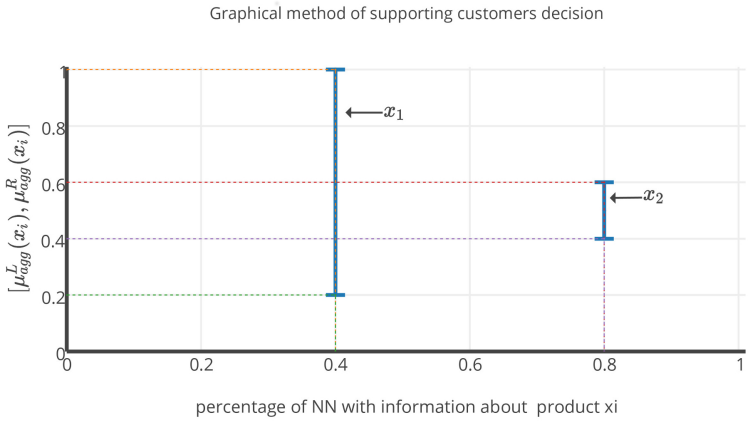


Fig. 3. Graphical method for supporting customer's decision

called an instance space, denote a set of elements (users, patients etc.) characterized by several attributes. Suppose that instead of classifying instances into separate classes, we associate each instance $u \in \mathbb{U}$ with a total order of all class labels $\mathbb{Y} = \{y_1, \ldots, y_M\}$. Moreover, we say that $y_i \succ_u y_j$ indicates that y_i is preferred to y_j given the instance u. A total order $\succ_u$ can be identified with a permutation π_u of the set $\{1, \ldots, M\}$, where $\pi_u(i)$ is the index j of the class label y_j put on the i-th position in the order. The class of permutations of $\{1, \ldots, M\}$ will be denoted by Ω.

The main goal required while creating recommendation is to predict a ranking of labels $y_1, \ldots, y_M$ for a new instance u, given some instances with known rankings of labels as a learning set. In practical issues, especially in recommender systems where the amount of available products is large, preference on instances known from the learning set does not usually contain all labels, i.e our information is of the form $y_{\pi_u(1)} \succ_u \ldots \succ_u y_{\pi_u(k)}$, where $k < M$.

Several methods are available in the literature but many of them are computationally exhaustive in a presence of vague data. In [13], we proposed the IF-set modification of an algorithm based on the Mallows model. The proposed modification effected in significant improvement in performance.

Below, we highlight some important details of a mentioned algorithm.

We may assume that every instance is associated with a probability distribution over Ω, i.e. for each instance $u \in \mathbb{U}$ there exists a probability distribution $\mathbb{P}(\cdot|u)$ such that, for every $\pi \in \Omega$, $\mathbb{P}(\pi|u)$ is the probability that $\pi_u = \pi$.

To evaluate the predictive performance of a label ranker a suitable loss function on Ω is needed, e.g. based on Kendall's tau (see [14]).

The Mallows model is a distance-based probability model defined by

$$\mathbb{P}(\pi|\theta,\pi_0) = \frac{exp(-\theta D(\pi,\pi_0))}{\phi(\theta)}, \tag{16}$$

where the ranking $\pi_0 \in \Omega$ is the location parameter (center ranking), D is a distance measure on rankings, $\phi = \phi(\theta)$ is a constant normalization factor and θ stands for a spread parameter which determines how quickly the probability decreases with the increasing distance between π and π_0.

The main idea of our modification proposed in [13] is to replace measure D in (16) with a substitute that admits vague data.

Having any two instances $u_1, u_2 \in \mathbb{U}$ we may compute a correlation between preference systems $\tilde{u}_1, \tilde{u}_1$ generated by these instances, using the generalized Kendall's tau, admitting incomplete preferences (see [4]):

$$\tilde{\tau} = \frac{1}{2M(M-1)} \sum_{i=1}^{M}\sum_{j=1}^{M}[sgn(\mu_{\tilde{u}_1}(y_j) - \mu_{\tilde{u}_1}(y_i)) \cdot sgn(\mu_{\tilde{u}_2}(y_j) - \mu_{\tilde{u}_2}(y_i)) \tag{17}$$
$$+sgn(\nu_{\tilde{u}_1}(y_j) - \nu_{\tilde{u}_1}(y_i)) \cdot sgn(\nu_{\tilde{u}_2}(y_j) - \nu_{\tilde{u}_2}(y_i))].$$

For possibly incomplete preferences we get incomplete permutation $\tilde{\pi} = \tilde{\pi}_u$ which might be identified with the corresponding IF-set $\tilde{u}$. Thus for any two instances $u_1, u_2 \in \mathbb{U}$ we have $\tilde{\tau} = \tilde{\tau}(\tilde{u}_1, \tilde{u}_2) = \tilde{\tau}(\tilde{\pi}_1, \tilde{\pi}_2)$. Hence, using (17), we may consider the following measure

$$D_{\tilde{\tau}}(\tilde{\pi}_1, \tilde{\pi}_2) = \frac{1 - \tilde{\tau}(\tilde{\pi}_1, \tilde{\pi}_2)}{2}, \tag{18}$$

which seems to be useful in the generalized Mallows model (16) admitting incomplete rankings and defined as follows

$$\tilde{\mathbb{P}}(\tilde{\pi}|\theta,\tilde{\pi}_0) = \frac{exp(-\theta D_{\tilde{\tau}}(\tilde{\pi},\tilde{\pi}_0))}{\phi(\theta)}. \tag{19}$$

Of course, when modeling preferences by IF-sets one can also consider other substitutes for the measure D in (16), including different distances, dissimilarity measures or divergences (see, e.g., [6]). However, we have chosen a measure based on the generalized Kendall's tau because it is common to use distances utilizing the classical Kendall's coefficient in the Mallows model (see, e.g., [14]).

One of proposed in [13] algorithms can be described as follows:

Algorithm 1 ***Mallows Best Probability Algorithm (MBP)***
{***Input****: u - new instance, U - learning set of instances, $\bar{\pi}$ - permutations of labels connected with instances, k - number of nearest neighbors*}
1. Find k nearest neighbors of u in U.
2. For (j in $1:M$) calculate $\sum_{\pi^* \in \bar{\pi}_{kNN(u)}} \tilde{\mathbb{P}}(y_j^{best}|\theta,\pi^*)$
3. MBP-rank $<-$ Sort labels according to the values obtained in step 2 (in case of ties a label with lower index is better in the ranking).
{***Output****: MBP-rank*}

where

$$\tilde{\mathbb{P}}(y_j^{best}|\theta,\pi^*) = \frac{exp(-\theta D^*(y_j^{best}, y_j^{\pi^*}))}{\phi(\theta)}, \tag{20}$$

where D^* is the Euclidean distance between IF-sets given by

$$D^*(y_j^{best}, y_j^{\pi^*}) = \sqrt{\frac{1}{2}\sum_{i=1}^{n}((\mu_{y_j^{best}} - \mu_{\pi^*}(y_j))^2 + (\nu_{y_j^{best}} - \nu_{\pi^*}(y_j))^2)}. \tag{21}$$

as we apply the Mallows model to express the probability corresponding to the best label.

The performance of proposed algorithm we show in Tables 1, 2 (see. [13]) for details).

Table 1. Comparison of label ranking algorithms for $p = 30\%$ missing labels in the learning set.

	accuracy			time [s]		
data set	IBLR	MBP	MMBP	IBLR	MBP	MMBP
glass (A)	0.781	0.784	0.788	3.504	0.26	3.7
vowel (A)	0.817	0.795	0.819	102.03	1.05	102.26
housing (B)	0.670	0.665	0.670	8.44	0.70	8.95
elevators (B)	0.622	0.617	0.624	1371.86	225.83	1583.55
wisconsin (B)	0.432	0.420	0.427	316.12	0.40	319.54
average	0.664	0.656	0.665	360.39	45.65	403.60

Table 2. Comparison of label ranking algorithms for $p = 50\%$ missing labels in the learning set.

	accuracy			time [s]		
data set	IBLR	MBP	MMBP	IBLR	MBP	MMBP
glass (A)	0.688	0.685	0.687	5.12	0.29	5.42
vowel (A)	0.725	0.700	0.715	119.84	0.95	126.04
housing (B)	0.579	0.570	0.573	12.53	0.7	13.12
elevators (B)	0.540	0.530	0.535	2326.23	272.67	2598.56
wisconsin (B)	0.381	0.351	0.363	502.22	0.37	508.74
average	0.583	0.567	0.575	593.19	55.00	650.38

Results given in Table 1 and Table 2 show that algorithms MBP, MMBP and IBLR have similar accuracy on our experimental sets. More precisely, MBP is

usually slightly worse than the two other algorithms, but it is significantly faster which is crucial due to applications in recommender systems.

4.1 Fitting decision strategy to the user

One may notice that the decision making strategy in mentioned MBP algorithm is a strategy that we choose in the process of designing the algorithm. It gives satisfactory overall results, but the rule how to choose final recommendation is assumed without any survey about the users of a recommender system.

In [8], we considered three common strategies of decision making based on proposed graphical tool:

- Strategy 1: choose the product with μ_R^{agg} not lower than 0.5 with maximal p_{nn} and minimal difference between μ_R^{agg} and μ_L^{agg} - this strategy would fit users with the strongest aversion to risk. Taking into consideration highest p_{nn} and lowest size of the interval $[\mu_L^{agg}, \mu_R^{agg}]$ is equivalent to base the recommendation on decision of the largest possible number of the most experienced users (with the highest level of knowledge about many different products).
- Strategy 2: choose the product with maximal μ_R^{agg} - is a kind of the risky, optimistic strategy, that takes into consideration only the highest possible value of the μ function for the unknown product.
- Strategy 3: choose the product with maximal μ_L^{agg} - is a kind of the pessimistic strategy, that maximizes only the lowest possible value of the μ function for the unknown product.

We will now adapt the proposed graphical tool for valuating the possible recommendations. We will show what improvement can be obtained by using different strategies of creating recommendation and we will compare the results with MBP algorithm.

To perform our experiments, we use semi-synthetic label ranking datasets downloaded from `www.cs.uni-paderborn.de/fachgebiete/intelligente-systeme/software/label-ranking-datasets.htm`. Below, we extend our experimental evaluation, presented in [8], to analyze the behavior of discussed methodology (see [8] for more detailed description). This time, as we consider content-based recommendations, we combine preference systems and vector of attributes from *wisconsin* and *vowel* datasets randomly (assuming that labels and attributes from wisconsin dataset are always after labels and attributes from vowel dataset) to obtain dataset containing 528 instances with 37 attributes, where each instance is connected with ranking of 27 labels. We then generate the missing ranks (every element in each ranking is removed with probability 0.5). To analyze the performance and behavior of proposed method, we use leave-one-out cross-validation. For every observation A, we first find it's 5 nearest neighbors using all 37 attributes. After finding the set of nearest neighbors, we specify the IF-set representations of their preference systems and the aggregated values of μ and ν functions for every label using the formulas (14,15) are calculated. Having

all statistics to use proposed graphical method and perform strategies $(1-3)$, we begin our experiment, taking into consideration also the best recommendation from MBP algorithm. The accuracy is calculated using the following formula:

$$acc_A(y_i) = 1 - \frac{1}{2}\sqrt{(\mu_A^*(y_i) - \mu_L^{agg}(y_i))^2 + (\nu_A^*(y_i) - (1 - \mu_R^{agg}(y_i)))^2}, \quad (22)$$

where $\mu_A^*(y_i)$ and $\nu_A^*(y_i)$ are the values obtained by representing the preference system for a given user A in a form of IF-set after inputing the true rank for label y_i (from the learning set before the process of random removing the ranks) into his preference system.

Results are presented in Table 3. The simulation of the user interaction is obtained by choosing the strategy with the highest accuracy for every user and compare the results with the case when the recommendation strategy is fixed for every user. As MBP is rather too complicated strategy to be adapted individually by the user, it is not included in "simulated user strategy".

Table 3. Values of averaged accuracy of recommendations for different recommendation strategies. "Acc simulated user" means the averaged accuracy of recommendation based on the best possible strategy for each user.

acc str1	acc str2	acc str3	acc MBP	acc simulated user
0.9133	0.8461	0.9056	0.9144	0.9343

We may notice that the simulation of user-interactive strategy gives the best results.

The results of analogical experiment in collaborative filtering context can be seen in Table 4 (see [8] for details).

Table 4. Values of averaged accuracy of recommendations for different recommendation strategies. "Acc simulated user" means the averaged accuracy of recommendation based on the best possible strategy for each user.

acc str1	acc str2	acc str3	acc simulated user
0.9047	0.8884	0.8967	0.9140

An interesting observation is the difference between accuracy for the same strategies with and without the information brought from the meta-data connected with our instances. The difference in the accuracy between the worst and the best strategy increases when we use the meta-data.

5 Conclusions and Future Work

In this paper we discuss several IFset based methods of dealing with vague preferences in different types of recommender systems. We show how the graphical tool of summarizing recommendations proposed in [8] can be applied to improve accuracy of content based recommender systems.

Some questions remain open and deserve future research. In particular, a natural desired extension of the proposed graphical method for comparing recommendations would be its special implementation in the form of the automatic user-adaptive algorithm. Such algorithm would learn user's decision-making strategy in order to propose him the appropriate recommendations automatically, even without his influence. We can imagine at least two types of data that can be used to train such algorithm. In the first case, the algorithm would learn the behavior of the user from his historical choices, e.g. from the proposals of selected recommendation presented in a graphical form. The second approach requires some meta-data connected with the user, like results of a psychological survey concerning his behavior in decision-making. Basing on such data we could deduce whether the user prefers something risky but with possible highest rates or a medium rated but well checked product.

Concerning the content based recommender systems, although the proposed MBP algorithm seems to be a promising candidate for creating recommendations especially in the presence of large number of labeled items, one may consider application of the proposed graphical tool to design global predictive models that would be more computationally efficient in the prediction step. The proposed method based on finding nearest neighbors is rather exhaustive when we have to deal with large databases of users. In this case consideration of i.e. GLM based models for estimating borders of intervals presented in proposed graphical method, seem to be desired extension of proposed algorithms.

Acknowledgments

Study was supported by research fellowship within "Information technologies: research and their interdisciplinary applications" project co-financed by European Social Fund (agreement no. POKL.04.01.01-00-051/10-00). [15]

References

1. Martinez-Cruz C., Porcel C., B.M.J.H.V.E.: A model to represent users trust in recommender systems using ontologies and fuzzy linguistic modeling. Information Sciences **311** (2015) 102–118
2. Moradi, P., A.S.: A reliability-based recommendation method to improve trust-aware recommender systems. Expert Systems with Applications **42** (2015) 7386–7398
3. Grzegorzewski, P.: The coefficient of concordance for vague data. Computational Statistics & Data Analysis **51** (2006) 314–322

4. Grzegorzewski, P.: Kendall's correlation coefficient for vague preferences. Soft Computing **13** (2009) 1055–1061
5. Atanassov, K.: Intuitionistic fuzzy sets: Theory and applications. Springer-Verlag (1999) 55–62
6. Montes S., Iglesias T., J.V..M.I.: A common framework for some comparison measures of if-sets. IWIFSGN 2012 (2012)
7. P.P. Ł adyżyński, P. Grzegorzewski, .: Comparing vague preferences in recommender systems. Proceedings of the 8th conference of the EUROFUSE 2013 Workshop on Uncertainty and Imprecision Modelling in Decision Making (2013) 149–156
8. P.P. Ł adyżyński, P. Grzegorzewski, .: Vague preferences in recommender systems. Expert Systems with Applications (2015) In press
9. Ziembińska, P.G..P.: Spearman's rank correlation coefficient. LNAI **7022** (2011) 342–353
10. E., G.P..M.: Some notes on atanassov's intuitionistic fuzzy sets. Fuzzy Sets and Systems **156** (2005) 492–495
11. Pal N.R., Bustince H., P.M.M.U.G.D..B.G.: Uncertainties with atanassov's intuitionistic fuzzy sets: Fuzziness and lack of knowledge. Information Sciences **228** (2013) 61–746
12. P. Grzegorzewski, M.E.: On the entropy of intuitionistic fuzzy sets and interval valued fuzzy sets. Proceedings of the Tenth International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (2004) 1419–1426
13. P.P. Ł adyżyński, P. Grzegorzewski, .: On incomplete label ranking with if-sets. Strengthening Links Between Data Analysis and Soft Computing, Springer (2014) 55–62
14. W. Cheng, K. Dembczynski, E.H.: Decision tree and instance-based learning for label ranking. ICML (2009)
15. W. Cheng, E.H.: A nearest neighbor approach to label ranking based on generalized labelwise loss minimization. International Joint Conference on Artificial Intelligence (IJCAI-13) (2013)

On the Use of BOWA Operators in Cluster Analysis for Collaborative Filtering

Hanna Łącka

Institute of Computer Science, Polish Academy of Sciences
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland

Abstract. In this paper a construction of a similarity measure between groups of rankings, based on a Bipolar OWA (BOWA) function is discussed. The measure possesses some interesting properties that make it useful in cluster analysis performed as a part of collaborative filtering process. An extended data representation model for consumer preferences and objects of their preferences is assumed. Practical issues of conducting and evaluating the clustering procedure are discussed.

Keywords: Aggregation function, association measure, bipolarity, cluster analysis, collaborative filtering, ordering, OWA operator, preferences, ranks, rating, recommender system.

1 Introduction

The most common web-based recommender systems predict what movies, books or other goods a user would prefer, based on the historical ratings, views or purchases of the user [1, 2]. Explicit user feedback once given continues to be useful. The most popular setting in which preferences are represented in such systems is a matrix (sometimes called a *utility matrix*) with rows corresponding to users, columns corresponding to items and cells containing values of ratings given to items by the users. In [3] a more complicated setting of representing user preferences was proposed. It is especially suited for recommending services, e.g. vacation trips, cultural events, conferences, for which no explicit feedback exists. The setting has a form of a matrix where the entries of the cells for each user-attribute pair contain rankings. Each ranking expresses user preference for items belonging to the domain of a given attribute.

Collaborative filtering approach is the most common technique successfully applied in recommender systems [4, 5]. It creates item recommendations based on similarity measures between users and/or items. An application of cluster analysis and grouping the users based on their similarity was considered a natural and interesting direction of inference about preferences.

In order to apply this approach for the assumed data representation, a similarity measure between groups of rankings [3] coming from a pair of users was

defined. The measure is based on a function that is a member of the family of BOWA operators (Bipolar Ordered Weighted Averaging function) proposed in [6], which are used to aggregate bipolar data.

In this paper a work on the similarity measure between groups of rankings is summarized and the application of the measure in cluster analysis is proposed. The clustering is assumed to be a part of a collaborative filtering process whose purpose is to detect natural similarity groups of consumers, as opposed to a possible segmentation approach [7], and to generate object recommendations for the consumers.

The paper is organized as follows. An introductory example explaining the difference between the classical situation considered in recommender systems and the suggested setting is given in Section 2. In Section 3 a data representation model for consumers is recalled. We also propose a data representation for objects of consumer preferences, based on groups of vectors and referring to consumer data representation model. A notion of a *history of choices* of a consumer is introduced. Section 4 recalls definition of the chosen similarity measure between groups of rankings and of the family of operators the measure is based on. In Section 5 we propose to apply the defined similarity measure in cluster analysis and describe example choice of existing methods to conduct and evaluate it.

2 Introductory example

Let us consider a travel agency, that gathers a history of vacation trips of its clients. For every client a number of times he or she attended a certain trip is known.

The agency stores data about trips in the following way. Every trip is described by the same set of attributes. For each separate attribute the availability of its all possible variants (which depend on a domain of the attribute) is marked. An exemplary output of trips (or trip types) data set is given in Table 1, where 1 means a certain variant is available and 0 it is not.

Data about clients are stored in a form of rankings of choices made by the clients. For each attribute of a trip, all available variants concerning this aspect of a trip have the ranks assigned, according to historical preferences of the client. In other words, for each client a ranking of variants of a given attribute from the most preferred to the least preferred variant is obtained. Table 2 presents an exemplary output of the clients data set, where numbers indicate the ranks assigned.

Such data sets as shown in Table 1 and Table 2 are collected because the agency plans to prepare new trip offers for each client. To maximize the possibility of accepting a new offer it should be prepared in a way that guarantees client's satisfaction when chosen. To achieve this, the agency plans to create recommendations based on similarity between clients and/or trip offers.

Table 1. Exemplary data set of trips.

	Accommodation	Means of transport	Activities
Trip T	tent - 0 guesthouse - 1 hotel - 0	car - 1 bus - 0 train - 1 airplane - 0	sunbathing - 1 sightseeing - 0
Trip U	tent - 1 guesthouse - 0 hotel - 0	car - 1 bus - 0 train - 0 airplane - 1	sunbathing - 1 sightseeing - 1
...	...	...	...

Table 2. Exemplary data set of clients' preferences.

	Accommodation	Means of transport	Activities
Client A	tent - 2 guesthouse - 1 hotel - 3	car - 1 bus - 4 train - 2 airplane - 3	sunbathing - 1 sightseeing - 2
Client B	tent - 3 guesthouse - 2 hotel - 1	car - 3 bus - 4 train - 1 airplane - 2	sunbathing - 2 sightseeing - 1
...	...	...	...

3 Data representation

Let $\mathcal{Y}$ be a finite set of attributes of size n. Moreover, assume that $\mathcal{U}_j$ is a domain of the attribute $Y_j \in \mathcal{Y}$ which consists of l_j variants, i.e. $\mathcal{U}_j$ is a finite set of size l_j, where $j = 1, \ldots n$.

Let $\mathcal{X}$ denote a set of consumers. For each attribute consumer preferences are expressed with respect to all available variants of that attribute. Hence, for any consumer $A \in \mathcal{X}$ we get n rankings corresponding to successive attributes, so the observation related to A might be perceived as a vector

$$\mathbb{R}_A = [R_{A1}, R_{A2}, \ldots, R_{An}], \tag{1}$$

where R_{Aj} is a ranking of variants belonging to the domain of the j-th attribute.

Consider now a ranking R_{Aj}. Since it reflects the consumer's preferences on variants belonging to the domain $\mathcal{U}_j$ of the attribute $Y_j \in \mathcal{Y}$, it is also a vector. Namely,

$$R_{Aj} = (r_{Aj}^{(1)}, r_{Aj}^{(2)}, \ldots, r_{Aj}^{(l_j)}), \tag{2}$$

where $r_{Aj}^{(k)}$, $k = 1, \ldots, l_j$ is a rank assigned to k-th variant belonging to $\mathcal{U}_j$ and where l_j stands for the size of the domain $\mathcal{U}_j$.

Next, let $\mathcal{Z}$ be a set of objects of preference (products, offers, events etc.). Each object $T \in \mathcal{Z}$ is characterized by available variants of the attributes $\mathcal{Y}$ already considered by consumers. Hence, object $T \in \mathcal{Z}$ is described by a vector

$$\mathbb{V}_T = [V_{T1}, V_{T2} \ldots, V_{Tn}] \tag{3}$$

where V_{Tj} is an l_j-element vector indicating available variants, i.e.

$$V_{Tj} = (v_{Tj}^{(1)}, v_{Tj}^{(2)}, \ldots, v_{Tj}^{(l_j)}), \tag{4}$$

where $v_{Tj}^{(k)} \in \{0, 1\}$, $k = 1, \ldots, l_j$, and $v_{Tj}^{(k)} = 1$ denotes that the k-th variant in $\mathcal{U}_j$ is available (in offer T), while $p_{Aj}^{(k)} = 0$ means that it is not available. In general there is no restriction on the number of simultaneously available variants.

Given a finite set of considered objects $\{T^{(1)}, T^{(2)}, \ldots, T^{(t)}\} \in \mathcal{Z}$ and a consumer $A \in \mathcal{X}$, let

$$H_A = [h_{A1}, h_{A2}, \ldots, h_{At}] \tag{5}$$

denote a consumer A's *history of choices*, where $h_{Ai} \in \{0, 1, \ldots, t\}$ for $i = 1, \ldots, k$, and $h_{Ai} = 0$ means the i-th object from the set $\{T^{(1)}, T^{(2)}, \ldots, T^{(t)}\}$ was never chosen by A, while $h_{Ai} \in \{1, \ldots, t\}$ denotes a rank assigned to the i-th object. For the most often chosen object we obtain $h_{A\cdot} = 1$, the second most often chosen object has rank 2, and so on till the least often chosen object.

We assume the client A's representation (1) is linked to the history of choices H_A in the following way. Observation $\mathbb{R}_A = [R_{A1}, R_{A2} \ldots, R_{An}]$ is generated on the basis of: a history of choices H_A and an additional information about exact number of times each object from the history was chosen. For a given j-th attribute we obtain a ranking R_{Aj} by summing up the number of times each of the l_j variants, if available, was chosen in the history and assigning ranks to each of the l_j obtained sums in the non-increasing order.

Example 1. Let $\{T, U, V\}$ be a considered set of objects of preference, such that

$$\begin{aligned}
\mathbb{V}_T &= [(0, 1, 0), (1, 0, 1, 0), (1, 0)] \\
\mathbb{V}_U &= [(1, 0, 0), (1, 0, 0, 1), (1, 1)] \\
\mathbb{V}_V &= [(1, 0, 0), (1, 1, 1, 1), (0, 1)].
\end{aligned}$$

Given a client A, his or her history of choices $H_A = [1, 2, 0]$ and additional information that the first object was chosen 13 times and the second object 4 times by the client A, we obtain the following vector of sums for each variant of each attribute:

$$\begin{aligned}
&[(1 \cdot 4, 1 \cdot 13, 0), (1 \cdot 4 + 1 \cdot 13, 0, 1 \cdot 13, 1 \cdot 4), (1 \cdot 13 + 1 \cdot 4, 1 \cdot 4)] = \\
&= [(4, 13, 0), (17, 0, 13, 4), (17, 4)].
\end{aligned}$$

After assigning ranks to the sums in the non-increasing order, we obtain client A's representation:

$$\mathbb{R}_A = [(2,1,3),(1,4,2,3),(1,2)].$$

□

4 Measure of association between groups of rankings

In order to group the consumers based on their similarity, for the assumed consumer data representation (1), a measure of association between two groups of rankings was searched for. A set of requirements which a measure should satisfy was specified [3] and the form of the desired measure between two groups of rankings corresponding to consumers A and B, $A, B \in \mathcal{X}$, was stated as

$$S(A,B) = F(s^1_{AB}, s^2_{AB}, \ldots, s^n_{AB}), \tag{6}$$

where $(s^1_{AB}, s^2_{AB}, \ldots, s^n_{AB})$ is a vector of pairwise correlations obtained for all attributes under study for two consumers $A, B \in \mathcal{X}$, i.e. $s^j_{AB} = s(R_{Aj}, R_{Bj})$, $j = 1, \ldots, n$, s denotes any pairwise correlation measure between two rankings, taking values in $[-1,1]$ (e.g. Kendall's τ or Spearman's r_S [8]) and $F : [-1,1]^n \to [-1,1]$ is a suitable function.

Since the goal of F is to aggregate several correlations to a single value, one may expect that it should be an appropriate aggregation function. The preservation of bounds property of any aggregation function coincides with the specified requirement that the measure should take its maximal (minimal) value when all rankings are pairwise perfectly concordant (discordant). One of the other requirements was to reward higher correlations, hence an OWA operator [9] might seem a good choice. However, the reward was postulated to be given regardless of the correlation signs. Hence, F cannot be monotone on the whole interval $[-1,1]$ and cannot fulfill the monotonicity condition (see, e.g., [10–12]) of any aggregation function.

A new family of semi-aggregation operators was therefore proposed [6]. It is a generalization of OWA operators for the case of bipolar data and, most importantly, was shown to be monotone for absolute values of arguments while still keeping track of signs.

Definition 1. *Let $\boldsymbol{w} = [w_1, \ldots, w_n]$ be a vector of weights such that $w_j \geq 0$ for $j = 1, \ldots, n$ and $\sum_{j=1}^n w_j = 1$. Suppose that $x_1, \ldots, x_n$ are realizations of the continuous random variable defined on the interval $[-1,1]$. A function $F : [-1,1]^n \to [-1,1]$ defined as*

$$F(x_1, \ldots, x_n) = \sum_{j=1}^{n} w_j \cdot x^*_{(j)} \tag{7}$$

is called the Bipolar OWA function (BOWA), where $x^*_{(j)}$ denotes the j-th largest absolute value of element in the collection of aggregated objects $x_1, \ldots, x_n$ multiplied by the original sign of that element.

Alternative notation to express BOWA operator is

$$F(x_1, \ldots, x_n) = \langle \boldsymbol{w}, \boldsymbol{x}^* \searrow_B \rangle, \tag{8}$$

where $\langle \cdot, \cdot \rangle$ is the scalar product of vectors and the symbol $\searrow_B$ indicates a non-increasing ordering (proposed to be called *bipolar*) of elements obtained for their absolute values and thus ignoring their signs.

The BOWA operator definition in the presence of ties in the *bipolar* ordering of arguments was separately defined in [6]. Arguments having the same absolute values are given identical weights, which are computed as the average of weights that would be gathered if the arguments had not been tied.

The basic BOWA operator properties [3] include idempotence, symmetry and homogeneity. Moreover, each BOWA function for absolute values of its arguments is an OWA operator. Similarly as OWA functions, BOWA operators do not have neutral or absorbing elements, except for the special cases. They are however not shift-invariant. BOWA operator with adequately chosen weights, such that higher correlations are rewarded whatever are their signs, is a suitable function F that satisfies all postulates required by the measure of association (6) searched for. An example of such vector of weights was suggested in [3] and is also recalled below.

Example 2. Consider two consumers A and B. Assume that the pairise correlation between their preferences for each of the three attributes under study was calculated using Spearman's coefficient. As a result we received the following three numbers: $s^1_{AB} = 0.5$, $s^2_{AB} = 0.4$ and $s^3_{AB} = -1$.

To aggregate these three coefficients the following operator $F_{LG} : [-1, 1]^n \to [-1, 1]$ was suggested in [3]

$$F_{LG}(x_i, \ldots, x_n) = \frac{2}{n(n+1)} \sum_{j=1}^{n} r(|x_j|) \cdot x_j, \tag{9}$$

where $r : [0, 1] \to \mathbb{R}^+$ is a function such that

$$r(z) = \frac{1}{2} + \sum_{i=1}^{n} c(z - |x_i|) \tag{10}$$

and where c is defined as

$$c(u) = \begin{cases} 0 & \text{if} \quad u < 0 \\ \frac{1}{2} & \text{if} \quad u = 0 \\ 1 & \text{if} \quad u > 0. \end{cases} \tag{11}$$

The suggested operator is a member of a family of BOWA operators (7). Let us consider given correlation coefficients as a vector $\boldsymbol{x} = [0.5, 0.4, -1]$. Hence we get a vector of argument values $\boldsymbol{x}^* \searrow_B = [-1, 0.5, 0.4]$ in the bipolar order. Using

ranks given by (10), we may compute a vector of weights $w = [0.5, 0.(3), 0.1(6)]$ and therefore, by (8) we get

$$F(0.5, 0.4, -1) = 0.5 \cdot (-1) + 0.(3) \cdot 0.5 + 0.1(6) \cdot 0.4 = -0.2(6).$$

□

5 Using BOWA-based similarity measure in cluster analysis for collaborative filtering

The constructed similarity measure (6) between a pair of consumers based on BOWA operator, allows us to conduct cluster analysis for a set of consumers represented as in (1). We assume the goal of such clustering proccess is to find natural similarity groups among consumers and characterize the groups.

Consider a finite set of consumers $X \subset \mathcal{X}$ and the K-medoids [7] as a cluster analysis method to be conducted on X. K-medoids is a combinatorial cluster analysis [7], a generalization of a popular K-means algorithm for observations with arbitrary attributes. It admits arbitrary dissimilarity measure instead of a squared Euclidean distance. The center of each cluster, the medoid, is the cluster member that minimizes a total dissimilarity to all other members of the cluster. The BOWA based S similarity measure (6) can be easily adopted to be used as a dissimilarity S' in K-medoids method, i.e. $S' = -S$.

Dissimilarity S' can then also be used for several distance-based clustering quality measures, as Silhouette coefficient [13], Gamma index [14], C-index [15] or Caliński and Harabasz index generalised for dissimilairites [16]. Another proposed way to assess clustering quality is to measure the averaged agreement among consumers belonging to the same cluster in relation to the agreement among medoids. To compute an overall agreement of a group of consumers we can use the analogy to how the BOWA based similarity measure between two groups of rankings is constructed (6). First, we measure the agreement for each attribute separately, i.e. concordance of a set of rankings, using e.g. the Kendall's coefficient of concordance [8] which ranges between 0 (no agreement) to 1 (perfect agreement). Then using OWA operator with a proper weight vector that rewards higher correlations, e.g. obtained by (10), we aggregate the coefficients to obtain single value agreement indicator.

Now, consider a certain resulting cluster and assume that the history of object choices (5) of each cluster member is known. Let $\{T^{(1)}, T^{(2)}, \ldots, T^{(t)}\}$ be a set of all considered objects that the histories are based on. The following procedures of obtaining a meaningful cluster description are suggested:

P1. Picking or creating a representative consumer. Obvious way of representing a cluster is to pick the consumer that serves as the medoid. However, an equivalent of a centroid [7] known from the K-means procedure could also be computed, i.e. the averaged member of the cluster having the form of a vector (1), such that its each element is obtained as a result of aggregating corresponding elements of vectors representing all cluster members.

P2. Creating a representative object. A single object $T^{(rep)} \in \mathcal{Z}$ of the form $\mathbb{V}_{T^{(rep)}} = [V_{T_1^{(rep)}}, V_{T_2^{(rep)}} \ldots, V_{T_n^{(rep)}}]$ is created such that for the j-th attribute each element of the vector $V_{T_j^{(rep)}}$ is obtained as a result of aggregating corresponding elements of vectors $V_{T_j^{(i)}}$, $i = 1, \ldots, t$, if the i-th object is among the ones most often chosen by the cluster members.

P3. Creating a representative history of object choices. A vector having the form of the history of choices (5) is created, such that its each element is obtained as a result of aggregating corresponding elements of vectors representing histories of choices of all cluster members.

Keeping in mind the introductory example discussed in Section 2, we observe that cluster analysis can be especially beneficial also for recommendation creation purpose, for the assumed data representation model.

Firstly, we notice that history of consumer choices (5) can serve as a ground truth for algorithms that learn to predict a preferred order of a given set of objects for a given consumer, i.e. object ranking [17] or preference-based [4] algorithms. In practice, histories of choices can vary a lot between consumers regarding the number and the types of objects chosen. Notice that it applies even to very similar consumers (where similarity is understood as defined in Section 4). Big differences in the types of objects chosen result in sparse history vectors. Here, the cluster analysis can be helpful in dealing with the sparsity. Any consumer A to be used in the learning or testing phase of the object ranking procedure can be replaced with his or her cluster's representative consumer (see P1.). Ground truth history of choices of the cluster's representant is enriched with the history of A, reducing the sparsity problem.

On the other hand, the fact that a given consumer is assigned to a certain cluster, can be used as additional information (a feature in the input vector) about the consumer, possibly improving the prediction quality of object ranking procedure.

6 Conclusions

In this paper an extended data representation model of consumer preferences and objects of their preferences was proposed. A construction of the similarity measure between groups of rankings coming from a pair of consumers, based on a family of semi-aggregation BOWA operators, was summarized. It was shown how the measure can be applied in cluster analysis performed as a part of collaborative filtering process and what are the motivations behind it. Practical issues of conducting and evaluating the clustering process were discussed. Further work assumes experimental verification of the proposed consumer clustering procedure, including the defined similarity measure, performed on real and generated data.

Acknowledgements

Study was supported by research fellowship within "Information technologies: research and their interdisciplinary applications" project co-financed by European Social Fund (agreement no. POKL.04.01.01-00-051/10-00).

References

1. Bennett, J., Lanning, S.: The Netflix Prize. In: Proceedings of KDD Cup and Workshop 2007. (2007)
2. Dror, G., Koenigstein, N., Koren, Y., Weimer, M.: The Yahoo! Music Dataset and KDD-Cup'11. Journal of Machine Learning: Workshop and Conference Proceedings **18** (2012) 3–18
3. Łącka, H., Grzegorzewski, P.: On Measuring Association between Groups of Rankings in Recommender Systems. In Rutkowski, L.e.a., ed.: Proceedings of the 13th International Conference on Artificial Intelligence and Soft Computing (ICAISC 2014), Part II, Lecture Notes in Artificial Intelligence 8468, Springer (2014) 423–432
4. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering **17**(6) (2005) 734–749
5. Shi, Y., Karatzoglou, A., Baltrunas, L., Larson, M., Hanjalic, A.: GAPfm: optimal top-n recommendations for graded relevance domains. In: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management. (2013) 2261–2266
6. Grzegorzewski, P., Łącka, H.: Recommender systems and BOWA operators. In Angelov, P.e.a., ed.: Proceedings of the 7th International Conference on Intelligent Systems IEEE IS 2014, Part I, Advances in Intelligent Systems and Computing 322, Springer (2015) 11–21
7. Hastie, T., Tibshirani, R., Friedman, J.: Cluster analysis. Practical issues. In: The Elements of Statistical Learning. Springer (2001) 518–520
8. Gibbons, J., Chakraborti, S.: Nonparametric Statistical Inference. Marcel Dekker Inc. (2003)
9. Yager, R.: On ordered weighted averaging aggregation operators in multicriteria decisionmaking. IEEE Transactions and Systems, Man and Cybernetics **18** (1988) 183–190
10. Beliakov, G., Pradera, A., Calvo, T.: Aggregation Functions: A Guide for Practitioners. Springer (2007)
11. Calvo, T., Kolesarova, A., Komornikova, M., Mesiar, R.: Aggregation operators: Properties, classes and construction methods. In: Aggregation Operators. New Trends and Applications. Volume 97 of Studies in Fuzziness and Soft Computing. Springer (2002) 3–104
12. Grabisch, M., Pap, E., Marichal, J., Mesiar, R.: Aggregation Functions. Cambridge (2009)
13. Rousseeuw, P.: Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics **20**(1) (1987) 53–65
14. Baker, F., Hubert, L.: Measuring the power of hierarchical cluster analysis. Journal of the American Statistical Association **70** (1975) 31–38

15. Hubert, L., Levin, J.: A general statistical framework for assessing categorical clustering in free recall. Psychological Bulletin **83** (1976) 1072–1080
16. Hennig, C., Liao, T.: How to find an appropriate clustering for mixed-type variables with application to socio-economic stratification. Journal of the Royal Statistical Society, Series C Applied Statistics **62** (2013) 309–369
17. Fürnkranz, J., Hüllermeier, E.: Preference learning: An introduction. In Fürnkranz, J., Hüllermeier, E., eds.: Preference Learning. Springer-Verlag (2010) 1–17

Modelling Spot Prices on the Polish Power Exchange

Michał Pawłowski[1] and Piotr Nowak[2]

[1] Institute of Computer Science, Polish Academy of Sciences, ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
[2] Systems Research Institute, Polish Academy of Sciences, ul. Newelska 6, 01-447 Warsaw, Poland

Abstract. In this paper a model for the dynamics of the Polish Power Exchange electricity spot prices is proposed. The model describes most important quantitative features of evolution of the price index which are characteristic to the Polish market.
The dynamics of the electricity spot prices is governed by a mean-reverting jump-diffusion stochastic process with mixed-exponentially distributed jumps. Estimation of the model's parameters is based on historical data. The model may be precisely calibrated to quoted forward contracts, making use of the analytical formula for a forward price.
In the paper valuation of plain vanilla options on the electricity spot price via the Monte Carlo method is also presented.

1 Introduction

The establishment of the Polish Power Exchange (POLPX) was a result of an implementation of the new law in Poland in April 1997, whose one of the most important assumptions was to liberalize the Polish energy market. At that moment the process of restructuring the energy sector was conducted in the majority of European countries. The main purposes behind the reforms were to disassociate electricity, as a tradable commodity, from its transmission services and to establish a market for electricity generators, energy suppliers, companies involved in energy trading and industry clients.

The POLPX began to operate in December 1999. Within six months the electricty spot market started to run. In this way, bilateral contracts gained a benchmark pricing index. The 2000s decade is a period of a very fast development of the POLPX: a property rights market, a register of certificates of origin for the electrical power generated from renewable sources and produced in cogeneration, a spot market for CO_2 emission certificates and finally an electrical power derivatives market were launched.

