Applying data mining in the context of Industrial Internet

Oliviu Matei
Dept. of Electrical Engineering
North University of Baia Mare
Baia Mare, Romania
Email: oliviu.matei@cunbm.utcluj.ro

Kevin Nagorny
ATB-Bremen
Bremen, Germany
Email: nagorny@atb-bremen.de

Karsten Stoebener
KLÖCKNER DESMA Schuhmaschinen GmbH
Achim, Germany
Email: k.stoebener@desma.de

Abstract—**Nowadays, (industrial) companies invest more and more in connecting with their clients and machines deployed to the clients. Mining all collected data brings up several technical challenges, but doing it means getting a lot of insight useful for improving equipments. We define two approaches in mining the data in the context of Industrial Internet, applied to one of the leading companies in shoe production lines, but easily extendible to any producer. For each approach, various machine learning algorithms are applied along with a voting system. This leads to a robust model, easy to adapt for any machine.**

Keywords—*machine learning; data mining; k-nearest neighbour; neural network; support vector machine; rule induction.*

I. INTRODUCTION

According to [1], the access to the Internet has grown from an estimated 10 million people in 1993, to almost 40 million in 1995, to 670 million in 2002, and to 2.7 billion in 2013. The Internet has started to be used so much in industry, that a consortium has been founded, called Industrial Internet Consortium, which covers energy, healthcare, manufacturing, public sector and transportation (online at www.iiconsortium.org). According to the Industrial Internet Insights Report for 2015 [2], issued by General Electric and Accenture, the Industrial Internet can be described as a source of both operational efficiency and innovation that is the outcome of a compelling recipe of technology developments. However, the companies are not ready for predictive and innovative kinds of value-creating solutions, but 80% to 90% of the surveyed companies indicated that bug data analytic is in the top three priorities.

Therefore the article presents a specific application of data mining in the context of Industrial Internet. That means that some industrial equipments, wearing several sensors, are connected to the internet and the data can be analysed either locally or in a cloud. The target is to predict a specific behaviour based on the inputs from the sensors.

We treat the specific case of KLÖCKNER DESMA Schuhmaschinen GmbH (Germany), which is a leading vendor of machine systems for automated shoe production. An export rate of more than 95% results in a globally distributed customer and business partner network. Up-to-date, communication and information exchange with customers and business partners is mainly done in personal conversations almost inhibiting an efficient systematic analysis of customer feedback or product experience. By establishing an Industrial Internet connecting

customers, partners and products DESMA extends its leading position in the market but also targets to explore new business fields in data analysis and provision.

The article is structured as follows. After discussing the related work in section II, the general concepts of data mining in section III, which means applying machine learning (presented in section III-A) on big data. The specific algorithms used are described in sections III-A1 through III-A8, respectively III-B.The experimental setup and the results are detailed in sections IV and V.

II. RELATED WORK

The field of internet connectivity (along with Internet of Things and Industrial Internet) is at the start of the research because until recently there were no circumstances to do it (see [3], [4], [5]). As the Internet has deeply penetrated the domestic and industrial fields, more and more companies want to know how their equipments work. Therefore, in 2004, Olsson et al. [6] did fault diagnosis using case-based reasoning on sensor readings. A more statistical approach was developed by Giudici in [7]. One year later, Harding et al. publishes in [8] an overview about the possibilities of applying data mining in manufacturing. Many application are developed in the context of quality assurance in industry, such as the ones by Braha [9] and Koeksal [10]. Almagrabi et al. [11] publish in 2015 a very good survey on the quality prediction of product review.

III. DATA MINING PROCESS

The data mining process consists in collecting the data from a data source, preprocessing it and then applying a machine learning algorithm [12]. The inputs come from the sensors of a specific equipment, in our case direct soling machines, automated material flow with integrated robots (AMIR), robots and automatic laser processing cells. For more details, see section IV. The output is usually a desired outcome of the system (either an action of the equipment, or a warning etc.).

In our work, several machine learning algorithms have been applied, as described further. However, for improving the accuracy of the process, a voting system is also used. This means that the outputs of all algorithms vote for the most probable outcome. If it is a classification, the voting result is the class generated by most particular algorithms; if it is a regression, the global outcome is the average of each specific output. The strong point of this voting system is that

more approaches (algorithms) are used for the same purpose, therefore the result is expected to be more robust.

