Matthew Ding
Professor Koehler
Data Bootcamp
5/12/25

# Predicting Customer Churn in Telecom

# Introduction

Customer retention is a major challenge for any business, especially in the telecom industry, where the competition is fierce and there is a high cost of customer acquisition compared to retaining existing customers. Thus, one of the most important factors that may pose a threat to profitability in this sector is customer churn, which is when customers stop using their products and services. Being able to accurately predict which customers and what characteristics they have make them likely to churn helps companies determine strategies for retaining customers. With proper data collection and strategy, companies can improve their profitability and keep a stable customer base.

In this project, we will address the problem of customer churn and predict it using a dataset originating from Kaggle. The dataset has a variety of features, including demographic features, account details, service usage, and billing information for more than 7000 customers. Our goal is to make a machine learning model that can accurately predict customer churn using the above data.

To do this, we will first give a brief overview of the dataset and its components, loaded using pandas dataframes, then start with exploratory data analysis to see patterns and relationships within the dataset that may indicate causes of churn for the company. Next, we preprocess the data using Scikit Learn and its libraries, with Standard Scaler and One Hot Encoder to make the data encoded and ready for model training. After that, we have three different machine learning models with Logistic Regression, K Nearest Neighbors, and a neural network made with PyTorch, trained and evaluated to see the most effective method for predicting churn.

A quick summary of our findings is that the simple Logistic Regression was at the top with high of 82% accuracy and churn recall of 52%. Closely, the PyTorch neural network model achieved a best accuracy of 79% and a churn recall of 54%. Key predictors of churn include the contract type, tenure, and payment method. Customers on month-to-month contracts with shorter tenures and electronic check payments are at higher risk of churning. This aligns with concepts of low customer loyalty, shown through low tenure, and low cost of switching, shown through month-to-month payments and electronic check payments. These insights can provide valuable guidance for business strategies for customer retention to make sure the business focuses on these high-risk segments.

# Data Description

The dataset used for this project is the Telco Customer Churn dataset from Kaggle, which contains 7043 rows of customer information and 18 base features with a binary churn column. An instance of the data is saved in GitHub and loaded. Key features are shown below:

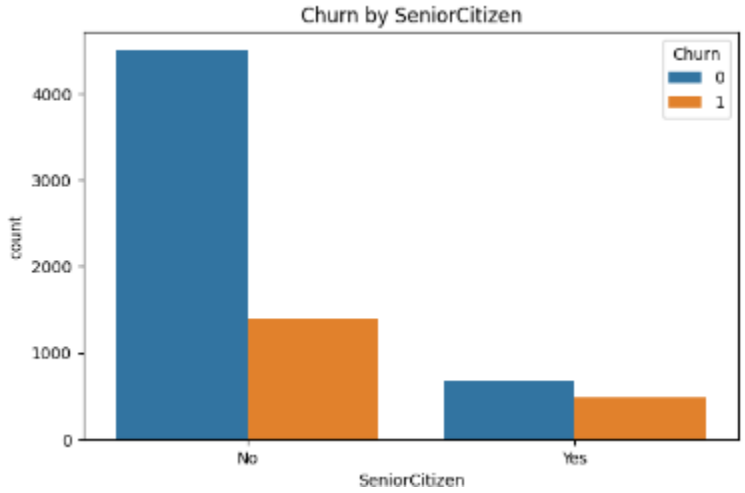
- Demographics: Gender, SeniorCitizen status, Partner, Dependents
- Account Information: Tenure (months with the company), Contract type (Month-to-month, One year, Two year), PaperlessBilling, PaymentMethod
- Services: PhoneService, MultipleLines, InternetService, OnlineSecurity, OnlineBackup, DeviceProtection, TechSupport, StreamingTV, StreamingMovies
- Financial: MonthlyCharges, TotalCharges

- Target Variable: Churn (Yes/No)

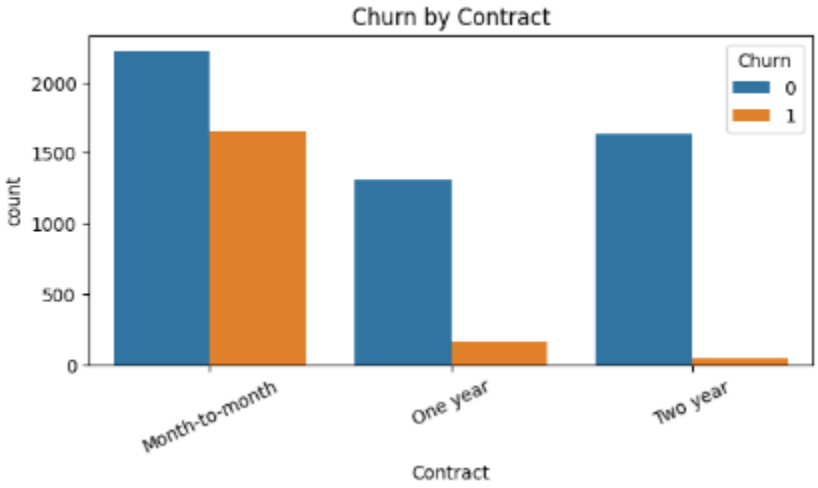
## Exploratory Data Analysis

Before building the predictive models, we first performed exploratory data analysis to understand the data and relationships between certain variables and features. This can help understand patterns and trends for feature engineering and model selection.

