ST JOSEPH'S UNIVERSITY

BENGALURU 560027

MACHINE LEARNING PROJECT
CHURN PREDICTION

Submitted by:
SHREYAS DASAN
24PGDDA12

Submitted to:
AARAN LAWRENCE DLIMA
Department of Advanced Computing
St. Joseph's University
36, Lalbagh Road
Bangalore- 560027

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Overview and background:

In the age of data driven decision making, businesses are increasingly relying on predictive analytics to improve customer retention, enhance revenue, and sustain long term growth. One of the most pressing challenges faced by subscription based and service-oriented businesses is customer churn, the phenomenon where customers discontinue using a company's products or services. The ability to accurately predict customer churn is immensely valuable, as it enables businesses to proactively engage at risk customers, implement retention strategies, and reduce revenue loss. This project focuses on developing a machine learning solution for customer churn prediction, with the goal of assisting businesses in identifying potential churners before they actually leave.

Churn prediction involves building models that can analyse customer data and estimate the likelihood of a customer leaving the business in the near future. This is a classic binary classification problem, where the objective is to assign a label (churn or no churn) to each customer based on a variety of input features. These models can be trained using historical data where churn outcomes are already known. Once trained, they can predict churn risk for current customers and help stakeholders take pre-emptive actions to minimize losses.

From a business perspective, customer churn directly affects profitability and long-term success. It is widely acknowledged that acquiring a new customer is often far more expensive than retaining an existing one. Studies suggest that increasing customer retention rates by as little as 5% can increase profits by 25% to 95%. Therefore, reducing churn has a direct impact on a company's bottom line. Moreover, existing customers are more likely to try new products, spend more, and refer others, making them invaluable assets. Businesses that fail to understand and address churn risk eventually suffer from customer base erosion, increased marketing spend, and weakened competitive positioning.

Churn prediction is especially important in industries such as telecommunications, banking, insurance, SaaS (Software as a Service), e-commerce, and streaming services, where customers typically have multiple service options and switching costs are low. In these domains, maintaining strong customer relationships is crucial. Predictive models that can pinpoint dissatisfied or disengaged customers offer an opportunity to tailor retention campaigns, offer personalized deals, or improve customer support interactions before the customer defects.

There are several key factors to consider when working on churn prediction. First, it is important to understand that customer behaviour is influenced by a combination of demographic, transactional, and service-related variables. Patterns such as reduced engagement, service complaints, frequent contract changes, or long inactivity periods can signal potential churn. These patterns are not always obvious, and effective churn prediction requires capturing subtle relationships and interactions among various features. Therefore, feature engineering, the process of transforming raw data into meaningful inputs for machine learning models plays a vital role in model success.

Second, churn is often an imbalanced classification problem, where the number of churned customers is much smaller than the number of non-churned ones. This imbalance can lead standard machine learning models to become biased toward the majority class, thereby reducing their ability to identify true churners. Special techniques such as oversampling the minority class, using class weighted algorithms, or applying ensemble methods become essential to ensure the model remains sensitive to the rare but important churn events.

Another important consideration is the business context of false positives and false negatives. A false positive occurs when a model wrongly predicts a customer will churn when they won't, possibly leading to unnecessary retention offers. A false negative, on the other hand, means the model failed to detect a customer who is actually going to churn, representing a missed opportunity. The cost of each type of error varies by business, and models must be evaluated using appropriate metrics (such as F-1 score, recall, or ROC-AUC) that align with business priorities.

However, despite its importance and clear benefits, churn prediction comes with its own set of challenges. One of the biggest hurdles is the dynamic nature of customer behaviour. Customer preferences, market conditions, and competitor actions evolve rapidly, and churn prediction models must be updated frequently to remain accurate. A model trained on old data may no longer reflect current churn patterns. Another challenge is the availability and quality of data. Many businesses struggle with incomplete, inconsistent, or noisy data, which limits model performance. Data privacy is also a growing concern, especially when sensitive customer information is used in training models.

Interpretability is another significant challenge, particularly in industries where decisions must be explained to stakeholders or customers. While complex machine learning algorithms such as ensemble methods and neural networks can achieve high accuracy, they are often treated as "black boxes." Balancing predictive performance with explainability is crucial for trust and transparency, especially in regulated sectors.

Against this backdrop, the aim of this project is to build a churn prediction system using machine learning that is accurate, scalable, and interpretable. The process involves collecting and analysing historical customer data, preprocessing it to handle missing values, outliers, and categorical variables, and then training multiple supervised learning models. Techniques such as class balancing and hyperparameter tuning are incorporated to enhance model quality. Finally, models are evaluated using comprehensive metrics to identify the best performer for deployment or further refinement.

The overarching goal is not just to build a technically sound model, but to develop a solution that can be seamlessly integrated into real world business decision making. Such a solution should enable companies to act on the insights derived from the model, for instance, by targeting high risk customers with loyalty programs or by improving touchpoints in the customer journey that commonly lead to dissatisfaction.

In summary, customer churn prediction is a vital application of machine learning with significant implications for business strategy and customer management. It allows businesses to shift from reactive to proactive modes of engagement, maximizing lifetime customer value and reducing unnecessary churn related losses. This project takes a systematic and practical approach to solving the churn problem using data science, with an emphasis on model accuracy, business interpretability, and real-world applicability. As companies become more data driven, the role of churn prediction will continue to grow in importance and this project aims to be a step in that direction.

