Nithin Sai K S

112721214011

**Play Store App reviews**

**Final Project**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error

# Step 1: Data Exploration

apps\_df = pd.read\_csv("playstore\_apps.csv")

reviews\_df = pd.read\_csv("playstore\_apps.csv")

# Check the structure of the data

print("Apps Data:")

print(apps\_df.head())

print("\nReviews Data:")

print(reviews\_df.head())

# Check for missing values

print("Missing Values in Apps Data:")

print(apps\_df.isnull().sum())

print("\nMissing Values in Reviews Data:")

print(reviews\_df.isnull().sum())

# Check data types

print("\nData Types in Apps Data:")

print(apps\_df.dtypes)

print("\nData Types in Reviews Data:")

print(reviews\_df.dtypes)

# Step 2: Data Cleaning

# Handling missing values

apps\_df.dropna(inplace=True)

reviews\_df.dropna(inplace=True)

# Drop duplicates

apps\_df.drop\_duplicates(inplace=True)

reviews\_df.drop\_duplicates(inplace=True)

# Convert data types if necessary

# Step 3: Descriptive Analysis

# Summary statistics

print("\nSummary Statistics for Apps Data:")

print(apps\_df.describe())

print("\nSummary Statistics for Reviews Data:")

print(reviews\_df.describe())

# Step 4: Feature Engineering

# Create new features if needed

# Step 5: Exploratory Data Analysis (EDA)

# Visualize relationships between variables

plt.figure(figsize=(10, 6))

sns.scatterplot(x='score', y='thumbsUpCount', data=apps\_df)

plt.title('score vs thumbsUpCount')

plt.show()

# Step 6: Modeling (Example)

# Extract features from reviews text

tfidf = TfidfVectorizer(stop\_words='english')

X = tfidf.fit\_transform(reviews\_df['reviewCreatedVersion'])

# Split data into train and test sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, reviews\_df['score'], test\_size=0.2, random\_state=42)

# Train a simple linear regression model

model = LinearRegression()

model.fit(X\_train, y\_train)

# Predict ratings on test data

y\_pred = model.predict(X\_test)

# Evaluate model performance

mse = mean\_squared\_error(y\_test, y\_pred)

print("Mean Squared Error:", mse)

# Step 7: Insights and Recommendations

# Provide actionable insights based on analysis

# Step 8: Visualization and Reporting

# Create visualizations and prepare a report or presentation

Apps Data:

reviewId userName \

0 495266a4-f451-48c3-a844-fb3c07560d55 Foysal Hossain

1 947fcd83-7a28-403d-b03b-d0bc20f52e0e S K VERMA

2 65856211-67ba-4560-84dd-a0055775ed90 Amanuel Abara

3 cd5ba250-3a26-43b4-a378-77d18f73a503 Vagarangas X Aopi

4 e8e886b4-d6c6-416b-b0a1-be90320c4024 Shafin islam

userImage \

0 <https://play-lh.googleusercontent.com/a-/ALV-U>...

1 <https://play-lh.googleusercontent.com/a/ACg8oc>...

2 <https://play-lh.googleusercontent.com/a/ACg8oc>...

3 <https://play-lh.googleusercontent.com/a/ACg8oc>...

4 <https://play-lh.googleusercontent.com/a-/ALV-U>...

content score thumbsUpCount reviewCreatedVersion \

0 Gett van for no reason 😂😂😂 1 0 NaN

1 better' than WhatsApp 4 0 NaN

2 That was good app for me 5 0 NaN

3 Love the app 5 0 10.0.4

4 🕳️🕳️🕳️ 1 0 NaN

at replyContent repliedAt appVersion userLang \

0 9/19/2023 15:05 NaN NaN NaN EN

1 9/19/2023 14:59 NaN NaN NaN EN

2 9/19/2023 14:55 NaN NaN NaN EN

3 9/19/2023 14:50 NaN NaN 10.0.4 EN

4 9/19/2023 14:48 NaN NaN NaN EN

app\_id

0 org.telegram.messenger

1 org.telegram.messenger

2 org.telegram.messenger

3 org.telegram.messenger

4 org.telegram.messenger

Reviews Data:

reviewId userName \

0 495266a4-f451-48c3-a844-fb3c07560d55 Foysal Hossain

1 947fcd83-7a28-403d-b03b-d0bc20f52e0e S K VERMA

2 65856211-67ba-4560-84dd-a0055775ed90 Amanuel Abara

3 cd5ba250-3a26-43b4-a378-77d18f73a503 Vagarangas X Aopi

4 e8e886b4-d6c6-416b-b0a1-be90320c4024 Shafin islam

userImage \

0 <https://play-lh.googleusercontent.com/a-/ALV-U>...

