POPULATION PREDICTION

Presenters:- Ritik Bhola

Shanmukhi Balagamsetty

Chukwuma Blessing

INTRODUCTION

- There has been tremendous growth in the size of the world's population in the last half century. Global population was around 3 billion in 1960. By 1987, in less than three decades, it had surpassed 5 billion and there were around 7.6 billion people in the world in 2018.
- This growth varies greatly across regions. Since 1960, the largest relative growth has taken place in Sub-Saharan Africa where the population expanded from 227 million in 1960 to more than 1 billion in 2018—a nearly fivefold increase. The second largest growth over the period can be seen in Middle East and North Africa, where the population increased more than 4 times, from 105 million to 449 million. So we need to build a model with higher accuracy to predict population growth of year 2050.

OBJECTIVES

- Predict population of year 2050 on the basics of previous years population.
- Build model with higher accuracy.

DATASET

```
[ ] #showing first 5 rows
    df.head()
```

	Country Name	Country Code	Series Name	Series Code	1960 [YR1960]	1961 [YR1961]	1962 [YR1962]	1963 [YR1963]	1964 [YR1964]	1965 [YR1965]	1966 [YR1966]	1967 [YR1967]	1968 [YR1968]	1969 [YR1969]	[YI
0	Afghanistan	AFG	Age dependency ratio (% of working-age populat...	SP.POP.DPND	81.717739	82.755907	83.304573	83.550740	83.734450	83.962159	85.099283	86.103695	86.970697	87.650154	88.
1	Afghanistan	AFG	Age dependency ratio, old	SP.POP.DPND.OL	5.086262	5.132609	5.139524	5.111895	5.056182	4.976304	5.024428	5.046187	5.039599	5.002078	4.
2	Afghanistan	AFG	Age dependency ratio, young	SP.POP.DPND.YG	76.631477	77.623299	78.165049	78.438845	78.678267	78.985855	80.074854	81.057508	81.931099	82.648076	83.
3	Afghanistan	AFG	Age population, age 0, female, interpolated	SP.POP.AG00.FE.IN	178556.000000	182674.000000	185434.000000	187513.000000	189945.000000	193553.000000	197672.000000	202860.000000	208967.000000	215766.000000	222986.
4	Afghanistan	AFG	Age population, age 0, male, interpolated	SP.POP.AG00.MA.IN	177635.000000	181588.000000	185852.000000	190462.000000	195498.000000	201014.000000	205918.000000	211276.000000	217164.000000	223692.000000	230854.

5 rows × 95 columns

CATEGORICAL COLUMNS:

```
# Print all columns with data type object
print(df.columns.values[df.dtypes=='object'])
```

```
['Country Name' 'Country Code' 'Series Name' 'Series Code']
```

