Association for Information Systems
AIS Electronic Library (AISeL)

BLED 2016 Proceedings | BLED Proceedings

2016

Forecasting terminal call rate with machine learning methods

Davorin Kofjač
University of Maribor, Faculty of Organizational Sciences, Slovenia, davorin.kofjac@fov.uni-mb.si

Andrej Škraba
University of Maribor, Faculty of Organizational Sciences, Slovenia, andrej.skraba@fov.uni-mb.si

Amira Mujanović
University of Maribor, Faculty of Organizational Sciences, Slovenia, amira.mujanovic@student.um.si

Gorenje gospodinjski aparati d.d., Slovenia

29th Bled eConference

Digital Economy

June 19 - 22, 2016; Bled, Slovenia

Forecasting terminal call rate with machine learning methods

Davorin Kofjač

University of Maribor, Faculty of Organizational Sciences, Slovenia

davorin.kofjac@fov.uni-mb.si

Andrej Škraba

University of Maribor, Faculty of Organizational Sciences, Slovenia

andrej.skraba@fov.uni-mb.si

Amira Mujanović

University of Maribor, Faculty of Organizational Sciences, Slovenia

amira.mujanovic@student.um.si

Aleš Brglez

Gorenje gospodinjski aparati d.d., Slovenia

ales.brglez@gorenje.com

Abstract

This paper deals with the development of a model to predict the products' terminal call rate (TCR) during the warranty period. TCR represents a key information for a quality management department to reserve the necessary funds for product repair during the warranty period. TCR prediction is often carried out by parametric models such as Poisson processes, ARIMA models and maximum likelihood estimation. Little research has been done with machine learning methods (MLM). Therefore, this paper addresses the utilization of machine learning methods (MLM), such as regression trees, ensembles of regression trees and neural networks in order to estimate the parameters of different models for TCR prediction. MLM were tested on exponential and logistic non-linear models, which best describe the shape of the cumulative density function of the failed products. The estimated cumulative density function was used to predict the TCR. The results have shown the ensembles of regression trees yield the smallest TCR prediction error among the tested MLM methods.

Keywords: product failure, prediction, machine learning, regression trees, neural networks, ensembles

1 Introduction

All production companies are faced with the problem of failure of products and the provision of opportunities to repair those products within the warranty period. The warranty provides security to customers in the event of early products' failure that occur during the warranty period and represents a contract between the buyer and the manufacturer, which comes into effect when a customer has purchased the product. The purpose of the warranty is basically to ensure accountability in the case of premature failure of the product, where the failure is considered as the inability of the product to perform its function (Kim et al., 2004). Offering warranty causes additional costs to the manufacturer. As a result, the manufacturer wants to minimize that cost (Blischke and Murthy, 1994; Zuo et al., 2000). Therefore, predicting the proportion of products' failures during the warranty period is crucial for determining this cost.

In order to minimize the aforementioned cost, the manufacturers have different ways of predicting failure of products, such as lifetime distributions, stochastic processes, artificial neural networks, Kalman filters and time series models (Wu, 2012). Nonparametric approaches, such as neural networks, have also been applied to predict product failures. Wasserman and Sudjianto (1992, 1996) have implemented a neural network (multilayer perceptron, MLP) to predict warranty claims, which is able to accommodate non-linearities in the data when it cannot be adequately fitted with a low-order polynomial. Rai and Singh (2005) used a special type of neural network, i.e. radial basis function (RBF), in order to research the forecast warranty performance in the presence of the maturing data. Grabert et al. (2005) developed an early detection system using neural networks and probability distribution estimation. However, those approaches do not consider the fact that warranty claims reported in the recent months might be more important in forecasting future warranty claims than those reported in the earlier months, and they are developed on the basis of repair rates, which can cause information loss through such an arithmetic-mean operation (Wu, 2012). Therefore, Wu and Akbarov (2011) have proposed a weighted support vector regression (SVR) model and a weighted SVR-based time series model, which show a better performance compared with that of MLP and RBF as well as with ordinary SVR models.

By examining the literature, we can conclude that machine learning methods (MLM) have been used scarcely in the area of product failure prediction. Therefore, our goal was to investigate, whether it is possible to use the MLM to make a quality prediction of non-linear model parameters, which describe the cumulative failure rate of a product batch in a form of a cumulative density function (CDF). The CDF is then used to predict the terminal call rate (TCR). Such approach presents a novelty in this research area.

