16th World Congress of the International Fuzzy Systems Association (IFSA)
9th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT)

Learning experts' preferences from informetric data

Marek Gagolewski[1,2] **Jan Lasek**[3]

[1] Systems Research Institute, Polish Academy of Sciences,
ul. Newelska 6, 01-447 Warsaw, Poland, gagolews@ibspan.waw.pl
[2] Faculty of Mathematics and Information Science, Warsaw University of Technology,
ul. Koszykowa 75, 00-662 Warsaw, Poland
[3] Interdisciplinary PhD Studies Program, Institute of Computer Science,
Polish Academy of Sciences, j.lasek@phd.ipipan.waw.pl

Abstract

In the field of informetrics, agents are often represented by numeric sequences of non necessarily conforming lengths. There are numerous aggregation techniques of such sequences, e.g., the g-index, the h-index, that may be used to compare the output of pairs of agents. In this paper we address a question whether such impact indices may be used to model experts' preferences accurately.

Keywords: preference learning, fuzzy relations, informetrics, aggregation, h-index

1. Introduction

Nowadays massive amounts of data are produced causing their users to suffer from so-called information overload. There is a need to accurately measure and retrieve relevant content using tools carefully tailored for this purpose. This is one of the goals in a field of informetrics that deals with measurable aspects of information processes.

In this paper we employ different tools for evaluation of so-called producers to model experts' preferences presented in questionnaire data which is an exemplary instance of the *Producer Assessment Problem* (PAP), cf. e.g. [1], which we shall now define. Let a set of k producers be given, $P = \{p_1, \ldots, p_k\}$. Each of the producers, say, ith, outputs certain finite number of products n_i. Further on, each of the products is given some qualitative rating, x_j for the jth product. Hence we may associate a producer with its output vector $\mathbf{x}^{(i)} = \left(x_1^{(i)}, \ldots, x_{n_i}^{(i)}\right) \in \mathbb{I}^{1,2,\cdots} = \bigcup_{n\geq 1} \mathbb{I}^n$ with elements in $\mathbb{I} = [0, \infty)$. We note that the number of supplied products may vary from producer to producer. The goal of PAP is to design tools for evaluation of their outputs or their ranking with respect to the quality of products they supply as well as their productivity.

The main goal of this study is to validate the hypothesis that certain standard impact functions may efficiently be used to compress (project) information included in an exemplary instance of PAP. To this end, we employ different prediction models which are estimated via two different approaches. Firstly, the prediction models are directly trained on raw inputs. The problem with such an approach is that the data has many variables, which are additionally highly correlated. Secondly, we extract certain features from output vectors and basing on these features we train several prediction models. We also present insights from an online survey data.

The paper is structured as follows. In Section 2 we present how the data for an online survey was generated and evaluated by the participants. In Section 3 we discuss certain properties of the gathered dataset emphasising the difference between questionnaire participants' responses. In Section 4 we proceed with the validation of the main hypothesis of our study – we verify the relevance of certain tools. Finally, Section 5 concludes the work.

2. Online questionnaire design

In this section we present the design of our online survey in an exemplary instance of PAP. The data collected via the questionnaire are the basis for our analysis in further parts of the paper. Our goal is to study the individual preferences of the participants (which we refer as to *experts*).

2.1. Data generating process

To create the dataset for the analysis we generated artificial data for PAP. A set of exemplary output vectors $\mathbf{x}^{(i)}$, $i = 1, 2, \ldots, 200$ were sampled according to the following algorithm:

- generate the number of products (the length n_i of the vector $\mathbf{x}^{(i)}$) from the Poisson distribution with expectation of 20, increased by 1 so as to assure that the generated number is positive:
$$n_i \sim Poisson(20) + 1$$
- generate the jth coordinate of a vector $\mathbf{x}^{(i)}$ independently from a truncated Pareto distribution with scale parameter equal to 1 and shape parameter α drawn from the uniform distribution on interval $[1, 2]$:
$$\mathbf{x}_j^{(i)} \sim \max\{\lfloor Pareto(1, \alpha) \rfloor, 50\}$$
with
$$\alpha \sim U([1, 2]).$$

The probability density function of a Pareto distribution is given by

$$f(y) = \frac{\alpha}{y^{\alpha+1}} \cdot \mathbb{1}_{\{y \geq 1\}}.$$

Each sampled coordinate was truncated at 50 in order to make the comparison of outputs on a graph display possible. This value was chosen by visual inspection of a few sample plots. A sample question from the questionnaire is presented in Figure 1[1]. The coordinates of output vectors were sorted in a nonincreasing order which is a standard operation applied to informetric data [1] – the order in which the products appear is not taken into account.

