University of Messina

A blue and white logo

Description automatically generated with low confidence

Bachelor of Data Analysis

ACADEMIC YEAR - 2023/2024

Machine Learning

(Project report)

Supervisor: Student:

Prof. Giacomo Fiumara Sahil Nakrani

**Table of Content**

* Introduction
* Understanding the Dataset
* Data Preprocessing
* EDA ( Exploratory Data Analysis )
* Feature Engineering
* Modeling and Evaluation
* Discussion
* Conclusion

**Introduction**

In the modern business landscape, retaining customers is just as important as acquiring new ones. Customer churn, which refers to the loss of clients or customers, is a critical metric for companies, especially in highly competitive markets. Predicting customer churn can help organizations take pre-emptive actions to retain customers, thereby enhancing profitability and growth. This project aims to predict customer churn using various machine learning models. The process includes data preprocessing, exploratory data analysis (EDA), feature engineering, model training, evaluation, and tuning to achieve the best possible predictive performance.

**Understanding the Dataset**

The dataset consists of 3749 entries with 17 columns. ( But here we are not considering 2 columns which are ‘Unnamed: 0’ and ‘CustomerId’. Because which are just indexes for the entries. )

In this dataset, there is two columns ‘Unnamed: 0’ and ‘CustomerID’ which are only showing the index. So, I have removed it to not have unnecessary feature.

A black background with yellow text

Description automatically generated

**Basic Information and Statistical Details**

A screenshot of a computer

Description automatically generated

A screenshot of a black screen

Description automatically generated

**Data Preprocessing**

**Handling the missing values**

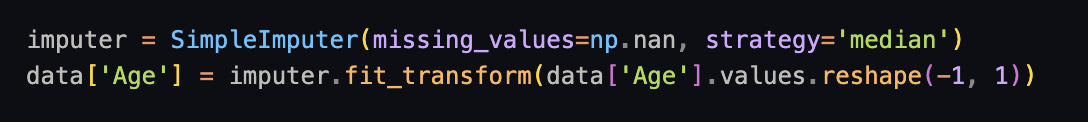
We utilize both the functions “isnull()” to inspect null values.

**A computer screen shot of a black background

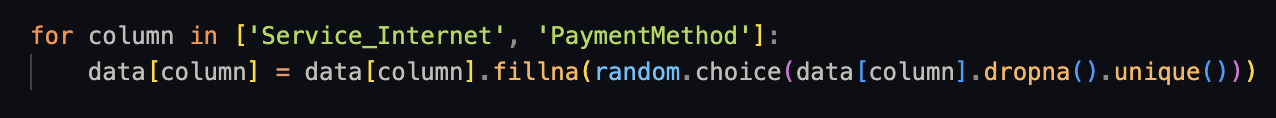
Description automatically generated**

Here, feature ‘Age’ is numerical feature and [ ‘Service\_Internet’, ‘PaymentMethod’ ] are categorical features.

* Replacing the numerical missing values with the median of the column.

****

* Replacing the categorical missing values with the random instance of the column.

****

Here, we can also replace categorical feature’s missing values with mode ( most frequent instance ) but we have replaced with random instance. Just because already mode is having majority of distribution and as per me, it is not good to put again the mode to let the model to rely on single instance ( category ).

**Label Encoding for the categorical features**

For the Exploratory Data Analysis (EDA) and for the model training part. For example terraforming the values of categorical value ‘yes’ and ‘no’ to 1 and 0.

First, I have listed all the categorical features. I’m using ‘**LabelEncoder()’** function to encode the categories of the features.

**A screen shot of a computer

Description automatically generated**

**Handling the Outliers**

Outliers are datapoints that deviate significantly from the rest of the dataset and can have a substantial impact on analysis.

A white graph with black lines

Description automatically generated

In the above box-plot, you can see there are outliers in two right sided features which are ‘MonthlyCharges’ and ‘TotalCharges’. And we are going to handle it with the technique called inter quartile range.

So, first I have made the function to handle the outliers and then I will apply it on all the numerical features and I will save the index of the outlier.