To start off, we evaluated the distribution of all the key features to see if any stand out from normal. Here are the ones that was particularly interesting:

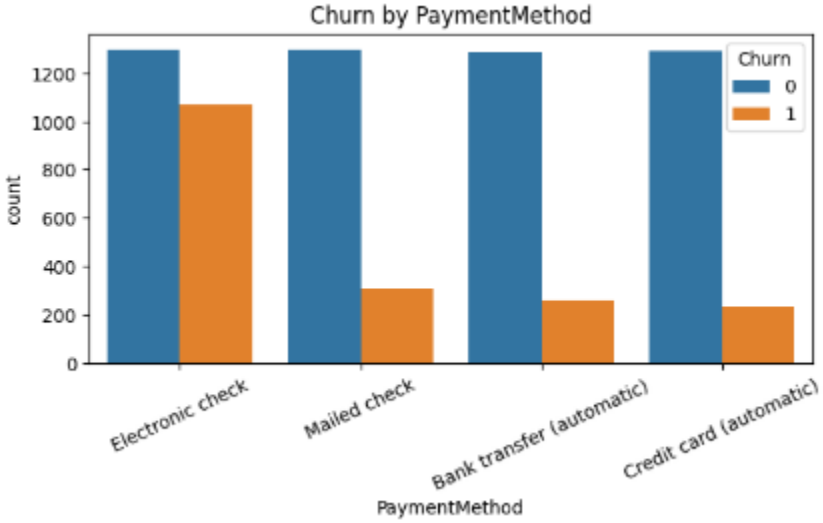



Unlike normal churn patterns, for senior citizens, the difference between churn and non-churn is much lower. This suggests that senior citizens are more likely to churn compared to non-senior citizens. Even though the total number of senior citizens is lower, their churn rate is considerably higher, which may indicate that age can be an important factor that influences customer churn.



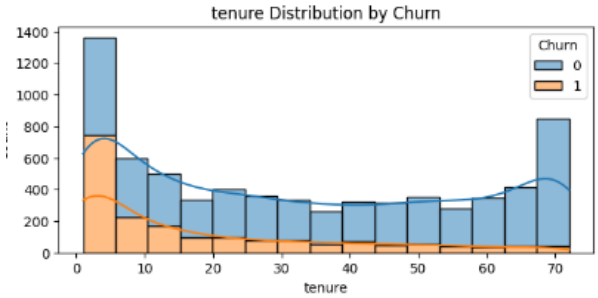

We can also see, for this bar chart, that the churn rate is considerably higher for month-to-month compared to the considerably lower churn rate for one year and two years, which shows contract type as a strong predictor of churn. Month-to-month customers are more prone to leaving, likely due to flexibility due to a lack of long-term commitment, thus lowering

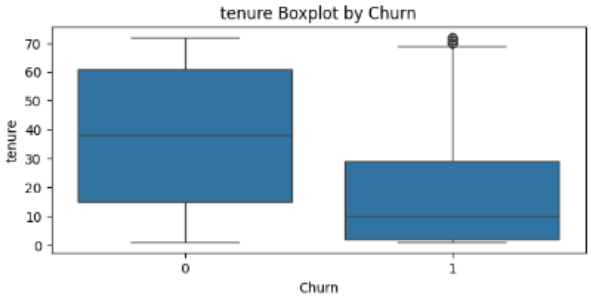
their cost of switching. On the other hand, customers on longer contracts are very stable and much less likely to churn, most likely due to incentives or penalties with contract terms. Retention efforts for customers with month-to-month contracts should be focused to reduce churn effectively.



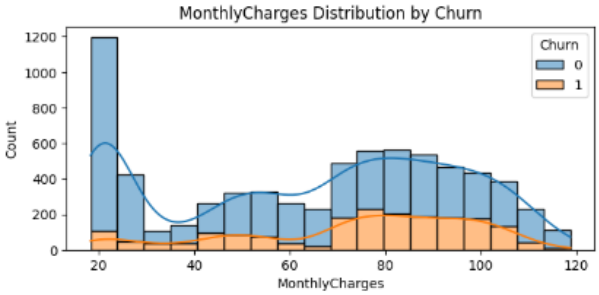

Similarly, the bar chart shows close churn and non-churn rates for customers paying with electronic checks compared to other payment methods. The convenience and reliability of automatic payments versus electronic checks may cause customers using electronic checks to become less engaged and more likely to churn. This insight suggests that encouraging customers to switch to automatic payment could be a better churn reduction strategy.

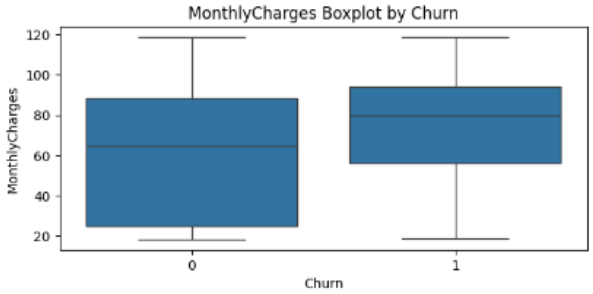
Moving on to more detailed EDA, we have side-by-side distributions and box plots of numerical features:



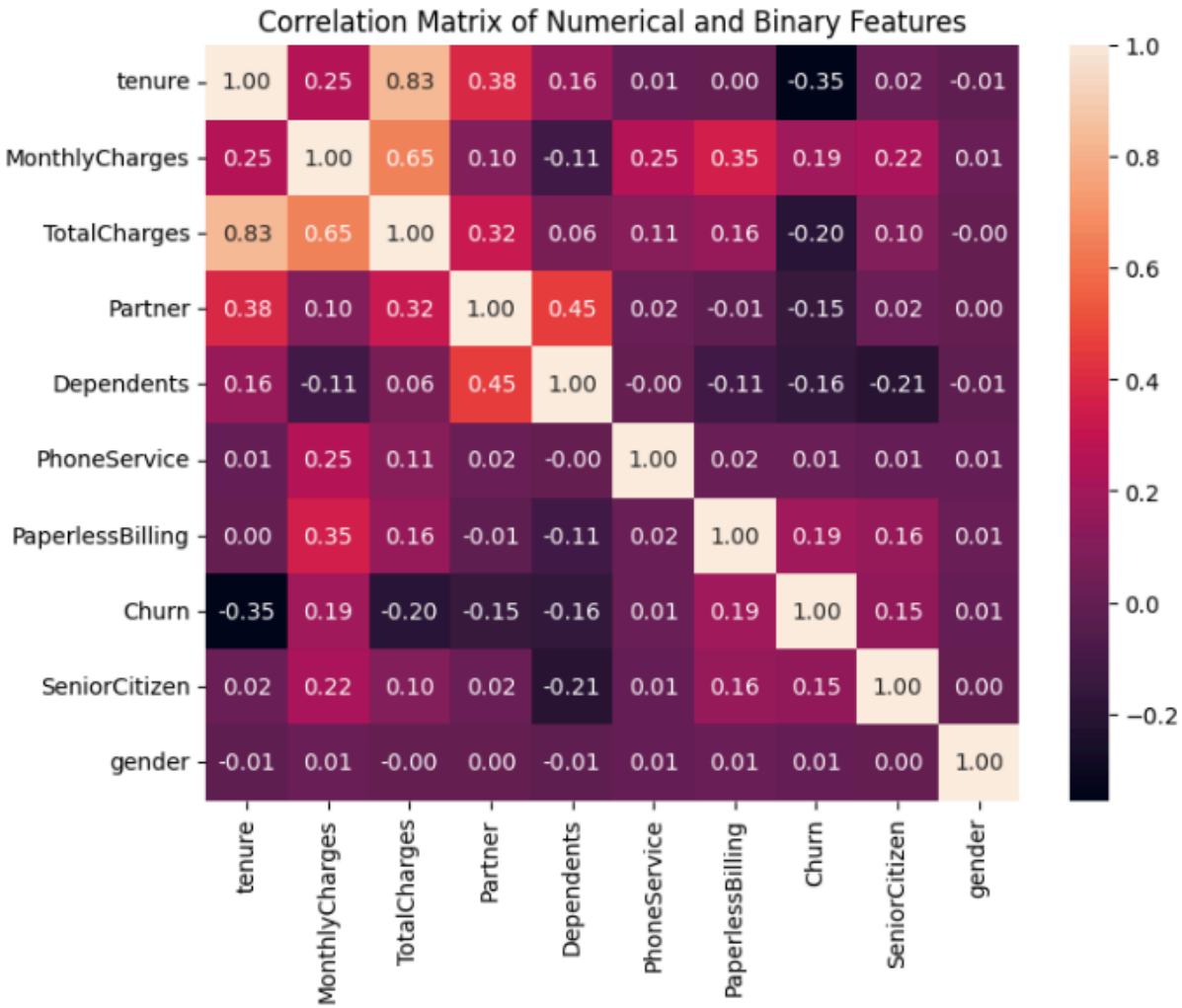

This curve shows a right-skewed distribution for churn with a relatively uniform distribution for non-churn, describing how customers who churn usually tend to have much shorter tenures, with most leaving within the first few months. On the other hand, customers who stay have longer tenures, along with a higher median and wider range. This indicates that the likelihood of churn decreases as tenure increases, which means early retention efforts are highly important.



