CSE422: Artificial Intelligence

Section: 17

Project Report on Telecom Customer Churn Prediction using Machine Learning

Submitted by:

Name	ID
Ahmed Rakin	23301039
Amreen Hassan	20201211

1. Introduction

This project's goal is to use machine learning techniques to forecast client attrition for a telecom company. Customer churn is the term for customers leaving the company's service, which has a negative effect on income for the businesses. Businesses can take preemptive measures like providing discounts, enhancing customer service, or changing subscription plans by anticipating churn. This project's goal is to find churn patterns and assess the efficacy of supervised and unsupervised learning learning algorithms..

2. Dataset Description:

The Telco Customer Churn dataset contains customer-level data collected from a telecommunications company and is stored in a comma-separated values (CSV) file named *telco_customer_churn.csv*. The dataset includes demographic information, service usage details, billing attributes, and a churn indicator that shows whether a customer has discontinued the service. The main purpose of this dataset is to analyze customer behavior and identify important factors that lead to customer churn.

Number of Features:
The dataset contains a total of 21 features. Out of these, 20 features are independent input variables, and one feature, **Churn**, is the target (output) variable. The input features describe customer demographics, subscription information, service usage, and payment behavior, while the target feature indicates the churn status of each customer.

Type of Dataset:
This dataset represents a binary classification problem. The target variable **Churn** has two categorical values, **Yes** and **No**, which indicate whether a customer has churned or not. Since the target variable is categorical and not continuous, the problem cannot be treated as a regression task. Instead, the goal is to classify customers into one of two classes, making the dataset suitable for classification-based machine learning models.

Amount of Data Points:
The dataset consists of 7,043 customer records, where each row represents one customer. When all features are considered together, the dataset contains a total of 147,903 data points, calculated as the number of rows multiplied by the number of columns. This sample size is sufficient for exploratory data analysis and for training classification models.

Types of Features Present in the Dataset:
The dataset includes both numerical and categorical features. The numerical features are few in number and include **SeniorCitizen**, **tenure**, and **MonthlyCharges**, which represent measurable values such as customer age category, length of service, and monthly billing amount. Most of the features are categorical, including **gender**, **Partner**, **Dependents**, **PhoneService**, **InternetService**, **OnlineSecurity**, **TechSupport**, **Contract**, **PaperlessBilling**, **PaymentMethod**, and the target variable **Churn**. The **customerID** feature is used only as an identifier and does not

provide useful information for prediction. Overall, the dataset contains 18 categorical features and 3 numerical features.

Encoding of Categorical Variables:

Yes, encoding of categorical variables is required. Most machine learning algorithms and statistical methods, including correlation analysis, require input data to be in numerical form. Since the dataset contains several text-based categorical features, these variables must be converted into numerical representations using encoding techniques such as one-hot encoding. Encoding allows the machine learning models to correctly process categorical information and ensures meaningful mathematical computations during model training and evaluation.

Correlation of all the features by the help of a plotted heatmap:

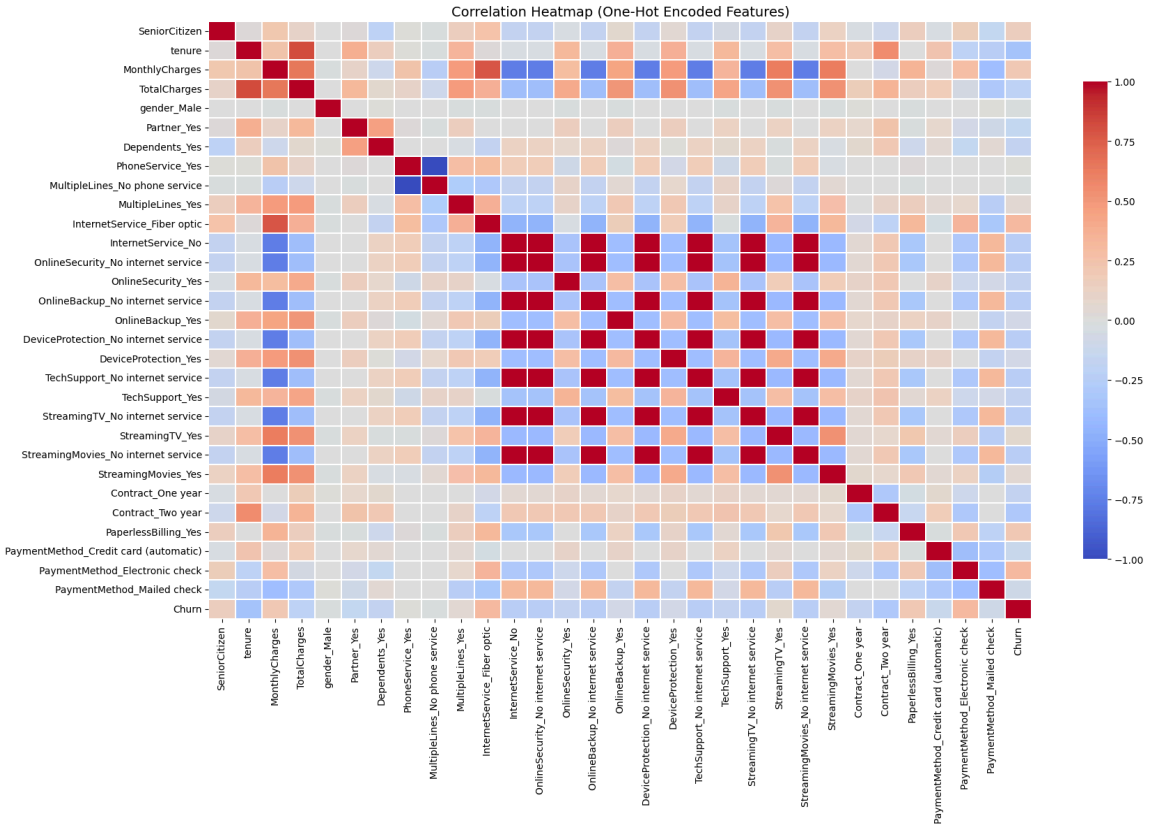
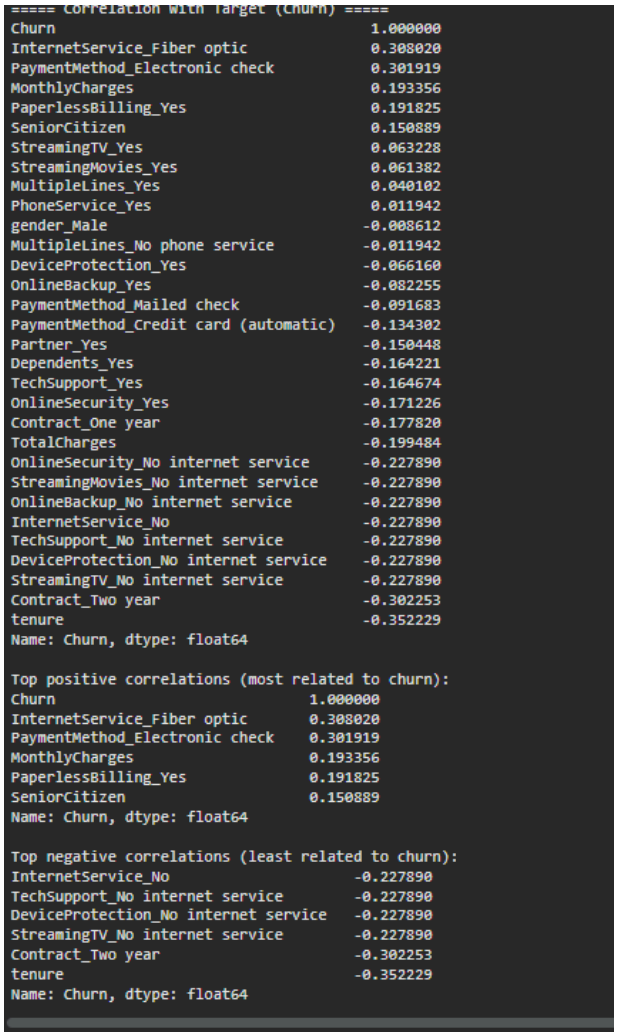