1.2 Definition:

In the context of customer churn prediction, several technical and business-related terms are frequently used. Understanding these terms is essential for grasping the methodology, interpretation of results, and implications of this project. Below are key definitions:

- **Customer Churn:** Customer churn refers to the phenomenon where a customer stops using a company's product or service. Churn may be voluntary (the customer chooses to leave) or involuntary (due to account closure, policy violation, etc.). In churn prediction, churn is typically represented as a binary variable (1 for churned, 0 for retained).
- **Churn Rate:** Churn rate is the percentage of customers who leave a company over a given time period. A high churn rate indicates potential problems in customer satisfaction, service quality, or product relevance. It is calculated as:

$$Churn\ Rate = \frac{Number\ of\ customers\ lost}{Total\ Customers\ at\ the\ start\ of\ the\ period} \times 100$$

- **Customer Retention:** Customer retention is the ability of a company to keep its existing customers over time. Retention strategies aim to minimize churn and maximize customer lifetime value. Churn prediction models are often used to improve retention.
- **Binary Classification:** Binary classification is a supervised machine learning task where the goal is to categorize data points into one of two groups. In churn prediction, the two classes are typically "Churn" and "No Churn."

1.3 Need for study:

The study of customer churn prediction is vital for several reasons. Below are key points that justify the need for this project:

1. **High Cost of Customer Acquisition**
   - Acquiring new customers is significantly more expensive than retaining existing ones.
   - Businesses can save on marketing and sales expenses by focusing on customer retention through churn prediction.
2. **Direct Impact on Revenue and Profit**
   - Losing loyal or high-value customers directly affects revenue.
   - Predicting and preventing churn helps maintain a stable and recurring income stream.
3. **Competitive Advantage**
   - In highly competitive markets like telecom, banking, and SaaS, churn prediction enables businesses to stay ahead by proactively addressing customer dissatisfaction.
4. **Improved Customer Retention**
   - Early identification of at-risk customers allows companies to implement targeted retention strategies such as discounts, personalized offers, or enhanced customer support.
5. **Data-Driven Decision Making**
   - Churn prediction transforms raw customer data into actionable insights.
   - Helps managers make informed business decisions backed by statistical evidence.
6. **Efficient Resource Allocation**
   - Retention efforts can be focused on high-risk customers, reducing unnecessary spending on those unlikely to churn.
   - Enables smarter use of marketing and service budgets.
7. **Identifying Root Causes of Churn**
   - Machine learning models can highlight key features influencing churn (e.g., service issues, billing complaints).
   - This insight helps improve products, policies, and service quality.

8. **Aligns with Customer Centric Strategies**
   - Businesses today prioritize customer experience.
   - Churn prediction aligns with this strategy by identifying pain points and improving customer satisfaction.
9. **Scalability of Prediction Models**
   - Once built, churn models can be applied to large datasets and integrated into real-time systems.
   - Enables continuous monitoring of customer behaviour.
10. **Enhances Long-Term Business Sustainability**
   - Reducing churn increases customer lifetime value (CLV).
   - Long-term customer relationships are crucial for sustainable business growth.

This study is not only a technical exercise but also a critical business initiative. By predicting churn effectively, businesses can reduce revenue loss, improve customer relationships, and stay competitive in fast-evolving markets.

1.4 Objectives:

- To understand customer churn behaviour and its impact in real-world business scenarios.
- To apply machine learning techniques to develop an accurate binary classification model for churn prediction.
- To address key data challenges such as class imbalance, preprocessing, and feature transformation.
- To evaluate and compare multiple classification algorithms for optimal performance.
- To build a deployable, scalable, and business-interpretable framework that supports proactive customer retention strategies.

2. DATASET

2.1 Information regarding dataset:

- The dataset used for this project was obtained from Kaggle, a well-known platform for data science and machine learning resources.
- It is titled **"Customer Churn Prediction: Analysis"** and is designed to help understand customer attrition in a telecommunications context.
- The dataset contains customer-related information including demographics, subscription details, billing behaviour, and churn status.
- There are a total of 1,000 rows (customers) and 10 columns (features) in the dataset which includes:
  - **CustomerID:** Unique identifier for each customer.
  - **Age:** Age of the customer, reflecting their demographic profile.
  - **Gender:** Gender of the customer (Male or Female).
  - **Tenure:** Duration (in months) the customer has been with the service provider.
  - **MonthlyCharges:** The monthly fee charged to the customer.
  - **ContractType:** Type of contract the customer is on (Month-to-Month, One-Year, Two-Year).
  - **InternetService:** Type of internet service subscribed to (DSL, Fiber Optic, None).
  - **TechSupport:** Whether the customer has tech support (Yes or No).
  - **TotalCharges:** Total amount charged to the customer (calculated as MonthlyCharges * Tenure).
  - **Churn:** Binary target variable indicating whether the customer has churned (Yes or No).
- Context from Kaggle: The dataset provides insight into customer decisions in a telecom setting and allows for exploratory data analysis and supervised machine learning for classification. It helps in

understanding why customers leave a service and provides a foundation for creating predictive models to prevent future churn.

- Why This Dataset?
  - It suitable for binary classification problems.
  - It supports both business insights and technical analysis, making it ideal for data analytics and machine learning project work.

2.2 Summary of dataset:

This section provides a statistical and structural summary of the dataset used for churn prediction, based on exploratory analysis performed using Python functions in the notebook.