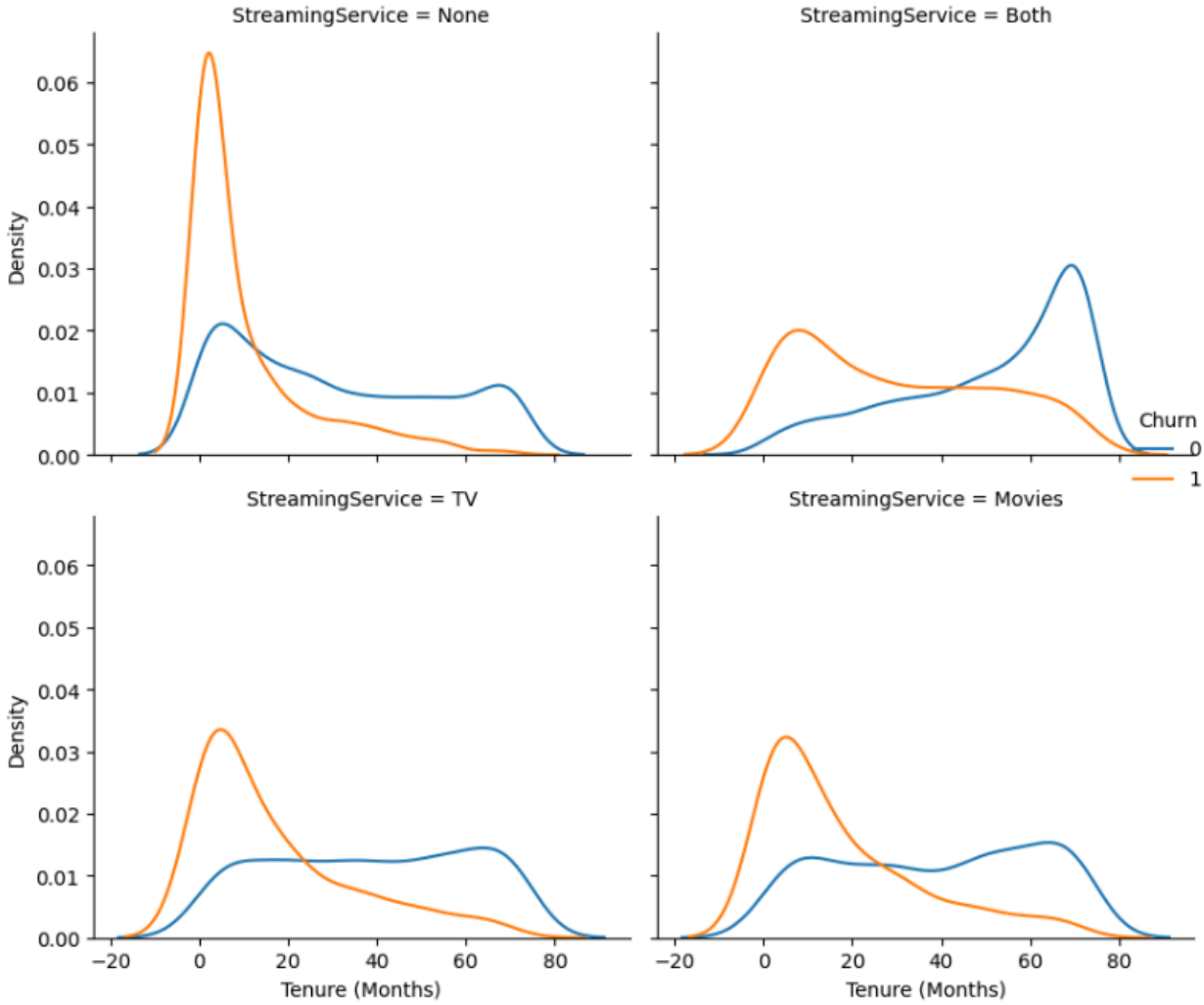




From these two visualizations, we can see that customers who churn generally have higher monthly charges, shown through the distribution and boxplot. The median monthly charge for churn is much higher than for non-churn, and churn is more common for customers with high bills. This means that higher monthly charges have a direct impact on churn rate, especially after $70 per month, which may be the price of a certain plan that is offered to users. Efforts can be made to reduce churn by lowering the cost of plans for certain customers to reduce churn if it is profitable in the long run.

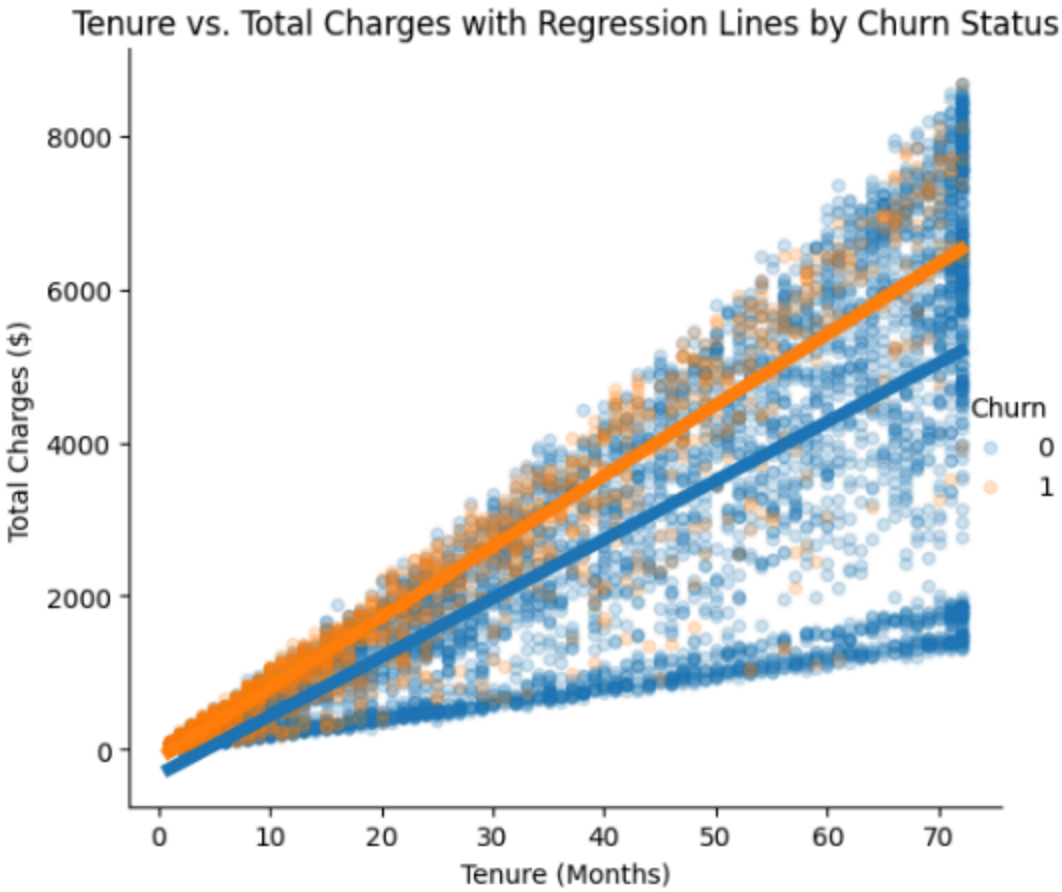
Next, we have a correlation matrix for numerical and binary features with churn:



