CUSTOMER CHURN ANALYSIS & PREDICTION REPORT

1. Introduction

- Customer churn is a critical business problem that directly impacts revenue and long-term growth.
- This project focuses on analyzing customer behavior, identifying churn drivers, and predicting churn probability using machine learning.
- The final outcome is an interactive analytics solution supported by a predictive model and actionable business insights.

2. Dataset Description

The dataset consists of **15 customer records** with demographic, behavioral, and subscription-related attributes.

Key Fields:

- Customer_ID
- Age, Gender, City
- Signup_Date, Last_Login_Date
- Subscription_Type
- Monthly_Charges, Tenure_Months, Total_Revenue
- Usage_Frequency, Support_Tickets
- Churn_Flag (0 = Active, 1 = Churned)

Customer_ID	Age	Gender	City	Signup_Date	Last_Login_Date	Subscription_Type	Monthly_Charges	Tenure_Months	Total_Revenue	Usage_Frequency	Support_Tickets	Churn_Flag
C001	28	Male	Bangalore	15-01-2022	20-12-2024	Premium	999	24	23976	18	1	0
C002	35	Female	Hyderabad	10-08-2021	12-09-2024	Basic	499	36	17964	5	6	1
C003	42	Male	Delhi	20-05-2020	28-12-2024	Standard	699	48	33552	14	2	0
C004	26	Female	Chennai	01-03-2023	15-06-2024	Basic	499	15	7485	4	5	1
C005	31	Male	Pune	18-11-2022	30-12-2024	Premium	999	25	24975	20	0	0
C006	45	Female	Mumbai	10-09-2019	02-04-2024	Standard	699	60	41940	6	8	1
C007	29	Male	Bangalore	22-01-2023	29-12-2024	Basic	499	23	11477	9	3	0
C008	38	Female	Kolkata	14-02-2021	18-03-2024	Standard	699	46	32154	7	7	1
C009	33	Male	Delhi	07-07-2022	10-12-2024	Premium	999	30	29970	16	1	0
C010	27	Female	Hyderabad	12-06-2023	20-05-2024	Basic	499	12	5988	3	4	1
C011	41	Male	Mumbai	05-01-2020	25-12-2024	Premium	999	60	59940	19	0	0
C012	36	Female	Pune	09-10-2021	01-08-2024	Standard	699	38	26562	8	5	1
C013	24	Male	Chennai	15-08-2023	22-12-2024	Basic	499	10	4990	11	1	0
C014	39	Female	Bangalore	11-11-2020	19-02-2024	Standard	699	50	34950	6	6	1
C015	34	Male	Delhi	03-04-2022	27-12-2024	Premium	999	32	31968	17	2	0

The dataset is structured and suitable for churn analysis and modeling.

3. Data Cleaning & Preprocessing

The dataset was reviewed for data quality and consistency.

Steps Performed:

- Verified absence of missing and duplicate values

```
[9]: #null counts
     df.isnull().sum()
```

```
[9]: Customer_ID          0
     Age                  0
     Gender               0
     City                 0
     Signup_Date          0
     Last_Login_Date      0
     Subscription_Type    0
     Monthly_Charges      0
     Tenure_Months        0
     Total_Revenue        0
     Usage_Frequency      0
     Support_Tickets      0
     Churn_Flag           0
     dtype: int64
```

```
[10]: #count of duplicate rows
      df.duplicated().sum()
```

```
[10]: np.int64(0)
```

- Validated numerical ranges (age, charges, tenure)

```
[8]: #statistical summary (numerical columns)
     df.describe()
```

[8]:

	Age	Signup_Date	Last_Login_Date	Monthly_Charges	Tenure_Months	Total_Revenue	Usage_Frequency	Support_Tickets	Churn_Flag
count	15.000000	15	15	15.000000	15.000000	15.000000	15.000000	15.000000	15.000000
mean	33.866667	2021-11-23 00:00:00	2024-09-14 00:00:00	732.333333	33.933333	25859.400000	10.866667	3.400000	0.466667
min	24.000000	2019-09-10 00:00:00	2024-02-19 00:00:00	499.000000	10.000000	4990.000000	3.000000	0.000000	0.000000
25%	28.500000	2020-12-28 12:00:00	2024-06-02 00:00:00	499.000000	23.500000	14720.500000	6.000000	1.000000	0.000000
50%	34.000000	2022-01-15 00:00:00	2024-12-10 00:00:00	699.000000	32.000000	26562.000000	9.000000	3.000000	0.000000
75%	38.500000	2022-12-20 12:00:00	2024-12-26 00:00:00	999.000000	47.000000	32853.000000	16.500000	5.500000	1.000000
max	45.000000	2023-08-15 00:00:00	2024-12-30 00:00:00	999.000000	60.000000	59940.000000	20.000000	8.000000	1.000000
std	6.345602	NaN	NaN	212.692490	16.280868	14875.096296	5.926534	2.640346	0.516398