A computer screen shot of a program

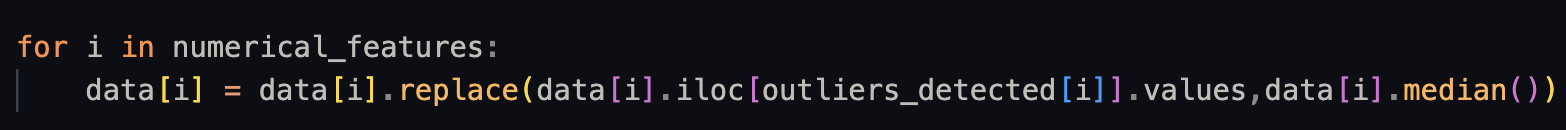
Description automatically generated

For example :

**A black background with white numbers

Description automatically generated**

Then, I have just replaced that detected outliers with the median of the column. We can also remove the outliers, but since our dataset is small so it is better to replace it with middle value not to remove.

****

Let’s see now the box-plot.

**A white graph with black lines

Description automatically generated**

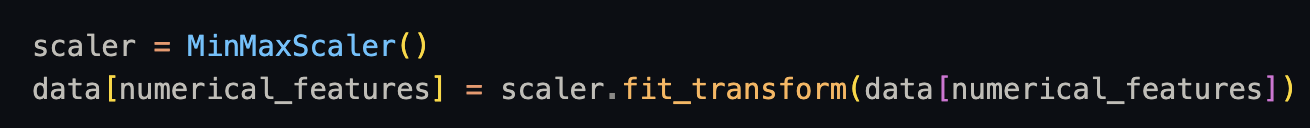
**Scaling the Feature**

Before scaling the data, features are not in one range. They are in different range which is not good for training phase.

**A white graph with blue lines

Description automatically generated**

here, I’m using a MinMaxScaler() to scale the data not a Standard Scaler. Because we have some features which is showing time period and standard scaler is converting the range between -1 and 1. Negative values are not usual for the time period. That’s why here I have used MinMaxScaler to have range between 0 and 1.



Here is the box-plot after scaling the numerical features.

**A diagram of a diagram

Description automatically generated with medium confidence**

**EDA ( Exploratory Data Analysis )**

Exploratory Data Analysis (EDA) is a crucial step in understanding the dataset and uncovering underlying patterns that inform feature engineering and model building. This section provides a detailed analysis based on the graphs and plots generated during the EDA process.

**Overview of the Dataset**

The dataset consists of 3749 entries with 17 columns, excluding 'Unnamed: 0' and 'CustomerID', which are irrelevant for analysis.

**Distribution of Numerical Features**

First, we are going to make a distribution graph for the numerical features using histogram.

**A graph of different numbers

Description automatically generated with medium confidence**

By this histogram, we can see the distribution, frequencies, and also skewness in the distribution.

For example, the histogram for 'TotalCharges' reveals a left-skewed distribution, indicating that most customers have total charges less than 3000. This skewness suggests that a large portion of the customer base incurs relatively lower total charges.

**Analysis of Categorical Features**

Categorical features were analyzed using bar plots to visualize the frequency of each category. The bar plot indicates that a significant number of customers involved in specific category.

A computer screen with text

Description automatically generated

A group of blue rectangular bars

Description automatically generated

**Relationship Between Features and Target Variable**

Now here is the relation between a numerical features and the target feature which is Churn. This relation is visualized with the box-plot.

**A diagram of a box plot

Description automatically generated**

**A diagram of a box plot

Description automatically generated**

**A diagram of a box plot

Description automatically generated**

Here, we can see some relationship and trends between the feature and target. For example, the box plot illustrates that customers with higher monthly charges are more likely to churn. This trend highlights 'MonthlyCharges' as a significant predictor of churn.

**Pairwise Scatter Plot**

Also I have visualized the relation with the help of pairwise scatter plot. Where we can identify the relational trends, actual trends in the data, and also distribution and outliers. Here, I have mentioned the pairwise scatter plot below.

**A screenshot of a graph

Description automatically generated**

Then, let’s create a heatmap of the correlation matrix. This is very important step of EDA because, by correlation matrix, we can identify the relation between the feature and other features and asl between the feature and target feature.

The scatter plot between 'MonthlyCharges' and 'TotalCharges' and also between ‘TotalCharges’ and ‘Tenure’ shows a positive linear relationship, confirming the correlation analysis. This plot also helps in visualizing the distribution of data points and identifying potential outliers.

**Correlation Analysis**

The heatmap of the correlation matrix visualizes the relationships between features. For example, 'MonthlyCharges' and 'TotalCharges' show a strong positive correlation, indicating that customers with higher monthly charges tend to accumulate higher total charges over time. This heatmap also helps in identifying multicollinearity.

By correlation matrix, we can also look the percentage of dependencies of the features on other features. Because, correlation coefficient ranges between -1 and 1. Where 1 mean strongly positive correlation and -1 means strongly negative correlation.