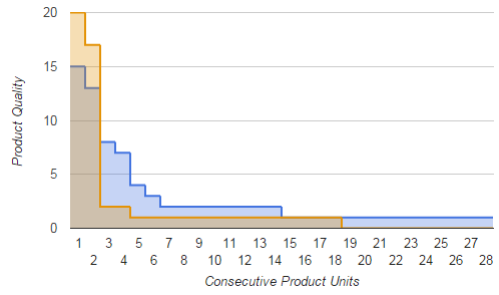

Figure 1: Sample comparison for two output vectors.

The choice of probability distributions is motivated by literature, cf. e.g. [2, 3, 4]. Intuitively, the length of vector $\mathbf{x}^{(i)}$ serves as a proxy for a producer's productivity. On the other hand, the lower the value of parameter α the higher the probability that a producer delivers high quality products.

2.2. Questionnaire

In total, 32 participants who took part in the online survey provided at least 100 answers each (responses by experts with fewer answers were disregarded). In each iteration, an expert was asked to indicate his/her preference toward one of the outputs presented at the same plot. Possible answers were:

- -2 or 2 indicating *strong preference* toward either of the options,
- -1 or 1 indicating *weak preference*,
- 0 representing *indifference*, and
- x representing *incomparability* relation.

Participation in the survey was rewarded with a small prize (sweets or money). The survey was carried out among PhD students in Information Technology as well as at a social network website among the authors' friends. The frequencies of answers are presented in Table 1. In total, 5330 answers were collected.

We also performed a sanity check for reliability of participants' answers by providing randomly a small fraction of repeated questions and by delivering questions in which the preference toward one of the option was fairly obvious (i.e., one of the output vectors dominated the other in terms of the preference relation discussed in Section 3.1 below). In general, the participants appeared to reveal truthful preferences and credible answers.

[1]The questionnaire is available online at http://lasek.rexamine.com/questionnaire/.

Table 1: The answers in the questionnaire and their associated counts.

Answer	*-2*	*-1*	*0*	*1*	*2*	*x*
Count	921	1180	696	1168	1044	321

3. Insights from the questionnaire data

In the following sections we discuss certain aggregation operators and their use to model experts' preferences. We also show how certain class of indexes (more precisely, a generalized h-index) can be tailored to a specific dataset and present some findings on the questionnaire data.

3.1. Interrelation of features and response variable

First of all, let us study how different tools for aggregating producers' output vectors are associated with given comparison outcomes. The idea is that these aggregation operators, mapping a vector of arbitrary length to a single number, should capture the quality and quantity of a producer's output. Assume we are given a vector

$$\mathbf{x} = (x_1, x_2, \ldots, x_n),$$

where the coordinates (product qualities) are sorted in a nonincreasing order, $x_1 \geq x_2 \geq \cdots \geq x_n$. A producer output may be characterized by a number of indicators about his/her productivity as well as quality of supplied products. We decided to use the following popular aggregation operators:

- mean quality of a product $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n} x_i$,
- sum of all qualities $\Sigma(\mathbf{x}) = \sum_{i=1}^{n} x_i$,
- maximal quality of a product x_1,
- number of products n,
- Egghe's g-index $i_G = \max\{i : \sum_{j=1}^{i} x_i \geq i^2\}$ [5],
- Hirsch's h-index $i_H = \max\{i : x_i \geq i\}$ [6],
- Woeninger's w-index $i_W = \max\{i : x_j \geq i - j + 1 \text{ for all } j = 1, \ldots, i\}$ [7].

Apart from the first measure, $\bar{\mathbf{x}}$, the above operators are instances of the so-called impact functions studied in e.g. [1]: they are monotone with respect to each element and with respect to the vector's length. For a compared pair of producers with indicated preference (a single answer in the questionnaire) we construct an explanatory variable (feature) as the difference in valuation of a particular

function for the two output vectors (the valuation for the second one minus the first one).