- Confirmed target variable consistency

```
[13]: #churn value counts
      df['Churn_Flag'].value_counts()
```

```
[13]: Churn_Flag
      0    8
      1    7
      Name: count, dtype: int64
```

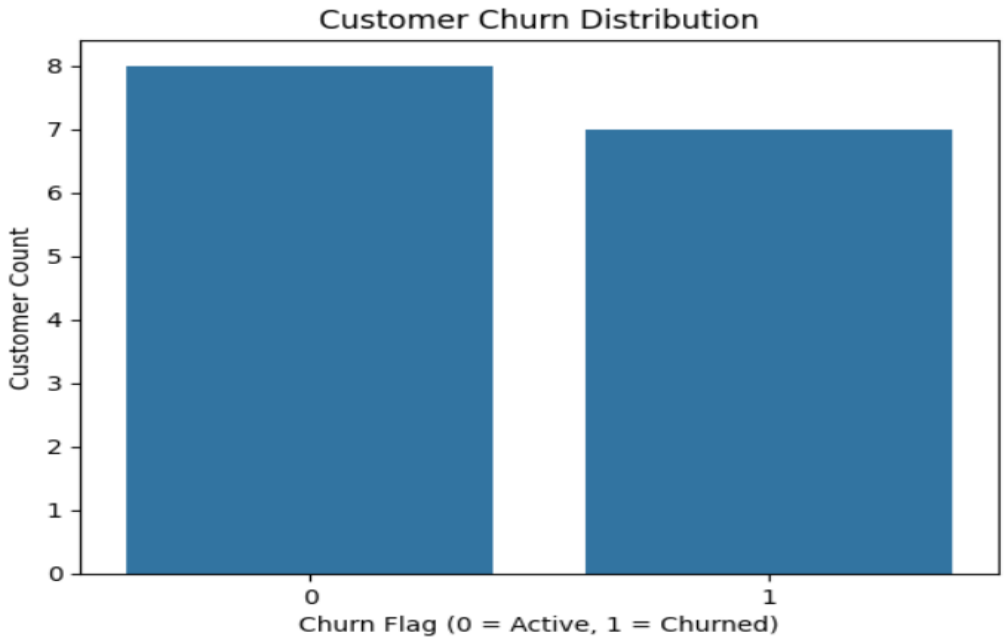
Outcome:
The dataset was clean and required minimal preprocessing, making it ready for analysis and modeling.

4. Exploratory Data Analysis (EDA)

EDA was conducted to understand customer behavior and churn patterns.

Key Observations:

- Churn rate is moderately balanced between active and churned customers

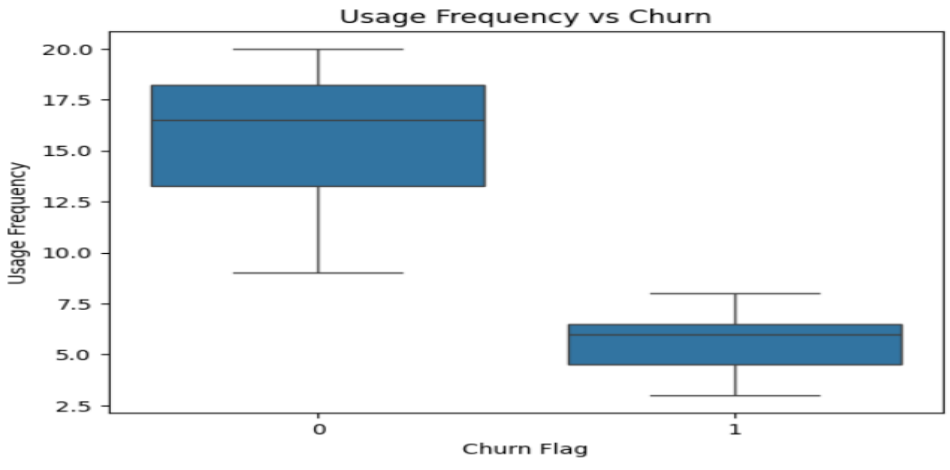



```
[14]: Churn_Flag
      0    53.333333
      1    46.666667
      Name: proportion, dtype: float64
```

