HW5_Part1_Henry_Romero

April 6, 2025

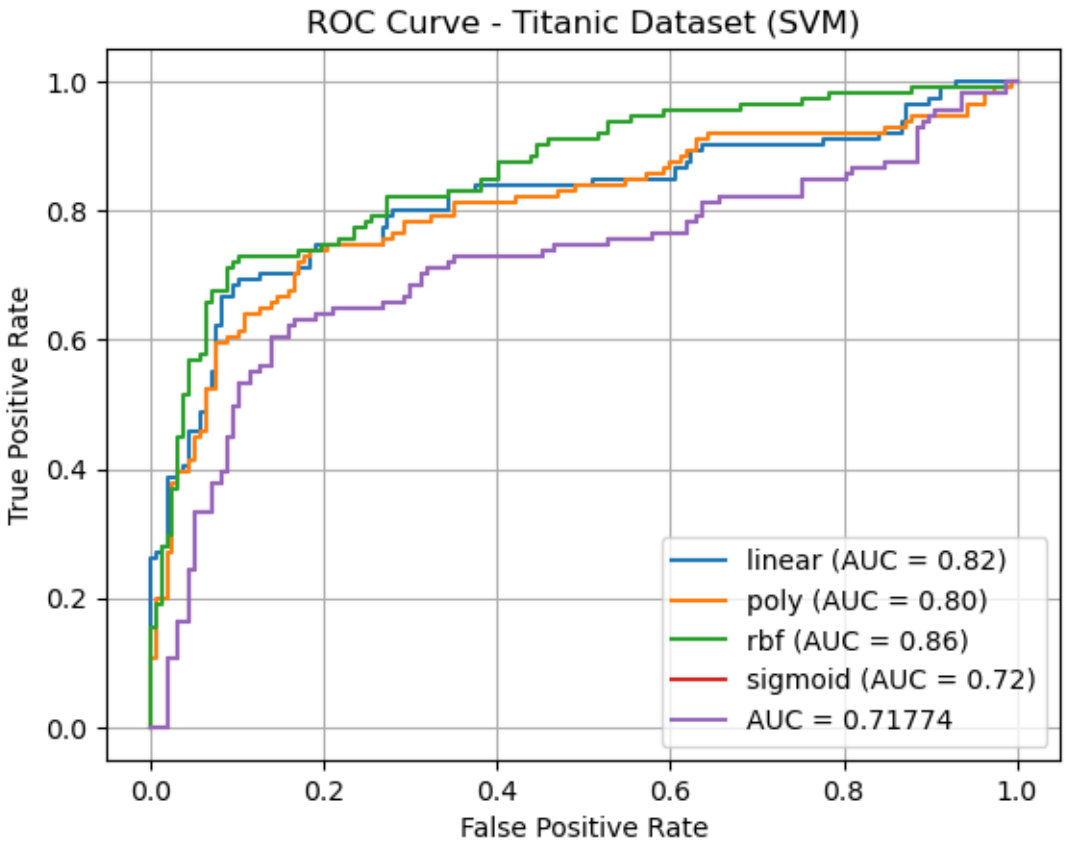
[24]:
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.svm import SVC
from sklearn.metrics import classification_report, precision_score,␣
 ↪recall_score, roc_curve, roc_auc_score
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import label_binarize
import matplotlib.pyplot as plt

# Loading Titanic dataset for SVM in Part 1
df = pd.read_csv('train.csv')

# Display basic info
print("Dataset Head:\n", df.head())
print("\nMissing Values:\n", df.isnull().sum())
print("Rows",len(df))
print(df.describe())

# Select features and target
features = ['Pclass', 'Sex', 'Age', 'SibSp', 'Parch', 'Fare', 'Embarked']
df = df[features + ['Survived']]

# Handle missing values
imputer = SimpleImputer(strategy='most_frequent')
df[['Embarked']] = imputer.fit_transform(df[['Embarked']])
df[['Age']] = SimpleImputer(strategy='mean').fit_transform(df[['Age']])

# categorical features
df['Sex'] = LabelEncoder().fit_transform(df['Sex'])
df['Embarked'] = LabelEncoder().fit_transform(df['Embarked'])

# Features and target
X = df[features]
y = df['Survived']

# Scale features
```

```
X_scaled = StandardScaler().fit_transform(X)

# Train/test split
# 30% of the data at a 42 state, will do the same for Part 2
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.3,␣
 ↪random_state=42)

# Experimenting with different kernels
# rbf had the highest accuracy at .85
kernels = ['linear', 'poly', 'rbf', 'sigmoid']

for kernel in kernels:
    print(f"\n SVM with kernel: {kernel}")
    clf = SVC(kernel=kernel, C=1.0, gamma='scale', probability=True) #SVM␣
 ↪classification on 4 kernels (rbf wins)
    clf.fit(X_train, y_train)

    y_pred = clf.predict(X_test)
    y_prob = clf.predict_proba(X_test)[:, 1]  # Prob for class '1' (Survived!)

