Increasing interpretability and prediction rate by combining self-organizing maps with modeling algorithms

Ivan Ryzhikov[1] Mikko Huovinen[2] Yrjö Hiltunen[1]

[1]Department of environmental and biological sciences, University of Eastern Finland, Finland, {ivan.ryzhikov,yrjo.hiltunen}@uef.fi

[2]Yara, Kuopio, Finland, mikko.huovinen@yara.com

Abstract

We consider supervised learning problems, for which we need not only the accurate model, but also the model, that explains the relation between inputs and a target variable. There are modeling problems, when production experts can measure their confidence in the modeling results by modeling metrics, such as accuracy, but need an explanation for what was the reason of desirable or undesirable situation or system state in the past. In this study we utilize a combination of self-organizing maps and multiple linear modeling to increase the interpretability and accuracy. We assume that the target variable can be explained differently by different patterns that characterizes inputs data. By solving clustering problem for subset of inputs, we have structured data and can relate each cluster to its representative or cluster profile, which explains the cluster. Based on that structure we build linear model for each cluster dataset, and coefficients of this model explain the influence of factors for particular inputs characteristics. To cut the number of inputs we use L1-regularization for linear model. Proposed approach was tested on several industry related problems and implemented in application.

Keywords: explanation, self-organizing map, risk estimation, postprocessing

1 Introduction

Digital transformation makes it possible for industries to find answers on many questions in mathematical models. Machine learning algorithms, statistical analysis and visualization reveal dependencies between production efficiency and processes factors based on observed data. Mathematical models and their applications become a main part of support decision making platforms. Since the models are data-driven, production experts need to measure the adequacy of models, but there is no general way to provide this estimation. Nonlinear models could give a very high prediction rate and good generalization, but due to its complexity it is difficult, if even possible, to interpret the model. On the other hand, simple models can be interpretable, but in some cases give lower prediction rate, so one cannot be confident in modeling results and use the extracted from the model knowledge. In this study we use a combination of clustering approach, such as self-organizing map, and simple modeling approaches proving that these combination makes the final composite model more flexible but still interpretable. Simpler model could be not good at generalizing, because the relation between the inputs and target variables cannot be identified with those simple rules. Another reason of bad generalization is when simple rules meet contradictions in data. But these contradictions could disappear if these are related to patterns in data.

We propose an approach that outperforms simple modeling approaches but keeps its interpretability benefits. This approach increases our confidence in data-driven models and clarifies effects between the target variable and inputs. Applying self-organizing maps helps one to understand the main patterns in the data and helps to see which pattern can be explained with simple models and which cannot and requires nonlinear models. Proposed approach discovers if the main influential factors are different for different patterns in data. This takes place in many cases, for example: seasonal effects or different input materials can lead to situations, where one inputs become more influential on target variable over another. The goal of this approach is to understand what one can do to improve the situation and why. In research we apply linear modeling with and without regularization, and Kohonen's self-organizing maps (SOM) (Kohonen, 1995). Linear models allow us to utilize the well-known statistics, such as p-values and F-score. When we apply l_1 and l_2 regularization (Gareth et al., 2013; Kuhn and Johnson, 2016) and cross-validation, we reduce the number of variables without loss of generalization and prediction rate. Self-organizing maps returns clusters, which can be characterized by their profiles. Profiles can be determined with reference vectors, or average or median values by cluster.

Combination of unsupervised learning and supervised learning can be met in different studies. In some cases, this combination improves the prediction metric. In (Lin et al., 2016) SOM is combined with support vector machine algorithm to improve the forecast of reservoir inflow during typhoon periods.

The proposed approach has been tested on several production data analysis problems and proved its reliability in decision making and understanding the causality between the effects appearing and the system state or input material characteristics.

SOM provides interpretable visualizations. One can see clusters and their properties: number of elements, model prediction rate on train or test data, data pattern that describes the cluster and the most influential variables for that cluster.