Moreover, we also used the valuation of the fuzzy preference relation studied in [8]: for two output vectors $\mathbf{x}$ and $\mathbf{y}$, the membership function of fuzzy preference relation $\mathbf{x} \blacktriangleleft \mathbf{y}$ is given by

$$\mu(\mathbf{x},\mathbf{y}) = \begin{cases} \frac{\pi_{yx}}{\pi_{xy}+\pi_{yx}} & \text{if } \pi_{xy}+\pi_{yx} > 0, \\ 0.5 & \text{otherwise,} \end{cases}$$

where $\pi_{xy} = \sum_i \cdot \max\{x_{(i)} - y_{(i)}, 0\}$ (for a vector $\mathbf{x}$ of length n we put $x_m = 0$ for $m > n$). This relation is additive reciprocal (or probabilistic), i.e., $\mu(\mathbf{x},\mathbf{y}) + \mu(\mathbf{y},\mathbf{x}) = 1$ for all pairs $\mathbf{x},\mathbf{y}$. It is also fuzzy transitive (under Łukasiewicz T-norm [9]). It is a fuzzy preference relation in the sense studied by, e.g., Tanino [10]. It is designed to measure preference in pairwise comparison with some degree of uncertainty.

In the next two subsections we discuss the association strength between the proposed features and the response variable.

3.1.1. Modeling individual experts' preference profiles

In order to analyze the dependencies between the proposed measures of quality together with productivity and the response variable, we use ordinal logistic regression model [11]. Let us briefly recall the model here. Let $l_1, l_2, \ldots, l_k$ be given labels and $z_1, z_2, \ldots, z_p$ be explanatory variables. We assume that labels are ordered, i.e., a total order on the set of labels is given, $l_1 < l_2 < \cdots < l_k$ (for example, *good* $<$ *better* $<$ *the best*). According to the model, there is latent variable x^* which follows the logistic distribution with scale parameter equal to 1 and the mean value modeled by a linear combination of response variables

$$\mathbb{E}(x^*) = \beta_1 z_1 + \beta_2 z_1 + \cdots + \beta_p z_p$$

and we assume that the value of variable x^* is known only up to an interval associated with some label,

$$label = \begin{cases} l_1 & \text{if } x^* \le \alpha_1, \\ l_2 & \text{if } x^* \in (\alpha_1, \alpha_2], \\ \cdots & \\ l_k & \text{if } x^* > \alpha_{k-1}, \end{cases}$$

where β_i, $i = 1, 2, \ldots, p$ and α_i, $i = 1, 2, \ldots, k-1$ are the model parameters that we want to estimate. Denoting $\mu_{\mathbf{z}} = \beta_1 z_1 + \beta_2 z_1 + \cdots + \beta_p z_p$, the probabilities of an instance being assigned a particular label are given as

$$\mathbb{P}(label = l_1) = \frac{1}{1 + e^{-\alpha_1 + \mu_{\mathbf{z}}}},$$

$$\mathbb{P}(label = l_k) = 1 - \frac{1}{1 + e^{-\alpha_{k-1} + \mu_{\mathbf{z}}}}$$

and for $i = 2, 3, \ldots, k-1$ we have

$$\mathbb{P}(label = l_i) = \frac{1}{1 + e^{-\alpha_i + \mu_{\mathbf{z}}}} - \frac{1}{1 + e^{-\alpha_{i-1} + \mu_{\mathbf{z}}}}.$$

The model is estimated using the maximum likelihood principle. In our study, to impose the symmetry of the output relation, thresholds levels α_i, $i = 1, 2, .., 4$ were restricted to be symmetric around 0.

We use an ordered logistic regression model to analyse the association between the response variable and the discussed aggregation functions. In this part, we skipped the Egghe's g-index as it was too highly correlated with both the sum ($\Sigma(\mathbf{x})$) and the mean quality ($\bar{\mathbf{x}}$) of the outputs, with over $\varrho = 0.9$ (in terms of the Spearman correlation coefficient).