- Churned customers generally have:
    - **Lower usage frequency**

Usage Frequency vs Churn (Visualization)

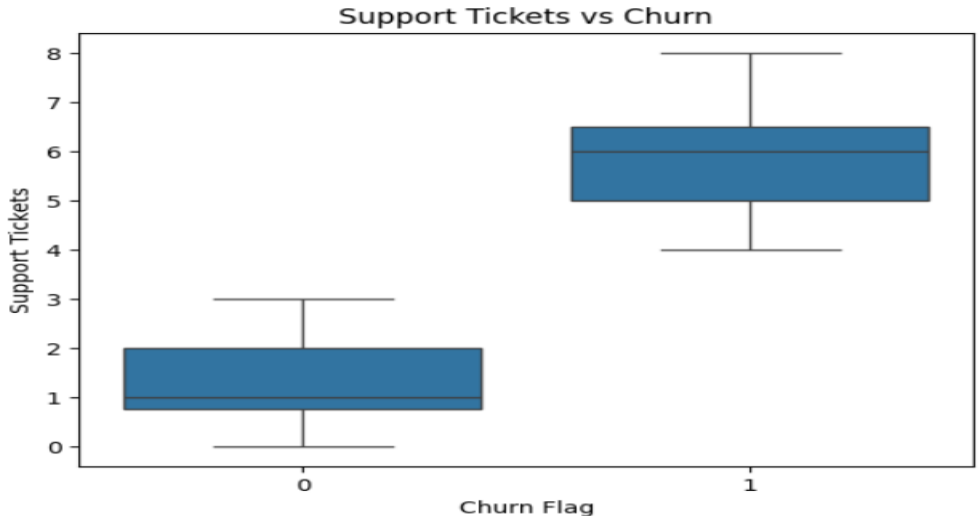
```
[20]: plt.figure()
      sns.boxplot(x='Churn_Flag', y='Usage_Frequency', data=df)
      plt.title("Usage Frequency vs Churn")
      plt.xlabel("Churn Flag")
      plt.ylabel("Usage Frequency")
      plt.show()
```



- **Higher support tickets**

Support Tickets vs Churn ¶

```
[21]: plt.figure()
sns.boxplot(x='Churn_Flag', y='Support_Tickets', data=df)
plt.title("Support Tickets vs Churn")
plt.xlabel("Churn Flag")
plt.ylabel("Support Tickets")
plt.show()
```



- Shorter tenure

Churn vs Numerical Features (Mean Comparison)

```
[16]: df.groupby('Churn_Flag')[num_cols].mean()
```

[16]:

Churn_Flag	Age	Monthly_Charges	Tenure_Months	Total_Revenue	Usage_Frequency	Support_Tickets
0	32.750000	836.500000	31.500000	27606.000000	15.500000	1.250000
1	35.142857	613.285714	36.714286	23863.285714	5.571429	5.857143

- Basic subscription users show higher churn compared to Premium users

Subscription Type vs Churn

```
[18]: pd.crosstab(df['Subscription_Type'], df['Churn_Flag'], normalize='index') * 100
```

[18]:

Subscription_Type \ Churn_Flag	0	1
Basic	40.0	60.0
Premium	100.0	0.0
Standard	20.0	80.0

EDA helped identify early indicators of churn and informed feature engineering.

5. Feature Engineering

Additional features were created to improve model performance and interpretability.

Engineered Features:

- **Recency_Days** – Days since last login

Recency Feature

```
23]: #Reference date = latest login date in dataset
reference_date = df['Last_Login_Date'].max()

df['Recency_Days'] = (reference_date - df['Last_Login_Date']).dt.days
```

- **Avg_Monthly_Spend** – Normalized spending behavior

Average Monthly Spend

```
[24]: df['Avg_Monthly_Spend'] = df['Total_Revenue'] / df['Tenure_Months']
```

- **Engagement_Score** – Usage frequency minus support tickets

Engagement Score

```
[25]: df['Engagement_Score'] = df['Usage_Frequency'] - df['Support_Tickets']
```

- Categorical encoding for gender, city, and subscription type

```
[32]: #one hot encoding
df_model = pd.get_dummies(
    df_model,
    columns=['Gender', 'City', 'Subscription_Type', 'Customer_Value'],
    drop_first=True
)
```

These features capture customer engagement, satisfaction, and inactivity patterns.

