Customer Churn Prediction Report

Problem: What We Solved

- Churn is when a customer leaves a company and stops using its services.
- For telecom companies, churn means **loss of revenue and higher costs** to acquire new customers.
- We aimed to **build a prediction system** that identifies which customers are likely to churn, based on their usage patterns and account details.
- With this system, the business can **act early to retain customers** who are at risk of leaving.

Approach: How We Did It

We used a real-world dataset of telecom customers and followed these main steps:

- **Explored the data** to understand patterns in churn.
- **Cleaned and prepared** the data: fixed missing values, converted text into numbers, and scaled important values.
- Built three predictive models:
  - **Logistic Regression** (good for interpretability)
  - **Decision Tree** (simple but effective)
  - **Random Forest** (a powerful ensemble model)
- The dataset had many more non-churners than churners, so we tried two techniques to fix the imbalance:
  - **Class Weights**: told the model to focus more on the churn class.

- SMOTE: added synthetic churn cases to balance the training data.

- We measured model performance using accuracy, recall, and F1-score.

Findings: What We Learned

Across all models, the following features consistently appeared as top drivers of churn:

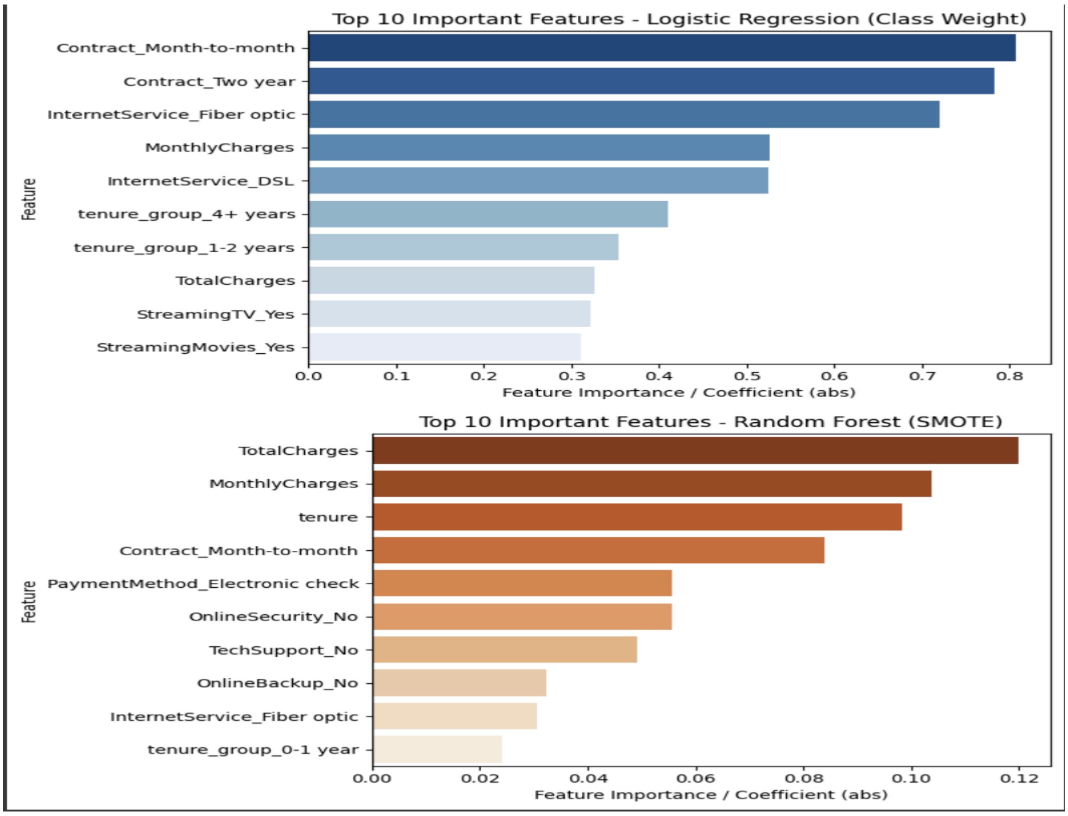
- **Contract type**: Customers with "Month-to-month" contracts were significantly more likely to churn compared to those with longer-term contracts.
- **Tenure**: Shorter tenure (i.e., newer customers) had a much higher churn rate.
- **Internet service type**: Customers with fiber optic connections showed higher churn rates, possibly due to higher costs or service dissatisfaction.
- **Monthly charges**: Higher monthly bills correlated with a higher likelihood of churn.
- **Payment method**: Those using electronic check payments had a greater tendency to leave, potentially indicating lower customer satisfaction or financial flexibility.