The trading volume on all electricity markets on the POLPX in 2014 was equal to 186.8 TWh which is 119.4% of the energy generation and 117.7% of

energy consumption in Poland in this year. Currently, there are 66 members of the electricity spot market.

Econometrical patterns, which are typical for most of electricity spot prices time series, are present also while analysing data coming from the POLPX's index. These include: daily, weekly, yearly seasonality, abrupt, usually unexpected jumps and mean-reversion - when the prices are in a spike regime and the extraordinary market situation (caused e.g. by a failure of a transmission network, outage of power plants, a sudden decrease or increase in temperature, low levels of water or droughts, changing possibilities of exploitation of renewable energy sources) finishes, the prices immediately recur to the former, normal level.

In the paper the dynamics of the spot electrical energy prices is modelled by a continuous-time stochastic process which takes into consideration the aforementioned features of prices, i.e. by a mean-reverting jump-diffusion with mixed-exponential jump size distribution. Our model belongs to the class of one-factor models which are characterized by their good matching to data, existence, in many cases, of analytical solutions to various considered problems (e.g. formulas for forward prices) and existence of numerous approximation methods (e.g. for pricing options, etc.). The basic, initial model was introduced by [1] in which the authors decomposed the signal to a seasonality and a mean-reverting to zero diffusion process. However, the authors did not take the possibility of jumps in prices into account.

In [2] a mean-reverting jump-diffusion model with normally distributed jumps was introduced. There is a possibility of deriving an analytical formula for a forward price. Unfortunately, the distribution of jumps on the POLPX's time series of spot electricity prices is not unimodal. What is more, the authors did not explain how to assure that after filtering the jumps from the series, the returns have normal distribution. Another reason for which the model cannot be applied to the Polish market is that the subject of calibration of the model to quoted on the market forward contracts, using the analytical formulas for the forward price, was not raised. Other authors in [3] also noticed that the usage of the Normal distribution of spikes has an effect of overestimation of skewness and kurtosis.

A very interesting approach was presented in a threshold model [4] where the mean-reverting diffusion was combined with a time-inhomogeneous Poisson process of a truncated exponential distribution of jumps. Moreover, to allow for downward, reverting to the mean jumps, the authors introduced a characteristic function indicating the sign of a jump which depends on a current value of a price. The proposed switching threshold is a constant positive spread over a seasonality.

Notwithstanding, the model has some drawbacks as well. There is no possibility to obtain the analytical formula for a forward price. Additionally, [3] critisizes the choice of the truncated exponential distrbution due to the fact that it disallows for big jumps exceeding the fixed threshold. There is also noted that two consecutive jumps of the same sign are impossible to occur and after estimating the model's parameters, the mean-reversion's parameter turned out to

be higher than expected for a base signal and smaller than required to dampen a spike.

Another interesting subclass of one-factor models are regime-switching models. Markov models are very popular nowadays and also in the field of electrical energy prices modelling they are widely applicable. The reason is that one can define separate forms of dynamics for all substantially different ranges of prices values, usually there are three of them: two spike regimes when the prices achieve anomalous values after the upward and downward jumps, and a normal, base regime. There is also a transition matrix which links the regimes by indicating how much the transition from one state to another is probable. For details, see [5, ?,?].

An alternative to all above-mentioned approaches may be a model in which a diffusion generated by a Wiener process is superseded by very frequent and small jumps (representing typical, daily movements of prices) generated by a Levy process of infinite activity. The Levy process is also responsible for big jumps in prices (substitution for the Poisson process). The model was described in [8].

In Section 2 the dynamics of our custom-made model for the Polish market is enunciated. The rest of the paper is organised as follows. Section 3 familiarizes the reader with historical data chosen for analysis. In Section 4 the method of adjusting seasonality to the historical time series is written up in details and also one becomes acquainted with the algorithm of detection of spikes in prices. The course of a process of the parameters estimation is comprised in Sections 5 and 6. In Section 7 the discretization of the continuous-time dynamics, as well as the comparison of the simulated this way trajectories with the historical series (tests for a goodness of fit) are performed. Section 8 demonstrates the form of the analytical forward price and introduces the notions of the market prices of risks. The next paragraph includes the results of option pricing. The last section concludes.

2 The model of the Polish Power Exchange spot prices

Let us now describe the model of the spot prices which reflects the distinctive features of the Polish energy market. The dynamics is governed by a mean-reverting jump diffusion process with the jump size distribution, the idea of which is borrowed from [9]. In that paper asset (not commodity) prices were considered, but inasmuch as the distribution of returns has fatter tails than the normal distribution, the authors decided to add a compound Poisson process component, jumps of which are sampled from the mixed-exponential distribution. This distribution can approximate any distribution with respect to weak convergence as closely as possible and this fact was an inspiration to use this jump distribution in our model. Existence of jumps in case of electricity spot prices trajectories is definitely more pronounced than in case of any other market's prices. Therefore, flexibility in fitting a theoretically described distribution to a dataset of jumps is an added value.

A relative simplicity of the model's formulation results in capability of deriving the closed-form forward price. This in turn enables to calibrate the model to the quoted forward contracts in a very precise way.

We start with the decomposition of the spot price process S_t:

$$S_t = \exp(g(t) + X_t), \tag{1}$$

$$dX_t = -\alpha X_t dt + \sigma dW_t + dJ_t, \tag{2}$$

where α and σ are constants, $(W_t)_{t\in\mathcal{T}}$ is a Wiener process, $(J_t)_{t\in\mathcal{T}}$ is a compound Poisson process of the form

$$J_t = \sum_{i=1}^{N_t} Z_i, \quad t \in \mathcal{T},$$

with constant intensity λ, Z_i are i.i.d. jump magnitudes of translated mixed-exponential distribution, i.e. with density

$$f(z) = q_d \sum_{i=1}^{m} q_i \xi_i e^{\xi_i (z-m_d)} \mathbb{1}_{\{z<m_d\}} + p_u \sum_{j=1}^{n} p_j \eta_j e^{-\eta_j (z-m_u)} \mathbb{1}_{\{z>m_u\}}, \tag{3}$$

where

$$q_d, p_u \geq 0, \ \ q_d + p_u = 1, \ q_i, p_j \in (-\infty, \infty), \ \sum_{i=1}^{m} q_i = \sum_{j=1}^{n} p_j = 1, \ \xi_i > 0, \eta_j > 1.$$

q_d and p_u are the probabilities of negative and positive jumps, respectively. $m_d < 0$ is a minimal (with respect to the absolute value) value of negative jumps, $m_u > 0$ is a minimal value of positive jumps. A necessary condition for $f(z)$ to be a density function is

$$q_1, p_1 > 0, \ \sum_{i=1}^{m} q_i \xi_i \geq 0, \ \sum_{j=1}^{n} p_j \eta_j \geq 0.$$

One of possible sufficient conditions is

$$\sum_{i=1}^{k} q_i \xi_i \geq 0, \ \sum_{j=1}^{l} p_j \eta_j \geq 0$$

for all $k \in \{1, \ldots, m\}$, $l \in \{1, \ldots, n\}$. A special case of the mixed-exponential distribution is a hyperexponential distribution, when all parameters q_i and p_j are nonnegative.

The separation from zero of the support of the density function is caused by the fact that either positive or negative jumps are extreme events, therefore highly greater than zero with respect to absolute value.

Using the Ito lemma, one obtains that S_t follows the stochastic differential equation

$$dS_t = \alpha(\rho(t) - \ln S_t)S_t dt + \sigma S_t dW_t + S_t(e^Z - 1)dN_t, \tag{4}$$

where

$$\rho(t) = \frac{1}{\alpha}\left(\frac{dg(t)}{dt} + \frac{1}{2}\sigma^2\right) + g(t).$$

3 Historical data

Data chosen for estimation of the model's parameters comes from the POLPX's IRDN spot index and covers the period of October 2011 – September 2015 (1443 historical prices). It is important to note here that by a spot price we mean a

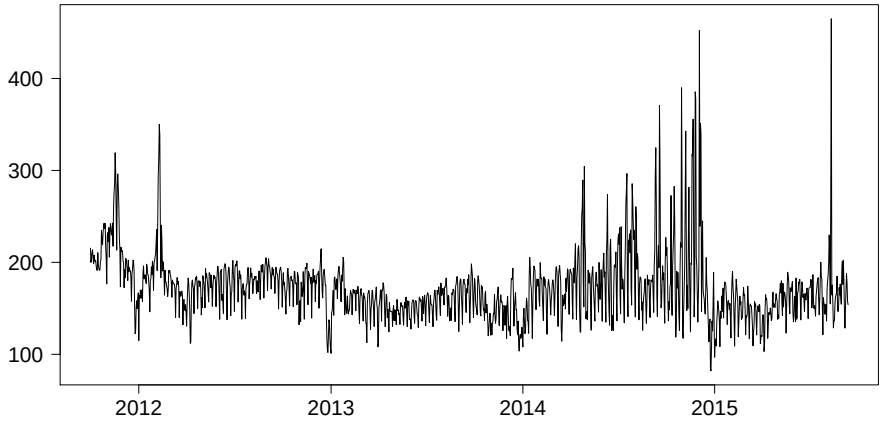


Fig. 1. Spot prices in PLN/MWh

weighted (by volume) average price of daily transactions – a standard day-ahead reference index for contracts with delivery of energy during the whole upcoming day.

At first sight one can state that prices undergo some yearly and weekly (prices on Sundays are decidedly smaller than on other days) seasonal movements and that from time to time a spike occurs.

4 Seasonality matching and filtering of spikes

In this paper an original, robust method of deseasonalisation is proposed. Before all, a very important aspect to consider is that after the logarithm transform and

deseasonalisation of prices (see eq. (1)) combined with the filtrating of jumps, the remaining residue is modelled by the zero-mean-reverting diffusion process which has normally distributed increases. It means that the way of seasonality matching and filtering of spikes must not on any account be arbitrary. Our idea is to perform spikes filtering so as to maximize a p-value of the appropriate statistical test for normality of the aforementioned increases.

The deseasonalisation itself is divided into several stages.

4.1 Downward spikes appearing on holidays

The Polish market has an attribute that if a day is a national holiday, then a negative spike takes place. These spikes are removed from the series as a first part of deseasonalisation. Every spike's value is replaced with a mean of 5 preceding and 5 following prices. There are 12 such deterministic downward spikes each year.

4.2 Matching of weekly and yearly oscillations

The weekly seasonality is computed as means of logarithms of prices of all days within a week. Afterwards, these values are subtracted from the log-index, but days of the holidays are excluded from this procedure. The yearly fluctuations are found by adjusting, by a nonlinear least-squares method, a one-year periodic, sinusoidal function of the form

$$a + bt + \sum_{k=1}^{3} c_k \sin\left(\frac{2k\pi t}{365}\right) + d_k \cos\left(\frac{2k\pi t}{365}\right).$$

The fitted this way function is shown in Figure 2.

4.3 Spikes filtering

Filtering of spikes is performed by an iterative procedure: in the first step all jumps which absolute value exceeds some predefined threshold, for instance three times the standard deviation of the deseasonalised log-returns, are removed from the series. In the next step the same action is made, but this time the standard deviation is calculated basing on the thinned series of returns. New jumps are filtered and deleted and the process continues until in some iteration no jumps are found.

The most important aspect of this method is to fix the threshold so as to maximize the p-value of the Anderson-Darling normality test for the deseasonalised, and with deleted jumps, log-returns – the assumptions of the model must be fulfilled. For our data the threshold turned out to be $2.45s$, where s is the standard deviation of the series obtained in each step of the described procedure. The maximized p-value is equal to 0.113. There is no evidence to reject the null hypothesis of the log-returns normality at the 10% significance level. A similar idea, referring to the shape of a seasonality function, was applied in [10].

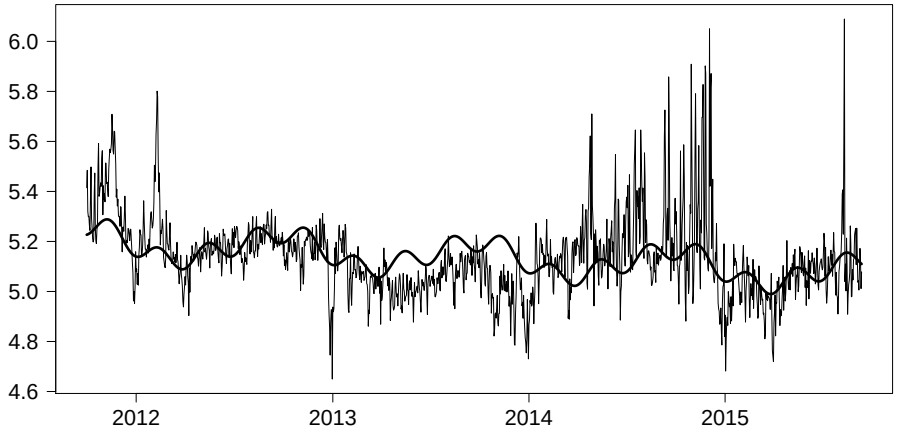


Fig. 2. Annual sinusoidal function fitted to the partially deseasonalised log-price series

The choice of the Anderson-Darling normality test is dictated by its one of the best capabilities of detecting most departures from normality, cf. [11].

After filtering of spikes, their values in the series are replaced by a mean of 5 preceding and 5 following prices.

4.4 Making seasonality independent from the spikes occurrences

If the seasonality was matched basing upon the raw historical data, then this estimation would be biased by the presence of jumps. To counteract this problem, we propose the following procedure:

1. logarithmize the input series of prices and remove holidays downward spikes (cf. Subsection 4.1),
2. eliminate the rest part of seasonality, i.e. weekly and yearly oscillations (cf. Subsection 4.2),
3. filter out and remove spikes (cf. Subsection 4.3),
4. add the seasonality fitted in point 2. to the deseasonalised and bereft of spikes series and then once again perform the (this time robust to spikes) deseasonalisation described in point 2.

Obtained this way seasonality is not influenced by the magnitudes of jumps in prices and thus should be used as a part of the formula (1) to achieve the historical realization of the process (2). A similar technique was adapted in [12].

It is worth to see this aggregate form of the seasonality applied for the upcoming years, see Figure 3. The seasonality in some magnification, around Christmas, is shown in Figure 4.

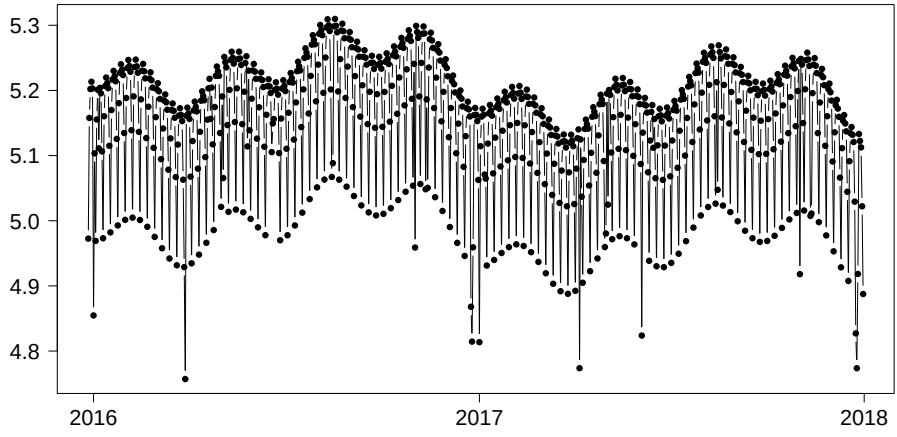


Fig. 3. The overall seasonality function in logarithmic scale in PLN/MWh

5 Estimation of the driving process's parameters

After the deseasonalisation and removal of spikes from the log-series, one may proceed to estimation of the jump-diffusion's parameters. The volatility σ from the equation (2) is estimated as a mean of the rolling standard deviation of the time-scaled increments $\frac{P_i - P_{i-1}}{\sqrt{t_i - t_{i-1}}}$ (see [13], formula 3.10):

$$\sigma(t_k) = \sqrt{\frac{1}{m-1} \sum_{i=k-m+1}^{k} \left(\frac{P_{i+1} - P_i}{\sqrt{t_{i+1} - t_i}} - \frac{1}{m} \sum_{j=k-m+1}^{k} \frac{P_{j+1} - P_j}{\sqrt{t_{j+1} - t_j}} \right)^2},$$

where P is the deseasonalised and devoid of spikes log-price index, $m = 30$, $M = 1305$ (after removing of jumps there are 1305 log-returns), $k \in \{m, \ldots, M\}$. For all $i \in \{1, \ldots, 1306\}$ $t_{i+1} - t_i = \frac{1}{365}$. The estimated value $\sigma = 1.14$.

Determination of the mean-reversion's velocity α is based on the deseasonalised log-prices, but in the presence of spikes. One has to regress the deseasonalised log-prices series bereft of its first element versus the deseasonalised log-prices series without its last element, which is a direct cause of the discretized form (see details in Subsection 7.1) of the equation (2):

$$X_{t_k} = e^{-\alpha \Delta t} X_{t_{k-1}} + \rho_{t_k},$$

where ρ_{t_k} is the sum of integrals of the Wiener process and the compound Poisson process between times t_{k-1} and t_k. The value of the regression coefficient $e^{-\alpha \Delta t}$ is significantly different from zero – the speed of mean-reversion achieved this way equals $\alpha = 0.3$.

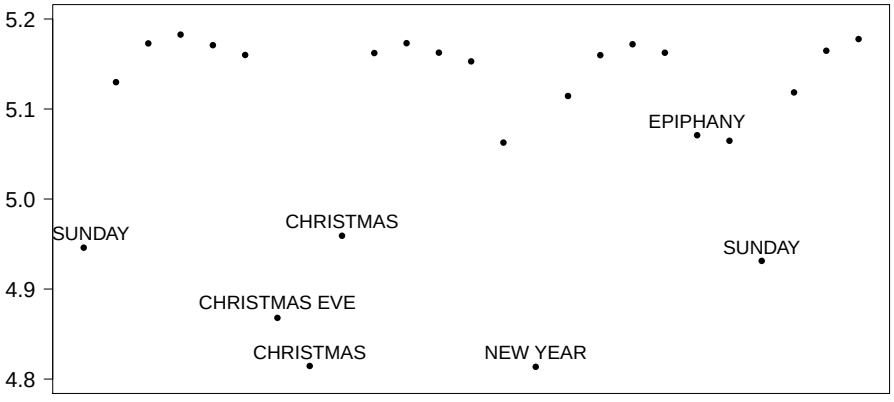


Fig. 4. Seasonality function around Christmas 2016

The results of the augmented Dickey-Fuller test applied for the deseasonalised log-prices indicate that there is no unit root in our time series data – the mean-reversion is indeed present.

6 Evaluation of the jump-size distribution's parameters

To filter jumps for the purpose of the jump-size distribution's parameters estimation, we use the algorithm described in Subsection 4.3, but a salient modification is necessary – some of the filtered jumps are mean-reversions of the process and thus have to be excluded from the analysis. Accordingly, if there are two or three consecutive jumps and the last one is of opposite sign, it is regarded as a mean-reversion. 137 returns are classified as jumps by the filtering algorithm and out of them 43 are assessed as mean-reversions, yielding the yearly frequency of the Poisson process $\lambda = (137 - 43)/1442 \cdot 365 = 23.8$. Counting downward and upward jumps brings $q_d = 0.32, p_u = 0.68$. The minimal sizes of negative and positive jumps are equal to $m_d = -0.152, m_u = 0.148$, respectively.

The remaining parameters are estimated by the maximum likelihood method – see Table 1 (in the density function specification (3) we take $m = n = 2$, which is a compromise between the accuracy and the number of parameters to be evaluated).

The parameters q_1, q_2, p_1, p_2 are all positive, so that the jump-size distribution turns out to be hyperexponential, a special case of the mixed-exponential distribution. Figure 5 illustrates the adjustment of the density to the empirical distribution of filtered jumps.

q_1	q_2	ξ_1	ξ_2	p_1	p_2	η_1	η_2
0.06	0.94	1.78	21.78	0.64	0.36	5.79	40.66

Table 1. Estimated parameters of the mixed-exponential jump size distribution

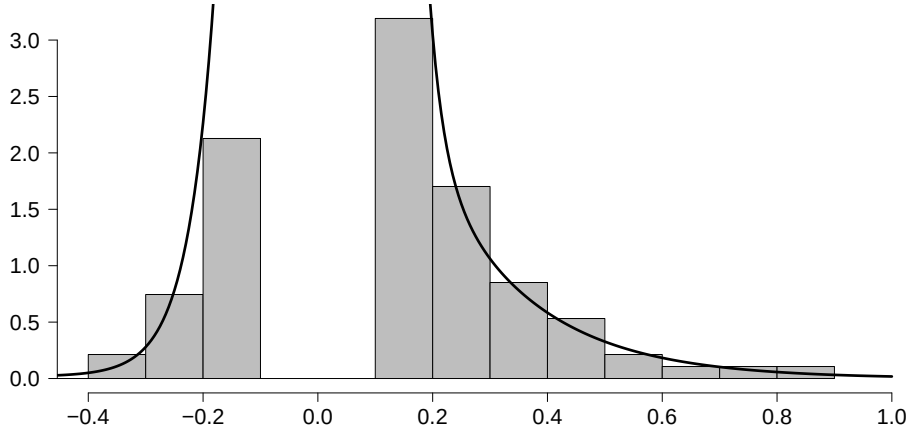


Fig. 5. Mixed-exponential distribution fitted to the empirical distribution of jumps

7 Simulation of the spot prices and tests for the trajectories

7.1 Discretization of the process

Lemma 1. *Let X_t follow the equation (2) and may $0 \leq s \leq t$, $t \in \mathcal{T}$. Then*

$$X_t = e^{-\alpha(t-s)} X_s + \int_s^t \sigma e^{-\alpha(t-u)} dW_u + \sum_{s<u\leq t,\ \Delta N_u \neq 0} e^{-\alpha(t-u)} \Delta J_u. \quad (5)$$

Moreover,

$$\int_s^t \sigma e^{-\alpha(t-u)} dW_u \sim N\left(0, \sigma \sqrt{\frac{1-e^{-2\alpha(t-s)}}{2\alpha}}\right). \quad (6)$$

Proof. We refer the reader to [14].

Hence, the discretized dependency between the consecutive "daily" values of the process X_t is of the form

$$X_{t_k} = X_{t_{k-1}} \exp\left(\frac{-\alpha}{365}\right) + \sigma\sqrt{\frac{1-\exp\left(\frac{-2\alpha}{365}\right)}{2\alpha}}N(0,1) + \sum_{i=1}^{N_{1/365}} Z_i, \qquad (7)$$

where $N(0,1)$ is a standard normally distributed variable, $N_{1/365}$ is a Poisson random variable with the intensity parameter $\frac{\lambda}{365}$, Z_i are mixed-exponentially distributed random variables. A sample trajectory put on the seasonality is shown in Figure 6.

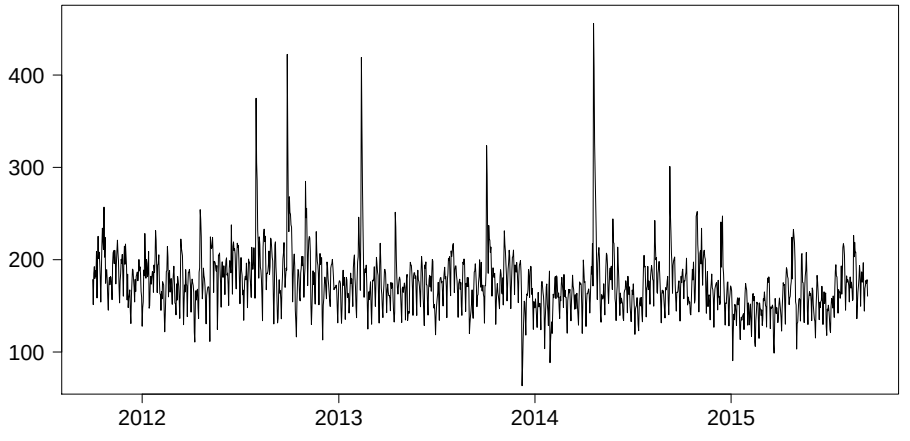


Fig. 6. Simulated sample path in PLN/MWh

7.2 Goodness of fit of the sample paths

The comparison of two moments and 15%, 85% quantiles of the historical log-returns and log-returns of 5000 simulated trajectories is shown in Table 2.

The Kolmogorov-Smirnov test for the equality of distributions of the real log-increases and the log-increases of the simulated data gives no evidence to reject the null hypothesis of the equality of these distributions at a reasonable level – the averaged p-value (over 5000 samples) is equal to 0.09.

The reestimation procedure was also conducted, i.e. for each simulated path all the parameters were estimated and then were averaged over samples – the resulting parameters' values were very similar to those computed during the estimation described in Section 5.

	mean	st. dev.	15% quantile	85% quantile
real data	-0.00023	0.153	-0.126	0.131
simulations	-0.00008	0.14	-0.128	0.133

Table 2. Moments and quantiles of the historical and 5000 simulated log-returns (averages)

8 Analytical formula for the forward price

One of the biggest advantages of the model is that it enables to derive an analytical formula

$$F(t,T) = \mathbb{E}^{\mathbb{Q}}[S_T|S_t] \tag{8}$$

for the forward prices F(t,T), $0 < t \leq T$, $T \in \mathcal{T}$, where $\mathbb{Q}$ is an equivalent risk-neutral measure. Thanks to this analytical formula one can create a forward curve and thanks to the change of measure it is possible to make the model suited to the actual prices. From mathematical point of view, there are uncountably many equivalent, potentially risk-neutral measures and the task is to pin down the appropriate one. From financial point of view, the considered electrical energy market is incomplete, as there are more sources of randomness (hence risk) than risky assets, thus not every payoff may be replicated (hedged) with this underlying asset and not risky one, for instance a bank account or a bond. In the model there are two sources of risk: diffusion risk connected to the Wiener process and jump risk related to the compound Poisson process. To deal with the problem of calibration of the model to the actual market situation and quoted forward contracts, the notions of the market prices of diffusion risk and jump risk are introduced. Ascribing the concrete numerical values to the parameters which denote the market prices of risks uniquely determines the choice of the appropriate risk-neutral measure. For details, we refer the reader to [14].

Theorem 1. *The analytical formula for the forward price within the model defined in Section 2 by (2) and (3) is equal to*

$$\begin{aligned} &F(t,T) = \mathbb{E}^{\mathbb{Q}}[S_T|\mathcal{F}_t] = \\ &G(T)\left(\frac{S_t}{e^{g(t)}}\right)^{e^{-\alpha(T-t)}} \exp\left(\int_t^T \sigma e^{-\alpha(T-s)}\left(\frac{1}{2}\sigma e^{-\alpha(T-s)} - \theta^{\mathbb{Q}}\right)ds\right)\cdot \\ &\exp\left(\int_t^T \left(e^{m_d} q_d \sum_{i=1}^{m} q_i \frac{\xi_i e^{\alpha(T-s)}}{\xi_i e^{\alpha(T-s)}+1} + e^{m_u} p_u \sum_{j=1}^{n} p_j \frac{\eta_j e^{\alpha(T-s)}}{\eta_j e^{\alpha(T-s)}-1}\right)\lambda^{\mathbb{Q}} ds\right. \\ &\left. - \lambda^{\mathbb{Q}}(T-t)\right), \end{aligned} \tag{9}$$

where $\theta^{\mathbb{Q}}$ is the market price of diffusion risk and $\frac{\lambda}{\lambda^{\mathbb{Q}}}$ is the market price of jump risk with $\lambda^{\mathbb{Q}}$ an intensity of the compound Poisson process after change of measure to the risk-neutral $\mathbb{Q}$.

Proof. We refer the reader to [14].

9 Valuing options on electricity spot price

A call option contract on the underlying asset, which in this case is electricity spot price, gives its holder at expiration date T the right (it is not an obligation as in case of a forward contract) to buy electricity for K instead of S_T. Likewise, a put option contract secures the right to sell electricity for K instead of S_T.

The problem of pricing at time t a call vanilla option $C_{t,T}(K)$ on electicity spot, expiring at time T and with strike K, is equivalent to finding value of the following expression

$$\begin{aligned} C_{t,T}(K) &= \exp(-r(T-t))\mathbb{E}^{\mathbb{Q}}[\max(S_T - K, 0)|S_t] = \\ &\exp(-r(T-t))\mathbb{E}^{\mathbb{Q}}[\max(\exp(X_T + g(T)) - K, 0)|X_t], \end{aligned} \tag{10}$$

where r is a discount rate. Analogously, a price of a put vanilla option $P_{t,T}(K)$ is given by

$$P_{t,T}(K) = \exp(-r(T-t))\mathbb{E}^{\mathbb{Q}}\left[\max(K - \exp(X_T + g(T)), 0)|X_t\right]. \tag{11}$$

Here we assume that $\mathbb{Q} = \mathbb{P}$, i.e. that we price options with respect to the physical measure, where the probabilities of events are induced by the historical realisation of prices values. It is due to the fact that methodology and results of the calibration of the model to the risk-neutral measure $\mathbb{Q}$ lie out of the scope of this article, and are the subject of a forthcoming paper. In this section we concentrate on a form of the price estimator and application of the driving process simulation method described earlier.

An adequate tool to cope with such defined problem is a Monte Carlo setup. By the strong law of large numbers, Monte Carlo estimators of (10) and (11) are equal to

$$\widehat{C}_{t,T} = \exp(-r(T-t))\frac{1}{n}\sum_{i=1}^{n}\max\left(\exp\left(X_T^{(i)} + g(T)\right) - K, 0\right) \tag{12}$$

and

$$\widehat{P}_{t,T} = \exp(-r(T-t))\frac{1}{n}\sum_{i=1}^{n}\max\left(K - \exp\left(X_T^{(i)} + g(T)\right), 0\right), \tag{13}$$

where $X_T^{(1)}, X_T^{(2)}, \ldots, X_T^{(n)}$ are sample values of the process X at time T obtained by simulating trajectories from t up to T according to the formula (7).

In Table 3 there are presented call and put option prices on electricity spot with time to expiry equal to 90 days, i.e. $T - t = \frac{90}{365}$, and with different strike prices K. The value of the seasonality function at expiry date is $\exp(g(T)) = 158$ PLN/MWh, discount rate $r = 0.02$.

K	120	130	140	150	160	170	180	190
$\widehat{C}_{t,T}$	43.90	34.11	24.66	16.24	9.89	6.00	3.85	2.69
$\widehat{P}_{t,T}$	0.14	0.30	0.80	2.33	5.93	11.00	19.79	28.59

Table 3. Call and put option prices on electricity spot with time to expiry equal to 90 days and the seasonality function at expiry date equal to 158 PLN/MWh

10 Conclusions

In the article the authors introduce the new model for electricity spot prices which are quoted on the Polish Power Exchange, taking into account all the specificity of the Warsaw market, as well as the electrical energy prices specificity in general. Several novel ideas concerning seasonality matching, spikes filtering, jump-size distribution, etc. are put into practice. The parameters are estimated basing on the historical data. The model is validated by performing simulations and tests for goodness of fit, which legitimize the proposed approach. Within the model there exists an analytical formula for the forward prices allowing for convenient calibration of the model to the forward contracts quoted on the exchange, making use of the notions of the market prices of diffusion and jump risks. Finally, valuing of options on electricity spot price is performed.

Acknowledgements

The paper is co-founded by the European Union from resources of the European Social Fund. Project PO KL "Information technologies: Research and their interdisciplinary applications", Agreement UDA-POKL.04.01.01-00-051/10-00.

References

1. Lucia, J., Schwartz, E.: Electricity prices and power derivatives: Evidence from the Nordic Power Exchange. Review of Derivatives Research **5**(1) (2002) 5–50
2. Cartea, A., Figueroa, M.: Pricing in Electricity Markets: a mean reverting jump diffusion model with seasonality. Applied Mathematical Finance **12**(4) (2005)
3. Benth, F.E., Kiesel, R., Nazarova, A.: A critical empirical study of three electricity price models. Review of Derivatives Research **34**(5) (2011) 1589–1616
4. Geman, H., Roncoroni, A.: Understanding the Fine Structure of Electricity Prices. Journal of Business **79**(3) (2006)
5. de Jong, C., Huisman, R.: Option Formulas for Mean-Reverting Power Prices with Spikes. Energy Global Research Paper (2002)
6. Janczura, J., Weron, R.: An empirical comparison of alternate regime-switching models for electricity spot prices. Energy Economics **32**(5) (2010) 1059–1073
7. Lindstrom, E., Regland, F.: Modelling extreme dependence between European electricity markets. Energy Economics **34**(4) (2012) 899–904

8. Benth, F.E., Saltyte-Benth, J.: The normal inverse Gaussian distribution and spot price modeling in energy markets. International Journal of Theoretical and Applied Finance **7**(2) (2004)
9. Cai, N., Kou, S.G.: Option Pricing Under a Mixed-Exponential Jump Diffusion Model. Management Science **57**(11) (2011) 2067–2081
10. Weron, R.: Market price of risk implied by Asian-style electricity options and futures. Energy Economics **30**(3) (2008) 1098–1115
11. Stephens, M.A.: EDF Statistics for Goodness of Fit and Some Comparisons. Journal of the American Statistical Association **69**(347) (1974) 730–737
12. Janczura, J., Truck, S., Weron, R., Wolff, R.C.: Identifying spikes and seasonal components in electricity spot price data: A guide to robust modeling. Energy Economics **38** (2013) 96–110
13. Eydeland, A., Wolyniec, K.: Energy and Power Risk Management. Wiley Finance (2003)
14. Pawłowski, M., Nowak, P.: Pricing forward contracts on the Polish Power Exchange. Research Report, RB/1/2015, SRI PAS (2015)

Personalised Simulation of Haemodynamic Response to the Valsalva Manoeuvre

Leszek Pstraś[1], Karl Thomaseth[2] and Jacek Waniewski[1]

[1] Nałęcz Institute of Biocybernetics and Biomedical Engineering, Polish Academy of Sciences,
ul. Ks. Trojdena 4, 02-109 Warsaw, Poland

[2] Institute of Electronics, Computer and Telecommunication Engineering, National Research Council,
Via G. Gradenigo 6/b, 35131 Padova, Italy

Abstract. The Valsalva manoeuvre is a simple and low-risk procedure used for assessing the autonomic nervous system or for diagnosing several heart conditions. The analysis of the cardiovascular alterations occurring during and after the manoeuvre due to the changes of the intrathoracic pressure can be facilitated using mathematical modelling. In this paper we present a method of employing a mathematical model to simulate the haemodynamic response to the Valsalva manoeuvre in a given individual. In particular, we present a method of adapting our own multicompartmental mathematical model of cardiovascular system (based on standard physiological data from the literature) to reflect the steady state of the cardiovascular system in the given subject before the manoeuvre is started. The structure of our cardiovascular model is also briefly discussed providing some particulars on the used modelling techniques.

Keywords: blood pressure, heart rate variations, baroreflex, autonomic function, mathematical model

1 Introduction

The Valsalva manoeuvre (VM) is often used as a simple, non-invasive, inexpensive and low-risk procedure of diagnosing several heart conditions (including heart failure and heart murmurs abnormalities) [1–5] or for testing the autonomic nervous system [6–9]. The manoeuvre consists in a forced expiratory effort against a closed airway, which increases the intrathoracic and intra-abdominal pressure and causes a specific haemodynamic response [10–12]. The VM triggers several cardiovascular regulatory mechanisms which are based mainly on the activity of baroreceptors (blood pressure sensors), with some influence from slowly adapting pulmonary stretch receptors, as well as central and peripheral

chemoreceptors [13–15, 11] and an almost negligible impact of nonautonomic humoral mechanisms (e.g. angiotensin II) [16].

For assessing the autonomic function, the VM is typically performed with the patient in the supine or sitting position with the intraoral pressure equal to 40 mm Hg maintained for 15 seconds [11, 12]. The diagnosis is based on heart rate variations, which can be recorded with electrocardiography or using a finger cuff device [12].

The changes in arterial blood pressure (BP) and heart rate (HR) during and after the typical VM can be divided into 4 physiological phases, as follows [17] (see Figure 1): (I) onset of strain with a rise of arterial pressure and a slight drop of HR, (II) continued strain with a decrease of arterial pressure, the corresponding tachycardia and ensuing partial pressure recovery, (III) pressure release with a sudden drop of BP and further increase in HR, (IV) arterial pressure overshoot and the resulting bradycardia [12].

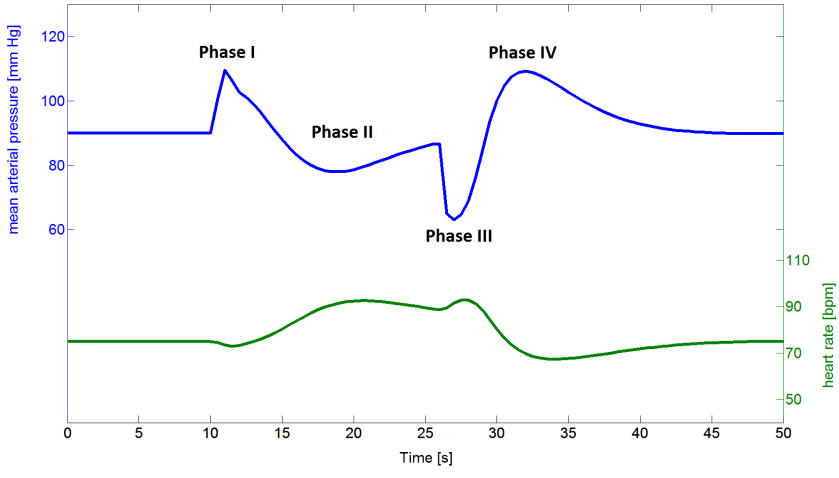


Fig. 1. Simulation of mean arterial pressure and heart rate changes during the Valsalva manoeuvre in a reference patient (the intrathoracic pressure increased to 40 mm Hg for 15 seconds)

2 Mathematical model

2.1 Model structure

We developed a multi-compartmental non-pulsatile model of the cardiovascular system with three baroreflex mechanisms controlling heart rate, peripheral resistance and venous capacity [18]. The proposed model (operating on mean

blood pressures) is much simpler than previously existing multi-compartmental pulsatile models of the VM (describing instantaneous changes of blood pressure with pulse pressure) [19, 20], but still provides a satisfactory representation of the haemodynamic response to the VM. The complete description of the model, its validation and limitations can be found in our previous work [18].

The cardiovascular part of the model involves 7 vascular compartments (aorta, systemic arteries, systemic capillaries, systemic veins, vena cava, pulmonary arteries and pulmonary veins) and 2 cardiac chambers (right heart and left heart, each combining the corresponding atrium and ventricle) [18]. To enable simulation of the VM, 6 intrathoracic compartments (vena cava, right heart, pulmonary arteries, pulmonary veins, left heart and aorta) are connected to a pressure source corresponding to the intrathoracic pressure (see Figure 2) [18]. The vena cava compartment includes both superior and inferior vena cavae. All vascular compartments are modelled as capacitors (representing the volume of blood stored in the compartment at a given pressure) with hydraulic resistances between the compartments corresponding to pressure and energy losses associated with the blood flow (the resistances change dynamically with the changes in the compartment volumes) [18]. With the compartmental structure of the cardiovascular model, we do not describe the continuous blood pressure decline along the blood vessels, but we approximate it instead with a step pressure reduction between the adjacent compartments. Therefore, within each compartment the blood pressure is uniform and equal to the pressure at the entry to the corresponding vasculature.

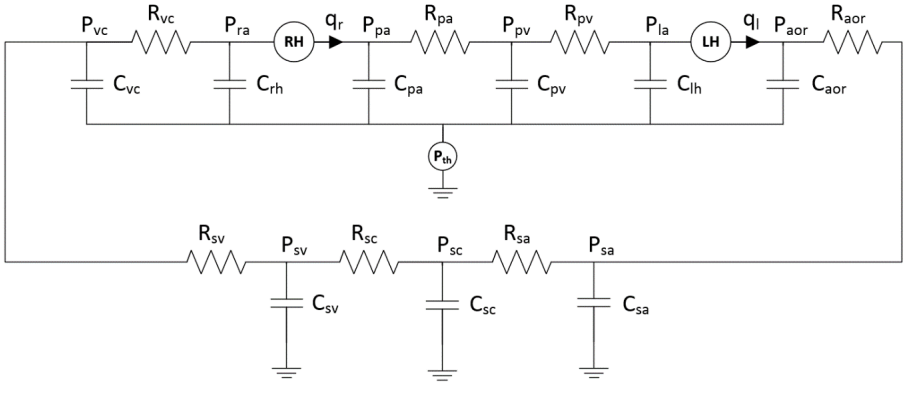


Fig. 2. Electric analogy of the cardiovascular model, where R denotes resistances, P – pressures, C – capacities, q_r and q_l – cardiac outputs from right and left heart ventricles respectively. The meaning of subscripts is: aor – aorta, sa – systemic arteries, sc – systemic capillaries, sv – systemic veins, vc – vena cava, pa – pulmonary arteries, pv – pulmonary veins, rh – right heart, lh – left heart, ra – right atrium, la – left atrium, th – intrathoracic [18].

