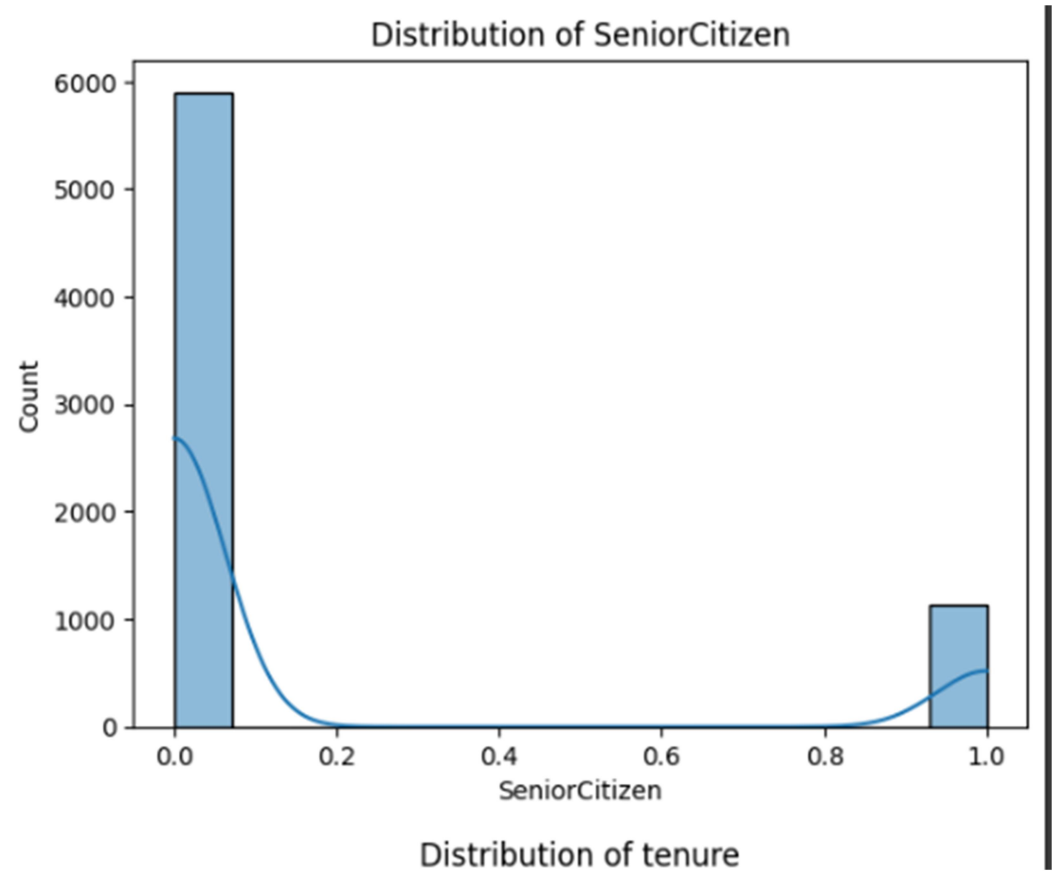
Customer churn prediction

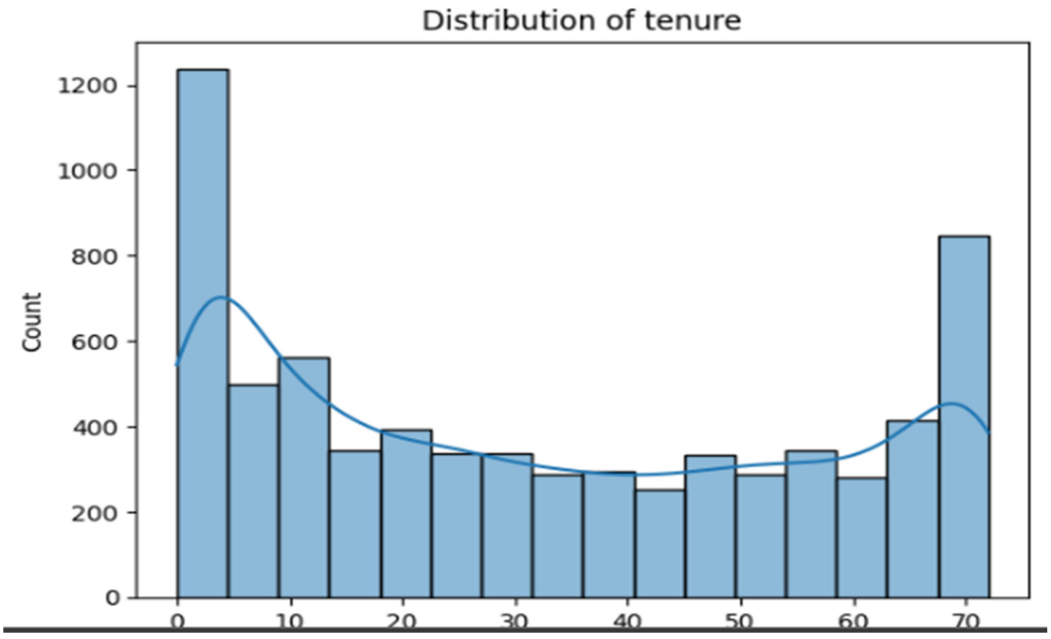
Code:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score

data = pd.read_csv('Customerchurn.csv')
X = data.drop('Churn', axis=1)
y = data['Churn']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
categorical_features = ['gender', 'Partner']
numeric_features = [col for col in X.columns if col not in categorical_features]
preprocessor = ColumnTransformer(
    transformers=[
        ('num', StandardScaler(), numeric_features),
        ('cat', OneHotEncoder(), categorical_features)
    ])
model = Pipeline([
    ('preprocessor', preprocessor),
    ('classifier', LogisticRegression())
])

