https://communications.science.ankara.edu.tr

Commun.Fac.Sci.Univ.Ank.Ser. A1 Math. Stat.
Volume 70, Number 2, Pages 827–836 (2021)
DOI: 10.31801/cfsuasmas.899206
ISSN 1303–5991 E-ISSN 2618–6470

Received by the editors: March 18, 2021; Accepted: April 19, 2021

PREDICTING CREDIT CARD CUSTOMER CHURN USING SUPPORT VECTOR MACHINE BASED ON BAYESIAN OPTIMIZATION

Kamil Demirberk ÜNLÜ

Department of Mathematics, Atilim University, Ankara, TURKEY

Abstract. In this study, we have employed a hybrid machine learning algorithm to predict credit card customer churn. The proposed model is Support Vector Machine (SVM) with Bayesian Optimization (BO). BO is used to optimize the hyper-parameters of the SVM. Four different kernels are utilized. The hyper-parameters of the utilized kernels are calculated by the BO. The prediction power of the proposed models is compared by four different evaluation metrics. Used metrics are accuracy, precision, recall and F_1-score. According to each metrics linear kernel has the highest performance. It has accuracy of %91. The worst performance achieved by sigmoid kernel which has accuracy of %84.

1. Introduction

Customer churn is a business term expression which describes loss of customers. Firms invest in order not to lose their customers. Marketing departments continuously investigate the behavior of their existing customers and potential customers to understand the underlying causes of churn. These investigations are costly and time consuming. For that reason, in this study we propose a hybrid machine learning algorithm to predict customer churn of a bank by using the available data. We propose a model based on Support Vector Machine (SVM) which has many applications on regression and classifications. We utilized SVM as the classifier in this study because it ensure to use the technique called kernel transformations, projects the features space to a higher dimension, which makes it easier to find the bound between the classification objects. These kernels are non-linear so SVM can

2020 *Mathematics Subject Classification.* Primary 62P30; Secondary 65C20.

Keywords and phrases. Churn analysis, support vector machine, machine learning, hyper-parameter optimization.

demirberk.unlu@atilim.edu.tr
0000-0002-2393-6691.

capture complex relations between the observations without making complex calculations. Some application areas of SVM are financial bubble detection [1], stock market movement forecasting [2], financial time series forecasting [3], oil price forecasting [4] and air pollution modelling [5].

The SVM has three hyper-parameters. The first one is C. It is the penalty parameter and it tells the magnitude of the margin of the hyperplane. Large values of C imply small margin while small values of C imply large margins. The second is the kernels. These can be radial basis, polynomial or sigmoid. The last one is the γ parameter. It decides the curvature of the hyperplane. A high value indicates more curvature while a low value represents less curvature. The parameters can not be predicted by the algorithm itself. They can be defined by the user or optimization algorithms can be employed to decide these parameters. In this study we use Bayesian Optimization to handle the hyper-parameter optimization problem.

[1] compares SVM with artificial neural networks (ANN), k-nearest neighbours (KNN) decision tress (DT), random forest (RF) and logistic regression (LR) to predict financial bubbles in the S&P 500 index. Their findings show that SVM is favourable among the others with almost %95 accuracy. [2] compares the performance of SVM with Linear Discriminant Analysis, Elman Backpropagation Neural Networks and Quadratic Discriminant Analysis to predict the markets movements of NIKKEI 225. Their results show that SVM outperforms the other classifiers. [3] compares SVM with multi-layer back-propagation (BP) neural network to forecast five futures contracts of Chicago Mercantile Market. The authors show that SVM outperforms BP based on weighted directional symmetry, mean absolute error, directional symmetry and normalized mean square error. [4] investigated the prediction power of SVM on oil price forecasting and compared it with auto regressive moving average (ARIMA) and BP. The findings show that the prediction power of SVM outperforms the others. Lastly, [5] use SVM to predict air pollution in the urban areas of Honk Kong and the proposed model compared with ANN. The findings reveal that SVM performs better than ANN. The literature above mentioned provides the necessary evidence of the performs of SVM in both classification and regression. For that reason, in this study we chose our classifier as SVM.