A slight modification of the representation of vena cava resistance in the model was employed for modelling backflow of blood from right atrium and vena cava during the onset of Valsalva manoeuvre, when the increased intrathoracic pressure compresses these two compartments and pushes some blood back into the systemic veins. Having vena cava resistance represented in the model as a hydraulic resistor between vena cava and right atrium (with the resistance being volume-dependent as described in [18]) distorts the process of emptying of these two compartments, since the significant increase of vena cava resistance (due to volume reduction) impedes emptying of right atrium, while not affecting emptying of vena cava itself. In order to make this process more realistic in the model, for backflow calculations, vena cava resistance was divided in two equal parts localized on each side of vena cava compartment. This way, the blood pushed from the right atrium back to vena cava flows across half of vena cava resistance, while backflow from vena cava to systemic veins takes into account the other half of vena cava resistance (the resistance of systemic veins, being relatively small, was neglected here). This solution was included mainly for technical correctness to avoid unnaturally high differences between pressures of vena cava and right atrium, however it does not affect the results significantly. For normal blood flow (towards the heart) vena cava resistance is always represented as one resistor localized between vena cava and right atrium (see Figure 2).

For all systemic venous compartments (ie. systemic veins and vena cava) nonlinear pressure-volume curves were used due to relatively high lumped compliance of these compartments and relatively high differences in their operating pressures (especially in systemic veins) [18]. For other compartments, linear P-V relationships were used assuming relatively small compliance changes, in which case a linear approximation does not lead to large errors.

A nonlinear (sigmoidal) function was used to describe the relationship between the ventricular stroke volume and atrial pressure (the Frank-Starling law). For the right ventricle we have the following equation [18]:

$$SV_r = \frac{SV_{max,r}}{1 + \exp\left(\frac{-P_{ra} - x_r}{s_r}\right)} a_r \tag{1}$$

where $SV_{max,r}$ represents the maximal right ventricular stroke volume, s_r determines the slope of the sigmoidal function, x_r describes the position of the curve with respect to the atrial pressure axis, a_r is a functional parameter describing the impact of afterload ie. the increase or decrease of stroke volume as a result of decreased or increased pressure downstream the ventricle ([18]).

An analogous equation is used for the left ventricle [18]:

$$SV_l = \frac{SV_{max,l}}{1 + \exp\left(\frac{-P_{la} - x_l}{s_l}\right)} a_l \tag{2}$$

Under the steady-state conditions the cardiac output from both ventricles (calculated as the product of stroke volume and heart frequency) must be equal:

$$q_l = SV_l f = SV_r f = q_r \tag{3}$$

The parameters s and x for both equations are calculated before each run of the model as follows. The parameter s_r is calculated so that when the system is operating at the nominal conditions (ie. at the nominal right atrial pressure $P_{ra,n}$ and the nominal cardiac output q_n) the slope of the SV_r curve (at the nominal operating point) is equal to cardiac output sensitivity to right atrial pressure $sens_r = 35$ ml/min/mmHg/kg [21]. Associating the slope of the curve with the derivative dSV_r/dP_{ra} (from equation 1) and neglecting the impact of afterload, we have hence the following equation for s_r:

$$s_r = \frac{SV_{max,r}\left(\frac{SV_{max,r}f_n}{q_n} - 1\right)}{k_r\left(\frac{SV_{max,r}f_n}{q_n}\right)^2} \tag{4}$$

where q_n is the nominal cardiac output ($q_n = 80$ ml/min/kg body weight [21]) and k_r is $sens_r$ transformed to ml/mmHg units:

$$k_r = \frac{sens_r BW}{60 f_n} \tag{5}$$

where BW is the patient body weight and f_n is the nominal heart frequency (heart rate) in beats per minute ($f_n = 75$ bpm [22]).

Similarly, we have:

$$s_l = \frac{SV_{max,l}\left(\frac{SV_{max,l}f_n}{q_n} - 1\right)}{k_l\left(\frac{SV_{max,l}f_n}{q_n}\right)^2} \tag{6}$$

$$k_l = \frac{sens_l BW}{60 f_n} \tag{7}$$

where $sens_l$ is the sensitivity of cardiac output to changes of left atrial pressure ($sens_l = 20.5$ ml/min/mmHg/kg [23]). $SV_{max,r}$ and $SV_{max,l}$ were both given the value 130 ml [22].

The parameters x_r and x_l are set so that at the nominal atrial pressures ($P_{ra,n}$ and $P_{la,n}$) both right and left ventricular outputs (being the product of stroke volume and heart rate) are equal to the nominal cardiac output.

$$x_r = -P_{ra,n} - s_r \log\left(\frac{SV_{max,r} f_n}{q_n} - 1\right) \tag{8}$$

$$x_l = -P_{la,n} - s_l \log\left(\frac{SV_{max,l} f_n}{q_n} - 1\right) \tag{9}$$

Sigmoidal functions were also used to describe the operation of baroreflex mechanisms based on the activity of three groups of baroreceptors – aortic baroreceptors located in the aortic arch, carotid baroreceptors located in carotid sinuses and cardiopulmonary baroreceptors located in the right atrium [18]. All baroreceptors measure continuously the blood pressure in each location and compare the measured values with the normal value for the given location. Based on the weighted sum of pressure deviations from normal levels, the baroreflex mechanisms modify then the controlled parameters (ie. heart rate, peripheral resistance and venous unstressed volume) in order to bring the pressures back to normal [18]. Note that during the VM the aortic transmural pressure measured by the aortic baroreceptors deviates from the normal aortic pressure to a much higher extent than the transmural pressure in the carotid sinuses.

The model is implemented in Matlab® (The Mathworks Inc.) and all simulations are performed using a built-in solver for stiff systems of ordinary differential equations (ode15s) [18].

2.2 Model parameters

The model includes the following parameters: general parameters of the cardiovascular system (nominal cardiac output, nominal HR, total blood volume), parameters related to P-V curves (blood distribution, nominal pressures and compliances, maximal volumes, parameter reflecting the relative level of vascular compliance etc.), parameters of cardiac stroke volume curve (maximal stroke volume etc.) and parameters of all baroreflex mechanisms (amplitudes, gains and time constants) [18].

All model parameters were taken from the literature and correspond to a normal healthy reference patient - an active, but untrained 70-kg mature male individual, as described in our previous work [18]. The initial (nominal) resistances have been calculated from the nominal pressure differences across the adjacent compartments in the normal steady state taking the nominal blood flow across the whole system of 80 ml/min/kg body weight [21].

Note that the nominal pressures used in the model for each compartment (see [18]) are not the average pressures of the corresponding part of the human vasculature, but the pressures at the entrance of the given vascular tree (as described earlier). Therefore the derived pressure-volume curves and their parameters (eg. unstressed volumes) do not necessarily reflect the real values from humans. In the model it is assumed that all the blood stored in a given compartment is subject to the same pressure and that the modelled vessels have a constant cross-section (constant radius) throughout their length. In reality, not

only the blood pressure drops continuously across the length of the vessels (due to friction losses), but also the total cross-section of the vessel (or a group of vessels) is subject to variations. This aspect can be mostly seen in the systemic arteries compartment which includes large arteries, small arteries and arterioles with a significant pressure difference between the beginning (large arteries) and the end of the compartment (arterioles). This issue applies to all compartments used in the model (although in other compartments the pressure differences are much smaller), however, it has no significant impact on the model outcomes.

2.3 Assumptions

The following assumptions were used when developing the cardiovascular model:

1. blood is an incompressible and Newtonian fluid
2. blood flow throughout the system is laminar
3. the body is in the supine position
4. the effects of muscle pump and respiratory pump are negligible
5. normal intrathoracic pressure is equal to the ambient pressure
6. systemic arteries (except aorta), systemic veins (except vena cava) and systemic capillaries are not compressed by the increased intrathoracic or intra-abdominal pressure
7. inertance effects associated with the blood flow are negligible
8. the cross-section of all vessels remain circular at all times (including vena cava during collapse)
9. active response of vascular smooth muscles to pressure changes is negligible
10. vascular viscoelastic effects (stress relaxation) are negligible
11. there is no hysteresis in the vascular pressure-volume curves
12. the pressure waves reflected from vessel bifurcations are negligible
13. there is no blood filtration or refilling across the capillaries
14. the Anrep effect (a mild increase in heart contractility at increased afterload) is negligible
15. there is no pressure "talk" between right and left ventricle (as modelled in [24])
16. the effects of respiration on heart rate variations are negligible
17. there is no time latency in baroreflex operation
18. baroreceptors are not sensitive to the rate of pressure changes
19. there are no other mechanisms controlling blood pressure (eg. chemoreceptors or lung mechanoreceptors)
20. there is no regional blood flow autoregulation (eg. in brain, heart or kidneys)

3 Model adaptation to the individual patient

3.1 Method overview

The main outputs of the VM are the variations of arterial blood pressure and heart rate measured before, during and after the manoeuvre. The same variables

are the main outputs of our model simulations, as shown in Figure 1. Since the model is based on the literature data representing the reference patient, the comparison of model simulations with experimental data from real patients is not straightforward.

Initially (before any changes of the intrathoracic pressure), the cardiovascular system (modelled as described above and in our previous work [18]) is in the steady state. This steady state is characterized by the cardiac output, heart rate, blood pressures, compartment volumes etc. as set for the reference patient. Before simulating the cardiovascular response to the VM in a real patient, one should modify some model parameters in order to better represent the analysed patient in the simulation. More specifically, one should shift the modelled system from the original steady state corresponding to the physiology of the reference patient to a new steady state corresponding, as well as possible, to the haemodynamics of the given patient. Obviously it is not feasible to provide all the individual pressures and blood distribution across all cardiovascular compartments or other physiological parameters. Therefore, we decided to use a simplified approach and concentrate only on two most important physiological parameters for analysing cardiovascular response to the VM. As already mentioned, these are the arterial blood pressure and heart rate, which are both easy to measure in the patient and which are monitored anyway during the VM. Consequently, we wanted to shift the cardiovascular system to the steady state corresponding to the average arterial blood pressure and average heart rate of the given patient measured before the manoeuvre (ideally recorded over a longer period of time in order to smooth out the natural individual variations and to represent as much as possible the normal values for the given patient, assuming that the patient is not overly excited or anxious). At the same time we wanted to keep as much parameters as possible on the level typically reported in the literature. The problem was hence to find a minimal number of model parameters that need to be changed in order to shift the system to a new steady state corresponding to the desired arterial blood pressure and heart rate.

Since heart rate is one of the model parameters (and hence one can set it directly to the desired value), the problem was how to obtain the required steady-state arterial blood pressure. Any steady state of the cardiovascular system is associated with a certain blood flow which must be equal to the output of both right and left heart ventricles. Therefore, in order to shift the system to a new steady state, one has to modify somehow the cardiac output. This can be done by changing the parameters of the relationships between the stroke volume and atrial pressure for right and left ventricles (equations 1 and 2 representing the Frank-Starling law of the heart).

We decided to keep the shape of both stroke volume curves unchanged ie. to keep the slopes (parameters s) and the maximal values (parameters SV_{max}) of both sigmoidal functions at the original level corresponding to the reference patient. To modify the steady state of the system we propose to shift horizontally the right ventricular stroke volume curve by changing the parameter x_r of the Frank-Starling relationship of the right heart (equation 1). Shifting the stroke

volume curve of the right ventricle results in a significant change of the output of the right ventricle. This in turn affects the amount of blood remaining in the right atrium and the amount of blood entering pulmonary arteries. Consequently, it affects blood pressure in right atrium and pulmonary arteries, which in turn affect the blood flow to and from the adjacent compartment and hence the blood flow in all other blood compartments across the whole cardiovascular system. These transient blood flow conditions continue until a new steady state is reached with a new level of blood flow across the system corresponding to the new cardiac output (as determined by the modified stroke volume curve). Using a Matlab built-in function *fminsearch*, which uses the simplex search method [25], we are able to find the value of parameter x_r needed to obtain the desired arterial blood pressure.

Hence, by changing directly only two model parameters i.e. heart rate and parameter x_r we are able to shift the system to a new steady state in which the cardiac output and the blood pressures, blood volumes and resistances of all compartments are modified so that the arterial blood pressure reaches the desired level.

The only other model parameters that can be easily adjusted for the given patient, are the parameters expressed per kg of body weight which can be scaled to the weight of the given patient. These include: nominal vascular compliances, total blood volume or cardiac output sensitivity to right and left atrial pressure changes. All other model parameters are assumed to remain at the original level corresponding to the physiology of the reference patient.

3.2 Case study

Below we present an example of shifting the system from the original steady state for the reference patient with the arterial blood pressure P_{sa} = 90 mm Hg and the heart rate f = 75 bpm to a new steady state corresponding to the arterial blood pressure P_{sa} = 110 mm Hg and the heart rate f = 70 bpm. As described above, to reach the new steady state we changed directly the value of heart rate (from 75 to 70 bpm) and we changed the value of parameter x_r (from -1.71 to 3.05), which corresponded to a horizontal shift of the right ventricular stroke volume curve, as shown in Figure 3.

The original steady-state conditions for the reference patient were as follows: right atrial pressure P_{ra} = 2 mm Hg, left atrial pressure P_{la} = 5 mm Hg, stroke volume SV = 74.67 ml, cardiac output q = 93.33 ml/s ($q = SVf$). The new steady-state conditions are as follows: right atrial pressure P_{ra} = -1.62 mm Hg, left atrial pressure P_{la} = 6.93 mm Hg, stroke volume SV = 105.55 ml, cardiac output q = 123.15 ml/s.

Figure 4 shows the operating points on the pressure-volume curves for each cardiovascular compartment before and after the changes. In this example we assumed that the new patient has the same weight as the reference patient (70 kg) and hence all model parameters (except heart rate and x_r) are the same as for the reference patient (based on the literature data). In particular, the

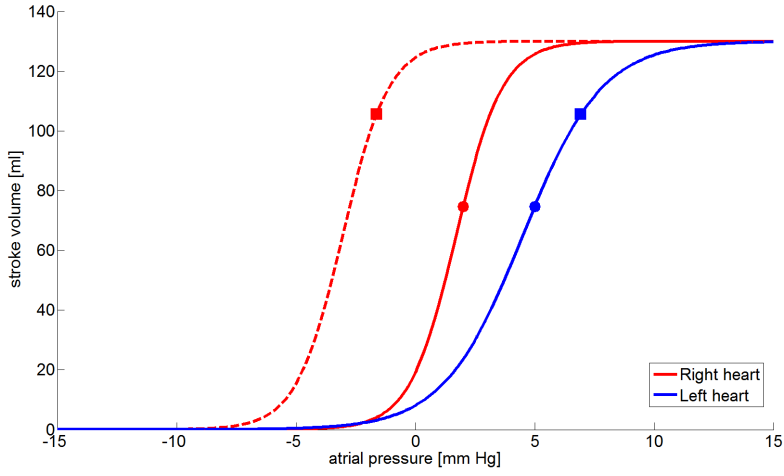


Fig. 3. Frank-Starling relationship between the stroke volume and atrial pressure modelled for a) the reference patient (solid lines with the dots representing the original steady-state conditions), b) the new patient (dashed lines with the squares representing the new steady-state conditions).

parameters of the pressure-volume curves of all cardiac and vascular compartments (ie. unstressed volumes, maximal volumes etc.) are exactly the same as for the reference patient, hence the new steady state corresponding to the new patient is obtained by forcing the model to change the operating points on all pressure-volume curves (ie. to change accordingly the volumes of blood stored in each compartment).

Finally, Figure 5 shows the simulation of the haemodynamic response to the typical 15-s VM (with the intrathoracic pressure increased to 40 mm Hg) starting from both the original steady state of the cardiovascular system in the reference patient and from the new steady state.

3.3 Technical considerations

In order to reach the desired steady state of the cardiovascular system, the baroreflex mechanisms and the stroke volume dependence on afterload have to be temporarily switched off in the model, as these mechanisms depend on the reference pressures set originally for the reference patient. After finding the new steady state of the system and the corresponding pressures and volumes of each compartment, the new steady-state pressures may be then used as the new reference pressures for these mechanisms (aortic, arterial and right atrial pressures for the baroreflex mechanisms; pulmonary arterial and aortic pressures for the afterload impact on the stroke volume of right and left heart respectively), assuming that these pressures represent the normal values for the given patient.

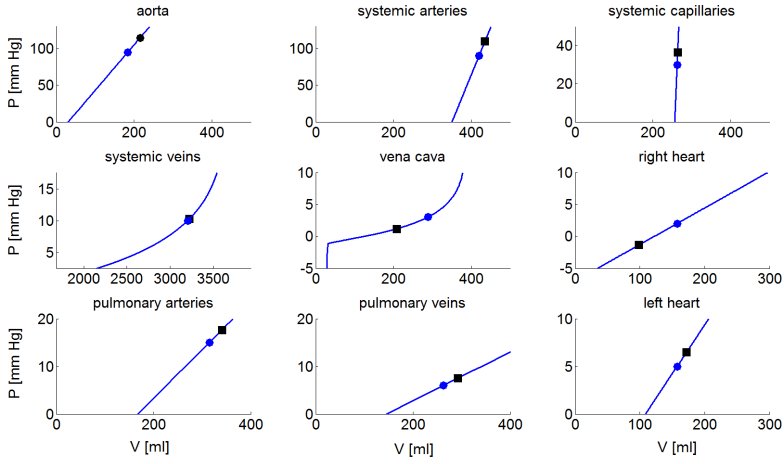


Fig. 4. Operating points on the cardiovascular pressure-volume curves for the original steady-state conditions for the reference patient (blue dots) and for the new steady-state conditions (black squares). The parameters of all pressure-volume curves remained unchanged.

Alternatively, in order not to switch off the baroreflex mechanisms or the impact of afterload, in each modelling step during the transient conditions one could take the current blood pressures as the normal values, thus indicating no deviations of pressure from the normal levels and, hence, effectively disabling the regulatory mechanisms. This approach leads to the same result, however, it is associated with a higher computational cost.

Following the change of heart rate, the minimal and maximal heart rate admissible by the baroreflex control of heart will also change accordingly (these parameters are calculated in the model so that in the steady state conditions the baroreflex operates in the middle point of the sigmoidal curve) [18].

Note that, instead of changing the parameter x_r of the Frank-Starling relationship for the right heart (ie. moving horizontally the curve relating the right ventricular stroke volume to the right atrial pressure), one could change the parameter x_l of the analogous relationship for the left heart. In this study, the former has been changed based on the assumption that any changes in the systemic arterial pressure (with respect to the reference patient) will affect more the right atrial pressure than the left atrial pressure.

As shown in Figure 6, the magnitude of hemodynamic response to the VM simulated in the model depends strongly on the initial state of the system and hence shifting the cardiovascular system to the steady-state conditions corresponding to the mean arterial pressure of the given patient is crucial for simulating the response to the VM. Obviously, the same holds for estimating some physiological parameters of the given patient based on the recorded data on arterial blood pressure and heart rate variations in response to the VM.

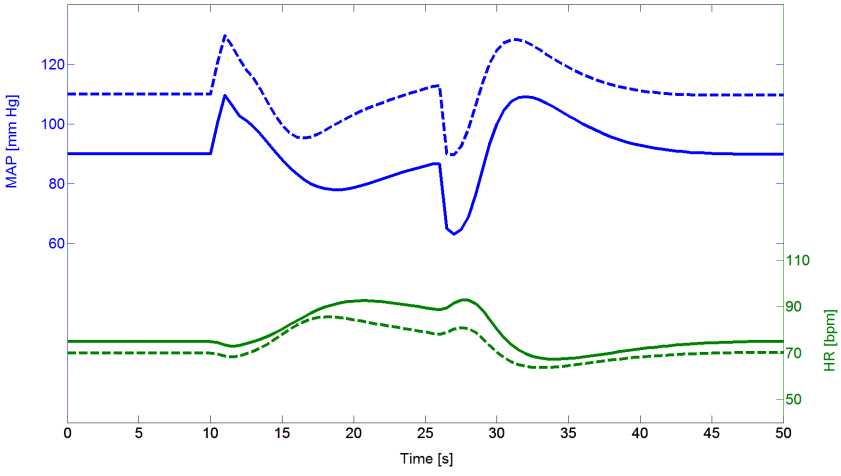


Fig. 5. Simulation of mean arterial blood pressure (MAP) and heart rate (HR) variations in response to the 15-s Valsalva manoeuvre with the intrathoracic pressure increased to 40 mm Hg modelled from the original steady state of the cardiovascular system in the reference patient (solid lines) and from the modified steady state for the given patient (dashed lines).

3.4 Discussion

One should be aware that the new steady state of the system reached using the above approach reflects only the arterial pressure and heart rate of the given patient and most likely does not represent correctly the actual state of all cardiovascular compartments in the given patient and hence such an approach is not ideal. In particular, the new value of cardiac output (equal to the blood flow across the system in the new steady state) will likely not match the real cardiac output of the given patient. Knowing the real cardiac output of the patient (which can be measured invasively or estimated non-invasively), one could change accordingly the values of both x_r and x_l parameters, thus having a better representation of the cardiac function in the modelled patient and reaching the correct steady-state cardiac output. Similarly, knowing the central venous pressure (which again can be measured invasively or estimated non-invasively), one could also change the parameter x_l so that the new steady-state venous pressure in the model would correspond to the measured value.

As far as the parameters scaled to patient's weight are concerned, in the future versions of the model, some of these parameters could depend not only on the weight of the given individual, but also on other anthropometrics, such as height or age. Some parameters, such as the maximal stroke volume (SVmax), could also depend on the physical fitness of the given individual.

All other model parameters (except x_r), as well as all per-kg values of the scalable parameters remain at the levels set for the reference patient (based on

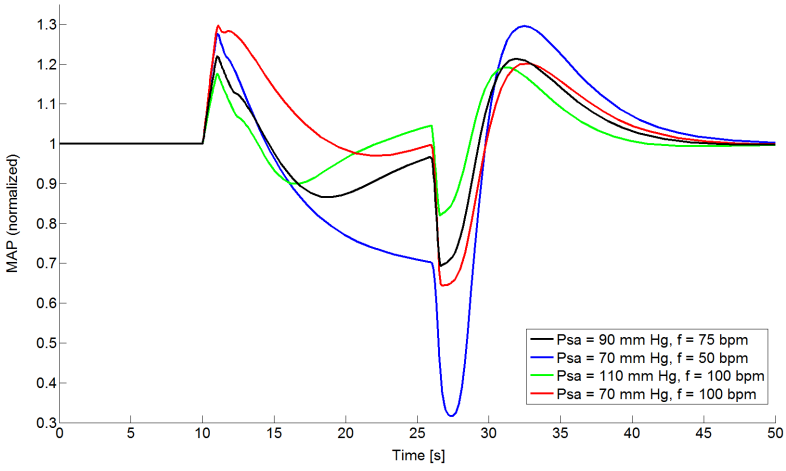


Fig. 6. Model simulations of the normalized mean arterial pressure during the Valsalva manoeuvre in patients with different cardiovascular steady-state conditions.

typical physiological data from the literature), unless one has some particular information or data on the given patient, which would allow changing individual parameters (e.g. decreased vascular compliance due to atherosclerosis).

We would like to point out also that using the presented approach we are not able to shift the cardiovascular system to all possible states. For instance there is an upper limit of the steady-state mean arterial pressure that one can reach in the model using the presented method. This is related to the closed nature of the cardiovascular system. When right and left ventricular outputs are initially increased (following the changes of stroke volume curves), the arterial pressure and the volume of blood in the arterial compartment increase as well. This means that the volume of blood on the low-pressure side of the system (ie. veins, vena cava, right atrium) must decrease and hence the right atrial pressure decreases. This in turn increases the pressure difference between the arteries and right atrium and hence increases the blood flow in the system (which is needed to keep the arterial pressure high). However, lowering right atrial pressure results in reducing the output of right heart ventricle (according to the Frank-Starling law, see equation 1 or Figure 3) which cannot keep up with the high blood inflow to the heart and hence the right atrial pressure starts to increase, which re-increases the right ventricular output, but also reduces the amount of blood in the arteries. This continues until the system finds a steady state and hence there is a maximal arterial pressure that can be reached (approximately 125 mm Hg).

Also, changing significantly the right ventricular stroke volume curve (moving it far left along the atrial pressure axis) is associated with increasing the

right ventricular output to a very high level. Depending on the initial blood distribution in the system this can quickly lead to emptying of right heart, which can compromise the computational efficiency of the model.

The aforementioned parameter SV_{max} (maximal ventricular stroke volume) may also pose limitation in some cases. For instance, it would be impossible to reach a steady state with a very high mean arterial pressure and a low heart rate without changing SV_{max}, as the cardiac output is the product of stroke volume and heart rate and hence it is upper bounded by $SV_{max}f$.

Nevertheless, in most cases the above limitations do not apply and the presented method is effective in the wide range of steady-state conditions of the cardiovascular system. Using this method we were able to simulate haemodynamic response to the VM in several patients (as described in [18]) without any problems. If, however, a need arises to shift the system to an extreme steady state (e.g. a very high mean arterial pressure), one would need to change some other model parameters (for instance, the initial blood distribution across the system).

4 Conclusions

We presented a method of employing our mathematical model for simulating the haemodynamic response to the Valsalva manoeuvre in a given individual. The same method could be used for comparing the simulation results with experimental data (ie. recorded blood pressure and heart rate variations in response to the manoeuvre) to estimate some physiological parameters of the given patient.

The presented method enables simulation of the cardiovascular system of a given subject starting from the steady-state conditions corresponding to the real arterial blood pressure and heart rate of the patient. We acknowledge the fact that this method is not ideal as it uses only two measured parameters, while assuming all other parameters at the standard physiological level (as reported in the literature). As already indicated, measuring additional parameters (such as cardiac output or central venous pressure) could obviously provide better results. Moreover, given any additional information on patient conditions, one can always adjust accordingly the corresponding model parameters to improve modelling accuracy.

Acknowledgements

This work was supported by the European Union from resources of the European Social Fund project PO KL „Information technologies: Research and their interdisciplinary applications" (agreement no. UDA-POKL.04.01.01-00-051/10-00).

References

1. Sharpey-Schafer, E.: Effects of valsalva's manoeuvre on the normal and failing circulation. Br Med J **1**(4915) (1955) 693–695
2. Zema, M., Restivo, B., Sos, T., Sniderman, K., Kline, S.: Left ventricular dysfunction - bedside valsalva monoeuvre. Br Heart J **44**(5) (1980) 560–569
3. Di Tullio, M., Sacco, R., Venketasubramanian, N., Sherman, D., Mohr, J., Homma, S.: Comparison of diagnostic techniques for the detection of a patent foramen ovale in stroke patients. Stroke **24**(7) (1993) 1020–1024
4. Nishimura, R., Tajik, A.: The valsalva maneuver-3 centuries later. Mayo Clin Proc **79**(4) (2004) 577–578
5. Ferguson 3rd, J., Miller, M., Aroesty, J., Sahagian, P., Grossman, W., McKay, R.: Assessment of right atrial pressure-volume relations in patients with and without an atrial septal defect. J Am Coll Cardiol **13**(3) (1989) 630–636
6. Baldwa, V., Ewing, D.: Heart rate response to valsalva manoeuvre. reproducibility in normals, and relation to variation in resting heart rate in diabetics. Br Heart J **39**(6) (1977) 641–644
7. Levin, A.: A simple test of cardiac function based upon the heart rate changes induced by the valsalva maneuver. Am J Cardiol **18**(1) (1966) 90–99
8. Zollei, E., Paprika, D., Rudas, L.: Measures of cardiovascular autonomic regulation derived from spontaneous methods and the valsalva maneuver. Auton Neurosci **103**(1-2) (2003) 100–105
9. Palmero, H., Caeiro, T., Iosa, D., Bas, J.: Baroreceptor reflex sensitivity index derived from phase 4 of te valsalva maneuver. Hypertension **3**(6 (Pt2)) (1981) II–134–137
10. Looga, R.: Valsalva manoeuvre–cardiovascular effects and performance technique: a critical review. Respir Physiol Neurobiol **147**(1) (2005) 39–49
11. Junqueira, Jr, L.: Teaching cardiac autonomic function dynamics employing the valsalva. Adv Physiol Educ **32**(1) (2008) 100–106
12. Pstraś, L., Thomaseth, K., Waniewski, J., Balzani, I., Bellavere, F.: The valsalva manoeuvre: physiology and clinical examples (in press). Acta Physiol (Oxf) (2015)
13. Eckberg, D.: Parasympathetic cardiovascular control in human disease: a critical review of methods and results. Am J Physiol **239**(5) (1980) H581–93
14. Looga, R.: Reflex cardiovascular responses to lung inflation: a review. Respir Physiol **109**(2) (1997) 95–106
15. Mateika, J., Demeersman, R., Kim, J.: Effects of lung volume and chemoreceptor activity on blood pressure and r-r interval during the valsalva maneuver. Clin Auton Res **12**(1) (2002) 24–34
16. Korner, P., Tonkin, A., Uther, J.: Reflex and mechanical circulatory effects of graded valsalva maneuvers in normal man. J Appl Physiol **40**(3) (1976) 434–440
17. Hamilton, W., Woodbury, R., Harper, Jr, H.: Physiologic relationships between intrathoracic, intraspinal and arterial pressures. JAMA **107**(11) (1936) 853–856
18. Pstraś, L., Thomaseth, K., Waniewski, J., Balzani, I., Bellavere, F.: Mathematical modelling of cardiovascular response to the valsalva manoeuvre (submitted). Math Med Biol
19. Lu, K., Clark, J., Ghorbel, F., Ware, D., Bidani, A.: A human cardiopulmonary system model applied to the analysis of the valsalva maneuver. Am J Physiol Heart Circ Physiol **281** (2001) H2661–H2679
20. Liang, F., Liu, H.: Simulation of hemodynamic responses to the valsalva maneuver: An integrative computational model of the cardiovascular system and the autonomic nervous system. J Physiol Sci **56**(1) (2006) 45–65

21. Rothe, C.: Reflex control of veins and vascular capacitance. Physiol Rev **63**(4) (1983) 1281–1342
22. Elad, D., Einav, S.: Physical and flow properties of blood. in: biomedical engineering and design handbook 2nd edn 1. McGraw-Hill (2009)
23. Greene, A., Shoukas, A.: Changes in canine cardiac function and venous return curves by the carotid baroreflex. Am J Physiol **251**(2) (1986) H288–96
24. Sun, Y., Beshara, M., Lucariello, R., Chiaramida, S.: A comprehensive model for right-left heart interaction under the influence of pericardium and baroreflex. Am J Physiol **272**(3 (Pt 2)) (1997) H1499–1515
25. Lagarias, J., Reeds, J., Wright, M., Wright, P.: Convergence properties of the nelder-mead simplex method in low dimensions. SIAM Journal of Optimization **9**(1) (1998) 1112–147

Boosting Techniques for Uplift Modelling

Michał Sołtys[1] and Szymon Jaroszewicz[1,2]

[1] Institute of Computer Science, Polish Academy of Sciences,
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
[2] National Institute of Telecommunications
ul. Szachowa 1, 04-894 Warsaw, Poland

Abstract. Predicting causal effects of actions taken was always one of the most important aims of human reasoning. Every human action is meant to increase probability of desired circumstances and reduce risks of unwanted ones. Actually, people seem to reason in the following way: if probability of desired outcome after a given action is high enough, it is worth trying. Yet prior chances of success (if action not adopted) are completely ignored, perhaps assumed to be negligibly small.
Unfortunately, this approach suffers from serious drawbacks. Consider for example a typical marketing campaign. Conducted on small random sample of customers, it is used to evaluate a probability of purchase (responding to a campaign) after the action was performed. Then a classification model is built to pick a group of customers, to which the campaign should be addressed. We achieve a model targeting customers most likely to buy *after* the campaign. But this is not what a marketer wants. Some of the customers would have bought regardless of the campaign, targeting them brought unnecessary costs. Other customers were actually going to make a purchase but were annoyed by the campaign. It is a well known phenomenon in the marketing literature; the result is a loss of a sale or even a complete loss of the customer (churn).
We should rather select customers who will buy *because* of the campaign, that is, those who are likely to buy if targeted, but unlikely to buy otherwise. Only then we actually can focus on performing the action to increase our chances, not just act when these chances are relatively high anyway. Notice also that similar problems arise in medicine where some patients may recover without actually being treated and some may be hurt by the therapy's side effects more than by the disease itself.
Uplift modelling provides a solution to the described problem. The approach uses two separate training sets: *treatment* and *control*. Individuals in the treatment group are subjected to the action, such as a medical treatment or a marketing campaign. The control dataset contains objects which are not subjected to the action and serve as a background against which its effect can be assessed. Instead of modelling class probabilities, uplift modelling attempts to model the *difference* between conditional class probabilities in the treatment and control groups. This way, the causal influence of the action can be modelled, and the method is able to

predict the true gain (with respect to taking no action) from targeting a given individual.
As the uplift approach is being developed and increasingly appears to be a prospective methodology, the need for more sophisticated tools becomes natural. In the case of classification, apart from more and better algorithms appearing, a hugely important milestone has been the invention of ensemble methods which strengthen existing classification algorithms. This powerful procedures allow to improve performance of many classifiers in a general way, often turning weak single models into highly capable ensembles. It becomes clear then, that a search for an uplift analogue of ensemble methods is needed.
We consider a few methods of applying an idea of the boosting procedure to an uplift approach. These are: a double (classifier) boosting approach being a natural way of implementing uplift boosting; a class variable transformation allowing for application of any ordinary classifiers to uplift modelling, and Uplift AdaBoost being a new algorithm for uplift modelling which realizes one of the basic assumptions of classic boosting: forgetting the last member added to the ensemble in each iteration.
We focus on the mechanism, used in classical boosting, of updating record weights such that its classification error is exactly $1/2$ after each iteration, which makes it likely for the next member to be very different from the previous one, leading to a diverse ensemble.
Implementation of this feature, known as forgetting the last member of the ensemble, is significantly more complex than in classification case. Since we have two datasets, treatment and control, reweighting instances can be done in infinite number of ways. Unlike the classification boosting, we have now two classification accuracies in each iteration, which should be used in establishing model weights; this makes the problem more challenging.
We construct an uplift AdaBoost algorithm preserving the feature of forgetting by setting weight update parameters for treatment and control datasets as well as model weights for each iteration in a way which guarantees convergence. We discuss analogies and dissimilarities between classification and uplift boosting algorithms, including theoretical properties and practical consequences.
We perform an experimental evaluation that demonstrate the usefulness of the methods considered. We compare their performance and performance of the base models on benchmark datasets. A proposed uplift boosting methods often dramatically improve performance of the base models and are thus new and powerful tools for uplift modelling.

1 Introduction

The main interest of machine learning is the problem of classification, where the task is to predict, based on a number of attributes, the class to which an instance belongs, or the conditional probability of it belonging to each of the classes. Unfortunately, classification is not well suited to many problems in marketing

or medicine to which it is applied. Let us discuss it on the example of a direct marketing campaign where potential customers receive a mailing offer.

A typical application of machine learning techniques in this context involves selecting a small pilot sample of customers who receive the campaign. Next, a classifier is built based on the pilot campaign outcomes and used to select customers to whom the offer should be mailed. As a result, the customers most likely to buy *after* the campaign will be selected as targets.

Unfortunately this is not what a marketer wants! Some of the customers would have bought regardless of the campaign; targeting them resulted in unnecessary costs. Other customers were actually going to make a purchase but were annoyed by the campaign. The result is a loss of a sale or even a complete loss of the customer (churn). While the second case may seem unlikely, it is a well known phenomenon in the marketing community [1, 2].

In order to run a truly successful campaign, we need, instead, to be able to select customers who will buy *because* of the campaign, i.e., those who are likely to buy if targeted, but unlikely to buy otherwise. Similar problems arise in medicine where some patients may recover without actually being treated and some may be hurt by the therapy's side effects more than by the disease itself.

Uplift modelling provides a solution to this problem. The approach employs two separate training sets: *treatment* and *control*. The objects in the treatment dataset have been subject to some action, such as a medical treatment or a marketing campaign. The control dataset contains objects which have not been subject to the action and serve as a background against which its effect can be assessed. Instead of modelling class probabilities, uplift modelling attempts to model the *difference* between conditional class probabilities in the treatment and control groups. This way, the *causal* influence of the action can be modelled, and the method is able to predict the true gain (with respect to taking no action) from targeting a given individual.

While in described problems uplift modelling is a better alternative for standard classification, we should expect a dynamic development of the approach. Yet, despite its practical appeal, uplift modelling has received surprisingly little attention in the literature. There are, however, papers concerning uplift models and successful applications to practical problems, especially in marketing, are reported. An American bank used uplift modelling to turn an unsuccessful mailing campaign into a profitable one [3]. Applications have also been reported in minimizing churn at mobile telecoms [4]. In [5] an approach to online advertising has been proposed which combines uplift modelling with maximizing the response rate in the treatment group to increase advertiser's benefits.

Although there would not be any reservations to use an algorithm to choose who should receive an advert or some marketing campaign, leaving a decision on treatment to some statistical procedure may seem too controversial in medicine. Still, doctors may be interested in factors indiated by the model to be responsible for chances of recovery after the treatment was applied. What is more, uplift modelling allows for any arbitrary number of factors, unlike typical medical trials with control groups.

As uplift approach is developed and seems to be a prospective methodology, a need for more sophisticated tools become natural. As it was in the case of classification, apart from more and better algorithms appearing, there was a marvelous milestone done: ensemble methods were invented to strengthen all existing classification algorithms. This powerful procedures allow to improve performance of any classifier in a generic way, often turning weak single models into highly capable ensembles. It becomes clear then, that search for uplift analogon of ensemble methods is needed.

This paper presents an adaptation of AdaBoost algorithm to the uplift modelling case. Boosting often dramatically improves performance of classification models, and in this paper we demonstrate that it can bring similar benefits to uplift modelling. We apply forgetting the last member of the ensemble to the described problem, trying to repeat the success of the classical algorithm in the uplift case. Experimental verification proves that the benefits of boosting extend to the case of uplift modelling and shows relative merits of the new approach.

In the remaining part of this section we introduce a definition of an uplift analogue of classification error and present two alternative ways to apply boosting procedures to the uplift case: a class variable transformation and a double classifier approach. We give an overview of the other related work and remind the property of forgetting the last member of the ensemble in classification boosting. But first we have to start with introducing a notation used throughout the paper.

1.1 Notation

We will now introduce the notation used further in the article. We use the superscript ^{T} for quantities related to the treatment group and the superscript ^{C} for quantities related to the control group. For example, the treatment training dataset will be denoted with $\mathcal{D}^T$ and the control training dataset with $\mathcal{D}^C$. Both datasets together constitute the whole training dataset, $\mathcal{D} = \mathcal{D}^T \cup \mathcal{D}^C$.

Each data record (x, y) consists of a vector of features $x \in \mathcal{X}$ and a class $y \in \{0, 1\}$ with 1 assumed to be the successful outcome, for example patient recovery or a positive response to a marketing campaign. Let N^T and N^C denote the number of records in the treatment and control datasets.

An uplift model is a function $h : \mathcal{X} \to \{0, 1\}$. The value $h(x) = 1$ means the action is deemed beneficial for x by the model, $h(x) = 0$ means that its impact is considered neutral or negative. By 'positive outcome' we mean that the probability of success for a given individual x is higher if the action is performed on her than if the action is not taken.

We will denote general probabilities related to the treatment and control groups with P^T and P^C, respectively. For example, $P^T(y = 1, h = 1)$ stands for probability that a randomly selected case in the treatment set has a positive outcome and taking the action on it is predicted to be beneficial by an uplift model h. We can now state more formally when an individual x should be subject to an action, namely, when $P^T(y = 1|x) - P^C(y = 1|x) > 0$.

