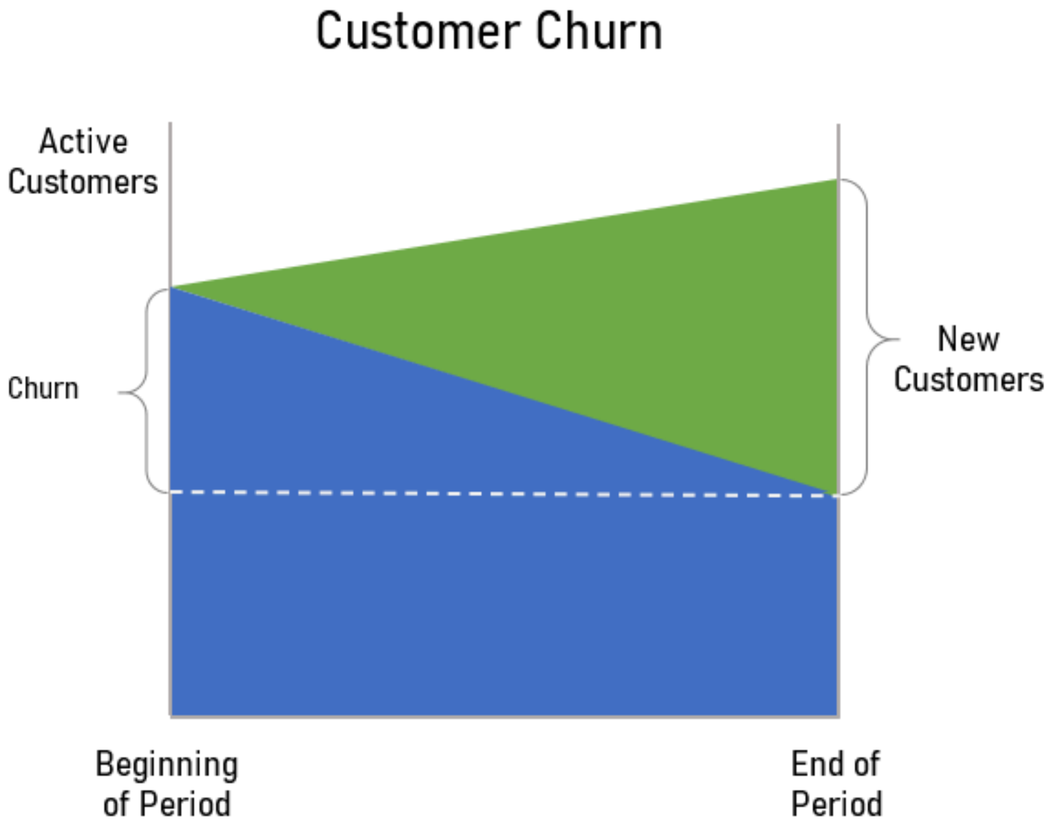
By Gauri Bhardwaj

Understanding Churn Prediction

How analysis can help companies and consumers

Table of Contents

Points for discussion



What is CHURN RATE?

Customer churn is the **percentage of customers that stopped using your company's product or service** during a certain time frame.

Churn rate is a key indicator of customer satisfaction.

- A **low** churn rate signifies satisfied customers
- A **high** churn rate indicates that customers are leaving

Churn serves as a valuable **measure of growth potential**. It tracks the customers you lose, while growth rates track the new customers you gain. Analyzing these metrics together reveals your business's overall growth. If your growth rate exceeds your churn rate, your business is expanding. Conversely, if churn surpasses growth, your business is contracting.

A Concept Learning Task – Enjoy Sport
Training Examples

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	**YES**
2	Sunny	Warm	High	Strong	Warm	Same	**YES**
3	Rainy	Cold	High	Strong	Warm	Change	**NO**
4	Sunny	Warm	High	Strong	Warm	Change	**YES**

ATTRIBUTES **CONCEPT**

- A set of example days, and each is described by six attributes.
- The task is to learn to predict the value of EnjoySport for arbitrary day, based on the values of its attribute values.

Features/ Attributes:

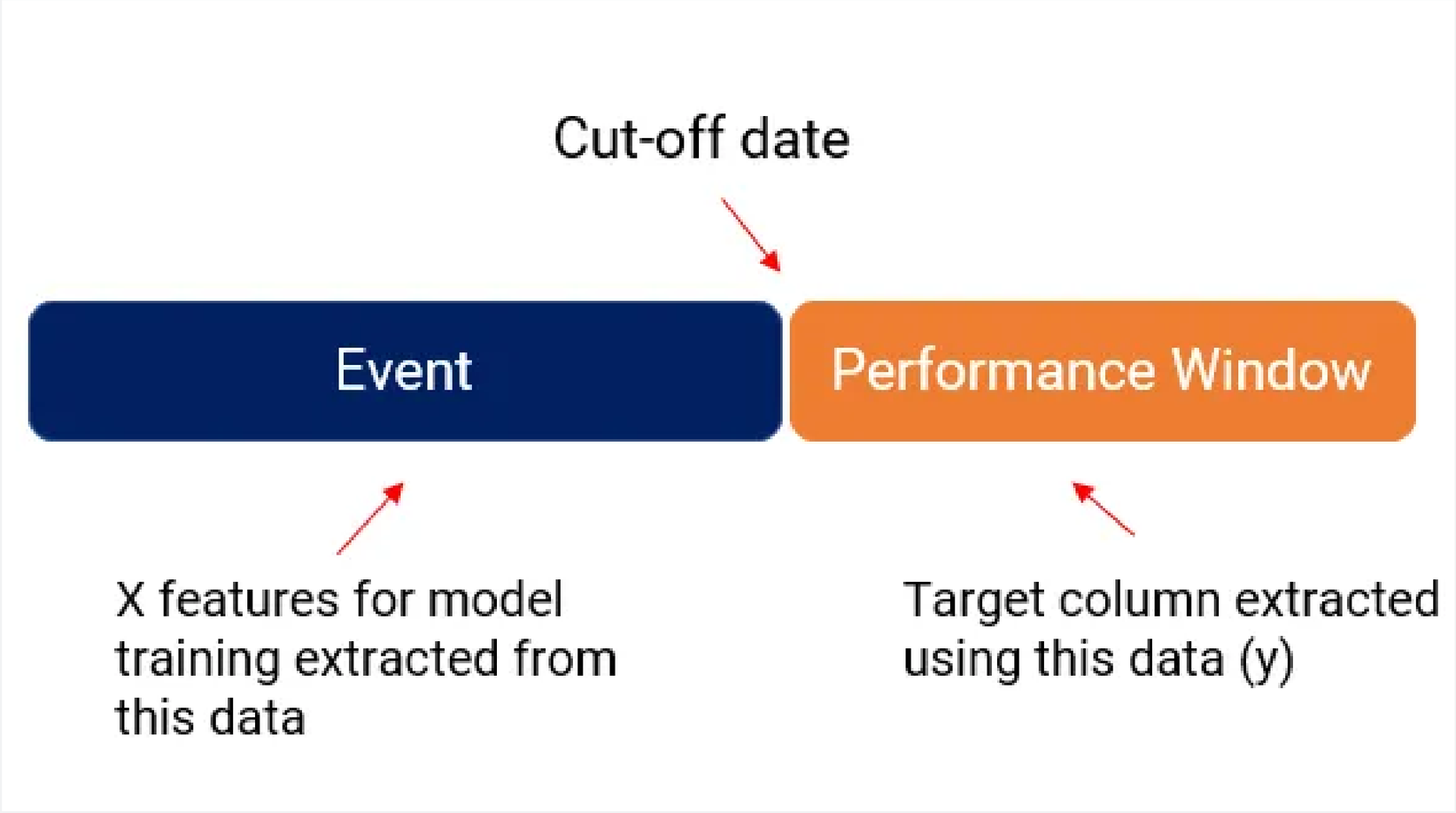
Properties of a sample that directly or indirectly influences the target variable we want to predict.

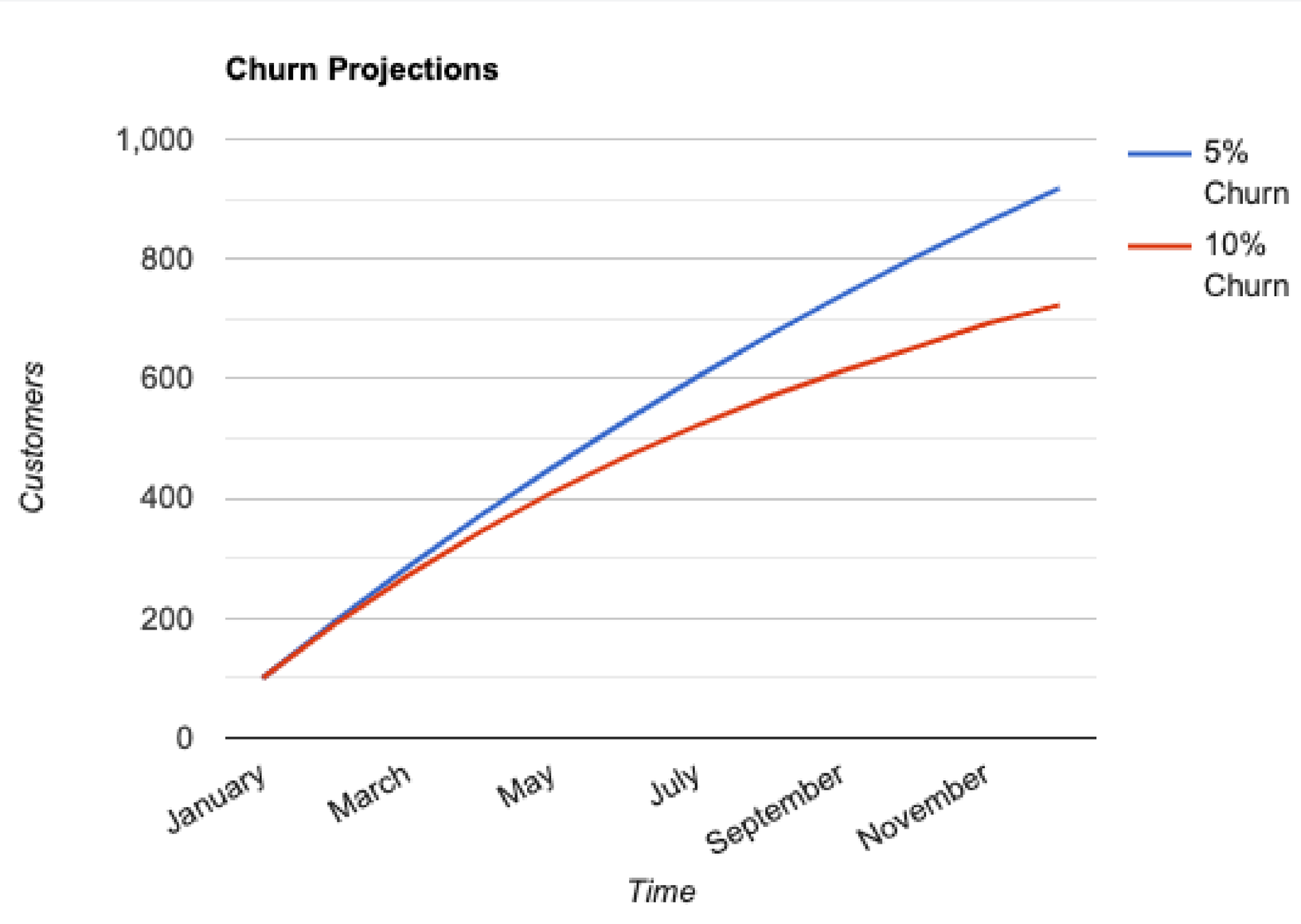
Target Variable:

Target Variable that we want to predict in this case is of type “Yes” or “No” i.e. it can be binary classified.