Summary of some related works which employ machine learning algorithms to predict customer churn are given in this paragraph. Customer churn prediction based on textual data is studied by [6]. The Convolution Neural Network (CNN) is proposed as the model. The data set contains structured information with textual information. The results show that using textual data as a feature of the model increases the performance of the proposed model. [7] use churn rate of the customer to predict the electricity sales of the power market. Credit card churn prediction is done by [8]. The used models are logistic regression and decision tree based methods. The comparison of the models show that logistic regression performs better than the tree algorithms. Extended SVM (E-SVM) and ANN are proposed by [9] to model customer churn in e-commerce sector. The results show that E-SVM has better performance based on accuracy, coverage rate, hit ratio and lift coefficient. Also, it is noted that the new algorithm handles data well when imbalanced is an issue. [10] propose SVM and RF to predict customer churn of telecom sector and the results reveal that the investigate learning models behave similarly. Ten different machine learning algorithms are compared by [11] to classify customer churn. The findings of the study indicate that best performance achived by RF and ADA boost with almost %96 accuracy and SVM with %94 accuracy. Some other recent machine learning approach on customer churn predictions are [12], [13], [14] and [15].

The remainder of this paper is organized as follows. Section 2 devoted to the methodology. Data and experimental results are given in Section 3 and finally Section 4 concludes the study.

2. Methodology

2.1. **Support Vector Machine.** Support vector machine is a supervised machine learning algorithm that can be used for regression or classification. It is introduced by [16]. The main idea under the algorithm is to find a hyperplane to separate a data set into multiple classes. For instance, if there are two linearly separable classes in a data set, multiple lines can divide the data into two parts. SVM proposes to find the line which maximize the margin between the closest data points. These data points are called support vectors. For more than two separable case the algorithm uses hyperplane for classification. If the data set contains classes which are not linearly separable than kernel tricks are used. It is the transformation of the features to the higher dimensions which makes it easier to separate.

Suppose it is given a data set which has n observations of d variables with features $(x_1, x_2, ..., x_d)$ where $x_i \in \mathbb{R}^n$ and labels $(y_1, y_2, ..., y_n)$ where $y_i \in \{-1, 1\}$. Define the linear classifier

$$y(x) = \text{sign}(w^T x + b), \tag{1}$$

where w is the weight vector and b is the bias term. If the data set is linearly separable than the hyperplane $w^T x + b$ separates the two class as:

$$\begin{aligned} w^T x + b \geq 1 \quad & \text{for} \qquad y = 1, \\ w^T x + b < 1 \quad & \text{for} \quad y = -1. \end{aligned} \tag{2}$$

These two equations can be combined in one equations by multiplying both by y that is

$$y(w^T x + b) \geq 1. \tag{3}$$

The margin between the support vectors and the hyperplane is $\frac{2}{\|w\|}$. The optimal solution is found by maximizing the margin that is to minimize the length of w:

$$\min \frac{1}{2}\|w\|^2 \quad \text{s.t} \quad y(w^T x + b) \geq 1. \tag{4}$$

Solution for the above optimization problem can be obtained by using the Lagrange's method as

$$L(w, b, \alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{n} \alpha_i (y_i(x_i w^T + b) - 1), \tag{5}$$

where

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i, \tag{6}$$

and α is the non-negative Lagrange multiplier. The classifier for the linear case can be obtained as

$$f(x) = \text{sign}\left(\sum_{i=1}^{n} \alpha_i y_i (x^T x_i) + b \right). \tag{7}$$

In the non linear case the classifier transformed to

$$f(x) = \text{sign}\left(\sum_{i=1}^{n} \alpha_i y_i K(x, x_i) + b \right), \tag{8}$$

where $K(x_i, x_j)$ is the kernel function of the form

$$K(x_i, x_j) = \phi(x_i)^T \phi(x_j). \tag{9}$$

Mostly used kernels are:

$$\text{Radial basis kernel} : K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2),$$
$$\text{Polynomial kernel} : K(x_i, x_j) = (\gamma {x_i}^T x_j + r)^d,$$
$$\text{Sigmoid kernel} : K(x_i, x_j) = \tanh(\gamma {x_i}^T x_j + r).$$