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   CustomerID       1000 non-null   int64
 1   Age              1000 non-null   int64
 2   Gender           1000 non-null   object
 3   Tenure           1000 non-null   int64
 4   MonthlyCharges   1000 non-null   float64
 5   ContractType     1000 non-null   object
 6   InternetService  703 non-null    object
 7   TotalCharges     1000 non-null   float64
 8   TechSupport      1000 non-null   object
 9   Churn            1000 non-null   object
dtypes: float64(2), int64(3), object(5)
memory usage: 78.3+ KB
```

- The data types include:
  - int64 for 3 numerical columns: CustomerID, Age, Tenure
  - float64 for 2 continuous variables: MonthlyCharges, TotalCharges
  - object for 5 categorical variables: Gender, ContractType, InternetService, TechSupport, Churn

The only column with missing values is InternetService, with 297 missing entries.

```
df.describe()
```

	CustomerID	Age	Tenure	MonthlyCharges	TotalCharges
count	1000.000000	1000.000000	1000.00000	1000.000000	1000.000000
mean	500.500000	44.674000	18.97300	74.391290	1404.364060
std	288.819436	9.797741	18.89257	25.712083	1571.755048
min	1.000000	12.000000	0.00000	30.000000	0.000000
25%	250.750000	38.000000	5.00000	52.357500	345.217500
50%	500.500000	45.000000	13.00000	74.060000	872.870000
75%	750.250000	51.000000	26.00000	96.102500	1900.175000
max	1000.000000	83.000000	122.00000	119.960000	12416.250000

- We observe that the Tenure and TotalCharges exhibit high variability, with many new customers (tenure = 0), indicating a need to study their relationship with churn.
- Distribution Observations:
  - The minimum Tenure is 0, indicating new customers. These may be more prone to churn.
  - TotalCharges has values starting from 0, corresponding to customers with either no or very short tenure.
  - MonthlyCharges range from ₹30 to over ₹100, which may reflect different subscription plans.
  - The age distribution is reasonably spread, with customers as young as 12 and as old as 83.

2.3 Data cleaning:

During the initial exploration of the dataset, the presence of missing values was evaluated in the InternetService column. To address the missing values in this column:

- The missing values were imputed using the label "Unknown", indicating customers for whom internet service type information was not available or not applicable.
- This was done using the .fillna("Unknown") method.

After imputation, a frequency count using .value_counts() on the InternetService column showed the following distribution:

- Fiber Optic: 395 customers
- DSL: 308 customers
- Unknown: 297 customers (i.e., previously missing values now categorized explicitly)

This approach not only preserves all records but also maintains the integrity of the dataset by clearly marking ambiguous or missing internet service information as "Unknown" instead of dropping or incorrectly imputing values.

3. MODEL

3.1 Model training:

Defining Dependent and Independent Variables

- The dataset was divided into independent variables (features) and a dependent variable (target).
- The dependent variable represents the churn status of customers.
- All other customer attributes serve as predictors used to model churn behaviour.

Feature Encoding