An application solving the data analysis and modeling problem with the proposed approach was implemented in R (R Core Team, 2018) and R Shiny (Chang et al., 2021) framework. It allows user to upload the dataset and set the clustering or learning parameters and build clusters and models. As a result, user sees statistics by cluster, modeling results by clusters and cluster profiles in interactive visualization made with "ggiraph" package (Gohel and Skintzos, 2020).

2 Modeling by Clusters

There are many modeling problems, when we are interested not only in the model accuracy, but the in model that explains what factors cause desirable situation. At the same time, model needs to be accurate, otherwise we cannot be confident in explanations that it brings. Linear model, ridge regression and lasso (Gareth et al., 2013; Kuhn and Johnson, 2016) regression give simple explanation on what factors have positive or negative effect on the target variable, but these modeling approaches have low accuracy if the relation between the inputs and outputs is nonlinear (Gareth et al., 2013). Flexible models need specific techniques to reveal the relation between inputs and output, which gives the relative importance of inputs, but leave behind the scenes the detailed explanation. Thus, the production expert cannot decide what condition leads to desirable or undesirable situation.

The main assumption of this study is that there are contradictions in effects of factors on the target variable, which make simple models inaccurate, and these contradictions can be caused by different relation between inputs and outputs for different patterns. The proposed approach is illustrated on Figure 1.

We assume that clusters performed only on a subset of input variable can already give us acceptable result. For example, we can leave out all the process variables and use only the characteristics of inputs materials. The approach consists of three steps, which are given on Figure 1 from the bottom to the top:

1 – We select variables that we will use in clustering analysis.

2 – We provide clustering analysis and reveal patterns in data.

3 – For each dataset related to pattern we solve modeling problem separately.

4 – We analyze the relation between the patterns in data and modeling results.

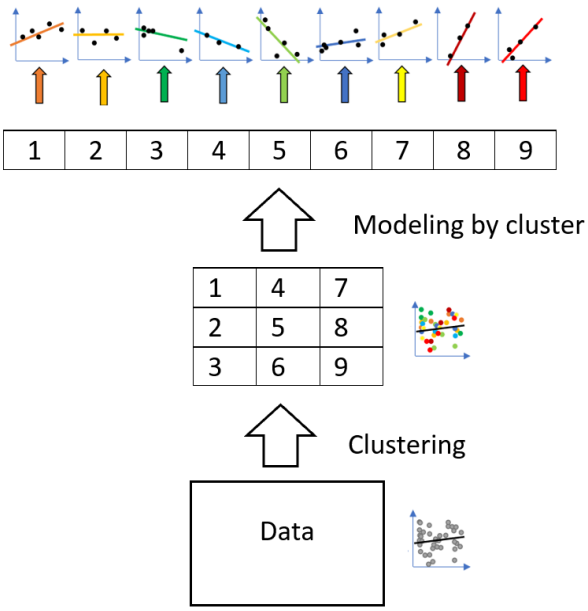

Figure 1. From data to clusters and model for each cluster.

As we mentioned above, linear model does not fit if the relation is nonlinear, but if nonlinearity caused by different patterns in data, we could solve modeling problem for each pattern. That is what one can see in Figure 1. If we use the whole dataset for training, we have a model, which does not give us the satisfactory accuracy level and thus, we cannot trust its coefficients as influence representation. But once we find clusters in the data and solve the modeling problem for each subset that is represented by a cluster, system predictability increases, and we can be more confident in explanations that these models provide. In that case we lessen the contradiction between the influence of different factors, which happens in different system state, according to the patterns found.

Let $X = \{x_1, x_2, \dots, x_s\}$, $x_i \in R^n$, $T = \{t_1, t_2, \dots, t_s\}$, be the observations and times, respectively, s is a dataset size. Let $Y = \{y_1, y_2, \dots, y_s\}$ be the target variable. In this paper we consider regression problem and binary classification problem. When solving binary classification problem, we search for model

$$\begin{gathered} \forall i\ y_i \in \{0,1\}, f_c(\cdot): R \to \{0,1\}, \\ m(\cdot): R^n \to R, \\ \sum_{i=1}^{s} [y_i = f_c(m(x_i))] \to \min, \end{gathered} \tag{1}$$

where f_c is a function that maps the linear model m prediction. Without loss of generality, let classification function f_c map model prediction to the 1st class, if the prediction value is smaller than 0.5.