These features align with business intuition, indicating that dissatisfaction often stems from billing terms and perceived service value.

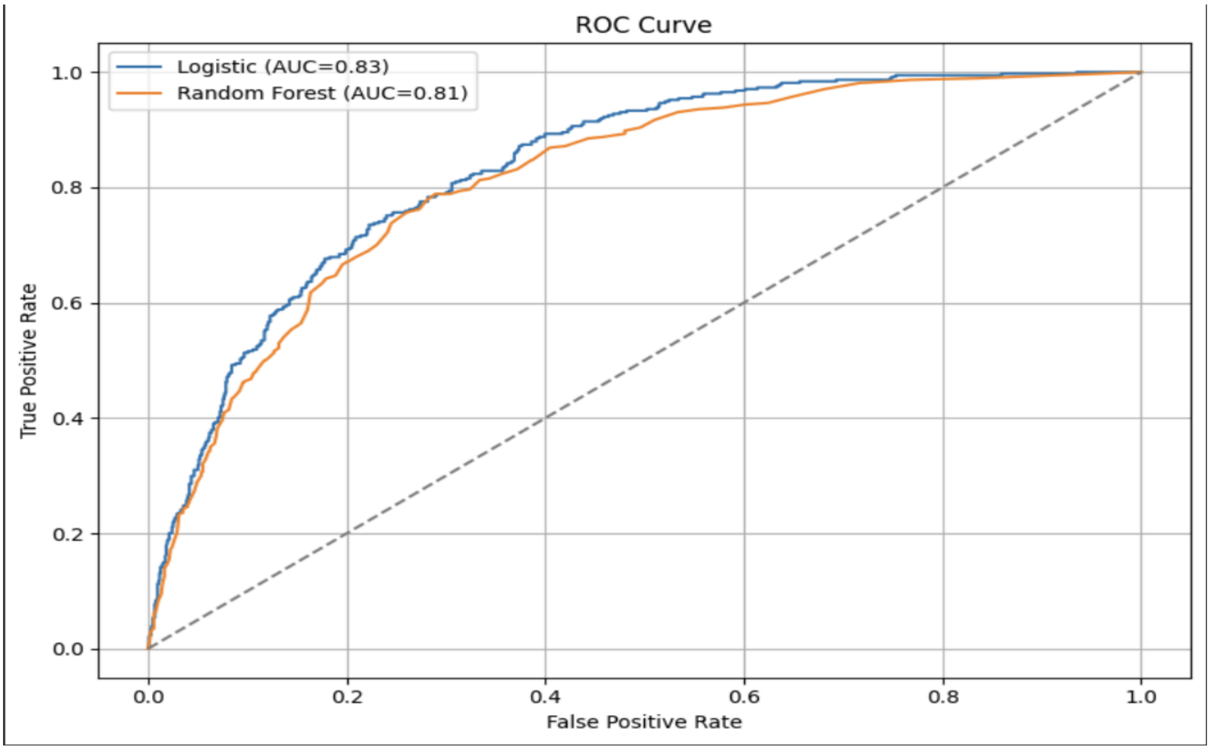
Model Performance Summary

- **Logistic Regression (with class weights)** achieved the highest **recall** for churn (0.79), meaning it was most successful at identifying customers who actually churned.
- **Random Forest (with SMOTE)** produced the best overall **balance between precision and recall**, making it the most reliable model when both false positives and false negatives matter.
- **Decision Trees**, while interpretable, performed slightly worse than the other models in terms of both recall and F1-score.