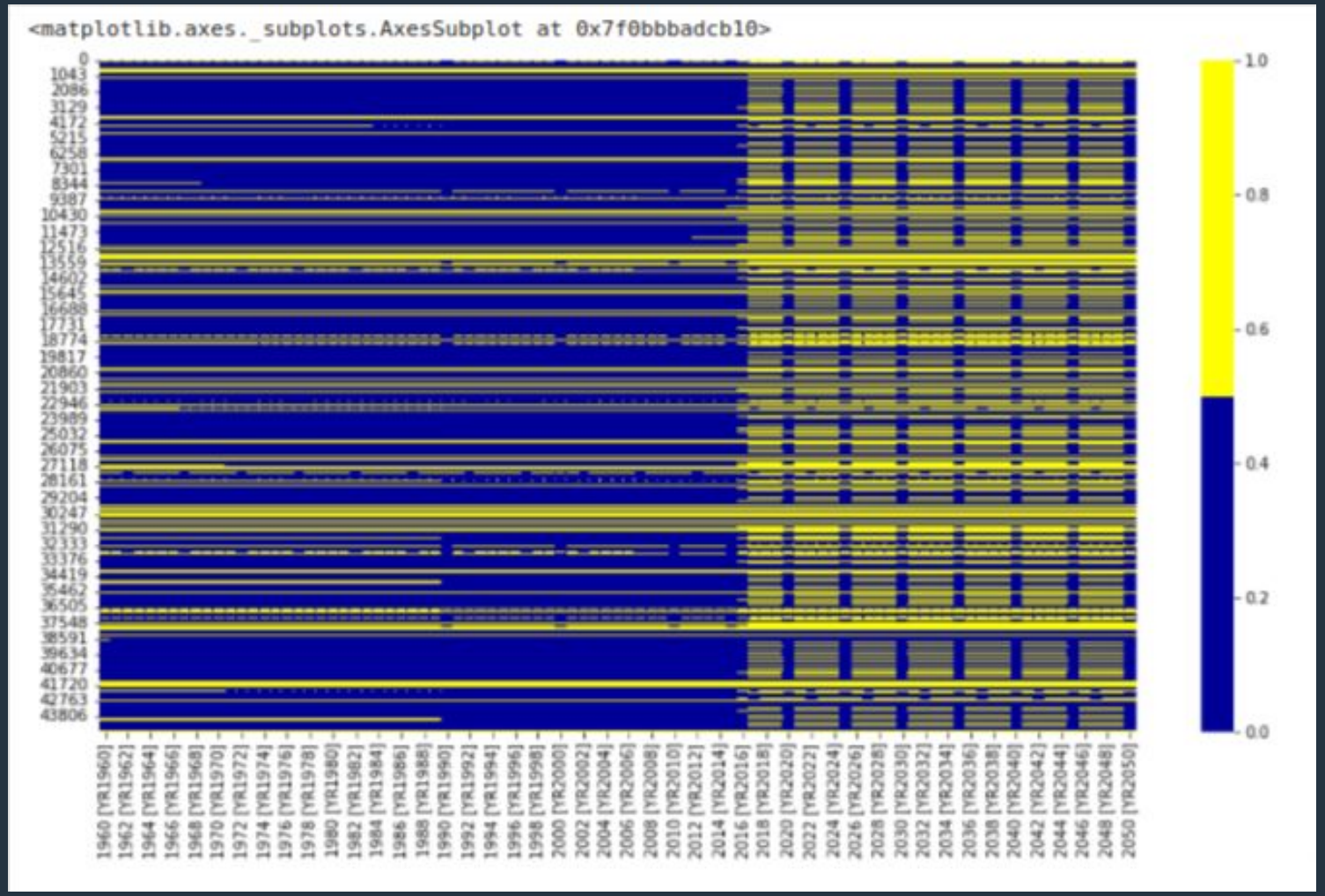
NUMERICAL COLUMNS:

```
['1960 [YR1960]' '1961 [YR1961]' '1962 [YR1962]' '1963 [YR1963]'
 '1964 [YR1964]' '1965 [YR1965]' '1966 [YR1966]' '1967 [YR1967]'
 '1968 [YR1968]' '1969 [YR1969]' '1970 [YR1970]' '1971 [YR1971]'
 '1972 [YR1972]' '1973 [YR1973]' '1974 [YR1974]' '1975 [YR1975]'
 '1976 [YR1976]' '1977 [YR1977]' '1978 [YR1978]' '1979 [YR1979]'
 '1980 [YR1980]' '1981 [YR1981]' '1982 [YR1982]' '1983 [YR1983]'
 '1984 [YR1984]' '1985 [YR1985]' '1986 [YR1986]' '1987 [YR1987]'
 '1988 [YR1988]' '1989 [YR1989]' '1990 [YR1990]' '1991 [YR1991]'
 '1992 [YR1992]' '1993 [YR1993]' '1994 [YR1994]' '1995 [YR1995]'
 '1996 [YR1996]' '1997 [YR1997]' '1998 [YR1998]' '1999 [YR1999]'
 '2000 [YR2000]' '2001 [YR2001]' '2002 [YR2002]' '2003 [YR2003]'
 '2004 [YR2004]' '2005 [YR2005]' '2006 [YR2006]' '2007 [YR2007]'
 '2008 [YR2008]' '2009 [YR2009]' '2010 [YR2010]' '2011 [YR2011]'
 '2012 [YR2012]' '2013 [YR2013]' '2014 [YR2014]' '2015 [YR2015]'
 '2016 [YR2016]' '2017 [YR2017]' '2018 [YR2018]' '2019 [YR2019]'
 '2020 [YR2020]' '2021 [YR2021]' '2022 [YR2022]' '2023 [YR2023]'
 '2024 [YR2024]' '2025 [YR2025]' '2026 [YR2026]' '2027 [YR2027]'
 '2028 [YR2028]' '2029 [YR2029]' '2030 [YR2030]' '2031 [YR2031]'
 '2032 [YR2032]' '2033 [YR2033]' '2034 [YR2034]' '2035 [YR2035]'
 '2036 [YR2036]' '2037 [YR2037]' '2038 [YR2038]' '2039 [YR2039]'
 '2040 [YR2040]' '2041 [YR2041]' '2042 [YR2042]' '2043 [YR2043]'
 '2044 [YR2044]' '2045 [YR2045]' '2046 [YR2046]' '2047 [YR2047]'
 '2048 [YR2048]' '2049 [YR2049]' '2050 [YR2050]']
```