- Several variables in the dataset were categorical (e.g., Gender, Internet Service Type, Contract Type).
- To convert these non-numeric values into a format suitable for machine learning models, encoding techniques were applied:
  - Label Encoding was used for binary categorical variables such as Gender, Tech Support, and Churn.
  - One-Hot Encoding was used for multi-class categorical variables such as Contract Type and Internet Service.

- This step ensures that models can process categorical data correctly without assuming any ordinal relationship between categories.

Splitting the Dataset

- The processed dataset was split into training and test sets using an 80:20 ratio.
- Stratification was applied during the split to maintain the same class distribution in both sets.
- This separation allows the model to be trained on one portion of the data and evaluated on unseen data for fair performance assessment.

Feature Scaling

- Since the dataset includes continuous numeric features with varying units and ranges, standardization was applied.
- All numeric variables were scaled to have a mean of zero and a standard deviation of one.
- This step is particularly important for distance-based and gradient-based algorithms that are sensitive to feature magnitudes.

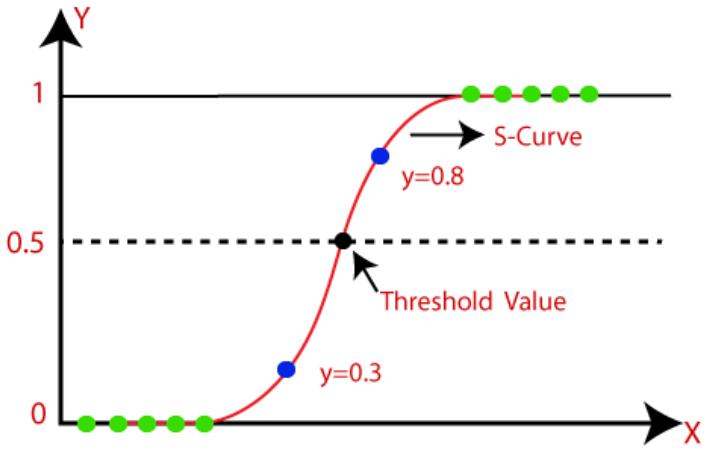
Handling Class Imbalance

- The target variable was found to be highly imbalanced, with significantly more non-churners than churners.
- To address this, SMOTE (Synthetic Minority Over-sampling Technique) was used on the training set.
- SMOTE generates new, synthetic examples of the minority class to balance the dataset and improve model learning.

Hyperparameter Tuning

- Hyperparameter tuning was performed to optimize model configurations for better performance.
- A grid search technique with cross-validation was applied to systematically evaluate different combinations of hyperparameters.
- This process helped in selecting the most effective model settings, avoiding both overfitting and underfitting.

3.2 Logistic Regression:

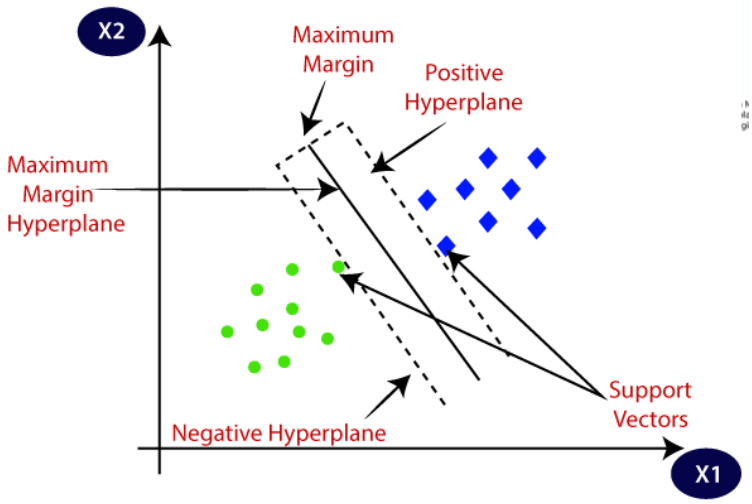


Definition:
Logistic Regression is a statistical classification algorithm used to model the probability of a binary outcome based on one or more input features. It uses the sigmoid function to output probabilities between 0 and 1.

Key Points:

- Used for binary classification problems (e.g., churn vs no churn).
- Outputs probabilities interpreted as class membership likelihood.
- Assumes a linear relationship between independent variables and the log-odds of the dependent variable.
- Sensitive to feature scaling.
- Easy to implement and interpret.
- Performs well when the classes are linearly separable.

3.3 Support Vector Machine:

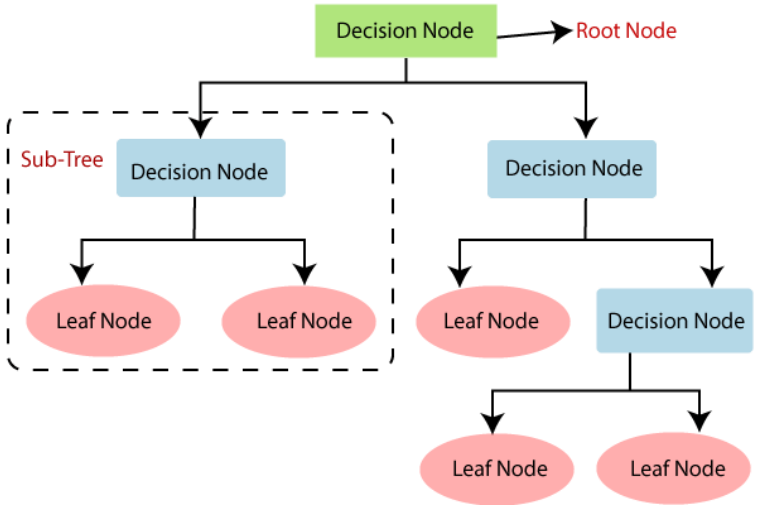


Definition:
SVM is a supervised learning algorithm that finds the optimal hyperplane that best separates the data into different classes by maximizing the margin between classes.

Key Points:

- Works well in high-dimensional spaces.
- Can use different kernels (linear, RBF, polynomial) to handle non-linear data.
- Effective when classes are clearly separable.
- Sensitive to feature scaling.
- Can be computationally intensive on large datasets.