Top Features That Predict Churn



Model Comparison Table or ROC Curve



Model Explainability Using LIME

To enhance model transparency and build trust in individual predictions, we applied **LIME (Local Interpretable Model-Agnostic Explanations)**. LIME provides an interpretable explanation for how the model arrives at a prediction by approximating the behavior of the complex model locally using a simpler, interpretable model (e.g., a linear model).

We applied LIME to a representative test instance and obtained the following insights:

- **Positive Contributors (reduce churn risk)**:
    - Customers with longer **tenure** (e.g., `tenure > 56`) are less likely to churn.
    - Customers on **long-term contracts** (especially `Contract_Two year`) show higher retention.
    - High **total charges** correlated with loyalty in some cases, indicating continued service usage.
- **Negative Contributors (increase churn risk)**:
    - Customers on **month-to-month contracts** are more prone to churn.
    - Users of **fiber optic internet** services tend to show a slightly higher churn probability—possibly due to pricing or service issues.
    - Lower tenure or intermediate values for **monthly charges** slightly increased churn risk.

This interpretability allows customer-facing teams to **understand and act on churn signals**. For example, customers with high risk and short tenure might benefit from **targeted retention offers or upgraded service plans**.

Web Application Deployment

To make our machine learning solution accessible to business stakeholders and customer service teams, we developed an **interactive web application using Streamlit**. The app allows non-technical users to input customer attributes and receive real-time churn predictions with visual insights.

Key Features:

- **Customer Input Form**: Intuitive sidebar interface for entering customer demographics, account info, and service details.

- **Live Churn Prediction**: Returns a prediction (Churn / Not Churn) along with a confidence score.

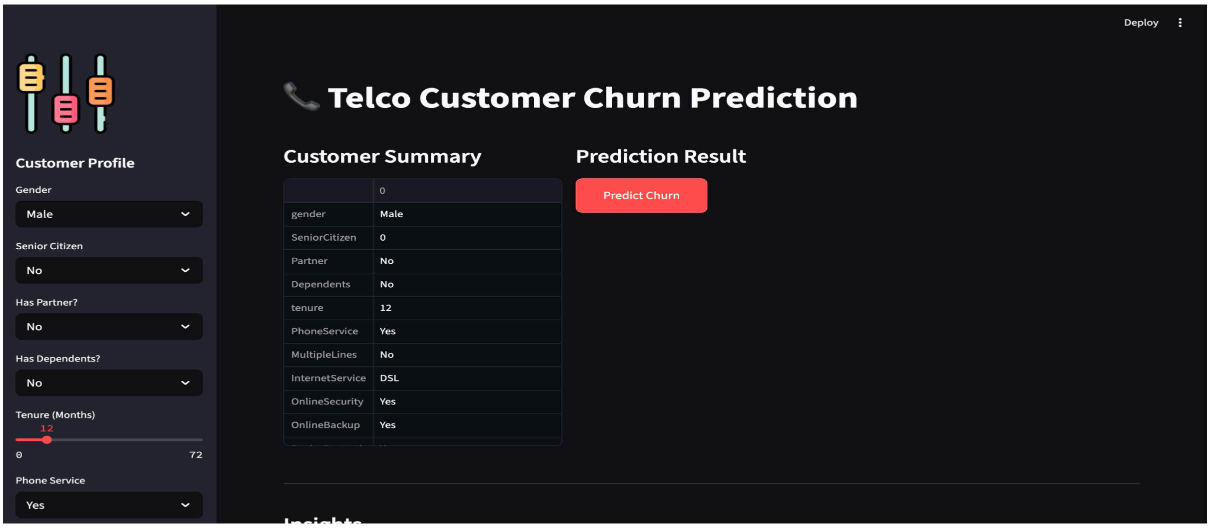
- **Feature Importance Visualization**: Bar chart showing the top factors contributing to the model's decision.

