1. Description of The Project and The Dataset

Examination of customer behavior has an important place today. If a relationship can be established between the loyalty of customers and some of their features, improvement can be achieved by going beyond these features.

Therefore, in this project, relationships will be sought in the loyalty of telecommunication customers and it is aimed to provide more investment to the customers who are connected or to work on increasing the loyalty of other customers through these connections.

The dataset is named as Telco Customer Churn which focused customer retention. Behavior is predicted to retain customers. All relevant customer data can be analyzed and focused customer retention programs can be developped.

There are 7043 rows, and 21 columns in the raw data, each row represents a customer, each column contains customer's attributes described on the column Metadata.18 of 21 columns are categorical.

The data set includes information about:

- Customers who left within the last month – the column is called Churn
- Services that each customer has signed up for – phone, multiple lines, internet, online security, online backup, device protection, tech support, and streaming TV and movies
- Customer account information – how long they've been a customer, contract, payment method, paperless billing, monthly charges, and total charges
- Demographic info about customers – gender, age range, and if they have partners and dependents

The target column is Churn. It is a binary classification. No values are in the majority.

As mentioned earlier, there are 21 columns in the raw data. 7043 entries exists and all of them attributes don't have any null value.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 21 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   customerID        7043 non-null   object
 1   gender            7043 non-null   object
 2   SeniorCitizen     7043 non-null   int64
 3   Partner           7043 non-null   object
 4   Dependents        7043 non-null   object
 5   tenure            7043 non-null   int64
 6   PhoneService      7043 non-null   object
 7   MultipleLines     7043 non-null   object
 8   InternetService   7043 non-null   object
 9   OnlineSecurity    7043 non-null   object
 10  OnlineBackup      7043 non-null   object
 11  DeviceProtection  7043 non-null   object
 12  TechSupport       7043 non-null   object
 13  StreamingTV       7043 non-null   object
 14  StreamingMovies   7043 non-null   object
 15  Contract          7043 non-null   object
 16  PaperlessBilling  7043 non-null   object
 17  PaymentMethod     7043 non-null   object
 18  MonthlyCharges    7043 non-null   float64
 19  TotalCharges      7043 non-null   object
 20  Churn             7043 non-null   object
dtypes: float64(1), int64(2), object(18)
memory usage: 1.1+ MB
```

Figure 1.1 - Dataset Information

There are 3 numerical attributes as MontlyCharges, SeniorCitizen, and tenure. TotalCharges also looks like numerical but it's not. It is kept as string data type and it has some of ' ' values. If the data quality report of these three attributes are examined;

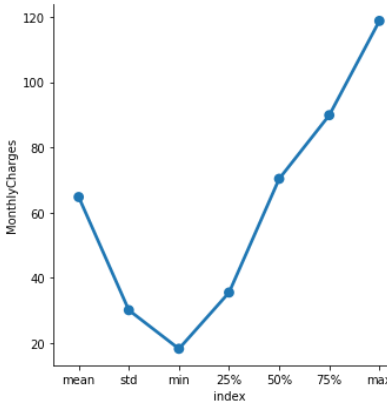


Figure 1.2 - MontlyCharges DQR

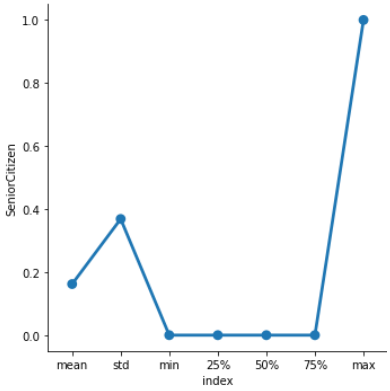


Figure 1.3- SeniorCitizen DQR

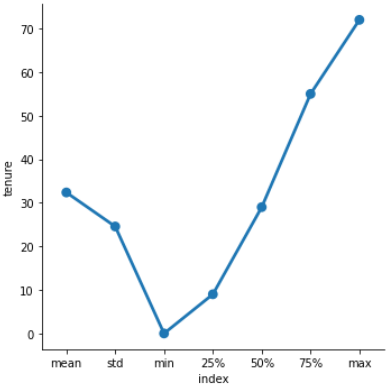


Figure 1.4 - tenure DQR

As can be seen here, some irregular distributions are observed in seniorCitizen.

The harmonious relationship between tenure and montlyCharges can also be observed.

When it comes to categorical data, it is generally observed that they have few different values. They usually take binary or 3 values.

