Paper 083-30

A Multi-Model-Approach to Improve Final Results

Ryszard Szupiluk, Polska Telefonia Cyfrowa Ltd., Al. Jerozolimskie 181, Warsaw, Poland
Piotr Wojewnik, Polska Telefonia Cyfrowa Ltd., Al. Jerozolimskie 181, Warsaw, Poland
Tomasz Zabkowski, Polska Telefonia Cyfrowa Ltd., Al. Jerozolimskie 181, Warsaw, Poland

ABSTRACT

In this paper we apply multidimensional decompositions to improve modelling results. Results generated by a model usually include both wanted and destructive components. In case of a few models, some of the components can be common to all of them. Our aim is to find basis elements and distinguish the components with the positive influence on the modelling quality from the negative ones. After rejecting the negative elements from the models' results we obtain better results in terms of some standard error criteria. The identification of the basis components can be performed by ICA and PCA transformations.

The procedure of models decomposition and improvement is implemented in SAS/IML®. The models' errors are analysed in SAS® BASE and the graphs are generated in SAS/GRAPH®. The automation is performed using SAS® MACRO language. Paper is addressed to the audience with data mining and statistical background.

INTRODUCTION

Data Mining (DM) is the process of finding trends and patterns in data [6,10]. Usually it aims at finding the previously unknown knowledge that could be used for business purposes such as fraud detection, client/ market segmentation, risk analysis, customer satisfaction, bankruptcy prediction, etc. The methodology of DM modelling can follow the SEMMA procedure introduced by SAS®. It consists of parts: Sample, Explore, Modify, Model and Assess [11].

Typically, in data mining problem many models are tested and then, according to particular criterion, the best one is chosen. The other models are left out. In this paper we propose to utilize information given by them. The motivation of such methodology can be based on somewhat ambiguity formulation "the best model". There are many different criteria which can indicate different models as the best one. On the other hand even if several model results are not the best according to specific criterion it is still possible to utilize them to improve the final effect.

Usually, solutions of the model aggregation problem propose to combine a few models by mixing their results or parameters [7,15]. Our aim is to integrate the knowledge uncovered by the set of the models applying decomposition of the models results into signals, rejecting from them the destructive ones and operation inverse to previous decomposition [13, 14]. Such transformation can be done by means of Independent Component Analysis (ICA) and Principal Component Analysis (PCA) [8,9]. The presented methods use many signals and different decompositions and can utilize different criteria to find more accurate final result what leads to multivariate analysis what can be performed with SAS/IML®.

MODEL RESULTS' INTEGRATION

The models try to represent the dependency between input data and target, so they bring some knowledge about the real value [5]. We assume that each model results include two types of components: positive associated with target and destructive associated with inaccurate learning data, individual properties of models etc. Many of good and bad components are common to all the models due to the same target, learning data set, similar model structures or optimization methods. Our aim is to explore information given simultaneously by many models to identify and eliminate components with destructive impact on model results.

We assume that result of i-th model x_i, $i = 1,...,m$, with N observations, is linear combination of positive impact components $t_1, t_2, ..., t_p$, and destructive components $v_1, v_2, ..., v_q$, what gives

$$x_i = \alpha_{i1} t_1 + ... + \alpha_{ip} t_p + \beta_{i1} v_1 + ... + \beta_{iq} v_q . \quad (1)$$

In close matrix form we have

$$x_i = \boldsymbol{\alpha}_i \mathbf{t} + \boldsymbol{\beta}_i \mathbf{v} , \quad (2)$$

where: $\mathbf{t} = [t_1, t_2, ..., t_p]^T$ is a $p \times N$ matrix of target components, $\mathbf{v} = [v_1, v_2, ..., v_q]^T$ is a $q \times N$ matrix of residuals, $\boldsymbol{\alpha}_i = [\alpha_1, ..., \alpha_p]$, $\boldsymbol{\beta}_i = [\beta_1, ..., \beta_q]$ are vectors of coefficients. In case of many models

$$\mathbf{x} = \mathbf{A}\mathbf{s} , \quad (3)$$

where $\mathbf{x} = [x_1, x_2, ..., x_m]^T$ is a $m \times N$ matrix of model results, $\mathbf{s} = \begin{bmatrix} \mathbf{t} \\ \mathbf{v} \end{bmatrix}$ is a $n \times N$ matrix of basis components ($n = p + q$), and $\mathbf{A} = \begin{bmatrix} \boldsymbol{\alpha}_1 & \boldsymbol{\beta}_1 \\ \vdots & \\ \boldsymbol{\alpha}_m & \boldsymbol{\beta}_m \end{bmatrix}$ is $m \times n$ matrix of mixing coefficients.