Figure X: Heatmap

From the correlation analysis and heatmap, we can identify which features are most strongly associated with customer churn and the direction of their relationship. The results show that InternetService_Fiber optic, PaymentMethod_Electronic check, MonthlyCharges, PaperlessBilling, and SeniorCitizen have the highest positive correlation with churn, indicating that customers with these characteristics are more likely to leave the service. In contrast, tenure and Contract_Two year show strong negative correlations, meaning customers who have stayed longer or are on long-term contracts are much less likely to churn. The correlation heatmap visually confirms these

patterns and also highlights strong correlations among one-hot encoded service-related features, especially those representing “no internet service,” which is expected since they originate from the same categorical variables. Overall, the correlation test helps highlight key churn drivers and supports feature understanding and selection, while indicating relationships rather than causation.



```
===== Correlation with Target (Churn) =====
Churn                                     1.000000
InternetService_Fiber optic               0.308020
PaymentMethod_Electronic check            0.301919
MonthlyCharges                            0.193356
PaperlessBilling_Yes                      0.191825
SeniorCitizen                             0.150889
StreamingTV_Yes                           0.063228
StreamingMovies_Yes                       0.061382
MultipleLines_Yes                         0.040102
PhoneService_Yes                          0.011942
gender_Male                              -0.008612
MultipleLines_No phone service           -0.011942
DeviceProtection_Yes                     -0.066160
OnlineBackup_Yes                         -0.082255
PaymentMethod_Mailed check               -0.091683
PaymentMethod_Credit card (automatic)    -0.134302
Partner_Yes                              -0.150448
Dependents_Yes                           -0.164221
TechSupport_Yes                          -0.164674
OnlineSecurity_Yes                       -0.171226
Contract_One year                        -0.177820
TotalCharges                             -0.199484
OnlineSecurity_No internet service       -0.227890
StreamingMovies_No internet service      -0.227890
OnlineBackup_No internet service         -0.227890
InternetService_No                       -0.227890
TechSupport_No internet service          -0.227890
DeviceProtection_No internet service     -0.227890
StreamingTV_No internet service          -0.227890
Contract_Two year                        -0.302253
tenure                                   -0.352229
Name: Churn, dtype: float64

Top positive correlations (most related to churn):
Churn                            1.000000
InternetService_Fiber optic      0.308020
PaymentMethod_Electronic check   0.301919
MonthlyCharges                   0.193356
PaperlessBilling_Yes             0.191825
SeniorCitizen                    0.150889
Name: Churn, dtype: float64

Top negative correlations (least related to churn):
InternetService_No                      -0.227890
TechSupport_No internet service         -0.227890
DeviceProtection_No internet service    -0.227890
StreamingTV_No internet service         -0.227890
Contract_Two year                       -0.302253
tenure                                  -0.352229
Name: Churn, dtype: float64
```

Pipeline: We used a pipelined approach where preprocessing steps such as encoding, scaling, and class imbalance handling were applied sequentially and consistently during training. This prevents data leakage and ensures fair evaluation on unseen data

Exploratory Data Analysis (EDA):

The data types reveal that the dataset contains a mixture of **categorical and numerical features**. Most of the features are categorical in nature, such as gender, Partner, Dependents, InternetService, Contract, PaymentMethod, and others. The numerical features include

SeniorCitizen, tenure, MonthlyCharges, and TotalCharges. The target variable Churn is encoded in binary form, where 0 represents non-churned customers and 1 represents churned customers, confirming that this is a binary classification problem.

Target Variable Distribution (Class Imbalance Analysis):

To understand the distribution of the target variable, the frequency of churned and non-churned customers was examined. The results show that 5,174 customers did not churn, while 1,869 customers churned. This indicates that the dataset is imbalanced, with a significantly higher number of non-churned customers compared to churned customers.

Such class imbalance is important to consider during model development, as it may bias the learning algorithm toward the majority class. Appropriate evaluation metrics and techniques may therefore be required to handle this imbalance effectively.

