

Do you know about the lastest rage among gen z?

# DATA SCIENCE PROJECT

Let's continue to explore the fascinating world of  
K-dramas through the lens of data analytics

# INTRODUCTION

Today's project combines two of my passions: Data Analytics and Korean dramas, also known as K-dramas.

In this project, I have analyzed a vast amount of data related to K-dramas, and I am thrilled to share with you some appealing insights.

# SOME DRAMAS THAT ARE MOST FAMILIAR WITH THE ENTHUSIAST

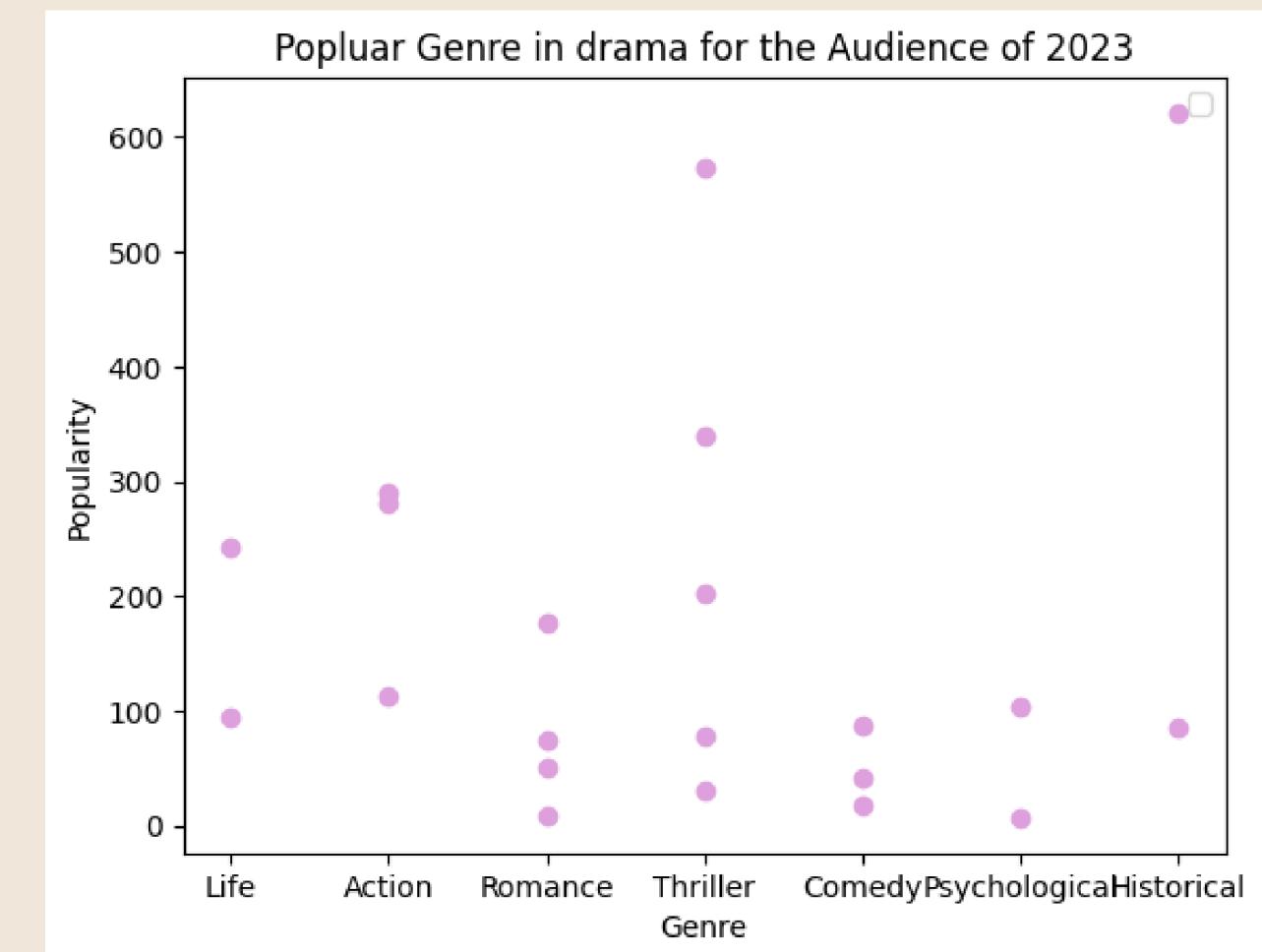
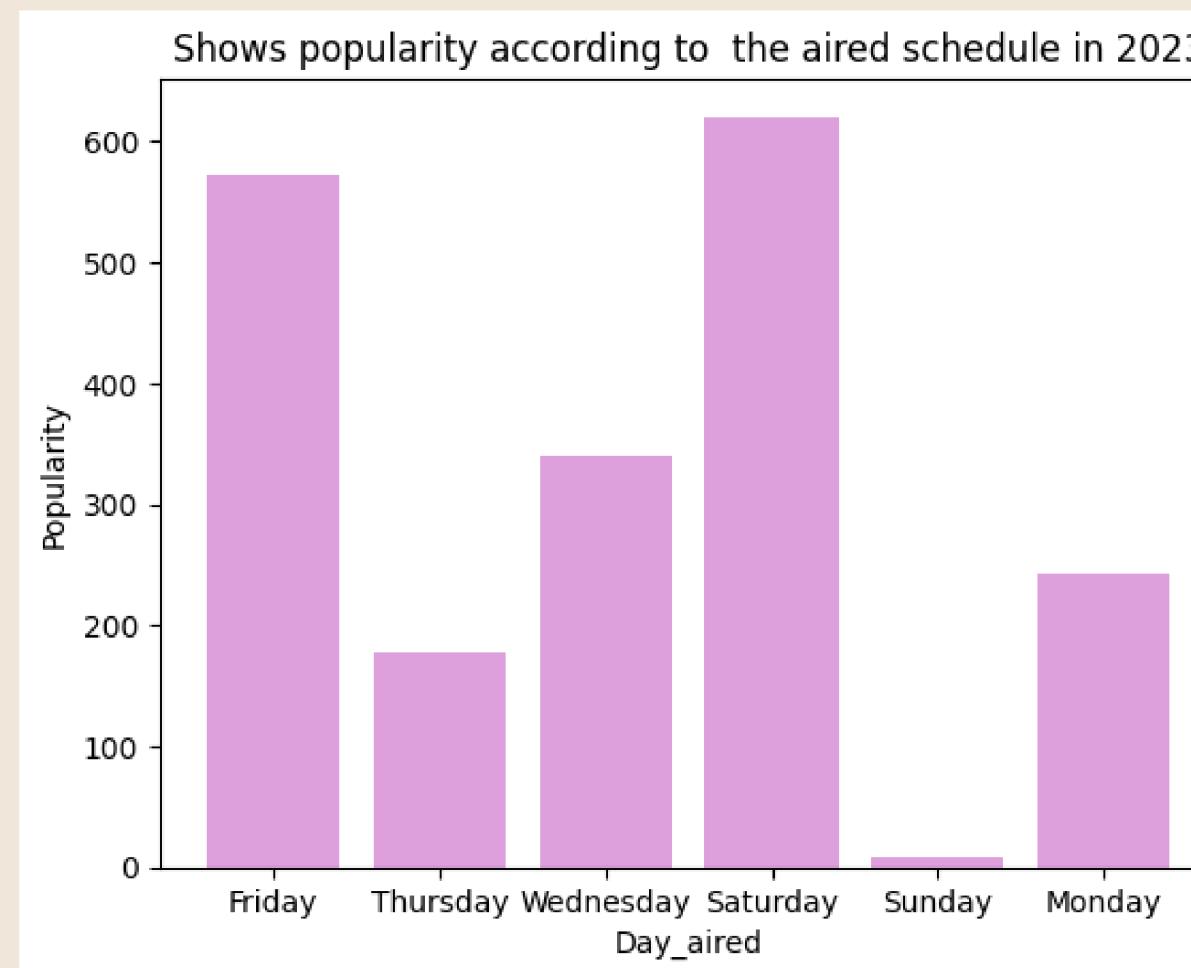