For regression problem we are interested in model that predicts the output accurately,

$$\forall i\ y_i \in R,\ m(\cdot): R^n \to R,$$
$$\sum_{i=1}^{s} \left(y_i - m(x_i)\right)^2 \to \min. \quad (2)$$

When solving problems (1) and (2) we are interested in model, which can satisfy our expectations in interpretability.

In general, one can solve any other supervised learning problem, but in this study, we focus on (1) and (2). Nevertheless, the developed analytical application allows solving multimodal classification problem.

2.1 Clustering

There are cases when the inputs can reflect different situations: different material types, different content of chemical elements, different shifts, seasons, etc. We expect to see that these factors would affect the relation between the inputs and outputs of the model. But these examples are not the only cases, there could be different patterns in the data that we need to reveal. Because of that we use clustering algorithms to find all the patterns in data.

In this study we utilize self-organizing maps to find the patterns in data. We chose SOM, because it preserves the data structure and makes it possible to visualize the clustering results on two-dimensional plot. Before we train SOM, we center and scale the inputs and keep the scaling parameters to preprocess the new observations. We train SOM on input data only because we need to apply it to new observations and for new observations, we do not know the output value yet.

It is important that clustering problem can be solved for a selection of inputs, which make our clusters more interpretable and allows using of the production experts experience. For example, the input data contain temperatures of machine tools and material analysis. Expert knows that temperatures is something that we cannot control and those change rapidly, but the material analysis changes once in two or three months and there could be some differences in how the process is going. In that part one can test different hypothesis on what variables should we select when do clustering. Let us denote

$$I = \{i_1^c,\ \dots,\ i_q^c\}, \quad (3)$$

as the set of indices of variables, we select for the clustering of the data and q is the number of selected variables.

Once the clustering problem is solved, we have labels or clusters number $C = \{c_1, c_2, \dots, c_s\}$, so each observation has one and only one label $x_i \leftrightarrow c_i$. Now we can split the dataset by clusters:

$$\begin{gathered}\tilde{X}_j = \{x_i \in X,\ i = 1, \dots, s:\ c_i = l_j\},\\ \tilde{T}_j = \{t_i \in T,\ i = 1, \dots, s:\ c_i = l_j\},\\ \tilde{Y}_j = \{y_i \in Y,\ i = 1, \dots, s:\ c_i = l_j\},\\ j = 1, \dots, n_c,\end{gathered} \quad (4)$$

where n_c is number of clusters or patterns and l_j is label of j-th cluster.

SOM requires several parameters. We need to set the grid dimension or the number of neurons and their topology. Let g_v be the number of neurons vertically and g_h be the number of neurons horizontally. The total number of neurons is $g_v \cdot g_h$. Algorithm has the following parameters: number of times the whole dataset will be presented to the network and the radius of neighborhood.

There is "kohonen" package in R (Wehrens and Kruisselbrink, 2018), which we use in this study, when solving the clustering problem. In numerical tests we used the default values for parameters and searched for the grid that is best for the dataset. In application it is possible to set the SOM algorithm parameters.

