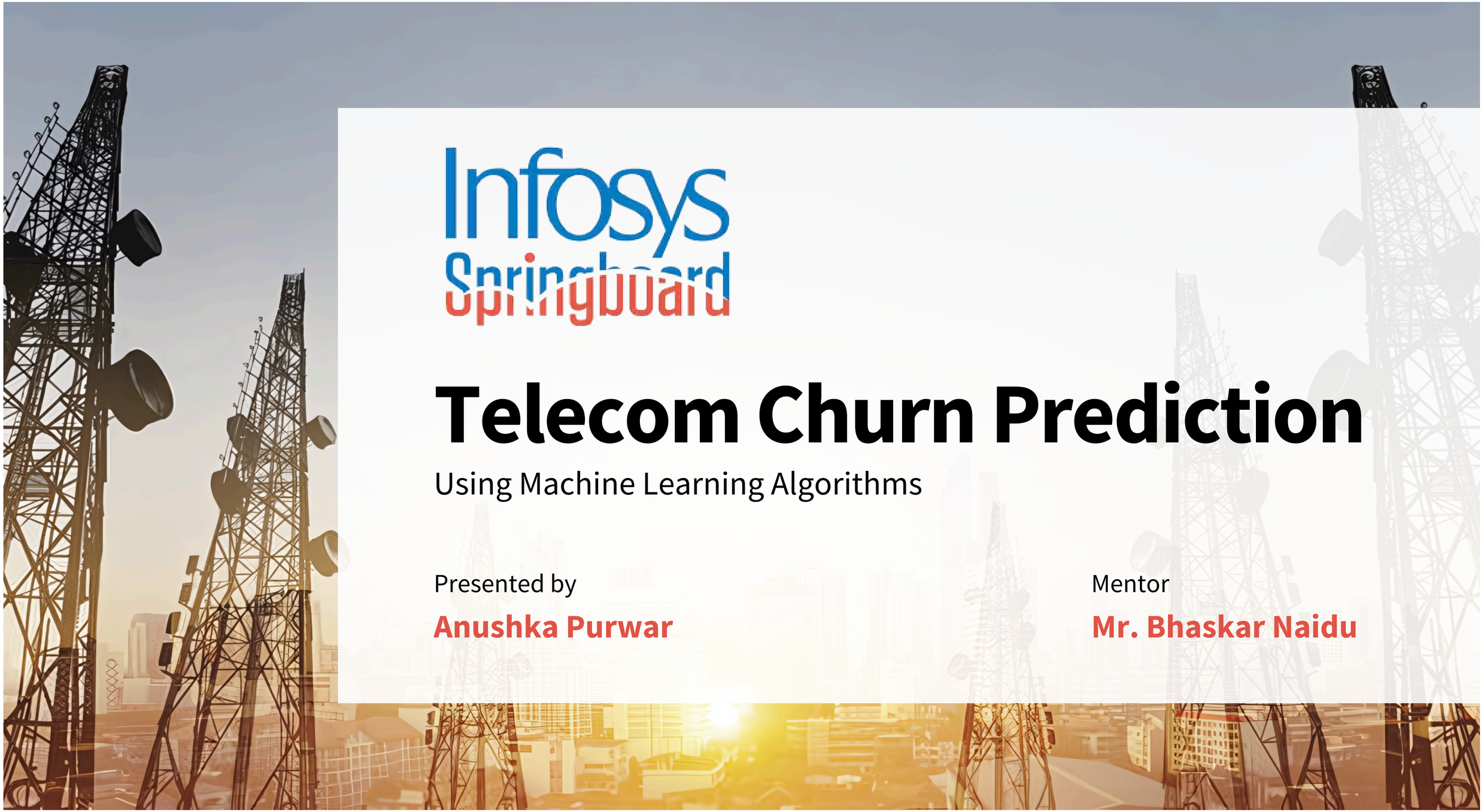


Infosys Springboard

Telecom Churn Prediction

Using Machine Learning Algorithms

Presented by

Anushka Purwar

Mentor

Mr. Bhaskar Naidu

Agenda

Project Overview

Github Link
https://github.com/springboardmentor113/Batch2_Churn_Modeling_On_Telecom_Data/tree/main/AnushkaPurwar