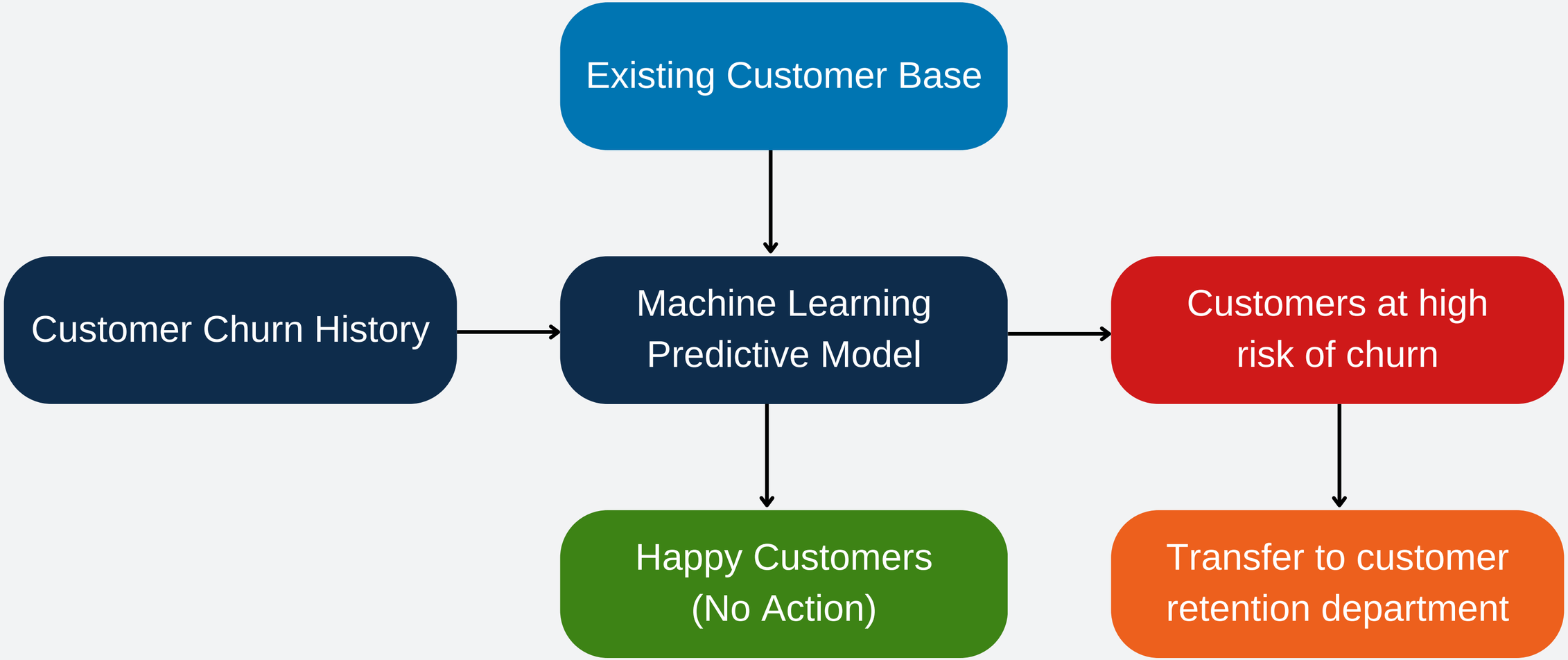
The response variable can also be a continous value (eg: predict house prices based on area and no. of rooms)

Dataset Composition




Churn Projections
1,000
800
600
400
200
0
Customers
January
March
May
July
September
November
Time
5% Churn
10% Churn

Customer Churn Model Workflow



More about
CLASSIFICATION MODELS

1. LOGISTIC REGRESSION

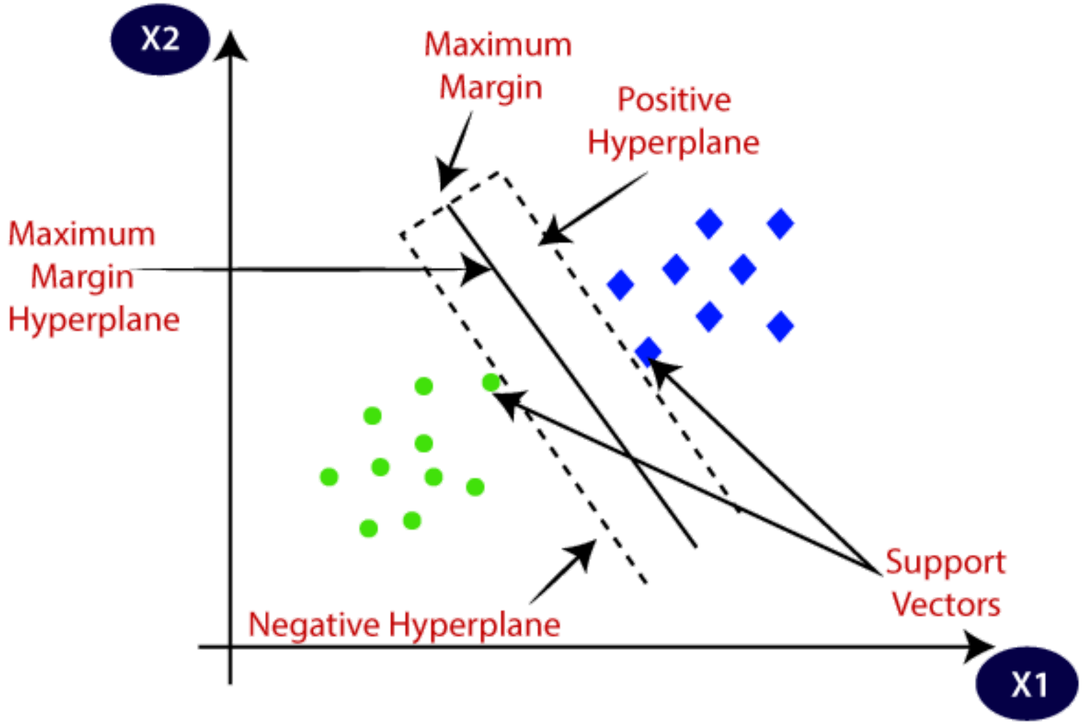
It is a predictive analysis algorithm and based on the concept of **probability**.