- Offers good performance on imbalanced data with appropriate class weights.

3.4 Decision Tree:

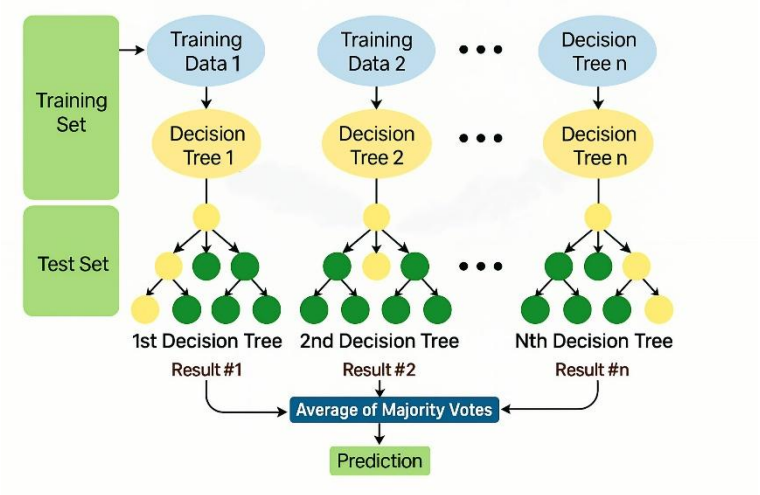


Definition:
A Decision Tree is a flowchart-like structure that recursively splits the dataset based on feature values to make predictions. Each internal node represents a decision rule, and each leaf node represents a class label.

Key Points:

- Simple and intuitive to interpret.
- No need for feature scaling.
- Can handle both categorical and numerical data.
- Prone to overfitting (can be controlled using max_depth, min_samples_split, etc.).
- Performs feature selection implicitly during training.

3.5 Random Forest:

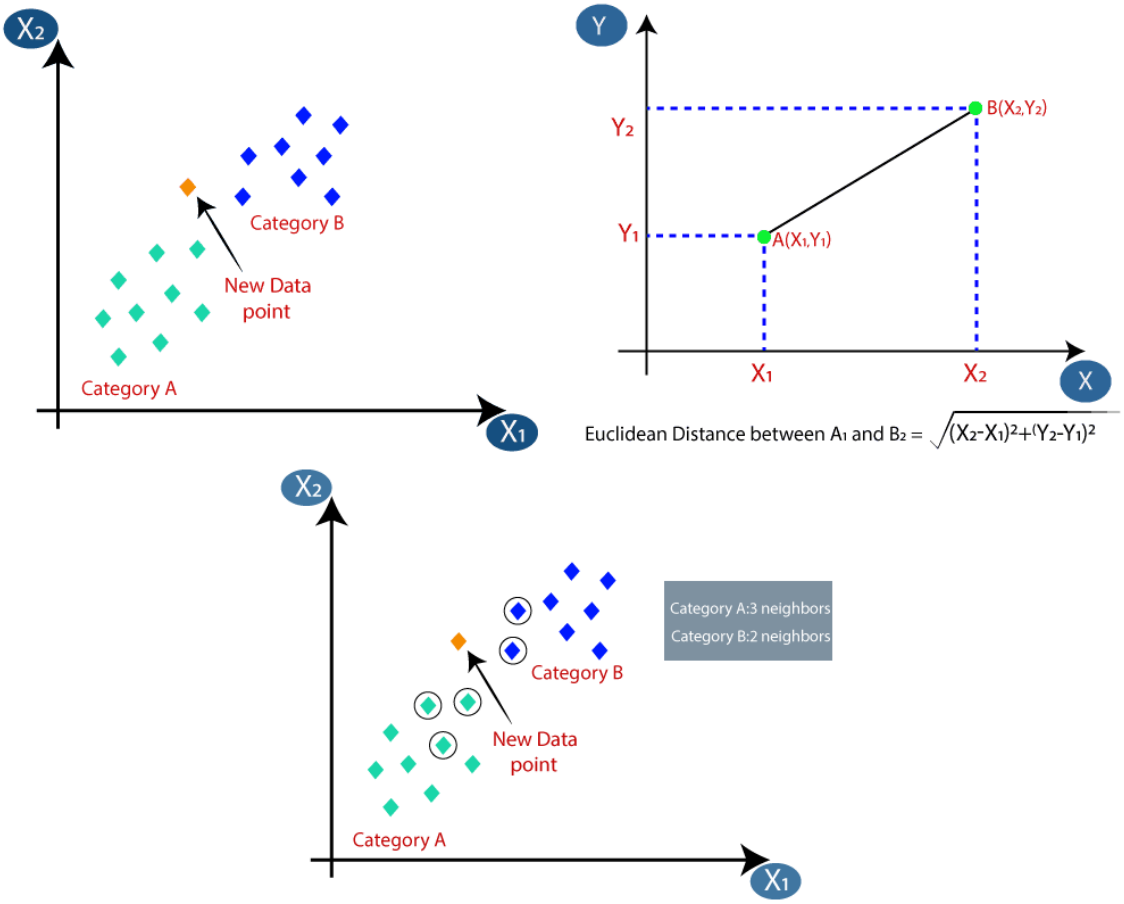


Definition:
Random Forest is an ensemble learning algorithm that builds multiple decision trees and combines their predictions through majority voting (classification) or averaging (regression).

Key Points:

- Reduces overfitting compared to individual decision trees.
- Handles missing values and categorical data effectively.
- Can rank feature importance.
- Robust to noise and class imbalance.
- Slower than a single tree, but more accurate and stable.

3.6 K-Nearest Neighbour:

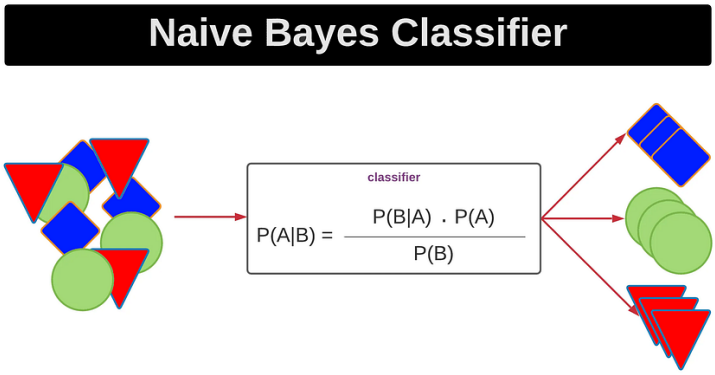


Definition:
KNN is a non-parametric algorithm that classifies a data point based on the majority label among its k nearest neighbours in the feature space.

Key Points:

- Simple and easy to understand.
- Sensitive to feature scaling and irrelevant features.
- Performance depends heavily on choice of k and distance metric.
- Computationally expensive during prediction (no training phase).
- Not ideal for high-dimensional data or large datasets.

3.7 Naïve Bayes:



Definition:
Naive Bayes is a probabilistic classifier based on Bayes' Theorem, assuming independence among features. Despite the "naive" assumption, it works surprisingly well in many real-world tasks.

Key Points:

- Very fast and scalable.
- Performs well on text classification and binary outcomes.
- Assumes features are conditionally independent given the class.
- Handles categorical and continuous features (with GaussianNB, MultinomialNB, etc.).
- Not sensitive to irrelevant features, but independence assumption can limit accuracy.

4. RESULTS AND ANALYSIS

- Classification algorithms help us automatically identify patterns in data and make informed decisions, such as predicting whether a customer will churn.
- They enable businesses to convert historical data into actionable insights, improving accuracy, efficiency, and strategic decision-making.
- In this case, we are using classification because churn prediction is a binary outcome problem. The goal is to classify each customer as either likely to churn (`Yes`) or not churn (`No`), making classification the most appropriate solution approach.

After training and testing multiple classification algorithms, each model was evaluated using key performance metrics such as accuracy, precision, recall, F1-score, confusion matrix, and AUC-ROC score. These metrics provide insight into the overall predictive capability, especially for the minority class (churners), which is crucial for business applications.

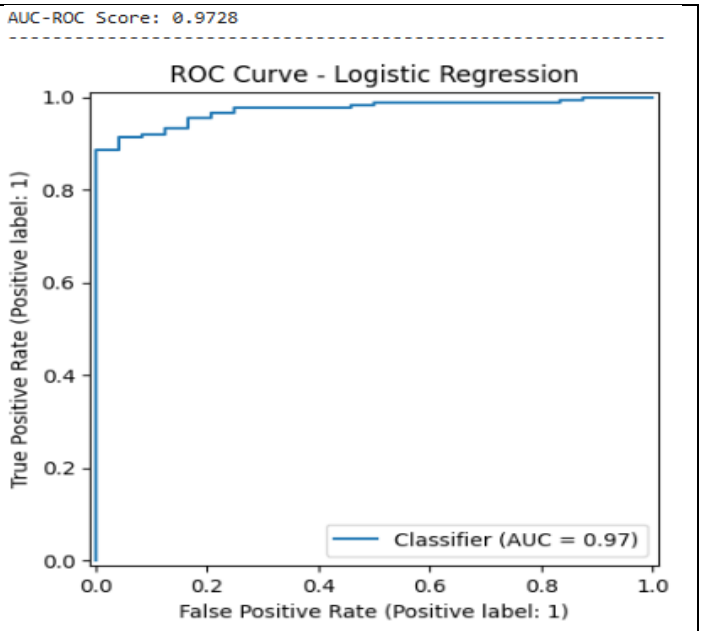
```
=> Model: Logistic Regression
```

