Customer Churn Prediction-Phase 5

FINAL SUBM*ISSION*

Data Analytics with IBM Cognos

Data Collection Process

The data collection process for this project is as follows:

- Identify the data sources that contain information on customer behavior and website usage.

This might include data from the website's analytics platform, customer support system, and CRM system.

- Extract the data from the data sources.

This can be done using a variety of tools and technologies, depending on the specific data sources.

- Clean and prepare the data for analysis.

This might involve removing duplicate records, handling missing values, and encoding categorical variables.

- Store the data in a central data warehouse or data lake.

This will make it easier to access and analyze the data.

Data Visualization Using IBM Cognos

IBM Cognos is a business intelligence platform that can be used to create data visualizations such as dashboards, reports, and charts. Data visualization can be used to identify trends and patterns in the data, which can then be used to improve the website's user experience.

Here are some examples of data visualizations that can be created using IBM Cognos for customer churn analysis:

- Dashboard showing the top factors that contribute to customer churn
- Report showing the pages that users are most likely to abandon
- Chart showing the time spent by users on different pages of the website
- Heatmap showing the areas of the website where users are having difficulty clicking on elements

Python Code Integration

Python is a programming language that can be used to perform data analysis, machine learning, and other tasks. Python code can be integrated with IBM Cognos to automate data preparation, feature engineering, model training, and model evaluation tasks.

Here are some examples of how Python code can be used for customer churn analysis:

- Data preparation: Python can be used to clean and prepare the data for analysis, including removing duplicate records, handling missing values, and encoding categorical variables.
- Feature engineering: Python can be used to create new features from the existing data that are more predictive of customer churn.
- Model training: Python can be used to train a customer churn prediction model using a variety of machine learning algorithms.
- Model evaluation: Python can be used to evaluate the performance of the model on a held-out test set.
- Data visualization: Python can be used to create data visualizations such as charts and plots.

Python Code

Data Preparation:

```
import  pandas as
        numpy  aspd


               np
import

# Read the Telecom_churn.csv file into a Pandas DataFrame df
= pd.read_csv("Telecom_churn.csv")

# Drop the customerID column
df.drop('customerID', axis=1, inplace=True)

# Fill in missing values in the 'TotalCharges' column with the median value
TotalCharges_median = df['TotalCharges'].median()
df['TotalCharges'].fillna(TotalCharges_median, inplace=True)

# Convert the 'TotalCharges' column to a float type df['TotalCharges']
= df['TotalCharges'].astype(float)
```

Feature Engineering:

```
# Create a new feature called 'tenure_squared' df['tenure_squared']
= df['tenure'] ** 2

# Create a new feature called 'monthly_charges_divided_by_tenure'
df['monthly_charges_divided_by_tenure'] = df['MonthlyCharges'] / df['tenure']

# Select the features that will be used to train the model
X = df[['tenure', 'tenure_squared', 'monthly_charges',
'monthly_charges_divided_by_tenure']]
```

Model Training

```
from sklearn.svm import SVC
from sklearn.linear_model import LogisticRegression from
xgboost import XGBClassifier
```