MISSING VALUES IN DATASET:

```
Country Name              3
Country Code              5
Series Name               5
Series Code               5
1960 [YR1960]          8962
                        ...
2046 [YR2046]         21753
2047 [YR2047]         20880
2048 [YR2048]         21753
2049 [YR2049]         21753
2050 [YR2050]         10307
Length: 95, dtype: int64
```

MISSING VALUES HEATMAP:



CHANGING NULL VALUES WITH MEAN | MEDIAN | ZERO:

-With median

```
[ ]  for col in numeric_cols:
         df[col] = df[col].fillna(df[col].median())
```

-with mean

```
[ ]  df2 = pd.read_csv('/content/drive/MyDrive/Datasets/data.csv'
```

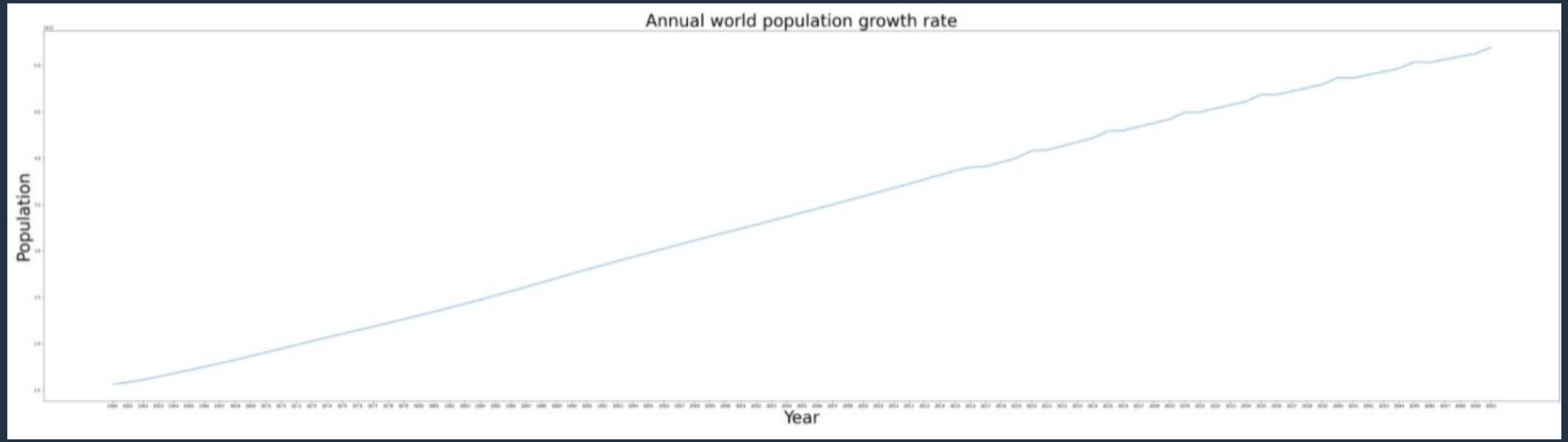
```
[ ]  for col in numeric_cols:
         df2[col] = df2[col].fillna(df[col].mean())
```

