ml-ssignment-2-2203a51813

March 11, 2024

[ ]:

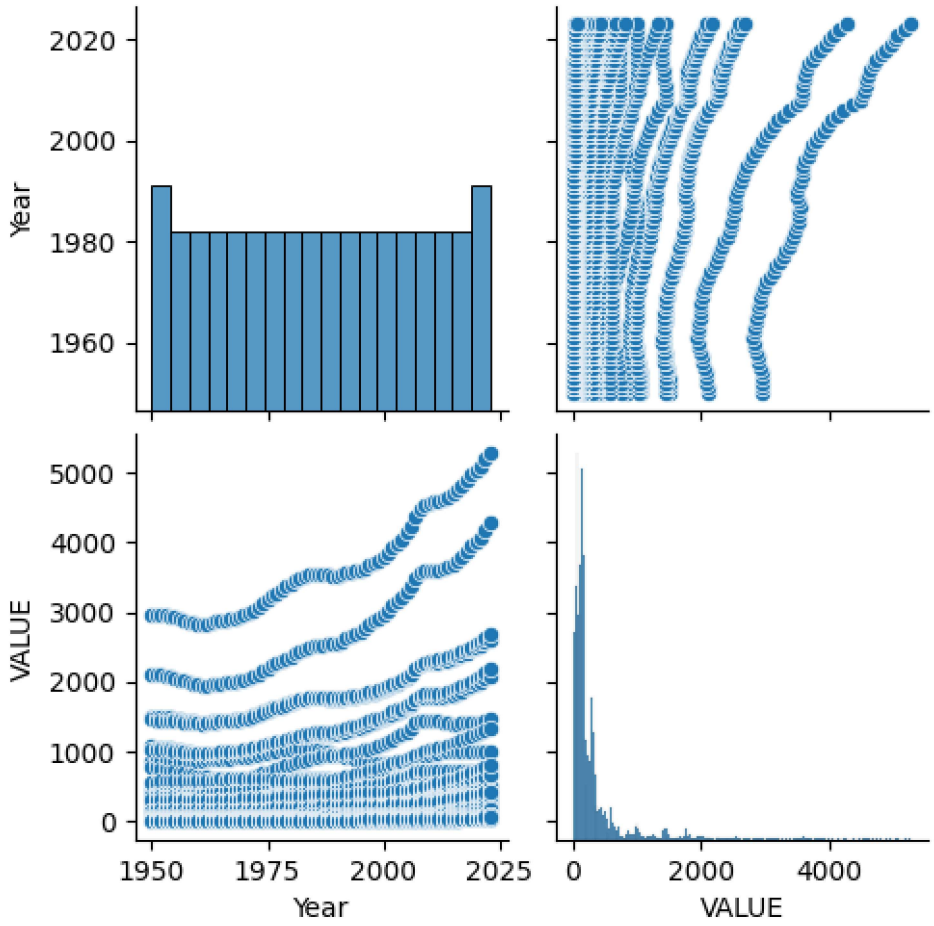
[9]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
df = pd.read_csv("/Population Estimates -1950-2023 project.csv")

print("Data Types of Each Feature:")
print(df.dtypes)
print("\nSummary of the Dataset:")
print(df.describe(include='all'))
# Step 4: List the names of columns/features in the dataset
print("\nColumn Names:")
print(df.columns)
# Step 5: Perform Exploratory analysis # Plot numeric features
numeric_features = df.select_dtypes(include=['int64', 'float64']).columns
sns.pairplot(df[numeric_features])
plt.show()

# Check relative size of survived/unsurvived sns.countplot(x='survived',␣
 ↪data=df) plt.title('Survival Count')
plt.show()

# Check if any pattern on gender sns.countplot(x='YEAR', hue='VALUE', data=df)␣
 ↪plt.title('Survival Count by Gender') plt.show()
# Passenger class
sns.countplot(x='Year', hue='VALUE', data=df)
plt.title('growth in population')
plt.show()

