Churn Prediction Via Customer Segmentation

Students' Names: Wahidullah Maqsood (12130822028) (3rd year)

Department: Computer Science Engineering with Specialization in *Artificial Intelligence & Machine Learning*

Semester: 6th

Bengal Institute of Technology

Aim of the Project:
The project aims to predict customer churn using machine learning by first segmenting customers based on their characteristics, then training models with these segments and other features.

Keywords:
Customer Churn, Customer Segmentation, k-means Clustering, Logistic Regression, Gradient Boosting, SMOTE, Feature Engineering, Telco.

---

1. Introduction

Customer churn when users stop using a service is a critical concern, especially for subscription-based industries like telecom, where retaining customers is more cost-effective than acquiring new ones.

This project tackles churn through a two-step approach:

1. **Customer Segmentation:**
   Uses K-means clustering on features like tenure, monthly charges, and total charges to group customers by usage patterns and identify key customer profiles.
2. **Churn Prediction:**
   These segments, combined with other attributes, are used to train machine learning models (Logistic Regression, Gradient Boosting) to predict churn likelihood. SMOTE is applied to handle class imbalance.

By identifying potential churners, telecom companies can implement targeted retention strategies—such as special offers or enhanced support—to reduce churn and strengthen customer loyalty.

---

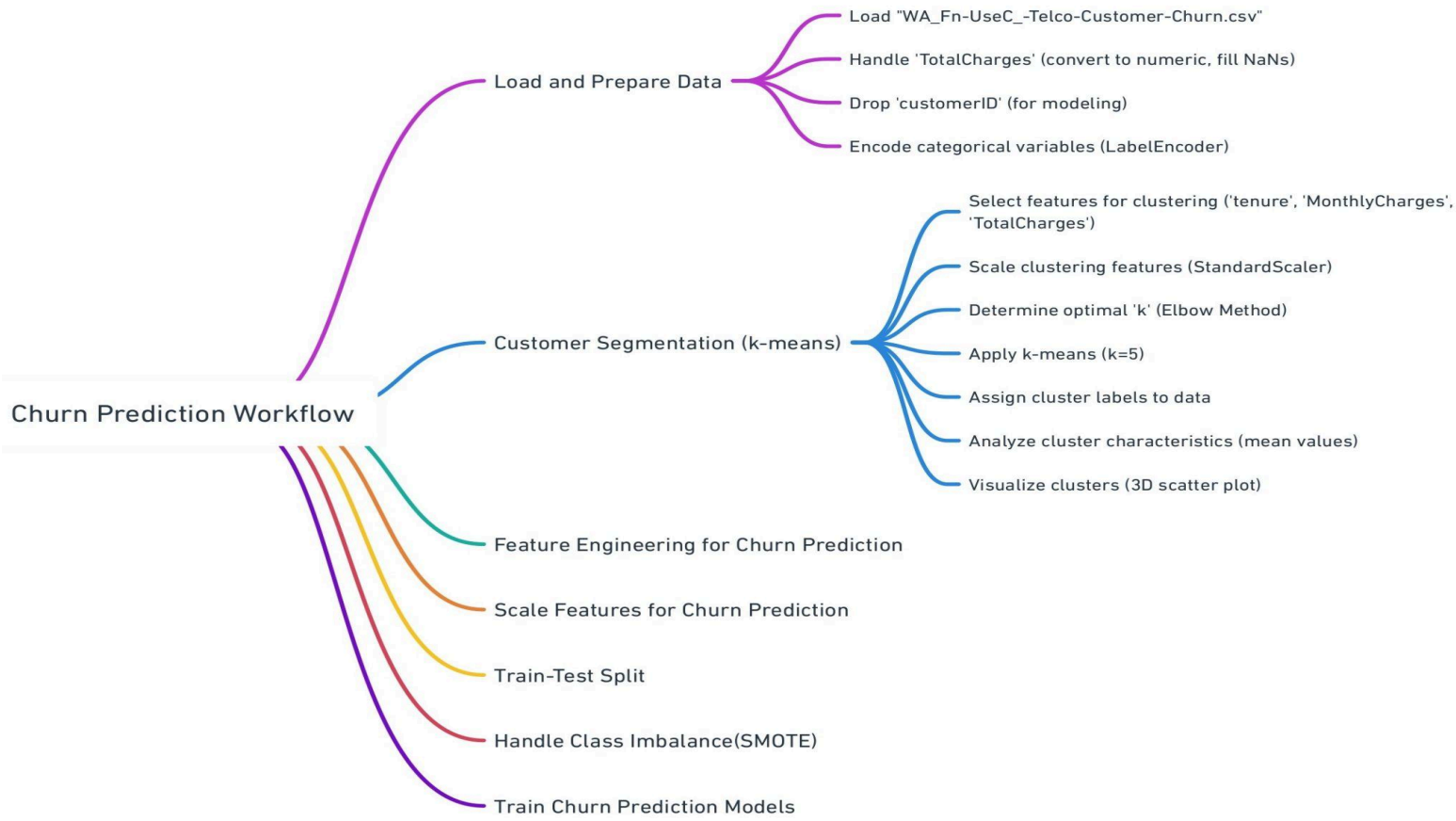
2. Literature Survey (Background study ~ relevant work study)