A. Machine learning algorithm

In our experiments we have used several algorithms along with their combinations. Only the ones with significant accuracy are presented in this paper, the other one being discarded because of the lack of interest and efficiency. Therefore, eight algorithms have some importance, such as:

- neural networks;
- Naive Bayes;
- support vector machine (SVM);
- fast large margin;
- k-nearest neighbour (k-NN);
- logistic regression;
- random forest;
- rule induction.

1) Neural networks: We apply the most common used model of neural networks, namely a multi-layer perceptron, which is a feed-forward neural network trained by a back propagation algorithm [13].

The learning is made using the back propagation algorithm, which is a supervised method that can be divided into two phases: propagation and weight update [14]. The two phases are repeated until the performance of the network is good enough. In back propagation algorithms, the output values are compared with the correct answer to compute the value of some predefined error-function. The error is then fed back through the network. Using this information, the algorithm adjusts the weights of each connection in order to reduce the value of the error function by some small amount. After repeating this process for a sufficiently large number of training cycles, the network will usually converge to some state where the error of the calculations is small [15].

For activation of the neurons, the sigmoid function is the most commonly used. Therefore, the values ranges of the attributes should be scaled to -1 and +1. This can be done through the normalize parameter. The type of the output node is sigmoid if the learning data describes a classification task and linear if the learning data describes a numerical regression task.

2) Naive Bayes: According to Lewis [16], a Naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem ([17]) with strong (naive) independence assumptions. In simple terms, a Naive Bayes classifier assumes that the presence (or absence) of a particular feature of a class (i.e. attribute) is unrelated to the presence (or absence) of any other feature.

The main advantage of the Naive Bayes classifier is that it requires a small amount of training data to estimate the means and variances of the variables necessary for classification. This is important because quite often the learning process is done on a very limited samples. Because independent variables are assumed, only the variances of the variables for each label need to be determined and not the entire covariance matrix.

3) Support Vector Machine: The support vector machine (SVM) is a fast algorithm with good results, which can be used for both regression and classification. Several kernel types include dot, radial, polynomial, neural, ANOVA, Epachnenikov, Gaussian combination and multiquadric [18].

An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

More formally [19], a support vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks. Intuitively, a good separation is done by the hyperplane that has the largest distance to the nearest training data points of any class. It is often the case that the sets to discriminate are not linearly separable in a lower dimensioanl space. For this reason, it was proposed that the original finite-dimensional space be mapped into a much higher-dimensional space, presumably making the separation easier in that space [20].

4) Fast Large Margin: Although the fast large margin provides results similar to those delivered by classical SVM or logistic regression implementations, this classifier, implemented as proposed by Fan et al in [21], is able to work on data set with millions of examples and attributes.

5) k-Nearest Neighbour: The k-Nearest Neighbor algorithm is based on learning by analogy, that is, by comparing a given test example with training examples that are similar to it [27]. The training examples are described by n attributes. Each example represents a point in an n-dimensional space. In this way, all of the training examples are stored in an n-dimensional pattern space. When given an unknown example, a k-nearest neighbor algorithm searches the pattern space for the k training examples that are closest to the unknown example. These k training examples are the k "nearest neighbors" of the unknown example. "Closeness" is defined in terms of a distance metric, such as the Euclidean distance [22].

The k-nearest neighbor algorithm is one the simplest of all machine learning algorithms: an example is classified by a majority vote of its neighbours, with the example being assigned to the class most common amongst its k nearest neighbours (k is a positive integer, typically small).

6) Logistic Regression: The logistic regression is based on the algorithm proposed by Keerthi et al. in [23]. The implementation uses the myKLR by Stefan Rueping [24]. Like most other SVM approaches, this one supports various kernel types including dot, radial, neural, anova, epachnenikov, Gaussian combination and multiquadric.

7) Random Forest: The Random Forest algorithm generates a set of a specified number of random tree models. The number of trees parameter specifies the required number of trees. The resulting model is a voting model of all the random trees [26].

According to Safavian and Landgrebe [25], the representation of the data in form of a tree has the advantage

compared with other approaches of being meaningful and easy to interpret. The goal is to create a classification model that predicts the value of a target attribute based on several input attributes of the training set. Each interior node of the tree corresponds to one of the input attributes. The number of edges of a nominal interior node is equal to the number of possible values of the corresponding input attribute. Outgoing edges of numerical attributes are labelled with disjoint ranges. Each leaf node represents a value of the label attribute given the values of the input attributes represented by the path from the root to the leaf.