6. Predictive Modeling

Two machine learning models were developed:

- Logistic Regression (baseline, interpretable)

logistic regression

```
[37]: #Logistic regression
log_model = LogisticRegression()
log_model.fit(X_train, y_train)

y_pred_log = log_model.predict(X_test)
y_prob_log = log_model.predict_proba(X_test)[:, 1]
```

- Random Forest Classifier (final selected model)

Random Forest

```
[38]: #Random forest

rf_model = RandomForestClassifier(
    n_estimators=100,
    random_state=42
)

rf_model.fit(X_train, y_train)

y_pred_rf = rf_model.predict(X_test)
y_prob_rf = rf_model.predict_proba(X_test)[:, 1]
```

Evaluation Metrics Used:

- Accuracy
- Precision
- Recall
- ROC-AUC

Model Selection:
Random Forest was selected due to better performance and ability to capture non-linear relationships.

Comparison table

[41]:
```
model_comparison = pd.DataFrame({
    'Model': ['Logistic Regression', 'Random Forest'],
    'Accuracy': [
        accuracy_score(y_test, y_pred_log),
        accuracy_score(y_test, y_pred_rf)
    ],
    'Precision': [
        precision_score(y_test, y_pred_log),
        precision_score(y_test, y_pred_rf)
    ],
    'Recall': [
        recall_score(y_test, y_pred_log),
        recall_score(y_test, y_pred_rf)
    ],
    'ROC_AUC': [
        roc_auc_score(y_test, y_prob_log),
        roc_auc_score(y_test, y_prob_rf)
    ]
})

model_comparison
```

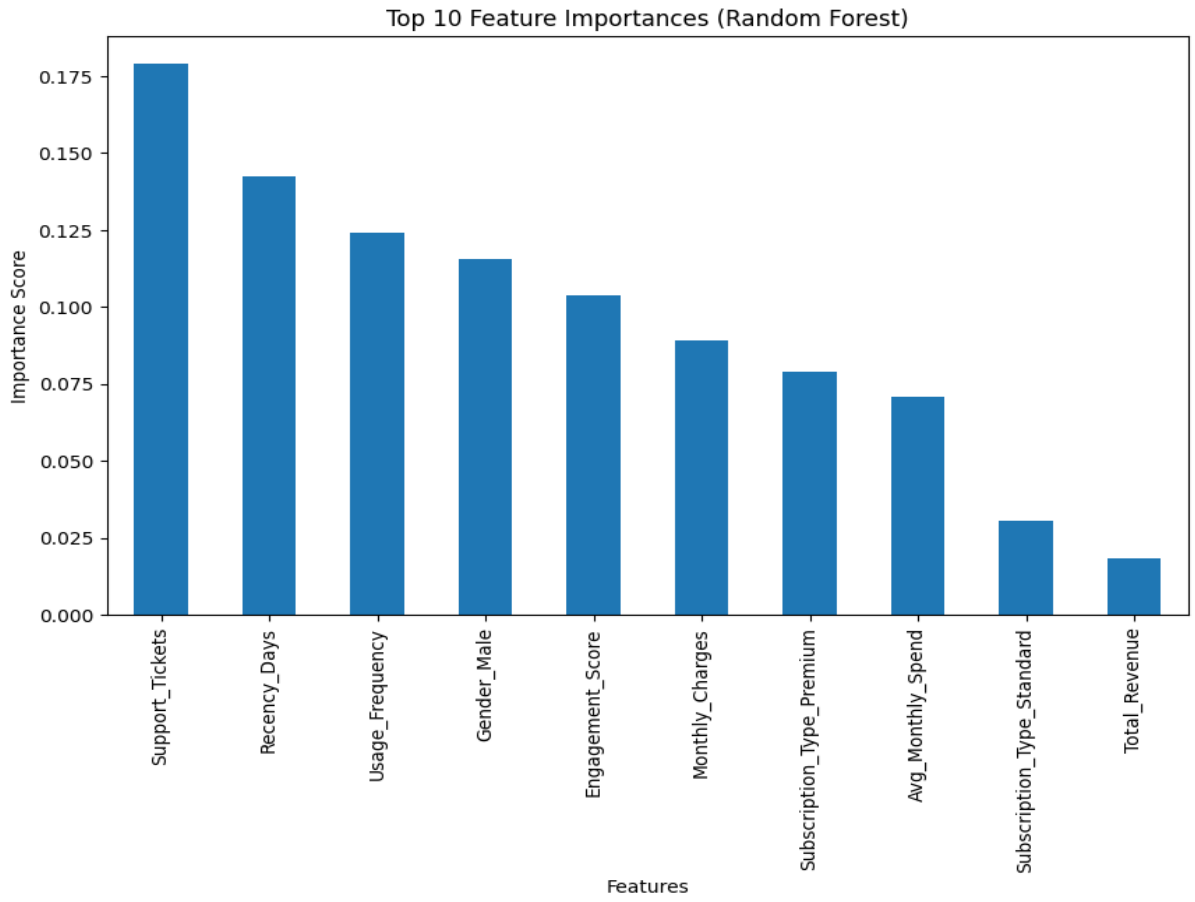
[41]:

	Model	Accuracy	Precision	Recall	ROC_AUC
0	Logistic Regression	0.8	0.666667	1.0	1.0
1	Random Forest	1.0	1.000000	1.0	1.0

7. Model Explainability

Model outputs were analyzed to understand churn drivers.

```
[43]: plt.figure(figsize=(10,6))
      feature_importance.head(10).plot(kind='bar')
      plt.title("Top 10 Feature Importances (Random Forest)")
      plt.ylabel("Importance Score")
      plt.xlabel("Features")
      plt.show()
```



Top Influencing Factors:

- High support ticket count
- Customer inactivity (Recency_Days)
- Low engagement score
- Short tenure
- Basic subscription type

Explainability techniques ensured model decisions were aligned with business logic.

8. Dashboard & Visualization

Due to system performance constraints with Power BI, dashboards were implemented using **Streamlit**.

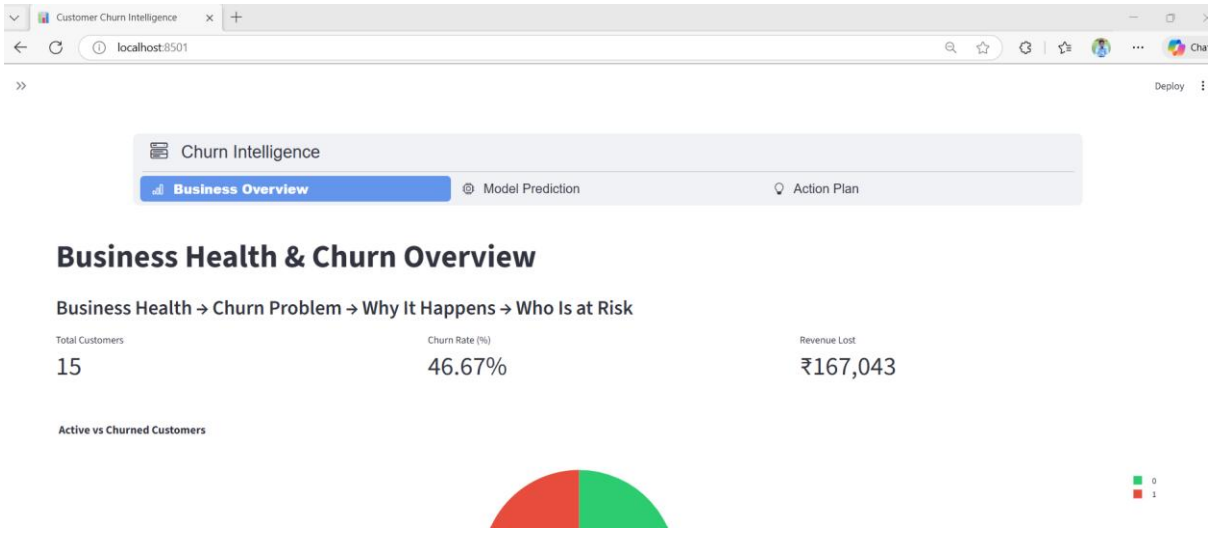
Streamlit Application Includes:

- Business health KPIs
- Churn distribution and drivers
- ML-based churn probability
- High-risk customer identification
- Actionable insights and recommendations