# DATA BASE

ID	Title	Genre	Episodes	Day_aired	Popularity
1	Move to Heaven	Life	10	Friday	94
2	Weak Hero Class 1	Action	8	Friday	290
3	Hospital Playlist Season 2	Romance	12	Thursday	177
4	Flower of Evil	Thriller	16	Wednesday	31
5	Hospital Playlist	Romance	12	Thursday	51
6	Alchemy of Souls	Action	20	Saturday	112
7	Reply 1988	Comedy	20	Friday	42
8	My Mister	Psychological	16	Wednesday	104
9	Under the Queen's Umbrella	Historical	16	Saturday	621
10	Prison Playbook	Comedy	16	Wednesday	88
11	Mr. Queen	Historical	20	Saturday	86
12	Alchemy of Souls Season 2: Light and Shadow	Action	10	Saturday	281
13	Mother	Thriller	16	Wednesday	340
14	Extraordinary Attorney Woo	Romance	16	Wednesday	75
15	Crash Landing on You	Romance	16	Sunday	8
16	Navillera	Life	12	Monday	242
17	Vincenzo	Comedy	20	Saturday	18
18	It's Okay to Not Be Okay	Psychological	16	Saturday	7
19	Signal	Thriller	16	Friday	79
20	The Glory	Thriller	8	Friday	202

# ★ SUB QUESTIONS ★

-Make 2 graphs on the Data Frames



# ★ SUB QUESTIONS ★

-Use any one algorithm and predict the new input

```
x = df1[['Popularity']]
y = df1['Day_aired']

dt_classifier = DecisionTreeClassifier()

dt_classifier.fit(x, y)

DecisionTreeClassifier()
DecisionTreeClassifier()

new_data = {
    'Popularity': [50]
}

new_df = pd.DataFrame(new_data)

prediction = dt_classifier.predict(new_df)

print("Prediction: ", prediction)
Prediction: ['Thursday']
```

```
x = df1[['Popularity']]
y = df1['Day_aired']

dt_classifier = DecisionTreeClassifier()

dt_classifier.fit(x, y)

DecisionTreeClassifier()
DecisionTreeClassifier()

new_data = {
    'Popularity': [600]
}

new_df = pd.DataFrame(new_data)

prediction = dt_classifier.predict(new_df)

print("Prediction: ", prediction)
Prediction: ['Saturday']
```