# YEAR WISE POPULATION
class_survival_rate = df.groupby('Year')['VALUE'].mean()
print("\npopulation mean::")

print(class_survival_rate)
# AGE GROUPS POPULATION
sns.countplot(x='Age Group', hue='VALUE', data=df)
```

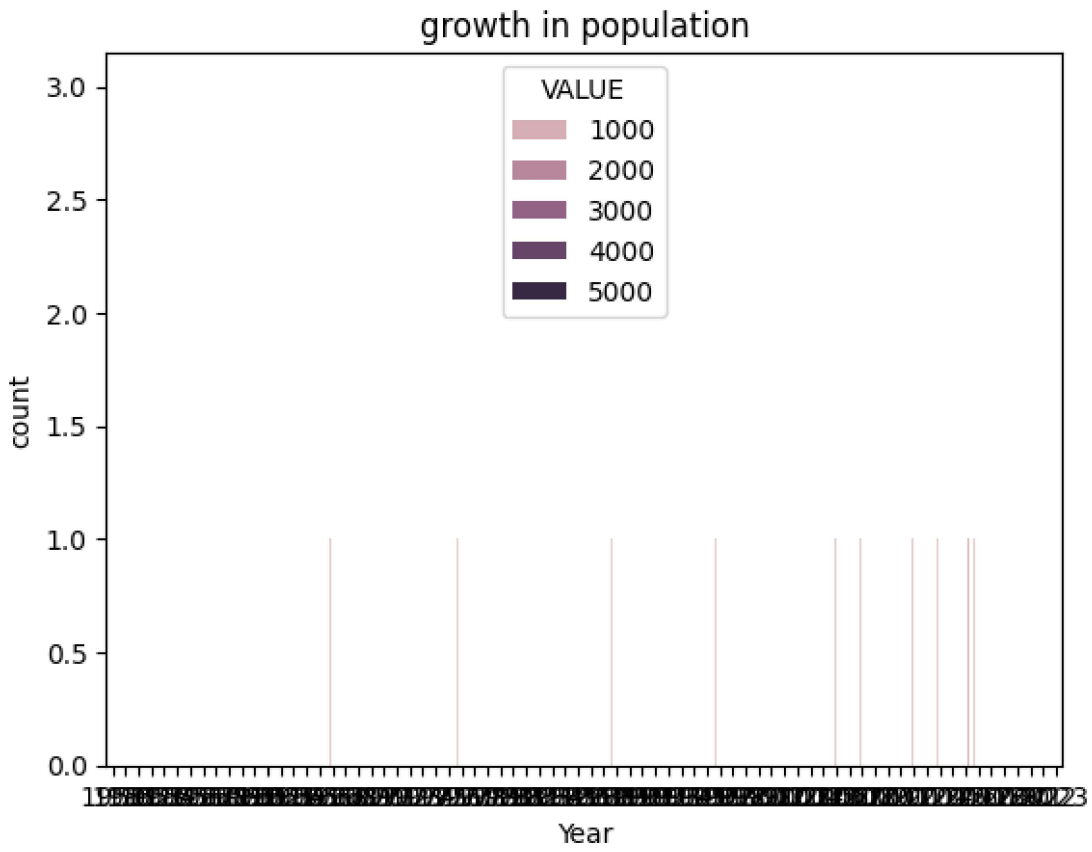
```
plt.title('age groups population')
plt.show()
# Overall age distribution sns.histplot(df['age'].dropna(), kde=True) plt.
 ↪title('Overall Age Distribution')

plt.xlabel('Age Group]')
plt.show()
# Class-wise age distribution plt.figure(figsize=(10, 6)) sns.
 ↪boxplot(x='pclass', y='age', data=df) plt.title('Age Distribution by␣
 ↪Passenger Class') plt.show()
# Step 6: Data wrangling # Impute age data
median_age = df['Age Group'].median()
df['Age Group'].fillna(median_age, inplace=True)
# Drop unnecessary features
columns_to_drop = ['PassengerId', 'Name', 'Ticket', 'Cabin']
existing_columns = df.columns
columns_to_drop = [col for col in columns_to_drop if col in existing_columns]
# Remove columns not present in the DataFrame
df.drop(columns=columns_to_drop, axis=1, inplace=True)

# Recode categorical features
df['sex'] = df['sex'].map({'male': 0, 'female': 1})
df = pd.get_dummies(df, columns=['embarked'])
# Display the updated dataframe print("\nUpdated DataFrame:") print(df.head())
```

```
Data Types of Each Feature:
STATISTIC Label     object
Year                 int64
Age Group           object
Sex                 object
UNIT                object
VALUE              float64
dtype: object