Customer churn prediction has been widely explored using machine learning techniques such as Logistic Regression, Decision Trees, SVM, Naive Bayes, and ensemble methods like Random Forest and Gradient Boosting.
Customer segmentation, often done using K-means clustering, helps identify behavioral patterns and high-risk groups for targeted retention.
Effective churn prediction also relies on strong feature engineering, using data on demographics, contracts, usage, and service interactions.
Class imbalance—fewer churners than non-churners is a key challenge, typically addressed using techniques like SMOTE to generate synthetic minority samples and improve model fairness.
This project builds upon these established methodologies by:

- Applying k-means clustering for customer segmentation based on key financial and tenure metrics.
- Utilizing the derived cluster information as an engineered feature for churn prediction.
- Comparing the performance of Logistic Regression and Gradient Boosting classifiers.
- Employing SMOTE to handle class imbalance in the training data.
- Evaluating models using standard metrics like precision, recall, F1-score, and AUC-ROC.

3. Methodology & Its Working

The project follows a systematic approach, The key steps are outlined below:
Flowchart:



Detailed Steps:

1. **Data Loading & Preparation**
    a. Loaded WA_Fn-UseC_-Telco-Customer-Churn.csv using pandas.
    b. Converted TotalCharges to numeric; invalid entries filled with 0.
    c. Dropped customerID for modeling; retained a copy for predictions.
    d. Categorical columns encoded using LabelEncoder.
2. **Customer Segmentation (K-Means)**
    a. Selected features: tenure, MonthlyCharges, TotalCharges.
    b. Scaled using StandardScaler.
    c. Used the Elbow Method to determine k=5.
    d. Applied K-Means clustering; assigned each customer to a cluster.
    e. Analyzed cluster characteristics and visualized.
3. **Feature Engineering for Prediction**
    a. Added cluster labels as a feature.
    b. Defined X (features) and y (Churn label).
4. **Feature Scaling**
    a. Applied StandardScaler to X before splitting to avoid data leakage.
5. **Train-Test Split**
    a. Split the dataset into 80% training and 20% testing using train_test_split.
6. **Class Imbalance Handling**
    a. Applied **SMOTE** to the training set to oversample the minority class (churners).
7. **Model Training**
    a. Trained **Logistic Regression** and **Gradient Boosting** on resampled training data.
8. **Model Evaluation**
    a. Evaluated on original test set using:
        i. **Classification Report** (Precision, Recall, F1-Score)
        ii. **AUC-ROC Score**

9. **Feature Importance**
    a. Visualized Gradient Boosting feature importance (feature_importance.png).