Pruning is a technique in which leaf nodes that do not add to the discriminative power of the tree are removed [30]. This is done to convert an over-specific or over-fitted tree to a more general form in order to enhance its predictive power on unseen datasets. In other words, pruning helps generalization.

8) Rule Induction: The Rule Induction operator works similar to the propositional rule learner named 'Repeated Incremental Pruning to Produce Error Reduction' [28]. Starting with the less prevalent classes, the algorithm iteratively grows and prunes rules until there are no positive examples left or the error rate is greater than 50% [29]. In the growing phase, for each rule greedily conditions are added to the rule until it is perfect (i.e. 100% accurate). The procedure tries every possible value of each attribute and selects the condition with highest information gain. In the prune phase, for each rule any final sequences of the antecedents is pruned with the pruning metric $p/(p + n)$.

B. Voting

For a more accurate result, a voting algorithm has been also applied. It uses a majority vote of all other algorithms. The results with most votes is the output of this algorithm.

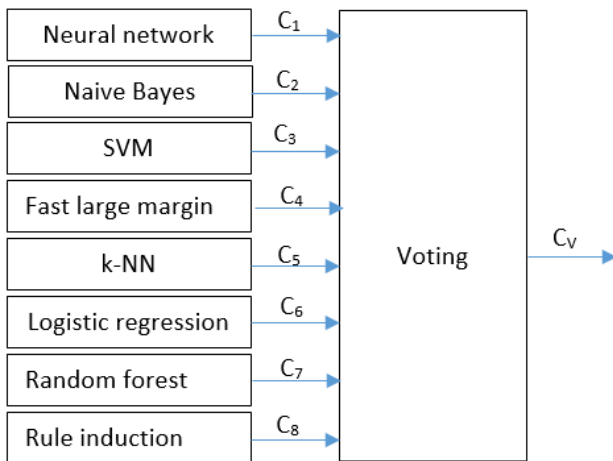The process of voting is depicted in figure 1.

Fig. 1: The voting process

The result C_v of the voting is the result of the most of the specific outputs $C_1, C_2, ..., C_8$.

IV. EXPERIMENTAL SETUP

The experiments have been carried out on a set of simulated samples provided by one of the world leading manufacturer of footwear production systems DESMA to validate the approach in a first step.

The data for the simulation is from following sensors which are representing a detailed status of a DESMA system or system component.

- `Status (on/off)` – showing the on/off status of a system or a system component;

- `Actor Positions` - represents the positions of actors inside of a system or a system module;

- `Speed` – is the configured system speed;

- `Revolution speed` - means the rotational speed of e.g. motors;

- `Duration time` - if the time of scheduled tasks;

- `Temperature` – shows material, ambient or system (component) temperatures;

- `Liquid pressure` - represents the pressure of liquid for the sole production;

- `Air pressure` - show the air pressure in the pipes.

The format of the data along with some sample values is described in table I.

These sensors are integrated in DESMA systems and system components as:

- Direct soling machines which directly injected the sole to the upper for single- or dual-density applications for the materials of rubber, polyurethan (PU), thermoplastic polyurethane (TPU) and other thermoplastics.

- Automated material flow with integrated robots (AMIR) systems which is an automation concept developed by DESMA used today in advanced footwear factories.

- Robots integrated in an automated production process

- Automatic Laser Processing cells which are roughing upper and offers new design opportunities in the direct soling technology.

Data mining results through (continuous) analyses of such sensor data of systems and system components as data base are useful to improve and optimizes company individual KPIs and are useful to improve extended system services as analysis, diagnosis and advanced monitoring of such systems.

A. Correlation matrix

The first important thing is to analyse the correlation between the attributes. Although this does not influence the behaviour of the algorithm, it is good to see if the inputs are dependent of each other. If so, some may be skipped. That would reduce the number of inputs, thus also the complexity of the system, increasing also the quality of the output.