2.2 Modeling

In this part we do linear model for each of the subset of the dataset (2). First, we consider regression problem and linear models with regularization. To minimize the values of coefficients we use l_2-regularization and to reduce the number of input variables we use l_1-regularization

$$\sum_{i=1}^{s_j} \left(\left(\tilde{Y}_j\right)_i - m_j\left(\left(\tilde{X}_j\right)_i, \theta_j\right)\right)^2 + \alpha P(\theta_j) = \min(\theta_j), \quad (5)$$

$$m_j\left(\left(\tilde{X}_j\right)_i, \theta_j\right) = \sum_{k=1}^{n} \theta_k^j \cdot \left(\left(\tilde{X}_j\right)_i\right)_k + \theta_0^j,$$
$$l_1: P(\theta) = \sum_{i=0}^{n} \left|\theta_i^j\right|,$$
$$l_2: P(\theta) = \sum_{i=0}^{n} {\theta_i^j}^2, \quad (6)$$

where m_j is the j-th linear model for $\tilde{X}_j$ subset that corresponds to cluster l_j, $\left(\tilde{X}_j\right)_i \in R^n$ is the i-th vector of observations in j-th subset, and $\left(a_0^j, a_1^j, \dots, a_n^j\right)$ are the coefficients of j-th linear model and α is parameter.

In this study we also consider binary classification problem, for which both regularizations are applicable:

$$\sum_{i=1}^{s_j} \log\left(\tilde{p}\left(\left(\tilde{Y}_j\right)_i, \left(\tilde{X}_j\right)_i, \theta_j\right)\right) + \alpha P(\theta_j) = \max(\theta_j), \quad (7)$$

$$\tilde{p}\left(\left(\tilde{Y}_j\right)_i, \left(\tilde{X}_j\right)_i, \theta_j\right) = \begin{cases} \sigma\left(m_j\left(\left(\tilde{X}_j\right)_i, \theta_j\right)\right), \left(\tilde{Y}_j\right)_i = 0, \\ \sigma\left(-m_j\left(\left(\tilde{X}_j\right)_i, \theta_j\right)\right), \left(\tilde{Y}_j\right)_i = 1, \end{cases} \quad (8)$$

$$\sigma\left(m_j\left(\left(\tilde{X}_j\right)_i, \theta_j\right)\right) = \frac{1}{1 + e^{-m_j\left(\left(\tilde{X}_j\right)_i, \theta_j\right)}}, \quad (9)$$

When we solve the modeling problem, we split each subset on training, validation, and test sets. Training and validation subsets are used to determined α parameter via the grid search. We pick a trial α value, train the model on the training subset and then calculate the criterion on validation subset and after we check all the trial values, we pick the best α^* value in a sense of validation dataset criterion value. Then we use this α^* to train model on the union of the training and validation dataset and calculate its accuracy on the testing set.

We use "glmnet" R package (Friedman et al., 2016), where lasso, ridge and elastic net regressions are implemented.

2.3 Visualization

Once we have solved the modeling problem for each cluster, it is possible to reveal the statistics of it. First, one can observe the criterion value calculated for the testing set of that cluster and additionally criterion value based on training and validation set. If the model is linear, we can see the p-values and F static. Second, if we are satisfied with the accuracy of the model, we can find the most influential variables by the corresponding coefficients of the linear model and find out which coefficient cause negative or positive effect on the target variable. Third, we can observe the cluster description or cluster profile by its reference vector, vector of medians or mean values of its observations. This profile gives us information about what specifies this cluster. It could be high or low values of the variable.

Visualization of the results includes 3 plots: one with the SOM clusters 2-d plot, another one with selected cluster profile and the last one with coefficients of the linear model built for this cluster.

2.4 Predicting New Observations

When we receive the new observations to make predictions, we need to recognize to which cluster these observations belong to and then use the corresponding model to make a prediction.

Each cluster is represented by its reference vector:

$$V = \{v_1, \ \dots, \ v_{n_c}\}, \tag{10}$$

where $v_k \in R^q$, since we selected q variables for clustering (3).

Let us denote $x_{new} \in R^n$ as new observation and its selected variables (3) for clustering projection is denoted by $x_{new}^c \in R^q$. Now we compare this projection to the cluster reference vectors and determine which cluster is the closest:

$$i_c = \arg\min_i \|x_{new}^c - v_i\| \tag{11}$$

Once we determine the cluster to which projection x_{new}^c belongs, we can make the prediction using the model for specific cluster i_c:

$$y_{new} = m_{i_c}(x_{new}). \tag{12}$$

Criterion (11) is not the only option to determine the cluster, but this question is out of scope in this study. We can also see if there is no cluster close to the new point and realize that this kind of inputs combination is new to us.