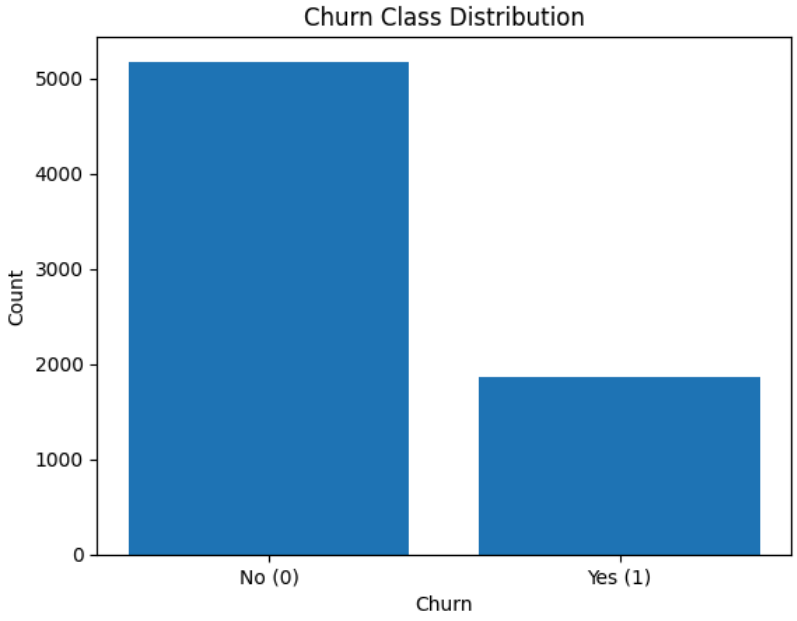


Figure X: Distribution of churned (1) and non-churned (0) customers.

Statistical Summary of Numerical Features: The statistical summary of the numerical features provides insight into the central tendency and variability of the data. The tenure feature ranges from 0 to 72 months, with an average tenure of approximately 32 months. The MonthlyCharges feature varies between 18.25 and 118.75, indicating a wide range of billing amounts among customers. The TotalCharges feature shows higher variance, reflecting cumulative charges over time. The SeniorCitizen feature is binary, where a smaller proportion of customers are senior citizens.

These statistics suggest that customer tenure and billing amounts may play an important role in determining churn behavior.

Missing Value Analysis:

A missing value analysis was conducted to identify incomplete data. The results show that all features are complete except TotalCharges, which contains **11 missing values**. Since the number of missing values is very small relative to the dataset size, appropriate preprocessing techniques such as imputation or row removal can be applied without significantly affecting the overall analysis.

```
=== Missing Values (BEFORE) ===
TotalCharges      11
gender             0
Partner            0
SeniorCitizen      0
Dependents         0
tenure             0
MultipleLines      0
PhoneService       0
OnlineSecurity     0
OnlineBackup       0
dtype: int64

Converted TotalCharges to numeric (Cause: stored as text/blanks).

=== Missing Values (AFTER converting TotalCharges) ===
TotalCharges    11
dtype: int64

=== Missing Rate % (Top) ===
TotalCharges      0.156183
gender            0.000000
Partner           0.000000
SeniorCitizen     0.000000
Dependents        0.000000
tenure            0.000000
MultipleLines     0.000000
PhoneService      0.000000
OnlineSecurity    0.000000
OnlineBackup      0.000000
dtype: float64
```

Relationship Between Tenure and Churn: To further explore the relationship between customer behavior and churn, a boxplot was used to compare tenure values for churned and non-churned customers. The visualization clearly shows that customers who churn tend to have much lower tenure, whereas non-churned customers generally have longer service durations.

This observation indicates a strong relationship between tenure and churn, suggesting that customers are more likely to leave the service during the early stages of their subscription.

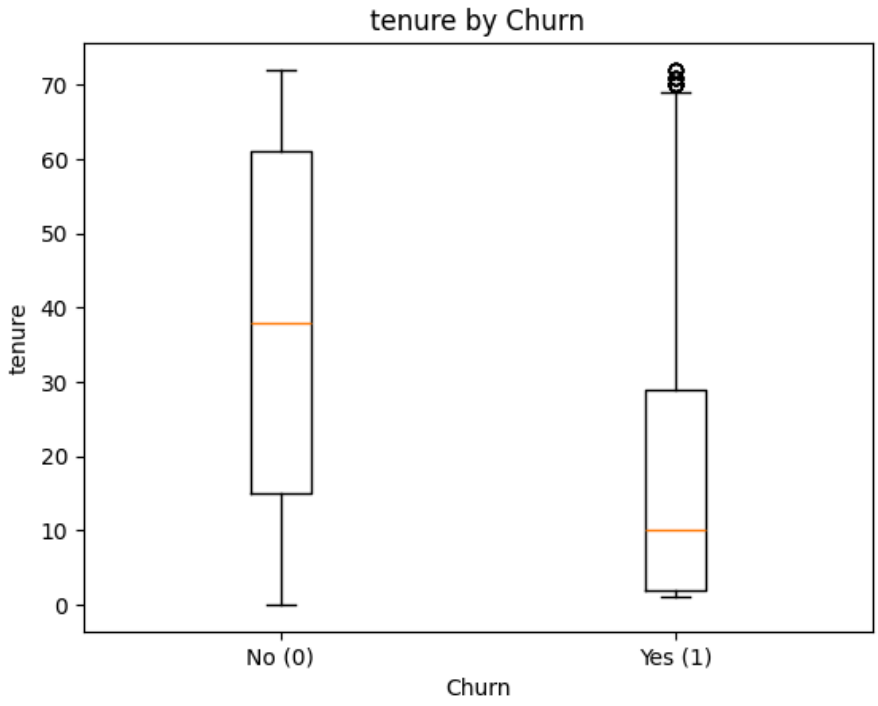


Figure Y: Boxplot showing tenure distribution for churned and non-churned customers.

3. Dataset Pre-processing:

The dataset was subjected to several pre-processing steps to remove inconsistencies and prepare it for effective analysis and machine learning model development. Raw datasets often contain structural and statistical issues that can negatively impact model performance if not addressed. The main issues identified in the dataset and the corresponding pre-processing techniques applied to resolve them are described below.