The rest of the paper is structured as follows. In Section 2 the literature review is provided as well as the methodology used to predict the product failures. Section 3 presents the machine learning prediction results. Finally, Section 4 gives the conclusions and the guidelines for future research.

2 Problem formulation and methodology

Manufacturer produces products in batches. Usually, batches are a monthly aggregate of a production of a certain product or a product family. When the failed product can be traced back

to a specific production batch, we are able to convert such data into a so-called layer cake format. This format combines produced and failed products on a monthly basis, as presented in Table 1. The data are aggregated as an upper triangular matrix with diagonal. Such a data representation allows for the implementation of best-fit approaches (Kleyner & Sandborn, 2005). The data in Table 1 shows that we can collect 6 data points about the failures, which can be used as an input for prediction models.

Month	Num. of produced products	Number of failures per month		
		Month 1	Month 2	Month 3
1	3256	43	87	120
2	3590		78	101
3	3478			66

Table 1: An example of a layer cake

The failure process is usually described with the following well-known distributions: Weibull, exponential, normal, and log-normal (Kleyner & Sandborn, 2005; Hall & Strut, 2003, Xie & Lai, 1995). In our case, the time of production of a certain batch is known as well as the time of product failure. Hence, we can model this process with the following probability density function (PDF, right axis) and its resulting cumulative density function (CDF, left axis), as presented in Figure 1.

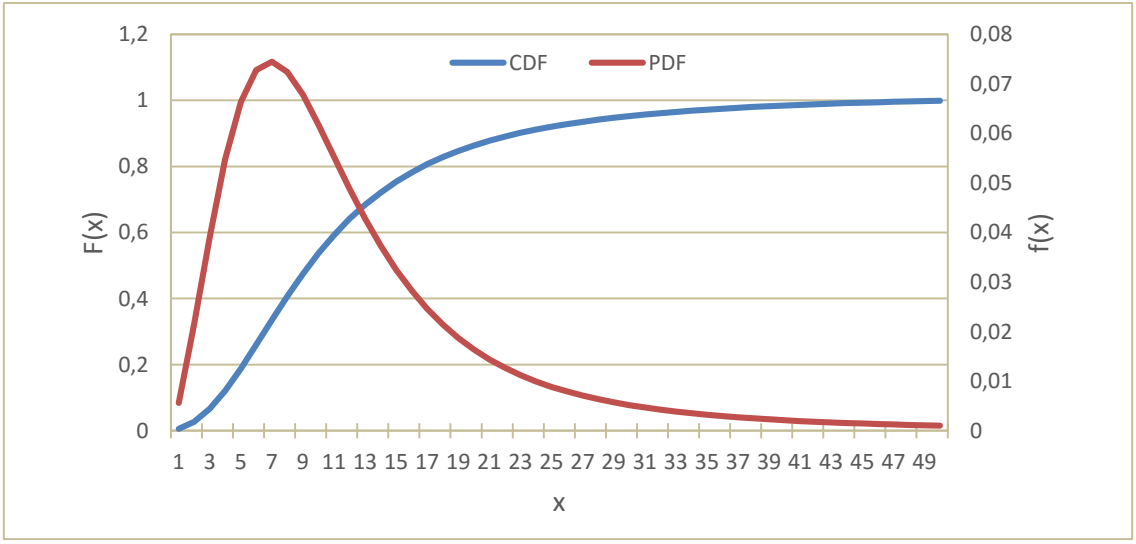

Figure 1: An example probability and cumulative density functions

An expected maximum value of the failed products within the warranty period is called the terminal call rate (TCR). To estimate the TCR, a cumulative failure rate over a specific period of time, e.g. a month, for a specific product batch must be known (Kofjač et al., 2014).

2.1 Curve fitting

In order to predict the TCR for a specific batch, its CDF must be determined. Hence, our goal is to estimate a real-valued variable $y \in \mathbb{R}$ given a pattern x (Smola and Vishwanathan, 2008).

The example is presented in Figure 2, where we are given a number of observations (presented by black dots) and we would like to estimate the function f which maps the observations X to such that $f(x)$ is as close to the observed values as possible.

