#read the dataset

pd.read\_csv("https://raw.githubusercontent.com/sunnysavita10/Statistics-With-Python-TheC ompleteGuide/main/clead\_google\_playstore\_data")

#check the dupliocate

df[df.duplicated()]

df=df.drop\_duplicates(subset=['App'],keep='first')

#segrate the feaature

numeric\_features = [feature for feature in df.columns if df[feature].dtype != 'O'] categorical\_features = [feature for feature in df.columns if df[feature].dtype == 'O']

#value counts

df["Type"].value\_counts(normalize=True)\*100

for col in categorical\_features:

print(f"{col}:{df[col].value\_counts(normalize=True)\*100}")

print("=============================================")

#category = [ 'Type', 'Content Rating'] wise count plot

# categorical columns

plt.figure(figsize=(20, 15))

plt.suptitle('Univariate Analysis of Categorical Features', fontsize=20, fontweight='bold', alpha=0.8, y=1.)

category = [ 'Type', 'Content Rating']

for i in range(0, len(category)):

plt.subplot(2, 2, i+1)

sns.countplot(x=df[category[i]],palette="Set2")

plt.xlabel(category[i])

plt.xticks(rotation=45)

plt.tight\_layout()

**plt.figure(figsize=(15, 15))**

**plt.suptitle('Univariate Analysis of Numerical Features', fontsize=20, fontweight='bold', alpha=0.8, y=1.)**

**for i in range(0, len(numeric\_features)):**

**plt.subplot(5, 3, i+1)**

**sns.kdeplot(x=df[numeric\_features[i]],shade=True, color='r')**

**plt.xlabel(numeric\_features[i])**

**plt.tight\_layout()**

**df["Category"].value\_counts().plot.pie(y=df["Category"],figsize=(15,15),autopct='%1.1f %%')**

**category = pd.DataFrame(df['Category'].value\_counts()) #Dataframe of apps on the basis of category**

**category.rename(columns = {'Category':'Count'},inplace=True)**

**plt.figure(figsize=(15,6))**

**sns.barplot(x=df\_cat, y ='Count',data = category[:10],palette='hls') plt.title('Top 10 App categories')**

**plt.xticks(rotation=90)**

**plt.show()**

**df\_cat\_installs=df.groupby(['Category'])['Installs'].sum().sort\_values(ascending=False ).reset\_index()**

**plt.figure(figsize = (14,10))**

**sns.set\_context("talk")**

**sns.set\_style("darkgrid")**

**ax = sns.barplot(x = 'Installs' , y = 'Category' , data = df2 )**

**ax.set\_xlabel('No. of Installations in Billions')**

**ax.set\_ylabel('')**

**ax.set\_title("Most Popular Categories in Play Store", size = 20)**

**### What are the Top 5 most installed Apps in Each popular Categories ?? dfa=df.groupby(['Category','App'])['Installs'].sum().reset\_index()**

**apps = ['GAME', 'COMMUNICATION', 'PRODUCTIVITY', 'SOCIAL' ] sns.set\_context("poster")**

**sns.set\_style("darkgrid")**

**plt.figure(figsize=(40,30))**

**for i,app in enumerate(apps):**

**df2 = dfa[dfa.Category == app]**

**df3 = df2.head(5)**

**plt.subplot(4,2,i+1)**

**sns.barplot(data= df3,x= 'Installs' ,y='App' )**

**plt.xlabel('Installation in Millions')**

**plt.ylabel('')**

**plt.title(app,size = 20)**

**plt.tight\_layout()**

**plt.subplots\_adjust(hspace= .3)**

**plt.show()**

**rating=df.groupby(["App"])["Rating"].sum().sort\_values(ascending=False).reset\_index ()**

**rating[rating.Rating==5.0]**

**rating2 = df.groupby(['Category','Installs',**

**'App'])['Rating'].sum().sort\_values(ascending = False).reset\_index() df.groupby("Category").agg({"Installs":"sum","Reviews":"sum"}).reset\_index() df.groupby(['Category','App'])["Reviews"].sum().reset\_index()**

**df.groupby("Category").agg({"Installs":"sum","Reviews":"sum"}).reset\_index()**

**fig, ax = plt.subplots(1,2,figsize=(20,7))**

**df.value\_counts('Type').plot.pie(y='Type',startangle=90, explode=(0.2,0), title='Percentage of the Free App and Paid App', legend=False, autopct='%.2f', ax=ax[0])**

**ax[0].set(ylabel='Type of Apps')**

**df.groupby('Type').agg({'Installs':sum}).plot.pie(y='Installs', startangle=90, explode=(0.2,0), title='Percentage of Installs Number for Free App and Paid App', legend=False, autopct='%.2f', ax=ax[1])**

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

**sns.boxplot(x="Installs",y="Rating",data=df)**

**plt.xticks(size=15,rotation=90)**

**plt.show()**

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

**sns.boxplot(x="Category",y="Rating",data=df)**

**plt.xticks(size=15,rotation=90)**

**plt.show()**

**plt.figure(figsize=(15,15))**

**plt.xticks(rotation = 90)**

**sns.barplot('Category','Size',data=df)**

**plt.show()**

**plt.figure(figsize=(15,8))**

**sns.countplot(x='Rating',data = df,palette="Set1\_r")**

**plt.xticks(rotation =90)**

**plt.title('Countplot for ratings')**

**plt.show()**

**plt.subplots(figsize=(20,10))**

**freq= pd.Series()**

**freq=df['year'].value\_counts()**

**freq.plot()**

**plt.xlabel("Dates")**

**plt.ylabel("Number of updates")**

**plt.title("Time series plot of Last Updates")**

**https://drive.google.com/drive/folders/1B-LeKjHsew1C6GUciWrr5y5XLlv5X\_OG**

**Dataset:**

**https://archive.ics.uci.edu/ml/datasets/Phishing+Websites**

**https://archive.ics.uci.edu/ml/datasets/Census+Income**

**Do the proper EDA and cleaning it is mandatory**

**Feature engineering is optional**

**Submit form:**

**https://forms.gle/7JoFujsHhNzc6sW18**

**Before next saturday you need to submit the task**