As easily seen in table II, some fields are fully correlated, such as `Status (on/off)` with `Speed`, `Duration time` and `Status air pressure`. Other fields, such as `Status (on/off)` and `Revolution speed` or

TABLE I: A sample of the data set

Date	Status (on/off)	Actor Position	Speed	Revolution speed	Duration time	Temp.	Liquid pressure	Air pressure	Article number
10.9.15 10:17:01	0	0	0	0	0	25	0	0	6565
10.9.15 10:17:10	1	90	10	72	1	40	1	1	6565
10.9.15 10:18:04	1	50	10	74	1	41	10	1	6565

TABLE II: The correlation matrix

	Status (on/off)	Actor position	Speed	Revolution speed	Duration time	Tempe-rature	Liquid pressure	Air pressure
Status (on/off)	1	0.399	1	0.763	1	0.586	0.406	1
Actor position	0.399	1	0.399	0.408	0.399	0.436	0.125	0.399
Speed	1	0.399	1	0.763	1	0.586	0.406	1
Revolution speed	0.763	0.408	0.763	1	0.763	0.762	0.352	0.763
Duration time	1	0.399	1	0.763	1	0.586	0.406	1
Temperature	0.586	0.436	0.586	0.762	0.586	1	0.241	0.586
Liquid pressure	0.406	0.125	0.406	0.352	0.406	0.241	1	0.406
Air pressure	1	0.399	1	0.763	1	0.586	0.406	1

`Revolution speed` with `Speed` and `Duration time`, are significantly correlated.

However, the desired output, namely `Actor position`, has rather low correlation with any of the attributes, as its highest correlation factor is 0.408.

Such observations are a good starting point for a data mining process, because otherwise the process itself does not make much sense.

V. EXPERIMENTAL RESULTS

Two sets of experiments have been carried out:

- one determining the alert temperature, which means determining if the temperature is more than 50 degrees, described in section V-A.

- one determining the temperature trend. If the trend is ascending and the temperature is above 50 degrees, an alert is emitted (as presented in section V-B).

Of course, in both set-ups, the temperature is neither an input nor an output. In the former case, the value of the logical expression $Temperature > 50$ is the output, whereas in teh latter scenario, the trend of the temperature is the desired outcome of the data mining process.

A. Predicting the temperature

In this scenario, the temperature is the output of the data mining process.

The approaches with good outcomes are: neural networks, SVM, k-NN, logistic regression, random forest and rule induction. The accuracy for each of them is presented in table III. The first column displays the approach, the second column the accuracy and the third one displays whether the accuracy is above or below the average, within or out of the standard deviation. In both tables III and V, the notations are as follows:

- "(+)" if the accuracy is above the average and within the standard deviation ($accuracy \in [avg, avg + stdev]$);

TABLE III: Computational results for predicting the temperature

Method	Accuracy	
Neural nets	98.9%	(+)
SVM	96.7%	(+)
k-NN	98.9%	(+)
Logistic regression	7.8%	(−)
Random forest	95.6%	(+)
Rule induction	92.2%	(+)
Average	81.68%	
Stdev.	33.11%	

- "(++)" if the accuracy is above the average and outside the standard deviation ($acccuracy > avg + stdev$);

- "(-)" if the accuracy is below the average and within the standard deviation ($accuracy \in [avg - stdev, avg)$);

- "(–)" if the accuracy is below the average and outside the standard deviation ($accuracy < avg - stdev$);

There are two important remarks related to the results:

- most of the results are very good (above 92%), which means that determining the temperature is very accurate and reliable;

- the accuracy in the case of logistic regression is extremely low (below 8%).

As the accuracies are so high, no further enhancements are considered.

B. Predicting the trend

In this scenario, the temperature can be measured, but it is not an input of the data mining process. The outcome is the trend of the temperature, which, corroborated with the value of the temperature will raise an alert if the temperature is above 50 degrees and the trend is upwards. The dataset looks like in table IV, where `Temp. trend` is the trend of the temperature to be predicted.

The eight approaches led to the following results regarding the accuracy of classification:

TABLE IV: A sample of the data set when predicting the temperature trend

Date	Status (on/off)	Actor Position	Speed	Revolution speed	Duration time	**Temp. trend**	Liquid pressure	Air pressure	Article number
10.9.15 10:17:01	0	0	0	0	0	**up**	0	0	6565
10.9.15 10:17:10	1	90	10	72	1	**up**	1	1	6565
10.9.15 10:18:04	1	50	10	74	1	**up**	10	1	6565

TABLE V: Computational results

Method	Accuracy		Optimized accuracy		Increase
Neural nets	52.9%	(+)	57.67%	(+)	9.02%
Naive Bayes	55.1%	(+)	62.14%	(+)	12.78%
SVM	56.9%	(++)	58.67%	(+)	3.11%
Fast large margin	51.5%	(-)	51.56%	(-)	0.12%
k-NN	48.2%	(-)	51.56%	(-)	6.97%
Logistic regression	57.1%	(++)	66.89%	(++)	17.15%
Random forest	48.6%	(–)	58.67%	(+)	20.72%
Rule induction	51.8%	(-)	46.56%	(–)	-10.12%
Average	52.76%		56.72%		7.49%
Stdev.	3.42%		6.52%		