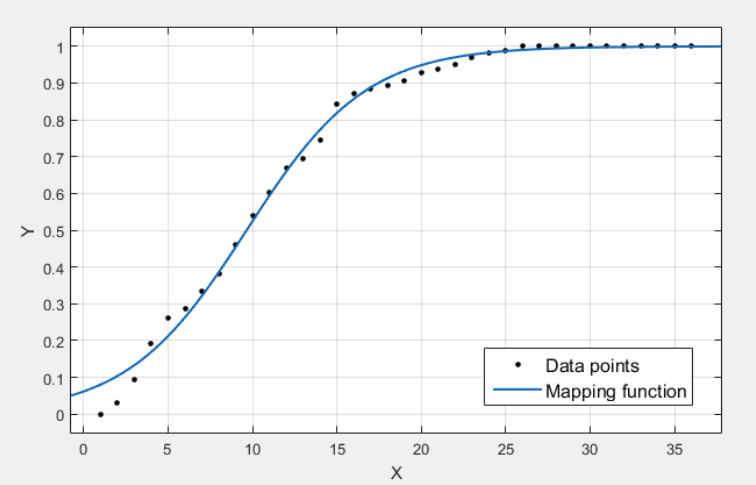

Figure 2: An example of regression estimation

The example in Figure 2 represents the CDF of a product batch failure rate (marked with dependent variable Y) over time in months (marked with independent variable X), normalized to the interval [0, 1]. Such a CDF can be estimated by using non-linear models, such as exponential and logistic. The logistic model is generally given by:

$$f(x) = \frac{a}{1+be^{-cx}} \tag{1}$$

while the exponential model is generally given by:

$$f(x) = e^{a+\frac{b}{x}} \tag{2}$$

To evaluate the goodness of fit of the abovementioned non-linear models, we have used standard error of the estimate (SEE), which measures the average distance of a non-linear model from the observations, and is given by:

$$SEE = \sqrt{\frac{\sum(f(i)-f(i))^2}{n-p}} \tag{3}$$

where $f(i)$ are the observations, $f(i)$ the predictions, n the number of observations and p the number of parameters (independent variables) in the non-linear model. Although R^2 and R^2_{adj} are frequently used in evaluation of non-linear regression models, Spies and Neumeyer (2010) argue that these measures do not adequately reflect the goodness of fit. Therefore, those measures were not used in our research.

Finally, the TCR prediction accuracy is measured by mean absolute percentage error, which is given by:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{\left|f(i)-f(i)\right|}{f(i)} \tag{3}$$

Manufacturer has provided us with a database of 313 product batches (instances). The database contained the produced batch quantities and their respective number of failures per month. On this basis we were able to calculate the 313 CDFs of product failures and normalize them to the interval [0,1]. An example of three CDFs is shown in Figure 3. In the next step, we have used the Matlab Curve Fitting Toolbox to obtain the values for parameters a, b, and c for the logistic model (Eq. 1) and the values for parameters a and b for the exponential model (Eq. 2). The parameter values were obtained for each instance. An example of the obtained parameters for both non-linear models is presented in Table 2.

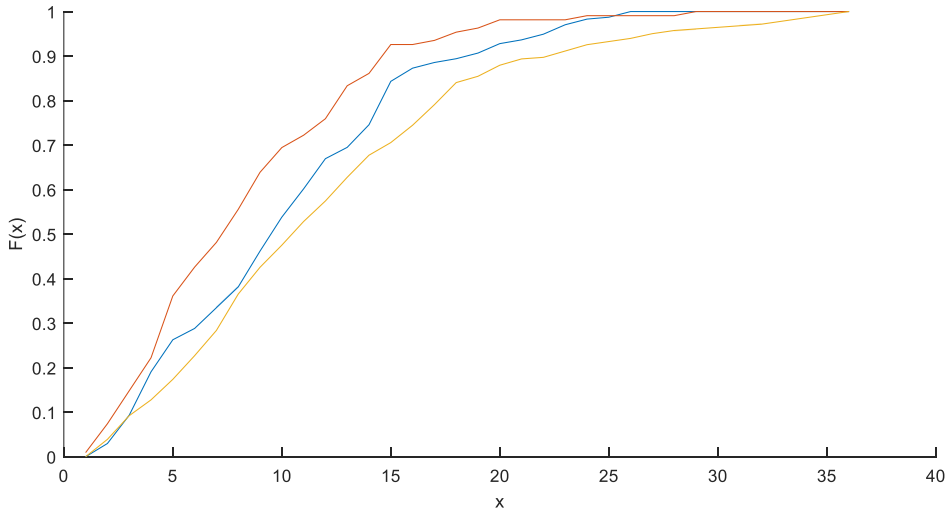

Figure 3: Examples of three different cumulative density functions, which represent a cumulative failure rate of three different product batches