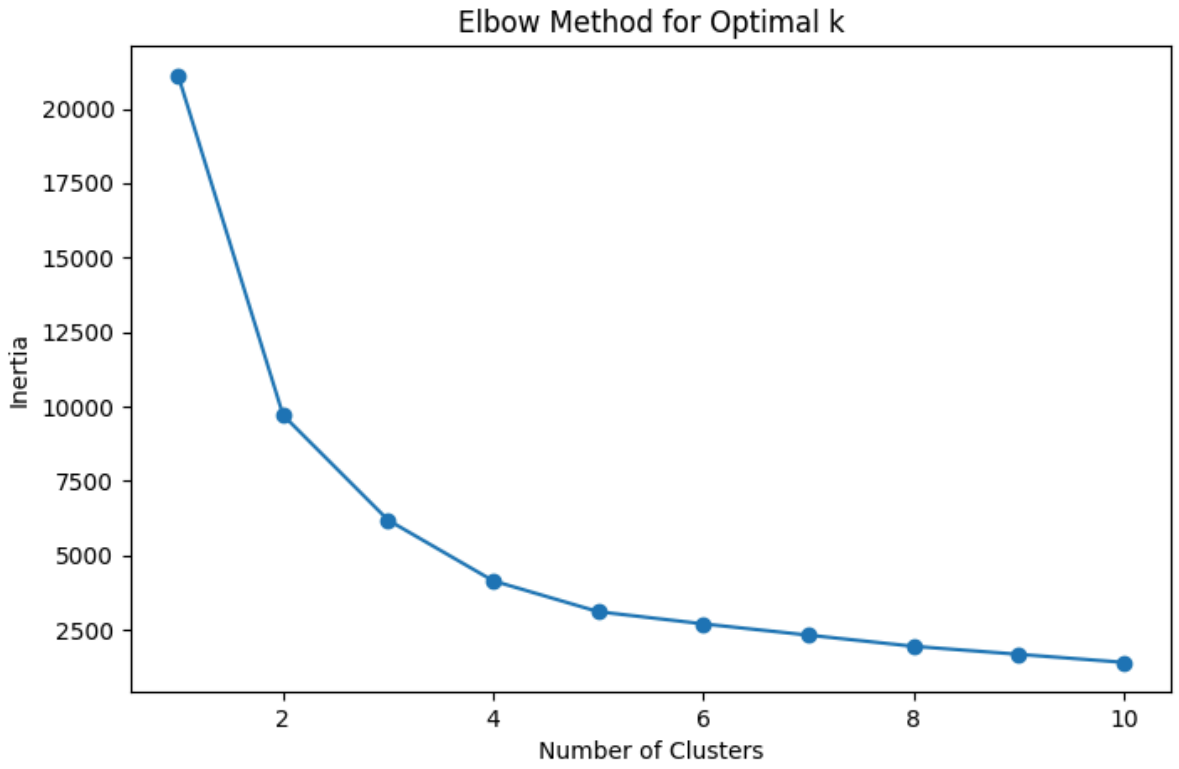
10. **Customer-Specific Prediction**
    a. On input customerID, data is preprocessed and passed to the Gradient Boosting model.
    b. Outputs churn prediction and probability

5. Outputs

The project generates several outputs, including analytical tables and visualizations.

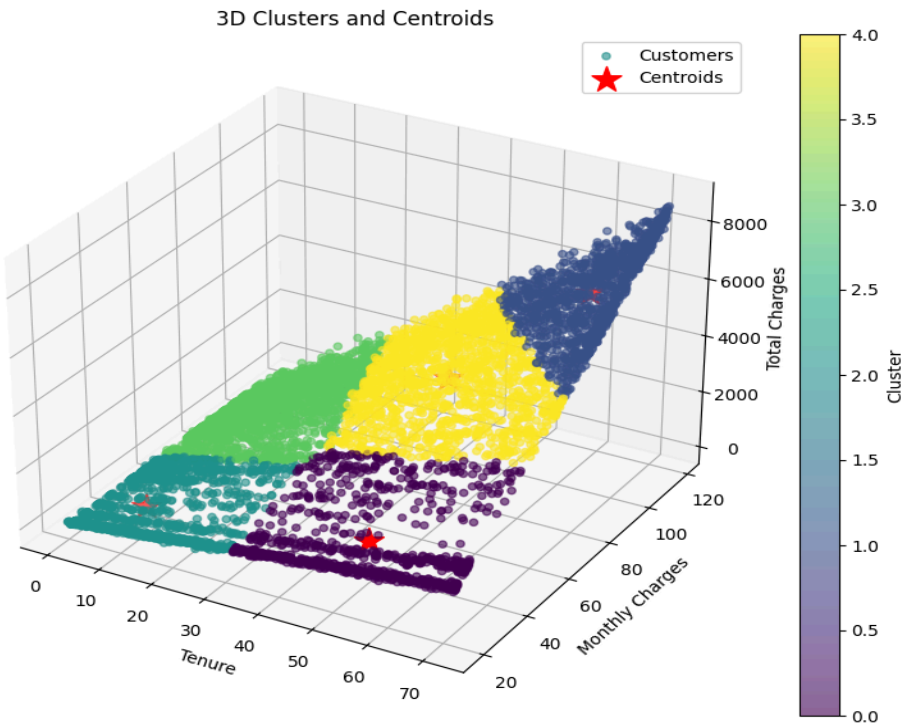
Figures/Diagrams:

1. **Elbow Method for Optimal k (elbow_curve.png):**



    a. **Description:** This line graph plots the number of clusters (k) on the x-axis against the inertia (sum of squared distances of samples to their closest cluster center) on the y-axis.
    b. **Explanation:** The "elbow" point, where the rate of decrease in inertia sharply slows, suggests an optimal value for k. In this project, k=5 was chosen based on this method.

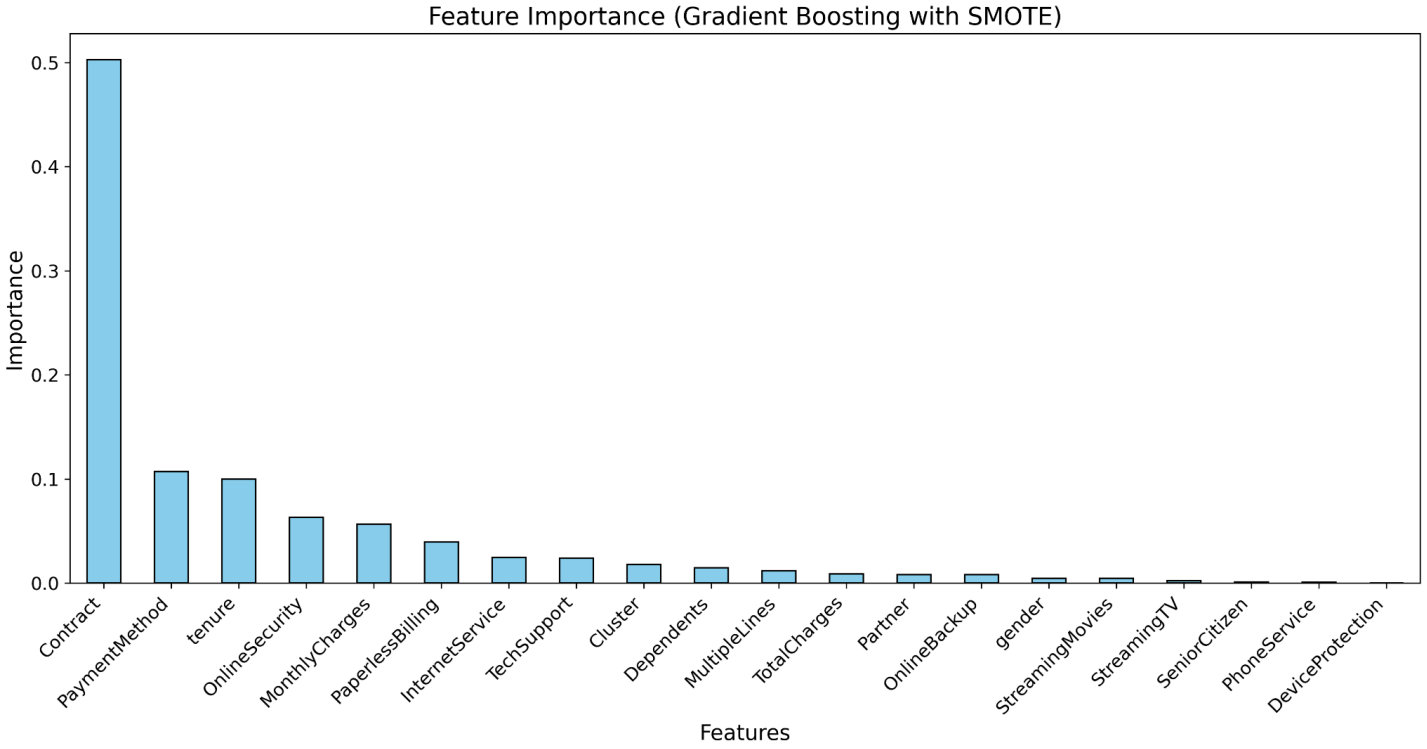
2. **3D Clusters and Centroids (cluster_3d_plot.png):**