In the m-th step of the boosting algorithm the i-th treatment group training record is assumed to have a weight $w^T_{m,i}$ assigned to it. Likewise a weight $w^C_{m,i}$ is assigned to the i-th control training case. Further, denote by

$$p^T_m = \frac{\sum_{i=1}^{N^T} w^T_{m,i}}{\sum_{i=1}^{N^T} w^T_{m,i} + \sum_{i=1}^{N^C} w^C_{m,i}}, \qquad p^C_m = \frac{\sum_{i=1}^{N^C} w^C_{m,i}}{\sum_{i=1}^{N^T} w^T_{m,i} + \sum_{i=1}^{N^C} w^C_{m,i}} \tag{1}$$

the relative sizes of treatment and control datasets at step m. Notice that $p^T_m + p^C_m = 1$ for every m.

1.2 An uplift analogue of classification error

We begin with mentioning a problem which is the biggest challenge of uplift modelling as opposed to standard classification. The problem has been known in statistical literature (see [6]) as the

> **Fundamental Problem of Causal Inference.** For every individual, only one of the outcomes is observed, after the individual has been subject to an action (treated) or when the individual has not been subject to the action (was a control case), *never* both.

As a result we never know whether the action performed on a given individual was truly beneficial. This is different from classification, where the true class of each individual in the training set is known.

Due to the Fundamental Problem of Causal Inference we cannot tell whether an uplift model correctly classified a given instance. We will, however, define an approximate notion of classification error in the uplift case. A record (x^T_i, y^T_i) is assumed to be classified correctly by an uplift model h if $h(x^T_i) = y^T_i$ and $(x^T_i, y^T_i) \in \mathcal{D}^T$; a record (x^C_i, y^C_i) is assumed to be classified correctly if $h(x^C_i) = 1 - y^C_i$ and $(x^C_i, y^C_i) \in \mathcal{D}^C$.

Intuitively, if a record (x^T_i, y^T_i) belongs to the treatment group and a model h predicts that it should receive the treatment $(h(x^T_i) = 1)$ then the outcome should be positive $(y^T_i = 1)$ if the recommendation is to be correct. Note that the gain from the action might also be neutral if a success would have occurred also without treatment, but at least the model's recommendation is not in contradiction with the observed outcome. If, on the contrary, the outcome for a record in the treatment group is 0 and $h(x^T_i) = 1$, the prediction is clearly wrong as the true effect of the action can at best be neutral.

In the control group the situation is reversed. If the outcome was positive $(y^C_i = 1)$ but the model predicted that the treatment should be applied $(h(x^C_i) = 1)$, the prediction is clearly wrong, since the treatment cannot be truly beneficial, it can at best be neutral. To simplify notation we will introduce

the following indicators:

$$e^T(x_i^T) = \begin{cases} 0 & \text{if } x_i^T \in \mathcal{D}^T \text{ and } h(x_i^T) = y_i^T, \\ 1 & \text{if } x_i^T \in \mathcal{D}^T \text{ and } h(x_i^T) \neq y_i^T, \end{cases} \tag{2}$$

$$e^C(x_i^C) = \begin{cases} 0 & \text{if } x_i^C \in \mathcal{D}^C \text{ and } h(x_i^C) \neq y_i^C, \\ 1 & \text{if } x_i^C \in \mathcal{D}^C \text{ and } h(x_i^C) = y_i^C. \end{cases} \tag{3}$$

An index m will be added to indicate the m-th step of the algorithm. Let us now define uplift analogues of classification error on the treatment and control datasets and a combined error:

$$\epsilon_m^T = \frac{\sum_{i:\ e_m^T(x_i)=1} w_{m,i}^T}{\sum_{i=1}^{N^T} w_{m,i}^T}, \quad \epsilon_m^C = \frac{\sum_{i:\ e_m^C(x_i)=1} w_{m,i}^C}{\sum_{i=1}^{N^C} w_{m,i}^C}, \quad \epsilon_m = p_m^T \epsilon_m^T + p_m^C \epsilon_m^C. \tag{4}$$

The sums above are a shorthand notation for summing over misclassified instances in the treatment and control training sets, which will also be used later in the paper.

1.3 Double classifiers

The most obvious approach to uplift modelling is to build two classification models h^T and h^C on the treatment and control groups respectively and to subtract their predicted probabilities:

$$h^U(\mathbf{x}) = h^T(\mathbf{x}) - h^C(\mathbf{x}).$$

We will call this approach the *double classifier* approach. Its obvious appeal is simplicity; however in many cases the approach may perform poorly. The reason is that both models can focus on predicting the class probabilities themselves, instead of making the best effort to predict the (usually much weaker) 'uplift signal', i.e., the difference between conditional class probabilities in the treatment and control groups. See [2] for a detailed discussion and an illustrative example[1]. Nevertheless, in some cases the approach is competitive. This is the case when the amount of training data is large enough to accurately estimate conditional class probabilities in both groups or when the net gain is correlated with the class variable, e.g. when people likely to buy a product are also likely to positively respond to a marketing offer related to that product.

1.4 Class variable transformation

In [7] a class variable transformation was presented which allows for converting an arbitrary classification model (the paper used logistic regression) into an

[1] The example is based on artificial data with two attributes, one strongly affecting the class probabilities independently from the treatment received, the other determining the relatively small sensitivity to the treatment. A model based on two decision trees uses only the first attribute.

uplift model. The transformation simply replaces class values y_i^C in the control group with their reverses $1 - y_i^C$ while keeping the treatment set class values unchanged. As a result, a single classifier is built which directly models the difference between success probabilities in the treatment and control groups. It is easy to see that the errors defined in Equation 4 are equivalent to standard classification errors for the transformed class.

1.5 Other related work

Despite its practical appeal, uplift modelling has seen relatively little attention in the literature. Here we shortly discuss some other work not mentioned above.

Several algorithms have thus been proposed which directly model the difference between class probabilities in the treatment and control groups. Many of them are based on modified decision trees. For example, [2] describe an uplift tree learning algorithm which selects splits based on a statistical test of differences between treatment and control class probabilities. In [8, 9] uplift decision trees based on information theoretical split criteria have been proposed.

Some work has also been published on using ensemble methods for uplift modelling, although, to the best of our knowledge, none of them on boosting. Bagging of uplift models has been mentioned in [2]. Uplift Random Forests have been proposed by [10]; an extension, called causal conditional inference trees was proposed by the same authors in [11]. A thorough experimental and theoretical analysis of bagging and random forests in uplift modelling can be found in [12] where it is argued that ensemble methods are especially well suited to this task and that bagging performs surprisingly well.

Other uplift techniques have also been proposed. Regression based approaches can be found in [13] or, in a medical context, in [14, 15].

[16] proposes a method for converting survival data such that uplift modelling can, under certain assumptions, be directly applied to it.

Some variations on the uplift modelling theme have also been explored. [5] proposed an approach in the context of online advertising, where it is necessary to not only maximize the net gain, but also to increase advertiser's benefits through maximizing response rate in the treatment group. This type of problems are beyond the scope of this paper.

1.6 Forgetting in classical AdaBoost

While many boosting algorithms are available, in this paper by 'boosting' we mean the discrete AdaBoost algorithm [17]. Forgetting the last member added to the ensemble means that after a new member is added, record weights are updated such that its classification error is exactly 1/2. This makes it likely for the next member to be very different from the previous one, leading to a diverse ensemble. Full details can be found for example in [17–19]. This key property will be important for adapting boosting to the uplift modelling case.

Now we can formulate an uplift analogon of AdaBoost algorithm.

2 Uplift AdaBoost

In this section we present the proposed algorithm and the property of forgetting the last ensemble member in the context of uplift modelling.

2.1 Algorithm

Algorithm 1 presents AdaBoost algorithm for uplift modelling.

Input: set of treatment training records, $\mathcal{D}^T = \{(x_1^T, y_1^T), \ldots, (x_{N^T}^T, y_{N^T}^T)\}$,
set of control training records, $\mathcal{D}^C = \{(x_1^C, y_1^C), \ldots, (x_{N^C}^C, y_{N^C}^C)\}$,
base uplift algorithm to be boosted,
integer M specifying the number of iterations

1. Initialize weights $w_{1,i}^T, w_{1,i}^C$
2. For $m \leftarrow 1, \ldots, M$
   (a) $w_{m,i}^T \leftarrow \frac{w_{m,i}^T}{\sum_j w_{m,j}^T + \sum_j w_{m,j}^C}$; $w_{m,i}^C \leftarrow \frac{w_{m,i}^C}{\sum_j w_{m,j}^T + \sum_j w_{m,j}^C}$
   (b) Build a base model h_m on $\mathcal{D}$ with $w_{m,i}^T, w_{m,i}^C$
   (c) Compute the treatment and control errors ϵ_m^T, ϵ_m^C
   (d) Compute $\beta_m = \frac{p_m^T \epsilon_m^T + p_m^C \epsilon_m^C}{1 - p_m^T \epsilon_m^T - p_m^C \epsilon_m^C}$
   (e) If $\beta_m = 1$ or $\epsilon_m^T \notin (0, \frac{1}{2})$ or $\epsilon_m^C \notin (0, \frac{1}{2})$:
      i. choose random weights $w_{m,i}^T$, $w_{m,i}^C$
      ii. continue with next boosting iteration
   (f) $w_{m+1,i}^T \leftarrow w_{m,i}^T \cdot (\beta_m)^{1[h_m(x_i^T) = y_i^T]}$
   (g) $w_{m+1,i}^C \leftarrow w_{m,i}^C \cdot (\beta_m)^{1[h_m(x_i^C) = 1 - y_i^C]}$
   (h) Add h_m with coefficient β_m to the ensemble

Output: The final hypothesis

$$h_f(x) = \begin{cases} 1 \; if \; \sum_{m=1}^M \left(\log \frac{1}{\beta_m}\right) h_m(x) \geq \frac{1}{2} \sum_{m=1}^M \log \frac{1}{\beta_m}, \\ 0 \; otherwise. \end{cases} \tag{5}$$

Algorithm 1: AdaBoost algorithm for uplift.

Note that the algorithm is a discrete boosting algorithm [17, 19], that is, the base learners are assumed to return a discrete decision on whether the action should be taken (1) or not (0). Algorithm 1, as presented in the figure, also returns a decision. However, it can also return a numerical score,

$$s(x) = \sum_{m=1}^{M} \left(\log \frac{1}{\beta_m}\right) h_m(x),$$

indicating how likely it is that the effect of the action is positive on a given case. In the experimental Section 3 we will use this variant of the algorithm.

AdaBoost can suffer from premature stops when the sum of weights of misclassified cases becomes 0 or is greater than 1/2. This problem turns to be even more troublesome in the uplift modelling case. Hence, in step 2e of Algorithm 1 we restart the algorithm by assigning random weights drawn from the exponential distribution to records in both training datasets. The technique has been suggested for classification boosting in [20].

2.2 Properties

Let us now examine what the property of forgetting the last model added to the ensemble means in the context of uplift error defined in Equation 4. To forget the member h_m added in step m we need to choose weights in step $m+1$ such that the combined error of h_m is exactly one half, $\epsilon_m = \frac{1}{2}$. From steps 2f and 2g of Algorithm 1 we get that to ensure the condition holds at step $m+1$, the following equation for β_m must be true:

$$\beta_m \sum_{i:\, e_m^T(x_i)=0} w_{m,i}^T + \beta_m \sum_{i:\, e_m^C(x_i)=0} w_{m,i}^C = \sum_{i:\, e_m^T(x_i)=1} w_{m,i}^T + \sum_{i:\, e_m^C(x_i)=1} w_{m,i}^C, \quad (6)$$

that is, the total new weights of correctly classified examples need to be equal to total new weights of incorrectly classified examples. After dividing both sides by $\sum_{i=1}^{N^T} w_{m,i}^T + \sum_{i=1}^{N^C} w_{m,i}^C$ the equation becomes

$$p_m^T(1-\epsilon_m^T)\beta_m + p_m^C(1-\epsilon_m^C)\beta_m = p_m^T\epsilon_m^T + p_m^C\epsilon_m^C. \quad (7)$$

Note that unlike classical boosting, this condition does not uniquely determine record weights.

Let us now give a justification of this condition in terms of performance of an uplift model.

Theorem 1. *Let h be an uplift model. If the balance condition holds and the assignment of cases to the treatment and control groups is random then the condition that the combined uplift error ϵ be equal to $\frac{1}{2}$ is equivalent to*

$$P(h=1)\left[P^T(y=1|h=1) - P^C(y=1|h=1)\right] + P(h=0)\left[P^C(y=1|h=0) - P^T(y=1|h=0)\right] = 0. \quad (8)$$

Proof. Note that the assumption of random group assignment implies $P^T(h=1) = P^C(h=1) = P(h=1)$ since both groups are scored with the same model and have the same distributions of predictor variables. Using the balance condition, the error ϵ of h, defined in Equation 4, can be expressed as (the second equality follows from $p^T = p^C = \frac{1}{2}$)

$$\begin{aligned} 2\epsilon &= 2P^T(h=1-y)p^T + 2P^C(h=y)p^C = P^T(h=1-y) + P^C(h=y) \\ &= P^T(h=1,y=0) + P^T(h=0,y=1) + P^C(h=y=0) + P^C(h=y=1) \\ &= P^T(y=0|h=1)P^T(h=1) + P^T(y=1|h=0)P^T(h=0) \\ &\quad + P^C(y=1|h=1)P^C(h=1) + P^C(y=0|h=0)P^C(h=0). \end{aligned}$$

Using the assumption of random treatment assignment and rearranging:

$$\begin{aligned}
&= P(h=1)\left[P^T(y=0|h=1) + P^C(y=1|h=1)\right] \\
&\quad + P(h=0)\left[P^T(y=1|h=0) + P^C(y=0|h=0)\right] \\
&= P(h=1)\left[1 - P^T(y=1|h=1) + P^C(y=1|h=1)\right] \\
&\quad + P(h=0)\left[P^T(y=1|h=0) + 1 - P^C(y=1|h=0)\right] \\
&= 1 + P(h=1)\left[-(P^T(y=1|h=1) - P^C(y=1|h=1))\right] \\
&\quad + P(h=0)\left[P^T(y=1|h=0) - P^C(y=1|h=0)\right].
\end{aligned}$$

After taking $\epsilon = \frac{1}{2}$ the result follows.

Note that the left term in (8) is the total gain in success probability due to the action being taken on cases selected by the model and the right term is the gain from not taking the action on cases not selected by the model. A good uplift model tries to maximize both quantities, so the sum being equal to zero corresponds to a model giving no overall gain over the controls.

When the balance condition holds, the forgetting property thus has a clear interpretation in terms of uplift model performance. When the balance condition does not hold, the interpretation is, at least partially, lost.

Note that β_m we choose:

$$\beta_m = \frac{p_m^T \epsilon_m^T + p_m^C \epsilon_m^C}{1 - (p_m^T \epsilon_m^T + p_m^C \epsilon_m^C)} \tag{9}$$

is identical to the result in classical boosting with the classification error being replaced by its uplift analogue.

3 Experimental evaluation

In this section we present an experimental evaluation of the three proposed algorithms and compare their performance with performance of the base models. We begin by describing the test datasets we are going to use, then review the approaches to evaluating uplift models and finally present the experimental results.

3.1 Benchmark datasets

A significant problem one encounters while working on uplift modelling is the lack of publicly available datasets. Even though control groups are ubiquitous in medicine and their use in marketing is growing, there are relatively few publicly available datasets which include a control group and a reasonable number of predictive attributes. In our experiments we are going to use datasets from the UCI repository artificially split into treatment and control groups. We describe

Table 1. Conversion of UCI datasets into treatment and control groups.

dataset	treatment/control split condition	#removed attributes / # original attributes
breast-cancer	menopause = 'PREMENO'	2/9
credit-a	a7 $\neq$ 'V'	3/15
dermatology	exocytosis ≤ 1	16/34
liver-disorders	drinks < 2	2/6
splice	attribute1 $\in \{$'A', 'G'$\}$	2/61
winequal-red	sulfur dioxide < 46.47	2/11

here the procedure used to split standard UCI datasets in a way suitable for uplift modelling. The details of the approach can be found in [8, 9].

The conversion is performed by first picking one of the data attributes which splits the data evenly into two groups. Details are given in Table 1. The first column contains the dataset name and the second provides the condition used to select records for the treatment group. The remaining records formed the control. A further postprocessing step removed attributes strongly correlated with the split itself; ideally, the division into treatment and control groups should be independent from all predictive attributes, but this is possible only in a controlled experiment. A simple heuristic was used for this purpose:

1. A numerical attribute was removed if its means in the treatment and control datasets differed by more than 25%.
2. A categorical attribute was removed if the probability of one of its categories differed between the treatment and control datasets by more than 0.25.

The number of removed attributes vs. the total number of attributes is shown in the third column of Table 1.

Further, multiclass problems were converted into binary problems with the majority class assumed to be class 1 (the desired outcome) and the remaining classes merged into class 0. We note that it is possible to use all analyzed uplift methods in the multiclass setting, however, we chose to use binarization in order to make the analysis (e.g. drawing curves) easier.

3.2 Methodology

Building uplift models requires two training sets. Consequently, we also have two test sets: treatment and control. A typical approach to assessing uplift models [2, 1] is to score both test datasets using the same uplift model and assume that objects in the treatment and control groups which have received similar scores are similar and can be compared with each other. In [1] the authors grouped treatment and control test cases by deciles of their scores and estimated net gains by subtracting success rates within each decile.

A more practical modification of this approach is to visualize model performance using *uplift curves* [8, 2]. Recall that one of the tools for assessing performance of standard classification models are lift curves[2], where the x axis corresponds to the number of cases subjected to an action and the y axis to the number of successes captured by the model.

In order to obtain an *uplift curve* we score both test sets using the uplift model and subtract the lift curve generated on the control test set from the lift curve generated on the treatment test set. The number of successes for both curves is expressed as percentage of the total population such that the subtraction is meaningful.

The interpretation of the uplift curve is as follows: on the x axis we select the percentage of the population on which the action is performed, and on the y axis we read the net gain achieved on the targeted group (the net gain on the remaining cases is zero since no action was performed on them). The point at $x = 100\%$ gives the gain in success probability we would obtain if the action was applied to the whole population. A diagonal uplift curve corresponds to performing the action on a randomly selected percentage of the population. More details can be found in [8, 2].

As with ROC curves, we can use the Area Under the Uplift Curve (AUUC) to summarize model performance with a single number. We subtract the area under the diagonal from this value in order to obtain more meaningful numbers. Note that the area under the uplift curve can be less than zero; this happens when the model gives high scores to cases for which the action has a predominantly negative effect.

All experiments have been performed by randomly splitting each dataset into training (80% of the data) and test (the remaining 20%) parts. Each experiment was repeated 128 times, and the resulting uplift curves have been averaged. The reason for this choice was to make the results repeatable and less sensitive to the random seed used. However, the disadvantage of such an approach is that it hides the variance of the predictions. To address this issue we also compute standard deviations of AUUCs computed over the 128 test sets in a manner similar to bootstrap estimates.

3.3 Experiments

As base models to be boosted we use two types of decision trees: unpruned J4.8 trees and decision stumps implemented in `Weka` package. We apply to them the three methods of boosting in the uplift approach: double (classical) boosting, class variable transformation and uplift AdaBoost algorithm proposed by us.

Thus we obtain two base models:

- a double classifier,
- a classifier with the class variable transformation

and four boosted models:

[2] Also known as cumulative gains curves or cumulative accuracy profiles.

- a doubled classical boosting ensemble,
- uplift AdaBoost ensemble with doubled classifiers,
- a classical boosting ensemble of classifiers with class variable transformation,
- uplift AdaBoost ensemble of classifiers with class variable transformation.

The class variable transformation is named shortly *Z model*, e.g. a decision tree with class variable transformation is named "Z decision tree". Note that "doubling" and "Z transformation" are two different ways of achieving uplift models, which than can be boosted with Uplift AdaBoost. Alternatively, we can double (classically) boosted classifiers or classically boost Z models.

In each ensemble we build $B = 101$ base models being members of the ensemble. This choice of the ensemble size is justified by the trade off: the B large enough to get a fully developed ensemble and not too big for practical applications.

Figures 1 to 6 present the uplift curves for chosen UCI datasets and the algorithms applied to J4.8 unpruned decision tree as a base model. In most cases boosting generally improves the base double model and often the proposed uplift model is superior to the ordinary double boosted model. In some cases the latter can eventually fail, which did not happen with the new algorithm (see Figure 2). Note also that the class variable transformation usually does not work properly with uplift AdaBoost.

For decision stumps the results are not so impressive. In fact, this base model sometimes works fine with classical boosting on the data with the class variable transormation, but not for the uplift AdaBoost with variable transformation (not presented on Figures).

4 Conclusions

In this paper we have developed a new boosting algorithm for the uplift modelling problem. We discuss some of its properties in relation to the classification AdaBoost algorithm and present the two other approaches to boosting in the uplift case.

Experimental evaluation showed that boosting has a potential to dramatically improve the performance of uplift models and the proposed algorithm often outperform the other two approaches. Our experiments demonstrate that ensemble methods often bring dramatic improvements in performance, turning useless single trees into highly capable ensembles. In some cases the Area Under the Uplift Curve of an ensemble was over double that of the base learner.

We conclude that further investigation of the designed algorithm is very promising and should be continued for various types of base models, as for some of them a possible improvement of model accuracy may be very remarkable.

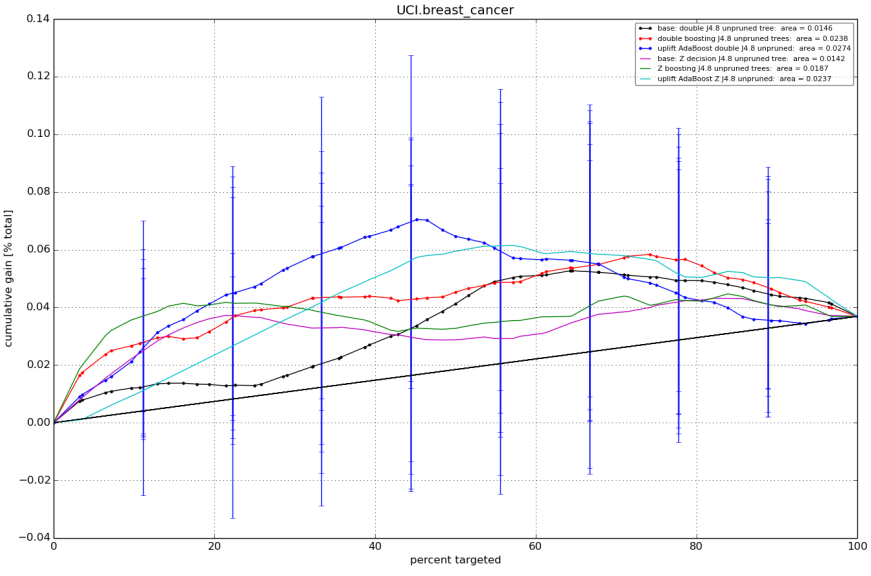


Fig. 1. Uplift curves for breast-cancer dataset.

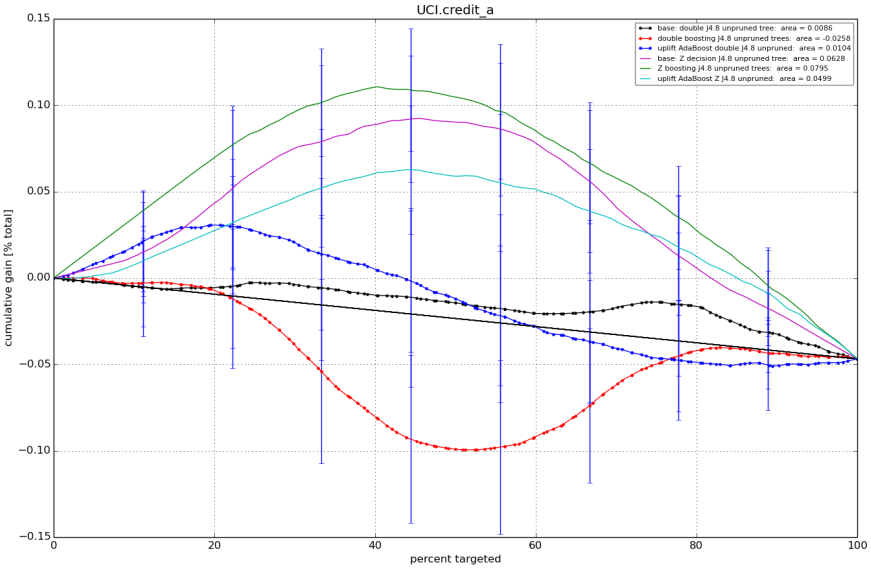


Fig. 2. Uplift curves for credit-a dataset.

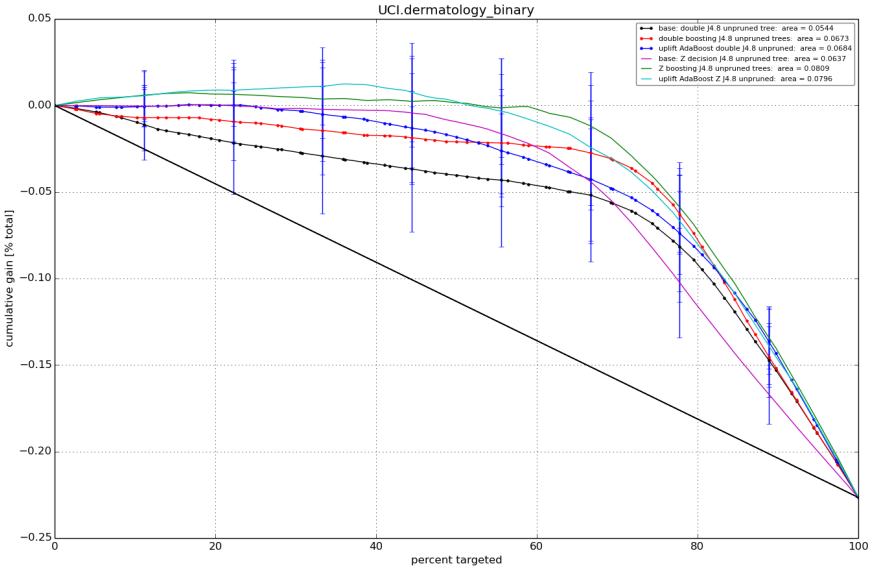


Fig. 3. Uplift curves for dermatology dataset.

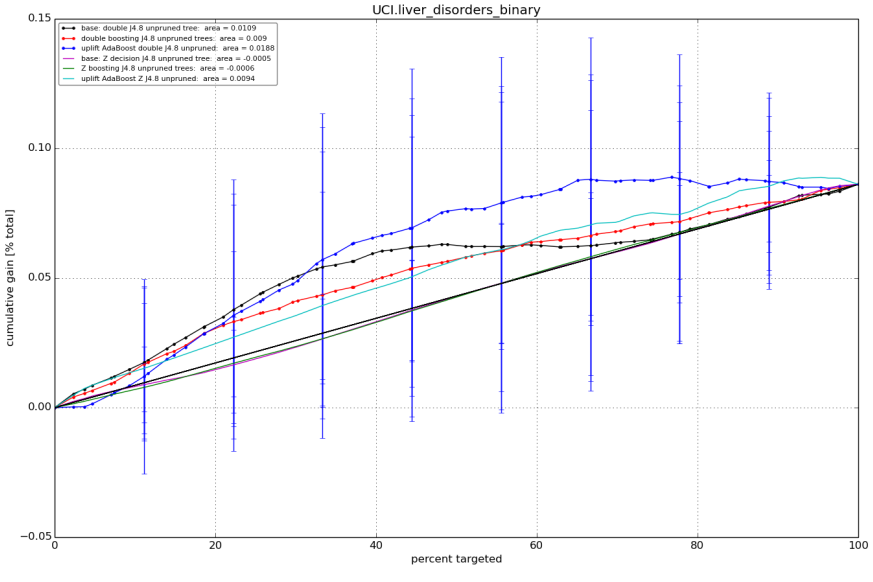


Fig. 4. Uplift curves for liver-disorders dataset.

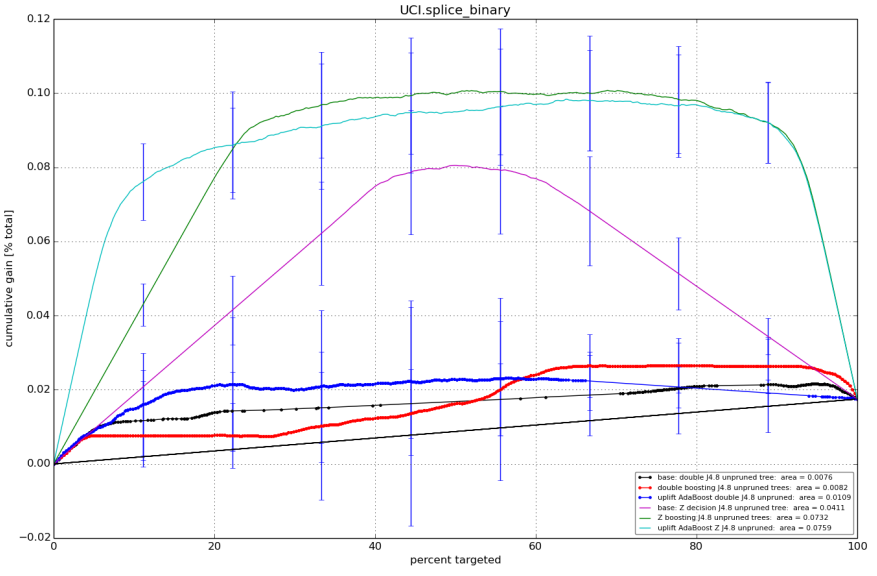


Fig. 5. Uplift curves for splice dataset.

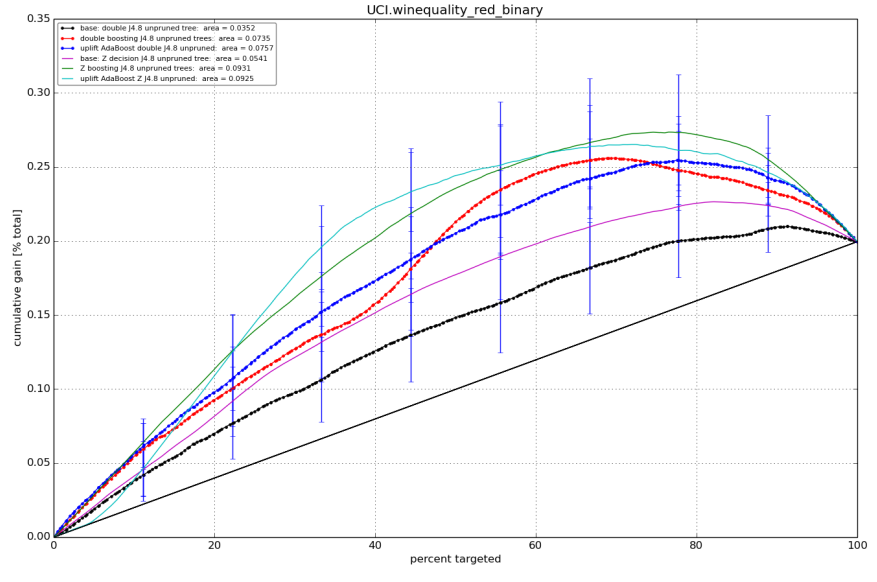


Fig. 6. Uplift curves for winequal-red dataset.

Acknowledgements

This work was supported by Research Grant no. N N516 414938 of the Polish Ministry of Science and Higher Education (Ministerstwo Nauki i Szkolnictwa Wyższego) from research funds for the period 2010–2014. M.S. was also supported by the European Union from resources of the European Social Fund: Project POKL 'Information technologies: Research and their interdisciplinary applications', Agreement UDA-POKL.04.01.01-00-051/10-00.

References

1. Hansotia, B., Rukstales, B.: Incremental value modeling. Journal of Interactive Marketing **16**(3) (2002) 35–46
2. Radcliffe, N., Surry, P.: Real-world uplift modelling with significance-based uplift trees. Portrait Technical Report TR-2011-1, Stochastic Solutions (2011)
3. Grundhoefer, M.: Raising the bar in cross-sell marketing with uplift modeling. Predictive Analytics World Conference (2009)
4. Radcliffe, N., Simpson, R.: Identifying who can be saved and who will be driven away by retention activity. Journal of Telecommunications Management **1**(2) (April 2008) 168
5. Pechyony, D., Jones, R., Li, X.: A joint optimization of incrementality and revenue to satisfy both advertiser and publisher. In: WWW 2013 Companion. (2013)
6. Holland, P.: Statistics and causal inference. Journal of the American Statistical Association **81**(396) (December 1986) 945–960
7. Jaśkowski, M., Jaroszewicz, S.: Uplift modeling for clinical trial data. In: ICML 2012 Workshop on Machine Learning for Clinical Data Analysis, Edinburgh, Scotland (June 2012)
8. Rzepakowski, P., Jaroszewicz, S.: Decision trees for uplift modeling. In: Proc. of the 10th IEEE International Conference on Data Mining (ICDM), Sydney, Australia (December 2010) 441–450
9. Rzepakowski, P., Jaroszewicz, S.: Decision trees for uplift modeling with single and multiple treatments. Knowledge and Information Systems **32** (August 2012) 303–327
10. Guelman, L., Guillén, M., Pérez-Marín, A.: Random forests for uplift modeling: An insurance customer retention case. In: Modeling and Simulation in Engineering, Economics and Management. Volume 115 of Lecture Notes in Business Information Processing (LNBIP). Springer (2012) 123–133
11. Guelman, L., Guillén, M., Pérez-Marín, A.: A survey of personalized treatment models for pricing strategies in insurance. Insurance: Mathematics and Economics (2014) to appear.
12. Sołtys, M., Jaroszewicz, S., Rzepakowski, P.: Ensemble methods for uplift modeling. Data Mining and Knowledge Discovery (2014) 1–29 online first.
13. Lo, V.: The true lift model - a novel data mining approach to response modeling in database marketing. SIGKDD Explorations **4**(2) (2002) 78–86
14. Robins, J., Rotnitzky, A.: Estimation of treatment effects in randomised trials with non-compliance and a dichotomous outcome using structural mean models. Biometrika **91**(4) (2004) 763–783
15. Vansteelandt, S., Goetghebeur, E.: Causal inference with generalized structural mean models. Journal of the Royal Statistical Society B **65**(4) (2003) 817—835

16. Jaroszewicz, S., Rzepakowski, P.: Uplift modeling with survival data. In: ACM SIGKDD Workshop on Health Informatics (HI-KDD'14), New York City, USA (August 2014)
17. Freund, Y., Schapire, R.: A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences **55**(1) (1997) 119–139
18. Schapire, R.: The strength of weak learnability. Machine Learning **5**(2) (July 1990) 197–227
19. Schapire, R., Singer, Y.: Improved boosting algorithms using confidence-rated predictions. Machine learning **37**(3) (1999) 297–336
20. Webb, G.I.: Multiboosting: A technique for combining boosting and wagging. Machine Learning **40** (2000) 159–196

Selection Consistency of GIC for Small-n-Large-p Sparse Logistic Regression Model

Hubert Szymanowski[1] and Jan Mielniczuk[1,2]

[1] Institute of Computer Science, Polish Academy of Sciences,
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
[2] Warsaw University of Technology, Faculty of Mathematics and Information Science,
ul. Koszykowa 75, 00-662 Warsaw, Poland

Abstract. We consider selection procedure for high-dimensional logistic regression problem which consists in choosing a subset of predictors which minimizes Generalized Information Criterion (GIC) over all subsets of variables of size not exceeding preset value k_n which may depend on a sample size. Nonasymptotic bound on probability of erroneous selection is proved which yields a range for GIC penalty parameter for which the procedure is consistent under mild assumptions and thus generalizes results of [1] and [2]. Various modifications of the procedure are analyzed using numerical examples.

1 Introduction

Let n be the number of observations and P_n be the number of variables which may depend on n. We consider a regression problem with matrix of experiment X of dimension $n \times (P_n + 1)$ and a binary response vector Y. Rows $x'_{i,\cdot}$ of X are thus transposed observations and its columns $x_{\cdot,j}$ contain predictors' values. The first column of X consisting of ones corresponds to the intercept. We assume that observations pertain to the standard logistic regression model with probability of success given observation $x_{i,\cdot}$ described by formula

$$\mathcal{P}(Y_i = 1|x_{i,\cdot}) = \frac{1}{1 + \exp(-\beta_0 x_{i,\cdot})}$$

where $\beta_0 = (\beta_{0,0}, \beta_{0,1}, ..., \beta_{0,P_n})$ is (P_n+1)-dimensional vector of true coefficients. We denote by s_0 the minimal true model $\{j : \beta_{0,j} \neq 0\}$. The conditions we impose later imply that s_0 is identifiable. We assume that the minimal true model always contains intercept i.e $0 \in s_0$.

We consider the problem of constructing selectors of s_0 incorporating Generalized Information Criterion (cf [3]) with objective function

$$GIC(s) = -2l_n(\hat{\beta}(s), Y|X(s)) + a_n|s|,$$

where s is a given submodel containing $|s|$ explanatory variables and an intercept, $\hat{\beta}(s)$ is a maximum likelihood estimator calculated for model s (augmented by zeros to (P_n+1)-dimensional vector if necessary), $X(s)$ is a matrix of experiment restricted to columns from s and a_n is a chosen penalty. Specific values of penalty term lead to popular selection criteria such as Bayesian Information Criterion (BIC) with $a_n = \log(n)$ or Akaike Information Criterion (AIC) with $a_n = 2$. It was established (cf [4]) that both of this criteria tend to choose too many predictors in the case when number of potential predictors is large. Therefore, in the last few years number of Information Criteria with penalty larger than $\log(n)$ have been proposed. In [5] generalization of BIC called Extended BIC (EBIC) with penalty $a_n = \log n + 2\gamma \log P_n$ for some $\gamma \geq 0$ is considered. It stems from putting a certain non-uniform prior on family of models. Note that EBIC penalty depends on the number of potential predictors and is of order $\log P_n$ when P_n is of a higher order than n.

Selection procedure based on GIC involves looking for a subset $\hat{s}_0$ which minimizes GIC objective function over predefined family of models $\mathcal{M}$. We consider $\mathcal{M} = \{s : |s| \leq k_n\}$ with threshold k_n which may depend on n. This selection method in the case of $k_n = k$ was introduced in [5] for the linear models and extended in [1] to the case of the generalized linear models (GLMs). In [2] properties of this selection method restricted to the standard logistic regression model were studied under two assumptions: Sparse Riesz Condition (SRC) for both Hessian matrix of loglikelihood function and experimental matrix and assumption of uniform continuity of the Hessian. Here we generalize and improve these results. We prove a nonasymptotic bound on the probability of erroneous selection from which selection consistency follows under certain relations between the minimal eigenvalue of a moment matrix, norms of observations and GIC penalty.

The paper is organized as follows. In Section 2 we introduce preliminaries and in Section 3 we state and prove the main results. In Section 4 numerical experiments are discussed.

2 Preliminaries

We partition all models including intercept of size not exceeding k_n into two disjoint families

$$\mathcal{A}_0 = \{s : |s| \leq k_n \wedge s \supseteq s_0\}$$

and its complement

$$\mathcal{A}_1 = \{s : |s| \leq k_n \wedge s \not\supseteq s_0 \wedge 0 \in s\}.$$

Let $p(t) = 1/(1 + \exp(-t))$ and $\sigma^2(t) = p(t)(1 - p(t))$. For the standard logistic regression with logit link function, conditional likelihood for a given model $s \in$

$\mathcal{A}_0 \cup \mathcal{A}_1$ and parameter $\beta \in \mathbb{R}^{|s|+1}$ is

$$l(\beta, Y|X(s)) = \sum_{i=1}^{n} \{y_i \log[p(x'_{i,\cdot}(s)\beta)] + (1-y_i)\log[1-p(x'_{i,\cdot}(s)\beta)]\}$$
$$= \sum_{i=1}^{n} \{y_i x'_{i,\cdot}(s)\beta - \log[1+\exp(x'_{i,\cdot}(s)\beta)]\},$$

where $X(s)$ stands for the design matrix X restricted to the columns from s and $x'_{i,\cdot}(s)$ is the i-th row of this matrix.
We denote by $\beta(s)$ $|s|$-dimensional vector augmented by zeros to higher-dimensional vector when necessary. The maximum likelihood estimator (ML) $\hat{\beta}(s)$ of parameter $\beta_0(s)$ is defined as

$$\hat{\beta}(s) = \arg \max_{\beta \in \mathbb{R}^{|s|+1}} l(\beta, Y|X(s)).$$

Define also the score function

$$S_n(\beta) = \frac{\partial l(\beta, Y|X)}{\partial \beta} = \sum_{i=1}^{n} [y_i - p(x'_{i,\cdot}\beta)]x_{i,\cdot} = X'(Y - \mathbf{p}(\beta)), \tag{1}$$

where $\mathbf{p}(\beta) = (p(x'_1\beta), \ldots, p(x'_n\beta))'$. The negative Hessian matrix will be denoted by

$$H_n(\beta) = -\frac{\partial^2 l(\beta, Y|X)}{\partial \beta \partial \beta'} = \sum_{i=1}^{n} \sigma^2(x'_{i,\cdot}\beta)x_{i,\cdot}x'_{i,\cdot} = X'\Pi(\beta)X, \tag{2}$$

where $\Pi(\beta) = \mathrm{diag}\{\sigma^2(x'_{1,\cdot}\beta), \ldots, \sigma^2(x'_{n,\cdot}\beta)\}$.
Define

$$\tilde{\lambda}_{min} = \min_{s \in \mathcal{A}_1} \lambda_{min}(X'(s \cup s_0)X(s \cup s_0)),$$
$$N = \max_{i=1,2,\ldots,n} ||x_{i,\cdot}(s_0)||,$$
$$\tilde{N} = \max_{s \in \mathcal{A}_1} \max_{i=1,2,\ldots,n} ||x_{i,\cdot}(s \cup s_0)||.$$

Results of the paper are proved under certain assumptions involving relations between the three quantities above and penalty a_n.