1 <https://play-lh.googleusercontent.com/a/ACg8oc>...

2 <https://play-lh.googleusercontent.com/a/ACg8oc>...

3 <https://play-lh.googleusercontent.com/a/ACg8oc>...

4 <https://play-lh.googleusercontent.com/a-/ALV-U>...

content score thumbsUpCount reviewcreated \

0 Gett van for no reason 😂😂😂 1 0 NaN

1 better' than WhatsApp 4 0 NaN

2 That was good app for me 5 0 NaN

3 Love the app 5 0 10.0.4

4 🕳️🕳️🕳️ 1 0 NaN

at replyContent repliedAt appVersion userLang \

0 9/19/2023 15:05 NaN NaN NaN EN

1 9/19/2023 14:59 NaN NaN NaN EN

2 9/19/2023 14:55 NaN NaN NaN EN

3 9/19/2023 14:50 NaN NaN 10.0.4 EN

4 9/19/2023 14:48 NaN NaN NaN EN

app\_id

0 org.telegram.messenger

1 org.telegram.messenger

2 org.telegram.messenger

3 org.telegram.messenger

4 org.telegram.messenger

Missing Values in Apps Data:

reviewId 0

userName 0

userImage 0

content 0

score 0

thumbsUpCount 0

reviewCreatedVersion 1604

at 0

replyContent 5809

repliedAt 5809

appVersion 1604

userLang 0

app\_id 0

dtype: int64

Missing Values in Reviews Data:

reviewId 0

userName 0

userImage 0

content 0

score 0

thumbsUpCount 0

reviewCreatedVersion 1604

at 0

replyContent 5809

repliedAt 5809

appVersion 1604

userLang 0

app\_id 0

dtype: int64

Data Types in Apps Data:

reviewId object

userName object

userImage object

content object

score int64

thumbsUpCount int64

reviewCreatedVersion object

at object

replyContent object

repliedAt object

appVersion object

userLang object

app\_id object

dtype: object

Data Types in Reviews Data:

reviewId object

userName object

userImage object

content object

score int64

thumbsUpCount int64

reviewCreatedVersion object

at object

replyContent object

repliedAt object

appVersion object

userLang object

app\_id object

dtype: object

Summary Statistics for Apps Data:

score thumbsUpCount

count 163.000000 163.000000

mean 2.466258 7.374233

std 1.599700 27.228817

min 1.000000 0.000000

25% 1.000000 0.000000

50% 2.000000 0.000000

75% 4.000000 2.500000

max 5.000000 271.000000

Summary Statistics for Reviews Data:

score thumbsUpCount

count 163.000000 163.000000

mean 2.466258 7.374233

std 1.599700 27.228817

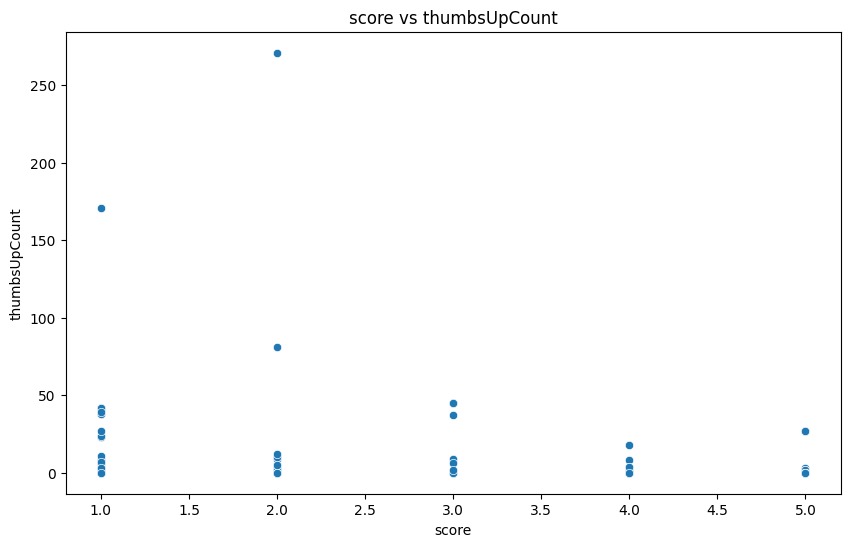
min 1.000000 0.000000

25% 1.000000 0.000000

50% 2.000000 0.000000

75% 4.000000 2.500000

max 5.000000 271.000000



Mean Squared Error: 2.1454661597701032