Some of the examples of classification problems:

- Classifying Emails: **spam or not spam**,
- Classifying Online transactions: Fraud or not Fraud

Logistic regression transforms its output using the logistic **sigmoid function** to return a probability value.

2. SUPPORT VECTOR MACHINE

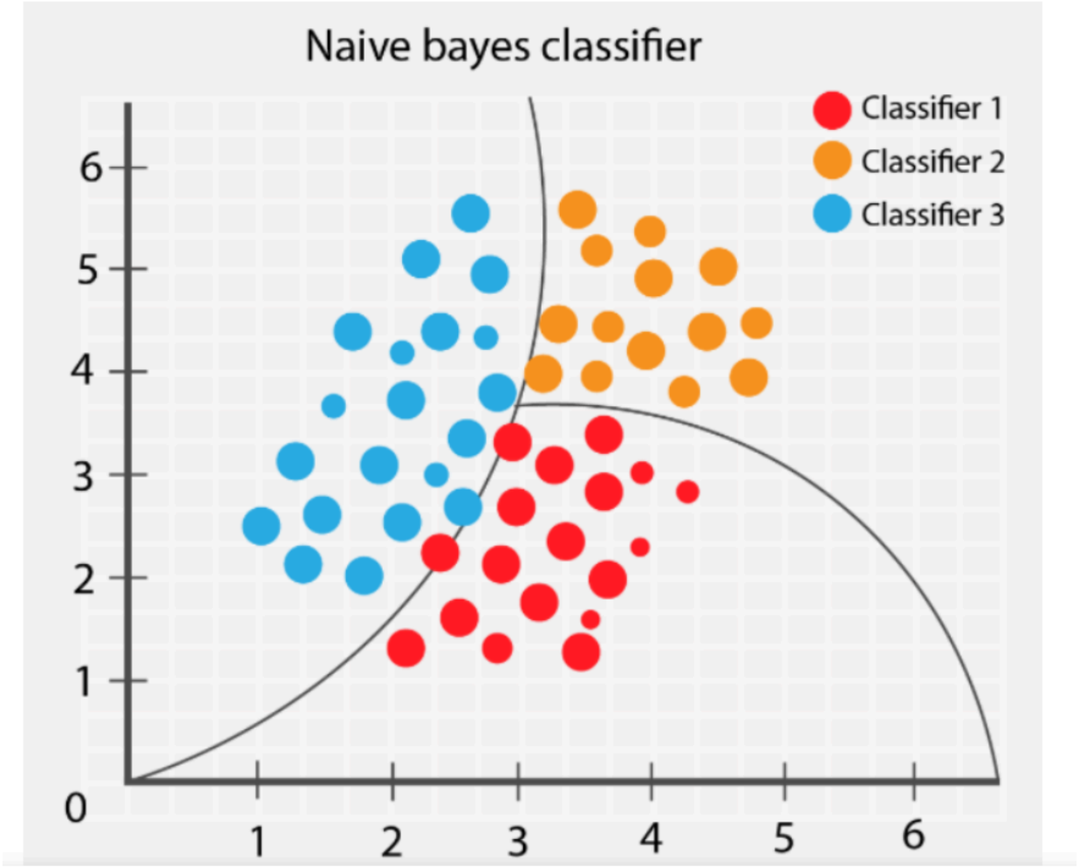


It is a **supervised machine learning** algorithm used for both classification and regression.

The main objective of the SVM algorithm is to find the **optimal hyperplane** in an N-dimensional space that can separate the data points in different classes in the feature space.

The **dimension of the hyperplane depends upon the number of features**. If the number of input features is two, then the hyperplane is just a line. If the number of input features is three, then the hyperplane becomes a 2-D plane.