Null / Missing Values:
During the initial inspection of the dataset, certain features were found to contain missing values. These were identified using the method df.isna().sum(), which helped determine both the presence and extent of missing data across all features. In particular, the **TotalCharges** attribute contained missing values due to incorrect data type representation, where numeric values were stored as text with blank entries.

Solution:
The **TotalCharges** feature was first converted from an object data type to a numeric data type using a coercion method, which automatically converted invalid or blank entries into null values. After this conversion, missing numerical values were imputed using the mean of the respective

feature. Rows containing missing values in the target variable were removed, as such records cannot be used in supervised learning.

Reason:
Missing values can lead to biased model training and unreliable predictions, as most machine learning algorithms cannot handle null values directly. Mean imputation helps preserve the size of the dataset while maintaining statistical consistency. Removing rows with missing target values is necessary because supervised learning models require complete output labels for training.

Categorical Values:
The dataset contains several categorical features, including **gender**, **contract type**, **payment method**, and various service-related attributes. The target variable **Churn** is also categorical. Since most machine learning algorithms require numerical input, categorical data cannot be used directly for model training.

Solution:
The target variable **Churn** was converted into numerical form by mapping its categories to binary values (No → 0, Yes → 1). The remaining categorical input features were transformed using one-hot encoding, which creates separate binary features for each category without implying any ordinal relationship.

Reason:
Machine learning models cannot process text-based data directly. Encoding converts categorical variables into numerical values that models can interpret. One-hot encoding is well suited for nominal categorical data, as it avoids introducing false order or priority among categories and ensures that each category is treated independently.

Feature Scaling:
The dataset includes numerical features such as **tenure**, **MonthlyCharges**, and **TotalCharges**, which exist on different numerical scales. This difference in scale can cause features with larger values to dominate the learning process, potentially leading to biased model performance.

Solution:
Feature scaling was applied using **StandardScaler**, which performs Z-score normalization by transforming numerical features to have a mean of zero and a standard deviation of one. Scaling was applied only to numerical features, as categorical features produced through one-hot encoding do not require scaling.

Reason:
Many machine learning algorithms, particularly distance-based and gradient-based models such as K-Nearest Neighbors, Logistic Regression, and Neural Networks, are sensitive to feature magnitude. Standardization ensures that all numerical features contribute equally to the learning process, which improves model stability, convergence speed, and overall predictive performance.

4. Dataset Splitting

Initially, the dataset showed an imbalance between churned and non-churned customers. Such imbalance can affect how well the model generalizes across different classes.

Solution: The dataset was split into training and testing sets using an 80:20 ratio. Stratified sampling was applied during the split to preserve the original class distribution of the target variable in both subsets.

Reason: Using a separate test set allows for an unbiased evaluation of the model's performance on unseen data. Stratified sampling ensures that both churn and non-churn classes are proportionally represented, preventing biased learning and misleading accuracy results.

5. Model Training and Testing

Evaluation Model:

```
MODEL PERFORMANCE (%)
==================================
Accuracy Precision Recall    F1 ROC_AUC
   80.6%     65.7%  55.9% 60.4%   84.2%

CONFUSION MATRIX
----------------------------------------
                Pred No      Pred Yes
Actual No   926\n(65.7%)   109\n(7.7%)
Actual Yes  165\n(11.7%)  209\n(14.8%)

SUMMARY
--------------------
Churn correctly predicted: 209 (55.9% of actual churn)
Total test samples: 1409
```

This output shows that the model performs well overall in predicting customer churn. It achieved an **accuracy of 80.6%**, meaning most predictions were correct. When the model predicts that a customer will churn, it is correct **65.7%** of the time (precision). It also correctly identifies **55.9%** of all actual churn customers (recall), which means just over half of the churn cases were detected. The **ROC-AUC score of 84.2%** indicates that the model is good at separating churn and non-churn customers. Overall, the model predicts non-churn customers very well and does a reasonable job of identifying churn customers, though some churn cases are still missed.

Supervised Learning:

In this study, three supervised learning models were applied to the customer churn prediction problem: Logistic Regression, Decision Tree, and a Neural Network (Multi-Layer Perceptron). Logistic Regression was used as a baseline linear classifier due to its interpretability and robustness on tabular data. A Decision Tree classifier was employed to capture non-linear relationships between customer attributes and churn behavior. Finally, a Neural Network was included to satisfy the requirement of using a deep learning model and to explore whether a more

complex architecture could improve predictive performance.

Logistic Regression:

```
=== Logistic Regression ===
Accuracy: 0.8055358410220014
ROC-AUC: 0.8418610659019866
Confusion Matrix:
 [[926 109]
 [165 209]]
Classification Report:
              precision    recall  f1-score   support

           0       0.85      0.89      0.87      1035
           1       0.66      0.56      0.60       374

    accuracy                           0.81      1409
   macro avg       0.75      0.73      0.74      1409
weighted avg       0.80      0.81      0.80      1409
```

The Logistic Regression model shows **good overall performance**. It achieved an **accuracy of about 80.6%** and a **ROC-AUC score of 0.84**, indicating a strong ability to distinguish between churn and non-churn customers.

From the confusion matrix, the model correctly predicted most **non-churn customers (Class 0)**, with high precision (0.85) and recall (0.89). This means it is very effective at identifying customers who stay. However, performance on **churn customers (Class 1)** is weaker, with a precision of 0.66 and a recall of 0.56, showing that some churn cases are missed.

Overall, the model is reliable and balanced, performing especially well for non-churn prediction, but it may require further tuning or class-balancing techniques to better capture churn customers.

Decision Tree:

```
=== Decision Tree ===
Accuracy: 0.7913413768630234
ROC-AUC: 0.835092614120747
Confusion Matrix:
 [[926 109]
 [185 189]]
Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.89      0.86      1035
           1       0.63      0.51      0.56       374

    accuracy                           0.79      1409
   macro avg       0.73      0.70      0.71      1409
weighted avg       0.78      0.79      0.78      1409
```

The Decision Tree model achieved an **accuracy of about 79.1%** with a **ROC-AUC score of 0.84**, indicating a good overall ability to separate churn and non-churn customers.