-with zero

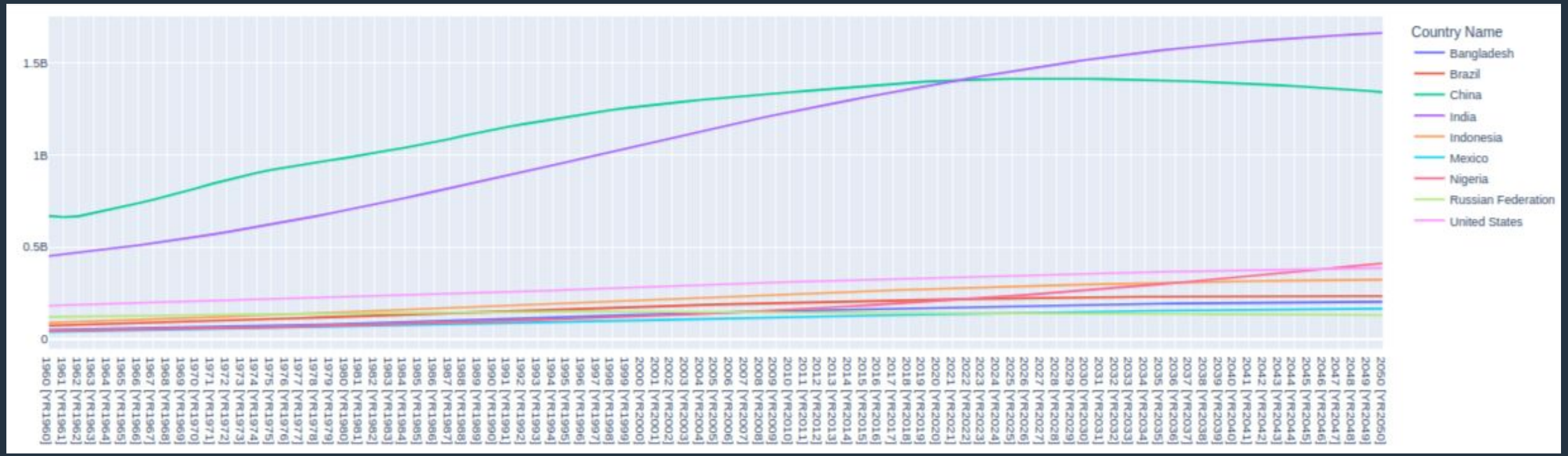
```
[ ]  df3 = pd.read_csv('/content/drive/MyDrive/Datasets/data.csv'
```

```
[ ]  for col in numeric_cols:
         df3[col] = df3[col].fillna(df[col].median())
```