3 Experimental Results

We applied approach to find the most influential factors of unwanted effects in a production line. To prevent leakage of commercial information, we rename the variables, and skip the analytical results that relates to the problem domain.

We have a dataset with 61 input variables and solve binary classification problem. Our first class is "good" production state, and our second class is "bad" one. Previously we cleaned the dataset and since observation rate is different for some variables, we modified it to the one we need. We made the standard normalization of the input data because approaches (5)-(6) and (7)-(9) and SOM requires that. When calculating regression or logistic regression with l_1 and l_2 regularization we split the train dataset on train and validation parts and keep 20% of data for validation. Then we use uniform grid on $G = G = [-5, 10]$ with 1000 points and try these values as exponential degrees for α in (5)-(6) and (7)-(9), in other words $\forall p \in G \Rightarrow \alpha = e^p$. Then we look for the best parameter by error on validation dataset and use it to train model on all the train data and after that check it on test dataset.

The next step was to determine the factors we use as the main ones for clustering. Since in the dataset we have sets of variables of different nature, we used one of those. Our choice was discussed with production experts. It is very important to receive a feedback from the production experts or business when selecting the inputs for clustering problem. Variables for clustering will be the first ones the production analyst or decision maker will use, when one needs to make a decision. It does not mean that these variables should be available in advance, but it should be available soon enough, so the decision maker will have explanation in time or not too late.

Once we selected variables (3), we solve the clustering problem and group the data according to the clustering labels (4). In this study we consider different number of clusters. We examined different combinations of clusters on horizontal and vertical axis: $(g_v^i, g_h^i), \forall i: 1 \le g_v^i, g_h^i \le 1, g_v^i \le g_h^i$ and $g_h^i = 1 \Leftrightarrow g_v^i \ne 1$, which means we try the following combinations: $1 \times 2, 1 \times 3, \dots, 1 \times 5, \dots, 2 \times 2, 2 \times 3, \dots, 5 \times 5$. For each of these parameters pair we solve the clustering problem and for each clustering problem solution, we produce the datasets and solve modeling problem.

When we have the combined clustering and models statistics, we can compare the clustering parameters by the total statistics and choose the best settings for considered problem. Let us compare overall train and

test ratings for different combinations of clusters on horizontal and vertical axis. The summary is given in Table 1. In this summary we calculate error rate mean value weighted by number of elements per cluster.

Table 1. Train and Test Error Rate by Number of Clusters

Model	*Train Accuracy*	*Test Accuracy*
No clusters	0.7833	0.7864
1x2	0.8324	0.7517
1x3	0.8371	0.7695
1x4	0.8629	0.7511
1x5	0.8675	0.8041
2x2	0.8582	0.7586
2x3	0.8985	0.8060
2x4	0.9300	0.8387
2x5	0.9599	0.8793
3x3	0.9432	0.8267
3x4	0.9913	**0.9116**
3x5	1.0000	**0.9205**
4x4	1.0000	0.8988
4x5	1.0000	0.9178
5x5	1.0000	0.9030

One can see that there are many combinations that outperforms approach without clustering and modeling by groups of data. One can also see that there are a few combinations which have high prediction rate on test data. Let us consider combination 3×5 as it has the best accuracy rate on test data. It is important to mention that 3×5 combination developed a cluster, which consists only of "good" class cases, so that its actual accuracy rate is higher. This must be considered when one chose the clustering parameters.

In this study we use the "kohonen" R package (Wehrens and Kruisselbrink, 2018) and make visualization with help of "ggplot2" R package (Wickham, 2016).