model.fit(X_train, y_train)
y_pred = model.predict(X_test)
print("Confusion Matrix:")
print(confusion_matrix(y_test, y_pred))
print("\nClassification Report:")
```

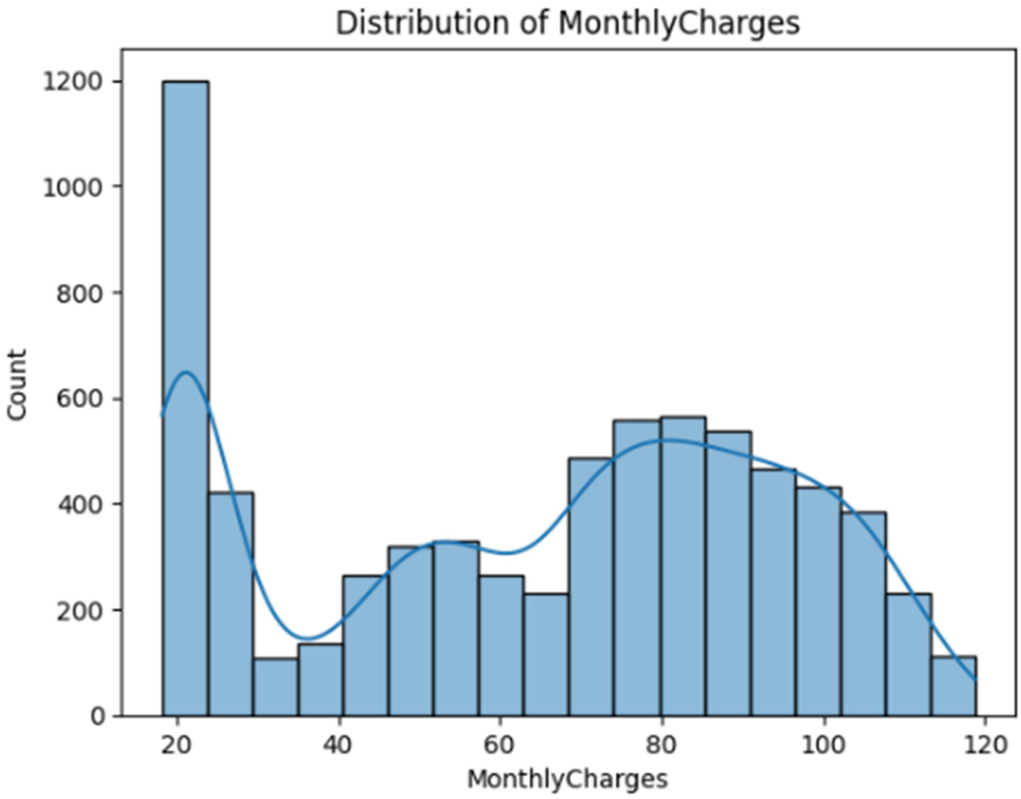
Output

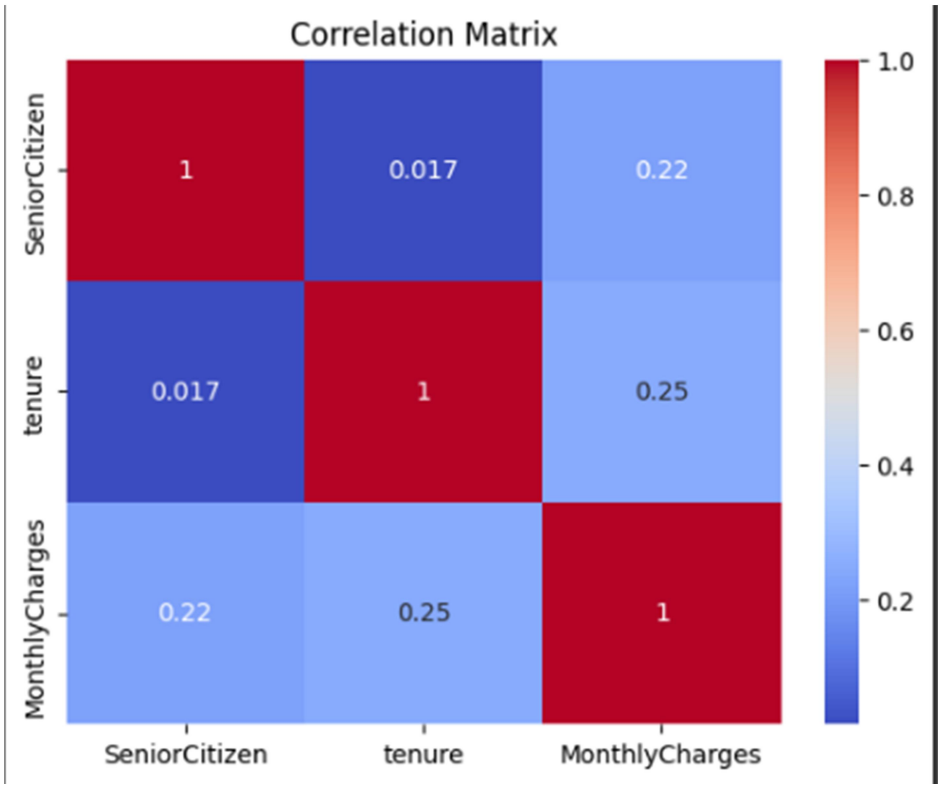


Distribution of tenure



tenure


Distribution of MonthlyCharges
Count
MonthlyCharges
0
200
400
600
800
1000
1200
20
40
60
80
100
120


Correlation Matrix
SeniorCitizen
tenure
MonthlyCharges
1
0.017
0.22
0.017
1
0.25
0.22
0.25
1
1.0
0.8
0.6
0.4
0.2