CUSTOMER CHURN PREDICTION

Prepared by -

Prerna Jasti

**Project Overview**

This report provides comprehensive documentation for a customer churn prediction project. The primary objective of this project is to empower the business to identify customers at high risk of discontinuing their services. By proactively predicting churn, the organization can address potential dissatisfaction, tailor interventions to retain customers, and enhance overall customer satisfaction. The insights gained from this initiative are expected to contribute significantly to long-term business sustainability by reducing revenue losses and fostering stronger customer relationships.

INDEX

|  |  |  |
| --- | --- | --- |
| Sr no. | TITLE | Page no. |
| 1 | Problem Statement | 4 |
| 2 | Project Goals | 5 |
| 3 | Research methodology | 6 |
| 4 | Data pre processing | 8 |
| 5 | Exploratory Data Analysis | 10 |
| 6 | Model development | 12 |
| 7 | Results | 17 |
| 8 | Conclusion | 19 |
| 9 | Bibliography | 20 |

**Problem Statement**

Customer churn is a significant challenge faced by businesses across industries. Losing customers not only impacts revenue but also increases the cost of acquiring new customers, which is often higher than retaining existing ones. Without a systematic approach to understanding customer behavior, businesses struggle to identify the early warning signs of churn. This inability to anticipate churn limits their ability to deploy effective strategies, leading to decreased customer loyalty and brand value. By accurately predicting churn and understanding the underlying factors, businesses can take preemptive actions to address customer dissatisfaction, strengthen engagement, and reduce the financial impact of churn.

**Project Goals**

1. **Develop a Predictive Model**:

The primary goal of this project is to build a predictive model that accurately identifies customers at risk of churning. The model will leverage machine learning techniques to analyze a range of customer data, including transaction history, service usage, demographic details, and behavioral patterns. By training the model on historical data, it will be able to flag high-risk customers in real time, enabling the business to focus its retention efforts more effectively.

1. **Analyze Customer Data**:

Conduct a detailed examination of customer-related data to uncover trends and factors that influence churn. This includes analyzing customer interaction frequency, feedback, service usage patterns, and demographic characteristics. By understanding these patterns, the project aims to identify key drivers of churn, such as poor service experiences, inadequate engagement, or unmet customer expectations. These insights will form the basis for designing targeted interventions and improving customer satisfaction.

**Research Methodology**

**(Data Collection Method)**

Research methodology is a way to systematically solve the research problem. It may be understood as a science of studying how research is done systematically. In that various steps, those are generally adopted by a researcher in studying his problem along with the logic behind them.

Data collection is an important step in any project and the success of any project will largely depend upon how accurate you will be able to collect and how much time, money, and effort will be required to collect that necessary data, this is also an important step.

Data collection plays an important role in research work. Without proper data available for analysis, you cannot do the research work accurately.

PYTHON LIBRARIES USED

* **Data preprocessing** - Numpy , pandas , matplotlib
* **Model development** - Sklearn
* **Production environment** – Streamlit

DATA SOURCE :

<https://www.kaggle.com/datasets/abdullah0a/telecom-customer-churn-insights-for-analysis/data>

DATA DESCRIPTION:

* 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.
* Monthly Charges: The monthly fee charged to the customer.
* Contract Type: Type of contract the customer is on (Month-to-Month, One-Year, Two-Year).
* Internet Service: Type of internet service subscribed to (DSL, Fiber Optic, None).
* Tech Support: Whether the customer has tech support (Yes or No).
* Total Charges: Total amount charged to the customer (calculated as Monthly Charges \* Tenure).
* Churn: Target variable indicating whether the customer has churned (Yes or No).

**Data Preprocessing**

1. **Data Cleaning: Handling Missing Values, Outliers, and Duplicates**  
   Data cleaning ensures the dataset is accurate, consistent, and free of issues that could affect model performance.

Key steps included:

1. **Handling Missing Values**: Missing data can arise from incomplete data entry or errors. In this project we have used
   * Filling missing categorical values with the placeholder such as

“ “.