The dashboard follows a storytelling approach:
Business Health → Churn Problem → Why It Happens → Who Is at Risk → What Action to Take

Streamlit App – Home / Navigation

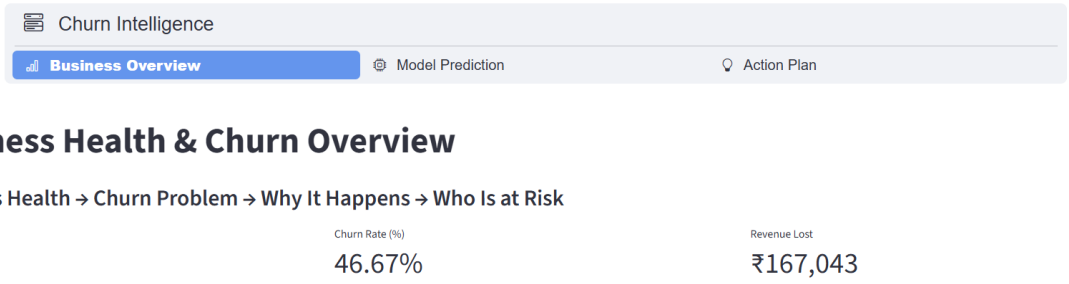
Customer Churn Intelligence – Application Navigation



Description :

This screenshot shows the main Streamlit application with the option menu–based navigation. The application is structured into three sections: Business Overview, Model Prediction, and Action Plan.

Business Health KPIs

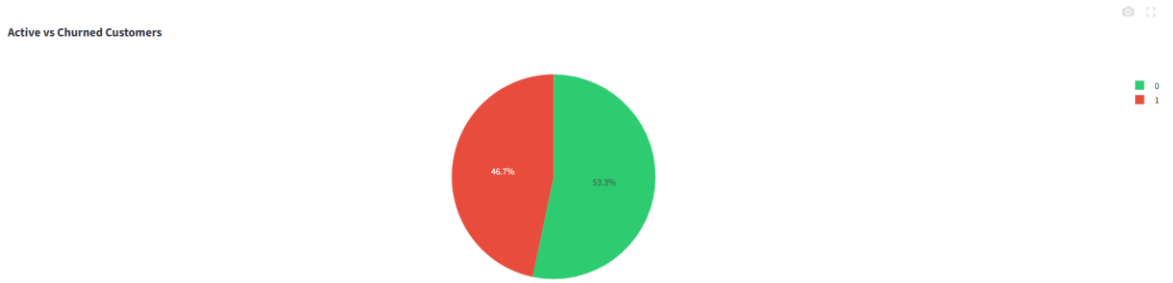


Description:

This view presents key business metrics including total customers, churn rate, and revenue lost due to churn. These KPIs provide a high-level understanding of overall business health.

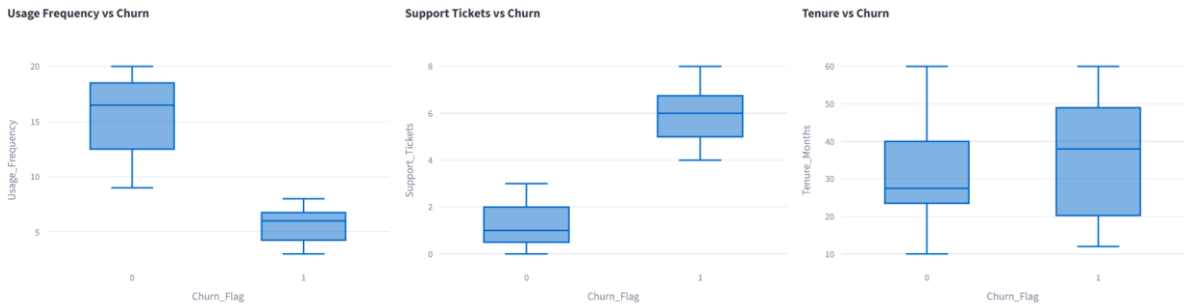
Churn Distribution

Churn Distribution (Active vs Churned Customers)