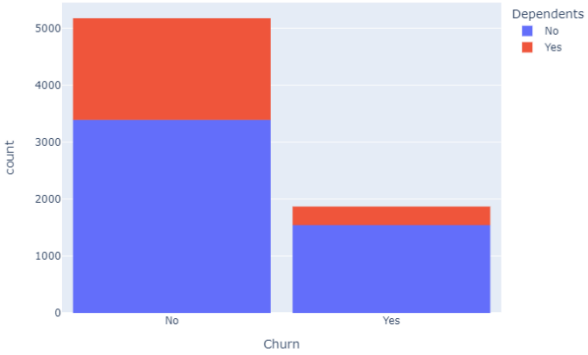


Figure 1.5 - Dependents - Churn Bar Chart

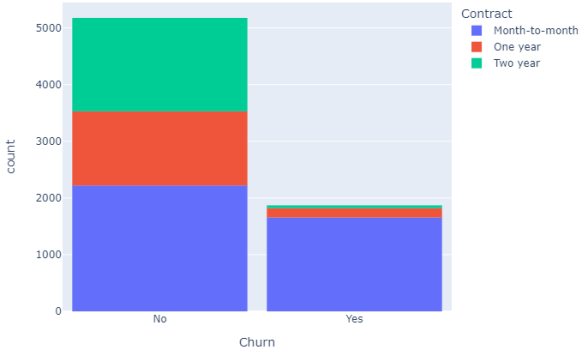


Figure 1.7 - Contract - Churn Bar Chart

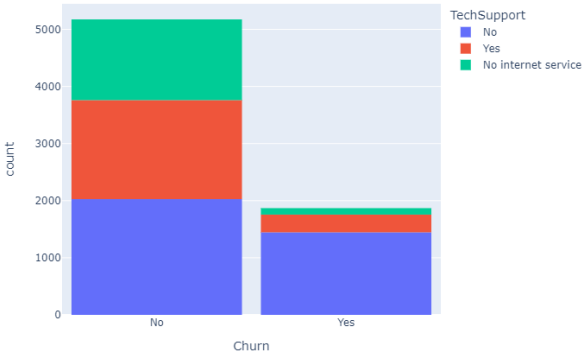


Figure 1.6 - TechSupport - Churn Bar Chart

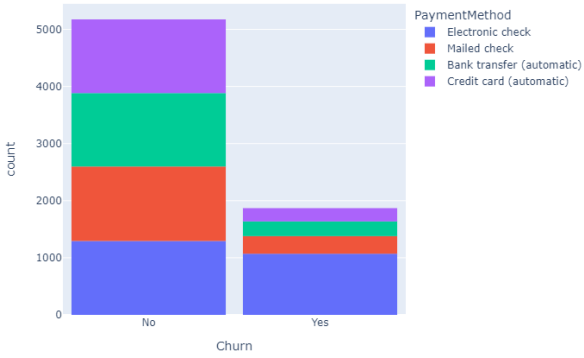


Figure 1.8 - PaymentMethod - Churn Bar Chart

A number of correlations can be observed from these tables. Often not having an opportunity at all produces churn. The most notable ones are month-to-month contrat, electronic check payment method, no techsupport and dependent.

2. Applied Data Preparation Techniques

2.1 Dimensionality Reduction

When columns were examined, can be seen there are 4 main types of columns. The churn column is the target feature, services that each customer was signed up for, account information of the customer and demographic information about the customer.

There were as many as diverse values numbers of customers, that situation was going to affect the prediction algorithms. Therefore, the *customerID* information has been removed as vertical data reduction.

2.2 Data Preprocessing

In the first phase of the study, duplicated rows were checked, and no such values were found on the dataset. Also, no null values were found when examining the data. While preparing the data quality report, it was noticed that the mode value of the TotalCharges column is ' ', that is, an empty string. And, it was detected that there are 11 entities with this empty string value. In order to solve this problem, it was preferred to fill in the blank value instead of the reduction techniques, and the attributes with this empty character were replaced with the median value of the relevant column.