Instance	Logistic model			Exponential model	
	a	b	c	a	b
1	1.309	0.839	0.092	0.304	-8.559
2	1.274	0.850	0.091	0.285	-8.438
3	1.318	0.827	0.092	0.312	-8.914
4	1.358	0.817	0.086	0.338	-9.741
5	1.323	0.827	0.088	0.316	-9.272
…	…	…	…	…	…

Table 2: An example of the obtained parameters per instance for both non-linear models

Next, the parameters obtained in the previous step were used to simulate 313 CDFs with both, the logistic and the exponential model. The simulated CDFs were compared them to the actual CDFs. The result of this modelling process was the evaluation measure SEE, for all instances and for both non-linear models. The average, minimum and maximum values and standard deviation for SEE are presented in Table 3. We can observe low average *SEE* values (with regard to the

interval [0,1]), indicating that both non-linear models provide a quality fit. However, the logistic model provides better results with respect to the exponential model. It yields lower average, minimum and maximum values than the exponential model. Further, by observing the histogram of SEE values in Figure 4, one can notice that logistic model SEE values are less dispersed, indicating a more robust model.

Measure	Non-linear model	
	Exponential	Logistic
Avg SEE	0.036	0.024
St.dev. SEE	0.015	0.010
Min SEE	0.012	0.008
Max SEE	0.140	0.123

Table 3: Initial goodness of fit evaluation for the exponential and the logistic models

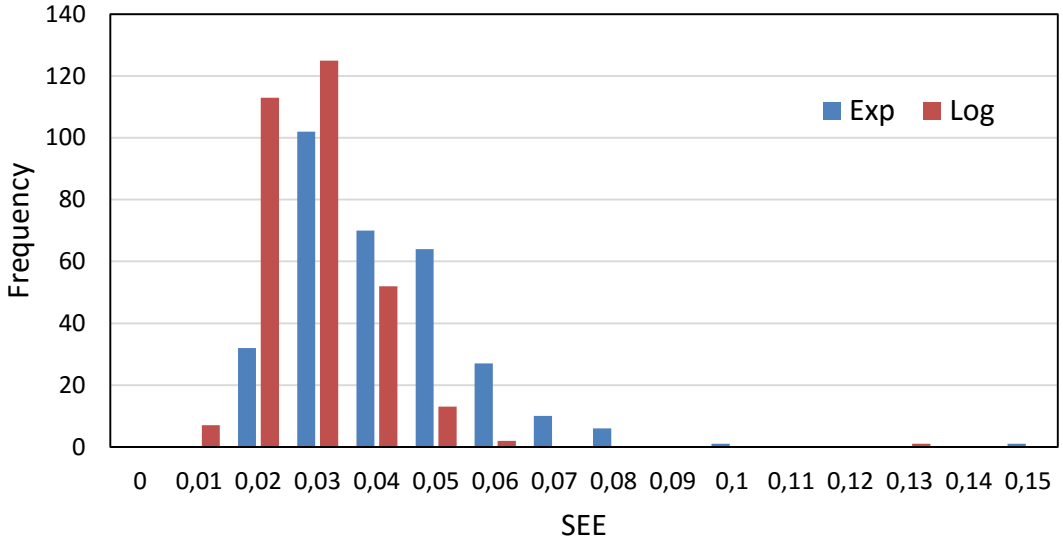

Figure 4: A histogram of SEE values in the initial goodness of fit evaluation for both non-linear models

2.2 Machine learning

By determining the parameter values for non-linear models we have obtained the instances, which can be used with MLM. Machine learning is a branch of artificial intelligence dealing with the development of techniques and methods that allow the learning of machines. Machine learning is used due to the two major reasons: some tasks are too complex to program and the need for adaptivity. First, several tasks are performed by humans routinely, yet we do not know how to sufficiently elaborate them to make an algorithm, such as driving, speech recognition, etc. Second, many tasks change over time or from one user to another. Therefore, it would be hard and inefficient to constantly change the algorithm “by hand” (Shalev-Schwartz and Ben-David, 2014).

Machine learning can tackle many problems, such as binary classification, multiclass classification, structured estimation, regression, novelty detection, etc. (Smola and

Vishwanathan, 2008). In our case, we use the regression estimation, where the goal is to estimate a real-valued variable TCR given an input pattern of cumulative failure rates.