```
Confusion Matrix:
----------------------------------------------------------
[[ 23   1]
 [ 18 158]]
```

```
Classification Report:
----------------------------------------------------------
              precision    recall  f1-score   support

           0     0.5610    0.9583    0.7077        24
           1     0.9937    0.8977    0.9433       176

    accuracy                         0.9050       200
   macro avg     0.7773    0.9280    0.8255       200
weighted avg     0.9418    0.9050    0.9150       200
```



- Accuracy: 90.5%
- Recall (churn class): 89.77%
- F1-Score (churn class): 0.9433
- AUC-ROC Score: 0.9728

Observation: Strong performance overall, especially in recall, but slightly lower than ensemble models in precision and F1.

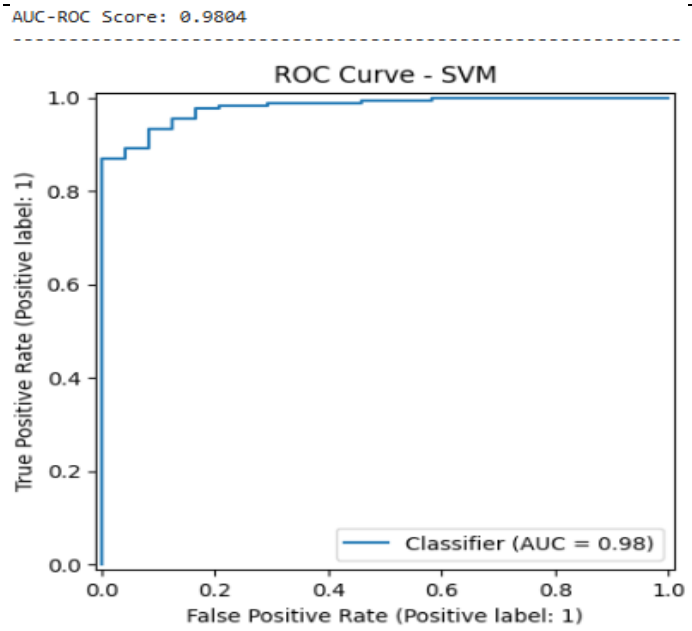
```
=> Model: SVM
```

```
Confusion Matrix:
----------------------------------------------------------
[[ 21   3]
 [  8 168]]
```

```
Classification Report:
----------------------------------------------------------
              precision    recall  f1-score   support

           0     0.7241    0.8750    0.7925        24
           1     0.9825    0.9545    0.9683       176

    accuracy                         0.9450       200
   macro avg     0.8533    0.9148    0.8804       200
weighted avg     0.9515    0.9450    0.9472       200
```



- Accuracy: 94.5%
- Recall (churn class): 95.45%
- F1-Score (churn class): 0.9683
- AUC-ROC Score: 0.9804

Observation: High performance and balanced precision-recall. SVM did well with optimized parameters and scaled features.