As a preprocessing step in the model estimation, we standardized the features so that they are of mean 0 and have standard deviation equal to 1. At first, we ran the model for five experts (in columns) with the largest number of provided answers. Table 2 presents the results of applying the estimation procedure. We assume here and throughout the paper the significance level of 0.05. Statistically significant estimates are denoted with $*$. The variable names are as discussed above and FP stands for the valuation of fuzzy preference relation.

Table 2: Results of models estimation (coefficients β) for five experts with the highest number of answers.

	1	2	3	4	5
$\bar{\mathbf{x}}$	1.375*	0.003	1.666	1.409	0.817
x_1	1.546*	0.624*	0.743	-1.861*	-1.083*
$\Sigma(\mathbf{x})$	0.716	1.451*	1.122	3.359*	3.286*
n	0.444	0.427	1.243*	-1.697*	-0.911
i_H	-0.03	-0.151	0.202	-0.279	-0.361
i_W	0.506*	0.935*	0.602	0.925*	1.508*
FP	0.839*	0.434*	0.899*	0.636*	0.243

Table 2 reveals that there is a considerable variability between experts' preference profiles. The magnitude of the estimated coefficients in columns show which indexes are more important than the other ones for particular users. In general, the sign of the estimated coefficients is in line with intuition – a positive value means that the higher the value of difference in certain aggregation functions, the stronger the preference toward the second one of the two compared output vectors. Notably, in case of the fourth expert we obtain significant, negative estimates of coefficients associated with variables x_1 (maximal quality) and n (length of $\mathbf{x}$). We also note that the h-index, i_H, turns out to be insignificant. However, as the variables are correlated to a certain extent, one should be careful when drawing conclusions from the models.

3.1.2. Overall analysis

In this part we present the result of model estimation for all the answers (all the experts altogether) in Table 3.

Table 3: Results of model estimation for all answers in the data set.

	Estimate	Std. error	z-value	p-value
$\bar{\mathbf{x}}$	0.386*	0.141	2.747	0.006
x_1	0.521*	0.089	5.861	0
$\Sigma(\mathbf{x})$	1.179*	0.158	7.468	0
n	-0.136	0.082	-1.654	0.098
i_H	-0.059	0.053	-1.118	0.264
i_W	0.637*	0.059	10.749	0
FP	0.568*	0.053	10.694	0

We note that the sign of the estimated coefficients is in line with intuition. In the case of h-index and vector length n we obtain negative estimates, however, they are insignificant. We may conclude that in overall, as n can be viewed as a proxy for productivity, there is rather an emphasis on quality rather than productivity in the evaluation process by experts. The sum of qualities over all products can be viewed as the best indicator for performance of a producer in pairwise comparisons – the changes of preferences are most significantly influenced by the changes in this quantity.

3.2. Parameter optimization for a certain class of scientific impact indexes

The functions that we used to extract information from data are just examples of a broad class of so-called impact functions. One of the most famous examples of such aggregation operators is the discussed Hirsch h-index. This impact function can be generalized as follows. For an output vector $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ let us define h_c as

$$h_c(\mathbf{x}) = \max\{i = 0, 1, ..., n : cx_i \geq i\}.$$

In a standard setup we have $c = 1$. Using questionnaire data we may tailor the parametrized class of impact indices to a specific expert's preferences. Figure 2 presents an example of computation of h_c for different values of parameter c and an output vector $\mathbf{x} = (10, 9, 8, 5, 4, 4, 3, 1)$. The parameter c serves as a kind of a trade-off between quality and productivity. The higher its value, the more emphasis is placed on productivity (more precisely, the number of positive elements in a sequence). On the other hand, the lower the value of c, the higher quality of the first output value (x_1) in a vector $\mathbf{x}$ is necessary to increase the value $h_c(\mathbf{x})$.

