reinforcement machine learning. Supervised learning is based on the logic of loading the training dataset and test dataset into the system, making the necessary labels for each data in the dataset and thus establishing a relationship between the input dataset and the output dataset. In supervised learning, the target variable is obvious. If the target variable is numeric, the regression model is used; if it is categorical, the classification model is used. Unsupervised learning is a machine learning technique that tries to find patterns in data without any labels. Unlike supervised learning, there is no target variable in unsupervised learning. The two main methods used in unsupervised learning are clustering and dimensionality reduction. Reinforcement learning is a machine learning technique that aims to train the machine with the reactions it receives from the environment without training data. The idea involved in reinforcement learning is that the machine continuously trains itself depending on the environment and applies its enriched knowledge to solve problems.

The classification model approach is suitable for the subject of this study, which choice a customer can make on an insurance product, whether to buy or not. There are classification model solutions that are frequently used in machine learning algorithms and provide good results. These are Logistic Regression, Support Vector Machine, Random Forest and XGBoost algorithms.

Logistic Regression is a type of regression applied when the target variable is a binary variable and belongs to a class of models called generalized linear models (Hosmer, 1989). Support Vector Machine is a vector space-based machine learning method with a decision boundary between the two classes that are furthest from any point in the training data (Schlkopf, 2018). Random Forest is a batch learning method for classification, regression and other tasks that works by building large numbers of decision trees at training time (Breiman, 2001). XGBoost, eXtreme Gradient Boosting is an open source software library that provides a streamlined gradient boosting framework for C++, Java, Python, R, Julia, Perl and Scala.

The model experiments and the decided model in this study were written in Python language, in the Jupyter Notebook program. Jupyter Notebook is an open source program that provides an interactive environment for various programming languages.

Methods

Dataset Description

In this study, the dataset used for the statistical and machine learning model to be established during the application phase is real customer data, obtained anonymously from the insurance industry. The variables included in the dataset are shown in the table below.

Table 1. List of Variables

Variable Name	Explanation	Data Type
URUN_ADET_TOPLAM	Total number of products	Numeric
URUN_ADET_FK	Number of personal accident products	Numeric
URUN_ADET_HAYAT	Number of life products	Numeric
URUN_ADET_EMEKLILIK	Number of pension products	Numeric
MUSTERILILIK_SURE	Time without first customer	Numeric
INAKTIF_SURE	Inactive time	Numeric
KISA_KONUSMA	Number of short-term conversations	Numeric
ORTA_KONUSMA	Number of medium-term conversations	Numeric
UZUN_KONUSMA	Number of long-term conversations	Numeric

Table 1. List of Variables continued

Variable Name	Explanation	Data Type
MAX_SURE	Maximum time spoken	Numeric
MIN_SURE	Minimum time spoken	Numeric
SON1YIL_ARANMA	Number of searches in the last year	Numeric
SON1YIL_ULASIM	Number of transportation in the last	Numeric
SON1YIL_SATIS	Number of sales in the last year	Numeric
YAS	Customer's age	Numeric
CINSIYET	Customer's gender	Categorical
MEDENI_DURUM	Customer's marital status	Categorical
GELIR_DUZEY	Customer's income level	Categorical
EGITIM_DUZEY	Education level of the customer	Categorical
SEHIR_DUZEY	Customer's city level	Categorical
TOPLAM_AKTIF_PRIM	Customer's active total premium	Numeric
TOPLAM_PRIM	Customer's total premium	Numeric
TAZMINAT_FLAG	Receiving compensation payment	Numeric
TARGET	Purchasing/Not purchasing a product	Categorical

Dataset Analysis

The distribution of the target variable in the dataset is summarized as follows. A value of 0 indicates that the customer has not purchased the product and a value of 1 indicates that the customer has purchased the product.

There are a total of 850 rows -customers- of data in the dataset. 305 of these data resulted in purchases.

Table 2. Distribution of Target Variable

Target	Count
0	545
1	350
Total	850

35.9% of the customers in the dataset purchased the product; it appears that 64.1% have not purchased the product.

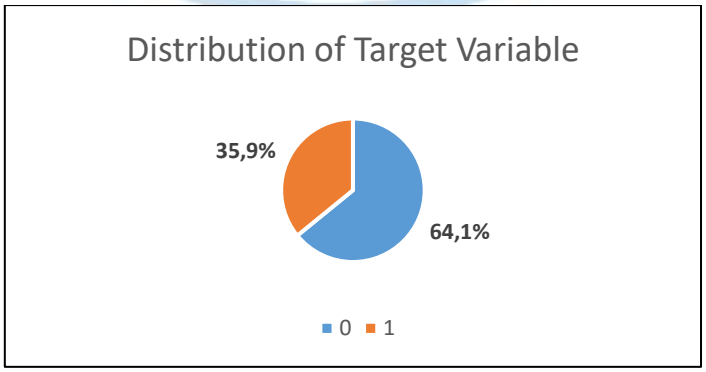

Figure 1. Distribution of Target Value

The correlation matrix, where the relationship between variables can be observed, is shown in Figure 2. Accordingly, the strongest relationship is the positive relationship between urun_adet_toplam and urun_adet_fk with 0.92. Strong relationships are not evident in most of the variables.

