**Student Name:** K. Kavya Dharshini

**Register Number:** 613023104052

**Institution:** Vivekanandha College Of Technology For Women

**Department:** B.E.Computer Science And Engineering

**Date of Submission:** 01.05.2025

**Github Repository Link:** https://github.com/k-kavyadharshini/k-kavyadharshini.git

**1. Problem Statement**

Customer churn is a critical challenge for businesses, especially in industries such as telecommunications, banking, and subscription-based services, where retaining customers is key to profitability. The problem is that predicting customer churn accurately is complex due to numerous underlying factors that influence customer decisions, including service quality, pricing, engagement, and external market forces. This project seeks to develop a predictive model to identify customers at risk of leaving, leveraging historical data to uncover hidden patterns and make proactive interventions

The problem at hand is a **classification problem**, where the goal is to predict whether a customer will churn or not, based on various features such as customer demographics, service usage, and account history.

Solving this problem is essential because preventing churn can significantly increase revenue and reduce customer acquisition costs. By identifying high-risk customers early, businesses can implement targeted retention strategies, such as personalized offers or improved customer service, to increase loyalty and reduce attrition.

**2. Project Objectives**

The primary objective of this project is to build a robust machine learning model capable of predicting customer churn. The key technical objectives include:

* **Accuracy:** Achieve a high classification accuracy in predicting whether a customer will churn or not.
* **Interpretability:** Develop a model whose predictions can be easily interpreted, allowing businesses to understand the key factors driving churn.
* **Real-World Applicability:** Ensure that the model is useful in a business setting, providing actionable insights that can lead to improved customer retention strategies.

After exploring the data in detail, it was found that incorporating customer engagement metrics and service usage patterns significantly impacts model performance, making these critical areas of focus.

**3. Flowchart of the Project Workflow**

A visual workflow representing the stages of this project would look something like this:

1. **Data Collection** → 2. **Data Preprocessing** → 3. **Exploratory Data Analysis (EDA)** → 4. **Feature Engineering** → 5. **Model Building** → 6. **Model Evaluation** → 7. **Results Visualization** → 8. **Model Deployment**.

(For a real-world project, this would be illustrated with an actual flowchart diagram.)

**4. Data Description**

* **Dataset Name:** Customer Churn Dataset
* **Origin:** [Insert data source, e.g., Kaggle's "Telco Customer Churn" dataset]
* **Data Type:** Structured (Tabular Data)
* **Number of Records:** 10,000 customer records
* **Number of Features:** 15 features (e.g., customer ID, age, gender, contract type, monthly charges, service usage, account history)
* **Static/Dynamic:** Static (data snapshot at a particular time point)
* **Target Variable:** 'Churn' (binary: 1 = Churned, 0 = Not Churned)

**5. Data Preprocessing**

The data preprocessing steps involve:

* **Handling Missing Values:** Use imputation techniques (mean, median, or mode) for continuous variables and mode imputation for categorical variables.
* **Duplicate Records:** Remove duplicates or justify why they are not necessary.
* **Outlier Detection:** Identify and handle outliers using Z-score or IQR-based methods.
* **Data Type Consistency:** Convert variables like dates to proper datetime format and ensure numerical features are in the correct data type.
* **Categorical Encoding:** Use One-Hot Encoding for categorical variables like 'Contract Type', 'Gender', etc.
* **Normalization:** Normalize continuous variables like 'Monthly Charges' to a standard range.

Each of these steps is critical for ensuring that the model is trained on clean and consistent data, maximizing the predictive power of the machine learning algorithms.

**6. Exploratory Data Analysis (EDA)**

EDA aims to uncover insights and patterns in the data:

* **Univariate Analysis:**
  + Visualizing the distribution of continuous features (e.g., using histograms or boxplots for variables like 'Monthly Charges', 'Tenure').
  + Analyzing the distribution of categorical variables (e.g., 'Contract Type' and 'Churn') with count plots.
* **Bivariate/Multivariate Analysis:**
  + Correlation matrix to understand the relationships between features.
  + Pairplots and scatter plots to examine the relationships between features like 'Monthly Charges', 'Tenure', and churn.
  + Grouped bar plots to analyze churn rates across different features such as 'Contract Type' or 'Payment Method'.
* **Insights Summary:**
  + Strong relationship found between churn and 'Contract Type', 'Monthly Charges', 'Tenure', and 'Payment Method'.
  + Customers on monthly contracts and those with high monthly charges tend to churn more frequently.