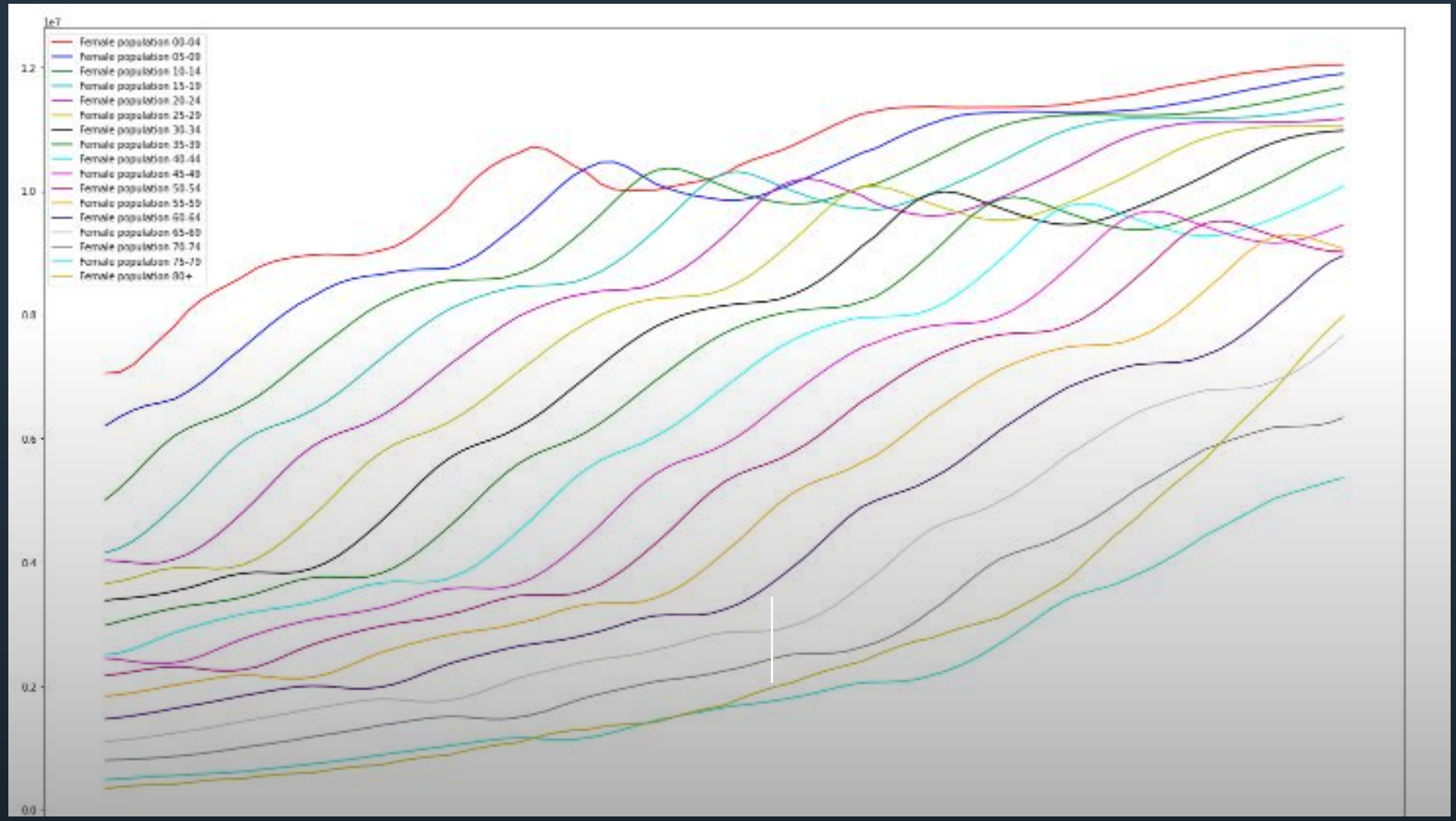
ANNUAL WORLD POPULATION GROWTH RATE



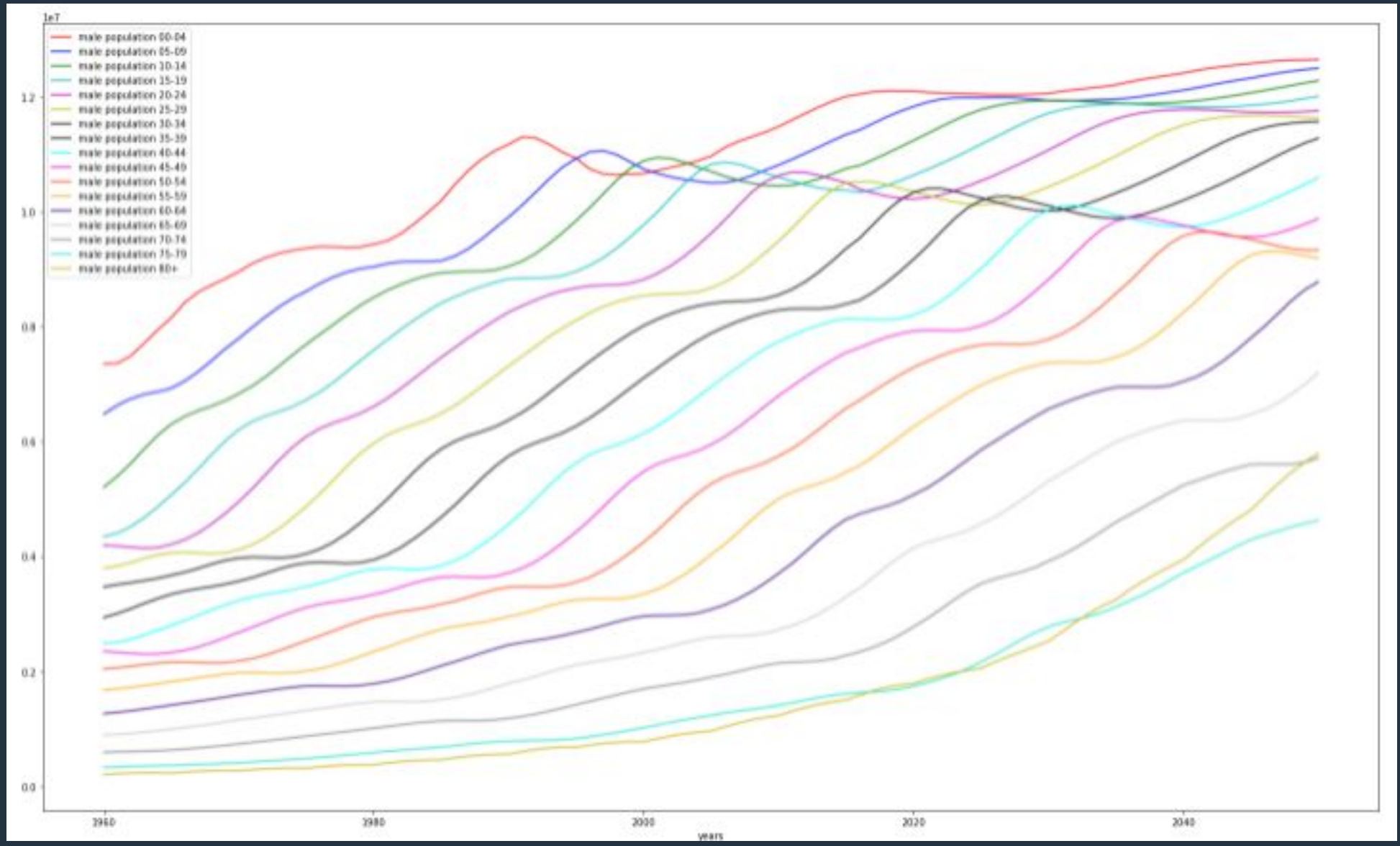
POPULATION GROWTH OF SOME SELECTED COUNTRIES