Machine learning methods must be trained (taught) in order to be used effectively. They are able to learn in several ways. The most common classification of learning techniques is the supervised and unsupervised learning (Shalev-Schwartz and Ben-David, 2014). In supervised learning, it is assumed that the learning output is known in advance, while in unsupervised learning this is not the case. Supervised learning usually includes neural networks, decision trees, etc., while unsupervised learning includes different types of clustering, self-organizing maps, etc. (Smola and Vishwanathan, 2008).

In our case, the output is known in advance; we have obtained the values for parameters a, b (and c), which represent the desired MLM output values. Hence, we are able to use the supervised learning to train the MLM.

2.3 TCR prediction model and training

In this section we will present the TCR prediction model shown in Figure 5. The input into the model are the values r_1, r_2, …, r_n, which represent the cumulative failure rates for the first n consecutive months since the production of a batch. The selected MLM then estimates the parameters a, b (and c). These parameters are used to calculate the CDF, which should adequately represent the cumulative failure rate of the selected batch. Finally, on the basis of the calculated CDF, we are able to predict the TCR.

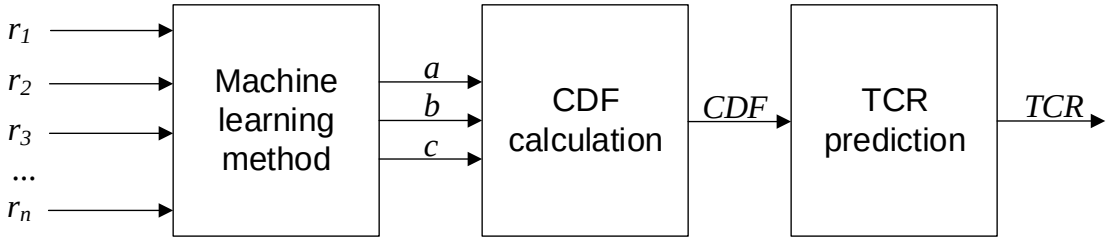

Figure 5: TCR prediction model

In order to train the MLM with supervised learning, we have to provide them with the inputs and the desired outputs. The input into the MLM are the partial CDFs, for example, the cumulative product failure rates for the first n months since the end of production of a particular product batch, where $n \in \{3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36\}$. For example, if $n = 3$ then the input into the MLM are the three values $\{0,\ 0.029,\ 0.093\}$. The desired output for the supervised learning are the estimated parameters a, b and c, which were obtained in the previous step, for example $\{1.309,\ 0.839,\ 0.092\}$. An example of this input-output mapping is presented in Table 4, where the partial cumulative product failure rates for the first 3 months (Mx denotes the month x) were taken as the input, while the parameters a, b and c were estimated for the logistic model.

	Input			Desired output		
Instance	M1	M2	M3	a	b	c
1	0	0.029	0.093	1.309	0.839	0.092

2	0	0.047	0.129	1.274	0.850	0.091
3	0.013	0.049	0.101	1.318	0.827	0.092
4	0	0.017	0.046	1.358	0.817	0.086
5	0	0.025	0.075	1.323	0.827	0.088
…	…	…	…	…	…	…

Table 4: Input and desired output data for machine learning methods

3 Results

To estimate the parameters for non-linear models, we have used regression tree, feedforward neural network and ensemble of regression trees. The MLM were tested in Matlab environment with the setup presented in Table 5. This setup has yielded the best fitting results in the preliminary study.

Regression tree	Type of tree: Binary Split criterion: MSE Number of folds in cross-validation: 10. Minimum instances in a leaf: 3 Minimum number of instances in a parent: 10
Neural network	Learning method: Levenberg-Marquardt backpropagation Max. number of epochs: 1000 Error goal: 0 Minimum gradient: 1e-7 Initial combination parameter μ value: 0.001 Decrease factor of μ: 0.1 Increase factor of μ: 10 Maximum μ value: 1e10 1 hidden layer with different number of neurons: 1, 5 and 10
Regression tree ensemble	The same parameters as with Regression tree. We have tested three different ensembles with 10, 50 and 100 trees, by using Tree Bagging.

Table 5: Experiment setup for machine learning methods

To train the MLM we have used the method of repeated random sub - sampling validation (RRSSV), also known as Monte Carlo cross-validation (MCCV) (Remesan and Mathew, 2015). We have run 10 iterations of MMCV. During each run the instances were divided into the training set (70%) and the test set (30%).