The model performs well in identifying **non-churn customers (Class 0)**, with high recall (0.89) and solid precision (0.83). However, its performance on **churn customers (Class 1)** is weaker,

with a recall of 0.51 and an F1-score of 0.56, meaning that nearly half of the churn cases are not correctly identified.

Overall, the Decision Tree provides reasonable performance but is slightly less accurate than Logistic Regression and struggles more with correctly detecting churn customers.

Neural Network:

```
••• === Neural Network (MLP, Tuned) ===
    Accuracy: 0.7892122072391767
    ROC-AUC: 0.8386292593453719
    Confusion Matrix:
     [[931 104]
     [193 181]]
    Classification Report:
                  precision    recall  f1-score   support

               0       0.83      0.90      0.86      1035
               1       0.64      0.48      0.55       374

        accuracy                           0.79      1409
       macro avg       0.73      0.69      0.71      1409
    weighted avg       0.78      0.79      0.78      1409

    Iterations used: 49
```

The tuned Neural Network (MLP) model achieved an **accuracy of about 78.9%** and a **ROC-AUC score of 0.84**, showing a good overall ability to distinguish between churn and non-churn customers.

The model performs strongly for **non-churn customers (Class 0)**, with a high recall of 0.90 and a solid F1-score of 0.86, meaning most non-churn cases are correctly identified. However, performance on **churn customers (Class 1)** is weaker, with a recall of 0.48 and an F1-score of 0.55, indicating that many churn cases are not detected.

Overall, while the neural network converged efficiently in 49 iterations and shows stable performance, it does not outperform Logistic Regression and remains limited in accurately identifying churn customers.

The dataset is highly imbalanced, with significantly fewer churned customers than non-churned ones. This causes models to bias toward predicting the majority class, leading to poor churn detection. Oversampling the minority class in the training set exposes the model to more churn examples, improving its ability to identify churned customers. Oversampling is a technique used to balance imbalanced datasets by increasing the number of minority class samples so the model can learn both classes equally.

Oversampling :

```
=== Neural Network (MLP, Oversampled Train Only) ===
Accuracy: 0.7494677075940384
ROC-AUC: 0.7970110310263763
Confusion Matrix:
 [[806 229]
 [124 250]]
Classification Report:
              precision    recall  f1-score   support

           0       0.87      0.78      0.82      1035
           1       0.52      0.67      0.59       374

    accuracy                           0.75      1409
   macro avg       0.69      0.72      0.70      1409
weighted avg       0.78      0.75      0.76      1409


=== Performance Metrics (Percentage) ===
Accuracy      : 74.95%
Precision     : 52.19%
Recall        : 66.84%
F1-score      : 58.62%
ROC-AUC       : 79.70%
```

This output shows that after oversampling, the neural network became **better at detecting churn customers**, even though overall accuracy decreased slightly. The model achieved an accuracy of about **75%**, meaning three out of four predictions were correct. More importantly, the **recall for churn customers increased to around 67%**, which indicates that the model successfully identified most customers who actually churned. However, precision for churn is lower (about **52%**), showing that some non-churn customers were incorrectly predicted as churn. The ROC-AUC score of **79.7%** suggests the model has a reasonably good ability to distinguish between churn and non-churn customers overall. This trade-off is expected when handling imbalanced data, and in a churn prediction context, improving recall is often more valuable than maximizing accuracy

```
results_df = pd.concat([
    metric_table("Logistic Regression", y_test, y_pred_lr, y_prob_lr),
    metric_table("Decision Tree", y_test, y_pred_dt, y_prob_dt),
    metric_table("MLP", y_test, y_pred_mlp, y_prob_mlp),
    metric_table("MLP (Oversampled)", y_test, y_pred_bal, y_prob_bal)
])

print(results_df.round(2))
```

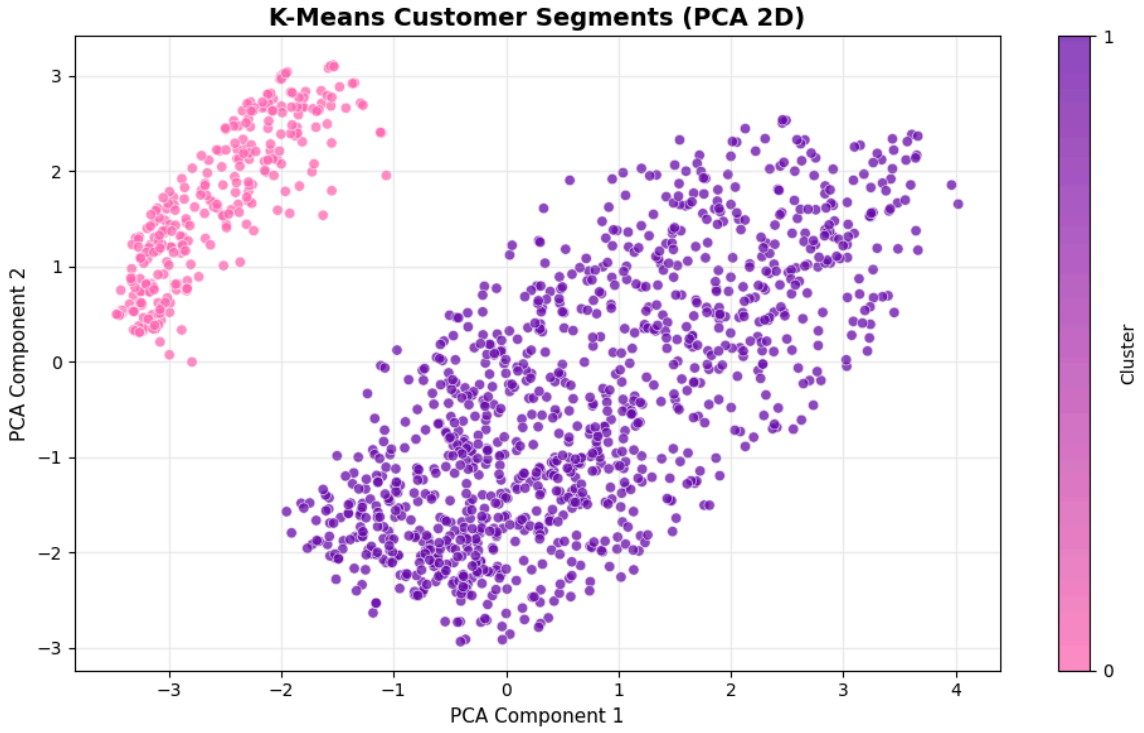
```
                Model  Accuracy (%)  Precision (%)  Recall (%)  F1-score (%)  \
0  Logistic Regression         80.55          65.72       55.88         60.40
0        Decision Tree         79.13          63.42       50.53         56.25
0                  MLP         78.92          63.51       48.40         54.93
0   MLP (Oversampled)         74.95          52.19       66.84         58.62

   ROC-AUC (%)
0        84.19
0        83.51
0        83.86
0        79.70
```