2.2. **Bayesian Optimization.** Bayesian optimization is an iterative optimization which is very popular in hyper-parameter optimization of machine learning algorithms [17]. It searches and finds the candidate values based on previously obtained values. It contains two important elements called acquisition function and surrogate model [18]. The observed data points are fit into an objective function by the surrogate model. The acquisition function determines which points are used to balance the distribution of the surrogate model by evaluating the arrangements between exploration and exploitation [19]. Exploration is the process to search the upsampled area while the exploitation is the process of searching the most promising area in which the global minima or maxima may occurs.

In this paragraph we try to summarize Bayesian Optimization based on the work [17]. Firstly, the algorithm builds a surrogate model for the objective function. Secondly, using the surrogate model, it determines the optimal parameter values. Thirdly, the determined values are tested in the real objective function. Finally, the surrogate model is updated by the new results. These procedure repeats until the maximum number of iterations are achieved based on the initially surrogate model. Gaussian process can be given as a classic example of a surrogate model.

This algorithm is more efficient than grid search and random search, for that reason it is employed in this study.

2.3. **Evaluation Metrics.** We use 4 different evaluation metrics to test the performance of the proposed hybrid model. These are precision, recall, F_1-score and accuracy. Precision is the ratio of true positives to the sum of true positives and false positives.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}.$$

It measures the classifier ability to not to label a sample as positive which is negative. Recall is the ratio of true positives to the sum of true positives and false negatives.

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negatives}}.$$

It measures the classifier ability to identify the all positive sample points. F_1-score is the weighted average of precision and recall. It can take values between 0 and 1. The performance of the algorithm is at the best when takes value 1 or near to 1. In the same manner it is the worst when takes 0 or values very near to 0. It is calculated by the following formula:

$$F_1 = 2\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}.$$

Finally, accuracy is the fraction that the model predicts correctly. It is calculated as the ratio of sum of the total true positive and true negative to the total predictions. That is

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{True Negative} + \text{False Positive} + \text{False Negative}}.$$

It can take values between 1 and 0. If the performance of the model is high, it will take values near to 1, otherwise near to 0.

3. Data and Analysis

The data set for this study is obtained from the Kaggle [20] which is a machine learning and data science community. The data set contains 20 variables and each contains 10127 observations with no missing values. The variable with their descriptions are given in Table 1.

The data set contains categorical and numerical variables. Categorical variables are: AF, EL, MS, IC, GN, ID and CC. AF is the target variable. Numerical variables are CA, DC, MB, TR, MI, CC12, CL, OB, TA, TT, TC, TC4, TR, RB and AU. Categorical variables are converted to binary and one hot encoding. The target variable AF takes 1 if it is existing customer and 0 otherwise. GN takes value 0 if it is male and 1 otherwise. The rest of the categorical variables are converted to the one hot encoding format.