The results of testing the MLM are presented in Table 6. With obtained parameters a, b (and c) we have calculated the logistic or exponential CDFs for each instance in the test set and then evaluated the goodness of fit with the actual data by the SEE. The values in Table 6 represent the mean value and standard deviation across the 10 iterations of the MMCV method and $n \in \{3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36\}$.

Method's name in Table 6 consists of the following parts, separated by a comma:

- Type of machine learning method (NN – neural network, RT – regression tree, Ensemble – ensemble of regression trees).

- Model (Exp – exponential, Log – logistic).
- The parameter specific for the machine learning method - number of regression trees in an ensemble or the number of neurons in a hidden layer for neural network.

The results in Table 6 are sorted by the mean value of SEE in an ascending order. The best CDF fit results yielded ensembles of regression trees with exponential model, by achieving the lowest SEE values (0.04) and the lowest SEE standard deviation (0.01), indicating a robust production model. The worst SEE score was achieved by a regression tree with logistic model with the mean value of 0.06.

Method	Mean	SD
Ensemble, Exp, 100	0.04	0.01
Ensemble, Exp, 50	0.04	0.01
Ensemble, Exp, 10	0.04	0.01
NN, Exp, 5	0.04	0.01
NN, Exp, 10	0.05	0.02
Ensemble, Log, 50	0.05	0.01
NN, Log, 10	0.05	0.01
Ensemble, Log, 100	0.05	0.01
RT, Exp	0.05	0.01
NN, Log, 5	0.05	0.01
Ensemble, Log, 10	0.05	0.01
NN, Exp, 1	0.05	0.02
RT, Log	0.06	0.01

Table 6: Mean values and standard deviation for SEE per machine learning method

We have conducted further research on how the number of input data into the ML method affects SEE. For this test, we have selected only ensembles of regression trees with exponential model which have yielded the best results in the previous step. The test was performed with ensembles containing 10, 50 and 100 trees. The results are presented in Figure 6. As expected, one can notice that the SEE decreases for all ensembles with the increased number of input data. Obviously, the more input data you provide, the better the prediction result. Second, ensembles with higher number of trees yield better SEE values. The largest difference, approximately 0.005 among ensembles is with the lowest number of inputs. The more input data we provide, the lesser this difference, approximately 0.001, which is almost negligible.

In the final test, we have researched the TCR prediction accuracy of the ensembles of trees. The results are shown in Figure 7. Similar to the experiment setup in the previous step, the test was performed with ensembles of regression trees, containing 10, 50 and 100 trees, together with the exponential model. The TCR prediction accuracy was measured with mean absolute percentage error ($MAPE$) against the actual TCR. The average $MAPE$ values for (Exp, 10), (Exp, 50) and (Exp, 100) models are 2.46%, 2.34% and 2.31%, respectively. As expected, (Exp, 100) model yielded the lowest $MAPE$ values, therefore being the most suitable for TCR prediction.