Feature	Count	Card	Mode	Mode Freq	Mode(%)	2nd Mode	2nd Mode Freq	2nd Mode(%)
customerID	7043	7043	0002-ORFBO	1	0.01	5046-NUHWD	1	0.01
gender	7043	2	Male	3555	50.48	Female	3488	49.52
Partner	7043	2	No	3641	51.70	Yes	3402	48.30
Dependents	7043	2	No	4933	70.04	Yes	2110	29.96
PhoneService	7043	2	Yes	6361	90.32	No	682	9.68
MultipleLines	7043	3	No	3390	48.13	Yes	2971	42.18
InternetService	7043	3	Fiber optic	3096	43.96	DSL	2421	34.37
OnlineSecurity	7043	3	No	3498	49.67	Yes	2019	28.67
OnlineBackup	7043	3	No	3088	43.84	Yes	2429	34.49
DeviceProtection	7043	3	No	3095	43.94	Yes	2422	34.39
TechSupport	7043	3	No	3473	49.31	Yes	2044	29.02
StreamingTV	7043	3	No	2810	39.90	Yes	2707	38.44
StreamingMovies	7043	3	No	2785	39.54	Yes	2732	38.79
Contract	7043	3	Month-to-month	3875	55.02	Two year	1695	24.07
PaperlessBilling	7043	2	Yes	4171	59.22	No	2872	40.78
PaymentMethod	7043	4	Electronic check	2365	33.58	Mailed check	1612	22.89
TotalCharges	7043	6531		11	0.16	20.2	11	0.16
Churn	7043	2	No	5174	73.46	Yes	1869	26.54

Figure 2.9 - Data Quality Report for Categorical Features

It can be easily seen that categorical data are in the majority of the features in the dataset. When we examine these categorical data closely, 8 of them have 2 values (e.g. no, yes, and female, male) and 10 of them have 3 values. In order to use this categorical data in mathematical algorithms, it had to be converted into numerical data, and some encoding techniques were applied to do this. In order to encode features with 2 values without generating extra columns, we assign ‘1’ to ‘Yes’ and ‘male' values, ‘0’ is assigned to ‘no’ and ‘Female’ values (0 and 1 values are assigned to these features alphabetically.) When converting features with 