# ★ SUB QUESTIONS ★

-Print the Accuracy

```
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score

df = pd.get_dummies(df, columns=['Genre'])

df = pd.get_dummies(df, columns=['Day_aired'])

X = df.drop('Episodes', axis=1)
y = df['Episodes']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

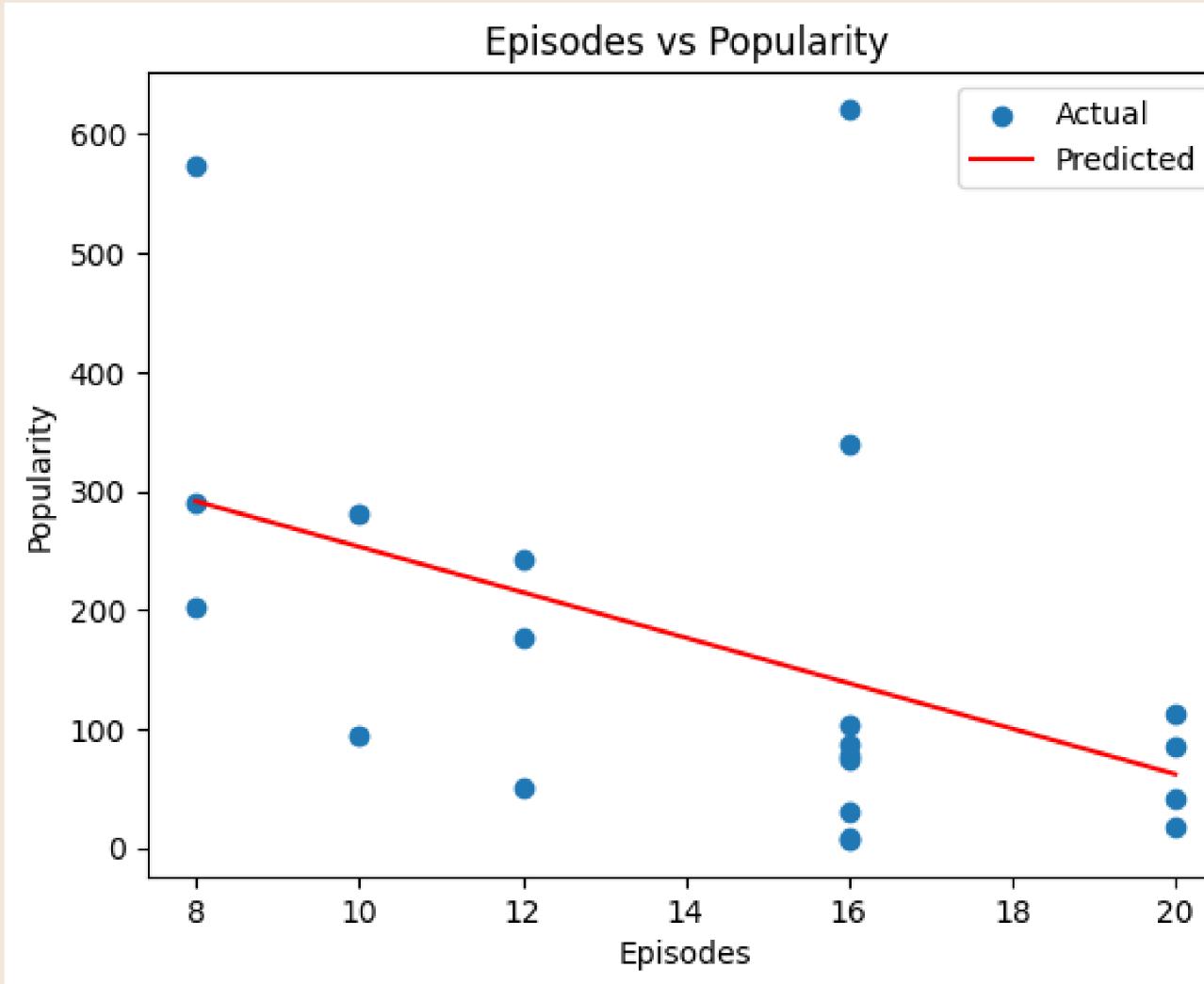
svm_classifier = SVC()
svm_classifier.fit(X_train, y_train)
y_pred = svm_classifier.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
print("Accuracy: ", accuracy)

Accuracy: 0.2
```

# ★ SUB QUESTIONS ★

-Make the predicted vs actual graph



An interesting trend: as the number of episodes increases, the popularity of a drama tends to decrease. This finding provides valuable insights into viewer preferences and engagement.



**THANK YOU  
FOR YOUR  
TIME**