3 Main results

We assume throughout that $P_n > 2$ for every n.

Lemma 1 *Let $Y = (y_1, \ldots, y_n)'$ be a vector consisting of independent binary variables having not necessarily the same distribution and*

$$A(s) = X(s \cup s_0)[X'(s \cup s_0)X(s \cup s_0)]^{-1}X'(s \cup s_0)$$

for $s \in \mathcal{A}_0 \cup \mathcal{A}_1$. Then for every $\varepsilon > 0$ and $c(\varepsilon) = 0.5(4\varepsilon - \sqrt{9+8\varepsilon} + 3) > 0$ the following inequalities hold for all n

$$\mathcal{P}(\max_{s\in\mathcal{A}_1} ||A(s)(Y-EY)||^2 > (\frac{5}{4}+\varepsilon)(k_n+|s_0|)\log P_n) \leq P_n^{-c(\varepsilon)(k_n+|s_0|)}. \quad (3)$$

$$\mathcal{P}(\max_{s\in\mathcal{A}_0} ||A(s)(Y-EY)||^2 > (\frac{5}{4}+\varepsilon)(k_n)\log P_n) \leq P_n^{-c(\varepsilon)k_n}. \quad (4)$$

$$\mathcal{P}(\max_{s\in\mathcal{A}_0, s\neq s_0} [||A(s)(Y-EY)||^2 - (\frac{5}{4}+\varepsilon)(|s|-|s_0|)\log P_n] > 0) \leq \exp(P_n^{-c(\varepsilon)}) - 1. \quad (5)$$

Proof

First we prove inequality (3). Since $A(s)$ is an idempotent matrix for any s, we have $\mathrm{tr}A^2(s) = \mathrm{tr}A(s) = |s \cup s_0|$ and $\lambda_{max}(A(s)) = 1$. It follows from Theorem 2.1 in [6] that

$$P(||A(s)(Y-EY)||^2 > \frac{1}{4}(\mathrm{tr}(A(s)) + 2\sqrt{\mathrm{tr}(A^2(s))t} + 2\lambda_{max}(A(s))t)) < e^{-t}.$$

Let $t = (1+c(\varepsilon))(k_n+|s_0|)\log P_n$. Note that $\sqrt{1+c(\varepsilon)} = \sqrt{2\varepsilon+9/4} - 1/2$. We have

$$\begin{aligned}
&\mathcal{P}(\max_{s\in\mathcal{A}_1} ||A(s)(Y-EY)||^2 > (\frac{5}{4}+\varepsilon)(k_n+|s_0|)\log P_n) \\
&= \mathcal{P}(\max_{s\in\mathcal{A}_1} ||A(s)(Y-EY)||^2 > \frac{1}{4}(1+2\sqrt{1+c(\varepsilon)}+2(1+c(\varepsilon)))(k_n+|s_0|)\log P_n) \\
&\leq \sum_{j=1}^{k_n} \binom{P_n}{j} \max_{s:|s|=j} P(||A(s)(Y-EY)||^2 \\
&\quad > \frac{1}{4}(k_n+|s_0|+2(k_n+|s_0|)\sqrt{(1+c(\varepsilon))\log P_n}+2(1+c(\varepsilon))(k_n+|s_0|)\log P_n)) \\
&\leq \exp(-(1+c(\varepsilon))(k_n+|s_0|)\log P_n) \sum_{j=1}^{k_n} \binom{P_n}{j} \\
&\leq \exp(-(1+c(\varepsilon))(k_n+|s_0|)\log P_n) \sum_{j=1}^{k_n+|s_0|} \frac{P_n^j}{j!} \\
&\leq \exp(-(1+c(\varepsilon))(k_n+|s_0|)\log P_n) \frac{P_n^{k_n+|s_0|}}{(k_n+|s_0|-1)!} \leq P_n^{-c(\varepsilon)(k_n+|s_0|)}.
\end{aligned}$$

For the last two inequalities we use an $\binom{n}{k} \leq n^k/k!$ and the fact that sequence $n^k/k!$ is non decreasing for fixed n and $k = 1, 2, ..., n$.

Proof of inequality (4) is similar. In order to prove (5) change t in the reasoning above to $t = |s| \log P_n$ and note that

$$\sum_{j=|s_0|}^{k_n} \binom{P_n}{j} \big(\exp(-(1+c(\varepsilon)) \log P_n) \big)^j \leq (1 + \frac{1}{P_n^{1+c(\varepsilon)}})^{P_n} - 1 \leq \exp(P_n^{-c(\varepsilon)}) - 1.$$

□

Remark 1 *If the number of variables P_n is constant, Lemma 1 does not give a suitable bounds on considered probability. In such a case we use a slightly modified version of the Lemma. For $P_n = P$ we set $k_n = P$. Let $f = \{0, 1..., P\}$ be a full model and 2^f be a set of all possible models. In the considered setting for $n > P$ and any constant $M \geq P$ we have*

$$\mathcal{P}(\max_{s \in 2^f, 0 \in s} ||A(s)(Y - EY)||^2 > \frac{5}{4}M) = P(||A(f)(Y - EY)||^2 > \frac{5}{4}M)$$
$$\leq \mathcal{P}(||A(f)(Y - EY)||^2 > \frac{1}{4}(P + 2\sqrt{PM} + 2M)) < e^{-M}$$

and

$$\mathcal{P}(\max_{s \in \mathcal{A}_0, s \neq s_0} [||A(s)(Y - EY)||^2 - \frac{5}{4}(|s| - |s_0|)M] \geq 0)$$
$$\leq \mathcal{P}(||A(F)(Y - EY)||^2] \geq \frac{5}{4}M) \leq e^{-M}.$$

In order to ensure that the minimal true model is selected with a large probability we need to find conditions under which the behaviour of $l_n(\hat{\beta}(s)) - l_n(\hat{\beta}_0(s))$ can be controlled uniformly over $s \in \mathcal{A}_1$ and $s \in \mathcal{A}_0 \setminus \{s_0\}$. This will be done using the following notion. For a given $s \in \mathcal{A}_0 \cup \mathcal{A}_1$ define

$$B(s, r) = \{\beta : ||X(s \cup s_0)(\beta(s) - \beta_0(s))||^2 \leq r^2\}. \tag{6}$$

Lemma 2 and Theorem 2 state conditions under which $\hat{\beta}(s) \in B(s, r)$ for $s \in \mathcal{A}_0$ whereas for $s \in \mathcal{A}_1$ we have $\hat{\beta}(s) \notin B(s, r)$. Define

$$\mathcal{B} = \{\forall s \in \mathcal{A}_0 \quad \hat{\beta}(s) \in B(s, \frac{\sqrt{\tilde{\lambda}_{min}}}{\tilde{N}})\}. \tag{7}$$

Lemma 2 *For all n such that inequality*

$$\frac{\sqrt{\tilde{\lambda}_{min}}}{\tilde{N}} \exp(-N||\beta_0||) \geq e\sqrt{(80 + 64\varepsilon)k_n \log P_n} \tag{8}$$

holds, we have

$$P(\mathcal{B}) \geq 1 - \exp(-c(\varepsilon)k_n \log P_n).$$

Proof

It is easily seen that $\beta \in B(s,r)$ can be represented as $\beta_0(s)+\gamma(X'(s)X(s))^{-\frac{1}{2}}u$ where $\gamma \in [0,r]$ and u is a vector with $||u|| = 1$. For any index i, $s \in \mathcal{A}_0$ and $\beta \in B(s,r)$ we have

$$\begin{aligned}\sigma^2(x'_{i,.}\beta) &= \sigma^2(x'_{i,.}\beta_0 + \gamma x'_{i,.}(X(s)'X(s))^{-\frac{1}{2}}u)\\ &\geq \sigma^2(||x'_{i,.}(s_0)|| \cdot ||\beta_0|| + r\sqrt{x'_{i,.}(X(s)'X(s))^{-1}x_{i,.}})\\ &\geq \sigma^2(||x'_{i,.}(s_0)|| \cdot ||\beta_0|| + r||x'_{i,.}(s)||/\sqrt{\lambda_{min}(X(s)'X(s))})\\ &\geq \sigma^2(N||\beta_0|| + r\tilde{N}/\sqrt{\tilde{\lambda}_{min}}).\end{aligned}$$

Let $\beta_u = \beta_0 + r(X(s)'X(s))^{-\frac{1}{2}}u$ for some u such that $||u|| = 1$. Note that β_u is a boundary point of $B(s,r)$. Using concavity of $l_n(\cdot)$ we have

$$P(\exists s \in \mathcal{A}_0 \;\; \hat{\beta}(s) \notin B(s, \frac{\sqrt{\tilde{\lambda}_{min}}}{\tilde{N}})) \leq P(\exists u : ||u|| = 1, \max_{s\in\mathcal{A}_0} l_n(\beta_u) \geq l_n(\beta_0))$$

and the bound above is in its turn not larger than

$$\begin{aligned}&P(\exists u : ||u|| = 1, \max_{s\in\mathcal{A}_0} \big[u'(X(s)'X(s))^{-\frac{1}{2}}X(s)'(Y-EY)\\ &-\frac{1}{2}ru'(X(s)'X(s))^{-\frac{1}{2}}H(\beta^*)(X(s)'X(s))^{-\frac{1}{2}}u\big] \geq 0)\\ &\leq P(\max_{s\in\mathcal{A}_0} ||A(s)(Y-EY)|| \geq \frac{1}{2}r\sigma^2(N||\beta_0|| + r\tilde{N}/\sqrt{\tilde{\lambda}_{min}}))\\ &\leq P(\max_{s\in\mathcal{A}_0} ||A(s)(Y-EY)|| \geq \frac{1}{8}r\exp(-N||\beta_0|| - r\tilde{N}/\sqrt{\tilde{\lambda}_{min}}))\end{aligned}$$

for some β^* belonging to the line segment between β_u and β_0. We used the fact that the scalar product $u'v$ for a given vector v and $||u|| = 1$ is maximized by $u = v/||v||$. The last inequality follows from the fact that $\sigma^2(t) = e^{-|t|}/(1+e^{-|t|})^2 \geq 0.25e^{-|t|}$. For $r = \sqrt{\tilde{\lambda}_{min}}/\tilde{N}$ by Lemma 1 the right hand side is bounded from above by $\exp(-c(\varepsilon)k_n \log P_n)$ if inequality (8) is satisfied. □

Theorem 1 *For all n such that inequality (8) and*

$$a_n \geq (5+4\varepsilon)e\log P_n e^{N||\beta_0||} \tag{9}$$

hold simultaneously, we have

$$P(\min_{s\in A_0, s\neq s_0} GIC(s) \leq GIC(s_0)) \leq P_n^{-c(\varepsilon)k_n} + \exp(P_n^{-c(\varepsilon)}) - 1$$

Proof
Let $s \in \mathcal{A}_0$. For some β^* being a vector belonging to the line segment with endpoints $\hat{\beta}(s)$ and $\beta_0(s)$ we have

$$\begin{aligned} & l_n(\hat{\beta}(s)) - l_n(\hat{\beta}_0(s)) \le l_n(\hat{\beta}(s)) - l_n(\beta_0(s)) \\ &= [\hat{\beta}(s) - \beta_0(s)]' S_n(\beta_0(s)) - \frac{1}{2}[\hat{\beta}(s) - \beta_0(s)]' H_n(\beta^*)[\hat{\beta}(s) - \beta_0(s)]. \end{aligned}$$

From convexity $\beta^* \in B(s,r)$ and on event $\mathcal{B}$ defined by (7) we have in view of the proof of Lemma 2 that $\sigma^2(x'_{i,\cdot}\beta^*) \ge \sigma^2(N||\beta_0|| + 1)$ for any i. On the event $\mathcal{B}$ we also have

$$\hat{\beta}(s) - \beta_0 = \gamma_s (X'(s)X(s))^{-\frac{1}{2}} u'_s$$

for some $\gamma_s \in [0, r]$ and vector u_s with $||u_s|| = 1$. This implies that on $\mathcal{B}$

$$\begin{aligned} l_n(\hat{\beta}(s)) - l_n(\hat{\beta}_0(s)) &\le \gamma_s ||(X'(s)X(s))^{-\frac{1}{2}} X(s)'(Y - EY)|| - \frac{1}{2}\gamma_s^2 \sigma^2 (N||\beta_0|| + 1) \\ &\le \frac{||(X'(s)X(s))^{-\frac{1}{2}} X(s)'(Y - EY)||^2}{2\sigma^2(N||\beta_0|| + 1)} = \frac{||A(s)(Y - EY)||^2}{2\sigma^2(N||\beta_0|| + 1)} \end{aligned}$$

Therefore,

$$\begin{aligned} & P(\min_{s \in \mathcal{A}_0, s \ne s_0} GIC(s) \le GIC(s_0)) \\ &= \mathcal{P}(\max_{s \in \mathcal{A}_0, s \ne s_0} (l_n(\hat{\beta}(s)) - l_n(\hat{\beta}(s_0)) - \frac{(|s| - |s_0|)a_n}{2}) \ge 0) \\ &\le \mathcal{P}(\max_{s \in \mathcal{A}_0, s \ne s_0} [||A(s)(Y - EY)||^2 - (|s| - |s_0|)a_n \sigma^2 (N||\beta_0|| + 1))] \ge 0) \\ &\le \mathcal{P}(\max_{s \in \mathcal{A}_0, s \ne s_0} [||A(s)(Y - EY)||^2 - \frac{(|s| - |s_0|)a_n \exp(-N||\beta_0||)}{4e}] \ge 0). \end{aligned}$$

By Lemma 1 the last expression is bounded from above by $\exp(P_n^{-c(\varepsilon)}) - 1$ if condition (9) is satisfied. Since it follows from Lemma 2 that $P(\mathcal{B}) \le \exp(-c(\varepsilon)k_n \log P_n)$ the last step is to use inequality $P(\mathcal{C}) \le P(\mathcal{C} \cap B) + P(B')$ for $\mathcal{C} = \{\min_{s \in \mathcal{A}_0, s \ne s_0} GIC(s) \le GIC(s_0)\}$.

□

Theorem 2 *Let $\beta_{min} = \min_{i \in s_0} |\beta_{0,i}|$. Fix $\eta \in (0,1)$. For all n such the following inequalities hold*

$$\tilde{N}\beta_{min} > 1 \tag{10}$$

$$\frac{\eta}{4e} \frac{\tilde{\lambda}_{min}}{\tilde{N}^2} e^{-N||\beta_0||} \ge a_n \tag{11}$$

and

$$(1 - \eta) \frac{\sqrt{\tilde{\lambda}_{min}}}{\tilde{N}} e^{-N||\beta_0||} \ge e\sqrt{(80 + 64\varepsilon)(k_n + |s_0|) \log P_n} \tag{12}$$

we have

$$P(\min_{s\in\mathcal{A}_1} GIC(s) \leq GIC(s_0)) \leq P_n^{-c(\varepsilon)(k_n+|s_0|)}$$

Proof

Consider set $B(s,r)$ defined in (6) for $r=\sqrt{\tilde{\lambda}_{min}}/\tilde{N}$ and $s\in\mathcal{A}_1$. Note that,

$$||X(s\cup s_0)(\beta(s)-\beta_0(s))|| \geq \sqrt{\lambda_{min}(X'(s\cup s_0)X(s\cup s_0))}||\beta(s)-\beta_0(s)|| \geq \sqrt{\tilde{\lambda}_{min}}\beta_{min}.$$

Thus if $r<\beta_{min}\sqrt{\tilde{\lambda}_{min}}$ then $\hat{\beta}(s)\notin\mathcal{A}_1$ and the last inequality is satisfied in view of (10). Using concavity of $l_n(\cdot)$ again we have

$$\begin{aligned}
&P(\min_{s\in\mathcal{A}_1} GIC(s) \leq GIC(s_0))\\
&\leq P(\max_{s\in\mathcal{A}_1}\ l_n(\hat{\beta}(s)) - l_n(\beta_0) \geq -\frac{a_n}{2})\\
&\leq P(\exists u: ||u||=1 \max_{s\in\mathcal{A}_1}\ l_n(\beta_u) - l_n(\beta_0) \geq -\frac{a_n}{2})\\
&\leq P(\exists u: ||u||=1, \max_{s\in\mathcal{A}_1}(u'(X(s\cup s_0)'X(s\cup s_0))^{-\frac{1}{2}}X(s\cup s_0)'(Y-EY)\\
&\quad -\frac{1}{2}ru'(X(s\cup s_0)'X(s\cup s_0))^{-\frac{1}{2}}H(\beta^*)(X(s\cup s_0)'X(s\cup s_0))^{-\frac{1}{2}}u) \geq -\frac{a_n}{2r})\\
&\leq P(\max_{s\in\mathcal{A}_1}||A(s)(Y-EY)|| \geq \frac{1}{2}r\sigma^2(N||\beta_0|| + \frac{\tilde{N}}{\sqrt{\tilde{\lambda}_{min}}}r) - \frac{a_n}{2r})\\
&\leq P(\max_{s\in\mathcal{A}_1}||A(s)(Y-EY)|| \geq \frac{1}{8e}r\exp(-N||\beta_0||) - \frac{a_n}{2r})\\
&\leq P(\max_{s\in\mathcal{A}_1}||A(s)(Y-EY)|| \geq \frac{1-\eta}{8e}r\exp(-N||\beta_0||))
\end{aligned}$$

where (11) is used for the last inequality. By Lemma 1 the last probability is bounded from above by $\exp(-c(\varepsilon)(k_n+|s_0|\log P_n)$ if

$$\frac{1-\eta}{8e}r\exp(-N||\beta_0||) \geq \sqrt{(\frac{5}{4}+\varepsilon)(k_n+|s_0|)\log P_n}$$

which is equivalent to (12).

□

Corollary 1 *It follows from Theorems 1 and 2 that for all n such that inequalities (9) (10), (11),(12) hold for some* $\eta\in(0,1)$ *and* $\varepsilon>0$, *we have*

$$P(\min_{s\in\mathcal{A}_0\cup\mathcal{A}_1, s\neq s_0} GIC(s) \leq GIC(s_0)) \leq 2P_n^{-c(\varepsilon)(k_n+|s_0|)} + \exp(P_n^{-c(\varepsilon)}) - 1.$$

Remark 2 *Write $w_n << z_n$ if $w_n = o(z_n)$ for $n \to \infty$. It follows from Theorems 1 and 2 that if*

$$1 << \tilde{N} \tag{13}$$

$$k_n \log P_n << \frac{\tilde{\lambda}_{min}}{\tilde{N}^2} e^{-2N||\beta_0||} \tag{14}$$

and

$$e^{N||\beta_0||} \log P_n << a_n << \frac{\tilde{\lambda}_{min}}{\tilde{N}^2} e^{-N||\beta_0||} \tag{15}$$

we have

$$P(\min_{s \in \mathcal{A}_0 \cup \mathcal{A}_1, s \neq s_0} GIC(s) \leq GIC(s_0)) \to 0$$

when n tends to infinity.

Remark 3 *If number of variables P is constant we use in Lemma 2 and Theorems 1 and 2 inequalities from Remark 1. This leads to the following conditions for consistency of GIC. If condition (13) is satisfied and*

$$e^{N||\beta_0||} << a_n << \frac{\tilde{\lambda}_{min}}{\tilde{N}^2} e^{-N||\beta_0||} \tag{16}$$

we have

$$P(\min_{s \in 2^f, s \neq s_0} GIC(s) \leq GIC(s_0)) \to 0$$

where f and 2^f are defined in Remark 1. Note that in considered case we have $\tilde{\lambda}_{min} = \lambda_{min}(X'(f)X(f))$ and $\tilde{N} = \max_{i=1,\ldots,n} ||x_{i,\cdot}||$.
If condition (13) does not hold, then in the proof of Theorem 2 we take $r = A\sqrt{\tilde{\lambda}_{min}}$ with $A < \beta_{min}$. This leads to the following conditions on consistency of GIC. If

$$1 << a_n << \tilde{\lambda}_{min} \tag{17}$$

we have

$$P(\min_{s \in 2^f, 0 \in s, s \neq s_0} GIC(s) \leq GIC(s_0)) \to 0.$$

4 Discussion of the assumptions

In this section we examine behavior of $\tilde{\lambda}_{min}$, $\tilde{N}$ and N when design matrix has some specific structure. We find the following lemma useful. It is a version of Preposition 1 in [7] for unnormalized predictors.

Lemma 3 *Let $\rho_{ij} = x'_{\cdot,i} x_{\cdot,j}$. The following inequality holds*

$$\tilde{\lambda}_{min} \geq \min_j \rho_{jj} - \max_{|s|=k_n} \inf_{\alpha>1} \left[\sum_{i \in s \cup s_0} \left(\sum_{j \in s \cup s_0 \setminus \{i\}} |\rho_{ij}|^{\alpha/(\alpha-1)} \right)^{\alpha-1} \right]^{1/\alpha}. \tag{18}$$

Since all the matrices $X'(s \cup s_0)X(s \cup s_0)$ are positively defined, the lemma is nontrivial if the right hand side of (18) is positive.

Proof

Fix s such that $|s| \leq k_n$ and denote by $j_1, j_2, ..., j_{|s \cup s_0|}$ elements of $s \cup s_0$. Let $b(s \cup s_0) = b = (b_{j_1}, ..., b_{j_{|s \cup s_0|}})'$ be an eigenvector of $X'(s \cup s_0)X(s \cup s_0)$ corresponding to its minimal eigenvalue $\lambda_{min}(s \cup s_0) = \lambda_{min}$. From the definition of eigenvector for any $j \in s \cup s_0$ we have

$$\sum_{i \in s \cup s_0} \rho_{ji} b_i = \lambda_{min} b_j.$$

Therefore by Hölder's inequality

$$\min_{j=1,...,p} |\lambda_{min} - \rho_{jj}|^\alpha \sum_{j \in s \cup s_0} |b_j|^\alpha \leq \sum_{j \in s \cup s_0} |(\lambda_{min} - \rho_{jj})b_j|^\alpha$$

$$= \sum_{j \in s \cup s_0} \left| \sum_{i \in s \cup s_0 \setminus \{j\}} \rho_{ji} b_i \right|^\alpha \leq \sum_{j \in s \cup s_0} \left(\sum_{i \in s \cup s_0 \setminus \{i\}} |\rho_{ij}|^{\alpha/(\alpha-1)} \right)^{\alpha-1} \sum_{i \in s \cup s_0} |b_i|^\alpha.$$

Let $\delta = \max_{|s|=k_n} \inf_{\alpha>1} \left[\sum_{i \in s \cup s_0} \left(\sum_{j \in s \cup s_0 \setminus \{i\}} |\rho_{ij}|^{\alpha/(\alpha-1)} \right)^{\alpha-1} \right]^{1/\alpha}$. After dividing both sides by $\sum_{i \in s \cup s_0} |b_i|^\alpha$ we obtain $\min_j |\lambda_{min} - \rho_{jj}| \leq \delta$ which implies that $\lambda_{min} \geq \min_j \rho_{jj} - \delta$. Since the right hand side does not depend on choice of s the lemma is proved.

□

Let

$$\rho_n = \max_{i \neq j} |\rho_{ij}|, \quad \tau_n = \min_j ||x_{\cdot,j}||, \quad M_n = \max_{i=1,...,n; j=1,...,P_n} |x_{ij}|.$$

Then (18) and Schwarz inequality implies

$$\frac{\sqrt{\tilde{\lambda}_{min}}}{\tilde{N}} \exp(-N||\beta_0||) > \frac{\sqrt{\tau_n^2 - (k_n + |s_0|)\rho_n}}{\sqrt{k_n + |s_0|}M_n} \exp(-\sqrt{s_0}M_n||\beta_0||).$$

The lower bound is positive if $\rho_n < \tau_n^2/k_n$ and the inequality (8) holds if

$$\tau_n^2 - (k_n + |s_0|)\rho_n > e^2(80 + 64\varepsilon)(k_n + |s_0|)^2 \log P_n M_n^2 \exp(2\sqrt{s_0}M_n||\beta_0||).$$

The assumption frequently used in the literature is Sparse Riesz Condition (SRC) see e.g. [7]. We say that the design matrix X satisfies left-sided SRC with rank k_n and a spectrum bound $0 < C_1 < +\infty$ if

$$\forall s : |s| \leq k_n \quad \forall v \in \mathbb{R}^{|s|} \quad C_1 \leq \frac{||X(s)v||^2}{n||v||^2}$$

which is equivalent to

$$\min_{s:|s| \leq k_n} \lambda_{min}(X'(s)X(s)) \geq C_1 n$$

Corollary 2 *Assume that matrix X satisfies left-sided SRC with rank $k_n + |s_0|$ and constant C_1. For all n such that*

$$e^2(80+64\varepsilon)M_n^2(k_n+|s_0|)^2\log P_n\exp(2\sqrt{s_0}M_n||\beta_0||) < C_1(1-\eta)^2 n \tag{19}$$

$$1 < C_1 k_n \beta_{min}^2 \tag{20}$$

$$(5+4\varepsilon)\exp(M_n\sqrt{s_0}||\beta_0||)\log P_n < a_n < \frac{\eta C_1 n}{4e(k_n+|s_0|)M_n^2}\exp(-M_n\sqrt{s_0}||\beta_0||) \tag{21}$$

hold for some $\eta \in (0,1)$, we have

$$P(\min_{s\in\mathcal{A}_0\cup\mathcal{A}_1, s\neq s_0} GIC(s) \le GIC(s_0)) \le 2P_n^{-c(\varepsilon)(k_n+|s_0|)} + P_n^{-c(\varepsilon)}.$$

Note that if $M_n \le M$, inequality (19) reduces to

$$n > A(k_n+|s_0|)^2\log P_n \quad \text{with} \quad A = \frac{e^2(80+64\varepsilon)M^2\exp(\sqrt{s_0}M||\beta_0||)}{C_1(1-\eta)^2}$$

and inequality (21) reduces to

$$B_1\log P_n < a_n < B_2\frac{n}{k_n+|s_0|}$$

$$\text{with} \quad B_1 = (5+4\varepsilon)\exp(M\sqrt{s_0}||\beta_0||) \quad \text{and} \quad B_2 = \frac{\eta C_1}{4eM^2}\exp(-M\sqrt{s_0}||\beta_0||).$$

Proof

We show that conditions (19)-(21) imply assumptions of Corollary 1. Left-sided SRC with rank $k_n + |s_0|$ implies that for fixed s with $|s| \le k_n$, we have

$$C_1 n \le \lambda_{min}(X'(s\cup s_0)X(s\cup s_0)) \le \frac{1}{|s\cup s_0|}\mathrm{tr}(X'(s\cup s_0)X(s\cup s_0))$$
$$= \frac{1}{|s\cup s_0|}\sum_{i=1}^{n}||x_{i,\cdot}(s\cup s_0)||^2 \le \frac{n}{|s\cup s_0|}\max_{i=1,\ldots,n}||x_{1,\cdot}(s\cup s_0)||^2.$$

Thus, when left-sided SRC is satisfied and all absolute values of design entries x_{ij} are bounded from above by M_n the inequality (12) holds for some $\eta \in (0,1)$ if

$$n > AM_n^2(k_n+|s_0|)^2\log P_n\exp(2\sqrt{s_0}M_n||\beta_0||) \quad \text{with} \quad A = \frac{e^2(80+64\varepsilon)}{C_1(1-\eta)^2}.$$

Analogously, inequality (19) implies (12). Moreover, the string of the inequalities above implies that

$$\sqrt{k_n+|s_0|}M_n \ge \tilde{N} \ge C_1\sqrt{k_n+|s_0|} \quad \text{and} \quad \sqrt{|s_0|}M_n \ge N \ge C_1\sqrt{|s_0|}$$

which shows that (9) and (12) follow from (21) and (20) implies (10). □

5 Simulation study

In this section we compare different methods of variable selection for high-dimensional logistic regression. We give detailed description of the experiment, its results and conclusions.

We perform simulations for 4 artificially generated data sets described in Table 1. Number of observations is equal to 100 for the first data set and it is increased by 80 for every subsequent data set. Number of variables is a function of n given by $\lfloor \exp((n-20)^{0.37}) \rfloor$ and number of relevant variables vary from 3 for the first data set to 6 for the last one. Vector of true coefficients β_0 is equal to $(-3.5, 1.5, -2)$ for the first data set and is augmented alternately by -2 or 2 for each new relevant variable. The same setting is considered in [8].

For each data set we consider three different dependence structures between variables, namely observations are generated independently from multivariate normal distribution with zero mean and covariance matrix Σ with $\Sigma(i,j) = \rho^{|i-j|}$ for $\rho = -0.5, 0, 0.5$.

Due to computational burden we cannot directly optimize GIC for all subsets of variables containing no more than k_n variables even if k_n is relatively small. Hence, we perform two-stage procedure to find minimal true model. In the first stage we screen moderate number of valuable variables and in the second one we optimize GIC on some subfamily of models consisting of this chosen variables only. There are many statistical procedures such as LASSO, SCAD, Dantzig Selector or Random Forests, which results in ordering variables according to some measure of importance and so can be used as screening methods. In the experiment we order variables according to LASSO for GLM. The most important variable is the one for which corresponding coefficient became nonzero for the largest value of penalty parameter in the LASSO objective function.

We compare three searching procedures: hierarchical (denoted by *hier*), exhaustive (*exh*) and semi-exhaustive (*semexh*). Hierarchical procedure involves minimization of GIC objective function on the nested family of 40 variables chosen in the first step. Since we take into account an empty model- intercept only model- the number of fitted models is 41. In exhaustive procedure we minimize GIC objective function on the family of all submodels of 10 variables chosen by LASSO. The number of fitted models is 1024. Semi-exhaustive method is a version of step forward algorithm with different stop condition. First we fit 40 models, one for each variable chosen by LASSO. Then we choose the best one, so the one witch minimize GIC. Next step is to fit 39 models with two variables- the one chosen previously and each remaining one. We chose the best pair of variables and proceed. The last fitted model is a full model. Including an empty model, we fit $\binom{41}{2} + 1 = 821$ models.

In the first part of the experiment we examine quality of LASSO for GLM as a screening method in considered scenarios. Figure 1 shows estimated probability that after initial screening relevant variables are separated from spurious ones for given values of ρ. This is equivalent to saying that minimal true model s_0 belongs to the nested family of 40 most important variables. We see that values differ significantly. In the easiest case, for $\rho = -0.5$ estimated probability

is nearly equal to 1 whereas in the most difficult case, for $\rho = 0.5$ it occurs to be nearly 0.

In the second part we compare searching procedures by taking into account probability of selecting s_0 and selection error. We use EBIC penalty with $\gamma = 1$ which was chosen as the best value in preliminary simulations. The estimated probability of selecting s_0 is shown in Figure 2. In the easiest case for $\rho = -0.5$ hierarchical method works significantly better than remaining two. However, for independent predictors when ordering after first step is of lower quality, exhaustive and semi exhaustive methods are superior to hierarchical one. The tendency is even stronger for positively correlated variables. When LASSO fails in ordering variables semi exhaustive method appears to be the best. In this case probability of selecting s_0 by hierarchical method is close to 0.

We measure error of each searching procedure by mean sum of false positives (FP) and false negatives (FN). Let s_j a set of features chosen in the j-th run. The measure is given by

$$FP + FN = \frac{\sum_{j=1}^{N} |(s_j \cup s_0) \setminus (s_j \cap s_0)|}{N}.$$

Figure (3) shows the result. Conclusions are in line with those from Figure (2). The case of independent variables is the only one when with growth of n error systematically decreases. For negative ρ we see again dominance of hierarchical method with error varying from 0.6 to 0.8. For positive ρ all methods work worse, with error close to the number of relevant variables. Although in this case the best method is semi exhaustive one.

Model	$\lvert s_0\rvert$	n	$p =$ $\lfloor \exp((n-20)^{0.37}) \rfloor$	$\beta_0(s_0)$
1	3	100	158	(-3,1.5,-2)
2	4	180	692	(-3,1.5,-2,2)
3	5	260	1993	(-3,1.5,-2,2,-2)
4	6	340	4680	(-3,1.5,-2,2,-2,2)

Table 1: Model specifications.

Acknowledgements

The study is cofounded by the European Union from resources of the European Social Fund. Project PO KL "Information technologies: Research and their interdisciplinary applications", Agreement UDA-POKL.04.01.01-00-051/10-00.

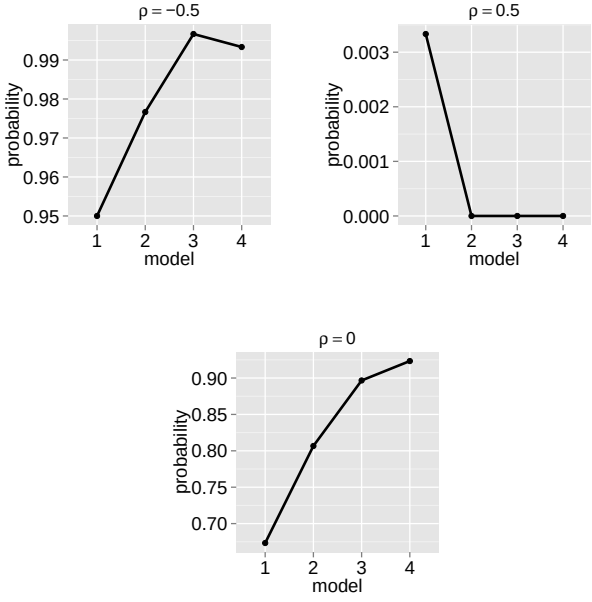


Fig. 1: Estimated probability that all relevant variables proceed the spurious ones after screening for $\rho = -0.5, 0, 0.5$.

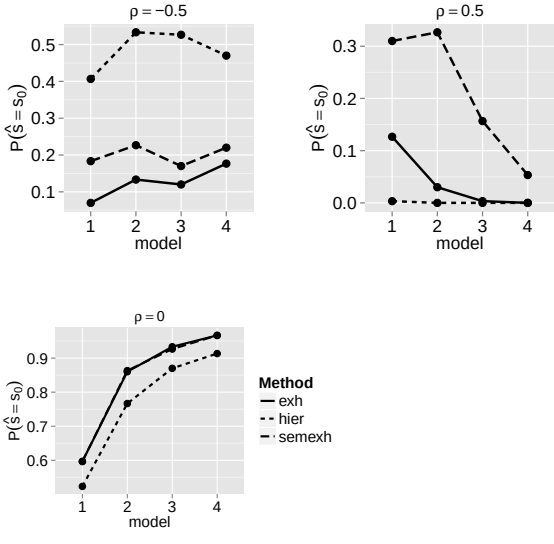


Fig. 2: Estimated probability of selecting true model.

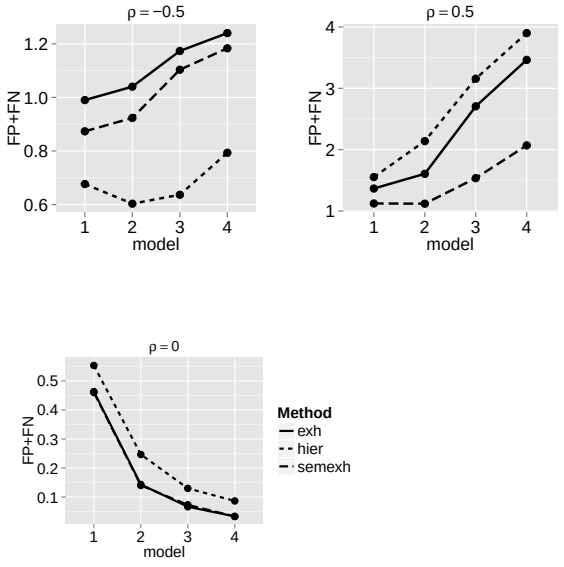


Fig. 3: Mean sum of false positives and false negatives.

References

1. Chen, J., Chen, Z.: Extended BIC for small-n-large-p sparse GLM. Statistica Sinica **22** (2012) 555–574
2. Mielniczuk, J., Szymanowski, H.: Selection consistency of generalized information criterion for sparse logistic model. In: Stochastic Models, Statistics and Their Applications. Volume 122 of Springer Proceedings in Mathematics & Statistics. (2015) 111–119
3. Sin, C., White, H.: Information criteria for selecting possibly misspecified parametric models. Journal of Econometrics **71** (1996) 207–225
4. Broman, K., Speed, T.: A model selection approach for the identification of quantitative trait loci in experimental crosses. Journal of the American Statistical Association **64** (2002) 641–656
5. Chen, J., Chen, Z.: Extended Bayesian Information Criteria for model selection with large model spaces. Biometrika **95** (2008) 759–771
6. Hsu, D., Kakade, S.M., T., Z.: A tail inequality for quadratic forms of subgaussian random vectors. Electronic Communications in Probability **17**(52) (2012) 1–6
7. Zhang, C.H., Huang, J.: The sparsity and bias of the lasso selection in high-dimensional linear regression. Annals of Statistics **36**(4) (2008) 1567–1594
8. Fan, Y., Tang, C.Y.: Tuning parameter selection in high dimensional penalized likelihood. Journal of the Royal Statistical Society: Series B **75** (2013) 531–552

Distributional Proteomics: Modelling Amino Acid Relationships by Measuring Their Patterns of Statistical Occurrence Across Proteins

Marcin Tatjewski[1,2] and Dariusz Plewczyński[2]

[1] Institute of Computer Science, Polish Academy of Sciences,
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
[2] Centre of New Technologies, University of Warsaw

Abstract.
Since 1990s the development of linguistic methods based on the *distributional hypothesis* has lead to significant achievements in extraction of word semantics from large corpora of texts in natural language. To date, there was no attempt made to apply algorithms of distributional semantics on biological data, despite many other successful method transfers between linguistic engineering and bioinformatics. Therefore, we constructed a distributional word-context matrix based on protein sequence data, making an analogy between words in natural language and single amino acids in proteins as most basic carriers of information. Our computational approach is inspired by the linguistic method of Correlated Occurrence Analogue to Lexical Semantics. In order to achieve our goal we also build a balanced set of protein sequences, as analogy to balanced text corpora in linguistics. The matrices which we obtained achieve correlations of up to 0.76 with amino acid substitution matrices. Substitution matrices are a widely used model of amino acid relationships, built using multiple sequence alignments and evolutionary data and useful in proteomics for sequence alignments. Our result suggests the potential to extract information about amino acids by purely statistical analysis of protein data. However, contrary to results in linguistic engineering, we obtain slightly higher correlation scores for matrices modelling simple tendency to co-occur than for matrices which model the more complex relationship of amino acids based on context.

1 Introduction

1.1 Linguistic distributional semantics

Linguistic distributional semantics is a part of a broader domain referred to as vector space models of semantics. This wide area aims at inferring word meaning from the statistical patterns of word usage in language. The foundations of the

field are build on theories like the *bag of words hypothesis*, the *latent relation hypothesis* or the *distributional hypothesis* [1]. The last one of them lies within the interest of our work. Distributional hypothesis states that the meaning of a word can be discovered by observation of the contexts in which the word occurs. Possibility of such inference is especially useful for detecting semantic similarity of words. It was already in 1950s that the *Distributional Hypothesis* was proposed [2]. However, it had to wait until the advent of computational methods in linguistic engineering before it could be applied on a larger scale.