Our concept is to identify source signals and mixing matrix from observed models' results x, and reject the common residuals v, which means replacing destructive signals in s by zero. After proper identification and rejection we have $\mathbf{s} = \begin{bmatrix} \mathbf{t} \\ \mathbf{0} \end{bmatrix}$ what allows us to obtain purified target estimation

$$\hat{\mathbf{x}} = \mathbf{A}\hat{\mathbf{s}} = \mathbf{A}\begin{bmatrix} \mathbf{t} \\ \mathbf{0} \end{bmatrix}. \tag{4}$$

The main problem is to distinguish $\mathbf{t}$ from $\mathbf{v}$. This task requires some decision system to resolve it. If we don't have any sophisticated method we can simply check impact of all components is s for final results. It means the rejecting one by one, source signal s_i and mix the rest in transformation inverse to decomposition system, Fig.1.

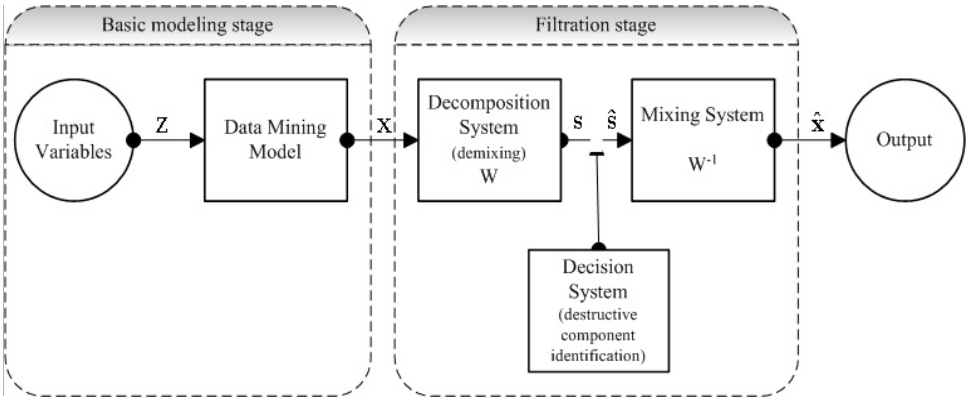

Fig. 1. Data Mining with multidimensional filtration.

DECOMPOSITION ALGORITHMS

In this paper the estimation of the source signals is realized using PCA and ICA. The methods use different features and properties of data but both of them can be considered as looking for data representation of the form (3). To find the latent variables $\mathbf{A}$ and $\mathbf{s}$ we often can use an transformation defined by matrix $\mathbf{W} \in R^{n \times m}$, such that

$$\mathbf{y} = \mathbf{W}\mathbf{x}, \tag{5}$$

where $\mathbf{y}$ is related to s and it satisfies specific criteria as e.g. decorrelation or independence. Particular methods of W estimation are given as follows. For simplicity we assume that $n = m$.

PRINCIPAL COMPONENT ANALYSIS (PCA) is a second order statistics method associated with model (1) where the main idea is to obtain orthogonal variables $y_1, y_2, ..., y_m$ ordered by decreasing variance [9]. To find the transformation matrix W the eigenvalue decomposition (EVD) of correlation matrix $\mathbf{R}_{xx} = E\{\mathbf{x}\mathbf{x}^T\}$ can be performed by:

$$\mathbf{R}_{xx} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^{\mathrm{T}}, \tag{6}$$

where $\boldsymbol{\Sigma} = \mathrm{diag}[\sigma_1, \sigma_2, \ldots, \sigma_m]$ is diagonal matrix of eigenvalues ordered by decreasing value, $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m]$ - is orthogonal matrix of eigenvectors of $\mathbf{R}_{xx}$ related to specific eigenvalues. The transformation matrix can be obtained as

$$\mathbf{W} = \mathbf{U}^T. \tag{7}$$

INDEPENDENT COMPONENT ANALYSIS (ICA) is a statistical tool, which allows us to decompose observed variable into independent components [2,8]. Typical algorithms for ICA explore higher order statistical dependencies in dataset, so after ICA decomposition we have got signals (variables) without any linear and non-linear statistical dependencies. This is the main difference from the standard correlation methods (PCA), which allow us to analyze only the linear dependencies. Form many existing ICA algorithms we focus on Natural Gradient method which basis on-line form is as follow:

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \mu(k)\left[\mathbf{I} - \mathbf{f}(\mathbf{y}(k))\mathbf{g}(\mathbf{y})^T(k)\right]\mathbf{W}(k), \tag{8}$$

where $\mathbf{W}(k)$ is the transformation matrix in k iteration step, $\mathbf{f}(\mathbf{y}) = [f_1(y_1),...f_n(y_n)]^T$ and $\mathbf{g}(\mathbf{y}) = [g_1(y_1),...g_n(y_n)]^T$ are vector of nonlinearities which can be adaptively chosen according to normalized kurtosis of estimated signals $\kappa_4(y_i) = E\{y_i^4\}/E^2\{y_i^2\} - 3$, see Table 1 [1-4].