3. NAIVE BAYES CLASSIFIER

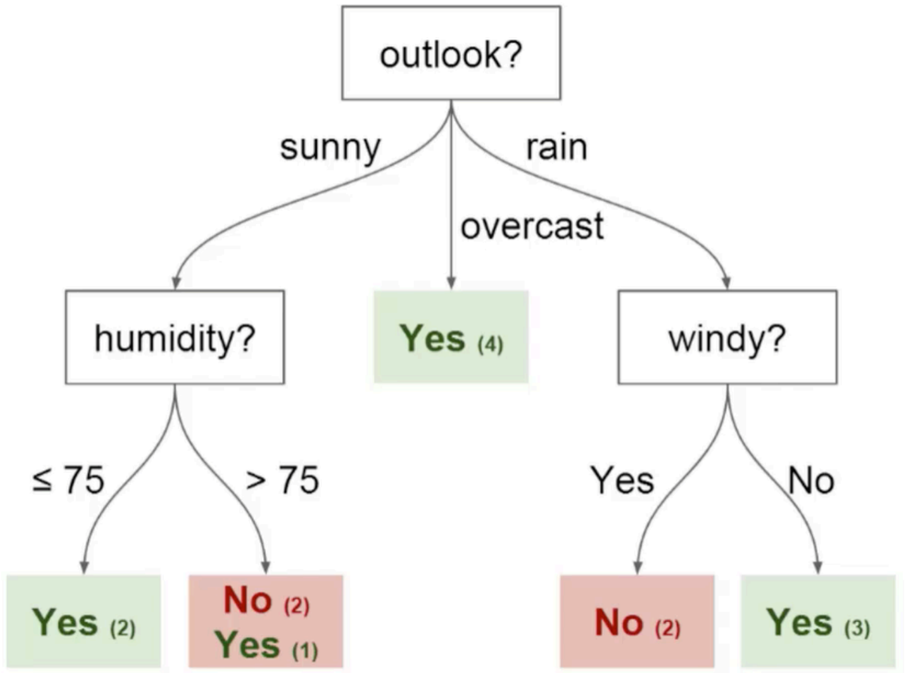


Naive Bayes classifiers is a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is **independent** of each other.

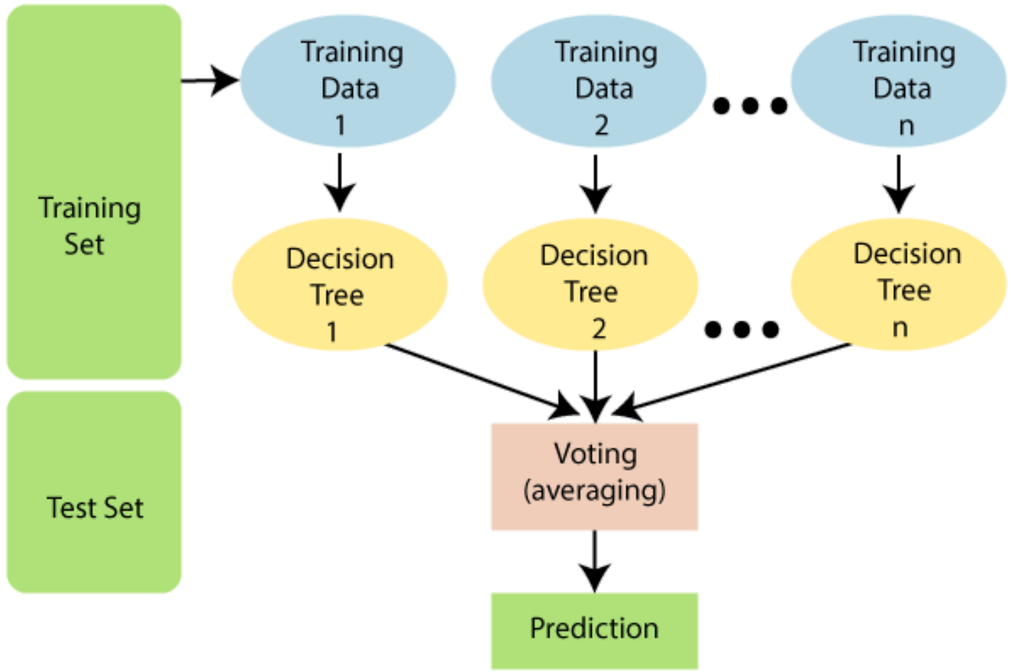
This model predicts the probability of an instance belongs to a class with a given set of feature value. It is a **probabilistic classifier**. In other words, each feature contributes to the predictions with no relation between each other. In real world, this condition satisfies rarely. It uses Bayes theorem in the algorithm for training and prediction.

Some other Classification Models

4. DECISION TREES



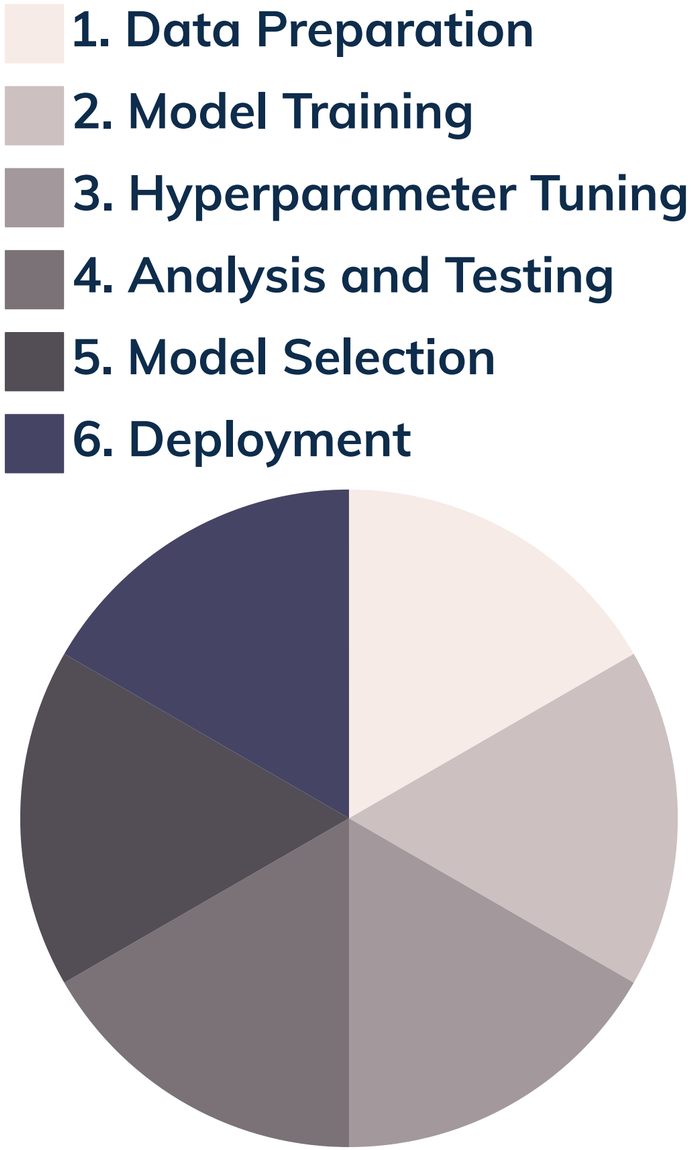
5. RANDOM FOREST



The Machine Learning Process

Points for discussion

More and more companies realize that specific job titles and profiles can be accomplished while working from home.