1. **Dealing with Outliers**: Outliers, which are extreme values, can distort model training. These can be handled by:
   * Removing outliers using statistical methods like the interquartile range (IQR).
   * Transforming the data using log or square root transformations to reduce the impact of outliers.
2. **Checking for Duplicates**: Duplicate entries can skew results and introduce redundancy. This involves:
   * Identifying duplicate rows or records based on unique identifiers.
   * Removing duplicates to ensure the dataset is clean and unbiased.
3. **Feature Selection: Identifying Most Important Features**  
   Feature selection involves selecting the most relevant features to improve model performance and reduce complexity. Technique used
   * **Filter Methods**: Using statistical measures like correlation.
4. **Encoding Categorical Variables: Converting Categorical Variables into Numerical Format**Since machine learning models work with numerical data, categorical variables must be encoded. Techniques include:

* **One-Hot Encoding**: Creating binary columns for each category (e.g., "Gender: Male"=0 and "Gender: Female"=1).
* **Target Encoding**: Replacing categories with their mean value of the target variable (e.g., churn probability for each category). Choosing the right encoding method depends on the type of model and dataset size.

1. **Normalization: Scaling Features for Better Model Performance**  
   Normalization ensures features are on a similar scale, which is critical for algorithms that rely on distance metrics or gradient-based optimization. We include:

* **Standardization**: Centeralizes data by subtracting the mean and dividing by the standard deviation (mean = 0, std = 1).

**Exploratory Data Analysis (EDA)**

1. **Descriptive Statistics: Summary Statistics for Numerical and Categorical Variables**  
   Descriptive statistics provide an overview of the dataset. Key measures include:
   * **Numerical Variables**: Mean, median, variance, standard deviation, minimum, and maximum values to understand the central tendency and spread of data.
   * **Categorical Variables**: Frequency counts and proportions for each category to identify distribution patterns. Summaries provide insight into data structure and potential anomalies.
2. **Visualization: Charts and Graphs to Visualize Customer Behavior and Churn Rates**  
   Visual tools make it easier to spot trends and patterns.
   * **Histograms**: To visualize distributions of features like customer tenure or monthly charges.
   * **Bar Charts**: To compare category-level distributions (e.g., churn rates across regions or genders).
   * **Pie Charts**: To represent the proportion of different categories visually, such as the percentage of churned vs. retained customers or customer types. Pie charts are effective for displaying relative proportions in an intuitive format.
3. **Correlation Analysis: Identifying Relationships Among Features and the Target Variable**  
   Correlation analysis helps identify how features relate to each other and the target variable (churn). Techniques include:
   * **Correlation Matrices**: Visualizing relationships among numerical variables using metrics like Pearson or Spearman coefficients.
   * **Feature-Target Analysis**: Measuring how strongly each feature influences the target variable.
   * Addressing multicollinearity (highly correlated independent features) to avoid redundancy and improve model interpretability.

**Model Development**

**Model Selection**

* **Logistic Regression**

Logistic regression is a simple and widely used algorithm for binary classification tasks, though it can be extended to multiclass problems. It predicts the probability of a class using the logistic (sigmoid) function, making it a great choice for datasets where the relationship between input features and the output class is linear. Logistic Regression is efficient, interpretable, and works particularly well with linearly separable datasets.

* **K-Nearest Neighbors (k-NN) classifier**

The k-Nearest Neighbors (k-NN) classifier is a non-parametric and lazy learning algorithm that assigns a class to a data point based on the majority class of its k closest neighbors in the feature space. It relies on distance metrics, such as Euclidean or Manhattan distance, to identify neighbors. While k-NN is easy to understand and implement, it can become computationally expensive as the dataset grows, since it needs to compute distances for every prediction.

* **Support Vector Classifier (SVC)**

Support Vector Classifier (SVC) is a supervised algorithm that aims to find the optimal hyperplane that separates data points of different classes in a high-dimensional space. Using kernel functions, such as linear, polynomial, or radial basis function (RBF), SVC can handle non-linear relationships in the data effectively. It performs well for small to medium-sized datasets and is especially useful for high-dimensional spaces where other algorithms might struggle.

* **Decision Trees**

Decision trees are tree-structured models that classify data by repeatedly splitting it based on feature values. Each node in the tree represents a decision rule, and the leaves correspond to the final class predictions. Decision Trees are intuitive and easy to visualize, making them interpretable models. However, they are prone to overfitting, especially when the tree grows too complex, which can be mitigated with techniques like pruning.