3 and 4 values to numeric data, One hot encoder from the scikit-learn library was used. While performing this encoding, the features were deleted from the dataset and replaced with 3 or 4 new columns representing each possible value. Finally, after all the encoding operations are done, the data set had has 41 columns.

After the operations with the categorical data are finished, when the numeric data is considered, it is seen that each of them has very different value ranges. These differences are clearly visible in the data quality report, and for these numeric features to be of equal importance in mathematical algorithms, they all had to have the same range of values. To do this, min-max normalization was used on the relevant features. In this way, all the features have new values between 0 and 1.

	count	mean	std	min	25%	50%	75%	max	card
tenure	7043.0	32.371149	24.559481	0.00	9.000	29.000	55.00	72.00	73.0
MonthlyCharges	7043.0	64.761692	30.090047	18.25	35.500	70.350	89.85	118.75	1585.0
TotalCharges	7043.0	2281.916928	2265.270398	18.80	402.225	1397.475	3786.60	8684.80	6531.0

Figure 2.10 – Numeric Features before the Normalization

	count	mean	std	min	25%	50%	75%	max	card
tenure	7043.0	0.449599	0.341104	0.0	0.125000	0.402778	0.763889	1.0	73.0
MonthlyCharges	7043.0	0.462803	0.299403	0.0	0.171642	0.518408	0.712438	1.0	1585.0
TotalCharges	7043.0	0.261149	0.261397	0.0	0.044245	0.159090	0.434780	1.0	6531.0

Figure 2.2 – Numeric Features after the Normalization

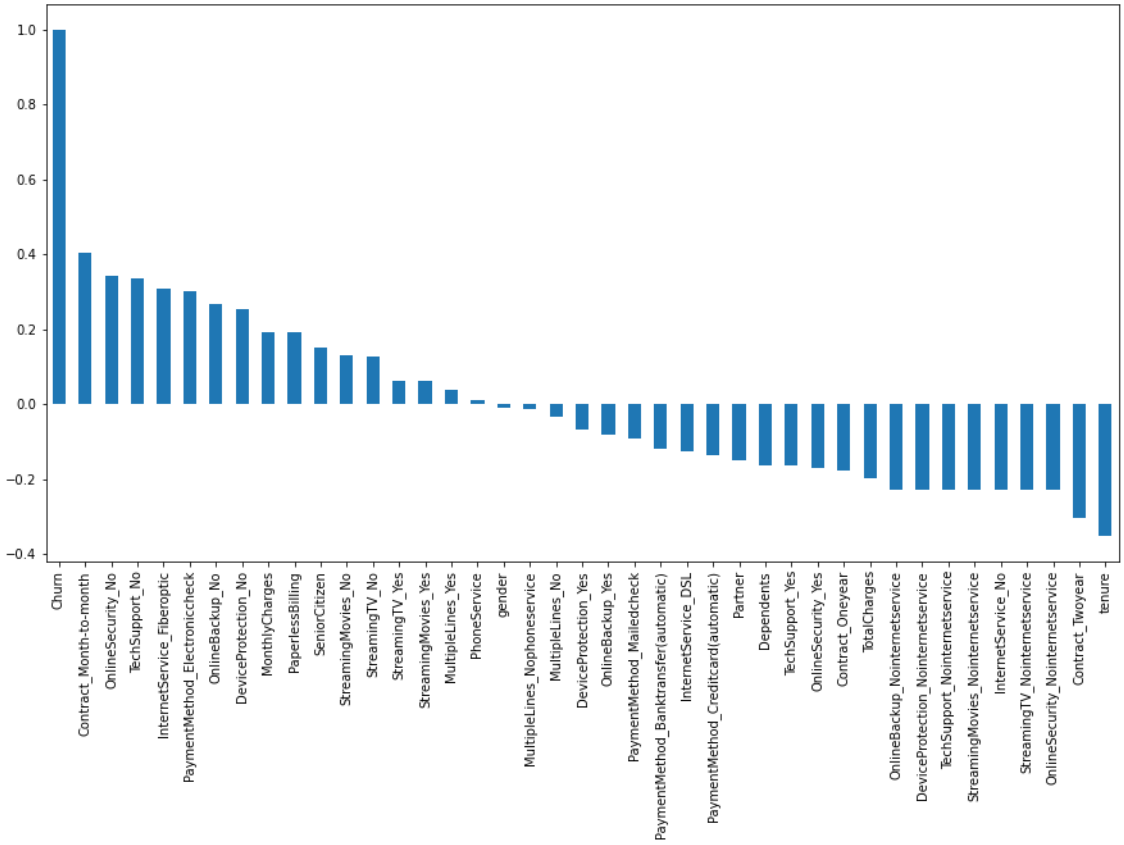
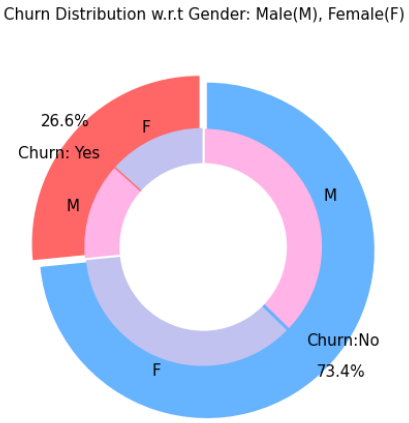


Figure 2.3 – Correlations of features with the target feature

3. Obtained Results