	$f_i(y_i)$	$g_i(y_i)$
$\kappa_4(y_i) > 0$	$\tanh(\beta_i y_i)$	$\text{sign}(y_i)\mid y_i \mid^{r_i-1}$
$\kappa_4(y_i) < 0$	$\text{sign}(y_i)\mid y_i \mid^{r_i-1}$	$\tanh(\beta_i y_i)$
$\kappa_4(y_i) = 0$	y_i	y_i
where $r_i \geq 2$, β_i is a suitable constant.		

Table 1. Nonlinearities for Natural Gradient algorithm.

For our purpose the more convenient form of the algorithm is the batch type, so we modify (8). We take expected value of (2) and utilize fact that after proper learning we have $\mathbf{W}(j+1) = \mathbf{W}(j)$. Therefore assuming $\mu(k) = 1$ we have

$$\mathbf{R}_{fg} = E\left\{\mathbf{f}(\mathbf{y})\mathbf{g}(\mathbf{y})^T\right\} = E\left\{\mathbf{f}(\mathbf{W}\mathbf{x})\mathbf{W}\mathbf{x}^T\right\} = \mathbf{I}, \tag{9}$$

what is a condition of proper W estimation. The full batch algorithm is as follows:

1. For initial $\mathbf{W} = \mathbf{W}_0$ and $\mathbf{y} = \mathbf{W}\mathbf{x}$,
2. Perform $\mathbf{R}_{fg} = E\left\{\mathbf{f}(\mathbf{y})\mathbf{g}(\mathbf{y})^T\right\}$ and next to obtain symmetric matrix compute $\overline{\mathbf{R}}_{fg} = \frac{1}{2}[\mathbf{R}_{fg} + \mathbf{R}_{fg}^T]$,
3. Use the eigenvalue decomposition $\overline{\mathbf{R}}_{fg} = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^T$,
4. Compute $\mathbf{Q} = \mathbf{\Sigma}^{-1/2}\mathbf{U}^T$ and next $\mathbf{z} = \mathbf{Q}\mathbf{y}$,
5. Substitute $\mathbf{W} \leftarrow \mathbf{Q}\mathbf{W}$, $\mathbf{y} \leftarrow \mathbf{z}$ and repeat steps 2 - 4.

To improve efficiency of ICA algorithms the data pre-processing such as decorrelation can be performed [2]. A typical number of iterations for batch algorithm are about 50-100.

CHOOSING THE BEST MODEL

Application of the method described above gives various possibilities to improve results. It can be done in two ways.
1. We can decide to use only one criterion and to choose the best model according to that criterion.
2. We can apply an approach where different criteria are considered for the purpose of model selection (proposed in this paper).

Let's assume there is a target value $t(j)$ estimated as $x(j)$ with some residual $e(j)$, where $x(j) = t(j) + e(j)$, and $j = 1,...,N$ means the number of observation. The one of most commonly used statistics are then: Mean Square Error $MSE = \frac{1}{N} \cdot \sum_{j=1}^{N} e(j)^2$ and Mean Absolute Deviation $MAD = \frac{1}{N} \cdot \sum_{j=1}^{N} |e(j)|$, [4, 5]. These statistics reflect various aspects of the modelled dependency. We propose to observe both of them in process of models decomposition and chose such transformation, that both of the statistics are improved. If many models and a few criteria are used simultaneously we can expect that the results will be more general.

PRACTICAL EXPERIMENT

The multi-model-improvement methodology will be applied to the invoice prediction task in telecommunications. The problem is to predict the amount of money the individual client spends on services of mobile operator. The models use nine variables: monthly subscription, the last and the previous invoice amount, unpaid amount left from the last invoice, average payment delay, number of cells visited in last month, customer lifetime value, sum of invoices and open amount from the last six months. Five linear and five MLP model structures where chosen for testing the approach described in this paper. The learning set included 6000 observations, the validating one - 3982, and the testing one - 1500.