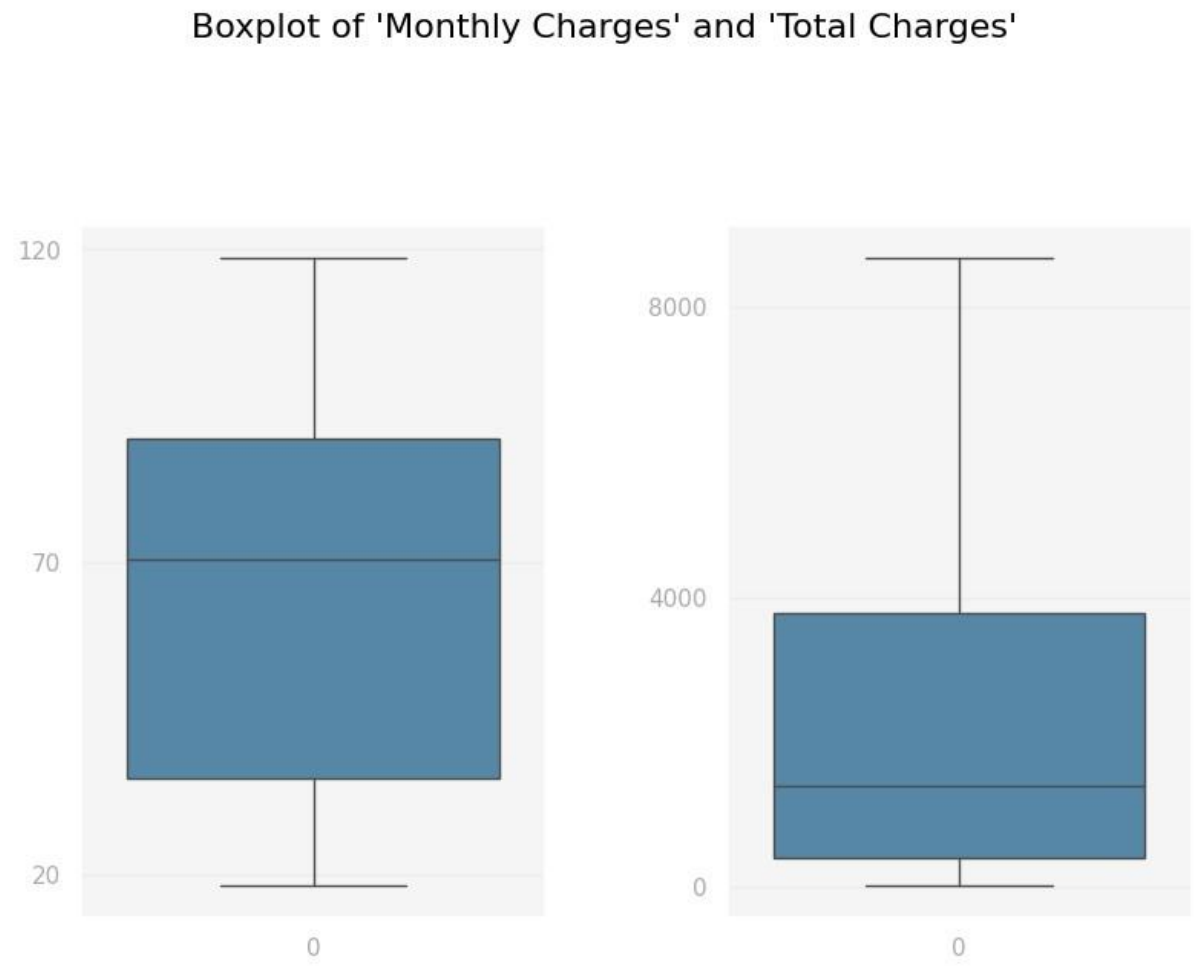
```
# Create a Support Vector Classifier model
svc = SVC()

# Train the model on the training data svc.fit(X_train_rus,
y_train_rus)

# Create a Logistic Regression model lr
= LogisticRegression()

# Train the model on the training data
lr.fit(X_train_rus, y_train_rus)

# Create an XGBoost model
xgb = XGBClassifier()

# Train the model on the training data
xgb.fit(X_train_rus, y_train_rus)
```