Two different test techniques were used. These tests were applied to many algorithms individually. Also, as seen from figure x, due to the uneven distribution of the target feature, oversampling was applied. Algorithms will be divided as before oversampling and after oversampling.



3.1. 90% Train, 10% Test Method

3.1.1. Support Vector Machines

Support vector machines are a learning algorithm that needs labeled data and can be used for classification and regression problems. SVMs are efficient in high dimensional spaces and memory usage. Our problem needs binary classification, it also has many attributes, for such reasons the use of Support vector machines will be effective.

- Before SMOTE

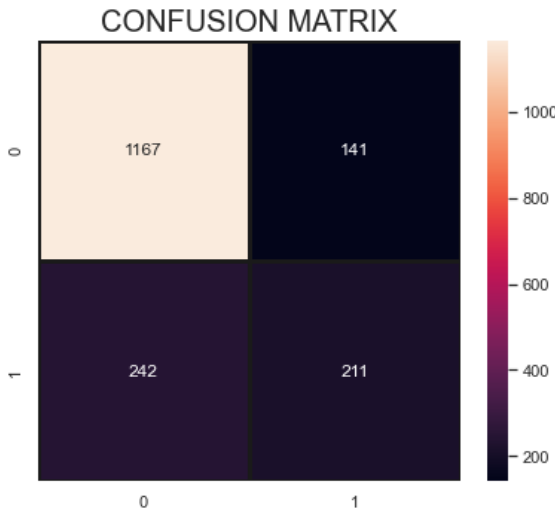


Figure 3.1– Confusion Matrix of SVM before SMOTE

	precision	recall	f1-score	support
0.0	0.83	0.89	0.86	1308
1.0	0.60	0.47	0.52	453
accuracy			0.78	1761
macro avg	0.71	0.68	0.69	1761
weighted avg	0.77	0.78	0.77	1761

```
R2 Score:  -0.1382880693440265
Mean Absolute Error:  0.2174900624645088
Mean Squared Error 0.2174900624645088
```

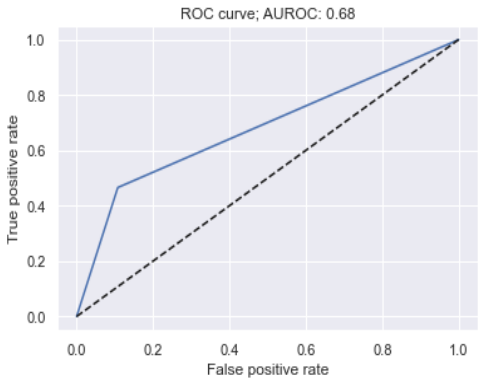


Figure 3.2 – ROC Curve of SVM before SMOTE

Figure 3.3 – Classification Metrics of SVM before SMOTE

- After SMOTE

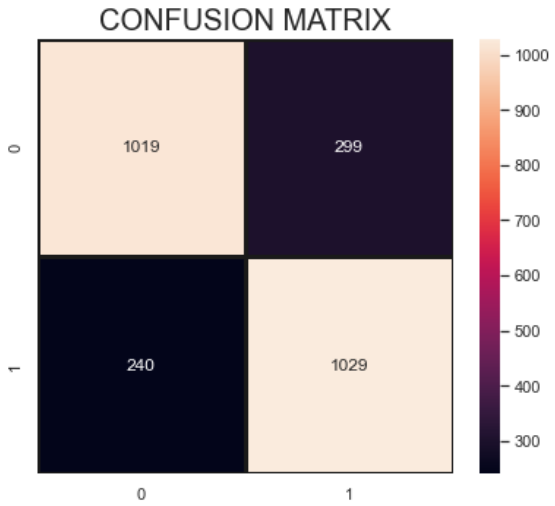


Figure 3.4 – Confusion Matrix of SVM after SMOTE

```
R2 Score:  0.26838847694108725
Mean Absolute Error:  0.1828372632392733
Mean Squared Error 0.1828372632392733
```

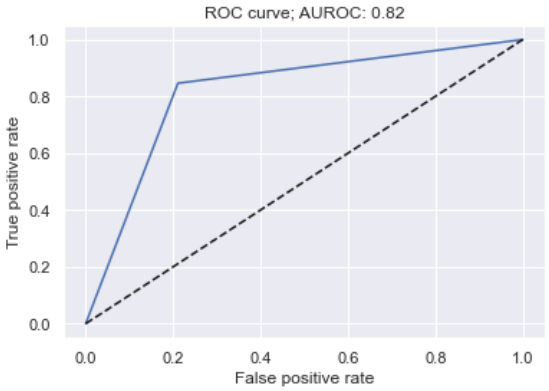


Figure 3.6– ROC Curve of SVM after SMOTE

	precision	recall	f1-score	support
0.0	0.81	0.77	0.79	1318
1.0	0.77	0.81	0.79	1269
accuracy			0.79	2587
macro avg	0.79	0.79	0.79	2587
weighted avg	0.79	0.79	0.79	2587

Figure 3.7 – Classification Metrics of SVM after SMOTE

3.1.1 KNN Nearest Algorithm