Churn Drivers – Usage & Support

Churn Drivers Analysis

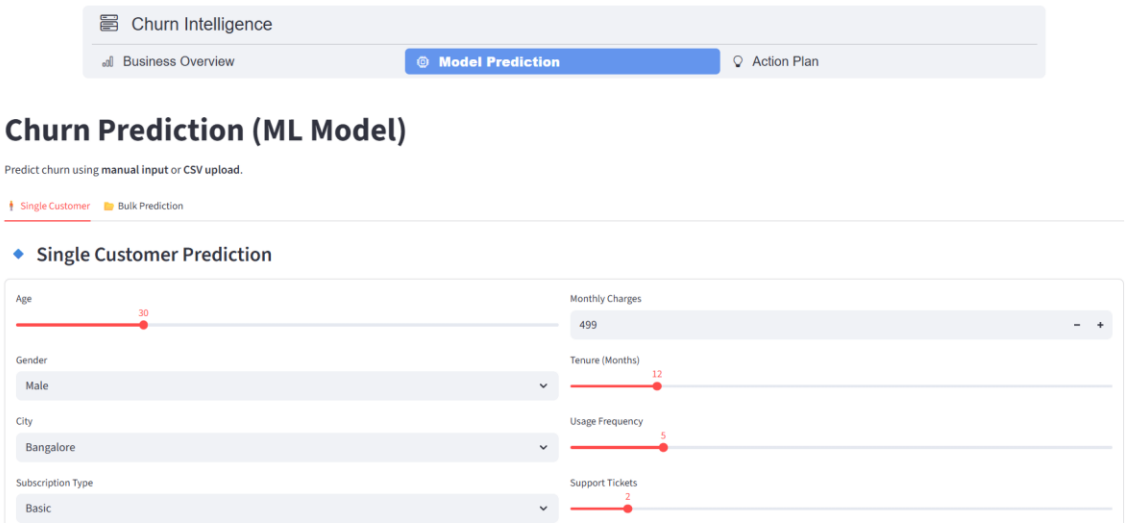


Description:

This section highlights the primary factors contributing to churn, including usage frequency, number of support tickets, and customer tenure.

Model Prediction – Churn Probability (Single Customer)

Churn Probability Prediction (Single Customer Input)

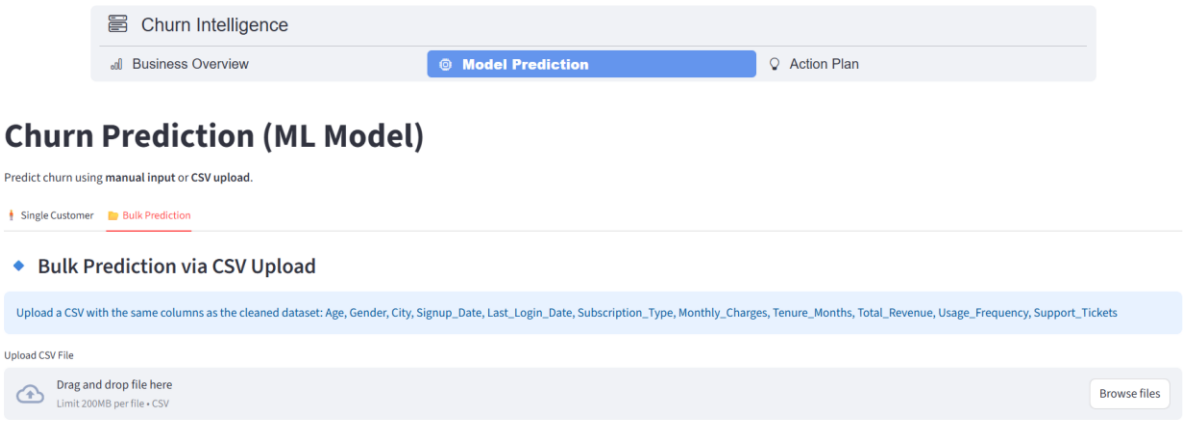


Description:

This screenshot represents the machine learning–based churn prediction for a single customer using manually adjusted input values in the Streamlit application.