Table 1. Variables with their descriptions

	Variables	Descriptions
1	ID	Unique identifier for the customer.
2	Attrition Flag (AF)	Customer activity. If the account is closed takes value 1 otherwise 0.
3	Customer Age (CA)	Demographic variable. Customer ages in years.
4	Gender (GN)	Demographic variable. M for male and F for female.
5	Dependent Count (DC)	Number of dependents.
6	Education Level (EL)	Demographic variable which takes the values of high school, graduate, college, post-graduate, doctorate, uneducated and unknown.
7	Marital Status (MS)	Demographic variable which can be married, single, divorced and unknown.
8	Income Category (IC)	Demographic variable which can be less than 40000$, between 40000$ and 60000$, between 60000$ and 80000$, between 800000$ and 120000$ and greater than 120000$.
9	Card Category (CC)	Product variable which represent the card category. It takes the values of blue, silver, gold and platinum.
10	Months on Book (MB)	Represents the time of period with the bank.
11	Months Inactive (MI)	Number of months in active in the last 12 months.
12	Contacts Count (CC12)	Number of contacts in the last 12 months.
13	Credit Limit (CL)	The amount of credit limit on the card.
14	Revolving Balance (RB)	Total revolving balance on the card.
15	Open to Buy Credit (OB)	Average of open to buy credit on line of 12 months.
16	Total Amount of Changes (TA)	Change in transaction from the first quarter to the fourth quarter.
17	Total Transaction Amount (TT)	Total transaction amount of the last 12 months.
18	Total Transaction Count (TC)	Total transaction count of the last 12 months.
19	Total Change in Transaction Count (TC4)	Change in transaction count from the first quarter to the fourth quarter.
20	Total Relationship Count (TR)	Total number of products held by the customer.
21	Average Utilization Ratio (AU)	Average credit card utilization ratio.

As an example consider the transformation of CC:

$$\text{Card Category} = \begin{cases} 1 \quad 0 \quad 0 \quad 0 & \text{if card category} = \text{blue}, \\ 0 \quad 1 \quad 0 \quad 0 & \text{if card category} = \text{silver}, \\ 0 \quad 0 \quad 1 \quad 0 & \text{if card category} = \text{gold}, \\ 0 \quad 0 \quad 0 \quad 1 & \text{if card category} = \text{platinum}. \end{cases}$$

For categorical variables which has more than 2 different observations, one hot encoding is used. It is used because there is no ordinary relations between the observations. Otherwise, algorithms would assume natural ordering between the categorical variables which leads poor performance.

According to the data 16% of the customer leaving the bank while 86% staying. The vast majority of the customers are married and female level is slightly higher than the male proportion by 3% . Mostly, blue credit cards are used and in general income levels are less than 40000$. More than 30% of the credit card users have graduate level. The age of the customers are between 26 and 73. Lastly, credit card limits are between 1.438 and 34.516. Correlation between the numerical variables are given in Figure 1. The colour codes of the figure is given in the right hand side of the table. Light red implies strong positive correlation while dark purple implies negative correlations. It is seen that there exists high positive correlation between MA - CA, OB - CL, TC4 - TA and AU - RB, high negative correlation between AU - CL and AU - OB.

The data set divided into test and train set. The test set contains the %20 of the data while the rest is the train set.

We have started our analysis with the linear kernel. The best parameter for C, the penalty parameter, is obtained as 37.5598. On the train set the algorithm with the given parameters has %91 accuracy. The other metrics are given in the Table 2.

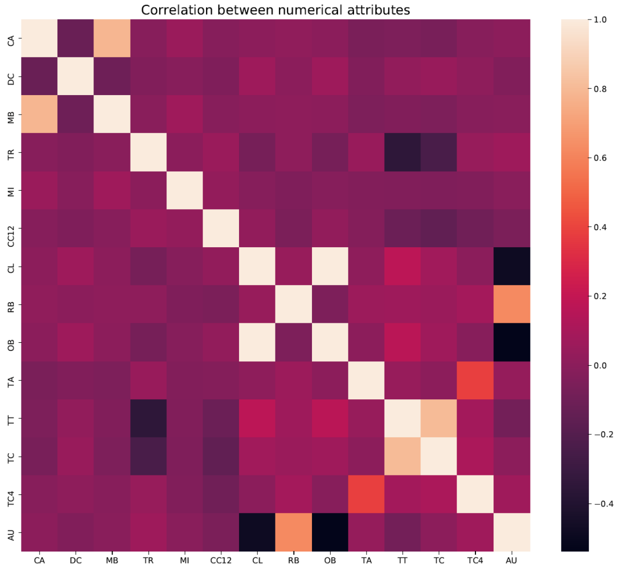

Figure 1. Correlation between numerical variables

Table 2. Evaluation Metrics for the Linear Kernel

Label	Precision	Recall	F_1-score	Support
0	0.78	0.60	0.68	319
1	0.93	0.97	0.95	1707
Macro Average	0.85	0.79	0.81	2026
Weighted Average	0.91	0.91	0.91	2026

As Table 2 shows linear kernel has weighted average, calculates the metrics for each label and takes the weighted average according to number of supports, of the precision, recall and F_1-score as 0.91 while it has accuracy of %91.