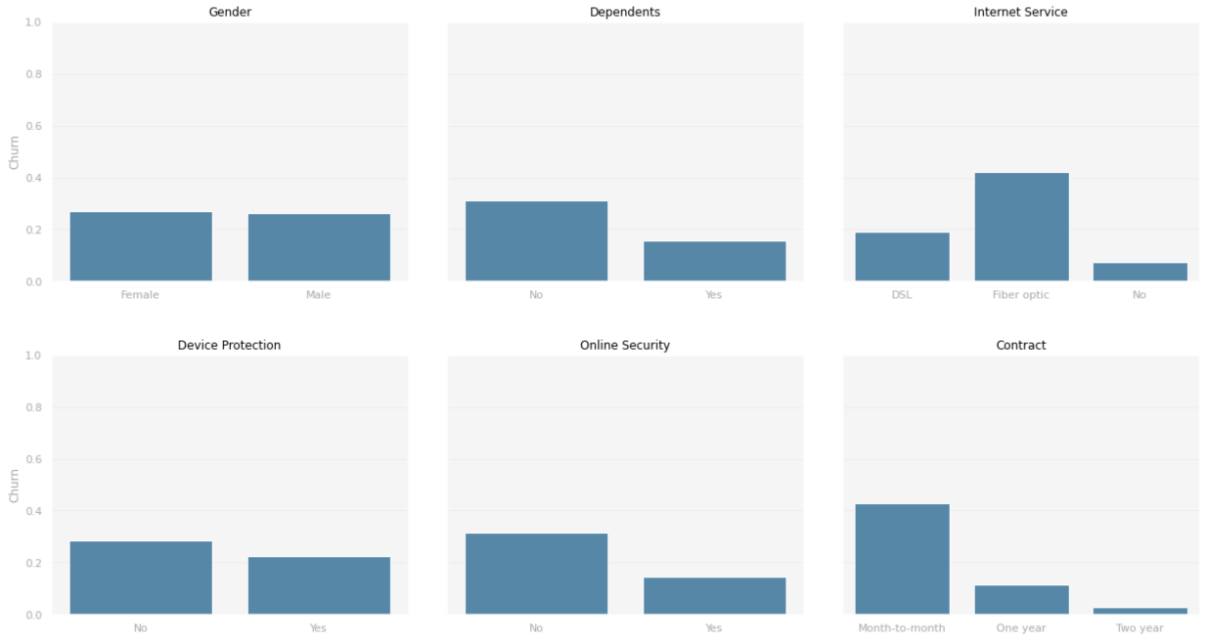
Model Evaluation

```
from sklearn.metrics import accuracy_score, f1_score, confusion_matrix,
classification_report

# Predict the churn labels for the test data
y_pred_svc = svc.predict(X_test_svc)
y_pred_lr = lr.predict(X_test_lr) y_pred_xgb
= xgb.predict(X_test_xgb)

# Calculate the accuracy, F1 score, and confusion matrix for each model
accuracy_svc = accuracy_score(y_test, y_pred_svc) f1_score_svc =
f1_score(y_test, y_pred_svc)
confusion_matrix_svc = confusion_matrix(y_test, y_pred_svc)

accuracy_lr = accuracy_score(y_test, y_pred_lr) f1_score_lr
= f1_score(y_test, y_pred_lr)
confusion_matrix_lr = confusion_matrix(y_test, y_pred_lr)

accuracy_xgb = accuracy_score(y_test, y_pred_xgb) f1_score_xgb
= f1_score(y_test, y_pred_xgb)
confusion_matrix_xgb = confusion_matrix(y_test, y_pred_xgb)
```