Unsupervised Learning:

K-Means:

K-Means clustering was applied as an unsupervised learning approach to identify natural groupings among customers without using churn labels. The algorithm was trained on the preprocessed training data only. PCA was then used to project the high-dimensional feature space into two dimensions for visualization purposes. After clustering, churn labels were used solely for post-hoc interpretation to examine how churn behavior differed across clusters. One cluster showed a significantly higher churn rate than the other, indicating that K-Means successfully identified meaningful customer segments, although it was less effective than supervised models for direct churn prediction.




```
Cluster
0    0.072488
1    0.318326
Name: Churn, dtype: float64
Churn Percentage per Cluster:
Cluster
0     7.25
1    31.83
Name: Churn, dtype: float64
```

Cluster 0 consists of customers with a very low churn rate, where only about 7.25% of customers left the service. This indicates that customers in this group are generally stable and loyal, with a low risk of churn. In contrast, Cluster 1 shows a much higher churn rate of approximately 31.83%, meaning nearly one-third of the customers in this group discontinued the service. This suggests that Cluster 1 represents a high-risk segment of churn-prone customers who may require targeted retention strategies.

Unsupervised learning was less effective than supervised models because K-Means does not use churn labels during training. As a result, clusters were formed based on overall similarity rather than churn-specific patterns, making supervised classifiers more suitable for direct churn prediction

6. Model Comparison and Evaluation

Comparison between the models:

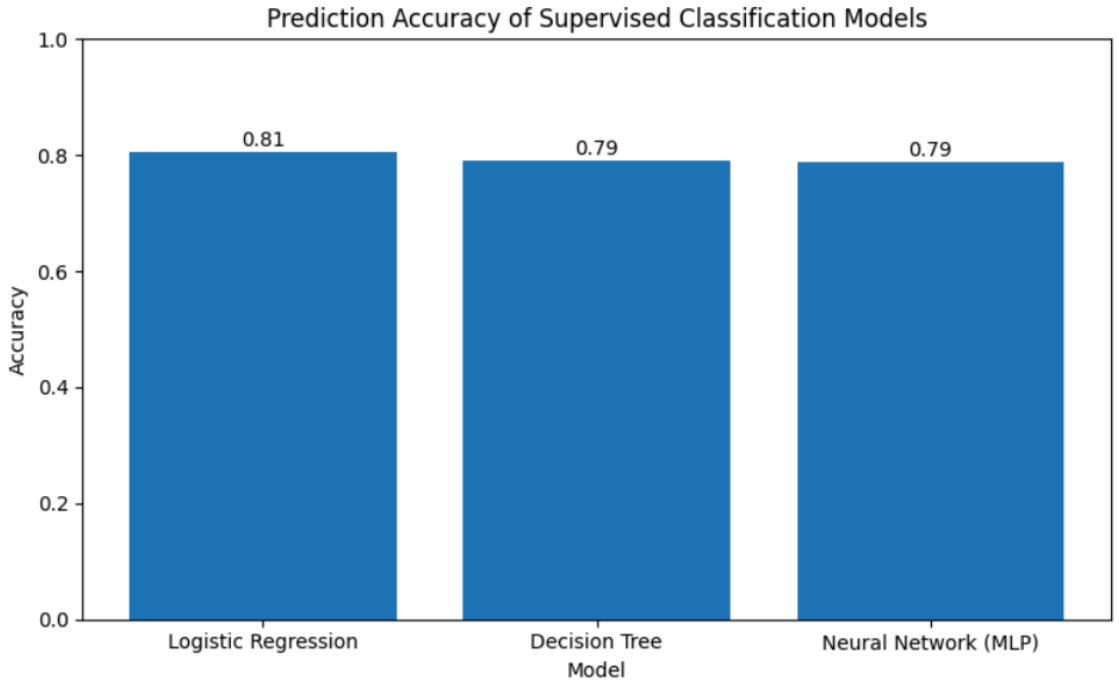
	Model	Accuracy	Precision (Churn=1)	Recall (Churn=1)	F1 (Churn=1)	ROC-AUC
0	Logistic Regression	0.805536	0.657233	0.558824	0.604046	0.841861
1	Decision Tree	0.791341	0.634228	0.505348	0.562500	0.835093
2	Neural Network (MLP Tuned)	0.789212	0.635088	0.483957	0.549317	0.838629

From the results, **Logistic Regression performs the best overall**. It achieves the highest accuracy (about **80.6%**) and the strongest ROC-AUC (**0.84**), indicating better overall discrimination between churn and non-churn customers. It also has the highest recall and F1-score for churn customers, meaning it detects more actual churn cases while maintaining a reasonable balance between precision and recall.

The **Decision Tree** performs slightly worse than Logistic Regression across all metrics, particularly in recall and F1-score, suggesting it misses more churn customers. The **Neural Network (MLP Tuned)** shows similar accuracy to the Decision Tree but has the **lowest recall for churn**, meaning it is the weakest at identifying customers who actually churn.