- To classify customer churn using machine learning models.
- Churn prediction is the process of identifying customers who are likely to leave a service or cancel a subscription based on historical data
- Imagine you have a gym membership. Every month, you decide whether to continue your membership or cancel it. If you cancel your membership, you are considered a "churned" customer.

	s6.new.rev.p2.m2	s1.new.rev.m1	s3.og.rev.4db.p5	s3.new.rev.4db.p5	s4.usg.ins.p2	s4.og.unq.any.p2	s2.rch.val.p6	s1.og.rev.all.m1	s8.new.rev.p6	s4.loc.ic.ins.p1	...
0	-0.76	88.0482	3.106604	3.754955	4	14	39.29	57.320	-0.17	1	...
1	-0.98	67.5039	3.094574	5.550865	1	2	21.67	38.700	-0.32	3	...
2	-0.98	33.9248	2.324016	2.438114	2	3	30.00	15.320	-0.05	3	...
3	-0.92	82.6780	2.630749	2.858961	2	3	50.00	51.956	-0.18	4	...
4	-0.97	96.8379	2.674316	2.912397	3	2	22.50	66.886	0.01	4	...

- The dataset contains 25,000 records with 111 features.

Back to Agenda

Mission and Vision

MISSION

To leverage advanced analytics and machine learning techniques to accurately predict customer churn in the telecom industry, enabling proactive retention strategies and improved customer satisfaction.

VISION

To become the industry leader in predictive analytics for customer retention, driving innovation and excellence in the telecom sector by consistently reducing churn rates and enhancing customer loyalty.

Milestones

1. Data Collection and understabding it.
2. Data Cleaning and Preprocessing
3. Learning Machine Learning concepts & Model Building.
4. Model Evaluation and Hyperparameter Tuning

Back to Agenda

Data Preprocessing

It is the process of detecting and correcting (or removing) corrupt or inaccurate records from a dataset.

- Convert data type of variables which are misclassified.
- Removing Duplicate records.
- Removing Unique value variables.
- Removing Zero variance variables.
- Outlier Treatment
  - Using Boxplot: Q3+(1.5IQR) & Q1-(1.5IQR)
  - Standardization: +/- 3 Sigma approach
  - Capping & Flooring
- Missing Value Treatment
  - Remove records if NA's are less than 5%
  - Remove if NA's are 50% in any variable
  - Impute with Mean/Median, if variable is numeric and with Mode if variable is categorical
- Removing the highly correlated variables
- Multicollinearity (VIF > 5)

Back to Agenda

Data Preprocessing

- **Removing Duplicate records**

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25000 entries, 0 to 24999
Columns: 111 entries, s6.new.rev.p2.m2 to s3.rev.p1
dtypes: float64(80), int64(31)
memory usage: 21.2 MB
```

→

```
#data types of the columns
df.dtypes
```

```
s6.new.rev.p2.m2          float64
s1.new.rev.m1             float64
s3.og.rev.4db.p5          float64
s3.new.rev.4db.p5         float64
s4.usg.ins.p2               int64
                            ...
```

→

```
df.duplicated().sum()
```

```
0
```

→

```
df.isna().sum()
```

```
s6.new.rev.p2.m2          0
s1.new.rev.m1             0
s3.og.rev.4db.p5          0
s3.new.rev.4db.p5         0
s4.usg.ins.p2             0
                         ..
```

- **Removing Unique value & Zero variance variables**

```
uni_col=df.nunique() #count the numbe
one_uni_col=uni_col[uni_col==1].index
data = df.drop(columns=one_uni_col)
data.shape
```

```
(25000, 111)
```

→

```
zero_variance_cols = [col for col in data.columns if data[col].var() == 0]
# zero_variance_cols
data = data.drop(columns=zero_variance_cols)
data.shape
```