a. **Description:** This is a 3D scatter plot visualizing the customer segments. Data points are colored by their assigned cluster. The features tenure, MonthlyCharges, and TotalCharges form the axes. Red markers indicate the centroids of each cluster.

b. **Explanation:** This visualization helps in understanding the separation and characteristics of the identified customer segments in a multi-dimensional space.

3. **Feature Importance (Gradient Boosting with SMOTE) (feature_importance.png):**



a. **Description:** This bar chart displays the relative importance of each feature used in the Gradient Boosting model for predicting churn. Features are listed on the x-axis, and their importance scores are on the y-axis.

b. **Explanation:** It helps identify the key drivers of churn. For example, features like 'Cluster', 'tenure', 'MonthlyCharges', 'Contract' type often show high importance.

Output Data (Tables):

Table 1: Cluster Analysis

a. **Description:** This table shows the mean values of `tenure`, `MonthlyCharges`, and `TotalCharges` for each of the 5 identified customer clusters.

b. **Data:**

Cluster	tenure (mean)	MonthlyCharges (mean)	TotalCharges (mean)
0	53.19	28.39	1485.14
1	65.22	99.02	6447.07
2	9.87	31.61	295.61
3	11.81	79.52	931.63
4	46.42	80.77	3646.25

c. **Explanation:** This table helps characterize each customer segment. For instance, Cluster 1 might represent long-tenured, high-spending customers, while Cluster 2 might represent new, low-spending customers.

Table 2: Logistic Regression Performance (with SMOTE)

d. **Description:** Classification report and AUC-ROC score for the Logistic Regression model.

e. **Data:**

```
                precision    recall  f1-score   support
0 (No Churn)         0.92      0.74      0.82      1036
1 (Churn)            0.53      0.83      0.65       373
accuracy                               0.76      1409
macro avg            0.73      0.78      0.73      1409
weighted avg         0.82      0.76      0.77      1409


AUC-ROC: 0.8610
```

f. **Explanation:** The Logistic Regression model achieved an accuracy of 76% and an AUC-ROC of 0.861. It shows good recall for the churn class (0.83), meaning it identifies 83% of actual churners, but precision is lower (0.53), indicating a fair number of false positives for churn.

Table 3: Gradient Boosting Performance (with SMOTE)

g. **Description:** Classification report and AUC-ROC score for the Gradient Boosting model.

h. **Data:**

```
                precision    recall  f1-score   support
0 (No Churn)         0.90      0.81      0.85      1036
1 (Churn)            0.59      0.75      0.66       373
accuracy                               0.79      1409
macro avg            0.74      0.78      0.76      1409
weighted avg         0.82      0.79      0.80      1409


AUC-ROC: 0.8574
```

i. **Explanation:** The Gradient Boosting model achieved an accuracy of 79% and an AUC-ROC of 0.857. It has a slightly higher precision for the churn class (0.59) and comparable recall (0.75) compared to

Logistic Regression. The overall F1-score for churn is slightly better.

Example Prediction Output for a Specific Customer:

j. **Description:** When a customerID is provided, the model predicts churn and the probability.

k. **Example Data (format):**

```
CustomerID: [Entered CustomerID]
Predicted Churn: Yes/No
Probability of Churn: [XX.XX]%
```

l. **Explanation:** This demonstrates the model's application in predicting churn for individual customers.