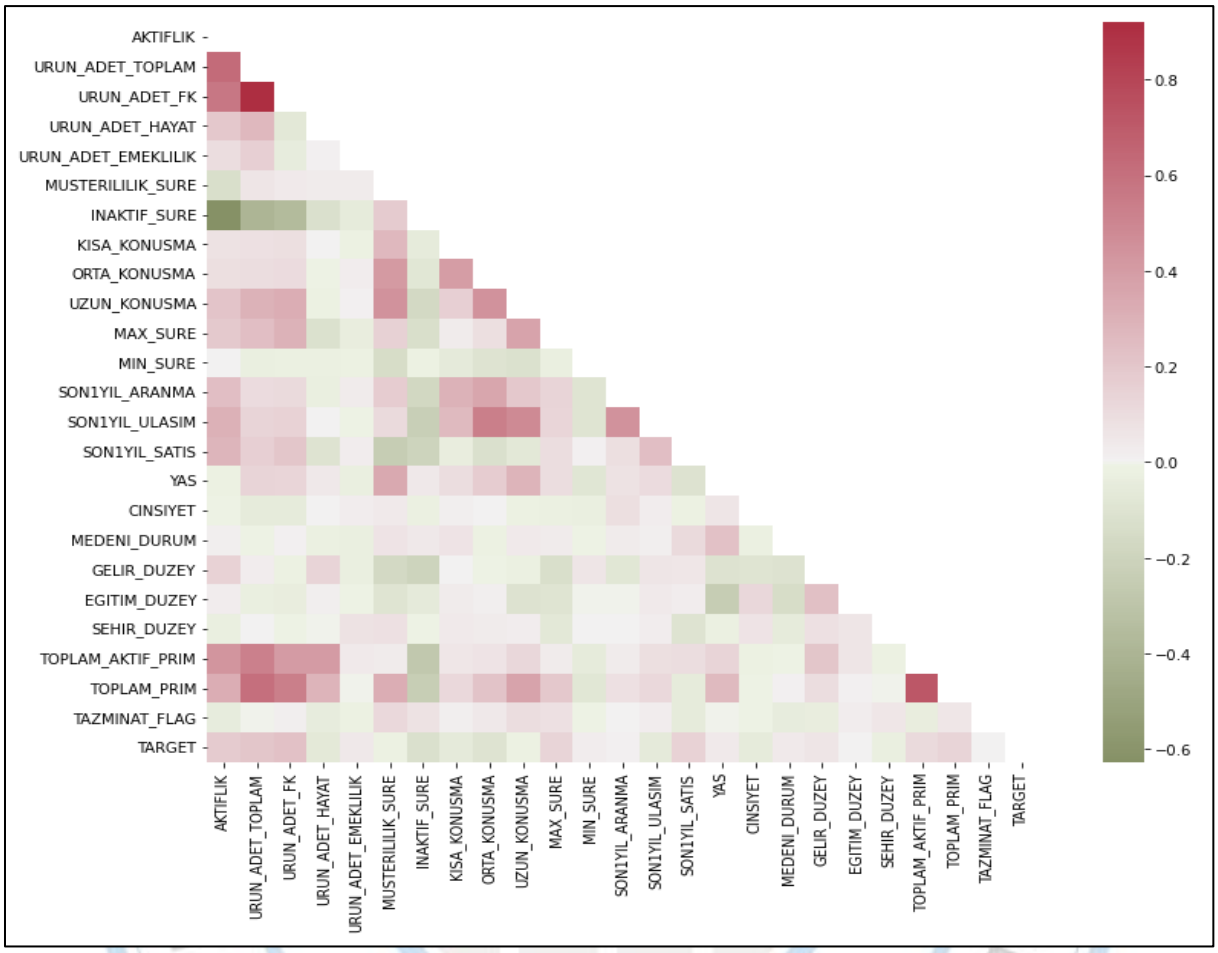

Figure 2. Correlation Matrix Between Variables

Data Preprocessing

Data preprocessing is a step involved in the data mining and data modeling process that converts raw data into a format that can be understood and analyzed by computers. Data preprocessing can refer to manipulating or reducing data before use to ensure or improve performance. Missing or defective values may be encountered during data collection or creation methods. Analyzing data that has not been carefully screened for such problems may yield misleading results. Therefore, the representation and quality of the data come first before performing any analysis (Pyle, 1999). The methods used in the literature as data preprocessing steps can be listed as data cleaning, data reduction, data transformation.