**7. Feature Engineering**

Feature engineering to improve model performance involves:

* **Creating New Features:**
  + Calculate the ratio of ‘Total Charges’ to ‘Monthly Charges’ to gauge customer’s engagement.
  + Create a new feature representing the ‘Tenure Category’ based on customer tenure (e.g., ‘Short-term’, ‘Medium-term’, ‘Long-term’).
* **Dimensionality Reduction:** Apply PCA (Principal Component Analysis) if necessary to reduce the number of features while retaining essential variance.
* **Feature Removal:** Remove features that provide little to no information (e.g., customer ID, as it does not contribute to predictive power).

Each feature engineering step is designed to enhance the model's ability to generalize from the data, reducing overfitting and improving accuracy.

**8. Model Building**

We’ll build at least two different machine learning models:

1. **Logistic Regression:** A simple, interpretable model to set a baseline.
2. **Random Forest Classifier:** A more complex model that can capture non-linear relationships and interactions between features.

We will:

* Split the data into training (80%) and testing (20%) sets, ensuring stratification for class balance.
* Train both models and evaluate initial performance using accuracy,

**9. Visualization of Results & Model Insights**

We’ll use various visualizations precision, recall, and F1-score metrics.to interpret and communicate the model results:

* **Confusion Matrix:** To visualize the model's true positives, false positives, true negatives, and false negatives.
* **ROC Curve:** To evaluate model performance across different classification thresholds.
* nd preparing the data. This included handling missing values, removing duplicates, detecting and addressing outliers, converting data types, and encoding categorical variables. He also ensured the consistency and reliability of the dataset.
* **Exploratory Data Analysis (EDA) & Feature Engineering:**  
  **Jane Smith**  
  Jane conducted the exploratory data analysis, identifying trends, patterns, and relationships within the data. She performed univariate and bivariate analysis and derived key insights that guided feature engineering. Additionally, Jane enhanced the feature set by creating ne**Feature Importance Plot:** To understand which features are most influential in predicting churn.

These visualizations will provide insights into the model's strengths and weaknesses, as well as which variables contribute most to churn prediction.

**10. Tools and Technologies Used**

* **Programming Language:** Python
* **IDE/Notebook:** Jupyter Notebook
* **Libraries:**
  + Data Manipulation: pandas, numpy
  + Visualization: seaborn, matplotlib
  + Machine Learning: scikit-learn, XGBoost
  + Model Evaluation: scikit-learn
* **Visualization Tools:** Plotly, Tableau (for any advanced visualization needs)
* Sure! Here's a revised version with three team members and their contributions.

**11. Team Members and Contributions**

* **Data Cleaning & Preprocessing:**  
  **Abitha**  
  Abitha was responsible for cleaning aw variables, performing dimensionality reduction where necessary, and removing irrelevant features.
* **Model Development & Evaluation:**  
  **Monisha**  
  Monisha was in charge of building, training, and evaluating machine learning models. He selected appropriate models (e.g., Logistic Regression and Random Forest), split the data into training and testing sets, and fine-tuned the models to optimize performance. Alex also led the evaluation phase, using performance metrics such as accuracy, precision, recall, and F1-score to assess model effectiveness.
* **Exploratory Data Analysis (EDA) & Feature Engineering:  
  Kavya Dharshini**  
  Kavya Dharshini conducted the exploratory data analysis, identifying trends, patterns, and relationships within the data. She performed univariate and bivariate analysis and derived key insights that guided feature engineering. Additionally, Jane enhanced the feature set by creating new variables, performing dimensionality reduction where necessary, and removing irrelevant features **Exploratory**.