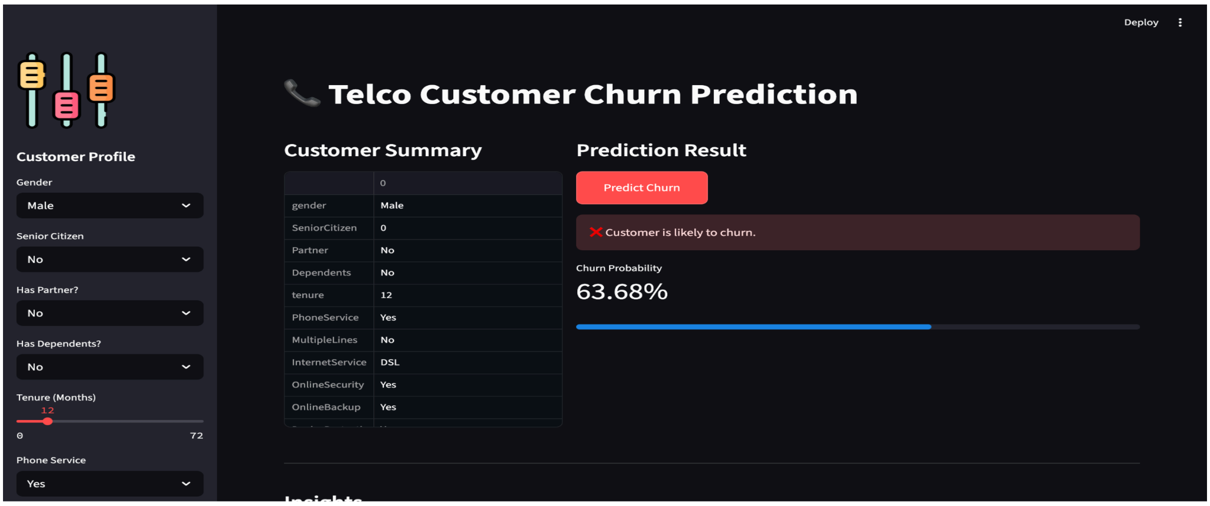
- **Interactive Feedback**: Users can adjust inputs dynamically and observe changes in churn probability.

- **Professional UI Design**: Clean layout with custom theming, metric cards, and charting (via Matplotlib & Seaborn).

- **Deployment-Ready**: The app runs locally or can be deployed using platforms like **Streamlit Cloud**, **Render**, or **Heroku** for enterprise use.

This app empowers business teams to **simulate customer scenarios**, understand churn drivers, and implement timely retention strategies—without needing direct access to the backend model or code.





Next Steps – What Could Come Next

1. **Integrate SHAP/LIME in Production**
   While LIME was used for exploratory explainability, integrating SHAP or LIME directly into the app can offer **instance-level interpretability** to support decision-making.

2. **Deploy the App Publicly**
   Hosting the Streamlit application on a cloud platform (e.g., **Streamlit Community Cloud**, **Render**, or **Heroku**) would allow wider usage by sales or retention teams.

3. **Enhance Data Pipeline**
   Build an **automated pipeline** to continuously pull updated customer data from CRM systems and retrain models periodically to adapt to business changes.

4. **Model Monitoring & Drift Detection**
   Implement monitoring to detect **concept drift** or a drop in prediction accuracy over time, which can be addressed through retraining or feature updates.

5. **Add Risk-Based Alerts**
   Extend the app to notify internal teams when **high churn-risk customers** are predicted, enabling proactive intervention.

6. **A/B Testing of Retention Strategies**
   Use the predictions to guide **targeted retention campaigns** and evaluate their effectiveness through **A/B testing** and uplift modeling.

7. **Multi-Model Comparison Dashboard**
   Integrate a backend feature to allow **model versioning** and comparison (e.g., Random Forest vs Logistic Regression) to track performance evolution.

8. **Include Customer Feedback Loop**
   Introduce a system to capture **actual outcomes** (e.g., whether the customer actually churned), enabling continuous **model evaluation and learning**.