When the dataset used in this study is examined, the first striking detail is the presence of variables with categorical data type. These are cinsiyet, medeni_durum, gelir_duzey, eğitim_duzey, sehir_duzey. In order for these variables to be read by machine learning algorithms, these variables must be converted to numeric values. Label Encoding is a method used to convert categorical columns into numeric columns to fit machine learning models that only receive numeric data. Five variables related to this method are converted into numeric columns.

After converting these variables into numeric columns, the data distribution is shown in Figure 3.

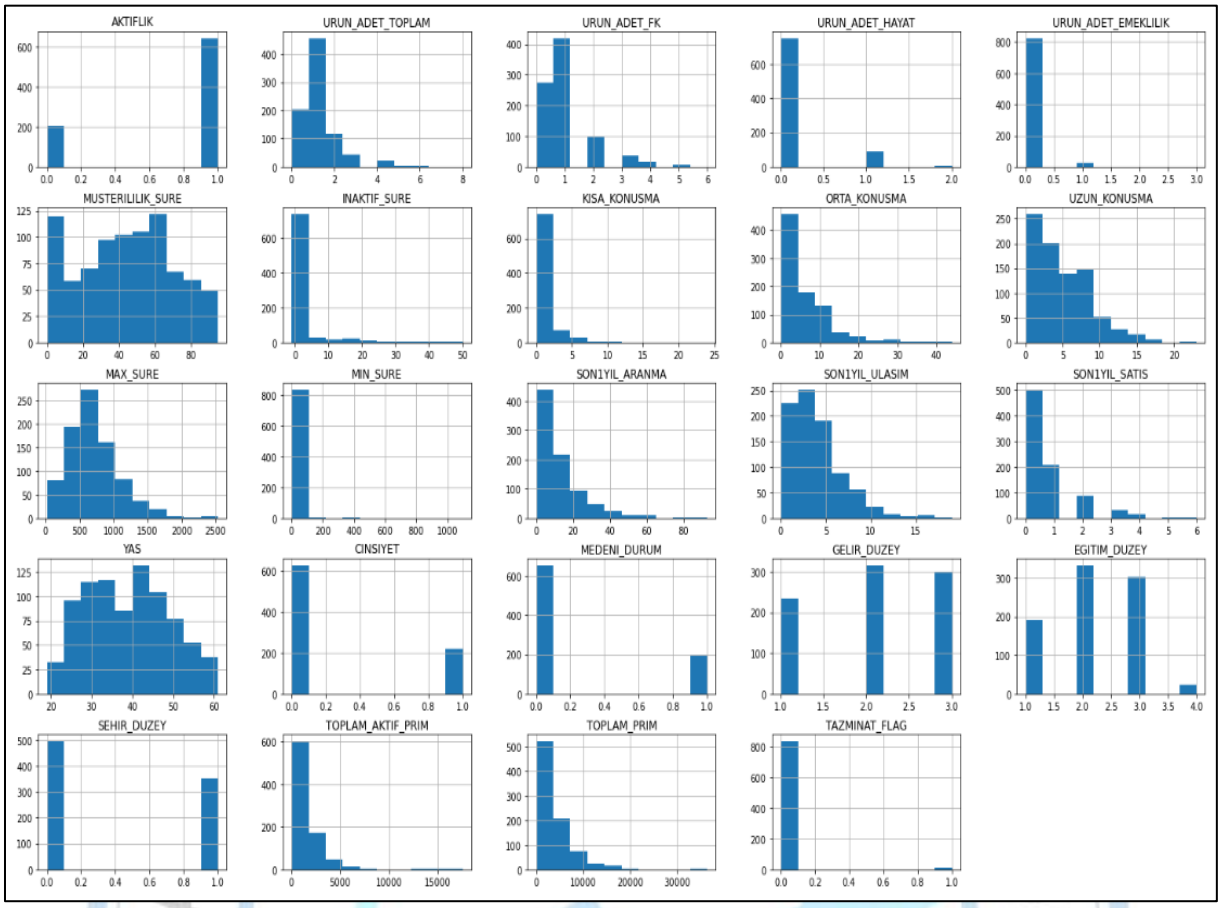

Figure 3. Distribution of Data

By looking at the summary histogram distribution graphs of all variables given in Figure 3, it is seen that some variables have outlier values when the data distribution is examined. These variables are urun_adet_toplam, urun_adet_fk, urun_adet_hayat, urun_adet_emeklilik, inaktif_sure, kisa_konusma, orta_konusma, uzun_konusma, max_sure, min_sure, son1yil_aranma, son1yil_ulasim, son1yıl_satis, toplam_aktif_prim, toplam_prim. Before making a decision about these variables, they should be examined in detail.