First, we can visualize clusters and color them according to the number of "bad" observations. In Figure 2 we can see the distribution of "bad" observations and their localization in particular clusters. One can use this information to reveal the relation between the clustering inputs values and pattern these values represent to the target variable. The same is possible for continuous output, for which we can use mean or median value.

Additionally, we can visualize the characteristics of each cluster by its statistics for specific variable or their combinations. For example, we can visualize the average sum of specific components by clusters, or we can see the distribution for a variable among clusters. The general profile or characteristics of the cluster will be described below.

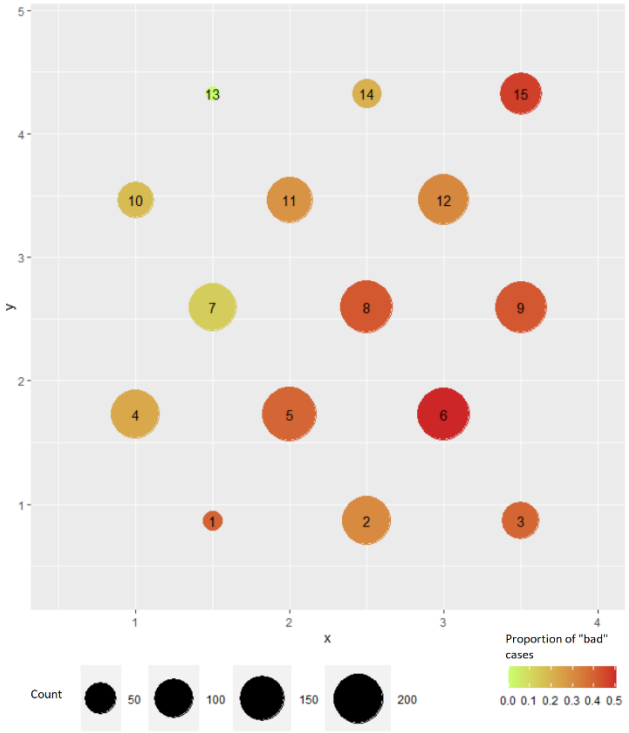

Figure 2. Percent of "bad" cases per cluster.

Second, we can visualize the error on training data for each model. In general, this plot can show us if there are some patterns for which we cannot apply the model we chose, or the data does not allow us to reveal the relation between the inputs and outputs.

Third, we can visualize accuracy on the testing data for each model. This plot is given in Figure 3.

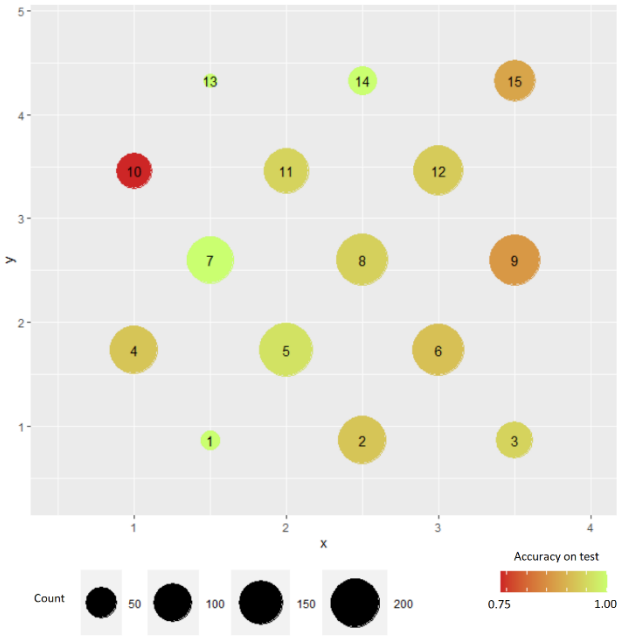

Figure 3. Accuracy on testing data per cluster.

Fourth, we can visualize the precision on the testing data for each model. This plot is given in Figure 4.