The basis tool for implementation above method is SAS/IML® which is a good tool for multivariate analysis. The eigen value decomposition (EVD) was employed to obtain ICA and PCA transformations. In ICA algorithm we put $\beta_i = 1$ and $r_i = 1$. The calculations of models' errors were done in SAS® BASE and the graphs are the output of the SAS/GRAPH® procedures. In order to automate the programs the SAS® MACRO language was used.

```
%macro iml_pca;/* PCA decomposition */

%let ns=10; /* number of signals*/
%do i=1 %to &ns;

proc iml;
use pca.model_results; read all var _all_ into x;
/* matrix x - models' results*/
x=x`; nr=nrow(x);nc=ncol(x);

R=x*x`/nc; /* cov matrix */
call eigen(Sigma,U,R);
W=U`;y=W*x; i=&i; /* component's number that equals 0 (removed)*/
y[i,]=0;
x_new=(inv(W)*y)`;

create pca.res_pca_out_m&i from x_new; append from x_new;
/*matrix x_new denotes x̂_i -improved results*/
run;quit;
%end;
%mend;
%iml_pca;

%macro iml_ica; /* ICA decomposition */
%let ns=10; /* number of signals*/
%do i=1 %to &ns;
proc iml;
use evd.model_results; read all var _all_ into x; x=x`; /* matrix x -model's
results*/
nr=nrow(x);n=ncol(x);

/*decorrelation stage*/

R=x*x`/n; call eigen(Si,U,R); W=U';x=U`*x;

do j=1 to 50; /*no of algorithm iterations*/
do p=1 to nr; /* nonlinearities computation*/
    xd = x[p,]-x[p,:];
    sd = sqrt(xd*xd`/(n-1));

/* k- kurtosis estimation*/
k=(n*(n+1)/((n-1)*(n-2)*(n-3))) *((xd##4)[,+]/(sd**4))-3*((n-1)*(n-1))/((n-2)*(n-
3));
fx=x; gx=x;
if  k>0 then do; fx[,p]=x[,p]##3;      gx[,p]=tanh(x[,p]);     end;
if  k<0 then do; fx[,p]=tanh(x[,p]);   gx[,p]=x[,p]##3;        end;
end;

R=fx*gx`/n;
call eigen(Sigma,U,R);
x=U`*x;
W=U`*W;
end;

i=&i; x[i,]=0; /*component's number that eq 0 (removed)*/
x_new=(inv(W)*x)`;
create evd.res_ica_out_m&i from x_new;
append from x_new; /*matrix x_new denotes x̂_i -improved results*/
run;quit;
%end;

%iml_ica;
```

Having the table with models' results we can perform these two macros below to do decompositions for ten models. Every time the macro does the iteration one of the components is removed and the results are written into the file.

In order to compare the performance of the models before and after decompositions the two error criteria MSE and

MAD were calculated. By inspecting MAD and MSE errors before and after decompositions we can discover that in many cases, rejecting of the specific component, improved the prediction quality of the model. In order to compare the modelling results before and after transformations it is recommended to put it on the graph. The visualization helps a lot to decide which transformation to use and how much we improved our results in terms of given measures (MSE and MAD). The graphs in Fig. 2-3 present the improved results after applying specific decomposition and after removing given component. The dots and triangles denote the models. It is clearly visible that, in general, the results can be improved.

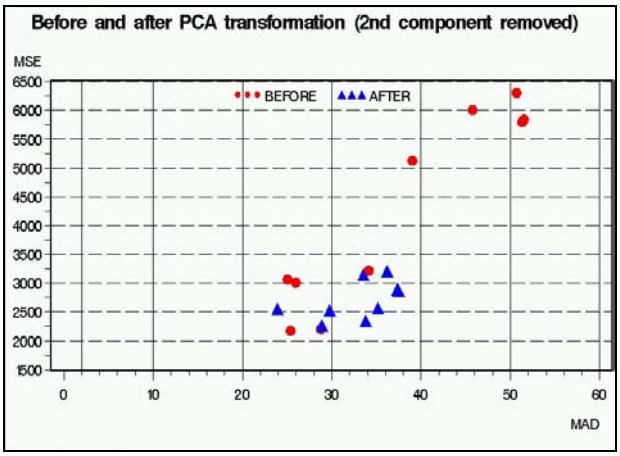

Fig. 2. Results after PCA transformations.