The first proposed algorithm which used this hypothesis is Hyperspace Analogue to Language (HAL) [3]. It served its authors to construct some of the first word *semantic spaces*, also called *matrices of semantic relatedness* [4]. HAL method also established the four main steps which till now are the main ingredients of distributional semantics procedures. These steps are:

1. Gathering and preparing the text corpus which will serve as the experimental base. Linguists pay special attention to this stage and try to build *balanced* corpora. A balanced corpus includes texts from diverse sources: spoken language, books, newspapers, letters etc. An example of a balanced corpus is the balanced version of the National Corpus of Polish [5].
2. Processing the text corpus with a sliding window in order to obtain a co-occurrence count matrix. Sliding windows can vary in size, can be ramped or flat.
3. Post-processing of the obtained co-occurrence matrix, which at least should involve normalization, yet it may contain more advanced transformations or dimensionality reduction.
4. Establishing a similarity measure between the word-vectors described in the finally obtained matrix.

Soon after HAL appeared the better-known Latent Semantic Analysis (LSA) [6], yet it was a move from word-context model to word-document model. On the other hand, a continuation and extension of the HAL's word-context approach was proposed in the Correlated Occurrence Analogue to Lexical Semantics (COALS) model [7]. Concepts from the COALS algorithm were the most inspirational for our work. Supplement A presents an extract of our previous work: an example of word similarity results obtained with use of COALS method in a study of word synonymy for Polish language.

Nowadays, vector space models of semantics are a well developed domain with many successful applications. They are also easy to use, as there are many available text corpora and dedicated software packages [8].

Our experience of work with linguistic distributional semantics lead us in our daily bioinformatics research to an attempt to apply similar techniques in order to model relationships (*semantics*) between amino acids.

1.2 Amino acid relationship modelling

Modelling differences between amino acids is an important task for proteomics. Whether we want to extract features for machine learning from amino acid

sequences or whether we attempt to align several proteins in order to reason about their common traits, we need to be able to quantitatively compare amino acids to each other. Several resources are available for such tasks. Many of them are gathered in the AAIndex Database [9]. Two types of such resources which we would like to cover are:

Amino acid indices - These are mostly physicochemical properties with specific values for each amino acid. They can relate to hydrophobicity, compositional properties, structural propensity, electric properties and others. Indices are useful, for example, in the task of feature extraction from protein sequences [10]. Their abundance might pose a challenge, yet this is often addressed with clustering or feature selection methods [11].

$$amino\ acid\ index : AA \rightarrow R$$

Equation 1: Amino acid indices - functions returning values of physicochemical attributes for each amino acid. Below AA is the set of amino acids, and R is the set of real values.

Substitution (or mutation) matrices - The idea behind them comes from the need to align protein sequences with each other. Before their appearance, alignment scoring algorithms counted sequence matches or mismatches equally - not taking into account which particular amino acids are compared. Later on, based on the observation that some amino acids are more likely to mutate than others (and also mutate to specific targets), biologists started to differently rate the proximity of protein sequences. To establish the substitution scores, scientist relied on analysis of multiple alignments. In PAM aligned were evolutionary similar proteins [12], while in BLOSUM alignment focused on very conserved regions in distant proteins [13]. Other methods were also designed, yet these two substitution matrices are one of the most popular and are commonly used in the popular BLAST program [14].

$$substitution\ matrix : AA \times AA \rightarrow R$$

Equation 2: Substitution matrices - functions returning evolutionary/chemical similarity/dissimilaity scores for pairs of amino acids. Below AA is the set of amino acids, and R is the set of real values.

Substitution matrices in proteomics serve similar purpose as semantic spaces in linguistic. They rate proximity of, respectively, amino acids and words. However, both methods are based on a very different approach. In our work we decided to apply the methods of distributional semantics to proteins, thus building

an *amino acid semantic space*. We base this approach on the analogy between words and amino acids - both are the basic units carrying information in their domains and both occur in large sequences that can be studied statistically.

2 Method and materials

2.1 Protein corpus preparation

As we mentioned in section 1.1, an important step of every linguistic engineering experiment is careful preparation of a text corpus. Therefore, we decided to pay special attention to this step in our biological experiment. Taking just a full set of proteins from a database like UniProt might have biased the results, as proteins in different domains are not equally well researched. For some types of organisms or some functional types of proteins we have much more objects sequenced than in other areas.

In order to obtain a balanced protein set we utilized UniProt20 database which was developed in the HHblits package [15]. UniProt20 is a clustered set of proteins from UniProt. It was constructed using similarity threshold of 20%. To build our protein corpus we took at maximum one sequence from each cluster of UniProt20. However, we only accepted sequences that are marked in the original UniProt as having experimental evidence at protein or transcript level [16]. Therefore, some UniProt20 clusters are not at all represented in our dataset. Statistics of the protein corpus which we obtained are displayed in Table 2.1.

Number of sequences	347 409
Number of amino acids	101 966 845
Mean sequence length	293.5
Median sequence length	193.0

Table 1. Statistics of the obtained protein corpus.

In order to make sure that short, medium and long sequences are relatively equally represented in our dataset we looked in detail into its composition from the perspective of elements' length, what is displayed in Figures 1 and 2. Moreover, Figure 3 presents the amino acid composition of our dataset.

2.2 Amino acid distributional matrix construction

Procedure which we used to construct our amino acid distributional matrix is highly inspired by the COALS algorithm [7]. However, many of steps in COALS are appropriate only for linguistic domain, thus we eliminated:

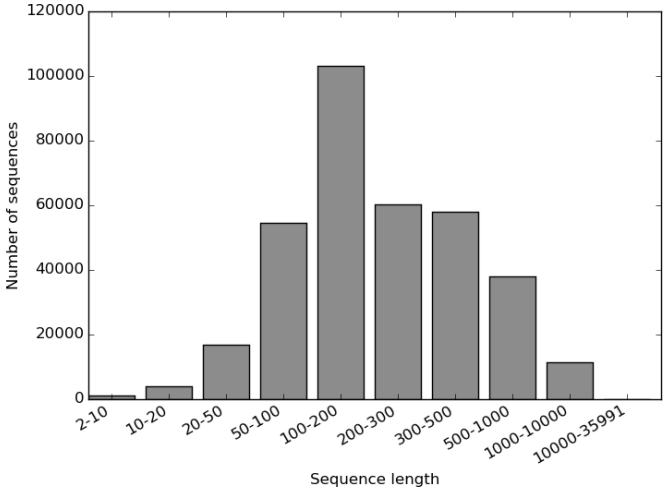


Fig. 1. Number of sequences in the protein corpus per sequence length group

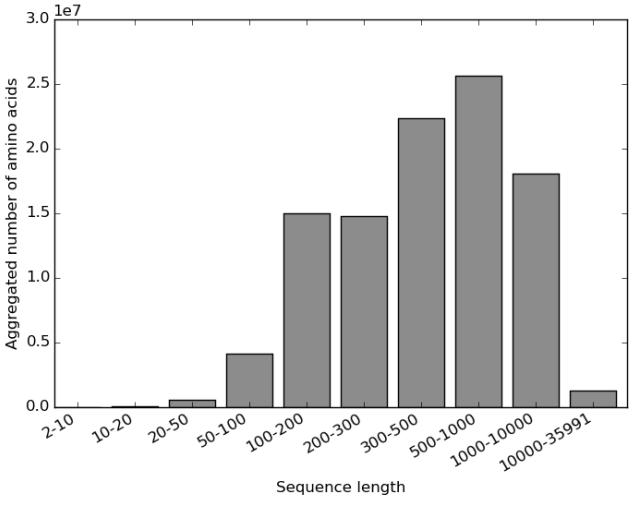


Fig. 2. Aggregated number of amino acids in the protein corpus per sequence length group

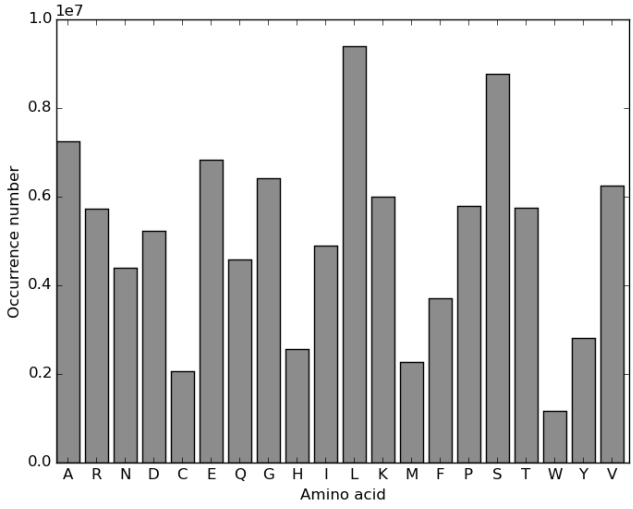


Fig. 3. Distribution of different amino acids in the protein corpus

- Dimensionality reduction, which is required in linguistics because of the huge size of the matrix based on words and is also credited with performance increase. Reduction of our 20×20 amino acid matrix does not seem to be necessary, yet it might be worth checking in the future whether it would not increase the final performance.
- Replacing negative correlation values with zeros. In linguistics this transformation step is explained by the fact that knowing a large set of words that have negative correlation with a target word is not very helpful for inferring the meaning of the target word. On the other hand, knowledge of just a handful of words that correlate positively with the target word provides a lot of insight into the target word's semantics. For example, information that an unknown word W has negative correlation with words: mountain, swimming, colorful, multiple and clumsy is not very useful when we want to infer the semantics of W. However, if we know that W correlates positively with words: dog, lion and pet, than we know much more about its meaning. Nevertheless, the world of amino acids is different. We cannot claim *a priori* that negative correlation between amino acids is meaningless. Intuitively it's seems to be quite the opposite. Therefore, we keep negative correlation values as equally valuable as positive correlations.

Therefore, our amino acid procedure consisted of the following steps:

1. Gathering co-occurrence counts in matrix of size 20×20 using a sliding window. We used flat and ramped windows with radius 4,10 and 16, thus obtaining 6 different matrices. Figure 2.2 shows how a ramped window of

radius 4 counts co-occurrence scores. The size of our final matrix is 20×20 as we decided to ignore all the non-standard amino acids as their number was not big enough.
2. Co-occurrence matrix normalization with use of formula from Figure 3. The formula should not be confused with correlation of two co-occurrence rows, as it is instead a correlation between the occurrences of two amino acids [7].
3. Obtaining similarity score for two amino acids by calculating correlation between their row-vectors. This step shifts the final results from looking at pure co-occurrence likelihood of amino acids towards the representation of their context similarity.

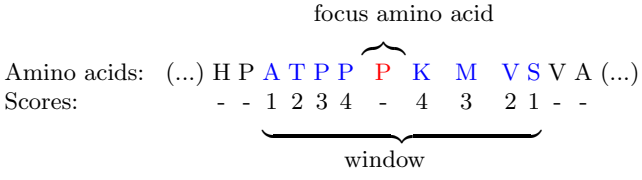


Fig. 4. Co-occurrence scoring for a ramped window with radius 4 on an example protein subsequence.

$$D_{i,j} = \frac{S * C_{a,b} - \sum_{j}^{20} C_{a,j} * \sum_{i}^{20} C_{i,b}}{(\sum_{j}^{20} C_{a,j} * (S - \sum_{j}^{20} C_{a,j}) * \sum_{i}^{20} C_{i,b} * (S - \sum_{i}^{20} C_{i,b}))^{1/2}}$$

where

$$S = \sum_{i}^{20} \sum_{j}^{20} C_{i,j}$$

Equation 3: Transforming raw co-occurrence counts (C matrix) into distribution-based tendency to co-occur (D matrix) through normalization based on Pearson's correlation coefficient. 20 is the number of basic amino acids, thus it determines the dimensionality of matrices C and D.

2.3 Comparison with substitution matrices

The most intuitive idea for checking whether the information gathered in our matrices does make biological sense is to compare them to substitution matrices.

Following works on comparing biological matrix resources with each other, we decided to use for this task a simple correlation of matrices flattened to rows, as presented in Figure 4 [17]. For comparison we took all the 93 matrices available in AAIndex2 Database [9]. It's important to note, that substitution matrices gathered in this resource are not all very different from each other, as this set contains many variations of matrices built by the same algorithms. An example of a substitution matrix is presented in Table 2.3.

For comparison with substitution matrices we did not only take our final row similarity matrices, but we also performed calculations for matrices which we obtain before performing step number 3 from procedure presented in section 2.2.

A	5																		
R	-2	7																	
N	-1	0	6																
D	-2	-1	2	7															
C	-1	-3	-2	-3	12														
E	-1	0	0	2	-3	6													
Q	-1	1	0	0	-3	2	6												
G	0	-2	0	-1	-3	-2	-2	7											
H	-2	0	1	0	-3	0	1	-2	10										
I	-1	-3	-2	-4	-3	-3	-2	-4	-3	5									
L	-1	-2	-3	-3	-2	-2	-2	-3	-2	2	5								
K	-1	3	0	0	-3	1	1	-2	-1	-3	-3	5							
M	-1	-1	-2	-3	-2	-2	0	-2	0	2	2	-1	6						
F	-2	-2	-2	-4	-2	-3	-4	-3	-2	0	1	-3	0	8					
P	-1	-2	-2	-1	-4	0	-1	-2	-2	-2	-3	-1	-2	-3	9				
S	1	-1	1	0	-1	0	0	0	-1	-2	-3	-1	-2	-2	-1	4			
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-1	-1	2	5		
W	-2	-2	-4	-4	-5	-3	-2	-2	-3	-2	-2	-2	-2	1	-3	-4	-3	15	
Y	-2	-1	-2	-2	-3	-2	-1	-3	2	0	0	-1	0	3	-3	-2	-1	3	8
V	0	-2	-3	-3	-1	-3	-3	-3	-3	3	1	-2	1	0	-3	-1	0	-3	-1	5
	A	R	N	D	C	E	Q	G	H	I	L	K	M	F	P	S	T	W	Y	V

Table 2. Example of a substitution matrix: BLOSUM45 substitution matrix (Henikoff-Henikoff, 1992). Missing values above the diagonal indicate that the matrix is symmetric. Please note that not all substitution matrices are symmetric.

3 Results and discussion

Our amino acid distributional matrices obtain surprisingly high correlations with the substitution matrices, e.g. 0.76 with matrix built by *Koshi et al*, 0.64 with BLOSUM45 or 0.51 with PAM120. It's especially interesting as our matrices are built using a very different paradigm. Distributional amino acid matrices

$$c = \frac{\sum_{i}^{N}\sum_{j}^{N}(D_{i,j} - \bar{D})(S_{i,j} - \bar{S})}{(\sum_{i}^{N}\sum_{j}^{N}(D_{i,j} - \bar{D})^2 \sum_{i}^{N}\sum_{j}^{N}(S_{i,j} - \bar{S})^2)^{1/2}}$$

Equation 4: Matrix correlation by flattening matrices to vectors. D - distributional amino acid matrix. S - substitution matrix.

are based on vertical analysis of protein sequences and they do not use any external knowledge about evolutionary relationships between proteins. On the other hand, most of the substitution matrices rely on horizontal analysis of protein multiple alignments and incorporate evolutionary information into their methodology. Also notable is the result of 0.73 correlation with a substitution matrix based on amino acid chemical properties [18]. These results show that it is possible to extract meaningful knowledge about amino acids from pure statistical analysis of protein sequences.

However, it's important to note that, contrary to the results in linguistic applications, better "performance" is achieved by distributional matrix built without the step 3 presented in method from section 2.2. This means that more related to substitution matrices is the pure likelihood of amino acid co-occurrence rather than semantic similarity driven by the context relationship.

Acknowledgements

The study is cofounded by the European Union from resources of the European Social Fund. Project PO KL "Information technologies: Research and their interdisciplinary applications", Agreement UDA-POKL.04.01.01-00-051/10-00; Polish National Science Centre (grant numbers: 2015/16/T/ST6/00493, 2014/15/B/ST6/05082 and 2013/09/B/NZ2/00121); EU COST BM1405 and BM1408 actions.

A Supplement: Example distributional semantics results for Polish language

Tables in Figure A present an example of distributional semantics results for Polish language. Usually, most valued and useful outcomes of these methods are lists of words' nearest neighbors, i.e. words having most similar vectors in the semantic space to a given word. In the case of Figure A we see nearest neighbors lists from space produced with COALS algorithm [7] run on the National Corpus of Polish [5]. Results were produced for the project APPROVAL[1], which was aimed at analysis of synonym pairs. This is why we present neighbors lists for two synonymous words [28].

[1] http://www.approval.uw.edu.pl/start

Substitution matrix AAIndex code	Correlation with matrix built without step 3. from section 2.2	Correlation with matrix built including step 3. from section 2.2	Substitution matrix description
KOSJ950102	**0.76**	0.58	Context-dependent optimal substitution matrices for exposed beta [19]
OVEJ920105	0.75	0.58	Environment-specific amino acid substitution matrix for inaccessible residues [20]
LINK010101	0.75	0.58	Substitution matrices from a neural network model [21]
MCLA720101	0.73	**0.68**	Chemical similarity scores [18]
CSEM940101	0.67	0.65	Residue replace ability matrix [22]
HENS920101	0.64	0.58	BLOSUM45 substitution matrix [13]
ALTS910101	0.51	0.47	The PAM-120 matrix [23]
AZAE970102	0.45	0.4	The substitution matrix derived from spatially conserved motifs [24]
GEOD900101	0.33	0.33	Hydrophobicity scoring matrix [25]
RUSR970101	-0.01	0.04	Substitution matrix based on structural alignments of analogous proteins [26]
GRAR740104	-0.43	- 0.41	Chemical distance [27]

Table 3. Correlation results (calculated according to Figure 4) of chosen substitution matrices with amino acid distributional matrices based on co-occurrences calculated by flat sliding window of radius 16.

Kolarz	
Similarity	**Neighbor**
0.859	biegacz
0.783	kolarski
0.775	maratończyk
0.766	kajakarz
0.762	peleton
0.761	lekkoatleta
0.758	pływak
0.755	wyścig
0.744	zawodnik
0.736	rajdowiec
0.727	szosowy
0.726	szachista
0.719	zapaśnik
0.717	kolarstwo
0.716	jaskuła
0.707	olimpijczyk

Cyklista	
Similarity	**Neighbor**
0.767	motocyklista
0.761	rowerzysta
0.715	pieszy
0.705	rajdowiec
0.704	rowerowy
0.688	kolarz
0.683	jednoślad
0.668	rower
0.667	zmotoryzowany
0.661	rajd
0.647	motocyklowy
0.641	biegacz
0.638	quad
0.624	kolarski
0.624	motocykl
0.621	spacerowicz

Fig. 5. Example nearest neighbors lists from a COALS [7] semantic space constructed over National Corpus of Polish [5] in the course of project APPROVAL (http://www.approval.uw.edu.pl/start).

References

1. Turney, P.D., Pantel, P.: From frequency to meaning: vector space models of semantics. Journal of Artificial Intelligence Research **37**(1) (January 2010) 141–188
2. Harris, Z.: Distributional structure. Word **10**(23) (1954) 146–162
3. Lund, K., Burgess, C.: Producing high-dimensional semantic spaces from lexical co-occurrence. Behavior Research Methods, Instruments, & Computers **28**(2) (June 1996) 203–208
4. Piasecki, M., Szpakowicz, S., Broda, B.: A Wordnet from the Ground Up. (2009)
5. Przepiórkowski, A., Bańko, M., Górski, R.L., Lewandowska-Tomaszczyk, B.: Narodowy Korpus Języka Polskiego. (2012)
6. Landauer, T.K., Foltz, P.W., Laham, D.: An introduction to latent semantic analysis. Discourse Processes **25**(2-3) (January 1998) 259–284
7. Rohde, D., Gonnerman, L., Plaut, D.: An improved model of semantic similarity based on lexical co-occurrence. Communications of the ACM **8** (2006) 627–633
8. Jurgens, D., Stevens, K.: The S-Space Package: An Open Source Package for Word Space Models. Proceedings of the ACL 2010 System Demonstrations (July) (2010) 30–35
9. Kawashima, S., Pokarowski, P., Pokarowska, M., Kolinski, A., Katayama, T., Kanehisa, M.: AAindex: Amino acid index database, progress report 2008. Nucleic Acids Research **36**(SUPPL. 1) (2008) 202–205
10. Plewczynski, D., Basu, S., Saha, I.: AMS 4.0: consensus prediction of post-translational modifications in protein sequences. Amino acids **43**(2) (August 2012) 573–82
11. Saha, I., Maulik, U., Bandyopadhyay, S., Plewczynski, D.: Fuzzy clustering of physicochemical and biochemical properties of amino acids. Amino acids **43**(2) (August 2012) 583–94
12. Dayhoff, M., Schwartz, R.: A Model of Evolutionary Change in Proteins. In Atlas of protein sequence and structure (1978) 345–352
13. Henikoff, S., Henikoff, J.G.: Amino acid substitution matrices from protein blocks. Proceedings of the National Academy of Sciences of the United States of America **89**(22) (November 1992) 10915–9
14. Altschul, S.F., Gish, W., Miller, W., Myers, E.W., Lipman, D.J.: Basic local alignment search tool. Journal of molecular biology **215**(3) (October 1990) 403–10
15. Remmert, M., Biegert, A., Hauser, A., Söding, J.: HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment. Nature methods **9**(2) (February 2012) 173–5
16. The UniProt Consortium: UniProt: a hub for protein information. Nucleic Acids Research **43**(D1) (October 2014) D204–D212
17. Tomii, K., Kanehisa, M.: Analysis of amino acid indices and mutation matrices for sequence comparison and structure prediction of proteins. Protein engineering **9**(1) (1996) 27–36
18. McLachlan, A.D.: Repeating sequences and gene duplication in proteins. Journal of molecular biology **64**(2) (March 1972) 417–37
19. Koshi, J.M., Goldstein, R.A.: Context-dependent optimal substitution matrices. Protein engineering **8**(7) (July 1995) 641–5
20. Overington, J., Donnelly, D., Johnson, M.S., Sali, A., Blundell, T.L.: Environment-specific amino acid substitution tables: tertiary templates and prediction of protein folds. Protein science : a publication of the Protein Society **1**(2) (February 1992) 216–26

21. Lin, K., May, A.C., Taylor, W.R.: Amino Acid Substitution Matrices from an Artificial Neural Network Model. Journal of Computational Biology **8**(5) (October 2001) 471–481
22. Cserzö, M., Bernassau, J.M., Simon, I., Maigret, B.: New alignment strategy for transmembrane proteins. Journal of molecular biology **243**(3) (October 1994) 388–96
23. Altschul, S.F.: Amino acid substitution matrices from an information theoretic perspective. Journal of molecular biology **219**(3) (June 1991) 555–65
24. Azarya-Sprinzak, E., Naor, D., Wolfson, H.J., Nussinov, R.: Interchanges of spatially neighbouring residues in structurally conserved environments. Protein engineering **10**(10) (October 1997) 1109–22
25. George, D.G., Barker, W.C., Hunt, L.T.: Mutation data matrix and its uses. Methods in enzymology **183** (January 1990) 333–51
26. Russell, R.B., Saqi, M.A., Sayle, R.A., Bates, P.A., Sternberg, M.J.: Recognition of analogous and homologous protein folds: analysis of sequence and structure conservation. Journal of molecular biology **269**(3) (June 1997) 423–39
27. Grantham, R.: Amino acid difference formula to help explain protein evolution. Science (New York, N.Y.) **185**(4154) (September 1974) 862–4
28. Tatjewski, M., Bańko, M., Kucińska, A., Rączaszek-Leonardi, J.: Computational distributional semantics and free associations: a comparison of two word-similarity models in a study of synonyms and lexical variants. In Pęzik, P., Waliński, J., Kosecki, K., eds.: Language, Corpora and Cognition. Peter Lang (In press)

Uplift Modelling Using Kernel Support Vector Machines

Łukasz Zaniewicz[1] and Szymon Jaroszewicz[1,2]

[1] Institute of Computer Science, Polish Academy of Sciences,
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
[2] National Institute of Telecommunications,
ul. Szachowa 1, 04-894 Warsaw, Poland

Abstract.
Uplift modeling is a relatively new branch of Machine Learning initially used for marketing campaigns to determine their incremental impact. In contrast to conventional classification methods, it does not predict the response (or response probabilities) itself, but instead, the difference in those probabilities resulting from this campaign. In other words, it aims to model the causal effect of an applied action on a given individual. But uplift modeling is not restricted only to marketing, the second straightforward application is controlled clinical trail. The main assumption here is that our population is divided into two groups: treatment, where the action was taken, and control, which plays the role of a background. In this paper we present a modification of L_1 Support Vector Machines designed specifically for needs of uplift modeling. The standard SVM optimization task has been reformulated in order to explicitly model the difference in response behavior between treatment and control datasets. The resulting model can make three different predictions on a given object: whether the response to an applied action will be positive, neutral or negative. Finally we compare nonlinear Uplift SVMs and demonstrate their performance.

1 Introduction

Unlike the traditional classification methods like logistic regression or Support Vector Machines, where the model predicts the conditional class probability distribution, the uplift modelling aims at predicting the incremental response of some investigated action. Standard classification methods focus entirely on the effect after the action has been taken and do not take into account possibility of not taking the action. And that is the idea behind the uplift modelling: it tries to model what happens *because* of the action. To achieve this, uplift methods require the dataset to be divided into two groups: the treatment group, where

the action has been performed, and the control, used as a background, which was not subject to this action.

The idea of uplift is the easiest to understand in the context of marketing campaigns. In fact, customers who were subject to a given campaign and had a positive outcome (bought some product) can be divided into two categories: those who bought because they were induced and those who would have bought it anyway. Of course, only the first group is really valuable from the marketers point of view. Notice that traditional classification methods are unable to distinguish between those two groups. Moreover, an analogous division can be made on clients with negative outcome, where we should pay special attention to detect those, who did not buy *because* they were targeted (e.g they got annoyed). Marketers, in all possible ways, should exclude this group from the campaign. Uplift methods try to explicitly model the difference in outcome probabilities between the control and treatment groups, hence they are able to predict whether the result of the action will be truly positive, neutral or negative.

In this paper we describe a modification of Support Vector Machines designed specifically for needs of uplift modeling. The resulting model handles two training datasets and can make three different predictions on a given object: whether the response to an applied action will be positive, neutral or negative. Moreover, the kernel trick was applied in order to create nonlinear classifiers. In Section 4 we will experimentally compare the performance of a linear uplift SVM with its nonlinear modifications, where polynomial and radial basis function kernels were used.

1.1 Previous work

Surprisingly, uplift modeling has received relatively little attention in the literature. The most obvious approach uses two separate probabilistic models, one built on the treatment and the other on the control dataset, and subtracts their predicted probabilities. The advantage of the two-model approach is that it can be applied with any classification model. Moreover, if uplift is strongly correlated with the class attribute itself, or if the amount of training data is sufficient for the models to predict the class probabilities accurately, the two-model approach will perform very well also in the uplift case. The disadvantage is, that when uplift follows a different pattern than the class distributions, both models will focus on predicting the class, instead of focusing on the weaker 'uplift signal'. See [1] for an illustrative example.

A few papers addressed decision tree construction for uplift modeling. See e.g. [2, 1]. In [3] uplift decision trees have been presented which are more in line with modern machine learning algorithms. The approach has been extended to the case of multiple treatments in [4].

Some regression techniques for uplift modeling are available. Most researchers, however, follow the two model approach either explicitly or implicitly [5, 6]. In [7] a method has been presented which makes it possible to convert a classical logistic regression model (or in fact any other probabilistic classifier) into an uplift

model. The approach is based on a class variable transformation. Recent, thorough literature overviews on uplift modeling can be found in [3] and [1].

Support Vector Machines with parallel hyperplanes, similar to our approach, have been analyzed in the context of ordinal classification [8]; here the situation is different as two training datasets are involved.

Recently another approach to Support Vector Machines applied to uplift modeling has been published [9]. It is based on structured SVMs used to directly maximize the area under uplift curves.

A preliminary version of this paper appeared in [10]. The current paper extends that first version mainly with an application of kernels. All experiments are focused on this issue. Moreover, the list of datasets under comparison has been largely refreshed.

2 Classic Support Vector Machines algorithm

Before we move on to Uplift SVM, let us make a short revision of the linear Support Vector Machines.

2.1 Linear SVM

The idea of Support Vector Machine (SVM) method was first introduced by Vapnik, Chervonenkis and Lerner in early 60's [11, 12]. Later, in 1992 Vapnik with his team [13] suggested a way to apply the so-called kernel-trick to create nonlinear classifiers and finally, in 1995, also Vapnik with Corinna Cortes [14] introduced the most popular nowadays soft-margin approach.

We start with introducing the notation. Let us consider n-points training sample $\mathbf{D} = \{(\mathbf{x}_i, y_i) : i = 1, \ldots, n\}$, where $\mathbf{x}_i \in \mathbb{R}^m$ are the values of the predictor variables and $y_i \in \{-1, 1\}$ is the class of the i-th data point. The class $+1$ is considered as the *positive*, or desired outcome. By $\langle \mathbf{x}_1, \mathbf{x}_2 \rangle$ we denote the scalar product of vectors $\mathbf{x}_1$ and $\mathbf{x}_2$.

At first we consider the simplest, linearly separable case. Then, there exists some hyperplane H which separates the positive from the negative data points. This is called the separating hyperplane and it has the following form

$$\langle \mathbf{w}, \mathbf{x} \rangle + b = 0, \tag{1}$$

where $\mathbf{w}$ is the normal vector to the hyperplane H and $b \in \mathbb{R}$. In the linearly separable case there also exist two hyperplanes $\langle \mathbf{w}, \mathbf{x} \rangle + b = -1$ and $\langle \mathbf{w}, \mathbf{x} \rangle + b = +1$ that also separate the training sample and there are no data points between them. The goal is to maximize the distance between them, which geometrically equals $\frac{2}{||\mathbf{w}||}$, hence, instead of this maximization one can minimize $||\mathbf{w}||$ (or equivalently its square $||\mathbf{w}||^2 = \langle \mathbf{w}, \mathbf{w} \rangle$) subject to constraints of the (simplified) form

$$y_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b) \geq +1 \text{ for } i = 1, \ldots, n, \tag{2}$$

which are the consequence of linear separability.

Hence, here we have a well-known quadratic optimization problem with linear constraints. It has a corresponding dual problem of the form

$$\max_{\alpha_i} L(\alpha) = \max_{\alpha_i} \min_{\mathbf{w}} \frac{1}{2}\langle \mathbf{w}, \mathbf{w}\rangle + \sum_{i=1}^{n} \alpha_i \left[y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) - 1\right] \tag{3}$$

$$= \max_{\alpha_i} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j\rangle. \tag{4}$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n), \alpha_i \geq 0$ are Lagrange multipliers. See [15] for more datailed description.

In case when the data points are not linearly separable, a small modification is needed to handle with mislabeled examples. In such situation there are introduced non-negative slack variables ξ_i, which measure the "degree of misclassification" of i-th example. The constraints become

$$y_i(\langle \mathbf{w}, \mathbf{x}_i\rangle + b) \geq +1 - \xi_i, \qquad \xi_i \geq 0 \quad \text{for } i = 1, \ldots, n. \tag{5}$$

The goal function is then increased by non-zero ξ_i and the optimization problem becomes a kind of a trade-off between a wide margin and low error caused by misclassification. Then the primal problem has the form

$$\arg\min_{\mathbf{w}, b, \xi} \frac{1}{2}\langle \mathbf{w}, \mathbf{w}\rangle + C \sum_{i=1}^{n} \xi_i, \tag{6}$$

subject to constraints (5). Constant C is the penalty coefficient, its large values pay more attention to the missclassified examples. The dual form is almost exactly the same as in the separable case, slack variables ξ_i vanish from the formulation and the only one difference is that Lagrange multipliers α_i are bounded by the penalty coefficient C i.e. $0 \leq \alpha_i \leq C$.

2.2 Kernel trick

The so-called "kernel trick" owes its name to the application of kernel functions in order to transfer the considered problem to high dimensional one, implicit feature space, where we even do not need to compute the coordinates of the training sample, but instead, we simply compute the scalar product between the images of all pairs of examples in the feature space, what usually is easier and computationally cheaper than explicit computation of the new coordinates. Support Vector Machines is probably the most popular application of kernel trick. Due to the fact that in the dual problem the data appears solely in the scalar products, the application of the kernel trick is absolutely straightforward. In general, we simply substitute those scalar products with some desired kernel function.

Let us suppose that we have a mapping $\phi : \mathbb{R}^m \to \mathcal{T}$, which maps our data to some other (possibly infinite dimensional) Euclidean space $\mathcal{T}$. Then

the only dependence on the data in the learning alogrithm would be through scalar products $\langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$ in the new space $\mathcal{T}$. Now, if there exists a kernel function K, such that $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$, then we may use only K in the optimization and, what is tricky here, we do not need to know ϕ at all - we only require that K is a positive-semidefinite, symmetrical kernel function, which, based on Mercer's theorem, represent the scalar product in the implicit feature space $\mathcal{T}$.

In case of the SVM, the (4) becomes

$$\max_{\alpha_i} L(\alpha) = \max_{\alpha_i} \big[\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \big]. \tag{7}$$

There are many possible kernel functions, but the most popular are

1. polynomial: $K(\mathbf{x}_i, \mathbf{x}_j) = (\langle \mathbf{x}_i, \mathbf{x}_j \rangle + 1)^d$,
2. gaussian radial basis function (RBF): $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma ||\mathbf{x}_i - \mathbf{x}_j||^2)$,
3. sigmoid: $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\alpha \langle \mathbf{x}_i, \mathbf{x}_j \rangle + 1)$.

In general, the parameters d, γ, α are used to adjust the degree of nonlinearity. In this paper we will present the results using polynomial and RBF kernel functions. We will compare it with the linear SVM (in fact, it might be also considered as a kernel SVM with $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$).

3 Uplift Support Vector Machines

SVMs are designed primarily for classification, not probability modeling, so in order to adapt the SVMs to the analyzed setting, we first recast the uplift modeling problem as a three-class classification problem. This differs from the typical formulation which aims at predicting the difference in class probabilities between treatment and control groups.

Unlike standard classification, in uplift modeling we have two training samples: the *treatment group*, $\mathbf{D}^T = \{(\mathbf{x}_i, y_i) : i = 1, \ldots, n^T\}$ and the *control group* $\mathbf{D}^C = \{(\mathbf{x}_i, y_i) : i = 1, \ldots, n^C\}$, where $x_i \in \mathbb{R}^m$ are the values of the predictor variables, and $y_i \in \{-1, 1\}$ is the class of the i-th data record, m is the number of attributes in the data, and n^T and n^C are the numbers of records in the treatment and control groups respectively. Objects in the treatment group have been subject to some *action* or *treatment*, while objects in the control group have not.

In the rest of the paper we will continue to follow the convention that all quantities related to the treatment group will be denoted with superscript ^{T} and those related to the control group with superscript ^{C}.

An *uplift model* is defined as a function

$$M(\mathbf{x}) : \mathbb{R}^m \to \{-1, 0, 1\}, \tag{8}$$

which assigns to each point in the input space one of the values $+1$, 0 and -1, interpreted, respectively, as positive, neutral and negative impact of the action.

In other words, the positive prediction +1 means that we expect the objects class to be +1 if it is subject to treatment and −1 if it is not, the negative prediction means that we expect the class to be −1 after treatment and +1 if no action was performed, and neutral if the object's class is identical (either +1 or −1) regardless of whether the action was taken or not.

The proposed Uplift Support Vector Machine (USVM), which performs uplift prediction, uses two parallel hyperplanes

$$H_1 : \langle \mathbf{w}, \mathbf{x} \rangle + b_1 = 0 \qquad H_2 : \langle \mathbf{w}, \mathbf{x} \rangle + b_2 = 0,$$

where $b_1, b_2 \in \mathbb{R}$ are the intercepts and if $b_2 \geq b_1$ then the model is valid; in Lemma 1 we will give sufficient conditions for this inequality to hold. The model predictions are specified by the following equation

$$M(\mathbf{x}) = \begin{cases} +1 & \text{if } \langle \mathbf{w}, \mathbf{x} \rangle + b_1 > 0, \\ 0 & \text{if } \langle \mathbf{w}, \mathbf{x} \rangle + b_1 \leq 0 \text{ and } \langle \mathbf{w}, \mathbf{x} \rangle + b_2 > 0, \\ -1 & \text{if } \langle \mathbf{w}, \mathbf{x} \rangle + b_2 \leq 0. \end{cases} \tag{9}$$

Intuitively, the point is classified as positive if it lies on the positive side of both hyperplanes, neutral if it lies on the positive side of hyperplane H_2 only, and classified as negative if it lies on the negative side of both hyperplanes. In other words, H_1 separates positive and neutral points, and H_2 neutral and negative points.

Let us now formulate the optimization task which allows for finding the model's parameters $\mathbf{w}, b_1, b_2$. We will use $\mathbf{D}_+^T = \{(\mathbf{x}_i, y_i) \in \mathbf{D}^T : y_i = +1\}$ to denote data points belonging to the positive class in the treatment group and $\mathbf{D}_-^T = \{(\mathbf{x}_i, y_i) \in \mathbf{D}^T : y_i = -1\}$ to denote points in that group belonging to the negative class. Analogous notation is used for points in the control group. Denote $n = |\mathbf{D}^T| + |\mathbf{D}^C|$.

The parameters of an USVM can be found by solving the following optimization problem, which we call the *USVM optimization problem*.

$$\begin{aligned} \min_{\mathbf{w}, b_1, b_2 \in \mathbb{R}^{m+2}} \frac{1}{2}\langle \mathbf{w}, \mathbf{w} \rangle + C_1 \sum_{\mathbf{D}_+^T \cup \mathbf{D}_-^C} \xi_{i,1} + C_2 \sum_{\mathbf{D}_-^T \cup \mathbf{D}_+^C} \xi_{i,1} \\ + C_2 \sum_{\mathbf{D}_+^T \cup \mathbf{D}_-^C} \xi_{i,2} + C_1 \sum_{\mathbf{D}_-^T \cup \mathbf{D}_+^C} \xi_{i,2}, \end{aligned} \tag{10}$$

subject to the following constraints

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b_1 \geq +1 - \xi_{i,1}, \text{ forall } (\mathbf{x}_i, y_i) \in \mathbf{D}_+^T \cup \mathbf{D}_-^C, \tag{11}$$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b_1 \leq -1 + \xi_{i,1}, \text{ forall } (\mathbf{x}_i, y_i) \in \mathbf{D}_-^T \cup \mathbf{D}_+^C, \tag{12}$$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b_2 \geq +1 - \xi_{i,2}, \text{ forall } (\mathbf{x}_i, y_i) \in \mathbf{D}_+^T \cup \mathbf{D}_-^C, \tag{13}$$

$$\langle \mathbf{w}, \mathbf{x}_i \rangle + b_2 \leq -1 + \xi_{i,2}, \text{ forall } (\mathbf{x}_i, y_i) \in \mathbf{D}_-^T \cup \mathbf{D}_+^C, \tag{14}$$

$$\xi_{i,j} \geq 0, \text{ forall } i = 1, \ldots, n, \ j \in \{1, 2\}, \tag{15}$$

where C_1, C_2 are penalty parameters and $\xi_{i,j}$ slack variables allowing for misclassified training cases. Note that $\xi_{i,1}$ and $\xi_{i,2}$ are slack variables related to the hyperplane H_1 and H_2 respectively. We will now give an intuitive justification for this formulation of the optimization problem.

Below, when we talk about distance of a point from a plane and point lying on a positive or negative side of the plane we implicitly assume that the width of the margin is also taken into account.

The situation is graphically depicted in Figure 1. Example points belonging to $\mathbf{D}_+^T$ are marked with T_+, points belonging to $\mathbf{D}_-^T$, respectively with T_-. Analogous notation is used for example points in the control group which are marked with C_+ and C_-.