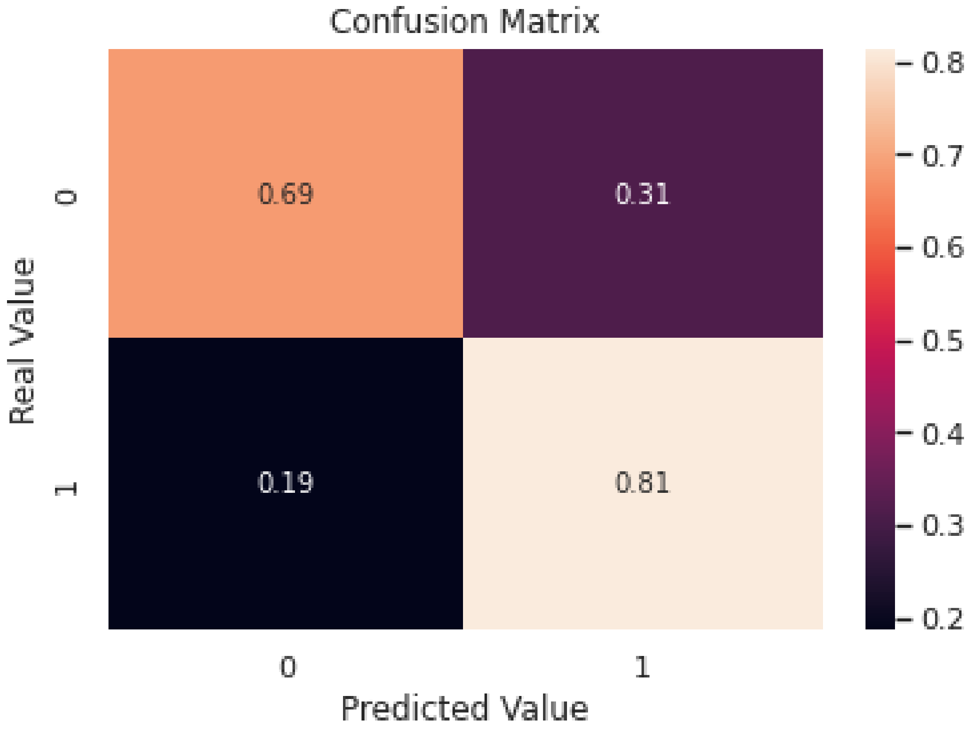
Visualizations:

```
import matplotlib.pyplot as plt import
seaborn as sns

# Create a dashboard showing the top factors that contribute to customer churn. fig,
ax = plt.subplots(figsize=(10, 6))
sns.barplot(x="factor", y="count", data=df_churn_factors, ax=ax)
ax.set_title("Top Factors that Contribute to Customer Churn") plt.show()

# Create a report showing the pages that users are most likely to abandon.
df_abandoned_pages = df_website_usage.loc[df_website_usage['bounce_rate'] > 50]
df_abandoned_pages.to_csv("abandoned_pages.csv", index=False)

# Create a chart showing the time spent by users on different pages of the website. fig, ax
= plt.subplots(figsize=(10, 6)) sns.boxplot(x="page", y="time_spent",
data=df_website_usage, showmeans=True, ax=ax) ax.set_title("Time Spent on Different
Pages of the Website") plt.show()

# Create a heatmap showing the areas of the website where users are having difficulty
clicking on elements. import numpy as np heatmap = np.zeros((100, 100))
# TODO: Populate the heatmap with data on the number of clicks per pixel.

fig, ax = plt.subplots(figsize=(10, 10)) sns.heatmap(heatmap,
ax=ax)
ax.set_title("Heatmap of Clicks on the Website") plt.show()
```