    # Evaluation metrics with a classification report
    print("Classification Report:\n", classification_report(y_test, y_pred))
    precision = precision_score(y_test, y_pred)
    recall = recall_score(y_test, y_pred)
    auc = roc_auc_score(y_test, y_prob)

    print(f"Precision: {precision:.2f}")
    print(f"Recall: {recall:.2f}")
    print(f"ROC AUC: {auc:.5f}") # Different to show how higher AUC can be

    # Ploting ROC curve with all kernels
    fpr, tpr, _ = roc_curve(y_test, y_prob)
    plt.plot(fpr, tpr, label=f'{kernel} (AUC = {auc:.2f})')

# ROC curve graph
fpr, tpr, _ = roc_curve(y_test, y_prob)
roc_auc = roc_auc_score(y_test, y_prob)
plt.plot(fpr, tpr, label=f'AUC = {roc_auc:.5f}') # Longer to distinguish AUC
plt.title('ROC Curve - Titanic Dataset (SVM)')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.legend()
plt.grid(True)
plt.show()
print("The best model ran with a SVM classification was the 'rbf' kernel with␣
 ↪an ROC accuracy of .857")
```

```
Dataset Head:
    PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                 Name     Sex   Age  SibSp  \
0                             Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                              Heikkinen, Miss. Laina  female  26.0      0
3        Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                            Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare Cabin Embarked
0      0         A/5 21171   7.2500   NaN        S
1      0          PC 17599  71.2833   C85        C
2      0  STON/O2. 3101282   7.9250   NaN        S
3      0            113803  53.1000  C123        S
4      0            373450   8.0500   NaN        S

Missing Values:
 PassengerId      0
Survived          0
Pclass            0
Name              0
Sex               0
Age             177
SibSp             0
Parch             0
Ticket            0
Fare              0
Cabin           687
Embarked          2
dtype: int64
Rows 891
       PassengerId    Survived      Pclass         Age       SibSp  \
count   891.000000  891.000000  891.000000  714.000000  891.000000
mean    446.000000    0.383838    2.308642   29.699118    0.523008
std     257.353842    0.486592    0.836071   14.526497    1.102743
min       1.000000    0.000000    1.000000    0.420000    0.000000
25%     223.500000    0.000000    2.000000   20.125000    0.000000
50%     446.000000    0.000000    3.000000   28.000000    0.000000
75%     668.500000    1.000000    3.000000   38.000000    1.000000
max     891.000000    1.000000    3.000000   80.000000    8.000000

            Parch        Fare
```

```
count  891.000000  891.000000
mean     0.381594   32.204208
std      0.806057   49.693429
min      0.000000    0.000000
25%      0.000000    7.910400
50%      0.000000   14.454200
75%      0.000000   31.000000
max      6.000000  512.329200

 SVM with kernel: linear
Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.85      0.83       157
           1       0.77      0.70      0.74       111

    accuracy                           0.79       268
   macro avg       0.79      0.78      0.78       268
weighted avg       0.79      0.79      0.79       268

Precision: 0.77
Recall: 0.70
ROC AUC: 0.81563

 SVM with kernel: poly
Classification Report:
              precision    recall  f1-score   support

           0       0.76      0.90      0.83       157
           1       0.82      0.60      0.69       111

    accuracy                           0.78       268
   macro avg       0.79      0.75      0.76       268
weighted avg       0.79      0.78      0.77       268

Precision: 0.82
Recall: 0.60
ROC AUC: 0.80312

 SVM with kernel: rbf
Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.91      0.85       157
           1       0.84      0.68      0.76       111

    accuracy                           0.82       268
   macro avg       0.82      0.80      0.80       268
```

```
weighted avg       0.82      0.82      0.81       268

Precision: 0.84
Recall: 0.68
ROC AUC: 0.85741

 SVM with kernel: sigmoid
Classification Report:
              precision    recall  f1-score   support

           0       0.76      0.83      0.79       157
           1       0.73      0.62      0.67       111

    accuracy                           0.75       268
   macro avg       0.74      0.73      0.73       268
weighted avg       0.74      0.75      0.74       268

Precision: 0.73
Recall: 0.62
ROC AUC: 0.71774
```



The best model ran with a SVM classification was the 'rbf' kernel with an ROC accuracy of .857