The K-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems. Similarities are established with distance calculations. Then, the most appropriate k value is selected according to these similarities. The K value is the value that decides the closest number of instances to be selected. After selecting the K value, for example, if the value is 3, the closest 3 values are taken and output is given according to the ratios of these values.

- Before SMOTE

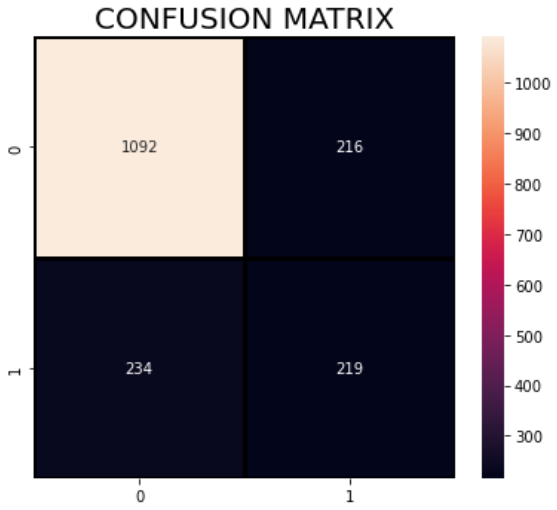


Figure 3.8 – Confusion Matrix of KNN before SMOTE

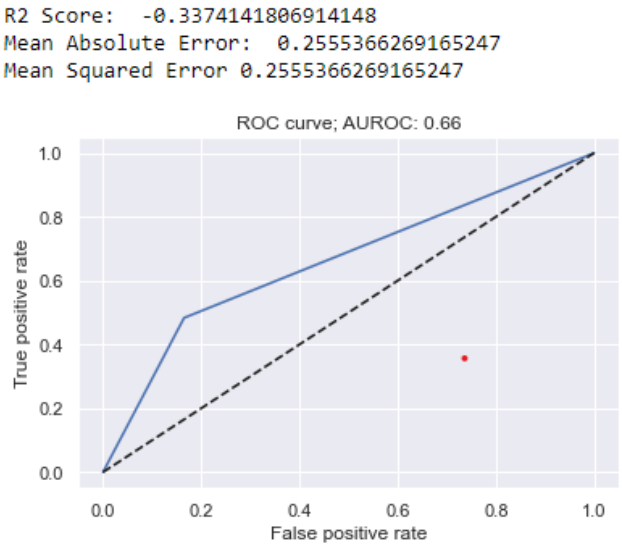


Figure 3.9 – ROC Curve of KNN before SMOTE

	precision	recall	f1-score	support
0.0	0.82	0.83	0.83	1308
1.0	0.50	0.48	0.49	453
accuracy			0.74	1761
macro avg	0.66	0.66	0.66	1761
weighted avg	0.74	0.74	0.74	1761

Figure 3.10 – Classification Metrics of KNN before SMOTE

- After SMOTE

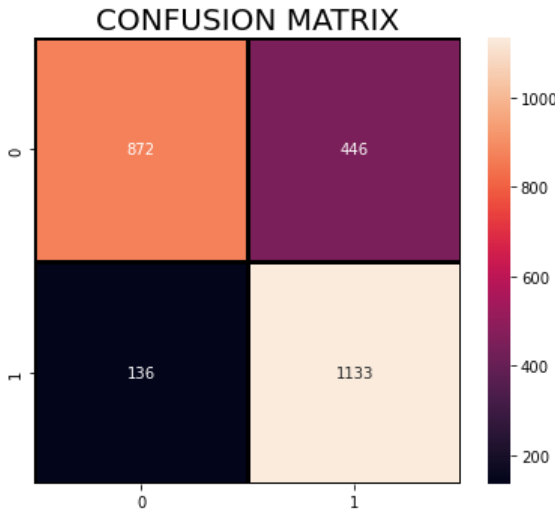


Figure 3.11 – Confusion Matrix of KNN after SMOTE

```
R2 Score:  0.11371373633666593
Mean Absolute Error:  0.22149207576343255
Mean Squared Error 0.22149207576343255
```

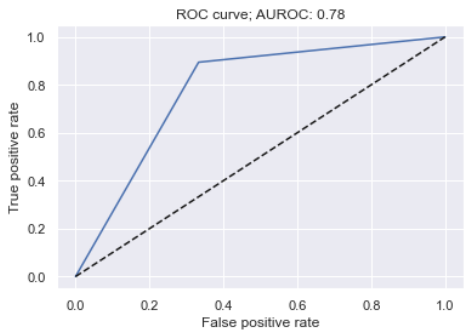


Figure 3.12 – ROC Curve of KNN after SMOTE

	precision	recall	f1-score	support
0.0	0.87	0.67	0.75	1318
1.0	0.72	0.89	0.80	1269
accuracy			0.78	2587
macro avg	0.79	0.78	0.78	2587
weighted avg	0.80	0.78	0.78	2587

Figure 3.13 – Classification Metrics of KNN after SMOTE

3.1.2 Decision Trees

It is important that this node is the most decisive node in the dataset. Some algorithms are run to find the root node and these algorithms generate values over some parameters such as the entropy and information gain of the features depending on executing what algorithm of the