In an ideal situation, we would want points for which a positive $(+1)$ prediction is made to contain only cases in $\mathbf{D}_+^T$ and $\mathbf{D}_-^C$, that is only points which do not contradict the positive effect of the action. Note that for the remaining points, which are in $\mathbf{D}_-^T$ or in $\mathbf{D}_+^C$, the effect of an action can at best be neutral. Therefore points in $\mathbf{D}_+^T$ and $\mathbf{D}_-^C$ (marked T_+ and C_- respectively in the figure) are not penalized when on the positive side of hyperplane H_1. Analogously points in $\mathbf{D}_-^T$ and $\mathbf{D}_+^C$ (marked T_- and C_+) which are on the negative side of H_2 are not penalized. Points in $\mathbf{D}_+^T$ and $\mathbf{D}_-^C$ which lie on the negative side of H_1 are penalized with penalty $C_1\xi_{i,1}$ where ξ_i is the distance of the point from the plane (in fact, the true distance is equal to $\frac{\xi_i}{||w||}$, but for simplicity we use only ξ_i) and C_1 is a penalty coefficient. Those penalties prevent the model from being overly cautious and classifying all points as neutral (see Lemmas 2 and 3 in the next section). Analogous penalty is introduced for points in $\mathbf{D}_-^T$ and $\mathbf{D}_+^C$ in the fifth term of (10). In Figure 1, those points are sandwiched between H_1 and H_2, and their penalties are marked with red arrows.

Consider now points in $\mathbf{D}_+^T$ and $\mathbf{D}_-^C$ which lie on the negative side of both hyperplanes, i.e. in the region where the model predicts a negative impact (-1). Clearly, model's predictions are wrong in this case, since if the outcome was positive in the treatment group the impact of the action can only be positive or neutral. Those data points are thus additionally penalized for being on the wrong side of the hyperplane H_2 with penalty $C_2\xi_{i,2}$. Analogous penalty is of course applied to points in $\mathbf{D}_-^T$ and $\mathbf{D}_+^C$ which lie on the positive side of both hyperplanes. Additional penalties are marked with dashed blue arrows in the figure.

To summarize, the penalty coefficient C_1 is used to punish points being on the wrong side of a single hyperplane (red arrows in Figure 1) and the coefficient C_2 controls additional penalty incurred by a point being on the wrong side of also the second hyperplane (dashed blue arrows in Figure 1). In the next section we give a more detailed analysis of how the penalties influence the model's behavior.

3.1 Properties of the Uplift Support Vector Machines (USVMs)

In this section we are going to analyze some mathematical properties of Uplift Support Vector Machines (USVMs), especially in the context of influence of the

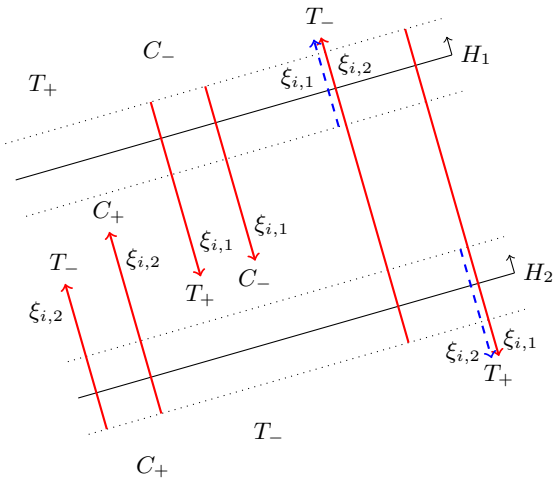


Fig. 1. The Uplift SVM optimization problem. Example points belonging to the positive class in the treatment and control groups are marked respectively with T_+ and C_+. Analogous notation is used for points in the negative class. The figure shows penalties incurred by points with respect to the two hyperplanes of the USVM. Positive sides of hyperplanes are indicated by small arrows at the right ends of lines in the image. Red arrows denote the penalties incurred by points which lie on the wrong side of a single hyperplane, blue dashed arrows denote additional penalties for being misclassified also by the second hyperplane.

parameters C_1 and C_2 on model's behavior. One of the more important results is how the ratio of penalty parameters $\frac{C_2}{C_1}$ directly *influences* the number of records which are classified as neutral, or, in other words, how it influences the distance between the two separating hyperplanes. This also sheds light on the interpretation of the model.

Lemma 1. *Let* $\mathbf{w}^*, b_1^*, b_2^*$ *be a solution to the Uplift SVM optimization problem given by Equations 10-15. If* $C_2 \geq C_1$ *then* $b_2^* \geq b_1^*$.

The proof of this and the remaining lemmas can be found in the in [10]. The lemma guarantees that the problem possesses a well defined solution in the sense of Equation 9. Moreover it naturally constrains the penalty C_2 to be greater than or equal to C_1. From now on, instead of working with the coefficient C_2, it will be more convenient to talk about the penalty coefficient C_1 and the quotient $\frac{C_2}{C_1} \geq 1$ determining how many times is C_2 is greater than C_1.

Lemma 2. *For sufficiently large value of* $\frac{C_2}{C_1}$ *none of the observations is penalized with a term involving the* C_2 *factor in the solution to the USVM optimization problem.*

Equivalently the lemma states that for a large enough value of $\frac{C_2}{C_1}$, none of the points will be on the wrong side of both hyperplanes. This is possible only when the hyperplanes are maximally separated, resulting in most (often all) points classified as neutral.

Lemma 3. *If $C_1 = C_2 = C$ and the solution is unique then both hyperplanes coincide: $b_1 = b_2$.*

We are now ready to give an interpretation of the C_1 and $\frac{C_2}{C_1}$ parameters of the Uplift SVM. The parameter C_1 plays the role analogous to the penalty coefficient C in classical SVMs controlling the relative cost of misclassified points with respect to the margin maximization term $\frac{1}{2}\langle \mathbf{w}, \mathbf{w} \rangle$. The quotient $\frac{C_2}{C_1}$ allows the analyst to decide what proportion of points should be classified as positive or negative. In other words, it allows for controlling the size of the neutral prediction.

Note that this is *not* equivalent to selecting thresholds in data scored using a single model. For each value of $\frac{C_2}{C_1}$ a different model is built which is optimized for a specific proportion of positive and negative predictions. We believe that this property of USVMs is very useful for practical applications, as it allows for tuning the model specifically to the desired size of the campaign.

3.2 The Uplift Support Vector Machine optimization task

Let us now present the dual of the Uplift Support Vector Machine optimization task and discuss methods of solving it.

We will first introduce a class variable transformation

$$z_i = \begin{cases} y_i, & \text{if } (\mathbf{x}_i, y_i) \in \mathbf{D}^T, \\ -y_i, & \text{if } (\mathbf{x}_i, y_i) \in \mathbf{D}^C. \end{cases}$$

In other words, z_i is obtained by keeping the class variable in the treatment group and reversing it in the control. Note that this is the same transformation which has been introduced in [7] in the context of uplift modeling and logistic regression.

This variable transformation allows us to simplify the optimization problem given in Equations 10-15 by merging (11) with (12) and (13) with (14). The simplified optimization problem is

$$\min_{\mathbf{w}, b_1, b_2 \in \mathbb{R}^{m+2}} \frac{1}{2}\langle \mathbf{w}, \mathbf{w} \rangle + C_1 \sum_{\mathbf{D}^T_+ \cup \mathbf{D}^C_-} \xi_{i,1} + C_2 \sum_{\mathbf{D}^T_- \cup \mathbf{D}^C_+} \xi_{i,1} + C_2 \sum_{\mathbf{D}^T_+ \cup \mathbf{D}^C_-} \xi_{i,2} + C_1 \sum_{\mathbf{D}^T_- \cup \mathbf{D}^C_+} \xi_{i,2},$$

subject to constraints

$$\begin{aligned} z_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b_1) - 1 + \xi_{i,1} \geq 0 \text{ forall } i = 1, \ldots, n, \\ z_i(\langle \mathbf{w}, \mathbf{x}_i \rangle + b_2) - 1 + \xi_{i,2} \geq 0 \text{ forall } i = 1, \ldots, n, \\ \xi_{i,j} \geq 0, \text{ forall } i = 1, \ldots, n, \ j \in \{1, 2\}. \end{aligned}$$

We will now obtain the dual form of the optimization problem. We begin by writing the following Lagrange function

$$\begin{aligned}
L(\mathbf{w}, b_1, b_2, \alpha_i, \beta_i, \xi_{i,1}, \xi_{i,2}, r_i, p_i) \\
&= \frac{1}{2}\langle \mathbf{w}, \mathbf{w} \rangle + C_1 \sum_{+1} \xi_{i,1} + C_2 \sum_{-1} \xi_{i,1} \\
&\quad + C_2 \sum_{+1} \xi_{i,2} + C_1 \sum_{-1} \xi_{i,2} \\
&\quad - \sum_{i=1}^{n} \alpha_i \big(z_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b_1) - 1 + \xi_{i,1} \big) \\
&\quad - \sum_{i=1}^{n} \beta_i \big(z_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b_2) - 1 + \xi_{i,2} \big) \\
&\quad - \sum_{i=1}^{n} r_i \xi_{i,1} - \sum_{i=1}^{n} p_i \xi_{i,2},
\end{aligned}$$

where $\sum_{+1}$ and $\sum_{-1}$ denote sums over all examples in $\mathbf{D}^T \cup \mathbf{D}^C$ for which $z_i = 1$ and $z_i = -1$ respectively; $\alpha_i, \beta_i \in \mathbb{R}$ are Lagrange multipliers and $r_i, p_i \geq 0$.

Now we need to calculate partial derivatives and equate them to 0 in order to satisfy Karush-Kuhn-Tucker conditions. We begin by deriving w.r.t. $\mathbf{w}$

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{n} \alpha_i z_i \mathbf{x}_i - \sum_{i=1}^{n} \beta_i z_i \mathbf{x}_i = 0,$$

from which we obtain

$$\mathbf{w} = \sum_{i=1}^{n} (\alpha_i + \beta_i) z_i \mathbf{x}_i. \tag{16}$$

We obtain the remaining derivatives in a similar fashion

$$\frac{\partial L}{\partial b_1} = -\sum_{i=1}^{n} \alpha_i z_i = 0, \qquad \frac{\partial L}{\partial b_2} = -\sum_{i=1}^{n} \beta_i z_i = 0, \tag{17}$$

$$\frac{\partial L}{\partial \xi_{i,1}} = C_1 \mathbb{1}_{[z_i = -1]} + C_2 \mathbb{1}_{[z_i = +1]} - \alpha_i - r_i = 0, \tag{18}$$

$$\frac{\partial L}{\partial \xi_{i,2}} = C_1 \mathbb{1}_{[z_i = +1]} + C_2 \mathbb{1}_{[z_i = -1]} - \beta_i - p_i = 0. \tag{19}$$

Plugging those equations back into the Lagrange function we obtain, after simplifications,

$$\begin{aligned}
L = \frac{1}{2}\langle \mathbf{w}, \mathbf{w} \rangle - \sum_{i=1}^{n} \alpha_i \big(z_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b_1) - 1 \big) \\
- \sum_{i=1}^{n} \beta_i \big(z_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b_2) - 1 \big).
\end{aligned}$$

Substituting $\mathbf{w}$ from Equation 16 and simplifying further we get

$$\begin{aligned} L =& \frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i+\beta_i)(\alpha_j+\beta_j)z_iz_j\langle\mathbf{x}_i,\mathbf{x}_j\rangle \\ &- \sum_{i,j=1}^{n}(\alpha_i+\beta_i)(\alpha_j+\beta_j)z_iz_j\langle\mathbf{x}_j,\mathbf{x}_i\rangle \\ &- b_1\sum_{i=1}^{n}\alpha_iz_i+\sum_{i=1}^{n}\alpha_i-b_2\sum_{i=1}^{n}\beta_iz_i+\sum_{i=1}^{n}\beta_i \\ =& \sum_{i=1}^{n}(\alpha_i+\beta_i)-\frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i+\beta_i)(\alpha_j+\beta_j)z_iz_j\langle\mathbf{x}_i,\mathbf{x}_j\rangle, \end{aligned} \tag{20}$$

which we maximize over α_i, β_i.

Finally, from the assumption that $r_i, p_i \geq 0$ and (18), (19) combined with the KKT condition on nonnegativity of α_i, β_i and from (17) we obtain the following constraints for the dual optimization problem

$$0 \leq \alpha_i \leq C_1\mathbb{1}_{[z_i=-1]} + C_2\mathbb{1}_{[z_i=+1]}, \tag{21}$$

$$0 \leq \beta_i \leq C_1\mathbb{1}_{[z_i=+1]} + C_2\mathbb{1}_{[z_i=-1]}, \tag{22}$$

$$\sum_{i=1}^{n}\alpha_iz_i = \sum_{i=1}^{n}\beta_iz_i = 0. \tag{23}$$

And the last thing that we mention in this section is the application of kernel functions in order to obtain nonlinear model. In case of uplift Support Vector Machines it is exactly the same as in the case of a standard SVM. We simply substitute the scalar product in (20) with some symmetric, positive-semidefinite kernel function

$$L = \sum_{i=1}^{n}(\alpha_i+\beta_i)-\frac{1}{2}\sum_{i,j=1}^{n}(\alpha_i+\beta_i)(\alpha_j+\beta_j)z_iz_jK(\mathbf{x}_i,\mathbf{x}_j), \tag{24}$$

while the Constraints 21 - 23 remain unchanged. As we mentioned before, in the experimental evaluation we have used the polynomial (of degree $d = 3$) and the radial basis function (with $\gamma = 1$) kernels.

The problem presented above we solve using the quadratic and convex solvers from the CVXOPT library [16]. We also have developed dedicated solvers for the Karush-Kuhn-Tucker (KKT) systems of equations needed to solve our USVM optimization problems, but we do not present the details here.

4 Experimental evaluation

4.1 Evaluation of uplift models

Let us now discuss evaluation of uplift models using so called uplift curves. One of the tools for assessing performance of standard classification models are lift

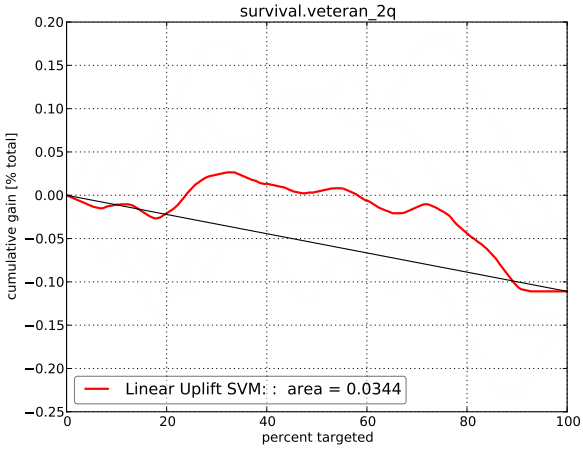


Fig. 2. Example uplift curve for the `survival-veteran` dataset for Uplift SVM with the linear kernel. The x-axis represents the percentage of the population to which the action has been applied and the y-axis the net gain from performing the action.

curves (also known as cumulative gains curves or cumulative accuracy profiles). In a lift curve, the x axis corresponds to the number of cases targeted and the y axis to the number of successes captured by the model. In our case both numbers will be expressed as percentage of the total population.

The *uplift curve* is computed by subtracting the lift curve obtained on the control test set from the lift curve obtained on the treatment test set. Both curves are generated using the same uplift model. Recall the number of successes on the y axis is expressed as a percentage of the total population which guarantees that the curves can be meaningfully subtracted. The interpretation of the uplift curve is as follows: on the x axis we select the percentage of the population on which an action is performed and on the y axis we read the difference between the success rates in the treatment and control groups. A point at $x = 100\%$ gives the gain in success probability we would obtain if the action was performed on the whole population. A diagonal line corresponds random selection. The Area Under the Uplift Curve (AUUC) can be used as a single number summarizing model performance. In this paper we subtract the area under the diagonal line from this value in order to obtain more meaningful numbers. More details on evaluating uplift models and on uplift curves can be found in [3, 1].

Figure 2 shows uplift curves for `survival-veteran` dataset(short description about this dataset is presented in the next paragraph) for Uplift SVM with the linear kernel. It can be seen that targeting between 20% and 80% of the population gives significant gains in net success rate over targeting nobody or the whole population. It can be seen that applying the action only to some proportion of the population leads to significant gains in net success rate. The curve in the

figure have been generated by averaging over 32 random train test splits; the same method, but with more splits, has been used for other experiments in this section and is described in detail below.

4.2 Benchmark datasets and experiment setup

One of the major difficulties in working on the uplift modeling is the lack of publicly available datasets. Despite the fact that control groups are quite common in clinical trails and marketing campaign, there are relatively few publicly available datasets with reasonably large control group and at least several predictors. Furthermore, clinical trials usually involve censored data and most machine learning methods, including uplift modeling tools, do not directly allow for the use of such data. But Rzepakowski and Jaroszewicz [17] demonstrated that, under reasonable assumptions, one can easily apply uplift modeling to survival data, without messing up the correctness of models' decisions. Before we present a very brief description of each dataset, we first explain how the conversion of survival time to binary (positive/negative) outcome was performed. In order to obtain a balanced class distribution we have used median of observed (censored) survival times. All but one datasets was transformed this way. The only exception was the dataset `colon_lev_recur`, where third quartile was used as a threshold. See [17] for more datailed description.

The first dataset called `tamoxifen` comes from a book on survival analysis by Pintilie [18] and contains the data on therapy of breast cancer with a drug tamoxifen. In this clinical trial the treated group received tamoxifen combined with radio-therapy, while the control group received tamoxifen alone. We attempt to model the target variable `stat` describing whether the patient was alive at the time of the last follow-up. The dataset contains six predictor variables. Details can be found in [18].

Two next datasets come from the `R` package called `KMsurv`. The `burn` dataset describes patients who suffered from underwent burns. The body cleansing was applied to patients in the treatment group while the control group had routine baths. Occurrence of staphylococcus aureus infection was the negative outcome. The second dataset is called `hodg` and it comes from clinical trail, where 43 patients underwent an allogeneic graft or an autologous graft (control group) as a lymphoma treatment.

Additional datasets come from the another `R` package called `survival`. We will not discuss them in details since full descriptions are easily accessible online. The `pbc` dataset is the result of a study of primary biliary cirrhosis (PBC) of the liver. 312 patients were randomly divided into the treated group (received D-penicillamine drug) and the control group (placebo). The next dataset called `bladder` contains the data about recurrences of a bladder cancer on 85 patients who received either the thiotepa drug or placebo.
The `colon` data comes from a study where clinicians examined an adjuvant chemotherapy for colon cancer. There are two types of treatment: levamisole and levamisole combined with 5-FU (Fluorouracil). The control group received placebo. Two types of outcomes were recorded: death and disease recurrence.

dataset	Linear	RBF	Poly
tamoxifen	**0.0138**	-0.0089	-0.0096
kmsurv-burn	0.0309	0.0336	**0.0339**
kmsurv-hodq	0.0691	**0.0884**	0.0789
survival-bladder	**-0.0358**	-0.0431	-0.0441
survival-pbc	-0.0138	**-0.0105**	-0.0114
survival-colon-death	**-0.0025**	-0.0107	-0.0121
survival-colon-lev-death	-0.0022	**0.0048**	**0.0048**
survival-colon-lev-recur	**-0.0174**	-0.0204	-0.0226
survival-colon-lev-5fu-death	-0.0045	**-0.0008**	-0.0018
survival-colon-lev-5fu-recur	-0.0026	0.0007	**0.001**
survival-colon-recur	**-0.001**	-0.0021	-0.0021
survival-veteran	0.0342	0.0396	**0.0411**

Table 1. Areas under the Uplift Curve for kernel Uplift Support Vector Machines.

Hence, we have two possible modeling targets and we analyzed it separately. All in all, `colon` data eventually gives us six datasets: three therapies (two treatment, one placebo) times two target attributes. Resulting datasets are called respectively `colon-death`, `colon-recur`, `colon-lev-death`, `colon-lev-recur`, `colon-lev-5fu-death` and `colon-lev-5fu-recur`.
Finally, the `veteran` data comes from a randomized clinical trial of lung cancer where 137 patients were involved.

4.3 Performance evaluation of uplift SVM

We will now compare the performance of Uplift Support Vector Machines with different kernel functions applied. As mentioned before, we used three of them: linear (classic SVM), polynomial with degree of 3 and Gaussian radial-basis function (with parameter $\gamma = 1$). Those parameters were not subject to tuning procedure. The penalty parameter C_1 in all models have been chosen from the set $\{10^{-2}, 10^{-1}, \ldots, 10^{3}\}$, while the proportion $\frac{C_2}{C_1}$ had 11 possible values uniformly distributed in the range $[1.0, 3.0]$. For each grid point 5-fold cross-validation was used to measure model performance.

Table 4.3 compares Areas under the Uplift Curve for Uplift SVMs with three used kernel functions on all benchmark datasets. The areas are given in terms of percentages of the total population (used also on the y-axis). Testing was performed by repeating 256 times a random train/test split with 80% of data used for training (and cross-validation based parameter tuning). The remaining 20% were used for testing. Large number of repetitions reduces the influence of randomness in model testing and construction, making the experiments repeatable. Cases when a given method performs best are marked in bold.

It can be seen that both introduced nonlinear versions of uplift SVM do not bring significant improvement, in fact, based on performed experiments, it is hard to point out the winner. The reason of such behavior should be, however,

subject to the further research. But we must remember that, even if the overall performance is not satisfactory, the uplift modeling may still be useful, since it is capable of selecting a subgroup of customers (patients) for which the campaign (treatment) is successful.

5 Conclusions

We have presented an adaptation of Support Vector Machines to the uplift modelling task. The proposed method has been analyzed theoretically including a problem reformulation and properties that clarifies the interpretation of model parameters. The Kernel trick was used in order to create a nonlinear classifiers. Resulting algorithms were tested on real clinical trails datasets. The results of experimental evaluation were good on some datasets, while the others were quite unsatisfactory. The causes of this fact are the direction of future research. We suspect that the proposed approach might suffer from the problem of an instability when small changes of parameter values result in large changes in the model behavior. Another way to improve the performance could be tuning of the kernel parameters. Future research will also include a theoretical analysis of the generalization properties.

6 Acknowledgements

This work was supported by Research Grant no. N N516 414938 of the Polish Ministry of Science and Higher Education (Ministerstwo Nauki i Szkolnictwa Wyższego) from research funds for the period 2010–2014. Ł.Z. was co-funded by the European Union from resources of the European Social Fund. Project POKL 'Information technologies: Research and their interdisciplinary applications', Agreement UDA-POKL.04.01.01-00-051/10-00.

References

1. Radcliffe, N.J., Surry, P.D.: Real-world uplift modelling with significance-based uplift trees. Portrait Technical Report TR-2011-1, Stochastic Solutions (2011)
2. Hansotia, B., Rukstales, B.: Incremental value modeling. Journal of Interactive Marketing **16**(3) (2002) 35–46
3. Rzepakowski, P., Jaroszewicz, S.: Decision trees for uplift modeling. In: Proc. of the 10th IEEE International Conference on Data Mining (ICDM), Sydney, Australia (December 2010) 441–450
4. Rzepakowski, P., Jaroszewicz, S.: Decision trees for uplift modeling with single and multiple treatments. Knowledge and Information Systems (2011)
5. Lo, V.S.Y.: The true lift model - a novel data mining approach to response modeling in database marketing. SIGKDD Explorations **4**(2) (2002) 78–86
6. Larsen, K.: Net lift models: Optimizing the impact of your marketing. In: Predictive Analytics World. (2011) workshop presentation.

7. Jaśkowski, M., Jaroszewicz, S.: Uplift modeling for clinical trial data. In: ICML 2012 Workshop on Machine Learning for Clinical Data Analysis, Edinburgh, Scotland (June 2012)
8. Shashua, A., Levin, A.: Ranking with large margin principle: Two approaches. Advances in neural information processing systems **15** (2002) 937–944
9. Kuusisto, F., Santos Costa, V., Nassif, H., Burnside, E., Page, D., Shavlik, J.: Support vector machines for differential prediction. In: European Conference on Machine Learning (ECML-PKDD). (2014) To appear.
10. Zaniewicz, Ł., Jaroszewicz, S.: Support vector machines for uplift modeling. In: The First IEEE ICDM Workshop on Causal Discovery (CD 2013), Dallas, Texas (December 2013)
11. Lerner, A., Vapnik, V.: Pattern recognition using generalized portrait method. Automation and Remote Control **24** (1963) 774–780
12. Vapnik, V., Chervonenkis, A.: A note on one class of perceptrons. Automation and Remote Control **25** (1963) 821–837
13. B. Boser, I.G., Vapnik, V.: A training algorithm for optimal margin classifiers. In: Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory, ACM Press (1992) 144–152
14. Cortes, C., Vapnik, V.: Support-vector networks. Machine Learning **20(3)** (1995) 273–297
15. Vapnik, V.: The Nature of Statistical Learning Theory. Springer (1999)
16. Andersen, M.S., Dahl, J., Liu, Z., Vandenberghe, L.: Interior-point methods for large-scale cone programming. In: Optimization for Machine Learning. MIT Press (2012) 55–83
17. Jaroszewicz, S., Rzepakowski, P.: Uplift modeling with survival data. In: ACM SIGKDD Workshop on Health Informatics (HI-KDD'14), New York City, USA (August 2014)
18. Pintilie, M.: Competing risks : a practical perspective. John Wiley & Sons Inc. (2006)

Evaluating Multi-level Machine Learning Prediction of Protein-protein Interactions

Julian Zubek[1,2], Marcin Tatjewski[1,2], Subhadip Basu[3] and Dariusz Plewczynski[2]

[1] Institute of Computer Science, Polish Academy of Sciences, ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
[2] Centre of New Technologies, University of Warsaw, ul. Stefana Banacha 2c, 02-097 Warsaw, Poland
[3] Department of Computer Science and Engineering, Jadavpur University, 188, Raja S. C. Mallick Road, Kolkata, West Bengal, India

Abstract. We introduce a novel procedure for evaluating prediction of protein-protein interactions. It takes into account fact that pairwise protein interactions form a larger interaction network. Our procedure guarantees that: a) true positives and true negatives of interacting proteins are formed from the same elements (i.e. they have identical protein composition), b) there is strict separation of proteins between training and test sets. This procedure was applied to previously developed MLPPI (Multi-level machine learning prediction of protein-protein interactions) method and established sequence-based methods. We performed evaluation on high-quality small and medium size data sets containing protein interactions from *Saccharomyces cerevisiae*, *Homo sapiens*, and *Escherichia coli*. Poor performance of all methods (AUC ROC below 0.6) raises a question whether the goal of protein-protein interaction prediction was correctly formulated.

Experimental code and data freely available at:
`http://zubekj.github.io/mlppi/`
(Python implementation, OS independent).

1 Introduction

Proteins are among the most important building blocks of living cells. They are compound objects which can be described in multiple scales: protein primary structure is a linear (1D) sequence of amino acids residues, secondary structure is a sequence of characteristic structural motifs formed along protein chain, and tertiary structure is a full 3D structure of a protein molecule. Interactions between proteins form complex signalling networks, which needs to be reconstructed in as much details as possible in order to understand properties of living organisms at the system level [12]. Various computational tools based

on machine learning are being developed to facilitate this process. Most of these tools focus on predicting binary interactions between pairs of proteins. Among them methods using only 1D protein sequence have the widest applicability, since this kind of information is available for all known proteins. Assessment and comparison of performance of such methods in a realistic setting is a non-trivial task which requires special considerations.

In this work we focus on a thorough evaluation of a multi-level machine learning method for predicting protein-protein interactions, which was developed by Zubek et al. [17]. It differs significantly from the established sequence-based methods because it uses residue-residue interaction prediction as an intermediate step during protein-protein interaction prediction (hence it is called a multi-level approach). In such fashion it introduces 3D structural information during classifier training but utilises only 1D sequence during prediction. The impact of this approach on prediction quality was not yet evaluated properly: so far the method was tested only on a relatively small subset of proteins from *Saccharomyces cerevisiae*. The goal of this work is to compare the performance of multi-level method by Zubek et al. [17] with that of classic sequence-based methods [10] in a realistic setting using larger and more diverse sets of proteins from different organisms: *Saccharomyces cerevisiae* (Yeast), *Homo sapiens* (Human), and *Escherichia coli* (the organisms were chosen based on the availability of the data). In order to meet our goal we develop a novel evaluation schema, which measures predictive power in the context of detecting real compatibility between previously unseen proteins. We construct a balanced set of true negative interactions using interaction network properties. We calculate the performance metrics using modified multi-level cross-validation schema, which takes into account internal structure of the classified objects. This approach allows to avoid a common problem in the evaluation of classifiers operating on compound objects, when the same components occur in different quantities in training and test set [11]. Our hypotheses are that: a) introducing indirectly 3D information in the multi-level classifier is beneficial for its performance, b) our evaluation schema reflects the real difficulty of protein-protein interaction prediction better than a näive approach which often overestimate classifier performance.

The decision to focus on prediction utilising protein primary structure and do not include methods based on protein functions [14, 13] in our comparison needs justification. We believe that those two types of prediction methods have different areas of application. First, functional features are generally available only for a subset of proteins from well studied organisms. Second, this kind of description is strongly dependent on biological pathways, which may be highly specific for a given organism. With functional features we are targeting high-level evolutionary designed mechanisms, while with sequence-based features we can hope to uncover basic physical properties of proteins, which govern their interactions. Knowing those properties it would be possible to predict protein interactions across different organisms and include some specific cases which distort normal protein interaction networks, such as host-patogen protein interactions.

In our experiments we obtained performance estimates much lower than usually reported, which is in line with our hypothesis. The multi-level approach was only marginally better than other methods. Our results raise the need to re-evaluate the usefulness of sequence-based features for protein interaction prediction. Lack of success of both standard methods aggregating global features of the sequence and the multi-level approach, which looks at the individual residues, suggests that protein interactions may be a phenomenon occurring primarily on a higher level and involving whole protein structures.

2 Materials and methods

2.1 Protein interactomes

We evaluated prediction methods by building classifiers separately for three organisms: *S. cerevisiae*, *H. sapiens*, and *E. coli*. Interactomes of all these organisms are relatively well studied, however reconstructed protein interaction networks are still far from being complete. For all three organisms we extracted 3D protein crystal complexes from Protein Data Bank (PDB) [2]. We were interested in complexes scanned with X-RAY with the resolution below 3 Å. Homologous structures were removed with 90% sequence identity threshold. The remaining complexes were used as a reliable source of information on residue-residue interactions (RRI) and protein-protein interactions (PPI). Residue-residue interaction is defined as a pair of amino acid residues from two different protein chains which are located within a close distance (4 Å) in the crystal structure. Protein-protein interaction is a pair of proteins for which at least one residue-residue interaction occurs. Only pairwise heterogenous protein interactions involving two different proteins were of interest to us.

In the work by Zubek et al. [17] a special procedure was used to filter RRIs and keep only the strongest interactions. The sliding window was moved along protein sequence and centred on each interacting residue. The window covered 21 residues – one central interacting residue, 10 residues to the left from it, and 10 residues to the right of it. Then the number of all interacting residues (including the central one) within the window was counted. Only when this number exceed certain threshold value the central residue was considered strongly interacting. We replicated this procedure in this work and set the threshold value to 15 (this value was reported as an optimal in the original publication).

Relatively small sets of PDB-derived PPIs were complemented with large scale data curated by Saha et al. [14]. They provided PPIs for *S. cerevisiae* and *H. sapiens* in two flavors: GOLD dataset contained only interactions which were confirmed independently with two different experimental methods, SILVER contained interactions reported by two different sources (possibly using the same experimental method). For *S. cerevisiae* and *H. sapiens* we used the available GOLD datasets. For *E. coli* we constructed our own SILVER dataset using iRefWeb interface [16].

PDB-based data sets were split into training and test set on the protein level (no protein occurred simultaneously in the two sets). Numbers of PPIs in each

set are given by Table 1. Proteins occurring in *S. cerevisiae* PDB training were removed from *S. cerevisiae* GOLD test, proteins occurring in *H. sapiens* PDB training were removed from *H. sapiens* GOLD test, and proteins occurring in *E. coli* PDB training were removed from *E. coli* SILVER. Because data available in PDB for *E. coli* was less abundant than for the other organisms, we did not construct a PDB-based test set, using *E. coli* SILVER as the only validation of prediction performance.

Table 1. Number of interacting protein pairs in the collected data sets.

Data set	Training RRI	Training PPI	Test PPI
S. cerevisiae GOLD	-	-	1284
S. cerevisiae PDB	5531	211	174
H. sapiens GOLD	-	-	1325
H. sapiens PDB	2774	195	204
E. coli SILVER	-	-	2763
E. coli PDB	1698	61	-

2.2 True negatives

The available experimental data identifies only positive interactions. True negative interactions for training machine learning classifier need to be artificially generated. Generating high quality negatives is generally very difficult. For RRIs we selected sequence fragments from known protein complexes such that not a single RRI occurred on those fragments. As the data was abundant and the risk of generating a false negative by chance was low, we generated 10 times more RRI negatives than the collected positives. This was done to represent class imbalance expected in real data.

The problem was more complicated for PPIs. Common methods for generating negatives include drawing random pairs of biomolecules from all known proteins found in a specific organism [14], or from the subset of whole proteome constituted by proteins occurring in positive examples [4]. We strongly believe that such methods have their inherent drawbacks, because they ignore network properties of the underlying protein interactome. Imagine that we have a set of 9 positive PPIs over 10 proteins which form a star subgraph in the interactome. The central protein in this subgraph has 9 interactions, the rest of the proteins have 1 interaction each. Then we generate negatives by drawing random pairs of these 10 proteins with equal probability. For each protein the probability of being included in a formed pair is equal to: $\frac{1}{10} + \frac{9}{10} \cdot \frac{1}{9} = 0.2$. If we draw 9 pairs, the expected number of negative interactions for each protein is 1.8. In such setting a classifier which recognises any pair containing the central protein

as positive automatically reaches 0.83 precision score, even though it completely ignores relative compatibility between two proteins. This scenario is biologically realistic, in real interactomes occurrence of a significant number of hub proteins is reported [15]. What is more, when the proteins are paired completely randomly there is always a risk of generating a false negative, i.e. previously undiscovered interaction. Because of this larger number negatives lower the quality of data.

As an alternative to uniform sampling we propose the following procedure:

- Let G_1 be a graph representing positive examples. Denote $V = v_1, \ldots, v_n$ as the set of its vertices. Each vertex in V represents a protein and each edge v_i, v_j represents an interaction. Let $[Deg(v_1), \ldots, Deg(v_n)]$ be a vector containing degrees of vertices from V. Let G_2 be a graph of negative interactions. At first it has vertices identical to G_1 and no edges.
- While there exist v such that $Deg(v) > 0$:
  1. Find vertex v with the largest $Deg(v)$.
  2. Find vertex u if exist such that:
     (a) There is no edge (v, u) in G_1.
     (b) u has as large $Deg(v)$ as possible.
     (c) Distance $d(u, v)$ in G_1 is as large as possible.
  3. If u exist:
     (a) Add edge (u, v) to G_2.
     (b) $Deg(v) \leftarrow Deg(v) - 1$
     (c) $Deg(u) \leftarrow Deg(u) - 1$
  4. else: $Deg(v) \leftarrow 0$

Such schema of constructing the negatives is unbiased, i.e. the protein composition of the positives and the negatives remains identical. Every single protein has the same number of positive and negative interactions. This forces the trained classifier to predict meaningful biophysical interactions rather than predicting general reactivity (the relative number of interactions) of a single protein. What is also important, our algorithm favours protein pairs which are remote to each other the interaction network, which reduces – but does not eliminate – the risk of generating false negatives by chance. Using the described procedure we generated the same number of negative PPIs as positive ones, thus obtaining a balanced dataset.

2.3 Multi-level prediction of protein-protein interactions

We were interested in benchmarking the multi-level method developed by Zubek et al. [17]. We will refer to it as MLPPI. It performs a two-stage prediction, first predicting RRIs and then using the results to predict PPIs. RRI classifier operates on sequence fragments of length 21 amino acid residues. Two fragments sliced from two proteins sequences constitute a single observation. Sliding window technique is used to extract fragments centred on each residue in a sequence. The result is a two dimensional matrix with dimensions corresponding to proteins' lengths. It can interpreted as a predicted potential contacts map.

This matrix is processed with various feature extraction algorithms to produce a fixed-length input for PPI classifier. General outline of this method is presented as Figure 1.

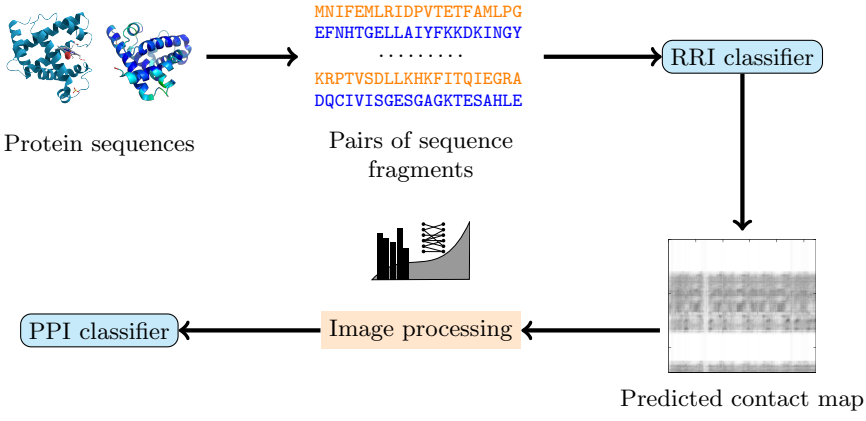


Fig. 1. Schematic depiction of the multi-level protein interaction prediction pipeline.

For classification on both levels we used Random Forest algorithm with 300 trees and maximum tree depth limited to 7 nodes. Sequence fragments which constituted input for RRI classifier were encoded using secondary structure symbols predicted from sequence by PSIPRED [8]. Features extracted from the predicted contact map to form an input for PPI classifier included:

- the mean and variance of values over the matrix (2),
- the sums of values in 10 best rows and 10 best columns (20),
- the sums of values in 5 best diagonals of the original and the transposed matrix (10),
- the sum of values on intersections of 10 best rows and 10 best columns (1),
- the histogram of scores distributed over 10 bins (10),
- features of the connection graph: fraction of nodes in the 3 largest connected components (3).

Features of the connection graph require further explanation. Predicted contacts between residues were represented as a bipartite graph. Nodes in the graph represented residues and edges represented predicted contact. To make the graph more consistent with the observed experimental data, for each node we left only 3 strongest outgoing edges. We set the value of this threshold (3) following the observation that in our PDB structures the mean number of interactions of a single interacting residue is between 2 and 3. In such trimmed graph we calculated fractions of nodes contained in 3 largest connected components. Those values were also appended to the feature vector.

2.4 Sequence-based methods

We compared our ensemble method with various sequence feature aggregation schemas that are commonly used to construct features for machine learning classifiers of protein interactions. To make the benchmarking results comparable between different algorithms, we used the same classification method (Random Forest) as for the MLPPI classifier. We benchmarked the following feature aggregation schemas:

1. AAC – Amino Acid Composition [10]. Feature set is the set of frequencies of all amino acids in the sequence.
2. PseAAC – Pseudo Amino Acid Composition [5]. Feature set consists of the standard AAC features with k-th tier correlation factors added. The k-th tier correlation factor represent correlation for residues separated from each other by k residues. We calculate those correlations on HQI8 indices.
3. 2-grams [10]. Feature set comprises of frequencies of all 400 ordered pairs of amino acids in the sequence.
4. QRC – Quasiresidue Couples [7]. A set of AAIndices is chosen. For each index d combined values of this property d for a given amino acid pair are summed up for all the pair's occurrences over the full protein sequence. Occurrences for pairs of residues separated from each other by $0, 1, 2 \ldots m$ residues. In effect, one obtains QRC^d vectors of length $400 \times m$. In this model we also use HQI8 indices.
5. VD – vector deviations, a variation of Liu's protein pair features [9]. The method starts from encoding each amino acid in a protein sequence with 7 chosen physicochemical properties, thus obtaining 7 feature vectors for each sequence. For each feature vector its "deviation" is calculated:

$$\gamma_{dj} = \frac{1}{n-d} \sum_{i=1}^{n-d} x_{ij} \times x_{(d+i),j} \quad j = 1, \ldots, 7 \quad d = 1, \ldots, L$$

where x_{ij} is the value of descriptor j for amino acid at position i in sequence P, n is the length of protein sequence P, and d is the distance between residues in the sequence. For the purpose of the comparison, we tested this method with the original 7 amino acid indices used by Liu. We tested different values of L from 5 to 30 in a quick cross-validation experiment on our data and chose $L = 9$ as yielding the best results.

2.5 Evaluation procedure

Created RRI training, PPI training, and PPI test data sets had their specific purposes. RRI training and PPI training data was used only to train RRI classifier in MLPPI. It was not used by any other method. Then, all sequence-based PPI classifiers and the PPI classifier of MLPII were trained and evaluated using PPI test data.