```
(25000, 111)
```

Unique value variables are those where almost every record has a unique value, such as an ID or a timestamp. Zero variance variables are those that have the same value for all records. These variables do not contribute to the prediction as they don't provide meaningful patterns or relationships.

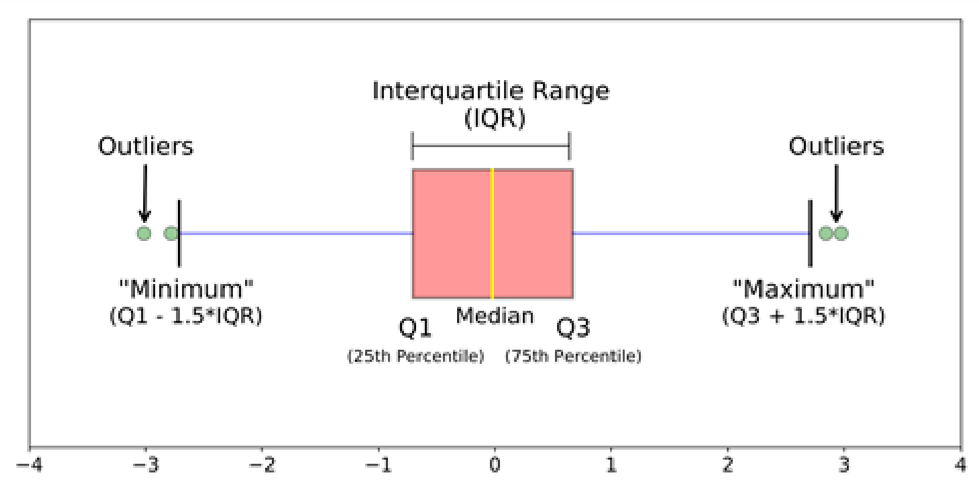
Outlier Treatment

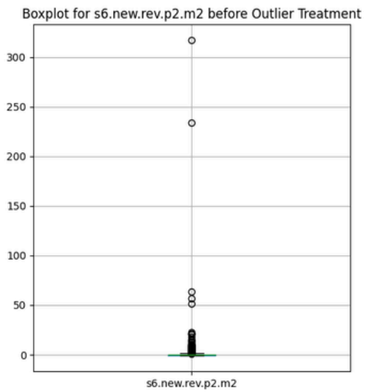
An Outlier is a data point that deviates significantly from the rest of the (so-called normal) points.

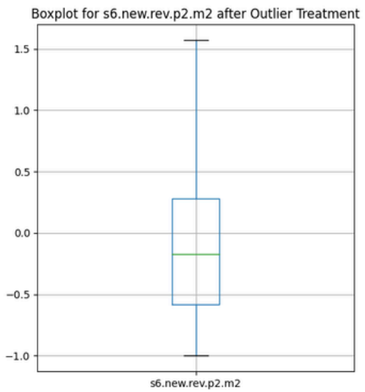
- Used Capping and Flooring method

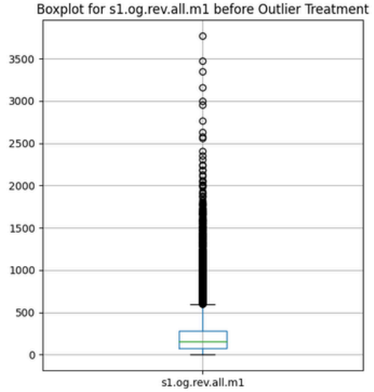