TABLE VI: The best parameters

Algorithm	Parameter	Value
Neural network	hidden layers	0
	training epochs	500
	learning rate	0.64
	momentum	0.0
Naive Bayes	estimation method	greedy
	number of kernels	10
	bandwidth	0.2
	minimum bandwidth	2.4
	number of kernels	10
SVM	kernel type	rbf
	γ	19.0
	C	0.0
Fast large margin	ϵ	0.001
k-NN	k	1
	numerical measure	correlation similarity
Logistic regression	kernel	radial
	kernel γ	5.0
	C	4.7
Random forest	number of trees	100
	maximal depth	46
	minimal leaf size	4
Rule induction	criterion	accuracy
	sample ration	0.9
	pureness	0.9
	minimal prune benefit	0.25

In table V, the first column represents the machine learning algorithm used for data mining. The second column displays the accuracy of each algorithm and then their relative performance. The results marked with "(+)" are above the average, and the ones marked with "(++)" are above the average plus the standard deviation. Pairwise, the results marked with "(-)" are below the average and the ones marked with "(–)" are below the average minus the standard deviation. The next column represents the accuracy of each approach, after a parameters optimization process, along with the relative performance. The last column shows the improvements brought by the optimization, which is, in average, 7.49%. To be noticed that only one algorithm namely `Rule induction`, reduced its accuracy (with 10.12%), while all the others improved up to 17.15% in the case of `logistic regression` and 20.72% for `random forest`.

Determining the trend did not lead to impressive levels of accuracy and this is somehow disappointing. This is because of the low correlation between the input attributes and the temperature trend. In other words, we could not find a significant correlation between the trend and the other attributes.

1) Best parameters: It is of importance to show what parameters of the learners lead to the best results. Therefore they are described in table VI.

2) Voting results: Voting for between all previous data mining techniques, brings the accuracy of the system 59.6%, which is less than the best two techniques running separately, namely Naive Bayes (with an accuracy of 62.14%) and logistic regression (with an accuracy of 66.69%). Even the optimized techniques with respect to parameters has not brought any improvement. However, voting accuracy is slightly higher than the average accuracy of the six methods (neural network, Naive Bayes, SVM, fast large margin, k-NN and logistic regression), which is 58.081%.

Voting all eight methods lead to an accuracy of 51.4%, which is much less than the average accuracy of each separate technique of 56.715%.

The very interesting thing is that removing the poorest quality method, which is rule induction (accuracy of 46.56%), the voting approach leads to an accuracy of 46.8%, thus even lower than using also rule induction.

VI. CONCLUSIONS

This article presents a data mining approach in the context of Industrial Internet. The specific studied case is a KLÖCKNER DESMA Schuhmaschinen GmbH, which is the leader company in shoe production lines. Their machines are equipped with several sensors and the collected data can be transferred into the cloud. Analysing the data thoroughly brings a lot of insight and useful knowledge about the usage of the machines, which can be further utilized for predictive maintenance and for improving the user experience.

The achieved accuracy is neither very high, nor very low. However, the experimental data set was pretty limited and the scope of the research was to achieve a robust and very easily extendible model. In this case, the accuracy will depend on the input/output data set. Therefore, the level of accuracy is not the most dominant here, if it is above some certain reasonable thresholds.

The implemented model and very robust and can be easily extended to other companies collecting data from their machines. We also show that two different approaches for

mining the same data set, lead to completely different results and accuracies.

References

[1] Brahima Sanou, "ITC Facts and Figures 2013", Telecommunication Development Bureau, International Telecommunications Union (ITU), Geneva, February 2013. Retrieved 23 May 2015.

[2] *, Industrial Internet Insights Report for 2015, General Electric, 2014.

[3] Hargittai, Eszter. "Weaving the Western Web: Explaining differences in Internet connectivity among OECD countries." Telecommunications policy 23.10 (1999): 701-718.

[4] Cohen, Reuven, et al. "Resilience of the Internet to random breakdowns." Physical review letters 85.21 (2000): 4626.

[5] Imane BOUIJ-PASQUIER, Anas ABOU EL KALAM and Abdellah AIT OUAHMAN, SmartOrBAC International Journal of Advanced Computer Science and Applications(IJACSA), 6(11), 2015.