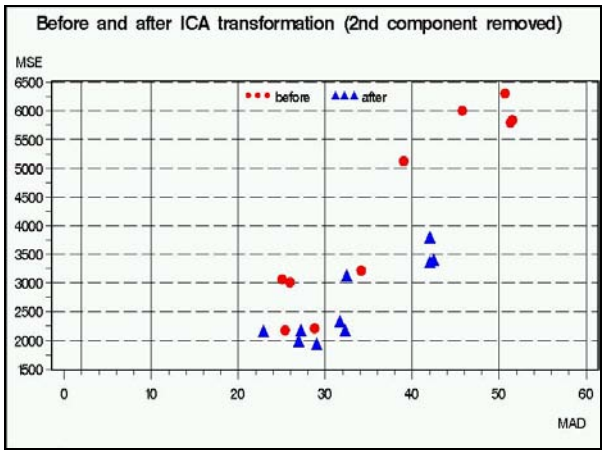

Fig. 3. Results after ICA transformations.

CONCLUSIONS

In this article we present a new approach to the concept, where the information from few models is integrated. Unlike many other works in this field our approach not only helps to improve the models accuracy but also enhances their generalization abilities due to combining the constructive components common to all the models.

It is an unquestionable fact, that various criteria can be used for model evaluation, but it is hard to find the superior one, because the criteria reflect different aspect of the problem. This independence of the model quality measures usually brings some confusion to the analysts. From our point of view, this fact is not confusing any more, but it is a valuable knowledge that can be successfully used for models improvement.

The natural question is what happens, if it is not possible to check the assumption of the linear dependency between models' results and source signals. We can still use the presented approach, but the interpretation of the estimated signals is not straightforward.

REFERENCES

[1] S. Amari, A. Cichocki, and H.H. Yang, A new learning algorithm for blind signal separation, in Advances in Neural Information Processing Systems, NIPS-1995, vole. 8, pp. 757-763. MIT Press: Cambridge, MA, 1996.
[2] A. Cichocki and S. Amari, Adaptive Blind Signal and Image Processing, John Wiley, Chichester, 2002.
[3] A. Cichocki, I. Sabala, S. Choi, B. Orsier, R. Szupiluk, Self adaptive independent component analysis for sub-Gaussian and super-Gaussian mixtures with unknown number of sources and additive noise, NOLTA-97, vol. 2, Hawaii, U.S.A. pp. 731-734, 1997.
[4] S. Cruces, A. Cichocki, L. Castedo, An iterative inversion method for blind source separation, Proceedings of the 1st International Workshop on Independent Component Analysis and Blind Signal Separation (ICA'99), pp. 307--312, Assois, France, January, 1999.
[5] W. H. Greene, Econometric analysis, NJ Prentice Hall, 2000.
[6] R. Groth, Data Mining. Building Competitive Advantage Prentice Hall Inc., Upper Saddle River, New Jersey 2000.
[7] J. Hoeting, D. Madigan, A. Raftery, C. Volinsky, Bayesian model averaging: a tutorial Statistical Science, 14, 382-417, 1999.
[8] A. Hyvärinen, J. Karhunen, E. Oja E, Independent Component Analysis, John Wiley, 2001.
[9] I. T. Jolliffe: Principal Component Analysis, Springer Verlag, July, 2002.
[10] R. L. Kennedy (Editor), Y. Lee, B. Van Roy, C. Reed and R.P. Lippman Solving Data Mining Problems with Pattern Recognition Prentice Hall, December 1997.
[11] SAS® System Help Enterprise Miner™ Release 4.1.
[12] G. Schwarz, Estimating the Dimension of a Model, The Annals of Statistics 6, 461-471.

[13] R. Szupiluk, P. Wojewnik, T. Ząbkowski, Independent Component Analysis for Filtration in Data Mining, Proc. of IIPWM'04 Conf., Zakopane, Poland. Published in Advances in Soft Computing, Springer Verlag, Berlin, 2004, 117-128.
[14] R. Szupiluk, P. Wojewnik, T. Ząbkowski, Model Improvement by the Statistical Decomposition. Artificial Intelligence and Soft Computing - ICAISC 2004. Published in Lecture Notes in Computer Science, Springer-Verlag, Heidelberg, 2004, 1199-1204.
[15] Y. Yang, Adaptive regression by mixing, Journal of American Statistical Association, 96, 574-588, 2001.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Author Name	Ryszard Szupiluk
Company	Polska Telefonia Cyfrowa Ltd.
Address	Al. Jerozolimskie 181
City state ZIP	Warsaw 02-222, Poland
Work Phone:	(+48 22) 413 61 71
Fax:	(+48 22) 413 62 04
Email:	rszupiluk@era.pl