A graph of a number of data

Description automatically generated with medium confidence

**Feature Engineering**

I have already scaled the features, and also have put the labels for the categorical features.

Feature engineering involves creating new features from existing ones to improve model performance. This process is guided by domain knowledge and the insights gained from EDA.

Combine Existing Features :

* **MonthlyCharges\_to\_Tenure :** identifies high-cost users who might be at risk of churn.
* **TotalCharges\_to\_Tenure :** highlights customers who have spent a lot over their tenure and might be more sensitive to service issues.

Domain Knowledge :

* **TotalServices :** helps to identify customers having reliant on the company’s services and might churn if dissatisfied.
* **isSeniorCitizen :** senior citizens might have different churn behaviour.

**A screen shot of a computer program

Description automatically generated**

**Modelling and Evaluation**

Frist we are going to split the data into target variables and other features.

A black background with yellow and green text

Description automatically generated

I have tried to train model now, but it is overfitting. May be because we have very less data and three features with high missing values. So, that’s why I have used feature selection technique to select most important features.

**A screen shot of a computer code

Description automatically generated**

So, we are selecting 10 most important features from our dataset to train the model on.The next step involves training various machine learning models to predict customer churn. The dataset was split into training (70%) and testing (30%) sets using the train\_test\_split() function with random\_state=42 to ensure reproducibility.

****

**A black background with white text

Description automatically generated**

**Model Selection**

We tested four models for this task:

* Random Forest Classifier
* Logistic Regression Classifier
* Gradient Boost Classifier
* Decision Tree Classifier

But first, I’m making a common function to train all the models by that function. Which is mentioned below.

A computer screen shot of a program code

Description automatically generated

Here, I’m also using a GridSearchCV() function to split the dataset in several part and train on each split to get best model and best parameters which will suitable to improve the accuracy of the model.

And by this function as you can see, I’m printing best parameters, classification report, confusion matrix, and returning the best model.

**Evaluation Metrics**

A computer screen shot of text

Description automatically generated

By the upper function, we can see the importance of the features for that specific model to train the model. And by below mentioned function, by checking the accuracy of the train and test, it is overfitting or not.

A screen shot of a computer code

Description automatically generated

And here is the example that how I’m training the model with some range of parameters, and with the help of functions which I made before. This is the example of the random forest classifier and below this I have mentioned result of this model. Like this I have trained other models and got the results which you can see in the code file.

A computer screen shot of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

A chart of confusion matrix

Description automatically generated

A black background with white text

Description automatically generated

A screenshot of a computer

Description automatically generated

But in the Logistic Regression, I have noticed that it is giving reliable result.

A screenshot of a computer screen

Description automatically generated

A chart of confusion matrix

Description automatically generated

A black background with white text

Description automatically generated

**Discussion**

The project revealed several key insights into predicting customer churn:

* **Data Quality:** Handling missing values and outliers is crucial for improving model performance. The chosen methods for imputation and outlier treatment maintained data integrity and variability.
* **Feature Engineering:** Creating new features based on domain knowledge significantly enhanced the models' ability to capture important patterns in the data.
* **Model Performance:** While all models showed some degree of overfitting, cross-validation and hyperparameter tuning helped mitigate this issue but here we are seeing overfitting after using those techniques so, it might be because of the dataset quality. The Random Forest model, in particular, demonstrated a balanced trade-off between bias and variance, making it a top performer.
* **Business Insights:** The findings provide valuable insights into customer behaviour. For instance, features like 'MonthlyCharges\_to\_Tenure' and 'TotalServices' highlight high-risk customer segments, enabling targeted interventions to reduce churn.

And in this project, I have tried to use many techniques to avoid and fix the overfitting problem. But, it was not fixable. So, as per my consideration and experience, there is no value of model if it is very perfect until you have good data. So, it is also very important to have a good related data structure of our dataset.

**Conclusion**

In this project, we undertook a comprehensive approach to predict customer churn using various machine learning models. Starting with data preprocessing and EDA, we gained an in-depth understanding of the dataset's characteristics and patterns. Feature engineering allowed us to create additional features that improved model performance.

We implemented and evaluated multiple machine learning models, addressing overfitting through feature selection, cross-validation, and hyperparameter tuning. The Random Forest model emerged as a robust predictor of customer churn, offering a balanced performance.

These findings offer strategic insights into customer behaviour, enabling organizations to implement targeted interventions to reduce churn and enhance customer retention. Future work could involve expanding the dataset, exploring more advanced machine learning techniques, and incorporating additional features to further improve prediction accuracy.