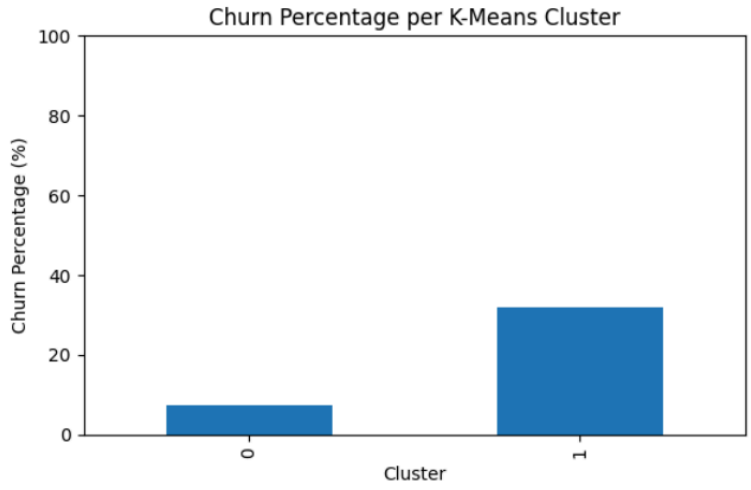
Overall, Logistic Regression is the most reliable supervised model in this comparison, especially when churn detection performance is prioritized over raw accuracy.

Bar Chart of the Supervised Classification Models:



This bar chart compares the **prediction accuracy** of the three supervised classification models. **Logistic Regression** achieves the highest accuracy at around **81%**, while both the **Decision Tree** and **Neural Network (MLP)** achieve slightly lower but similar accuracies of about **79%**. The differences between the models are relatively small, indicating that all three perform comparably in terms of overall correctness. However, Logistic Regression has a slight edge, suggesting better generalization on this dataset. It is important to note that because the churn dataset is imbalanced, accuracy alone does not fully reflect model performance, and metrics such as recall and F1-score are more informative for evaluating how well the models identify churn customers.

Churn Percentage:



This chart shows the **churn percentage within each K-Means cluster**. **Cluster 0** has a very low churn rate of roughly **7%**, indicating a group of mostly stable and loyal customers. In contrast, **Cluster 1** has a much higher churn rate of about **32%**, meaning nearly one-third of customers in this cluster left the service. This clear difference suggests that K-Means successfully identified two distinct customer segments with different churn behaviors. However, because clustering does not use churn labels during training, it is better suited for **customer segmentation and insight generation** rather than direct churn prediction, which explains why supervised models performed better overall.

Precision, recall comparison of each model:

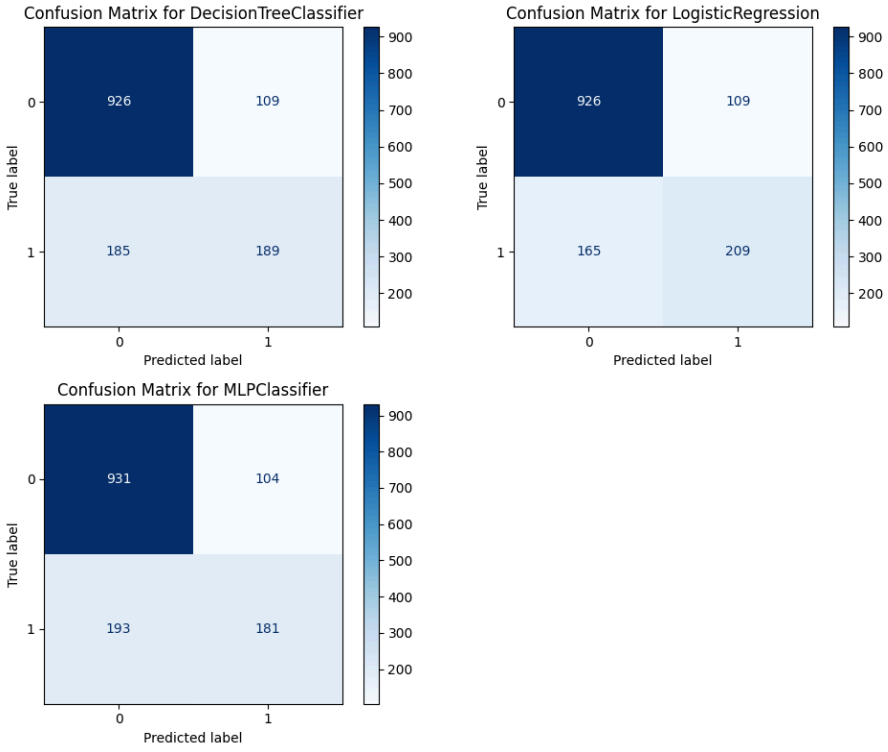
```
LogisticRegression:
 Precision       : 0.6572 (65.72%)
 Recall          : 0.5588 (55.88%)
 Confusion Matrix:
[[926 109]
 [165 209]]
----------------------------------------
DecisionTreeClassifier:
 Precision       : 0.6342 (63.42%)
 Recall          : 0.5053 (50.53%)
 Confusion Matrix:
[[926 109]
 [185 189]]
----------------------------------------
MLPClassifier:
 Precision       : 0.6351 (63.51%)
 Recall          : 0.4840 (48.40%)
 Confusion Matrix:
[[931 104]
 [193 181]]
----------------------------------------
```

The results show clear differences in how well each model identifies churn customers. **Logistic Regression** performs the best overall, with the highest precision (65.72%) and recall (55.88%), meaning it identifies more actual churn customers while keeping false alarms relatively lower. The **Decision Tree** shows slightly lower precision (63.42%) and recall (50.53%), indicating it misses more churn cases than Logistic Regression. The **Neural Network (MLP)** has similar

precision (63.51%) but the lowest recall (48.40%), meaning it fails to detect more than half of the customers who actually churn. Overall, Logistic Regression provides the most balanced performance for churn prediction among the three models.

Confusion Matrix:

Confusion matrices were generated only for the supervised classification models, namely Logistic Regression, Decision Tree, and Neural Network (MLP). These models produce explicit churn predictions, making confusion matrix analysis appropriate for evaluating true positives, false positives, true negatives, and false negatives. K-Means clustering was excluded from this evaluation because it is an unsupervised learning technique and does not generate class-based predictions required for confusion matrix computation.