Data Preparation

1) Data Gathering

- Identifying various data sources
- Collect Data
- Integrate data obtained from different sources

2) Data Exploration

- Understand quality, characteristics and nature of your data
- Find correlations, general trends and outliers

3) Data Wrangling

- Cleaning the data and converting raw data into usable format
- Deal with **missing values, invalid data, outliers, duplicates**
- Filtering Techniques of feature selection

4) Train- Test Split: CROSS-VALIDATION

- Technique for evaluating the performance of a machine learning algorithm.

- The procedure involves taking a dataset and dividing it into two subsets. The **training subset** is used to fit the model and the **testing subset** is not used to train the model; instead, the input element of the dataset is provided to the model, then predictions are made and compared to the expected values.

Nevertheless, common split percentages include:

- Train: 80%, Test: 20%
- Train: 67%, Test: 33%

Data Preprocessing Steps

Data Pre-processing

1) Convert data type of variables which are misclassified.
2) Removing Duplicate records
3) Removing Unique value variables
4) Removing Zero variance variables
5) Outlier Treatment
    - Using Boxplot: Q3+(1.5*IQR) & Q1-(1.5*IQR)
    - Standardization: +/- 3 Sigma approach
    - Capping & Flooring
6) 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
7) Removing the highly correlated variables
8) Multicollinearity (VIF > 5)

MODEL TRAINING

Model training is a crucial process in the field of Machine Learning, which involves **teaching a model to identify patterns, relationships, or trends within a given dataset** in order to generate meaningful insights. During the model training process, a dataset is carefully curated as the primary source of knowledge.

WRT SUPERVISED LEARNING

In this method, the training dataset is labeled, by **comparing the model's predictions with these labeled data points**, the algorithm identifies any discrepancies and adjusts the model accordingly. This iterative process continues until the model achieves a desired level of accuracy.

WRT UNSUPERVISED LEARNING

By **exploring the relationships between data points**, the unsupervised learning algorithm identifies clusters or groups that share similar characteristics.

Hyperparameter Tuning

Process of selecting the optimal values for a machine learning model's hyperparameters. Hyperparameters are **settings that control the learning process** of the model, such as the **learning rate**, the number of neurons in a neural network, or the kernel size in a support vector machine. The goal of hyperparameter tuning is to find the values that lead to the best performance on a given task.

DECISION TREE

- Accuracy Before: 88.08%
- Accuracy After: 91.2%

SVM

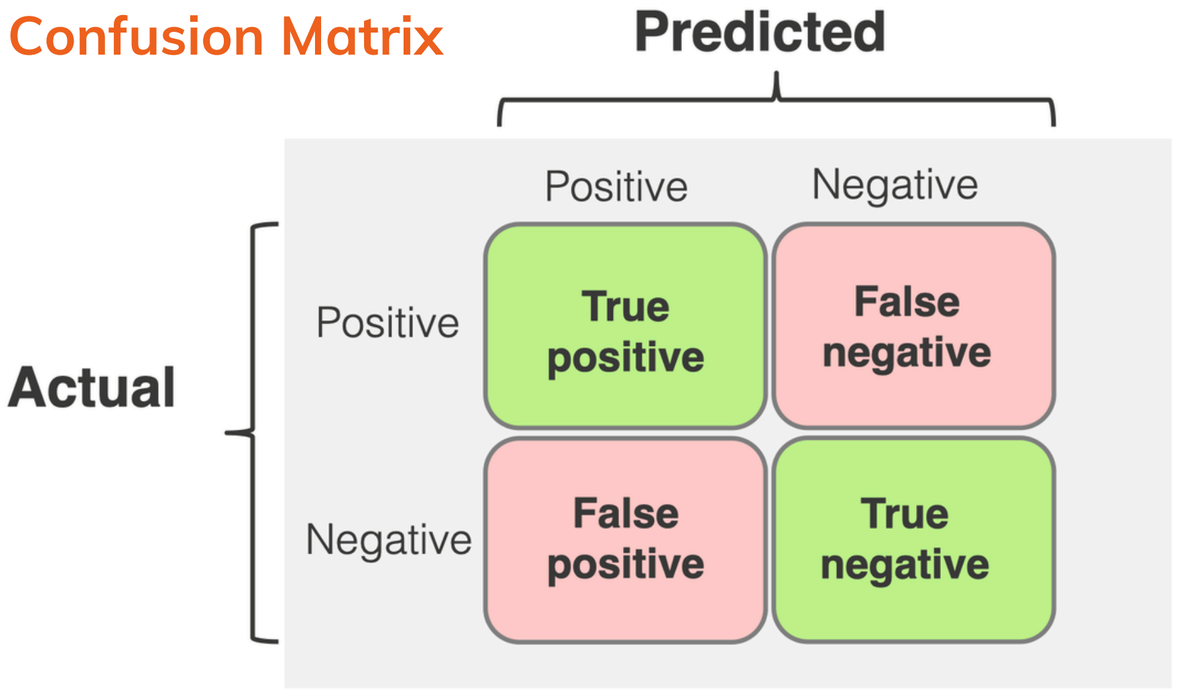
- Accuracy Before: 90.2%
- Accuracy After: 78.2%