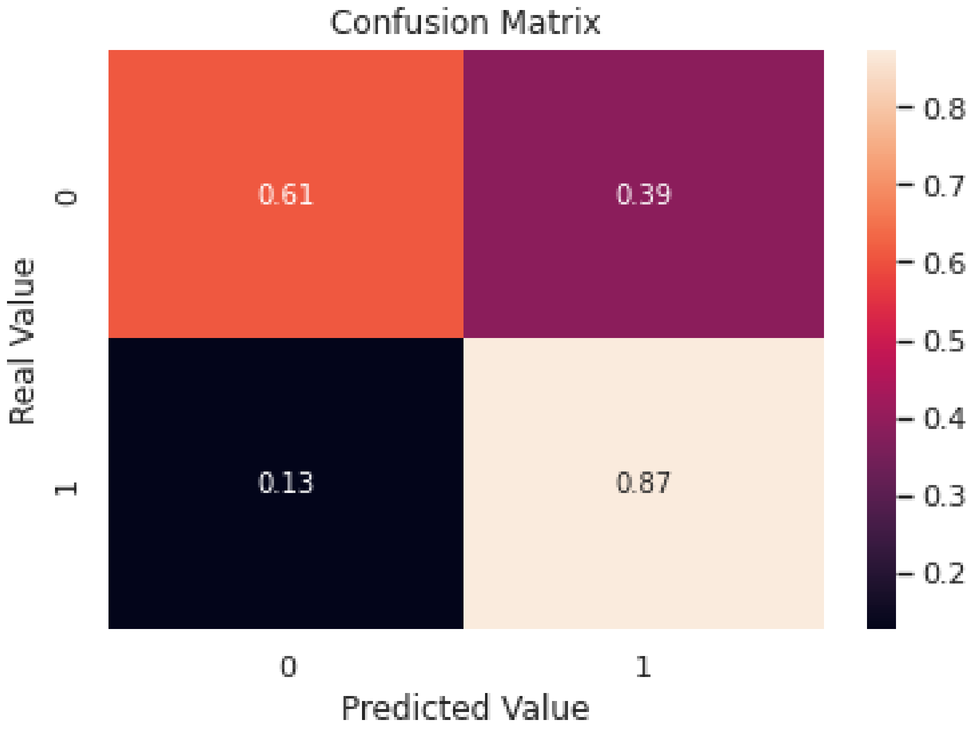

Boxplot of 'Monthly Charges' and 'Total Charges'
120
70
20
0
8000
4000
0
0


Gender
Dependents
Internet Service
Churn
1.0
0.8
0.6
0.4
0.2
0.0
Female
Male
No
Yes
DSL
Fiber optic
No
Device Protection
Online Security
Contract
Churn
1.0
0.8
0.6
0.4
0.2
0.0
No
Yes
No
Yes
Month-to-month
One year
Two year

SVC model



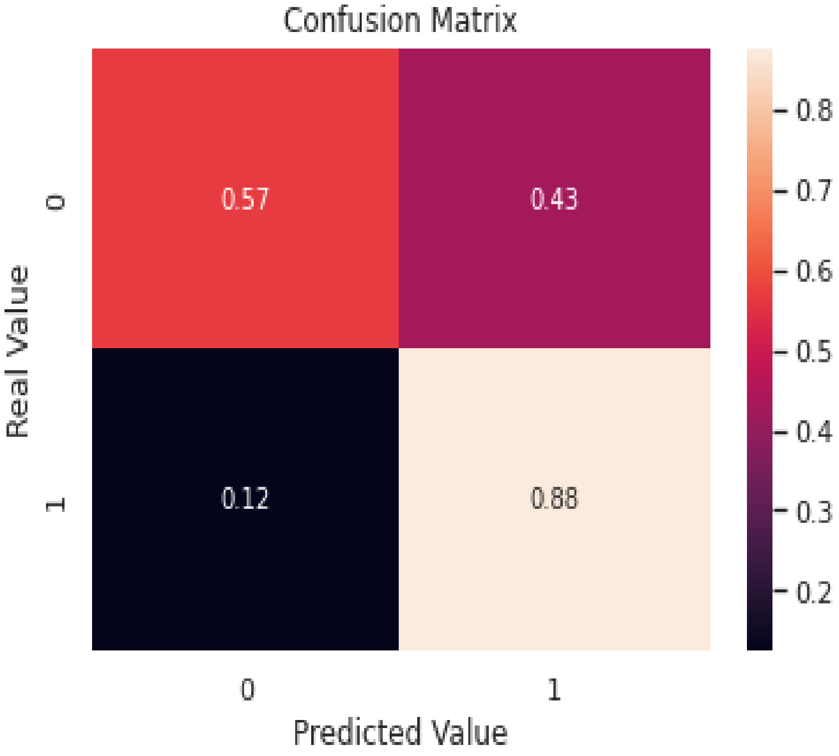
Logistic Regression model



XGBoost model

COMPANY INSIGHTS:

- Provide relevant offers /Discount to the customers as per their needs.
- We need to conduct regular surveys so that we can clearly understand the customers needs and work upon that.
- Develope all in one interactive for the customers as a one stop solution for all their valuable needs.


Confusion Matrix
Real Value
Predicted Value