Given a specific dataset, as the one obtained from our online questionnaire, we may ask what is the optimal value of the parameter c for individual experts. To this end, we estimate the ordinal logistic regression model for a single variable as the difference in the h_c-indexes. We find the optimal value of

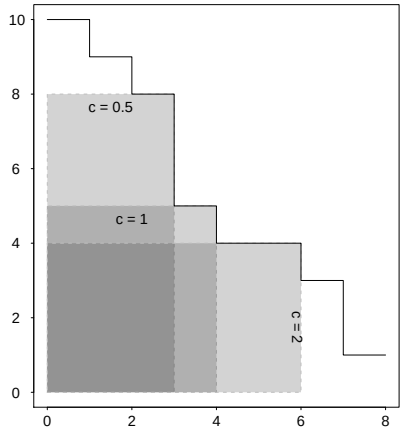

Figure 2: Computation of h_c for three different values of c. We have $h_{0.5} = 3$, $h_1 = 4$ and $h_2 = 6$.

parameter c along with the associated β coefficient in the model by the maximum likelihood method. The results are presented in Figure 3.

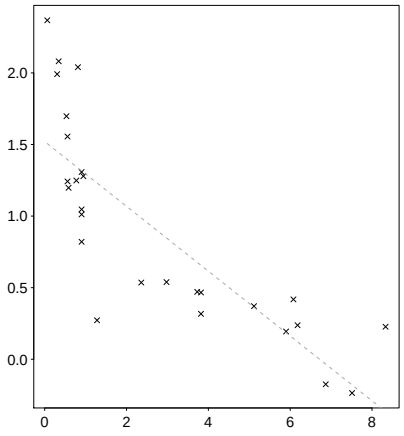

Figure 3: Plot of estimated value of parameter c (x-axis) against associated coefficient β (y-axis) along with regression line presenting linear relationship between the two quantities. Only statistically significant estimates are presented (27 points).

The figure demonstrates that the lower the estimates $c^{(i)}$, the greater the magnitude of the estimated coefficient β (y-axis). This may further indicate that the emphasis in the evaluation process was put on items' quality.

4. Learning the preference relation

The next step of the analysis is the verification of a hypothesis whether extracting features from data

based on existing aggregation operators is more effective an approach than supplying the data in an unprocessed manner. Each producer output is given as a vector of arbitrary length. To compare vectors we employ two approaches:

- compare vectors on each coordinate and equalize their lengths by padding the shorter one with zeros (i.e. consider first few greatest values of the vectors), and
- extract certain features of output vectors (as discussed in Section 3).

In order to validate which approach yields better results, we train several prediction models and examine their predictive accuracy. Under consecutive headings we describe the prediction models used, the evaluation metrics employed and the steps taken to extract features from data and finally we report the results.

4.1. Prediction models

In the numerical experiments, we used several models: ordered logistic regression (OLR), k-Nearest Neighbors (kNN) and the Random Forest (RF) classifier [12, 13] and their implementations [14, 15, 16] in the R language [17]. We use the kNN version of the model for ordinal data [18] as the response variable has natural ordering. We also experimented with the technique for dealing with such kind of response variables suggested by Frank and Hall [19] for RF model. However, this approach did not improve the results in our case.

4.2. Evaluation metrics

To compare different models we employ several accuracy metrics. First, let us introduce some notation. For ith instance in the data set (a pair of compared output vectors), $i = 1, 2, \ldots, N$, its associated true label is denoted by $l_t^{(i)},\ t \in \{-2, -1, 0, 1, 2\}$. Further, let a given model assign probability $\mathbb{P}(l_k^{(i)})$ to label l_k, $k = -2, -1, \ldots, 2$. A decision made by classifier is to assign a label according to the rule $\hat{l}_p^{(i)} = \operatorname{argmax}_k \mathbb{P}(l_k^{(i)})$.

Below we describe evaluation metrics used in our study.

Misclassification rate

Misclassification rate is the basic evaluation measure and equals to the proportion of incorrectly classified instances

$$Misscl = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}_{\left\{ l_t^{(i)} \neq \hat{l}_p^{(i)} \right\}}.$$