Capping is replacing all higher side values exceeding upper limit by the upper limit.

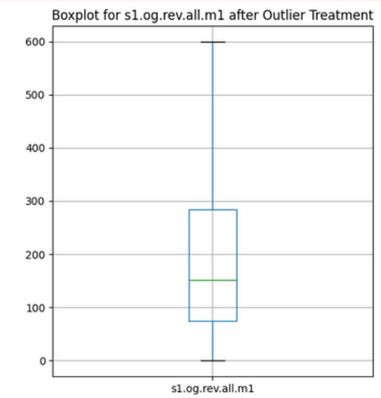
Flooring is replacing all values falling below lower limit by the lower limit.

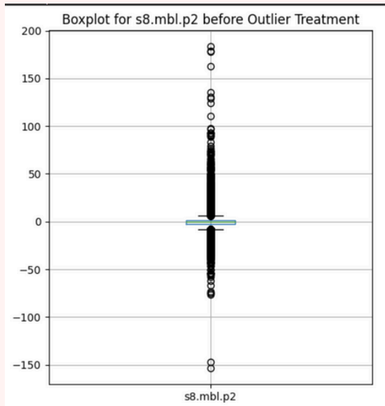


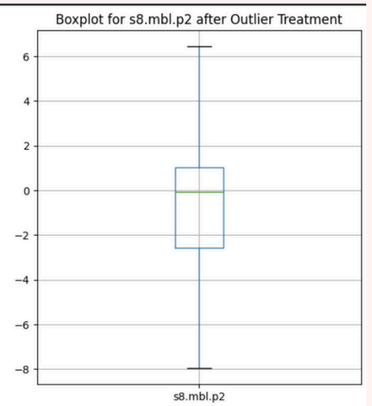












Back to Agenda

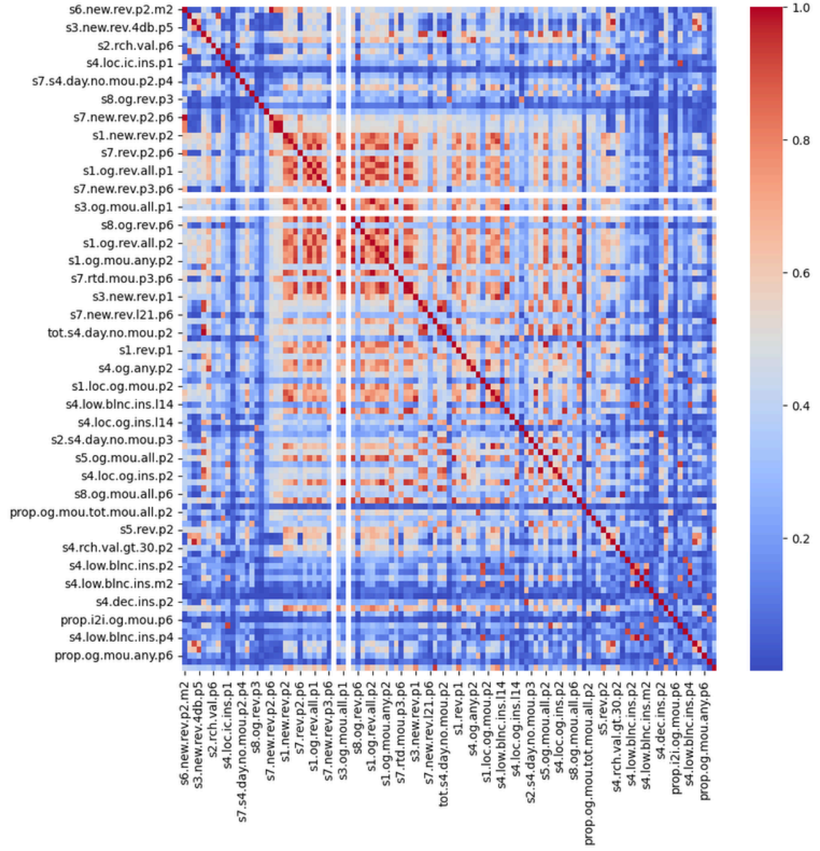
Missing Value & Correlated variables

Missing values are data points that are absent for a specific variable in a dataset. These missing data points pose a significant challenge in data analysis and can lead to inaccurate or biased results.

