📊 Churn Analysis & Prediction

A complete end-to-end Customer Churn Analysis & Prediction system leveraging SQL, Power BI, and Python (Machine Learning) to understand churn drivers, predict churn probability, and deliver actionable business insights.

🚀 Features

🔹 Full ETL pipeline implemented in SQL to extract, clean, and prepare the churn dataset

🔹 Data preprocessing, feature engineering, and machine-learning modelling in Python

🔹 Comparison of multiple models (e.g., logistic regression, random forest, XGBoost)

🔹 Evaluation using accuracy, F1-score, ROC-AUC, confusion matrix

🔹 Visual analytics / dashboard (e.g., via Power BI) to present churn patterns and revenue loss

🔹 Professional documentation and a workflow summary report

📁 Project Structure

```
Churn_Prediction/
├── data/
│ ├── raw/Customer_Data.csv ← raw data file
│ └── clean/Prediction_Data.xlsx ← cleaned/transformed data ready for modelling
├── sql/
│ ├── Upload_file.sql ← initial data load into staging
│ ├── Check_Null.sql ← data quality checks (NULLs, missingness)
│ ├── Remove_Nulls.sql ← data cleaning / standardization
│ └── View_Power_BI.sql ← final views used in dashboard layer
├── power IB/
│ └── Churn Analysis.pbix ← Power IB dashboard
├── notebook/
│ └── Churn_Prediction.ipynb ← Jupyter notebook with EDA, modelling
├── reports/
│ ├── Unified_Presentation.pptx ← full presentation of results
│ └── README.pdf ← PDF version of documentation
├── requirements.txt ← list of Python dependencies
└── README.md ← this file
```

🛠 SQL Workflow (ETL Pipeline)

1. Upload_file.sql – Initial Exploration Loads data into staging table Generates demographic, contract, and revenue insights
2. Check_Null.sql – Data Quality Audit Inspects NULL counts for all fields
3. Remove_Nulls.sql – Cleaning & Standardization Replaces missing values with defaults Loads final dataset into prod_Churn
4. View_Power_BI.sql – BI Layer Creates:
   - vw_ChurnData
   - vw_JoinData

Used directly in Power BI.

📈 Power BI Dashboard Insights

- Churn vs Retained customer overview
- High-risk customer segments
- State-wise churn heatmaps
- Revenue loss breakdown
- Contract-wise churn patterns

🤖 Machine Learning Workflow

✔ Data Preprocessing

- One-hot encoding
- Scaling
- Handling class imbalance

✔ EDA

- Churn distribution
- Correlation matrix
- Service usage patterns

✔ Models Used

- Logistic Regression
- Random Forest
- XGBoost

✔ Evaluation Metrics

- Accuracy
- F1 Score
- ROC–AUC
- Confusion Matrix

Installation & Setup

Follow these steps to set up the project environment and run the analyses:

1. Clone the repository git clone https://github.com/JaskarJeyabalan/Churn_Prediction.git
2. cd Churn_Prediction
3. Install Python dependencies pip install -r requirements.txt
4. Run SQL scripts Execute scripts in the following order:
5. Upload_file.sql
6. Check_Null.sql
7. Remove_Nulls.sql
8. View_Power_BI.sql
9. Open Power BI Dashboard power IB/Churn Analysis.pbix
10. Open ML Notebook notebook/Churn Prediction.ipynb
11. Reports reports/Presentation.pptx reports/README.pdf

📫 Author

Jaskar Jeyabalan S

jaskarjeyabalan@gmail.com

Churn Analytics | SQL | Power BI | Machine Learning

☆ Support the Project If you found this project helpful, feel free to ☆ star the repository on GitHub!