Detailed analysis for relevant variables that need to be re-examined is shown in Figure 4.

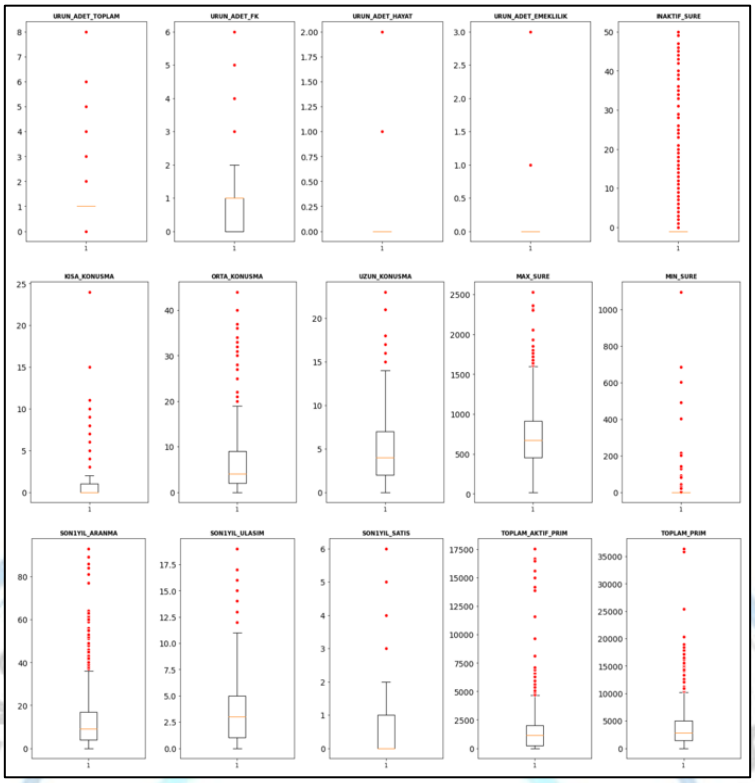

Figure 4. Variables with Outliers

The data was updated by replacing the outliers with the 99th percentile value of the relevant variable. The resulting data distribution is shown in Figure 5.