---

6. Limitations of this project

1. **Dataset Specificity**
   The models are trained on a specific telecom dataset and may not generalize to other industries without retraining.
2. **Limited Feature Scope**
   Features are focused on billing and tenure; adding behavioral and demographic data could improve performance.
3. **K-Means Algorithm Sensitivity**
   K-means is sensitive to centroid initialization and the choice of k; results may vary without robust validation.
4. **SMOTE Assumptions and Risks**
   SMOTE may generate unrealistic samples and risk overfitting if not applied carefully.
5. **Static Nature of the Model**
   Models are static and require regular updates to adapt to changing customer behaviors.
6. **Model Interpretability Trade-offs**
   Gradient Boosting is less interpretable than Logistic Regression, which may hinder understanding of predictions.
7. **Lack of Hyperparameter Tuning**
   Default parameters are used; tuning could lead to better model accuracy and generalization.

---

7. Challenges & Future Work

Challenges Encountered:

- **Class Imbalance**
  The dataset had a disproportionate number of non-churners; SMOTE was used to balance the classes.
- **Optimal Cluster Selection**
  Selecting the right number of clusters (k) using the Elbow method was subjective and required careful interpretation.
- **Feature Engineering**
  Choosing relevant features and encoding categorical variables effectively was crucial for model performance.
- **Model Interpretability vs. Performance**
  Trade-off between interpretable models and more accurate but complex ones

Future Work:

- **Extensive Hyperparameter Optimization**
  Use techniques like Grid Search or Randomized Search with cross-validation to improve model performance.
- **Ensemble Methods**
  Explore stacking or blending multiple models to enhance predictive accuracy.
- **Time-Series Analysis**
  Incorporate temporal trends if timestamped data becomes available to capture churn dynamics over time.
- **Deployment**
  Build a deployable pipeline to integrate the model into CRM systems for real-time churn risk scoring.
- **Cost-Sensitive Learning**
  Factor in the business cost of misclassifications to make more informed retention decisions.
- **Dynamic Feature Engineering**
  Develop features that reflect behavioral shifts over time, like changes in usage or billing patterns.

---

8. Conclusion

This project demonstrated an end-to-end approach to customer churn prediction in the telecommunications sector by integrating customer segmentation with supervised machine learning.

- **Data Preprocessing**
  Cleaned and prepared the Telco customer churn dataset for analysis and modeling.
- **Customer Segmentation**
  Applied K-means to segment customers into five distinct groups based on tenure and spending, providing actionable insights.
- **Churn Prediction Models**
  Built and evaluated Logistic Regression and Gradient Boosting models; addressed class imbalance using SMOTE.
- **Model Performance**
  Gradient Boosting achieved 79% accuracy and 0.857 AUC-ROC; Logistic Regression reached 76% accuracy and 0.861 AUC-ROC, both showing strong recall for churners.
- **Feature Engineering**
  Adding cluster labels as a feature improved model performance, with notable importance in the Gradient Boosting model.

References

- Pandas Dev Team. (2020). *pandas*. Zenodo. https://doi.org/10.5281/zenodo.3509134
- Pedregosa et al. (2011). Scikit-learn. *JMLR*, 12, 2825–2830.
- Lemaître et al. (2017). Imbalanced-learn. *JMLR*, 18(17), 1–5.
- Hunter, J. D. (2007). Matplotlib. *CiSE*, 9(3), 90–95.
- Harris et al. (2020). NumPy. *Nature*, 585, 357–362.
- Ames & Ponder. (2020). *Applied Data Science for Telecom Churn*. O'Reilly.
- Verbeke et al. (2012). Churn prediction. *EJOR*, 218(1), 211–229.
- Chawla et al. (2002). SMOTE. *JAIR*, 16, 321–357.
- MacQueen, J. (1967). K-means. *Proc. Berkeley Symp.*, 1(14), 281–297.
- Friedman, J. H. (2001). Gradient boosting. *Ann. Stat.*, 29(5), 1189–1232.
- IBM/Kaggle Dataset: *Telco Customer Churn.*
- Blogs: *Towards Data Science*, *KDnuggets*, etc.