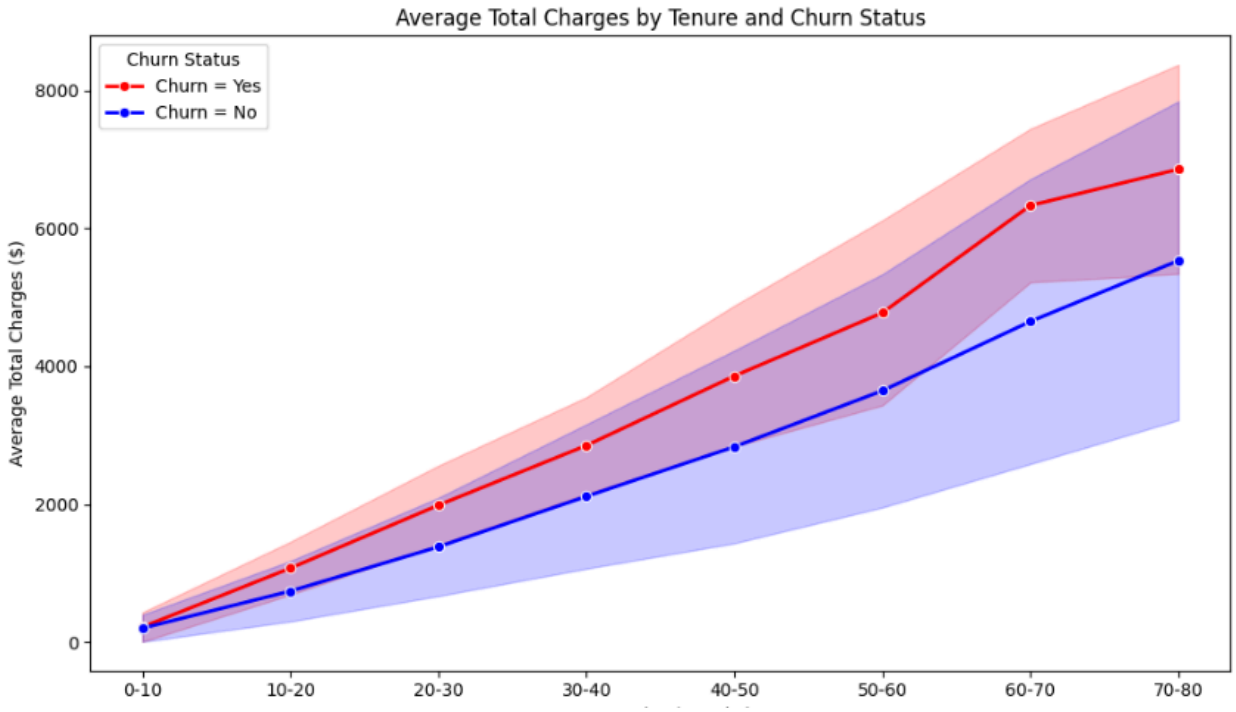
This correlation matrix shows that tenure has a moderate negative correlation on chur, which shows that customers with shorter tenure are more likely to leave. Monthly charges and paperless billing have a weak positive correlation with churn, which shows that a monthly charge increase and the implementation of paperless billing slightly increase churn. Features like partners and dependents have a low negative correlation with churn, and senior citizen has a low positive correlation with churn, showing they have less direct impact compared to others. Overall, tenure, monthly charges, and paperless billing seem to be the most relevant numerical predictors of churn for this dataset.




In this above facet grid, we can see that customers with no streaming services or only one of TV or movies tend to churn early in their tenure, shown by the high density of churn at lower tenure values in these groups. On the other hand, customers with both streaming services are less likely to churn early and show higher tenure overall. This suggests that customers who subscribe to both are more engaged and loyal customers, while those with fewer streaming services are more at risk of churn. Bundles can also be implemented for improving retention.




The above scatterplot with regression lines shows the relationship between tenure and total charges, using regression lines for churned and non-churned customers. Churned customers in orange show a steeper upwards trend than non-churned customers in blue, showing that they accumulated higher total charges over similar tenure periods. This suggests that customers who spend more relative to tenure time are likely churning due lack of long-term satisfaction and loyalty due to higher charges. A strategy to assist with this is to have special benefits to lower costs for loyal customers.




This line plot shows the average total charges by tenure group for churned and non-churned customers, with the filled regions showing the variability. Across all the tenure ranges, there is a higher average total charge than for those who stayed, similar to our previous graph. The shaded variability shows that there is a significant difference between the two churn statuses, and that pricing over time has a major impact on churn. The wide range for non-churned also demonstrates that high spending, long tenure customers may still be at risk of churn, which shows the importance of not overlooking loyal, high-value customers.

## Data Preprocessing

In this project, several steps were taken to prepare the Telco Customer Churn dataset for modeling. First, TotalCharges had missing values within its data, primarily for new users with no tenure (less than one month). These rows were removed since completely new customers would not contribute meaningfully to churn prediction.