Model Training → Hyperparameter Tuning → Find Accuracy and evaluate performance

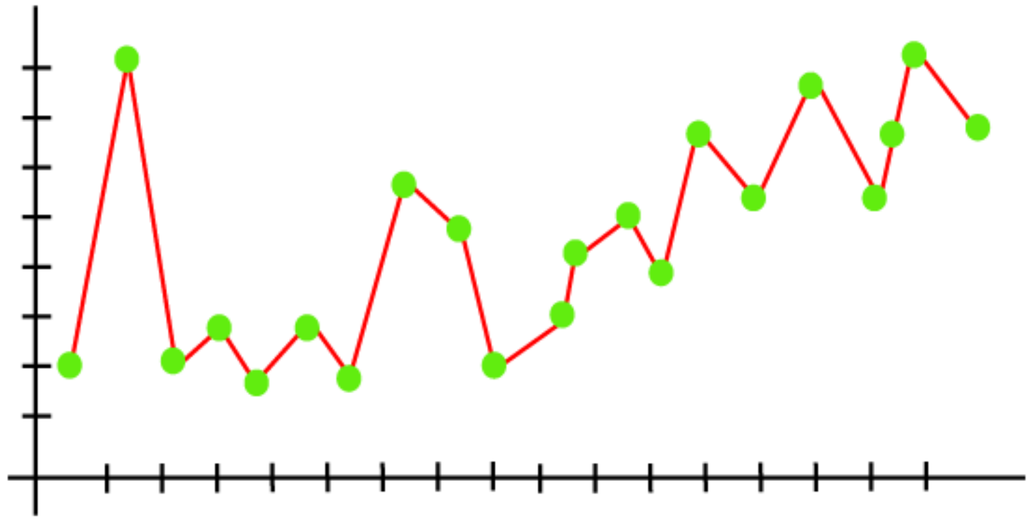
Popularly used performance metrics for classification problems are:

- Accuracy
- Precision
- Recall
- F-score
- ROC

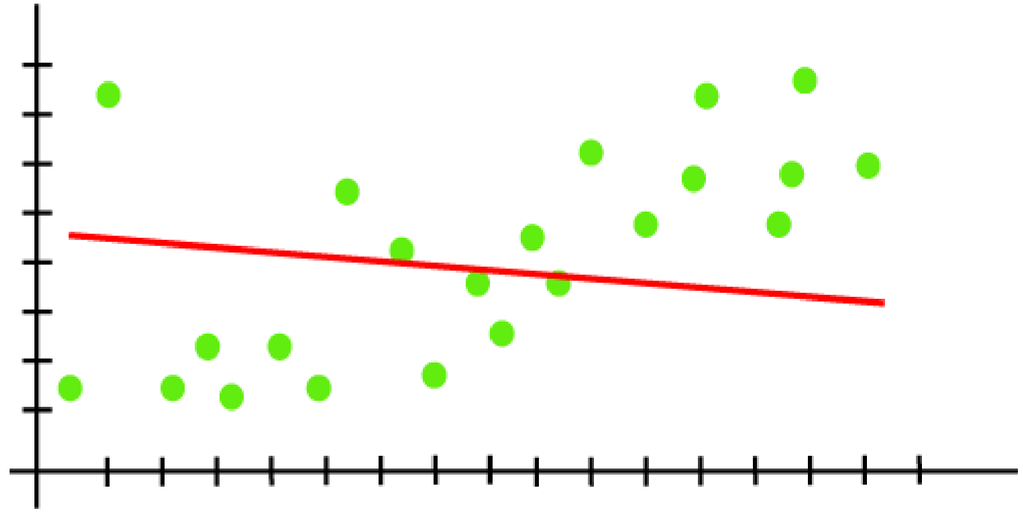
Confusion Matrix



Model Optimization: Reduce Overfitting and Underfitting



Overfitting occurs when our machine learning model **tries to cover all the data points** or more than the required data points present in the given dataset. Even if model works well for training data, it fails for new samples.



In the case of underfitting, the model is not able to learn enough from the training data, and hence it reduces the accuracy and produces unreliable predictions. High bias and Low variance.

Model Selection

DECISION TREE (77.4%)

```
Best Parameters found by GridSearchCV:
{'max_depth': 10, 'max_features': None}

Tuned Model Confusion Matrix:
[[2924  513]
 [ 615  948]]

Tuned Model Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.85      0.84      3437
           1       0.65      0.61      0.63      1563

    accuracy                           0.77      5000
   macro avg       0.74      0.73      0.73      5000
weighted avg       0.77      0.77      0.77      5000


Tuned Model Accuracy: 77.44 %
```

RANDOM FOREST (80.34%)

```
Fitting 5 folds for each of 8 candidates, totalling 40 fits
Best Hyperparameters: {'max_depth': 10, 'max_features': 'sqrt', 'n_estimators': 50}

Tuned Model Confusion Matrix:
[[3103  334]
 [ 649  914]]

Tuned Model Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.90      0.86      3437
           1       0.73      0.58      0.65      1563

    accuracy                           0.80      5000
   macro avg       0.78      0.74      0.76      5000
weighted avg       0.80      0.80      0.80      5000


Tuned Model Accuracy: 80.34 %
```

Model Selection

LOGISTIC REGRESSION (80.28%)