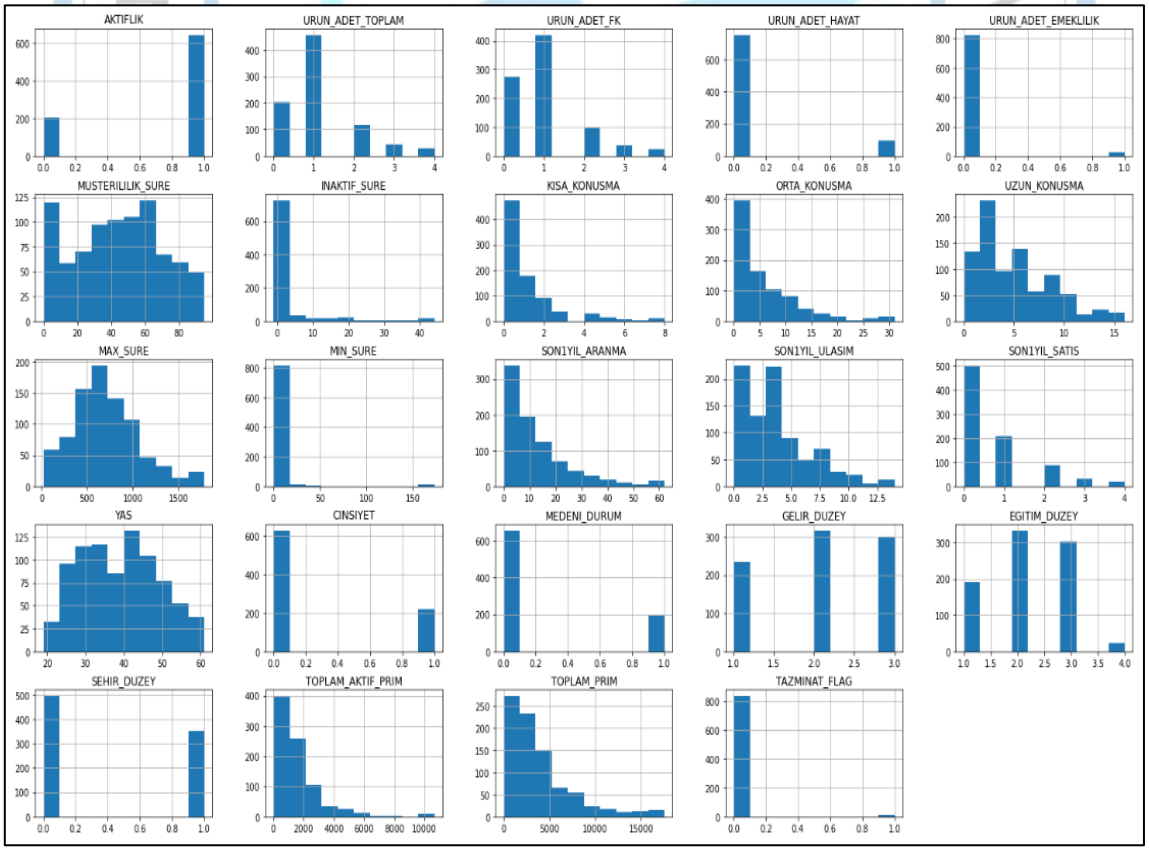

Figure 5. Distribution of Data After Data Preprocessing

The changes made to the data after the data preprocessing process are summarized as follows.

• Converting categorical columns to numeric columns
• Detection and suppression of outliers

Feature Engineering

Feature engineering or feature extraction or feature discovery is the process of using domain knowledge to extract features (characters, attributes) from raw data. The main motivation in feature engineering is to use extra features to improve the quality of the results obtained from a machine learning process, compared to using only raw data in a machine learning process.

Feature engineering aims to increase the predictive power of machine learning algorithms and achieve better results by creating new features that help streamline the machine learning process.

The following applications were made on the existing variables in this study.

- The son1yil_ulasim_oran variable was created based on the ratio of last year transportation and last year search variables.
- The son1yil_satis_oran variable was created based on the ratio of last year sales and last year transportation variables.
- The variable kisa_konusma_oran was created based on the ratio of the number of short speeches to the number of short, medium and long speeches. By calculating this ratio for all variables in this group, the variables orta_konusma_oran and uzun_konusma_oran were created.
- The toplam_aktif_prim_oran variable was created based on the ratio of total active premium and total premium variables.
- The average of the total active premium value in the entire dataset was determined and the toplam_aktif_prim_grup variable was created by assigning values of 1 and 0 for the data higher and lower than this average.
- The toplam_prim_grup variable was created by determining the average of the total premium value in the entire dataset and assigning values of 1 and 0 for the data higher and lower than this average.
- The variables urun_adet_fk_flag, urun_adet_hayat_flag and urun_adet_emeklilik_flag were created using the values 1 and 0 to define the expression yes and no, over the variables showing the product quantities.
- Customer service duration and inactive period variables were converted from monthly values to annual values and customer musterililik_sure_grup and inaktif_sure_grup variables were created.
- A new variable was created by age grouping based on the age variable. By following an intuitive method to meet business needs, age groups of 18-25, 26-35, 36-45, 46-55, 56-65 and 65+ were created. These groups are defined as yas_grup_1, yas_grup_2, yas_grup_3, yas_grup_4, yas_grup_5, yas_grup_6, respectively.
- Among the raw variables, gelir_duzey and egitim_duzey variables were converted into dummy variables. Among these variables, gelir_duzey_1 is defined as low, gelir_duzey_2 is medium, gelir_duzey_3 is high, egitim_duzey_1 is defined as primary education, egitim_duzey_2 is high school, egitim_duzey_3 is defined as university, egitim_duzey_4 is defined as master's degree and above.
- Among the newly created variables, musterilik_sure_grup, inaktif_sure_grup, yas_grup variables have been converted into dummy variables.

After that iteration, the total number of columns in the dataset increased from 25 to 60 and the total number of variables increased from 23 to 58.

Feature Selection

Feature selection is the process of reducing the number of input variables when developing a predictive model. In machine learning models such as regression or classification, there are often too many variables or features to work with. As the number of features increases, the difficulty of modeling them may increase. Irrelevant or partially relevant features can negatively impact model performance. It is desirable to reduce the number of input variables, both to improve the performance of the model and, in some cases, to reduce the computational cost (Murphy, 2012).

In this study, Lasso Regression method was used in the feature selection process. The main purpose

of Lasso Regression is feature selection and organization of data models. With the increasing number of variables in the feature engineering step, the variables that were meaningful among all variables were wanted to be used as model input.

Apart from the id and target columns, there are 58 columns on the dataset, that is, variables that can be entered as input into the model. The number of variables was reduced to 22 after the Lasso Regression method was applied, with the optimum number of variables at the $\alpha = 0.005$ significance level. The coefficients were close to zero and the number of variables minimized to zero was 36.

Data Scaling

Variables measured at different scales do not contribute equally to the model fitting and model learning function and may create a bias in the result. Therefore, to deal with this potential problem, the scaling method is applied on the variables.

Standardization or Standard Scaler is a method performed before many machine learning models are built to standardize the range of functionality of the input dataset. This method is used to resize the distribution of values so that the observed values have a mean of 0 and a standard deviation of 1.

Before establishing the machine learning model, all variables remaining after the Lasso Regression process in the dataset were standardized.