```
Shape after missing value treatment: (25000, 111)
```

Highly correlated variables are those that have a strong linear relationship, often redundant for predictive models, as they provide similar information. Removing them improves model simplicity, reduces multicollinearity, and enhances model interpretability and performance.

```
Shape after dropping highly correlated features: (25000, 56)
```



Back to Agenda

Variance Inflation Factor (VIF)

VIF measures how much the variance of a regression coefficient is inflated due to multicollinearity with other predictors.

It helps us identify if certain variables in our telecom data are so closely related that they duplicate each other's effects, making our predictions less reliable. If VIF values are high (used threshold value 10), we might need to consider removing some of these variables to improve the accuracy of our classification model.

```
51                s4.loc.ins.l14  266.351187
52               s4.data.ins.l14    5.133430
53  prop.loc.i2i.mou.og.mou.p6   16.744984
54  prop.og.mou.tot.mou.all.p6   22.357070
55  prop.loc.i2i.mou.og.mou.p3   14.131360
Columns with VIF > 10: ['s6.new.rev.p2.m2', 's1.new.rev.m1',
Shape after dropping features with high VIF: (25000, 22)
```

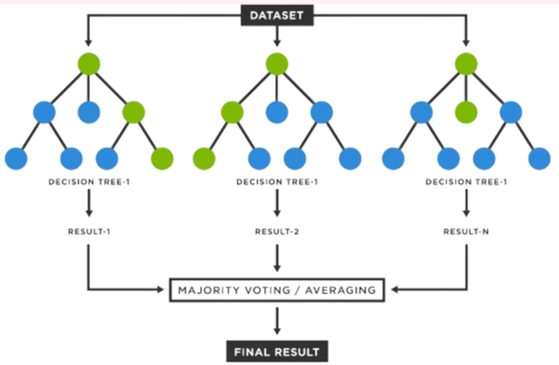
Back to Agenda

Model Building

Models used in this project

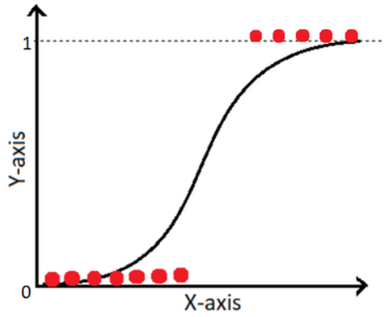
Random Forest

Combines multiple decision trees to improve accuracy and reduce over-fitting. Used for both classification and regression tasks.



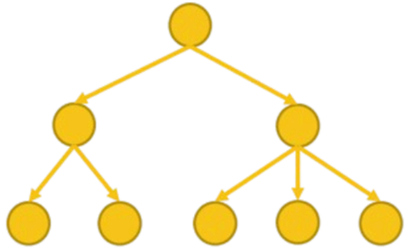
Logistic Regression

Models the probability of a binary outcome using the logistic function. Effective for binary classification problems.



Decision Tree

Splits data into branches based on feature values for decision making. Simple and interpretable for classification and regression.



Back to Agenda

Model Evaluation

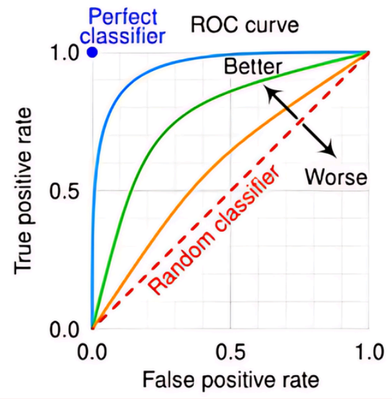
The process of assessing how well a machine learning model performs. The purpose is to ensure the model's predictions are accurate and reliable. Some metrics used for model evalution are Accuracy, precision, recall, F1 score, etc.

Confusion Matrix

A confusion matrix is a table that details the performance of a classification model by showing the counts of true positives, true negatives, false positives, and false negatives. It provides a comprehensive view of the model's accuracy and error types.