Secondly, polynomial kernel is utilized and by the help of the Bayesian optimization the best parameter for C is obtained as 0.28860 with $\gamma = 5.3504$. The accuracy of the train set with the given parameters are obtained as %87. The other metrics are given in Table 3.

Table 3. Evaluation Metrics for the Polynomial Kernel

Label	Precision	Recall	F_1-score	Support
0	0.61	0.63	0.62	319
1	0.93	0.92	0.93	1707
Macro Average	0.77	0.78	0.77	2026
Weighted Average	0.88	0.88	0.88	2026

As Table 3 shows polynomial kernel has weighted average of the precision as 0.77, recall as 0.78, F_1-score as 0.77 while it has accuracy of %88. It can be said that polynomial kernel is worse than the linear kernel according to the calculated metrics.

Thirdly, radial basis kernel is employed to predict credit card churns. The best parameter for C is obtained as 11.6085 with $\gamma = 3.2151$. The accuracy of the kernel in the train set is obtained as %86. The other metrics are given in Table 4.

Table 4. Evaluation Metrics for the Radial Kernel

Label	Precision	Recall	F_1-score	Support
0	0.62	0.31	0.42	319
1	0.88	0.96	0.92	1707
Macro Average	0.75	0.64	0.67	2026
Weighted Average	0.84	0.86	0.84	2026

As Table 4 shows radial kernel has weighted average of the precision as 0.75, recall as 0.64, F_1-score as 0.67 while it has accuracy of %86. It can be said that polynomial kernel is worse than the linear kernel and polynomial kernel according to the calculated metrics.

Lastly, sigmoid function is used as a kernel. The best parameters for the model are observed as $C = 45.4489$, $\gamma = 6.3796$. The model with these parameters have %83 accuracy. The metrics on the test set are given in Table 5.

Table 5. Evaluation Metrics for the Sigmoid Kernel

Label	Precision	Recall	F_1-score	Support
0	0.20	0.00	0.01	319
1	0.84	1.00	0.91	1707
Macro Average	0.52	0.50	0.46	2026
Weighted Average	0.74	0.84	0.77	2026

The worst result upon the investigated kernels are achieved by the sigmoid functions. The algorithm made 2026 forecasts and 2021 were identified as 1. As Table 5 shows it has accuracy of %84 while it has very low scores on precision, recall and F_1-score.

4. Conclusion

In this study, it is aimed to use a hybrid machine learning algorithm to classify the credit card churn of a bank. It is shown that the best kernel to predict churn behaviour of the customers is the SVM with linear kernel. Although, the data set is complex and contains many explanatory variables, a linear model fits the data better than the non-linear ones. The hyper-parameters of the algorithm is obtained by another algorithm called Bayesian optimization. Although, Bayesian optimization is not the only choice, it is utilized because of the flexibility and the speed of the algorithm. For the future studies the hyper-parameter optimizations tools can be compared and other machine learning and deep learning algorithms can be utilized to classify the churn behaviour of the customers.

Declaration of Competing Interests No potential conflict of interest was reported by the author.

Acknowledgement The author thanks to the associate editor and the anonymous reviewers for their valuable suggestions and corrections.

References

[1] Başoğlu Kabran, F., Ünlü K. D., A two-step machine learning approach to predict S&P 500 bubbles, *Journal of Applied Statistics*, (2020), 1-19. https://doi.org/10.1080/02664763.2020.1823947

[2] Huang, W., Nakamori, Y., Wang, S. Y., Forecasting stock market movement direction with support vector machine, *Computers & Operations Research*, 32(10) (2005), 2513-2522. https://doi.org/10.1016/j.cor.2004.03.016

[3] Tay, F. E., Cao, L., Application of support vector machines in financial time series forecasting, *Omega*, 29(4) (2001), 309-317. https://doi.org/10.1016/S0305-0483(01)00026-3

[4] Xie, W., Yu, L., Xu, S., Wang, S., A new method for crude oil price forecasting based on support vector machines, *International Conference on Computational Science*, (2006) 44-451. Springer, Berlin, Heidelberg.