Model Performances

Machine learning models were established with Logistic Regression, Support Vector Machine, Random Forest and XGBoost algorithms for creating propensity to buy model. The models were built on the same training and test set. The training set ratio is 80% of dataset and the test set ratio is 20% of dataset.

Classification reports of the created models are shown in Figure 6. Classification reports include accuracy, precision, recall and F1 score performance metrics of the models.

Lojistik Regresyon:
Model Accuracy: 73.0

	precision	recall	f1-score	support
0	0.78	0.85	0.81	117
1	0.59	0.45	0.51	53
accuracy			0.73	170
macro avg	0.68	0.65	0.66	170
weighted avg	0.72	0.73	0.72	170

SVM:
Model Accuracy: 75.0

	precision	recall	f1-score	support
0	0.76	0.94	0.84	117
1	0.72	0.34	0.46	53
accuracy			0.75	170
macro avg	0.74	0.64	0.65	170
weighted avg	0.75	0.75	0.72	170

Random Forest:
Model Accuracy: 77.0

	precision	recall	f1-score	support
0	0.78	0.93	0.85	117
1	0.73	0.42	0.53	53
accuracy			0.77	170
macro avg	0.76	0.67	0.69	170
weighted avg	0.76	0.77	0.75	170

XGBoost:
Model Accuracy: 77.0

	precision	recall	f1-score	support
0	0.82	0.85	0.84	117
1	0.65	0.58	0.61	53
accuracy			0.77	170
macro avg	0.73	0.72	0.73	170
weighted avg	0.77	0.77	0.77	170

Figure 6. Performance Metrics of the Models

According to the results of the classification reports:

- Accuracy rates were over 70 points in each model and the best algorithms were Random Forest and XGBoost algorithms.
- The XGBoost algorithm achieved the best F1 score in 1 prediction. In the prediction of 0, the F1 score is the second model.

AUC-ROC curve comparison of the models is shown in Figure 7.

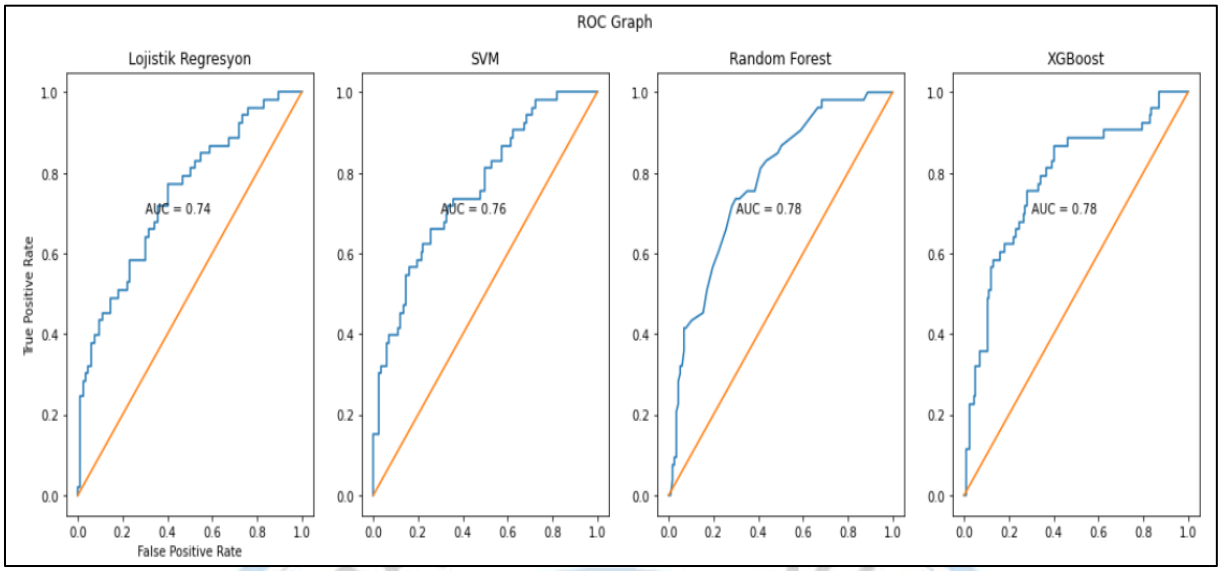

Figure 7. AUC-ROC Curve of the Models

According to the AUC-ROC curve results:

- The AUC score of the Logistic Regression model was 0.74; the AUC score of the Support Vector Machine model was 0.76; the AUC score of the Random Forest algorithm was 0.78; the AUC score of the XGBoost algorithm was 0.78.

RESULTS

A comparison of the results obtained with the original variables and the results obtained after the feature engineering and feature selection stages is shown in Table 3.