Average distance between ranks

The next evaluation metric is based on the distance between the true and predicted label, $d(l_t^{(i)}, \hat{l}_p^{(i)})$. In our setting, we use

$$d(l_t^{(i)}, l_p^{(i)}) = |t - p|,$$

Other choices for the distance metric are possible. Average distance between ranks is calculated as

$$AvgDist = \frac{1}{N} \sum_{i=1}^{N} d(l_t^{(i)}, \hat{l}_p^{(i)}).$$

Rank probability score

Rank probability score is calculated as rescaled, squared L^2 distance between the estimated and observed cumulative distribution function for ordered labels. Let $\hat{F}^{(i)}$ denote estimated distribution function for probability of given labels for ith instance

$$\hat{F}^{(i)}(j) = \sum_{k=-2}^{j} \mathbb{P}(l_k^{(i)}),$$

with $j = -2, -1, \ldots, 2$. Rank probability score is derived as

$$RPS = \frac{1}{4N} \sum_{i=1}^{N} \sum_{j=-2}^{2} \left(\hat{F}^{(i)}(j) - F^{(i)}(j) \right)^2,$$

where $F^{(i)}(j) = 1$ for $j \geq t$ and 0 otherwise for an instance with true label $l_t^{(i)}$.

Concordance index (C-index)

The last evaluation metric we discuss is the concordance index. This statistics is defined for every *usable* pair of compared objects. A pair is usable when a preference toward either of the option was indicated (they are not classified as indifferent neither incomparable). A usable pair is said to be concordant if the predicted and the true label indicate the preference in the same direction, i.e., for an instance labelled with l_1 or l_2 we have $\hat{l}_p \in \{l_1, l_2\}$. Let number of usable pairs be equal M. Formally, the concordance index is defined as

$$C = \frac{1}{M} \sum_{i:\ l_t^{(i)} \neq 0} \mathbb{1}_{\left\{ l_t^{(i)},\ \hat{l}_p^{(i)}\ concordant \right\}} + 0.5 \cdot \mathbb{1}_{\left\{ \hat{l}_p^{(i)} = 0 \right\}}.$$

The second component of the sum is a correction introduced for predictions in which two compared output vectors are considered equal by an expert.

Misclassification is a standard metric to evaluate efficacy of a model [12]. However, we note that this measure does not take into account the ordering of responses. In particular, for an instance with true label l_2, the misclassification error is the same regardless the object is classified as l_1 or l_{-2}. The other discussed metrics address this issue. Average distance between ranks is a measure adapted from [20]. This error measure could be further refined by applying other rank distance functions. For example, for a heavier cost for misclassifying, e.g.,

label l_2 with l_{-2}. Rank probability score has been widely used for evaluation of ordinal responses, e.g., in weather forecasting [21] or match outcome prediction in association football [22]. The last proposed measure - the concordance index - has been used to evaluate forecasts in medical research [23, 24, 25] and also adopted for evaluation of models in the field of preference learning [26].

4.3. Experiment setup

In our experiments we use only examples for which a level of preference was indicated (we disregard 321 samples labelled with *'x'*). First of all, we split the data into two sets: training and test set in proportion 80:20. Parameter optimization (for kNN model) and feature selection for the models was done using 10-fold crossvalidation on the training set. The models were trained for every expert (questionnaire respondent) separately.

4.3.1. Feature extraction

To describe output vectors we use several tools to extract their characteristics as discussed in Section 2. Next, we create a feature which is a difference of the value of the value of a specific function for the two producers. The features were scaled to be 0 in mean and have standard deviation equal 1. As far as the second approach is considered, for a training set consisting of pair of compared output the following procedure is employed. First of all, we find the maximal length of an output vector in the training set $n_{\max}$. Next, each vector is equalized in length according to $n_{\max}$. For a pair of compared vectors we take differences in corresponding coordinates as features to train the model. If in a test set (or, a fold in crossvalidation) we encounter a vector of length $> n_{\max}$ we truncate in to $n_{\max}$.

4.3.2. Feature selection

Random Forest model has an inherent method for selecting relevant features. The parameters of the model were set to default values in R package [14]. Such an approach is valid as long as we focus on comparison of model performance for different features as in the working hypothesis of our paper. As feature selection procedure (indexes) for the ordinal logistic regression and k-Nearest Neighbor classifier we employ the following algorithm. First, we estimate each model based on single feature. We order the features according to average distance between the target and the predicted label (*AvgDist* evaluation metric). Next, we build a model based on the first two the best performing features, first three the best performing features, and so forth. As far as the second approach is concerned, based of pairwise differences in order statistics, we estimate the model for first m elements of vectors, $m = 1, 2, \ldots, n_{\max}$, and choose the optimal value of m. In this way, we assume that first order statistics are more important. This view is motivated by the observation that the evaluation process was more focused on quality as discussed in Section 3.