Next, the dataset contained multiple categorical features like gender, contract, payment method, and many more. To fix this for machine learning, used One Hot Encoder to encode features to ensure compatibility with our models. For numerical columns like tenure, monthly charges, and total charges, they were standardized to have a zero mean and unit variance using Standard Scaler. This is important and helpful for models like KNN and neural network training stability. There was also the removal of any previous columns that were added and aggregated based on previous existing information to reduce double-counting.

## Models and Methods

For this customer churn prediction, we used three different machine learning models: Logistic Regression, K Nearest Neighbors, and a Neural Network. These models were selected to provide both simplicity and the ability to capture more complex relationships within the data. Logistic Regression is the interpretable baseline for the code, and KNN is made to be sensitive the the data structure provided, and finally Neural Network has the capability to have advanced modeling for patterns for more complex relationships. Here is the rationale for the selected models:

1. Logistic Regression: Is good for binary classification problems like churn, and it is very interpretable and efficient. It directly uses feature importance and is good as a starting benchmark for other models.
2. KNN: It can model complex decision boundaries without heavily relying on data distribution and making assumptions about it. It can help with churn prediction to see patterns, especially as many of our data relationships seem to be nonlinear.
3. Neural Network: These models are very powerful and good for churn prediction, especially as our dataset is relatively large and has many nonlinear features. This flexibility makes it good to find more complex interactions between variables that the simpler models above can't collect.

This combination of models allows for a variety of predictions to examine performance and overall relationships between data models, and to determine which one is potentially better for long-term prediction.

Hyperparameter Tuning was also made to assist with some models. The dataset was split into training and test data, and testing is reserved for final accuracy tests. Here is what we did for them:

1. For KNN, the number of neighbors and distance metrics were optimized using grid search, with performance evaluated on the validation set.
2. For the Neural Network, hyperparameters such as the number of layers, units per layer, and batch size were tuned using trial and error using validation loss and accuracy. Trial and error with epochs was also done to prevent overfitting.

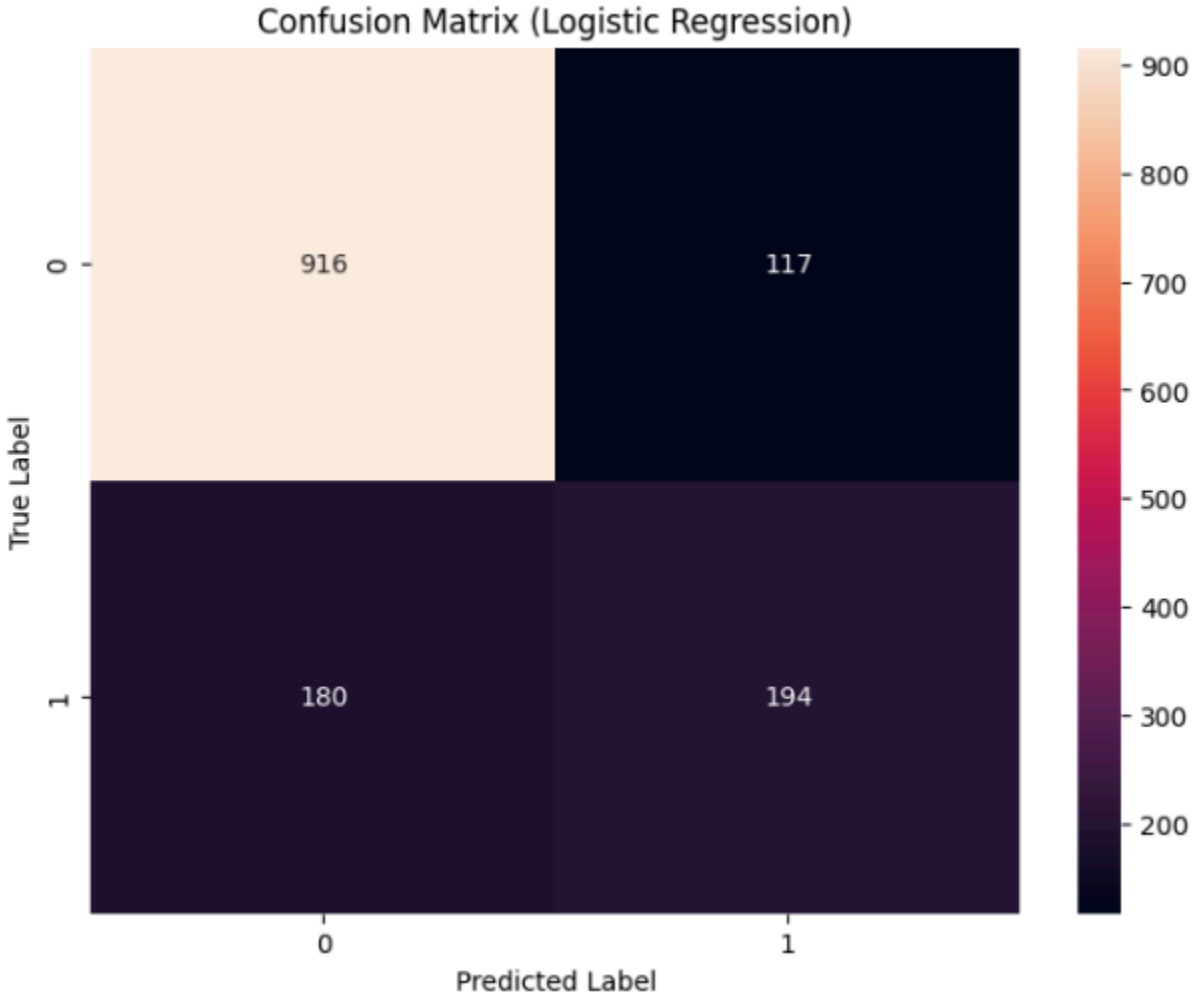
To implement the models, logistic regression and KNN were made using sklearn along with the same preprocessor and data for standardization. Both were fitted into a pipeline for cleaner implementation and prediction. For Neural Network, it was done using PyTorch for custom architecture and computation, with the same preprocessor and data as before.