Summary of the Dataset:
                                STATISTIC Label         Year     Age Group  \
count                                      5994  5994.000000          5994
unique                                        1          NaN            27
top     Population Estimates (Persons in April)          NaN  Under 1 year
freq                                       5994          NaN           222
mean                                        NaN  1986.500000           NaN
std                                         NaN    21.361791           NaN
min                                         NaN  1950.000000           NaN
25%                                         NaN  1968.000000           NaN
50%                                         NaN  1986.500000           NaN
75%                                         NaN  2005.000000           NaN
max                                         NaN  2023.000000           NaN
```

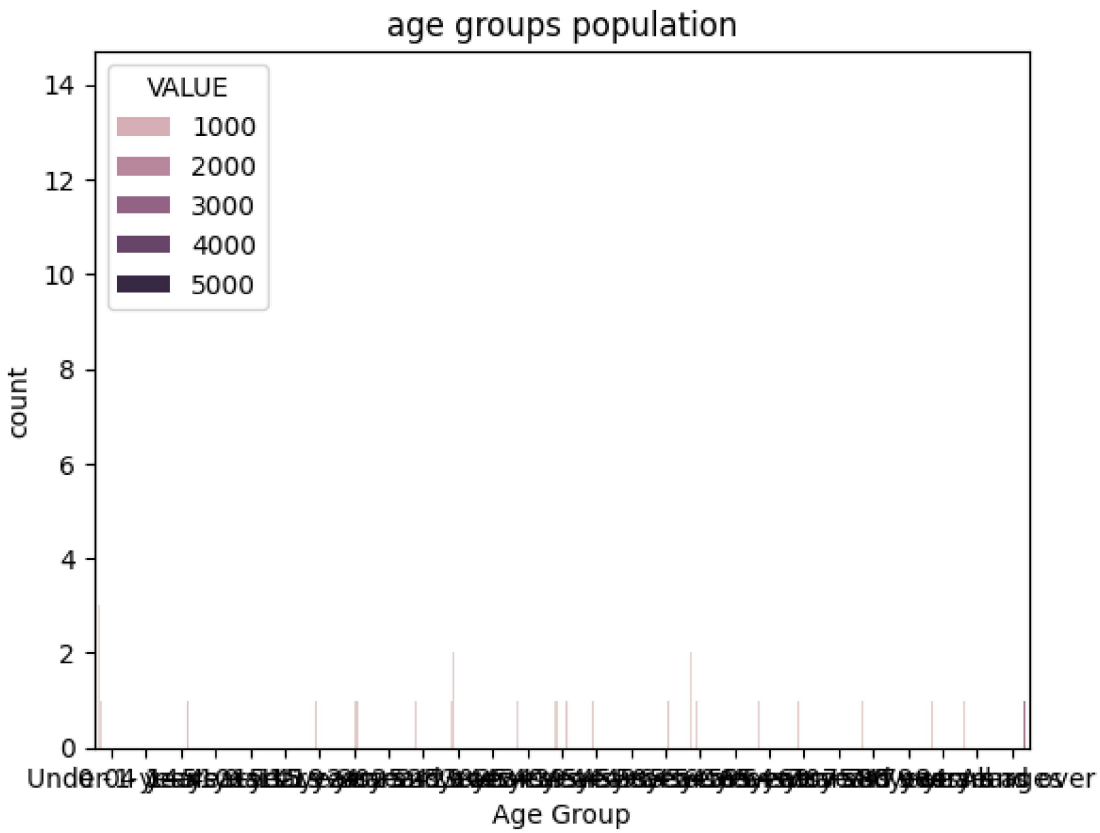
	Sex	UNIT	VALUE
count	5994	5994	5880.000000
unique	3	1	NaN
top	Both sexes	Thousand	NaN
freq	1998	5994	NaN
mean	NaN	NaN	347.001088
std	NaN	NaN	601.822682
min	NaN	NaN	5.500000
25%	NaN	NaN	79.475000
50%	NaN	NaN	151.700000
75%	NaN	NaN	307.725000
max	NaN	NaN	5281.600000

```
Column Names:
Index(['STATISTIC Label', 'Year', 'Age Group', 'Sex', 'UNIT', 'VALUE'],
dtype='object')
```






```
population mean::
Year
1950    282.687179
1951    279.013580
1952    280.820513
1953    280.274359
1954    279.353846
           …
2019    472.425926
2020    479.371605
2021    483.755556
2022    494.254321
2023    503.909877
Name: VALUE, Length: 74, dtype: float64
```


age groups population
VALUE
1000
2000
3000
4000
5000
count
0
2
4
6
8
10
12
14
Age Group