The confusion matrices show how each model predicts churn (1) and non-churn (0) customers.

Logistic Regression

Logistic Regression correctly classified 926 non-churn customers and 209 churn customers. However, it incorrectly predicted 109 non-churn customers as churn and missed 165 actual churn

customers. Overall, this model shows a good balance between correctly identifying churn and non-churn customers, which explains why it achieved the highest accuracy and ROC-AUC score among the models.

Decision Tree

The Decision Tree correctly identified 926 non-churn customers and 189 churn customers, but it failed to detect 185 churn customers, which is higher than Logistic Regression. This indicates that the Decision Tree misses more churn cases, leading to a lower recall for churn customers, even though it performs well on non-churn predictions.

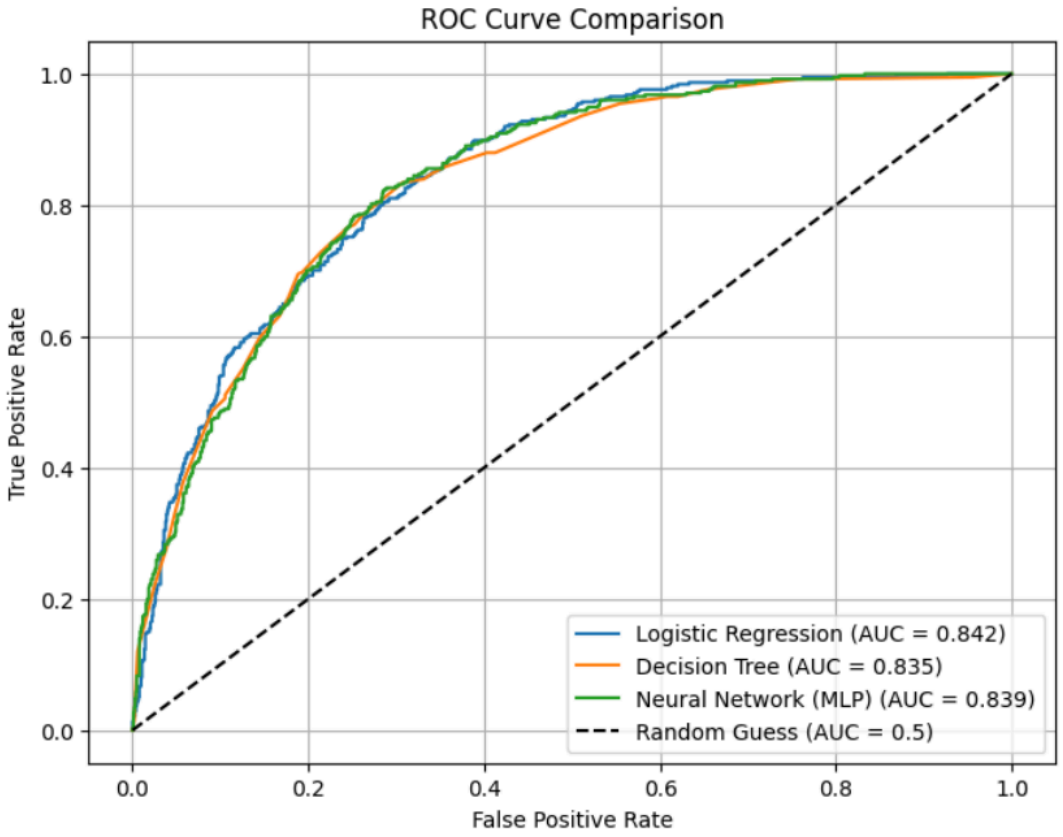
Neural Network (MLP)

The Neural Network correctly predicted 931 non-churn customers and 181 churn customers. While it performs slightly better than the Decision Tree in identifying non-churn customers, it misses the highest number of churn customers (193), resulting in the lowest churn recall among the three models. Despite tuning and early stopping, the Neural Network does not outperform Logistic Regression on this dataset.

Overall Comparison

From the confusion matrices, Logistic Regression performs the best overall, as it correctly identifies more churn customers while maintaining strong non-churn accuracy. The Decision Tree and Neural Network show similar behavior but miss more churn cases. This comparison highlights that simpler models like Logistic Regression can be more effective than complex models when working with structured, tabular datasets such as customer churn data.ped improving.

ROC Curve Comparison:



This ROC curve comparison shows that all three supervised models perform **significantly better than random guessing**, as their curves lie well above the diagonal baseline. **Logistic Regression** achieves the highest AUC (≈ 0.84), indicating the strongest overall ability to distinguish between churn and non-churn customers. The **Neural Network (MLP)** follows closely with an AUC of about **0.84**, while the **Decision Tree** performs slightly lower with an AUC of around **0.83**. The close overlap of the curves suggests that the models have similar discriminatory power, but Logistic Regression maintains a small and consistent advantage, making it the most reliable model for churn prediction in this comparison.

7. Conclusion

In this project, a complete churn prediction workflow was implemented, starting from exploratory data analysis (EDA) and data preprocessing to supervised and unsupervised learning, followed by thorough model evaluation. EDA revealed class imbalance and meaningful relationships between customer features and churn, justifying the use of stratified splitting and multiple evaluation metrics beyond accuracy. After proper preprocessing—including encoding, scaling, and handling missing values—three supervised models were trained: Logistic Regression, Decision Tree, and a Neural Network (MLP). Among these, Logistic Regression consistently performed the best,

achieving the highest accuracy and ROC-AUC, as well as the most balanced precision and recall for churn customers. The Decision Tree and MLP showed comparable but slightly weaker performance, particularly in identifying churn cases. Oversampling was applied to the neural network to address class imbalance, which improved recall for churn customers at the cost of lower overall accuracy, highlighting the expected trade-off in imbalanced classification problems. Unsupervised learning using K-Means clustering successfully identified distinct customer segments with significantly different churn rates, providing useful business insights, although it was less effective than supervised methods for direct churn prediction. Overall, the results demonstrate that supervised learning—especially Logistic Regression—is more suitable for churn prediction, while unsupervised learning adds value through customer segmentation and interpretability.