decision tree. And, determines the distribution of the nodes accordingly. This process continues until all leaf nodes have a single label. For example, the main purpose of creating a Decision tree in the ID3 tree algorithm is to produce the tree with the shallowest and most consistent information.

- Before SMOTE

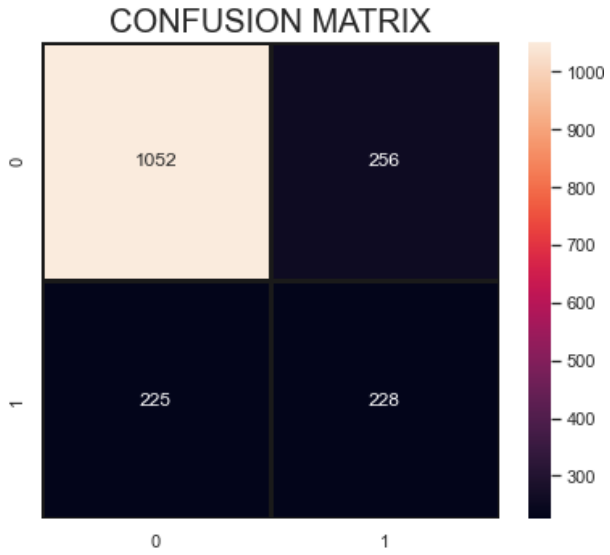


Figure 3.14 – Confusion Matrix of DT before SMOTE

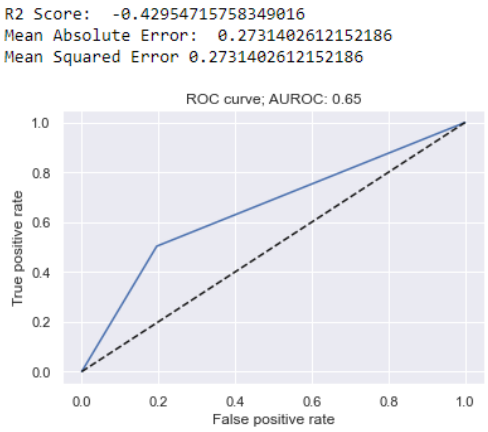


Figure 3.16 – ROC Curve of DT before SMOTE

	precision	recall	f1-score	support
0.0	0.82	0.80	0.81	1308
1.0	0.47	0.50	0.49	453
accuracy			0.73	1761
macro avg	0.65	0.65	0.65	1761
weighted avg	0.73	0.73	0.73	1761

Figure 3.15 – Classification Metrics of DT before SMOTE

- After SMOTE

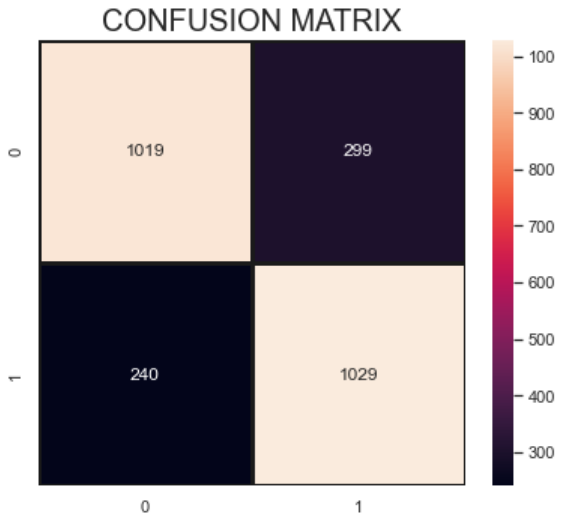


Figure 3.17 – Confusion Matrix of DT after SMOTE

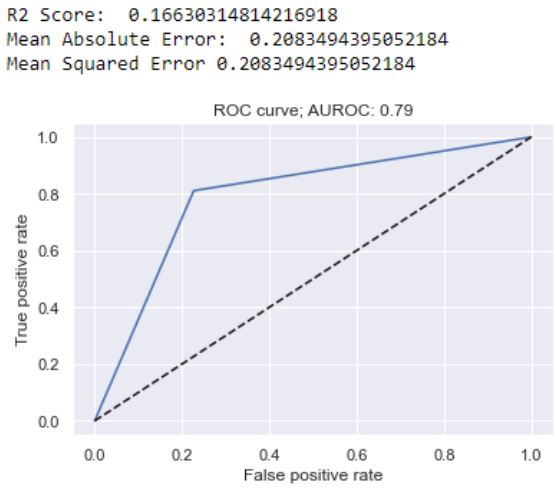


Figure 3.19 – ROC Curve of DT after SMOTE

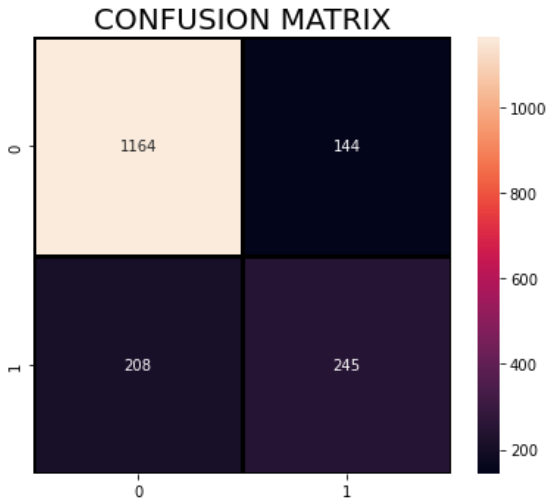
	precision	recall	f1-score	support
0.0	0.81	0.77	0.79	1318
1.0	0.77	0.81	0.79	1269
accuracy			0.79	2587
macro avg	0.79	0.79	0.79	2587
weighted avg	0.79	0.79	0.79	2587

Figure 3.18 – Classification Metrics of DT after SMOTE

3.1.3 Logistic Regression

Logistic regression is another algorithm to be used to develop the model with the highest accuracy. In this technique, which will be used for binary classification, the main purpose is to create a mathematical prediction function. In order to determine the parameters of this function, reference values used to reach the optimum value are assigned first. In order to reach optimum parameter values, estimation is improved by using additional methods such as Loss Error Function, Cost Function and gradient descent.