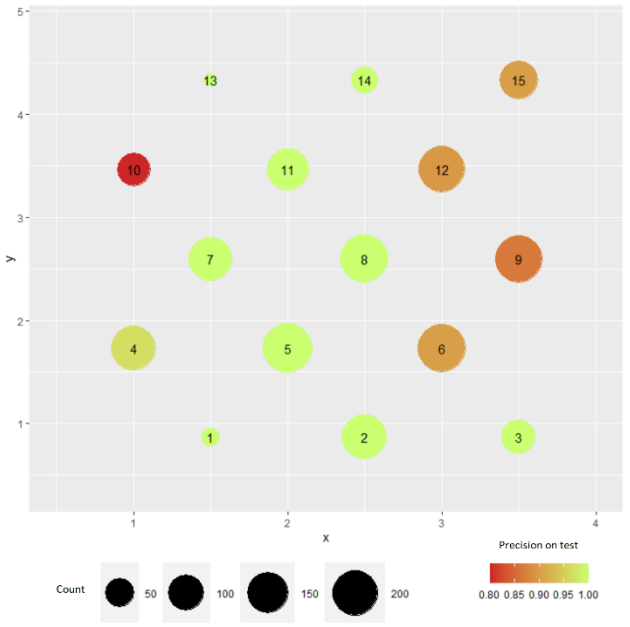

Figure 4. Precision on testing data per cluster.

The final two plots reveal what pattern can we be confident in, when predict the target variable or make any conclusion about relation between inputs and outputs. If the model metric on the test data is low, it means that we cannot be confident in any explanations given by the model that is built on the model of this cluster. For example, one can see that cluster 10 has lower accuracy and precision, comparing to other clusters.

Generally, we can add different statistical characteristics visualizations by clusters.

Now for each cluster one can see its profile and the most influential variables. In application we developed, it is possible to interactively choose the cluster of interest and observe its profile and linear model coefficients with help of R Shiny and "ggiraph" R package. Let us pick cluster "1", which is in left bottom corner in Figures 2-5. An example of cluster profile is given in Figure 5, and model's coefficients for that cluster is given in Figure 6.

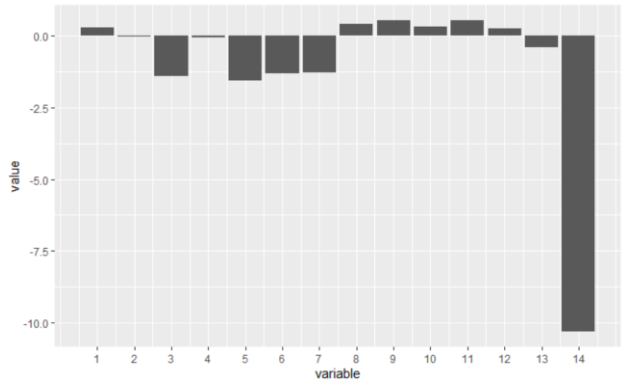

Figure 5. Cluster profile: cluster reference vector.

On profile figure we see the scaled data: 0 means average value for each variable and values above 0 correspond to cases, where these variables were greater than their average. If the value is smaller than 0, we know that this variable usually takes value that is smaller than average. As one can see this cluster can be described by 14th variable, because this variable is sufficiently smaller the mean value. One can also observe that variables 3, 5-7 are also smaller than the mean value. Expert can name this cluster according to its profile and variables nature.

Each cluster can be described by its characteristics. We can use mean, median values or any other metrics that help decision maker understand what each pattern represents. In this study we used reference vectors (10), since we utilize SOM. Reference vectors show values that characterize the cluster in a way, that if there is a new observation, we will compare the reference vector with that observation to make a decision (11) on what cluster does this observation belongs to (12).