Table 3. Model Comparison

Model Name	Accuracy	AUC-ROC	F1 Score
Logistic Regression with Original Variables	0.69	0.71	0.39
Support Vector Machine with Original Variables	0.68	0.72	0.18
Random Forest with Original Variables	0.76	0.76	0.54
XGBoost with Original Variables	0.72	0.74	0.49
Logistic Regression with New Features	0.73	0.74	0.51
Support Vector Machine with New Features	0.75	0.76	0.46
Random Forest with New Features	0.77	0.78	0.53
XGBoost with New Features	0.77	0.78	0.61

When the classification reports and AUC-ROC curves were examined, the most successful model was chosen as the model belonging to the XGBoost with New Features algorithm.

The classification report of the improved model was run again. The difference between this model and the previous model is the addition of selected hyperparameters to the model function. The classification report of the improved model is shown in Figure 8.

```
XGBoost Tuned:
Model Accuracy: 77.06
              precision    recall  f1-score   support

           0       0.82      0.85      0.84       117
           1       0.64      0.60      0.62        53

    accuracy                           0.77       170
   macro avg       0.73      0.72      0.73       170
weighted avg       0.77      0.77      0.77       170
```

Figure 8. XGBoost Tuned Model Classification Report

The confusion matrix output of the final model obtained, the XGBoost Tuned model, is shown in Figure 9.

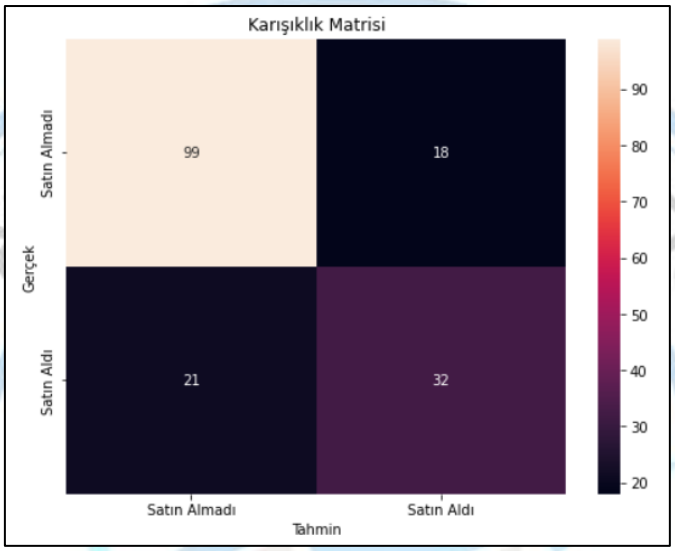

Figure 9. XGBoost Tuned Model Confusion Matrix

In machine learning, feature importance scores are used to determine the relative importance of each feature in a dataset when building a predictive model. Calculating feature importance is a method for ranking features according to their contribution to the prediction. Feature importance refers to techniques that calculate a score for all input features for a given model.

The importance level of the variables selected for the XGBoost Tuned model, which is the final model obtained, is shown in Table 4.

Table 4. Feature Importance Scores

Variable Name	Importance Score
URUN_ADET_FK	0.104066
SON1YIL_ULASIM_ORAN	0.056268
SON1YIL_SATIS	0.056192
MAX_SURE	0.054433
YAS	0.051663
MEDENI_DURUM	0.049584
YAS_GRUP_3	0.048799

Table 4. Feature Importance Scores continued

Variable Name	Importance Score
ORTA_KONUSMA	0.047688
UZUN_KONUSMA	0.047350
TOPLAM_PRIM_GRUP	0.045580
MUSTERILILIK_SURE	0.043117
SON1YIL_ARANMA	0.041984
EGITIM_DUZEY_1	0.041684
KISA_KONUSMA	0.040309
TOPLAM_AKTIF_PRIM_GRUP	0.039079
INAKTIF_SURE	0.038413
GELIR_DUZEY_3	0.037984
SON1YIL_ULASIM	0.037656
CINSIYET	0.032812
URUN_ADET_HAYAT	0.030648
EGITIM_DUZEY_2	0.028587
MIN_SURE	0.026104

DISCUSSION AND CONCLUSION

Using resources efficiently is always important for businesses. A method that can provide this efficiency is for businesses to find the propensity of their customer base with the help of statistical models and to implement marketing activities by targeting the right customer base in line with the results they find. Based on this idea, in this study, the factors affecting customers' tendency to purchase an insurance product were investigated with real data collected from the insurance industry. The collected data consists of 1 column expressing customer identity, 1 dependent variable and 24 independent variables. The data was examined, introduced and analyzed in detail. Data preparation stages were carried out before establishing a model with the help of machine learning algorithms. In these stages, data preprocessing was performed, new variables were created from the variables in the data, variable selection was made using statistical methods and data scaling was performed. In the final stage, classification models were established with machine learning algorithms. The XGBoost algorithm was selected as the champion model algorithm.