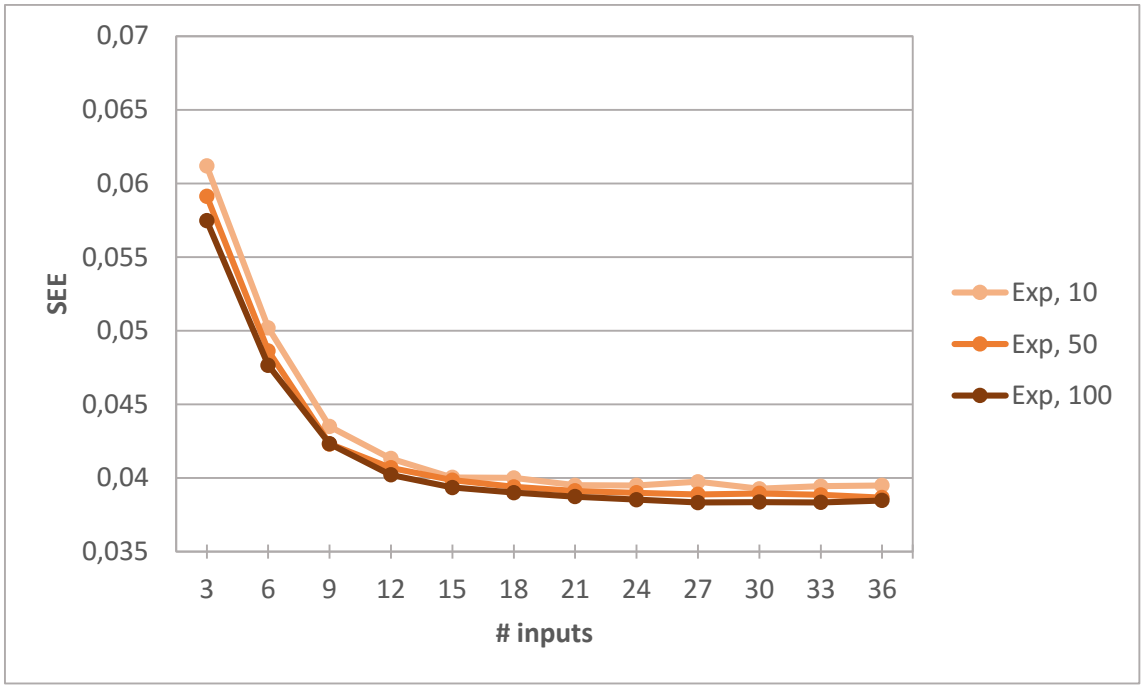

Figure 6: A comparison of ensembles with exponential model containing 10, 50 and 300 trees with regard to the different number of input data for SEE

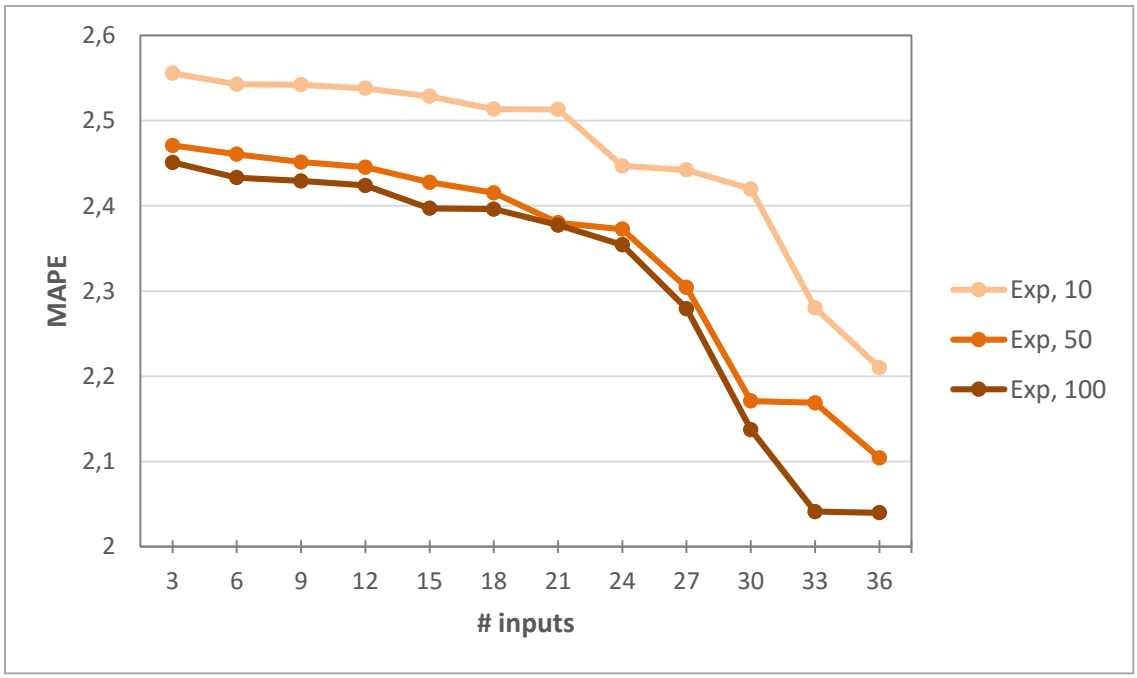

Figure 7: A comparison of ensembles with exponential model containing 10, 50 and 300 trees with regard to the different number of input data for $MAPE$

4 Conclusion

In the scope of this paper we have investigated the estimation of cumulative density function with MLM and its impact on the TCR prediction accuracy. The cumulative density functions were modelled with exponential and logistic models and their parameters were estimated with MLM, such as regression trees, neural networks and ensembles of regression trees.