## Modeling Results and Interpretations

All three models were evaluated on the customer churn prediction task, and shown similar accuracy and recall throughout (~75% and 50% respectively). Model performance was assessed using standard classification metrics: accuracy, precision, recall, F1-score, and ROC-AUC. These performance measures show each model's strengths and weaknesses, and

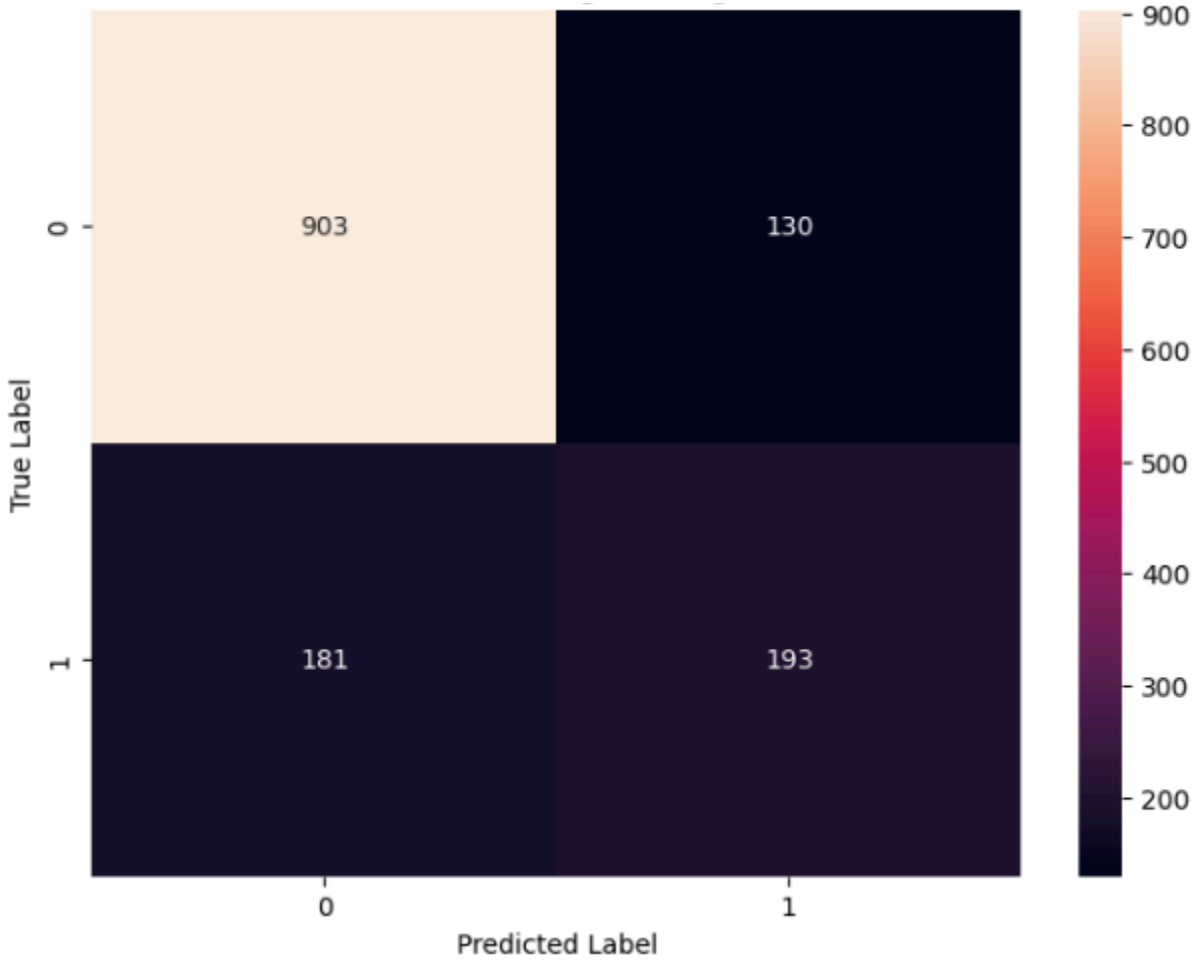
their compatibility with the data. Of the models used, Logistic Regression and Neural Network had some of the highest accuracy scores.

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.84 | 0.89 | 0.86 | 1033 |
| 1 | 0.62 | 0.52 | 0.57 | 374 |
| accuracy | | | 0.79 | 1407 |
| macro avg | 0.73 | 0.70 | 0.71 | 1407 |
| weighted avg | 0.78 | 0.79 | 0.78 | 1407 |




Logistic Regression achieved an accuracy of 0.79 on the test set, with a precision of 0.62, a recall of 0.52, and an F1-score of 0.57 for the churned class. The model performed well in predicting non-churned customers, but was less effective at capturing all churned customers, as reflected in the lower recall for the positive class. However, this might be due to the imbalanced dataset. Adding on to that, Logistic regression's interpretability is very helpful for actionable business insights, making it one of the best models with usability and simplicity.