References

Breiman, L., (2001). “Random Forests” Machine Learning, 45: 5–32.

Hosmer, D.W., Lemeshow, S., (1989). Applied Logistic Regression, The Multiple Logistic Regression Model, 25-37.

Murphy, K.P., (2012). Machine learning: a probabilistic perspective, MIT Press, 86.

Pyle, D., (1999). Data Preparation for Data Mining. Morgan Kaufmann Publishers, Los Altos, California.

Schlkopf, B., (2018). Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond, MIT Press, ISBN 0-262-53657-9. OCLC 1039411838.

Acknowledgment

I would like to my endless gratitude to Prof. Dr. Birsen EYGİ ERDOĞAN, who showed interest and support in the planning, research and execution of the study and to my family and loved ones who have always supported me throughout my life.

Conflict of Interest

The authors have declared that there is no conflict of interest.

Weighted Approximation of the Stancu Type Generalisation of the Chlodowsky-Szazs Mirakyan Hybrid Operator

Afşin Kürşat Gazanfer[1*], Şeyda Ünal[1]

[1] Zonguldak Bülent Ecevit University, Faculty of Science, Department of Mathematics, 67100, Zonguldak, Türkiye

*Corresponding author e-mail: afsinkursat@beun.edu.tr

Abstract

The following text describes the use of positive linear operators that are incorporated with the Stancu type generalization of Chlodowsky and Szasz type operators. The study focuses on the approximation properties of these operators in the space of continuous functions of two variables on a compact set. The convergence rate of these operators is determined through the modulus of continuity, and weighted approximation properties are also explored for these positive linear operators in a weighted space of functions of two variables. The convergence rate for these operators is established using the weighted modulus of continuity.

Keywords: *Modulus of continuity, weighted approximation, rate of convergence.*

INTRODUCTION

Let $x \in \mathbb{R}^{+,0} = [0,\infty)$ and $f \in \mathrm{C}(\mathbb{R}^{+,0})$. The Chlodowsky polynomial, commonly referenced in mathematics and denoted by $B_n^*(f;x)$, is a well-known concept in which the degree is denoted as n

$$B_n^*(f;x) = \sum_{k=0}^{n} \binom{n}{k} \left(\frac{x}{b_n}\right)^k \left(1-\frac{x}{b_n}\right)^{n-k} f\left(\frac{k}{n} b_n\right) \quad , \qquad 0 \le x \le b_n \tag{1}$$

where (b_n) is characterized by a sequence of positive numbers with specific properties

$$\lim_{n\to\infty} b_n = \infty, \qquad \lim_{n\to\infty} \frac{b_n}{n} = 0. \tag{2}$$

Stancu provided a generalisation of Bernstein operators for $x \in [0,1]$ with two parameters $r, s \in \mathbb{N} \cup \{0\}$:

$$C_{k,s}(f;x) = \sum_{k=0}^{n-sr} \binom{n-sr}{k} x^k (1-x)^{n-sr-k} \sum_{\mu=0}^{s} \binom{s}{\mu} x^\mu (1-x)^{s-\mu}.$$

When $s = r = 0$, the classical Bernstein operators are obtained. İbikli studied the Stancu generalisation of Bernstien-Chlodowsky polynomials in 2000. According to the this, $r, s \in \mathbb{N} \cup \{0\}$ and for (b_n) satisfying the properties in (2)

$$C_{k,s}^*(f;x) = \sum_{k=0}^{n-sr} \binom{n-sr}{k} \left(\frac{x}{b_n}\right)^k \left(1-\frac{x}{b_n}\right)^{n-sr-k} \sum_{\mu=0}^{s} \binom{s}{\mu} \left(\frac{x}{b_n}\right)^\mu \left(1-\frac{x}{b_n}\right)^{s-\mu} \tag{3}$$

can be written as. For $s = r = 0$, we get Bernstein-Chlodowsky operators.

Let $x \in \mathbb{R}^{+,0}$ and $f \in \mathrm{C}(\mathbb{R}^{+,0})$. The modified Szasz-Mirakjan operators, denoted by $\Im_n(f;\xi_n,\zeta_n,x) = \Im_n(f;x)$, are defined as

$$\Im_n(f;x) = e^{-\xi_n x} \sum_{k=0}^{\infty} \frac{(\xi_n x)^k}{k!} f\left(\frac{k}{\zeta_n}\right) \tag{4}$$

with (ξ_n) and (ζ_n) being increasing and unbounded sequences of positive numbers, resulting in

$$\lim_{n\to\infty} \frac{1}{\zeta_n} = 0, \qquad \lim_{n\to\infty} \frac{\xi_n}{\zeta_n} = 1 + O\left(\frac{1}{\zeta_n}\right). \tag{5}$$

In (Walczak 2000), Walczak presented the modified Szasz-Mirakjan operators in the polynomial