[6] OLSSON, Erik; FUNK, Peter; XIONG, Ning. Fault diagnosis in industry using sensor readings and case-based reasoning. Journal of Intelligent & Fuzzy Systems, 2004, 15.1: 41-46.

[7] Giudici, Paolo. Applied data mining: statistical methods for business and industry. John Wiley & Sons, 2005.

[8] Harding, J. A., M. Shahbaz, and A. Kusiak. "Data mining in manufacturing: a review." Journal of Manufacturing Science and Engineering 128.4 (2006): 969-976.

[9] Braha, Dan, and Armin Shmilovici. "Data mining for improving a cleaning process in the semiconductor industry." Semiconductor Manufacturing, IEEE Transactions on 15.1 (2002): 91-101.

[10] Koeksal, Guelser, nci Batmaz, and Murat Caner Testik. "A review of data mining applications for quality improvement in manufacturing industry." Expert systems with Applications 38.10 (2011): 13448-13467.

[11] H.Almagrabi , A. Malibari and J. McNaught, A Survey of Quality Prediction of Product Reviews International Journal of Advanced Computer Science and Applications(IJACSA), 6(11), 2015.

[12] Larose, Daniel T. Discovering knowledge in data: an introduction to data mining. John Wiley & Sons, 2014.

[13] Pal, Sankar K.; Mintra, Sushmita. Multilayer perceptron, fuzzy sets, and classification. IEEE Transactions on Neural Networks, 1992, 3.5: 683-697.

[14] Hecht-Nielsen, Robert. Theory of the backpropagation neural network. In: Neural Networks, 1989. IJCNN., International Joint Conference on. IEEE, 1989. p. 593-605.

[15] Yadav, Amit Kumar, Hasmat Malik, and S. S. Chandel. "Application of rapid miner in ANN based prediction of solar radiation for assessment of solar energy resource potential of 76 sites in Northwestern India." Renewable and Sustainable Energy Reviews 52 (2015): 1093-1106.

[16] Lewis, David D. Naive (Bayes) at forty: The independence assumption in information retrieval. In: Machine learning: ECML-98. Springer Berlin Heidelberg, 1998. p. 4-15.

[17] D'Agostini, Giulio. A multidimensional unfolding method based on Bayes' theorem. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 1995, 362.2: 487-498.

[18] Smola, Alex J.; Schoelkopf, Bernhard. A tutorial on support vector regression. Statistics and computing, 2004, 14.3: 199-222.

[19] Burges, Christopher JC. A tutorial on support vector machines for pattern recognition. Data mining and knowledge discovery, 1998, 2.2: 121-167.

[20] Cortes, Corinna; Vapnik, Vladimir. Support-vector networks. Machine learning, 1995, 20.3: 273-297.

[21] Fan, Rong-En, et al. LIBLINEAR: A library for large linear classification. The Journal of Machine Learning Research, 2008, 9: 1871-1874.

[22] Weinberger, Kilian Q.; Blitzer, John; Saul, Lawrence K. Distance metric learning for large margin nearest neighbor classification. In: Advances in neural information processing systems. 2005. p. 1473-1480.

[23] Keerthi, S. Sathiya, et al. A fast dual algorithm for kernel logistic regression. Machine learning, 2005, 61.1-3: 151-165.

[24] Rueping, Stefan. mySVM-Manual. University of Dortmund, Lehrstuhl Informatik, 2000, 8: 1207-1245.

[25] Safavian, S. Rasoul; Landgrebe, David. A survey of decision tree classifier methodology. IEEE transactions on systems, man, and cybernetics, 1991, 21.3: 660-674.

[26] Svetnik, Vladimir, et al. Random forest: a classification and regression tool for compound classification and QSAR modeling. Journal of chemical information and computer sciences, 2003, 43.6: 1947-1958.

[27] Beyer, Kevin, et al. When is "nearest neighbor" meaningful?. In: Database Theory(ICDT99). Springer Berlin Heidelberg, 1999. p. 217-235.

[28] Cohen, William W. Fast effective rule induction. In: Proceedings of the twelfth international conference on machine learning. 1995. p. 115-123.

[29] Langley, Pat; Simon, Herbert A. Applications of machine learning and rule induction. Communications of the ACM, 1995, 38.11: 54-64.

[30] Mingers, John. An empirical comparison of pruning methods for decision tree induction. Machine learning, 1989, 4.2: 227-243.