In order to evaluate the goodness of fit of cumulative density functions to the actual data, we have used the SEE measure. Generally, the best fit results were achieved by ensembles of regression trees with *SEE* as low as 0.04. As expected, it was also shown that the more input data you provide, the more accurate estimation you get. With only 3 months of input data for ensemble with exponential model and 100 trees, the SEE value is 0.0575. On the contrary, if you provide the same ensemble with 12 or 36 months of input data, the SEE drops significantly to 0.04 or 0.038, respectively. Finally, cumulative density functions were used to predict the TCR. The TCR prediction accuracy was measured with $MAPE$ against the actual TCR. Ensembles of 100 regression trees yielded the highest prediction accuracy, with the average $MAPE$ value of 2.31%. Aforementioned results offer a possibility to use the developed TCR prediction model in practice. Such low $MAPE$ results could substantially improve the existing TCR prediction methods. A possible consequence of using the developed methodology are the substantially lower funds needed to be reserved for repairs during the warranty period.

The results achieved during this research phase reflect only the fundamental research in prediction of TCR with MLM. In the future, to improve the prediction accuracy, we intend to investigate the impact of other attributes, such as mean time to failure (MTTF), seasonality, market, etc. Further, ML methods setup also plays an important role in achieving greater prediction accuracy. Therefore, we will also investigate the optimization of ML methods attributes, for example, number of instances in leaves for regression trees.

Acknowledgement

This work was partially supported by the Ministry of Higher Education, Science and Technology of the Republic of Slovenia; Program No. P5-0018.

References

Blischke, W.R. & Murthy, D.N.P. (1994). Warranty cost analysis. New York: Marcel Dekker.

Grabert, M., Prechtel, M., Hrycej, T. & Günther, W. (2004). An early warning system for vehicle related quality data. Lecture Notes in Artificial Intelligence. 3275, 88–95.

Hall, P. & Strutt, J. (2003). Probabilistic physics-of-failure models for component reliability using Monte Carlo simulation and Weibull analysis: a parametric study. Reliability Engineering and System Safety. 80, 233–242.

Kim, C.S., Djamaludin, I. & Murthy, D.N.P., (2004). Warranty and discrete preventive maintenance. Reliability Engineering and System Safety. 84, 301–309. DOI: 10.1016/j.ress.2003.12.001.

Kofjač, D., Škraba, A. & Brglez, A. (2014). Warranty claims prediction with a combined model of market absorption and failure process. International journal of computing anticipatory systems. 30, 81-91.

Rai, B. & Singh, N. (2005). Forecasting warranty performance in the presence of the 'maturing data' phenomenon. International Journal of Systems Science. 36, 381–394.

Remesan, R., & Mathew, J. (2015). Machine Learning and Artificial Intelligence-Based Approaches. In Remesan, R. & Mathew, J. (eds), Hydrological Data Driven Modelling: A Case Study Approach (71-110). Switzerland: Springer International Publishing.

Shalev-Shwartz, S. & Ben-David, S. (2014). Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, New York, USA.

Smola, A. & VIshwanathan, S.V.N. (2008). Introduction to Machine Learning. Cambridge: Cambridge University Press.

Spiess, A.-N. & Neumeyer, N. (2010). An evaluation of R2 as an inadequate measure for nonlinear models in pharmacological and biochemical research: A Monte Carlo approach. BMC pharmacology. 10(6). DOI: 10.1186/1471-2210-10-6.

Wasserman, G. S. & Sudjianto, A. (1992). Neural networks for forecasting warranty claims. In C. H. Dagli, L. I. Burke, & Y. C. Shin (Eds.), Intelligent Engineering Systems Through Artificial Neural Networks. (2, 901-906). Fairfield, NJ, United States: ASME.

Wasserman, G. S. & Sudjianto, A. (1996). A comparison of three strategies for forecasting warranty claims. IIE Transactions. 28, 967–977.

Wu, S. (2012). Warranty data analysis: A review. Quality and Reliability Engineering International. 28, 795-805. DOI: 0.1002/qre.1282.

Wu, S. & Akbarov, A. (2011). Support vector regression for warranty claim forecasting. European Journal of Operational Research. 213(1), 196-204.

Xie, M. & Lai, C.D. (1995). Reliability analysis using an additive Weibull model with bathtub-shaped failure rate function. Reliability Engineering and System Safety. 52, 87–93.

Zuo, M.J., Liu, B., Murthy, D.N.P. (2000). Replacement-repair policy for multistate deteriorating products under warranty. European Journal of Operational Research. 123(3), 519–530. DOI: 10.1016/S0377-2217(99)00107-1.