- Before SMOTE



	precision	recall	f1-score	support
0.0	0.85	0.89	0.87	1308
1.0	0.63	0.54	0.58	453
accuracy			0.80	1761
macro avg	0.74	0.72	0.73	1761
weighted avg	0.79	0.80	0.79	1761

Figure 3.20 – Confusion Matrix of LG before SMOTE

```
R2 Score:  -0.04615509245195115
Mean Absolute Error:  0.19988642816581487
Mean Squared Error 0.19988642816581487
```

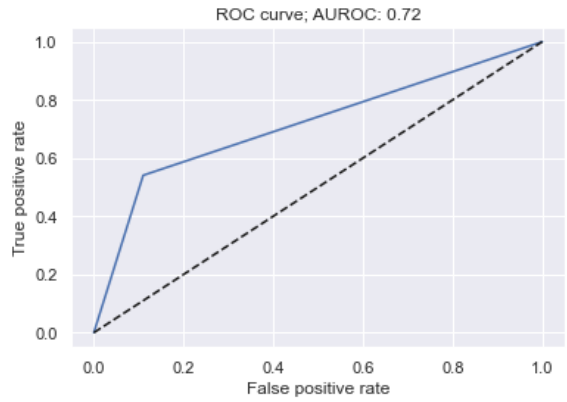


Figure 3.22 – ROC Curve of LG before SMOT

Figure 3.21 – Classification Metrics of LG before SMOTE

- After SMOTE

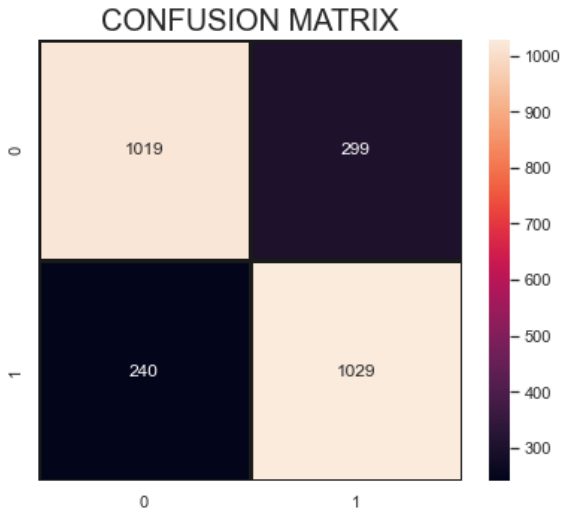


Figure 3.22 – Confusion Matrix of LG after SMOTE

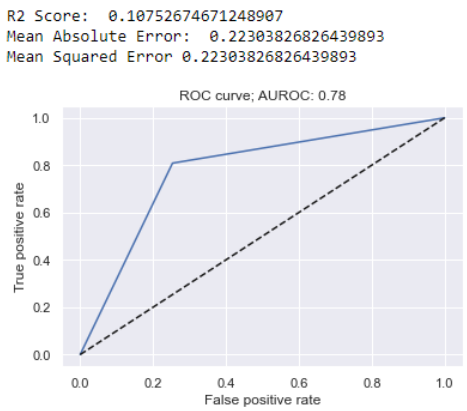


Figure 3.23 – Confusion Matrix of LG after

SMOTE

	precision	recall	f1-score	support
0.0	0.80	0.75	0.77	1318
1.0	0.75	0.81	0.78	1269
accuracy			0.78	2587
macro avg	0.78	0.78	0.78	2587
weighted avg	0.78	0.78	0.78	2587

Figure 3.24 – ROC Curve of LG after SMOTE

	Accuracy	**Presicion (Churn=Yes)**	**Recall (Churn=Yes)**	**F1 – Score (Churn=Yes)**
SVM	**0.79**	**0.77**	**0.81**	**0.79**
LG	**0.78**	**0.75**	**0.81**	**0.78**
KNN	**0.78**	**0.72**	**0.89**	**0.80**
DT	**0.79**	**0.77**	**0.81**	**0.79**

Figure 3.25 – Comparison of the algorithms due to classification metrics

In this study, in which customer behavior analysis was performed, the parts where the churn attribute of the customers is equal to the 'Yes' value is the focus of this study. Because, after predicting that the customer will leave the company, the company must take action before losing the customer. Therefore, even if the churn property is 'No', guessing 'Yes' does not cause a huge loss. However, estimating that the customer will not be lost while losing will cause the company to lose the customer without taking action. For these reasons, the recall value is important for this study, as well as the accuracy value.

3.2 N-Fold Cross Validation (n=10)

This method first shuffles the dataset, then divides it into k-numbers, that is, 10 pieces, and in each iteration, 1 piece of the dataset is used for testing and 9 pieces are used for training. In this way, each part of the dataset is used for both testing and training. Finally, 10 models are used for each algorithm and the average accuracy value is specified.

Algorithm	**Average Accuracy**
LogisticRegression	0.75
MLPClassifier	0.80
GaussianNB(),	0.76
SGDClassifier	0.75
KNeighborsClassifier	0.78
DecisionTreeClassifier	0.77
RandomForestClassifier	0.85
Support Vector Machine	0.80
AdaBoostClassifier	0.80
XGBClassifier	0.84

Figure 3.26 – Average accuracy of the algorithms

4. Conclusion

In this study, in which customer behavior analysis is performed, the parts where the loss attribute of the customers is equal to the value of 'Yes' is the focus of this study. Because after predicting that the customer will leave the company, the company must take action before losing the customer. So even if the churn property is 'No', guessing 'Yes' won't cause a huge loss. However, estimating that the customer will not lose while losing will cause the company to lose its customer without taking any action. For these reasons, the recall value is as important as the accuracy value for this study. When various algorithms are run, the differences in the evaluation criteria are noticeable. The algorithms most suitable for our study are as follows. In addition, when cross validation and normal testing are examined, it seems more logical to apply cross validation. Because correct evaluation values are important and more accurate evaluation rates can be found with more different tests.