[5] Lu, W. Z., Wang, W. J., Potential assessment of the "support vector machine" method in forecasting ambient air pollutant trends, *Chemosphere*, 59(5) (2005), 693-701. https://doi.org/10.1016/j.chemosphere.2004.10.032

[6] De Caigny, A., Coussement, K., De Bock, K. W., Lessmann, S., Incorporating textual information in customer churn prediction models based on a convolutional neural network, *International Journal of Forecasting*, 36(4) (2020), 1563-1578. https://doi.org/10.1016/j.ijforecast.2019.03.029

[7] Qu, Z., Wang, W., Qu, N., Liu, Y., Lv, H., Hu, K., Song, J. A., Forecasting method of electricity sales considering the user churn rate in a power market environment, *Journal of Electrical Engineering* & *Technology*, 14(4) (2019), 1585-1596. https://doi.org/10.1007/s42835-019-00215-9

[8] Nie, G., Rowe, W., Zhang, L., Tian, Y., Shi, Y., Credit card churn forecasting by logistic regression and decision tree. *Expert Systems with Applications*, 38(12) (2011), 15273-15285. https://doi.org/10.1016/j.eswa.2011.06.028

[9] Yu, X., Guo, S., Guo, J., Huang, X., An extended support vector machine forecasting framework for customer churn in e-commerce, *Expert Systems with Applications*, 38(3) (2011), 1425-1430. https://doi.org/10.1016/j.eswa.2010.07.049

[10] Andrews, R., Zacharias, R., Antony, S., James, M. M., Churn prediction in Telecom sector using machine learning, *International Journal of Information*, 8(2) (2019), https://doi.org/10.30534/ijiscs/2019/31822019

[11] Sabbeh, S. F., Machine-learning techniques for customer retention: A comparative study, *International Journal of Advanced Computer Science and Applications*, 9(2) (2018).

[12] Pamina, J., Raja, B., SathyaBama, S., Sruthi, M. S., VJ, A., An effective classifier for predicting churn in telecommunication, *Journal of Adv Research in Dynamical* & *Control Systems*, 11 (2019).

[13] Vo, N. N., Liu, S., Li, X., Xu, G., Leveraging unstructured call log data for customer churn prediction, *Knowledge-Based Systems*, 212 (2021), 106586,https://doi.org/10.1016/j.knosys.2020.106586

[14] Lalwani, P., Mishra, M. K., Chadha, J. S., Sethi, P., Customer churn prediction system: a machine learning approach, *Computing*, (2021), 1-24, https://doi.org/10.1007/s00607-021-00908-y

[15] Zhuang, Y., Research on e-commerce customer churn prediction based on improved value model and XG-boost algorithm, *Management Science and Engineering*, 12(3) (2018), 51-56

[16] Cortes, C., Vapnik, V., Support-vector networks. *Machine Learning*, 20(3) (1995), 273-297.

[17] Snoek, J., Larochelle, H., Adams, R. P., Practical Bayesian optimization of machine learning algorithms. *arXiv preprint*, (2012), arXiv:1206.2944.

[18] Injadat, M., Salo, F., Nassif, A. B., Essex, A., Shami, A., Bayesian optimization with machine learning algorithms towards anomaly detection. *IEEE Global Communications Conference*, (2018), 1–6.

[19] Yang, L., Shami, A. On hyperparameter optimization of machine learning algorithms: theory and practice, *Neurocomputing*, 415 (2020), 295-316. https://doi.org/10.1016/j.neucom.2020.07.061

[20] Kaggle `https://www.kaggle.com/sakshigoyal7/credit-card-customers`, Acceded:January 3, 2021.