In case of kNN classifier, the feature selection procedure was carried out for different values of k ranging from 5 to 30 with a step size of 5. In this way, the optimal number of neighbors was selected along with relevant attributes.

4.3.3. Feature importance evaluation

From the described feature selection procedure we derive a ranking of most important features for classification based on their predictive capabilities. For OLR model and different versions of kNN model (for various values of parameter k) we calculated how many times a given feature was selected during training procedure for different users. In this way, we obtain that the most important for classification are: 1. i_G (picked 42 times), 2. $\Sigma(\mathbf{x})$ (30), 3. $\bar{\mathbf{x}}$ (27), 4. x_1 (16) and 5. FP (15). For RF model we obtain feature importance measure (a ranking based on mean decrease in accuracy of classification) during model training [14]. In case of this model we derived ranking of features based on individual importance rankings for 32 participants by, e.g., Borda count, a method for ranking lists aggregation [27]. We obtain the following ranking of features: 1. FP, 2. i_G, 3. $\Sigma(\mathbf{x})$, 4. $\bar{\mathbf{x}}$, 5. x_1, 6. n, 7. i_W and 8. i_H. The results of feature importance evaluation for different models appear to be consistent.

The results of our experiment are presented in the next section for a test set of size 1001 compared pairs.

4.4. Experiment results

Table 4 presents the best performing models. In order to have all evaluation metrics expressed as error measures, in the table we report $1 - C$ rather than concordance index value C. We denote this metric with C'. In this way, for each of the evaluation metrics the lower its value the better the predictive power of a given model.

The models are trained using two approaches: based on features extracted from indexes (denoted with subscript i) and based on coordinatewise differences in output vectors (c). For each model M, M $\in$ {OLR, kNN, RF} and each error metric E, E $\in$ {Misscl, AvgDist, RPS, C}, we performed paired Wilcoxon test to investigate whether mean error estimates of E for M_i and M_c differ significantly (at 0.05 significance level). If they do, the significantly higher mean are marked with $*$.

To provide context for the reported numbers we also include in comparison evaluation statistics for a benchmark model in which every class is assigned equal probability of 0.2 and taking the predicted label to be $l_p^{(i)} = 0$ for every instance i.

Table 4: Results of classification.

	Misscl	*AvgDist*	*RPS*	C'
OLR_i	0.409	0.465	0.086	0.08
OLR_c	0.394	0.454	0.082*	0.075
$k\text{NN}_i$	0.401*	0.457*	0.085*	0.083*
$k\text{NN}_c$	0.453	0.548	0.099	0.122
RF_i	0.385*	0.452*	0.076*	0.078
RF_c	0.434	0.537	0.094	0.092
Equal	0.865	1.255	0.202	0.5

In case of kNN and RF model, based on the test results, we see that the models trained on indexes perform better. A worse performance of "the coordinatewise" approach may be due to noise contained in the data or high correlations between consecutive elements of output vectors. Only in case of OLR model and RPS error metric this approach turned out to yield worse results. We claim that the studied indexes are effective in compressing information contained in data. With the use of a few numbers we can effectively describe a vector of length 21 (on average; according to our data generating process described in Section 2). Therefore, the main hypothesis of our paper if verified positively.

5. Discussion

In this paper we studied the efficacy of certain aggregation operators used in, among others, informetrics. By designing an online survey within the framework of the Producer Assessment Problem, we collected preference data toward pairs of presented numeric sequences. We studied the difference in evaluation process among producers and presented an example of how to tailor a parametrized class of impact indexes to a specific expert's preferences. We observed that the emphasis was put on quality rather than productivity in the revealed preferences. The main goal of the paper was to evaluate certain aggregation operators and test whether they extract information from output vectors effectively. Based on good performance of several prediction models estimated with the use of these operators' valuations on input data, we claim that several tools proved to be effective in this task. Among the best performing aggregation tools in our experiment we identified Egghe's g-index i_G, sum of product qualities $\Sigma(\mathbf{x})$, the fuzzy preference relation FP, mean quality of a product $\bar{\mathbf{x}}$ and the maximal quality of a product x_1.