* **Random Forest Classifier**

Random Forest classifier is an ensemble learning method that combines multiple decision trees to improve prediction accuracy and reduce overfitting. It uses bootstrap aggregation (bagging) to create diverse trees and averages their predictions (or uses majority voting for classification). Random Forest works well for both classification and regression tasks and handles large datasets effectively while being robust to noise.

**Model training**

1. **Splitting the Dataset**:  
   When building a machine learning model, the dataset is divided into two parts:
   * **Training Set**: Used to train the model so it learns patterns and relationships in the data.
   * **Testing Set**: Used to evaluate the model’s performance on unseen data to ensure it generalizes well.  
     This split helps to prevent overfitting and gives a realistic estimate of how the model will perform in the real world.
2. **Training the Model**:  
   The model is trained by fitting it to the training data, where it learns to map inputs (features) to outputs (target). Different algorithms (e.g., decision trees, support vector machines, or logistic regression) can be applied based on the problem type (classification or regression).
3. **Cross-Validation**:  
   Cross-validation is a technique to further ensure the model's reliability. Instead of relying on a single train-test split, the dataset is divided into multiple folds. The model is trained and tested on different folds, and the results are averaged. This provides a more robust estimate of model performance and reduces the chance of overfitting or underfitting.
4. **Grid Search for Hyperparameter Tuning**:  
   Models often have hyperparameters (settings) that can influence their performance. Grid search is an automated method to find the best combination of hyperparameters by testing multiple configurations systematically. It evaluates each combination using cross-validation and selects the one with the best performance.

**Model Evaluation**  
The model is evaluated on the testing set using appropriate metrics .This helps determine how well the model generalizes to new data.

* **Accuracy**:  
  Measures the overall correctness of the model by calculating the ratio of correctly predicted instances to the total instances. It’s useful when the dataset is balanced.
* **Precision**:  
  Indicates the proportion of correctly predicted positive cases out of all predicted positives. High precision means fewer false positives, making it important when false alarms are costly.
* **Recall**:  
  Measures the proportion of actual positive cases that the model correctly identifies. High recall is critical when missing positive cases has severe consequences**.**
* **F1 Score**:  
  The harmonic mean of precision and recall, balancing the two metrics. It is useful when there’s an uneven class distribution or when both false positives and false negatives need to be minimized.

RESULTS

1. **Logistic regression**

Accuracy score : 91.5 %

Precision score : 96.15384615384616 %

Recall score : 94.5945945945946 %

F1 Score : 95.36784741144415 %

1. **KNeighbour Classifier**

Accuracy score : 91.5 %

Precision score : : 97.19101123595506 %

Recall score : : 93.51351351351352 %

F1 Score : 95.31680440771349 %

1. **Support Vector Classifier**

Accuracy score : 89.0 %

Precision score : 94.53551912568307 %

Recall score : 93.51351351351352 %

F1 Score : 94.02173913043478 %

1. **Desion Tree Classifier**

Accuracy score : 88.5 %

Precision score : 97.09302325581395 %

Recall score : 90.27027027027027 %

F1 Score : 93.55742296918767 %

1. **Random Forest Classifier**

Accuracy score : 92.0 %

Precision score : 96.68508287292818 %

Recall score : 94.5945945945946 %

F1 Score : 95.62841530054644 %

CONCLUSION

**Model Evaluation and Selection**

Among the evaluated machine learning models, the **Random Forest Classifier (RFC)** demonstrated the highest **Recall** score, making it the most suitable choice for the customer churn analysis. Recall is a critical metric in this context, as it ensures the model effectively identifies customers likely to churn, minimizing the risk of missing true churners. By prioritizing recall, the RFC enables proactive retention strategies, which are crucial for mitigating revenue loss. Other metrics, such as precision, accuracy, and the F1 score, were also considered, but recall was prioritized due to its relevance to the business objective.

BIBLIOGRAPHY

<https://www.kaggle.com/>

<https://chatgpt.com/>

<https://scribehow.com/>

<https://scholar.google.com/>

<https://www.youtube.com/@onurdatascience>