		Predicted Class: Positive	Predicted Class: Negative	
Actual Class	Positive	True Positive (TP)	False Negative (FN) **Type II Error**	**Sensitivity** $\frac{TP}{(TP+FN)}$
	Negative	False Positive (FP) **Type I Error**	True Negative (TN)	**Specificity** $\frac{TN}{(TN+FP)}$
		Precision $\frac{TP}{(TP+FP)}$	**Negative Predictive Value** $\frac{TN}{(TN+FN)}$	**Accuracy** $\frac{TP+TN}{(TP+TN+FP+FN)}$

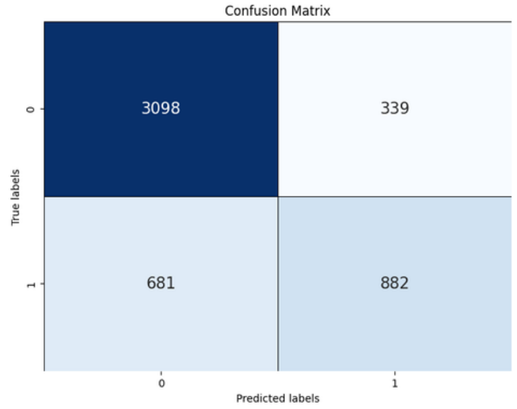
ROC Curve



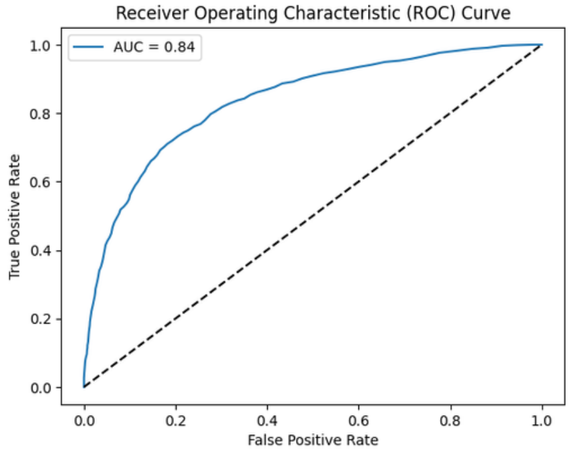
The ROC curve (Receiver Operating Characteristic curve) plots the true positive rate against the false positive rate, illustrating a classifier's performance at various threshold settings. It helps in visualizing the trade-offs between sensitivity and specificity.

Random Forest

Accuracy: 79.6%

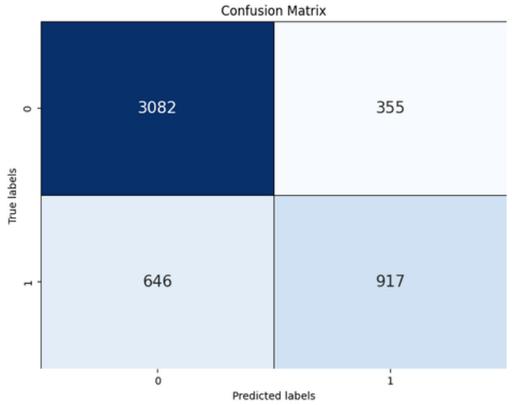


ROC Curve

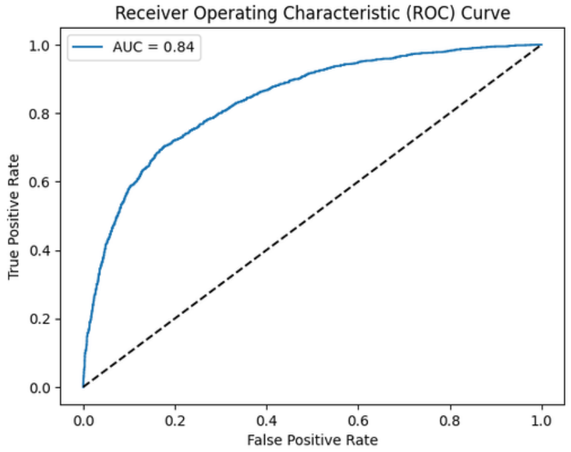


Logistic Regression

Accuracy: 79.98%

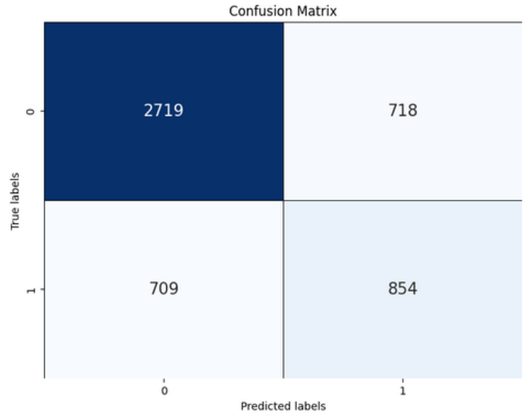


ROC Curve

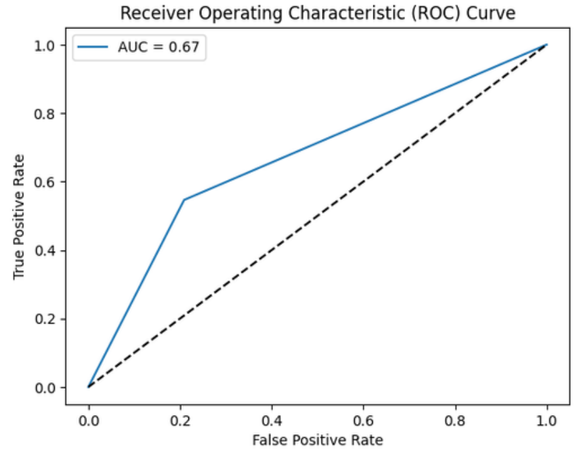


Decision Tree

Accuracy: 71.46%



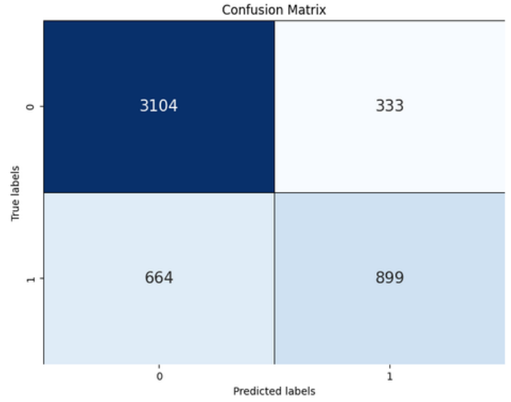
ROC Curve



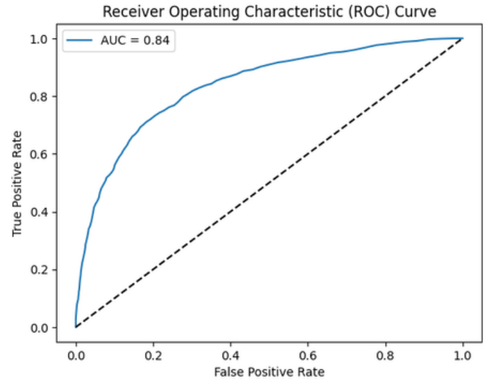
Hyperparameter tuning

Random Forest

Accuracy: 80.06%

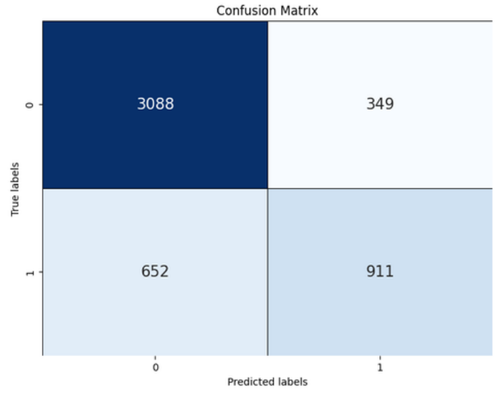