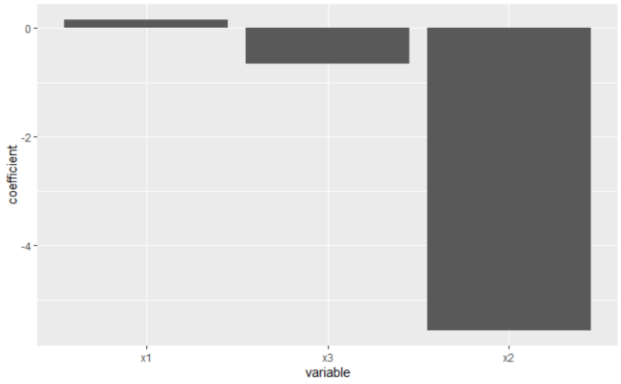

Figure 6. Linear model coefficients for selected cluster.

As one can see, lasso-regression keeps only 3 from 61 variables. Variables selected in model that corresponds to 1st cluster can be interpreted straightforward: increase of **x3** or **x2** lead to negative consequences and increase of **x1** lead to positive ones. Since we applied regularization we cannot calculate the $\boldsymbol{p}$ −value for any of the inputs coefficients the same way as one can do it for linear models without regularization. Nevertheless, one can apply linear modeling without regularization as the step 3, where we solve modeling problems for each data in cluster.

4 Conclusion

When solving modeling problem for business or production we are commonly interested in interpretability of the model. Interpretability lets decision makers and production experts understand the mechanics of the model prediction making. Sometimes this is necessary to validate the model, to be confident in it or to understand the process better. Data-driven modeling provides different view on the interaction between the inputs and outputs, which could reveal the unknown causality. Better understanding of the process is necessary, when one is looking for actions to improve

the process performance, to avoid unwanted states, and lessen production costs.

Proposed approach outperformed modeling without clustering and revealed the patterns that relates to the "bad" cases. We can observe it comparing error rate on train and test in Table 1 for model in the first row, which is built without clustering and any combined model. We can also observe that the model accuracy on train data is increasing with increase of clusters number. At the same time model accuracy on test data increase to some number of clusters. Because of that it is important to investigate what is the best combination of cluster numbers.

Powerful computational and visualization libraries in R along with R Shiny framework allows implementing analytical system, which can solve the combined clustering and modeling problem, reveal the dependencies and pattern and helps looking for actions to improve the process, when new observations appear.

Acknowledgements

This research is a part of the *AI-DA* projects, which is funded by Business Finland, European Regional Development Fund (ERDF), and seven companies.

References

Winston Chang, Joe Cheng, JJ Allaire, Carson Sievert, Barret Schloerke, Yihui Xie, Jeff Allen, Jonathan McPherson, Alan Dipert and Barbara Borges, shiny: Web Application Framework for R. R package version 1.6.0. https://CRAN.R-project.org/package=shiny. 2021

James Gareth, Daniela Witten, Trevor Hastie, Robert Tibshirani, *An Introduction to Statistical Learning*. New York, NY: Springer. 2013

David Gohel, Panagiotis Skintzos, ggiraph: Make 'ggplot2' Graphics Interactive. R package version 0.7.8. https://CRAN.R-project.org/package=ggiraph. 2020

Jerome Friedman, Trevor Hastie, Robert Tibshirani, Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, 33(1): 1-22. 2016. URL https://www.jstatsoft.org/v33/i01.

Teuvo Kohonen. *Self-Organizing Maps*. Springer, New York. doi:10.1007/ 978-3-642-97610-0. 2001

Max Kuhn, Kjell Johnson, *Applied predictive modeling*. Springer. 2016.

Gwo-Fong Lin, Tsung-Chun Wang, Lu-Hsien Chen, A Forecasting Approach Combining Self-Organizing Map with Support Vector Regression for Reservoir Inflow during Typhoon Periods, *Advances in Meteorology*: 1-12, 2016. https://doi.org/10.1155/2016/7575126

R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org. 2018

Ron Wehrens, Kruisselbrink, Flexible Self-Organizing Maps in kohonen 3.0. *Journal of Statistical Software*, 87(7): 1 - 18. doi : http://dx.doi.org/10.18637/jss.v087.i07. 2018

Hadley Wickham, *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016.