Acknowledgments

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

Jan Lasek 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] A. Cena and M. Gagolewski. OM3: Ordered maxitive, minitive, and modular aggregation operators – Axiomatic and probabilistic properties in an arity-monotonic setting. *Fuzzy Sets and Systems*, 264:138–159, 2015.

[2] K. Barcza and A. Telcs. Paretian publication patterns imply Paretian Hirsch index. *Scientometrics*, 81(2):513–519, 2009.

[3] W. Glänzel. H-index concatenation. *Scientometrics*, 77(2):369–372, 2008.

[4] W. Glänzel. On some new bibliometric applications of statistics related to the h-index. *Scientometrics*, 77(1):187–196, 2008.

[5] L. Egghe. Theory and practise of the g-index. *Scientometrics*, 69(1):131–152, 2006.

[6] J. E. Hirsch. An index to quantify individual's scientific research output. *Proceedings of the National Academy of Sciences*, 102(46):16569–16572, 2005.

[7] G. J. Woeginger. An axiomatic characterization of the Hirsch-index. *Mathematical Social Sciences*, 56(2):224–232, 2008.

[8] M. Gagolewski and J. Lasek. The use of fuzzy relations in the assessment of information resources producers' performance. In D. Filev et al., editors, *Intelligent Systems'2014*, volume 323 of *Advances in Intelligent Systems and Computing*, pages 289–300. Springer, 2015.

[9] J. Fodor and M. Roubens. *Fuzzy Preference Modelling and Multicriteria Decision Support*. Springer, 1994.

[10] T. Tanino. *Fuzzy Preference Relations in Group Decision Making*, pages 54–71. Springer Berlin Heidelberg, 1988.

[11] P. McCullagh. Regression models for ordinal data. *Journal of the Royal Statistical Society*, 42(2):109–142, 1980.

[12] I. M. Hall, I. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers, 2005.

[13] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, October 2001.

[14] A. Liaw and M. Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.

[15] K. Schliep and K. Hechenbichler. *kknn: Weighted k-Nearest Neighbors*, 2014. R package version 1.2-5.

[16] R. H. B. Christensen. ordinal—regression models for ordinal data, 2013. R package version 2013.9-30 http://www.cran.r-project.org/package=ordinal/.

[17] R Core Team. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria, 2015.

[18] K. Hechenbichler and K. Schliep. Weighted k-nearest-neighbor techniques and ordinal classification. Technical report.

[19] E. Frank and M. Hall. A Simple Approach to Ordinal Classification. In *EMCL '01: Proceedings of the 12th European Conference on Machine Learning*, pages 145–156, London, UK, 2001. Springer-Verlag.

[20] E. Hüllermeier and J. Fürnkranz. Preference learning: models, methods, applications – Proceedings of the KI-2003 Workshop. Technical report, Österreichisches Forschungsinstitut für Artificial Intelligence.

[21] M. T. Hamill. Hypothesis tests for evaluating numerical precipitation forecasts. *Weather and Forecasting*, 14(2):155–167, 1999.

[22] A. Constantinou and N. Fenton. Solving the problem of inadequate scoring rules for assessing probabilistic football forecast models. *Journal of Quantitative Analysis in Sports*, 8(1), 2012.

[23] F. W. Dekker G. Tripepi, J. K. Jager and C. Zoccali. Statistical methods for the assessment of prognostic biomarkers (part i): discrimination. *Nephrology Dialysis Transplantation*, 25(5):1399–1401, 2010.

[24] B. Krishnapuram C. Dehing-oberije V. C. Raykar, H. Steck and P. Lambin. On ranking in survival analysis: bounds on the concordance index. In Y. Singer J. C. Platt, D. Koller and S. Roweis, editors, *Advances in Neural Information Processing System*, pages 1209–1216, Cambridge, Mass, USA, 2008. MIT Press.

[25] K. L. Lee E. H. Frank and D. B. Mark. Tutorial in biostatistics multivariable prognostic models: issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in medicine*, 15:361–387, 1996.

[26] J. Fürnkranz and editors. E. Hüllermeier, editors. *Preference Learning*. Springer-Verlag, 2011.

[27] W. Gaertner. *A Primer in Social Choice Theory*. Oxford University Press, 2009.