Performance of PPI classifiers was evaluated through a repeated 2-fold cross-validation (split between two folds of equal size). However, splitting data on

the level of individual observations was unsatisfactory because training and test sets could still overlap on the protein level, which introduced a huge bias into evaluation results (similar observation was previously made by [11]). To fix this problem we decided to perform split on the protein level. We used the following algorithm:

1. Let X be a set of all observations (protein pairs), P set of all proteins, X_A, X_B observations in the two splits, P_A, P_B protein in the two splits.
2. Initialise set $X_A \leftarrow \emptyset$ to empty set, $X_B \leftarrow X$.
3. While $|X_A| < |X_B|$ repeat:
   (a) Add a random observation x not included in X_A to X_A.
   (b) Complement X_A with all observations $x = (x^1, x^2)$ such that $x^1 \in P_A$ and $x^2 \in P_A$.
   (c) Let $X_B = \{(x^1, x^2) : x^1 \notin P_A \wedge x^2 \notin P_A\}$.

The relations between all datasets used in the evaluation procedure are depicted by Figure 2.

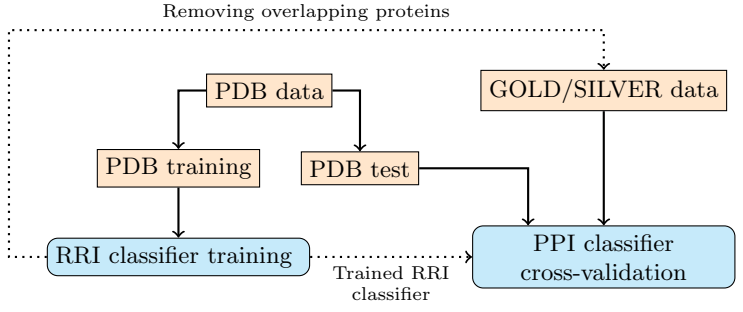


Fig. 2. Schematic depiction of relations between different data sets in the evaluation procedure.

The above described procedure differs from the standard cross-validation, since the number of observations in constructed test sets vary slightly, but this variance is small, and does not influence the estimated performance. Such evaluation schema does not allow for any information leak: the datasets are always balanced, and the classifier is tested on previously unseen proteins.

Using this form of cross-validation reduced the effective size of training and test data, because in each split some observations need to be dropped. Average size of a single cross-validation fold for all data sets is given in Table 2.

3 Results and discussion

We repeated cross-validation split 5 times and calculated average AUC ROC (area under the receiver operating characteristic curve) over splits and folds.

Table 2. Average size of a single cross-validation fold with estimated standard deviation.

Data set	CV fold size
S. cerevisiae GOLD	723 ± 7
S. cerevisiae PDB	116 ± 3
H. sapiens GOLD	763 ± 9
H. sapiens PDB	139 ± 3
E. coli SILVER	1591 ± 24

Results for different methods are presented as Table 3. As can be seen, AUC ROC values are generally very low, never exceeding 0.6. This means that under the conditions imposed by our strict evaluation procedure none of the methods was especially successful. This is especially true for *E. coli* SILVER data set for which the performance of all methods is at the level of a random baseline. Because of this we excluded *E. coli* data set from further analyses.

Table 3. AUC ROC (Area under the receiver operating characteristic curve) score for different methods. MLPPI – Multi-level Prediction of Protein Interactions, VD – vector deviations, AAC – amino acid composition, PseudoAAC – pseudo amino acid composition, 2-grams – bigram frequencies, QRC – quasiresidue couples.

Data set	MLPPI	VD	AAC	PseudoAAC	2-grams	QRC
E. coli SILVER	0.50	**0.51**	0.51	0.50	0.49	0.48
S. cerevisiae GOLD	**0.57**	0.56	0.55	0.54	0.51	0.52
S. cerevisiae PDB	**0.59**	0.52	0.52	0.54	0.47	0.47
H. sapiens GOLD	**0.56**	0.53	0.53	0.54	0.51	0.52
H. sapiens PDB	**0.56**	0.49	0.53	0.53	0.52	0.53

To establish statistical differences between methods we employed combined 5x2cv F test proposed by Alpaydin [1]. It is a modified version of 5x2cv t test introduced by Dietterich [6]. It strives to exploit the benefits of multiple train-test splits while minimising the bias introduced by lack of independence between splits. Each split i contains two cross-validation folds, which results in two values $p_i^{(1)}$ and $p_i^{(2)}$ which are the differences between scores obtained by two methods. They can be used to estimate mean and variance for each split separately:

$$\bar{p}_i = \frac{p_i^1 + p_i^2}{2}$$

$$s_i^2 = \left(p_i^{(1)} - \bar{p}\right)^2 + \left(p_i^{(2)} - \bar{p}\right)^2$$

The test statistic f has the following form:

$$f = \frac{\sum_{i=1}^{5} \sum_{j=1}^{2} \left(p_i^{(j)}\right)^2}{2 \sum_{i=1}^{5} s_i^2}$$

Under the null hypothesis, when two methods have identical performance, the f statistic is F distributed with 10 and 5 degrees of freedom.

Table 4. f statistic values and p-values for tests comparing MLPPI against the best performing sequence-based method.

Data set	Test	f	p-value
S. cerevisiae GOLD	MLPPI vs VD	1.637	0.250
S. cerevisiae PDB	MLPPI vs PseudoAAC	4.873	0.020
H. sapiens GOLD	MLPPI vs PseudoAAC	2.074	0.160
H. sapiens PDB	MLPPI vs AAC	2.604	0.097

We wanted to check whether our multi-level method performed better than methods aggregating global characteristics of protein sequence. On each data set separately we tested MLPPI against the best performing sequence-based method. Test statistic values and p-values are given by Table 4. Although MLPPI had the best AUC score on all four data sets, the difference was significant at $\alpha = 0.05$ level only for *S. cerevisiae* PDB – the data set on which MLPPI method was initially devised and calibrated.

The difference between *E. coli* and other data sets needs special attention. The performance of all predictors on *E. coli* was equal to a random baseline. The number of examples from PDB complexes was smaller than for the other organisms, while the number of examples from high-throughput experiments was larger, albeit of possibly lower quality (see Table 1). To assess whether the differences were also present in interaction network structure, we calculated mean node degree for PPI networks of the three organisms. For *H. sapiens* GOLD we obtained mean degree 2.01, for *S. cerevisiae* GOLD 2.05, and for *E. coli* SILVER 4.76. Such difference in numbers suggests a possibility that the data contained more gaps and false positives, making it impossible for a classifier to learn any relations. On the other hand, *E. coli* is the only prokaryotic organism among the three and its proteins may have different characteristics. Brocchieri and Karlin [3] showed that median protein length in prokaryotes is significantly smaller than in eukaryotes. They speculated that the difference may be due to eukaryotic proteins being composed of multiple functional units and additional sequence motifs acting as function regulators. This would definitely affect interaction landscape, however it is difficult to state in what way. Further research into this matter is needed before drawing conclusions.

Results obtained in our study were much lower than usually reported for methods concerning protein interaction prediction, even lower than in Park and Marcotte [11] suggesting that the performance may be routinely overestimated. Those differences are striking, for instance VD representation introduced by Liu [9] was reported to obtain 0.86 AUC ROC on large yeast proteins interactions data set. In the study of Nanni et al. [10] simple AAC representation achieved 0.72 AUC ROC on human PPI data set. Such differences were the result of a different evaluation strategies which, implicitly, led to different problem formulations. We believe that two conditions must be satisfied for an evaluation procedure to correctly represent the problem of predicting meaningful interactions between unknown proteins: a) proteins occurring in training and test sets must be strictly separated, b) protein composition of true positives and true negatives must be as close as possible (including characteristics such as node degree). To our knowledge, ours is the only procedure so far satisfying these conditions.

In the light of our strict evaluation schema and high-quality datasets the problem of predicting meaningful interactions between proteins occurs to be very hard, possibly even harder than generally expected. Success of simple sequence-based methods was limited and introduction of local structural information in our multi-level method yielded only minor and not statistically significant improvement. This raise a question as to how biological information regulating interactions is encoded? We know that protein sequences describe and identify proteins unambiguously, but is it sufficient to know proteins' sequences to fully characterise their behaviour? Our results suggests that the situation is more complex than that. While contacts between single residues of two different proteins occur only in interfaces, whole protein structures may be involved in mediating those interactions.

4 Conclusions

In this work we evaluated some sequenced-based approaches to protein interaction prediction. The main focus was put on the previously developed multi-level predictor (MLPPI). While MLPPI predictor was not worse than global sequence methods, obtained results are far from satisfactory. We believe that making a real breakthrough in protein-protein interaction prediction requires exploiting 3D structural information.

Further research is needed to develop evaluation strategies for multi-level biological input data and fully understand their properties. As our work demonstrates, the impact of evaluation procedure on the results is never overemphasized. We showed that unbalanced train-test splits may be the source of false results in previously published works. We believe that methodological unification and futher discussion is needed for the development of the field.

Acknowledgements

Study was financed by research fellowship within Project „Information technologies: Research and their interdisciplinary applications" (agreement number UDA POKL.04.01.01-00-051/10-00); Polish National Science Centre (grant number UMO 2013/09/B/NZ2/00121 and 2014/15/B/ST6/05082); COST BM1405 and BM1408 EU actions. Research were partially performed using the computational resources of Interdisciplinary Centre of Mathematical and Computational Modelling (ICM), University of Warsaw.

References

[1] Alpaydin, E.: Combined 5 × 2 cv F Test for Comparing Supervised Classification Learning Algorithms. Neural Computation 11(8), 1885–1892 (1999), http://dx.doi.org/10.1162/089976699300016007

[2] Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E.: The Protein Data Bank. Nucleic Acids Research 28(1), 235–242 (Jan 2000), http://nar.oxfordjournals.org/content/28/1/235

[3] Brocchieri, L., Karlin, S.: Protein length in eukaryotic and prokaryotic proteomes. Nucleic Acids Research 33(10), 3390–3400 (2005), http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1150220/

[4] Chang, D.T., Syu, Y.T., Lin, P.C.: Predicting the protein-protein interactions using primary structures with predicted protein surface. BMC Bioinformatics 11(Suppl 1), S3 (Jan 2010), http://www.biomedcentral.com/1471-2105/11/S1/S3/abstract

[5] Chou, K.C.: Prediction of protein cellular attributes using pseudo-amino acid composition. Proteins 43(3), 246–255 (May 2001)

[6] Dietterich, T.G.: Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. Neural Comput. 10(7), 1895–1923 (1998), http://dx.doi.org/10.1162/089976698300017197

[7] Guo, J., Lin, Y.: A novel method for protein subcellular localization: Combining residue-couple model and SVM. In: Proceedings of the 3rd Asia-Pacific Bioinformatics Conference. pp. 117–129. Singapore (2005)

[8] Jones, D.T.: Protein secondary structure prediction based on position-specific scoring matrices. Journal of Molecular Biology 292(2), 195–202 (1999), http://www.sciencedirect.com/science/article/pii/S0022283699930917

[9] Liu, H.w.: Protein-Protein Interaction Detection by SVM from Sequence. In: The Third International Symposium on Optimization and Systems Biology. pp. 198–206 (2009)

[10] Nanni, L., Lumini, A., Brahnam, S.: An Empirical Study of Different Approaches for Protein Classification. The Scientific World Journal 2014, e236717 (Jun 2014), http://www.hindawi.com/journals/tswj/2014/236717/abs/

[11] Park, Y., Marcotte, E.M.: Flaws in evaluation schemes for pair-input computational predictions. Nature Methods 9(12), 1134–1136 (2012), http://www.nature.com/nmeth/journal/v9/n12/full/nmeth.2259.html

[12] Rual, J.F., Venkatesan, K., Hao, T., Hirozane-Kishikawa, T., Dricot, A., Li, N., Berriz, G.F., Gibbons, F.D., Dreze, M., Ayivi-Guedehoussou, N., Klitgord, N., Simon, C., Boxem, M., Milstein, S., Rosenberg, J., Goldberg, D.S., Zhang, L.V., Wong, S.L., Franklin, G., Li, S., Albala, J.S., Lim, J., Fraughton, C., Llamosas, E., Cevik, S., Bex, C., Lamesch, P., Sikorski, R.S., Vandenhaute, J., Zoghbi, H.Y., Smolyar, A., Bosak, S., Sequerra, R., Doucette-Stamm, L., Cusick, M.E., Hill, D.E., Roth, F.P., Vidal, M.: Towards a proteome-scale map of the human protein–protein interaction network. Nature 437(7062), 1173–1178 (2005), http://www.nature.com/nature/journal/v437/n7062/full/nature04209.html

[13] Saccà, C., Teso, S., Diligenti, M., Passerini, A.: Improved multi-level protein-protein interaction prediction with semantic-based regularization. BMC bioinformatics 15(1), 103 (2014), http://www.biomedcentral.com/1471-2105/15/103/

[14] Saha, I., Zubek, J., Klingström, T., Forsberg, S., Wikander, J., Kierczak, M., Maulik, U., Plewczynski, D.: Ensemble learning prediction of protein–protein interactions using proteins functional annotations. Molecular BioSystems 10(4), 820–830 (Mar 2014), http://pubs.rsc.org/en/content/articlelanding/2014/mb/c3mb70486f

[15] Song, J., Singh, M.: From Hub Proteins to Hub Modules: The Relationship Between Essentiality and Centrality in the Yeast Interactome at Different Scales of Organization. PLoS Comput Biol 9(2), e1002910 (2013), http://dx.doi.org/10.1371/journal.pcbi.1002910

[16] Turner, B., Razick, S., Turinsky, A.L., Vlasblom, J., Crowdy, E.K., Cho, E., Morrison, K., Donaldson, I.M., Wodak, S.J.: iRefWeb: interactive analysis of consolidated protein interaction data and their supporting evidence. Database: The Journal of Biological Databases and Curation 2010, baq023 (2010)

[17] Zubek, J., Tatjewski, M., Boniecki, A., Mnich, M., Basu, S., Plewczynski, D.: Multi-level machine learning prediction of protein–protein interactions in Saccharomyces cerevisiae. PeerJ 3, e1041 (Jul 2015), https://peerj.com/articles/1041

Geometric Approach to Stepwise Regression

Barbara Żogała-Siudem[1] and Szymon Jaroszewicz[2,3]

[1] Systems Research Institute, Polish Academy of Sciences,
ul. Newelska 6, 01-447 Warsaw, Poland
[2] Institute of Computer Science, Polish Academy of Sciences,
ul. Jana Kazimierza 5, 01-248 Warsaw, Poland
[3] National Institute of Telecommunications,
ul. Szachowa 1, 04-894 Warsaw, Poland

Abstract. Stepwise feature selection is one of the most popular variable selection techniques for linear models. The procedure, however, is computationally demanding, especially when the number of potential variables is large. In our previous work we proposed a way to speed up stepwise algorithm on large data, based on multidimensional indices and a bound based on correlations between variables. This paper presents an alternative proof of the bound and shows that it cannot be improved.

1 Introduction

Nowadays, many sophisticated methods for data analysis are available. However, a very important issue is not only the modeling itself, but also finding relevant variables to include in the model. Unfortunately, there are currently no methods to assist the researcher (except his/her own intuition) in finding external sources of relevant data such as public datasets available on the web.

Potentially, an answer to this problem could be Linked Open Data (LOD): a project to make statistical data collected by various organizations, government statistical offices, etc. publicly available on the internet in a way which is well suited for automated access. The movement has recently gained momentum and huge amounts of data became available online from sources such as Eurostat [1], United Nations, International Monetary Fund, etc. The current state of Linked Open Data can be seen in the diagram [2] which shows data sources and links between them. More information on Linked Open Data can be found e.g. in [3–6].

We believe however, that in its current form Linked Open Data is not suitable for statistical practice. Linking new datasets is based on purely syntactic criteria, which can easily result in huge amount of unrelated data being downloaded to researcher's computer. Building models on such data would then be extremely time consuming and prone to overfitting.

A solution, in our opinion, is a linking procedure based on statistical, not syntactic properties. One example of such a solution (and at the same time the

most relevant previous work) is Google Correlate [7–9], which given a *query dataset* finds the most correlated Google query. The service has several limitations: it is restricted to finding correlated Google queries and does not include other publicly available datasets. Moreover it is only able to find single most correlated variables, while in practice we are interested in building complete statistical models.

In [10] we presented a method to build fast stepwise linear regression models by using a multidimensional indexes to search for relevant variables. As the multidimensional index we used FLANN (Fast Library for Approximate Nearest Neighbors) [11, 12]. Since the indices only allow for finding single most correlated variables, the stepwise procedure had to be rewritten using only this operation. The method is based on forward stepwise regression (see e.g. [13]) but during each step a spatial index is used to search for candidate variables; only those candidates are then used for classical stepwise selection.

The paper is organized as follows. In Section 2 we give a brief summary of [10], introduce the necessary notation, followed by a short introduction to multidimensional indexing. Forward stepwise feature selection is explained in Section 2.2 together with the theorem guaranteeing its correctness. Later, in Section 3 a geometric proof of Theorem 1 is described, and a theorem is given proving that the bound cannot be improved. Finally, in Section 4 we conclude the paper.

2 Fast stepwise regression

The main idea of our approach to speed up the stepwise regression procedure is based on Theorem 1, which was proved and discussed shortly in [10]. To present this theorem let us start with introducing some notation and explaining the stepwise regression procedure.

2.1 Notation

Lowercase letters will denote n-dimensional vectors. In particular, $y \in \mathbb{R}^n$ will be the response variable of a linear model, $r \in \mathbb{R}^n$ a residual vector of the currently considered model, and $x \in \mathbb{R}^n$ a predictor variable. The set of all possible predictors will be denoted as $X = \{x_1, \ldots, x_p\}$. Subsets of X will be denoted as X_I, where $I \subseteq \{1, \ldots, p\}$ is the set of indices of variables. So, if $I = \{l_1, \ldots, l_k\}$, then $X_I = \{x_{l_1}, \ldots, x_{l_k}\}$.

In the paper we assume that each vector $x_i \in X$ as well as the response y are normalized i.e. they have zero mean ($\bar{x}_i = 0$) and l_2 norm equal to 1 ($\|x_i\| = 1$).

Let $I = \{l_1, \ldots, l_k\}$. The projection of a vector y onto the space spanned by $X_I = \{x_{l_1}, \ldots, x_{l_n}\}$ will be denoted as $Proj_{X_I} y$ and by $y \sim x_{l_1} + \ldots + x_{l_k}$ or $y \sim X_I$ we will denote a linear model with y as response and $x_{l_1}, \ldots, x_{l_k}$ as predictors.

For brevity, correlations of specific vectors x_i and x_j will be written as $c_{i,j} = \mathrm{cor}(x_i, x_j)$ and correlation of the variable x_i and current residual vector r as $c_{res,i} = \mathrm{cor}(r, x_i)$.

In Section 3 volumes of parallelotopes spanned by a set of vectors will be denoted as $\mu_k(\cdot)$. For example, the volume of a k-dimensional parallelotope spanned by $\{x_{l_1}, \ldots, x_{l_k}\}$ is $\mu_k(x_{l_1}, \ldots, x_{l_k})$.

2.2 Stepwise regression

The idea of stepwise regression was introduced in 1960 by Efroymson [14]. Here, by stepwise procedure we mean *forward stepwise selection* (see e.g. [13]). The algorithm works as follows. First we start with an empty model ($y \sim 1$) and find a variable (say x_{l_1}) which gives the lowest residual sum of squares (RSS) when added to the model. The variable is then included in the model which becomes: $y \sim x_{l_1}$. Then we check all two-variable models which include the variable x_{l_1}, that is $y \sim x_{l_1} + x_i$, for all $x_i \in X \setminus \{x_{l_1}\}$, select a variable x_{l_2} for which the RSS was lowest and add it to the model. We continue this procedure until the model no longer improves according to an appropriate criterion (such as AIC [15] or BIC [16]) or the maximum number of variables allowed in the model is reached. The algorithm is presented in Table 1.

Algorithm: Stepwise

1) $r := y$
   $I := \emptyset$
2) **For** $k = 1, \ldots, k_{max}$:
   1. **For** each $i \in \{1, \ldots, p\} \setminus I$:
      compute the residual of the model obtained
      by adding x_i to the current model: $r_i = y - Proj_{X_{I \cup \{i\}}} y$
   2. Find $l_k = \arg\min_{i \in \{1,\ldots,p\} \setminus I} {r_i}^T r_i$,
   3. **If** the model: $y \sim X_{I \cup \{l_k\}}$ is *better* than $y \sim X_I$:
      Add l_k to I: $I := I \cup \{l_k\}$ and **goto** 2)
      else break.

Fig. 1. The stepwise regression algorithm

The main problem with the stepwise algorithm is that in each iteration it requires building as many models as there are possible predictors (although some work can be shared between all models in some circumstances) and, as a result, becomes very inefficient for datasets with a large number of variables, such as the ones that may be obtained using Linked Open Data.

2.3 Multidimensional indices and correlations

To speed up the stepwise procedure described in Section 2.2 we proposed [10] an algorithm which limits the number of models built in each iteration, by using multidimensional indexing. We will now summarize the results of that paper.

A multidimensional index can be used to store a large number of points from an n-dimensional Euclidean space. Afterwards, we can use the index to quickly

answer two types of queries: (1) *nearest neighbor queries*, where given a query vector, find k nearest vectors in the index, and (2) *range queries*, where given a query vector and a radius, find all points within the given radius from the query. As a multidimensional index we used the FLANN library [11] which is very fast but gives approximate results and Ball Trees (see e.g. [17]) which are much slower, but give exact results.

A key observation is that, for appropriately normalized vectors, searching for a nearest neighbor corresponds to looking for the most correlated vector. Let x_i, x_j be vectors with zero mean and l_2 norm equal to 1 (i.e. $\bar{x}_i = \bar{x}_j = 0$ and $\|x_i\| = \|x_j\| = 1$), then

$$\|x_i - x_j\| = \sqrt{2 - 2\langle x_i, x_j\rangle} = \sqrt{2 - 2\mathrm{cor}(x_i, x_j)}.$$

Due to the above, in order to search for a vector most correlated with a given query vector x we need to normalize it

$$x' = \frac{x - \bar{x}}{\|x - \bar{x}\|}, \tag{1}$$

and perform a nearest neighbor search for both x' and $-x'$.

2.4 Fast stepwise regression

The main result of the paper [10] was to show how to quickly build a stepwise model on data with a large number of indexed variables. Here we restate this result briefly, starting with the following theorem.

Theorem 1. *Assume that the variables $x_{l_1}, \ldots, x_{l_{k-1}}$ currently in the model are orthogonal, let $r = y - Proj_{\{x_{l_1},\ldots,x_{l_{k-1}}\}} y$ denote the residual vector of the current model and take two variables x_{l_k}, $x_{l'_k}$. Then*

$$\|y - Proj_{\{x_{l_1},\ldots,x_{l_{k-1}},x_{l'_k}\}} y\| \leqslant \|y - Proj_{\{x_{l_1},\ldots,x_{l_{k-1}},x_{l_k}\}} y\| \tag{2}$$

implies

$$\max\{|c_{l_1,l'_k}|, \ldots, |c_{l_{k-1},l'_k}|, |c_{res,l'_k}|\} \geqslant \frac{|c_{res,l_k}|}{\sqrt{1 - \sum_{i=1}^{k-1} c^2_{l_i,l_k} + (k-1)c^2_{res,l_k}}}. \tag{3}$$

Suppose we considered x_{l_k} as a candidate for the model and computed the residual sum of squares for it. Theorem 1 states that if any variable is better than x_{l_k}, then it must be correlated to a sufficient degree either with the current residual vector or one of the predictors already in the model. This condition can easily be translated into a series of range queries to the index. The query points are $\pm r$, where r is the current residual and $\pm x_{l_i}$, where x_{l_i} are variables currently in the model. The query radius is given by the right hand side of (3).

The algorithm for fast stepwise regression is given in Figure 2. Lines 2 and 3.2 use a nearest neighbor query to the index, and lines 3.5 and 3.6 use range queries. The speed up comes from using the stepwise procedure only on the candidate set C which can be efficiently obtained using the multidimensional index.

Algorithm: Fast stepwise

1) $r := y$
   $I := \emptyset$
2) Find a variable $x_{l_1} \in X$ most correlated with r
   and add its index l_1 to the active index set $I := I \cup \{l_1\}$
3) **For** $k = 1, \ldots, k_{max}$:
   1. Compute the new residual vector $r = y - Proj_{X_I} y$
   2. Find a candidate variable index $u_k \in \{1, \ldots, p\} \setminus I$
      such that x_{u_k} is most correlated with r
   3. Initialize the candidate index set $C := \{u_k\}$
   4. $\eta := \frac{|\mathrm{cor}(r, x_{u_k})|}{\sqrt{1 - \sum_{l_i \in I} \mathrm{cor}(x_{l_i}, x_{u_k})^2 + |I| \mathrm{cor}(r, x_{u_k})^2}}$
   5. $C := C \cup \{i \in \{1, \ldots, p\} \setminus I : \|x_i - r\|^2 \leqslant 2 - 2\eta\}$
   6. **For** $j = 1, \ldots, k-1$:
      $C := C \cup \{i \in \{1, \ldots, p\} \setminus I : \|x_i - x_{l_j}\|^2 \leqslant 2 - 2\eta\}$
   7. Find the best variable x_{l_k} in X_C using an iteration
      of stepwise procedure
   8. Add l_k to the current active index set: $I := I \cup \{l_k\}$

Fig. 2. The fast stepwise regression algorithm based on a multidimensional index.

3 Geometric approach

In [10] a proof of Theorem 3 was given, which was based on linear algebra techniques. In this paper we would like to show a different approach concentrating on a geometric structure of variables and correlations between them. The geometric proof is based on the following lemma.

Lemma 1. *If adding the variable $x_{l'_k}$ to the model decreases the residual sum of squares more than adding x_{l_k}, i.e.*

$$\|y - Proj_{\{x_{l_1}, \ldots, x_{l'_k}\}} y\| \leqslant \|y - Proj_{\{x_{l_1}, \ldots, x_{l_k}\}} y\|, \tag{4}$$

then the following inequality is satisfied

$$\frac{c^2_{res, l_k}}{1 - \sum_{i=1}^{k-1} c^2_{l_i, l_k}} \leqslant \frac{c^2_{res, l'_k}}{1 - \sum_{i=1}^{k-1} c^2_{l_i, l'_k}}.$$

To prove Lemma 1 let us first state two facts:

FACT 31
For any $x_1, \ldots, x_n \in \mathbb{R}^n$ and matrix $X = [x_1|\ldots|x_n]$ we can find a rotation matrix R such that RX is upper triangular.

FACT 32
The volume of a parallelotope spanned by vectors $x_1, \ldots, x_n \in \mathbb{R}^n$ is equal to

$$\mu_n(x_1, \ldots, x_n) = \det([x_1|\ldots|x_n]).$$

Let us now back to the proof of lemma 1.

Proof (Proof of lemma 1). First let us notice that

$$\|y - Proj_{\{x_{l_1},\ldots,x_{l_k}\}}y\| = \frac{\mu_{k+1}(r, x_{l_1}, \ldots, x_{l_k})}{\mu_k(x_{l_1}, \ldots, x_{l_k})}. \tag{5}$$

Due to Fact 31 and the fact that without loss of generality we may assume the vectors $x_{l_1}, \ldots, x_{l_{k-1}}$ already added to the model to be orthogonal, we can rotate matrices $[r|x_{l_1}|\ldots|x_{l_k}]$ and $[x_{l_1}|\ldots|x_{l_k}]$ such that matrices M_1 and M_2 are obtained with respectively only $k+1$ and k nonzero rows.

$$M_1 = \begin{pmatrix} z & 0 & \cdots & 0 & \frac{c_{res,l_k}}{z} \\ 0 & 1 & \cdots & 0 & c_{l_1,l_k} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & c_{l_{k-1},l_k} \\ 0 & 0 & \cdots & 0 & \sqrt{1 - \frac{c^2_{res,l_k}}{z^2} - \sum_{i=1}^{k-1} c^2_{l_i,l_k}} \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}, M_2 = \begin{pmatrix} 1 & \cdots & 0 & c_{l_1,l_k} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & c_{l_{k-1},l_k} \\ 0 & \cdots & 0 & \sqrt{1 - \sum_{i=1}^{k-1} c^2_{l_i,l_k}} \\ 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix},$$

where $z = \|r\|$. Due to Fact 32 the volumes in Equation 5 can be calculated as follows

$$\mu_{k+1}(z, x_{l_1}, \ldots, x_{l_k}) = \det M_1 = \sqrt{z^2(1 - \sum_{i=1}^{k-1} c^2_{l_i,l_k}) - c^2_{res,l_k}},$$

$$\mu_k(x_{l_1}, \ldots, x_{l_k}) = \det M_2 = \sqrt{1 - \sum_{i=1}^{k-1} c^2_{l_i,l_k}}.$$

And then equation (5) can be written as

$$\|y - Proj_{\{x_{l_1},\ldots,x_{l_k}\}}y\| = \frac{z^2(1 - \sum_{i=1}^{k-1} c^2_{l_i,l_k}) - c^2_{res,l_k}}{1 - \sum_{i=1}^{k-1} c^2_{l_i,l_k}},$$

which combined with (4) leads to

$$\frac{c^2_{res,l_k}}{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k}} \leqslant \frac{c^2_{res,l'_k}}{1-\sum_{i=1}^{k-1} c^2_{l_i,l'_k}}.$$

As we can see the above proof is based on the geometric structure of the variables of a linear model. The rest of the proof of theorem 1 is the same as in [10]. We restate it below for the sake of completeness.

Proof (Proof of Theorem 1). If for any $i = 1, \ldots, k-1$:

$$|c_{l_i,l'_k}| \geqslant \frac{|c_{res,l_k}|}{\sqrt{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k} + (k-1)c^2_{res,l_k}}}$$

then the inequality is true. Otherwise for all $i = 1, \ldots, k-1$:

$$|c_{l_i,l'_k}| < \frac{|c_{res,l_k}|}{\sqrt{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k} + (k-1)c^2_{res,l_k}}} \tag{6}$$

and we need to show that this implies $|c_{res,l'_k}| \geqslant \frac{|c_{res,l_k}|}{\sqrt{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k}+(k-1)c^2_{res,l_k}}}$. Notice first that the inequalities (6) imply

$$1-\sum_{i=1}^{k-1} c^2_{l_i,l'_k} > \frac{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k}}{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k} + (k-1)c^2_{res,l_k}}. \tag{7}$$

Using inequality (7) and Lemma 1 we get the desired result:

$$c^2_{res,l'_k} \geqslant c^2_{res,l_k} \frac{1-\sum_{i=1}^{k-1} c^2_{l_i,l'_k}}{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k}} > \frac{c^2_{res,l_k}}{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k} + (k-1)c^2_{res,l_k}}.$$

3.1 Optimality of the constraint

We will now show that the inequality (3) in Theorem 1 cannot be improved. This is illustrated graphically in Figures 3 and 4 and proved in Theorem 2.

Figures 3 and 4 present results on simulated data illustrating Theorem 1. Each point in each figure corresponds to a single simulation run, where random vectors were drawn, normalized and values of both sides of the bound calculated.

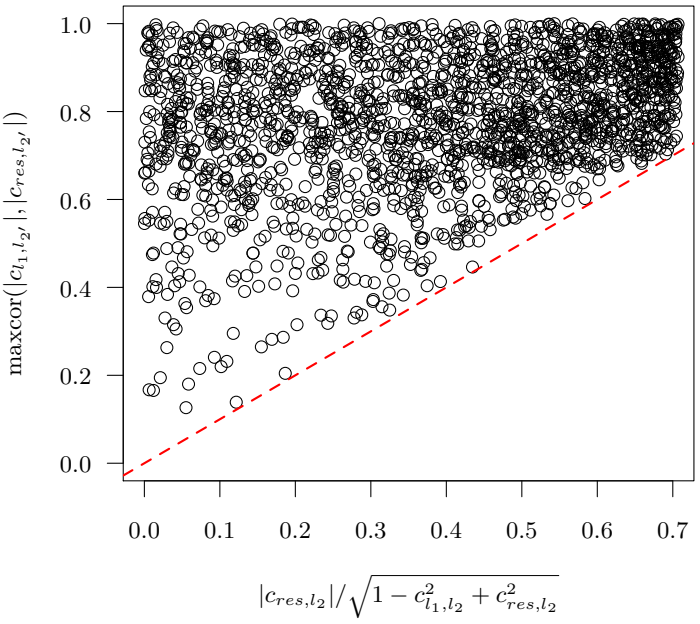


Fig. 3. Illustration of theorem 1 for adding 2nd variable for $n = 4$. Presented points correspond to vectors satisfying theorem assumptions and dashed red line is identity.

The first simulation (Figure 3) was performed as follows. The first predictor x_{l_1} was sampled as a normally distributed vector of a given length $n = 4$, then the response variable y was built as a sum of the vector x_{l_1} and some normally distributed noise. Two other vectors were then sampled similarly to x_{l_1}. The better of them (in the sense of lower RSS) was chosen as $x_{l_2'}$, the worse as x_{l_2} and the values $\max\{|c_{l_1,l_2'}|, |c_{res,l_2'}|\}$ and $|c_{res,l_2}|/\sqrt{1-c^2_{l_1,l_2}+c^2_{res,l_2}}$ were calculated. Then results were plotted as a scatterplot. The red dashed line corresponds to identity, so all vectors for which the inequality $\max\{|c_{l_1,l_2'}|, |c_{res,l_2'}|\} \geqslant |c_{res,l_2}|/\sqrt{1-c^2_{l_1,l_2}+c^2_{res,l_2}}$ is satisfied lie above that line. As we can see, vectors tend to get arbitrarily close to the line, suggesting that the inequality is tight.

The second simulation (Figure 4) is very similar, but instead of adding the second variable we add the third one. Moreover $n = 5$ was chosen. First, two variables x_{l_1} and x_{l_2} ware sampled and orthogonalized, then y was calculated as the sum of x_{l_1}, x_{l_2} and a normally distributed noise. Then two more variables were sampled, and the better one was used as $x_{l_3'}$ and the worse as x_{l_3}. Again,

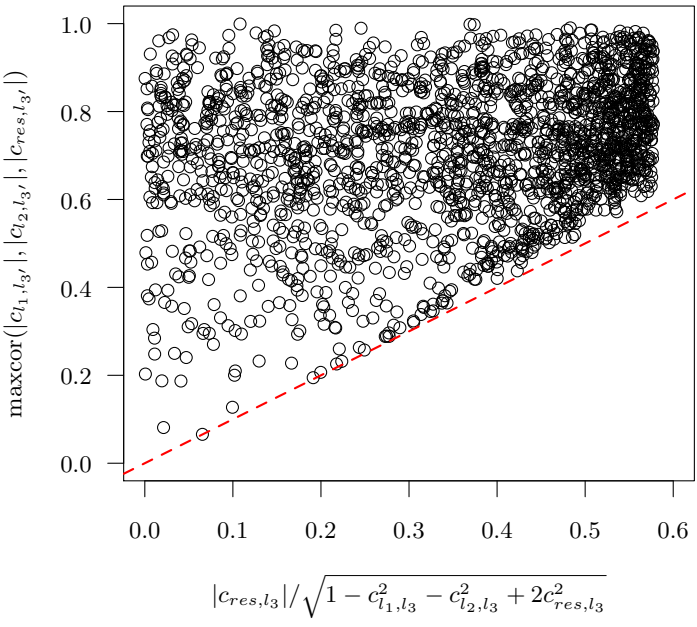


Fig. 4. Illustration of theorem 1 for adding 3rd variable for $n = 5$. Presented points correspond to vectors satisfying theorem assumptions and dashed red line is identity.

the values $\max\{|c_{l_1,l_3'}|, |c_{l_2,l_3'}|, |c_{res,l_3'}|\}$ and $|c_{res,l_3}|/\sqrt{1-c^2_{l_1,l_2}-c^2_{l_1,l_3}+c^2_{res,l_3}}$ were calculated and plotted in the figure. Again, points get arbitrarily close to the line, suggesting tightness of the bound. The theorem below proves that this is indeed the case.

Theorem 2. *The inequality*

$$\max\{|c_{l_1,l'_k}|,\ldots,|c_{l_{k-1},l'_k}|,|c_{res,l'_k}|\} \geqslant \frac{|c_{res,l_k}|}{\sqrt{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k}+(k-1)c^2_{res,l_k}}}$$

form theorem 1 cannot be improved.

Proof. To prove the theorem it is enough to find vectors x_{l_k} and $x_{l'_k}$ such that

$$\max\{|c_{l_1,l'_k}|,\ldots,|c_{l_{k-1},l'_k}|,|c_{res,l'_k}|\} = \frac{|c_{res,l_k}|}{\sqrt{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k}+(k-1)c^2_{res,l_k}}}$$

Let $x_{l_k} = \frac{1}{\sqrt{k}}\frac{r}{\|r\|} + \frac{1}{\sqrt{k}}x_{l_1} + \ldots + \frac{1}{\sqrt{k}}x_{l_{k-1}}$ and $x_{l'_k} = x_{l_k}$, then x_{l_k} is properly normalized ($\bar{x}_{l_k} = 0$, $\|x_{l_k}\| = 1$). Due to the fact that $r, x_{l_1}, \ldots, x_{l_k}$ are uncorrelated, the following correlations are equal to

$$c_{res,l_k} = c_{res,l'_k} = \frac{1}{\sqrt{k}},$$
$$c_{l_i,l_k} = c_{l_i,l'_k} = \frac{1}{\sqrt{k}},$$

thus

$$\max\{|c_{l_1,l'_k}|,\ldots,|c_{l_{k-1},l'_k}|,|c_{res,l'_k}|\} = \frac{1}{\sqrt{k}} = \frac{|c_{res,l_k}|}{\sqrt{1-\sum_{i=1}^{k-1} c^2_{l_i,l_k}+(k-1)c^2_{res,l_k}}}.$$

4 Conclusions

The paper presents an alternative, geometric proof of theorem enabling finding stepwise regression model faster on large data sets, presented in paper [10]. It also shows that the bound in this theorem cannot be improved. Paper discusses stepwise regression with no penalties, which is left for the future research.

5 Acknowledgements

The paper is co-founded by the European Union from resources of the European Social Fund. Project PO KL „Information technologies: Research and their interdisciplinary applications", Agreement UDA-POKL.04.01.01-00-051/10-00.

References

1. http://ec.europa.eu/eurostat
2. Schmachtenberg, M., Bizer, C., Jentzsch, A., Cyganiak, R.: Linking open data cloud diagram. http://lod-cloud.net/
3. http://stack.lod2.eu
4. http://linkeddata.org/
5. Heath, T., C., B.: Linked Data: Evolving the Web into a Global Data Space. 1 edn. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool (2011)
6. Bizer, C., Heath, T., Berners-Lee, T.: Linked data - the story so far. International Journal on Semantic Web and Information Systems (IJSWIS) **5**(3) (2009) 1–22
7. Mohebbi, M., Vanderkam, D., Kodysh, J., Schonberger, R., Choi, H., Kumar, S.: Google correlate whitepaper. (2011)
8. http://www.google.com/trends/correlate
9. Vanderkam, D., Schonberger, R., Rowley, H., Kumar, S.: Technical report: Nearest neighbor search in google correlate. (2013)
10. Zogala-Siudem, B., Jaroszewicz, S.: Fast stepwise regression on linked data
11. Muja, M., Lowe, D.: FLANN - Fast Library for Approximate Nearest Neighbors. (2013)
12. Muja, M., Lowe, D.G.: Scalable nearest neighbor algorithms for high dimensional data. IEEE Trans. on Pattern Analysis and Machine Intelligence **36** (2014)
13. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer-Verlag (2009)
14. Efroymson, M.A.: Multiple Regression Analysis. Wiley (1960)
15. Akaike, H.: A new look at the statistical model identification. IEEE Transactions on Automatic Control **AC-19**(6) (1974) 716–723
16. Schwarz, G.: Estimating the dimension of a model. The Annals of Statistics **6**(2) (1978) 461–464
17. Kibriya, A.M., Frank, E.: An empirical comparison of exact nearest neighbour algorithms. In: PKDD, Springer (2007) 140–151

ITRIA 2015 |

COMPUTATIONAL METHODS IN DATA ANALYSIS

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

The Project is co-financed by the European Union from resources of the European Social Fund

ISBN 978-83-63159-22-1