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.82 | 0.84 | 0.83 | 1033 |
| 1 | 0.53 | 0.50 | 0.52 | 374 |
| accuracy | | | 0.75 | 1407 |
| macro avg | 0.68 | 0.67 | 0.67 | 1407 |
| weighted avg | 0.75 | 0.75 | 0.75 | 1407 |

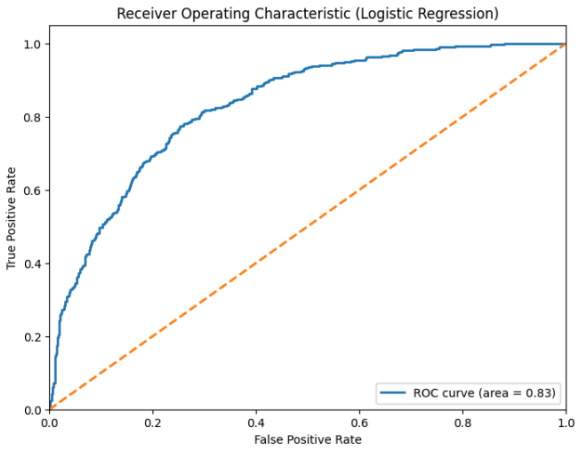


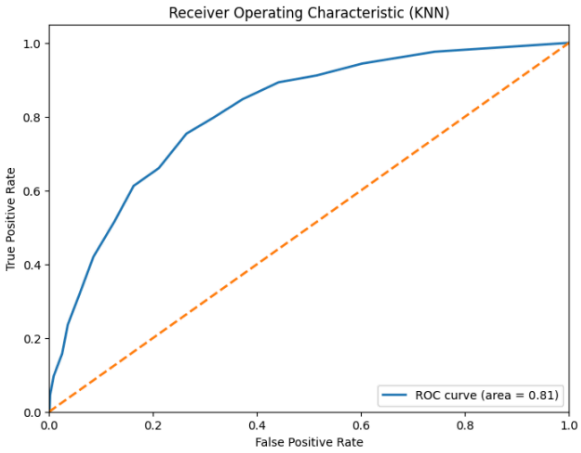

KNN got an accuracy of 0.75. For the churned class, precision was 0.53, recall 0.50, and F1-score 0.52. While KNN performed relatively well for the non-churned class, it also fell i performance for churn due to the same imbalance. KNN's lower recall and F1-score for churned customers indicate it is less effective than Logistic Regression for this task.


| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.82 | 0.84 | 0.83 | 1033 |
| 1 | 0.53 | 0.50 | 0.52 | 374 |
| accuracy | | | 0.75 | 1407 |
| macro avg | 0.68 | 0.67 | 0.67 | 1407 |
| weighted avg | 0.75 | 0.75 | 0.75 | 1407 |

The neural network also achieved an accuracy of 0.79. For the churned customers, precision was 0.61, recall 0.54, and F1-score 0.58. It performed similarly to logistic regression but showed a slight improvement in recall and F1-score for identifying churned customers, showing that there is a better balance between detecting churn and lowering false positives.

Here are some supporting graphs to identify the accuracy and development of our models:




Both Logistic Regression and KNN had strong ROC curves, showing a good overall performance for churned and non-churned customers. The AUC for ROC for Logistic Regression and KNN were 0.83 and 0.81, respectively. This means that Logistic Regression has a slightly better ability to rank customers for probably of churning, but both models are much above just random chance (AUC of 0.5). These results show that the models are effective at separating the two classes, with Logistic Regression having a very small advantage in predictive ability.


Here is the training overview for Neural Networks, showing training loss curve, training accuracy curve, and test accuracy curve for 300 epochs. The training loss curve shows a slowing exponential decrease reaching toward zero as the epoch increases, showing effective model learning over the epochs. Both the training and test accuracy curves quickly rise and then plateau, showing how the model converged and has a stable performance without too much overfitting. The plateau in accuracies shows that further training would likely not have significant improvements. Overall, these curves show that the neural network trained efficiently and is close to reaching its optimal performance within 300 epochs.

## Conclusion and Next Steps

In this project, all three machine learning models performed relatively well in predicting customer churn. Both Logistic Regression and the Neural Network achieved strong, balanced performance, with accuracy scores of 0.79 and F1 scores for churned around 0.57 to 0.58. KNN was slightly worse in identifying churned customers. Across all models, key predictors of churn included contract type, tenure, and payment method. ROC curve analysis showed that both Logistic Regression and KNN were good in effectively distinguishing between churned and non-churned customers, with Logistic Regression showing a slightly higher AUC.

For next steps, we can tackle the class imbalance with more resampling techniques using SMOTE, which was attempted but not providing good results, and other methods to improve the recall rate for churn. Also, we can implement more feature engineering in order to use and see more features or interactions that may show additional patterns like customer engagement metrics or service usage trends. More models like RFC and more advanced neural networks can be used to boost prediction performances as well.

For presentation, more model interpretation tools and visualization tools can be used to display the model and its effectiveness to others, as well as automatically identifying and generating pain points or strategies based on patterns shown. Finding more or merging this data with other datasets to compare across industries or geographies can also be done to make the resulting predictions more valuable.

Thank you for reviewing my project. Looking forward to any feedback.