```
 => Model: Decision Tree
```

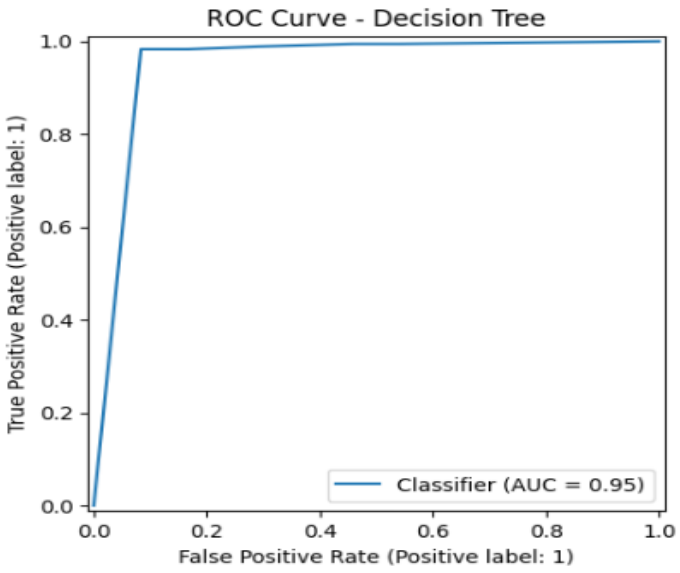
```
Confusion Matrix:
----------------------------------------------------------
[[ 22   2]
 [  3 173]]
```

```
Classification Report:
----------------------------------------------------------
              precision    recall  f1-score   support

           0     0.8800    0.9167    0.8980        24
           1     0.9886    0.9830    0.9858       176

    accuracy                         0.9750       200
   macro avg     0.9343    0.9498    0.9419       200
weighted avg     0.9755    0.9750    0.9752       200
```

```
AUC-ROC Score: 0.9512
```



- Accuracy: 97.5%
- Recall (churn class): 98.30%
- F1-Score (churn class): 0.9858
- AUC-ROC Score: 0.9512

Observation: Excellent performance but risk of overfitting; high precision and recall indicate strong training but may not generalize as well.

```
 => Model: Random Forest
```

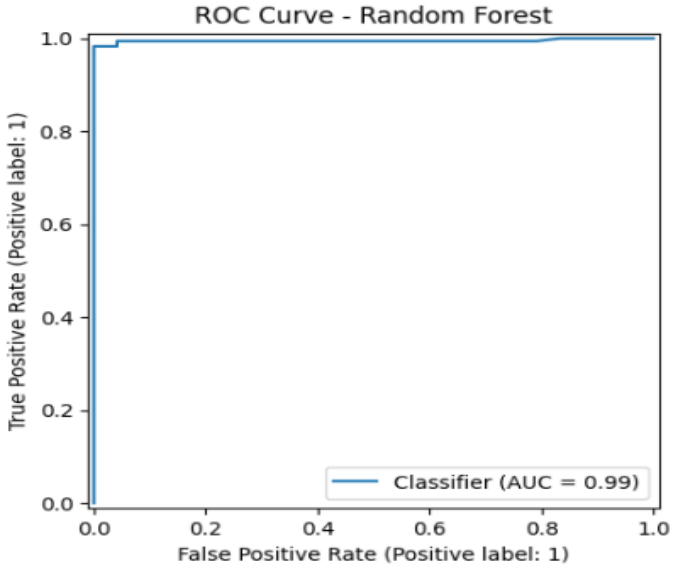
```
Confusion Matrix:
----------------------------------------------------------
[[ 23   1]
 [  2 174]]
```

```
Classification Report:
----------------------------------------------------------
              precision    recall  f1-score   support

           0     0.9200    0.9583    0.9388        24
           1     0.9943    0.9886    0.9915       176

    accuracy                         0.9850       200
   macro avg     0.9571    0.9735    0.9651       200
weighted avg     0.9854    0.9850    0.9851       200
```

```
AUC-ROC Score: 0.9949
```



- Accuracy: 98.5%
- Recall (churn class): 98.86%
- F1-Score (churn class): 0.9915
- AUC-ROC Score: 0.9949

Observation: Best-performing model overall with the highest scores across all metrics. Strong generalization due to ensemble nature.

```
=> Model: KNN
```

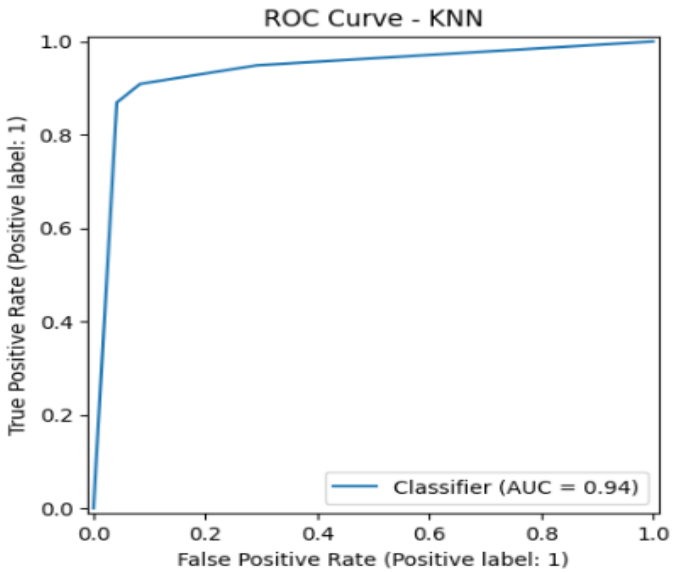
Confusion Matrix:

```
[[ 22   2]
 [ 16 160]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.5789	0.9167	0.7097	24
1	0.9877	0.9091	0.9467	176
accuracy			0.9100	200
macro avg	0.7833	0.9129	0.8282	200
weighted avg	0.9386	0.9100	0.9183	200

AUC-ROC Score: 0.9389



- Accuracy: 91.0%
- Recall (churn class): 90.91%
- F1-Score (churn class): 0.9467
- AUC-ROC Score: 0.9389

Observation: Performed well but more sensitive to data scaling and suffers on small sample edge cases.

```
=> Model: Naive Bayes
```

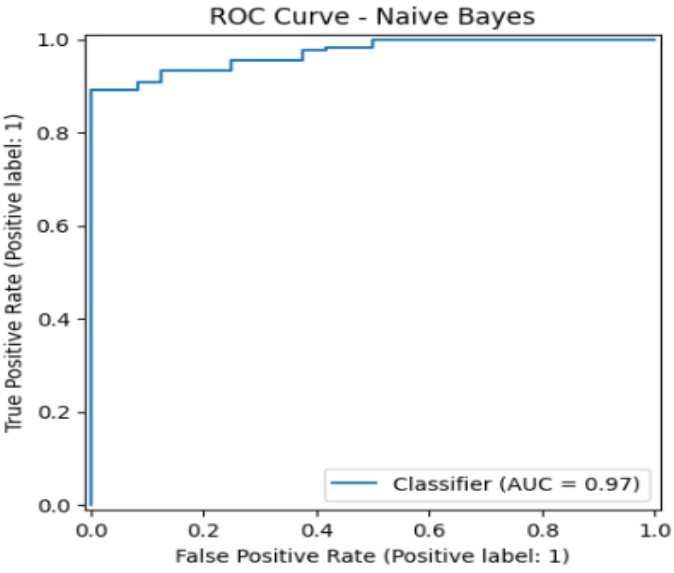
Confusion Matrix:

```
[[ 24   0]
 [ 38 138]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.3871	1.0000	0.5581	24
1	1.0000	0.7841	0.8790	176
accuracy			0.8100	200
macro avg	0.6935	0.8920	0.7186	200
weighted avg	0.9265	0.8100	0.8405	200

AUC-ROC Score: 0.9706



- Accuracy: 81.0%
- Recall (churn class): 78.41%
- F1-Score (churn class): 0.8790
- AUC-ROC Score: 0.9706

Observation: Quick and simple, but performance is limited due to strong independence assumptions.

5. CONCLUSION

This project aimed to develop an effective machine learning solution to predict customer churn using real-world telecom customer data. By leveraging supervised learning techniques, the objective was to help businesses proactively identify customers who are likely to leave and design strategic interventions to retain them.

The project followed a structured data science pipeline, beginning with an understanding of the dataset, followed by thorough data cleaning, handling missing values, encoding categorical variables, scaling numerical features, and addressing class imbalance using SMOTE. Exploratory Data Analysis (EDA) provided valuable business insights such as the fact that customers with month-to-month contracts, no tech support, and higher monthly charges were more prone to churn. These insights were supported by statistical summaries and visualizations that revealed important patterns in customer behaviour.

Six classification algorithms were implemented: Logistic Regression, Support Vector Machine (SVM), Decision Tree, Random Forest, K-Nearest Neighbours (KNN), and Naive Bayes. Hyperparameter tuning was applied to optimize model performance, and evaluation was done using multiple metrics including accuracy, recall, F1-score, and AUC-ROC score.

Among all the models, the Random Forest classifier emerged as the best performer, achieving the highest accuracy (98.5%), recall (98.86%), and AUC-ROC score (0.9949), making it the most reliable and generalizable model for this classification task. It managed to capture the churn pattern effectively while avoiding overfitting. In contrast, models like Naive Bayes underperformed, especially in identifying churners, due to its simplistic assumptions. The Decision Tree model, while achieving high accuracy, showed signs of overfitting and may not generalize well to unseen data.

The use of performance metrics like recall and AUC-ROC was especially important in this business context, as they helped assess how well the models could identify churners, a priority for retention focused strategies.

Overall, the project demonstrates the power and practicality of using machine learning for business decision-making. The churn prediction model developed here can be further enhanced and integrated into a larger customer relationship management (CRM) system to support real-time retention efforts. In the future, additional features, time series behaviour, or customer feedback data could be incorporated to improve accuracy and personalization even further.

6. REFERENCES

- Kaggle Dataset: Telecom Customer Churn Insights
  Source of project dataset.
  https://www.kaggle.com/datasets/abdullah0a/telecom-customer-churn-insights-for-analysis
- Images used in the report:
  https://encord.com
  https://medium.com
- Report:
  A business-focused explanation of churn, its impact, and how prediction helps.
  https://www.ibm.com/topics/customer-churn
  https://www.geeksforgeeks.org/machine-learning/python-customer-churn-analysis-prediction
  https://www.simplilearn.com/tutorials/machine-learning-tutorial/classification-in-machine-learning