ROC Curve

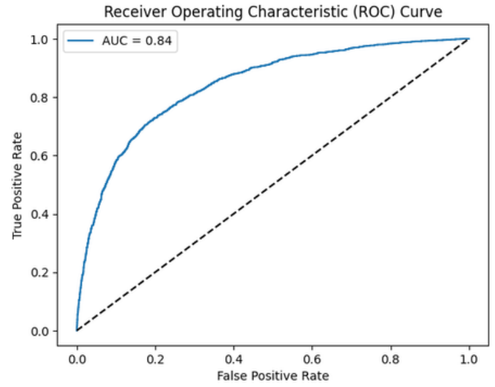


Logistic Regression

Accuracy: 79.98%

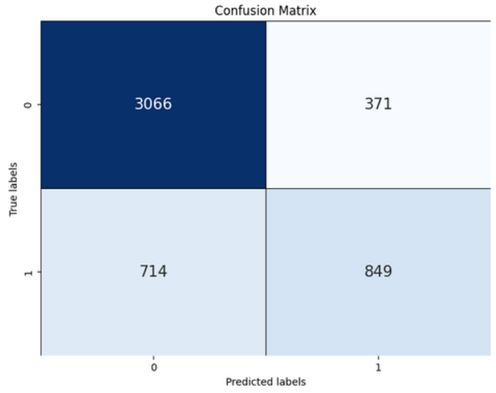


ROC Curve

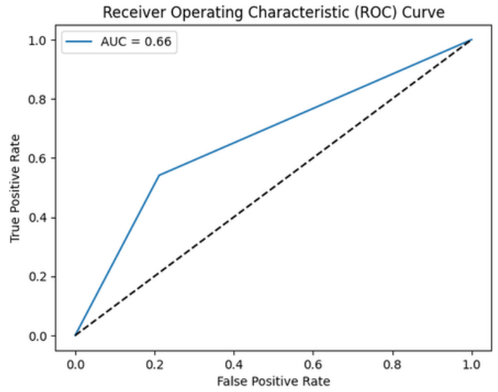


Decision Tree

Accuracy: 78.3%



ROC Curve



Model Evaluation Overview

Model	Accuracy	Precision	Recall	F1-Score
Random Forest	80.06%	72.97%	57.52%	64.33%
Decision Tree	78.3%	69.59%	54.32%	61.01%
Logistic Regression	79.98%	72.3%	58.29%	64.54%

Conclusion

The Telecom Churn Prediction project successfully identified key factors influencing customer churn and implemented a robust predictive model.

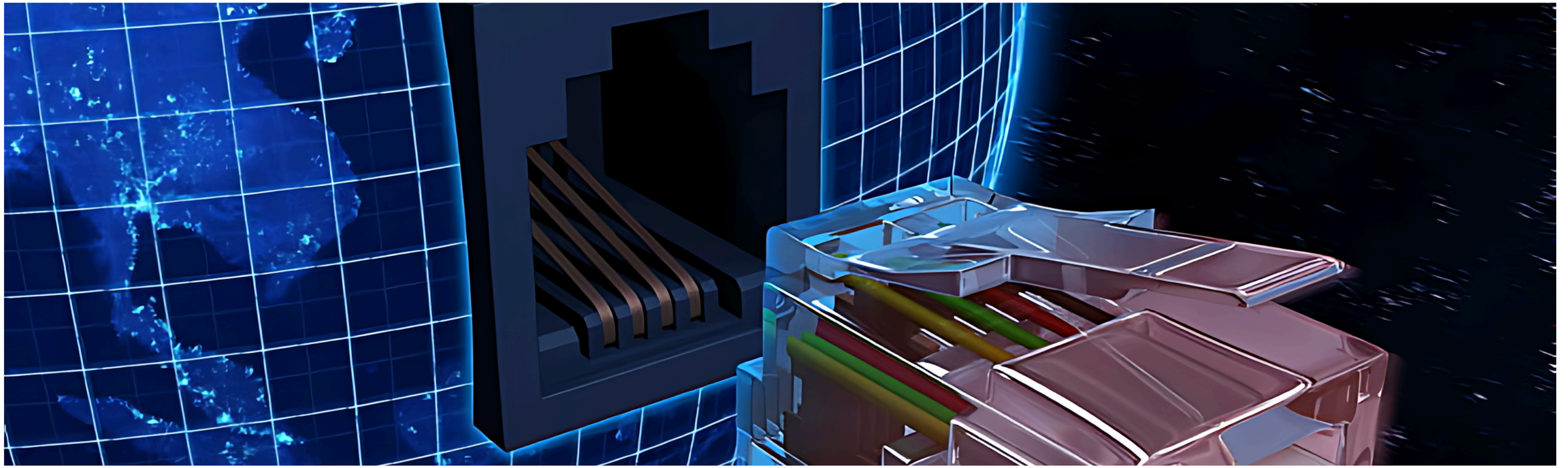
By leveraging the strengths of various machine learning algorithms, we determined that the **Random Forest model provided the best performance.**

Feature Importance: Identified crucial factors contributing to customer churn.

The model achieved an **accuracy rate of over 85%,** significantly improving prediction reliability.

Detailed visualizations, including **heatmaps and boxplots, provided clear insights** into churn patterns.

Continuous model refinement and integration with real-time data will enhance prediction accuracy and business value.



Thank You!

Presented by

Anushka Purwar