Model Prediction – Bulk CSV Upload

Bulk Churn Prediction via CSV Upload

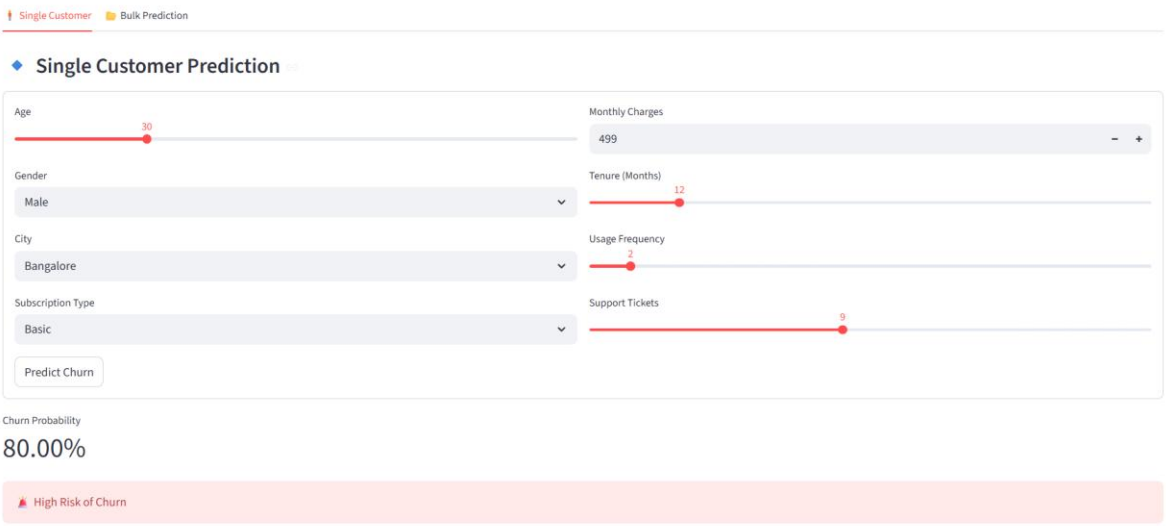


Description:

This view demonstrates bulk churn prediction, where users can upload a CSV file and receive churn probability scores for multiple customers.

High-Risk Customers Identification

High-Risk Customer Identification

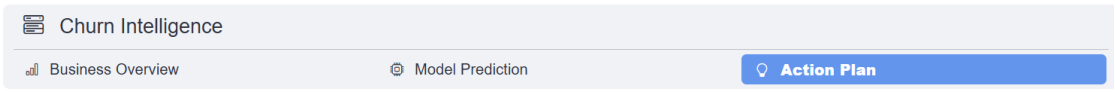


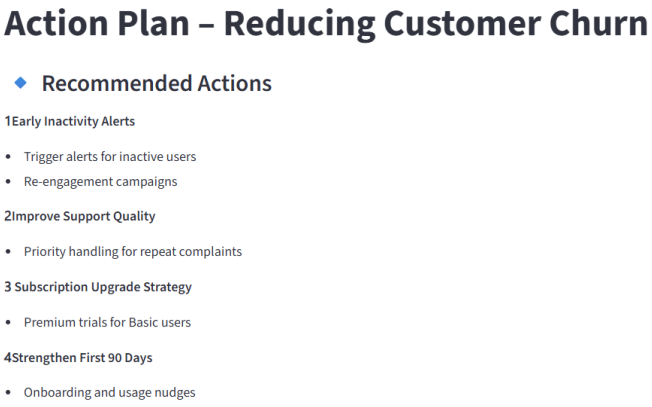
Description:

This table lists customers with high churn probability, enabling targeted retention actions by business and customer success teams.

Action Plan & Recommendations

Churn Reduction Action Plan





Description:

This section outlines actionable business strategies derived from churn analysis and model insights to reduce customer attrition.

9. Business Insights

Key insights derived from analysis and modeling:

- Low engagement and inactivity are the strongest churn indicators
- Customers with frequent support issues are more likely to churn
- Early-stage customers require proactive engagement
- Revenue at risk can be identified before actual churn occurs

10. Churn Reduction Recommendations

Recommended Actions:

1. Early inactivity alerts and re-engagement campaigns
2. Improved customer support for high-ticket customers
3. Subscription upgrade incentives for Basic users
4. Strong onboarding during the first 90 days

These actions can help reduce churn and improve customer lifetime value.

11. Deployment Readiness

- Trained model and preprocessing artifacts saved using Joblib
- Streamlit app supports:
    - Single customer prediction (manual input)
    - Bulk prediction (CSV upload)
- Solution can be integrated with BI tools or CRM systems in the future

12. Conclusion

This project demonstrates an end-to-end churn analytics solution combining data analysis, machine learning, visualization, and business strategy.
The approach is scalable, explainable, and aligned with real-world business needs.

13. References / Tools Used

- Python (Pandas, NumPy, Scikit-learn)
- Streamlit, Plotly
- Machine Learning (Random Forest)