**📊 Churn Prediction Web Application – Comprehensive Documentation**

**🧩 Problem Statement**

Customer churn is a critical metric for telecom companies as it directly impacts profitability. The goal of this project is to **predict customer churn** using historical customer information and service usage patterns. Accurate churn prediction allows companies to proactively engage with high-risk customers, thereby improving retention.

**📘 Data Overview**

The dataset includes customer-level information such as:

* **Demographics**: Gender, SeniorCitizen, Partner, Dependents
* **Service Info**: PhoneService, InternetService, StreamingTV, etc.
* **Account Info**: Contract type, MonthlyCharges, Tenure, PaymentMethod
* **Target Variable**: Churn (Yes/No)

**🛠️ Data Preprocessing Pipeline**

To ensure clean and reliable input for machine learning models, a robust preprocessing pipeline was developed:

**✅ Steps:**

* **Type Casting**: Convert SeniorCitizen to categorical, handle missing TotalCharges with coercion and drop NaNs.
* **Encoding**: One-hot encoding for 16 categorical variables.
* **Scaling**: RobustScaler used on numerical columns (tenure, MonthlyCharges, TotalCharges) to manage outliers.
* **Consistency**: Saved feature columns and scaler are reused in prediction to maintain parity between training and inference.

**🧠 Model Development**

Multiple machine learning algorithms were evaluated using cross-validation and hyperparameter tuning:

**✅ Models Built:**

* Logistic Regression
* Decision Tree
* Random Forest
* Extra Trees
* K-Nearest Neighbors
* Gradient Boosting
* Support Vector Machines
* Gaussian Naive Bayes

**✅ Evaluation Metrics:**

* Accuracy
* AUC-ROC
* Confusion Matrix

Each model was trained on the same processed data and saved as .pkl files for later inference.

**📈 Evaluation Highlights**

|  |  |  |
| --- | --- | --- |
| Model | Accuracy | AUC-ROC |
| Logistic Regression | 0.8032 | 0.8437 |
| Random Forest | 0.8214 | 0.8596 |
| Gradient Boosting | 0.8301 | 0.8723 |
| Extra Trees | 0.8289 | 0.8692 |
| SVM (Linear) | 0.7897 | 0.8314 |

✅ **Gradient Boosting** emerged as the best performer with the highest AUC.

**🚀 Deployment: Streamlit Web App**

A fully functional and interactive **Streamlit-based web application** was developed for end-to-end prediction.

**🔍 Key Features:**

* **📁 File Upload**: Accepts raw customer data in CSV format.
* **📄 Preview**: Displays uploaded data, schema, nulls, and descriptive stats.
* **🧠 Model Predictions**: Automatically applies **all trained models** to predict churn.
* **📊 Visualizations**: AUC-ROC curves plotted dynamically for each model.
* **📥 Download**: Per-model prediction CSV export available.
* **🏆 Best Model Selection**: Intelligent capture of the best-performing model.

**🖼️ UI Innovations:**

* Modern **fade-in animations** for headings
* Tab-based architecture (Summary, All Predictions, model-specific tabs)
* Clean Plotly visuals for ROC curves
* DataFrames are styled and scrollable

**💡 Innovations & Uniqueness**

|  |  |
| --- | --- |
| Innovation | Description |
| Multi-model comparison | Evaluates multiple models side-by-side in real-time within the web UI. |
| Training-inference consistency | Ensures identical preprocessing at both stages using saved scaler and feature columns. |
| Auto-handling categorical levels | Robust to category mismatches or missing columns by reindex with fill. |
| ROC Curve per model | Dynamically generates and displays ROC curves in the app using Plotly. |
| Export-ready predictions | Results are downloadable per model to compare performance offline. |

**📁 Project Structure**

📂 churn\_app/

├── app.py (Streamlit frontend)

├── models/

│ ├── scaler.pkl

│ ├── train\_dummy\_columns.pkl

│ ├── model\_rf.pkl

│ ├── model\_gb.pkl

│ └── ...

├── data/

│ └── test.csv

**🔐 Robustness & Security**

* Error handling for missing columns, corrupt models, invalid CSV files
* Download buttons wrapped with file-safe naming conventions
* Modularized architecture for easy extension (e.g., adding more models)

**🧪 Future Enhancements**

* ✅ **SHAP/LIME** explainability integration
* ✅ **Model selection toggle** for production usage
* ✅ **Drift detection** to alert if incoming data diverges from training distribution
* ✅ **Model retraining trigger** based on performance

**🏁 Conclusion**

This churn prediction app delivers a powerful yet user-friendly interface to explore customer churn through multiple lenses — with real-time predictions, visual analytics, and flexibility to handle different models. It’s designed for **business stakeholders, data scientists, and analysts** alike to make informed retention strategies.

**📬 Submission Highlights (if part of competition):**

| **Section** | **Details** |
| --- | --- |
| **Team Name** | Vijaykumar Radhakisan Kalaskar |
| **Project Name** | Customer Churn Predictor |
| **Link to Web App** | [e.g., Streamlit Cloud URL] |
| **GitHub Repo** | [e.g., <https://github.com/yourrepo>] |
| **Demo Video** | [Optional link] |