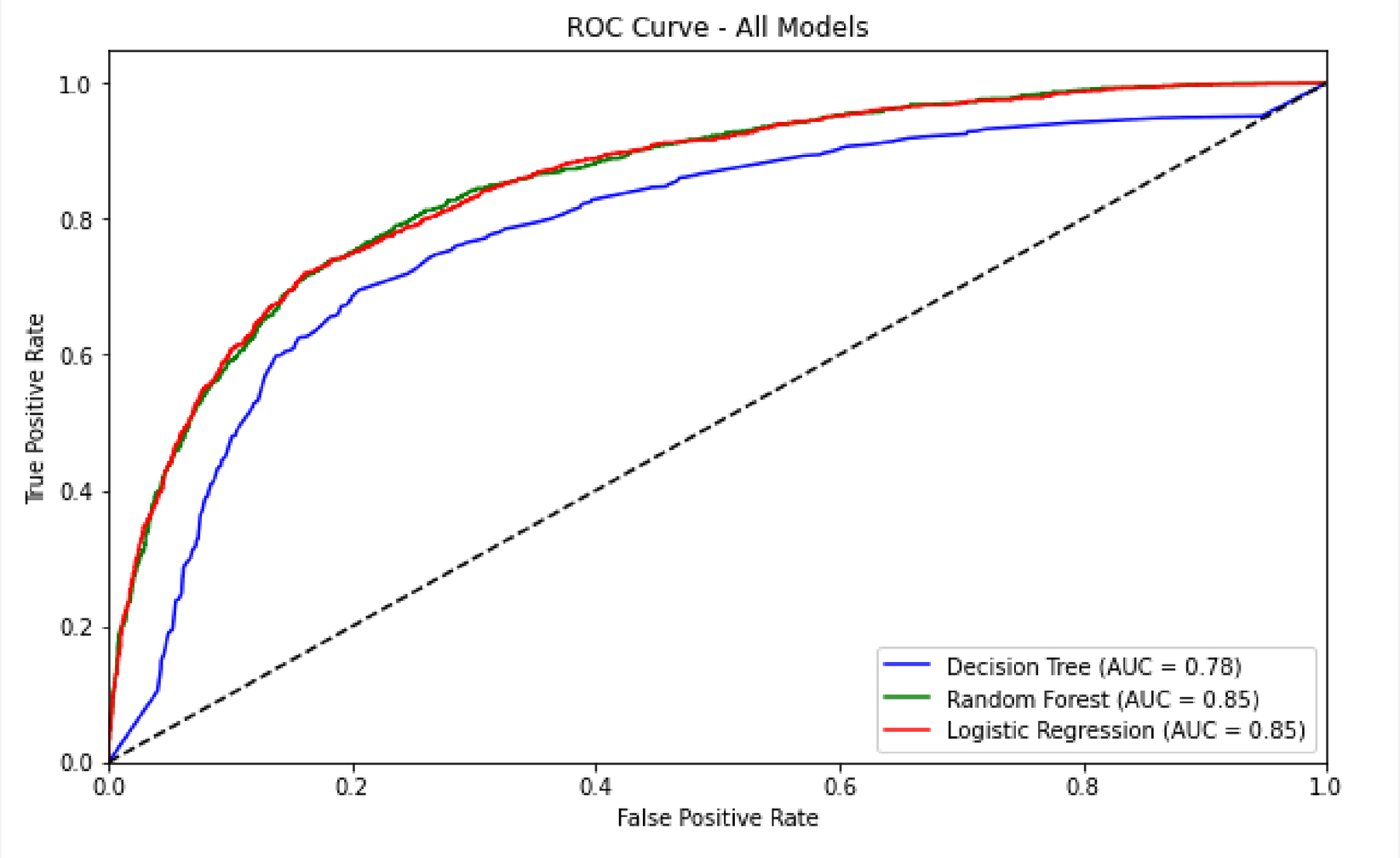
```
Fitting 5 folds for each of 60 candidates, totalling 300 fits
Best Parameters: {'C': 0.01, 'max_iter': 1000, 'penalty': 'l1', 'solver': 'liblinear'}
Accuracy of Best Model: 0.8028
Confusion Matrix of Best Model:
[[3092  345]
 [ 641  922]]
Classification Report of Best Model:
              precision    recall  f1-score   support

           0       0.83      0.90      0.86      3437
           1       0.73      0.59      0.65      1563

    accuracy                           0.80      5000
   macro avg       0.78      0.74      0.76      5000
weighted avg       0.80      0.80      0.80      5000


Accuracy Percentage of Best Model: 80.28 %
```

Cumulative ROC Curve

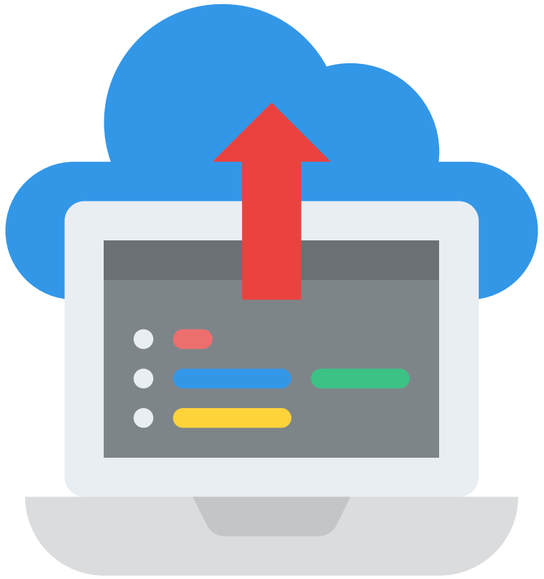


Final Testing and Deployment

LAST STEP OF MACHINE LEARNING CYCLE:

If the model we developed is producing an accurate result as per the customer's requirements with an acceptable speed then we deploy our model into the real-world system.

Before deploying ensure whether the model is increasing its performance using available data or not.



Thank You!