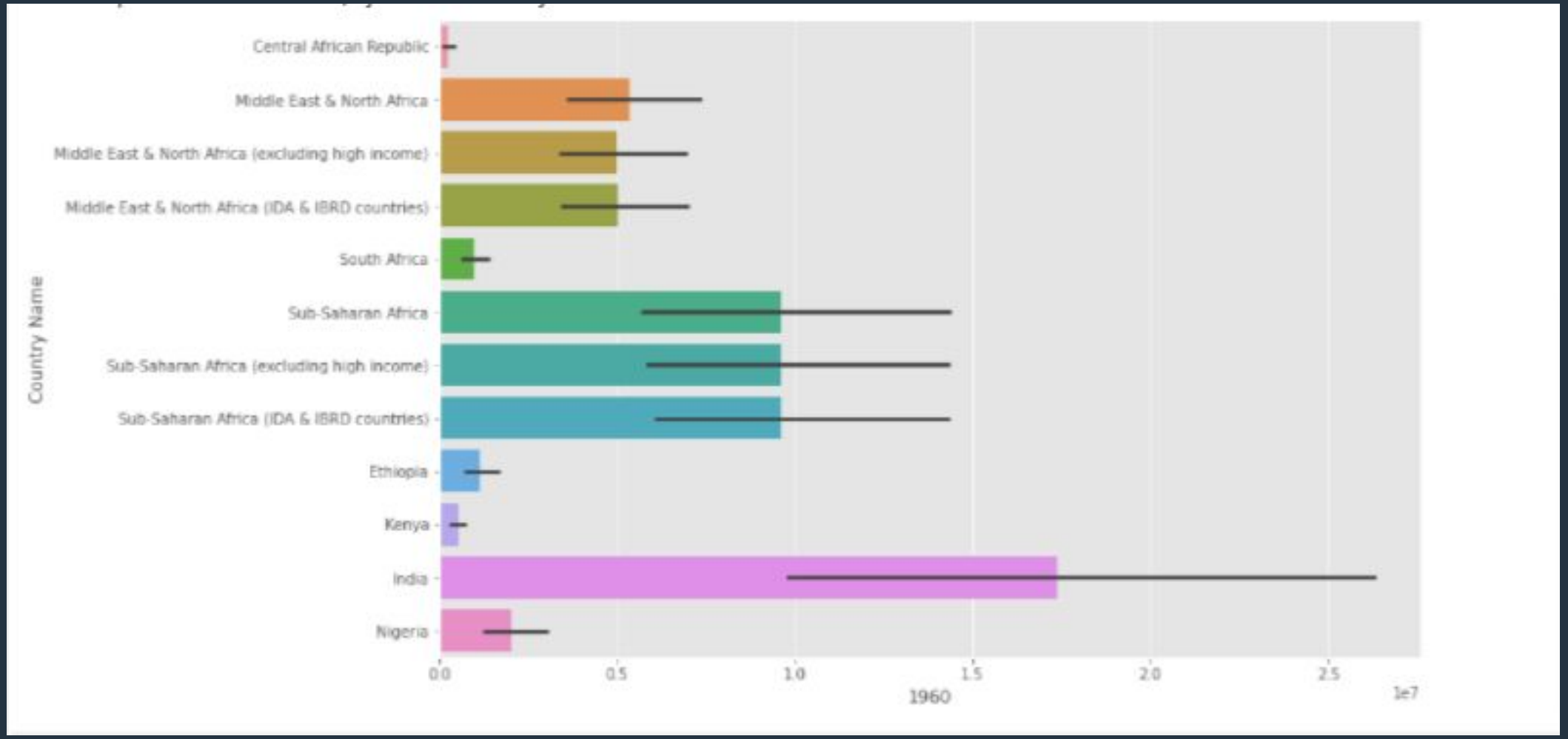
FEMALE POPULATION GROWTH ON THE BASICS OF THEIR AGE GROUPS:



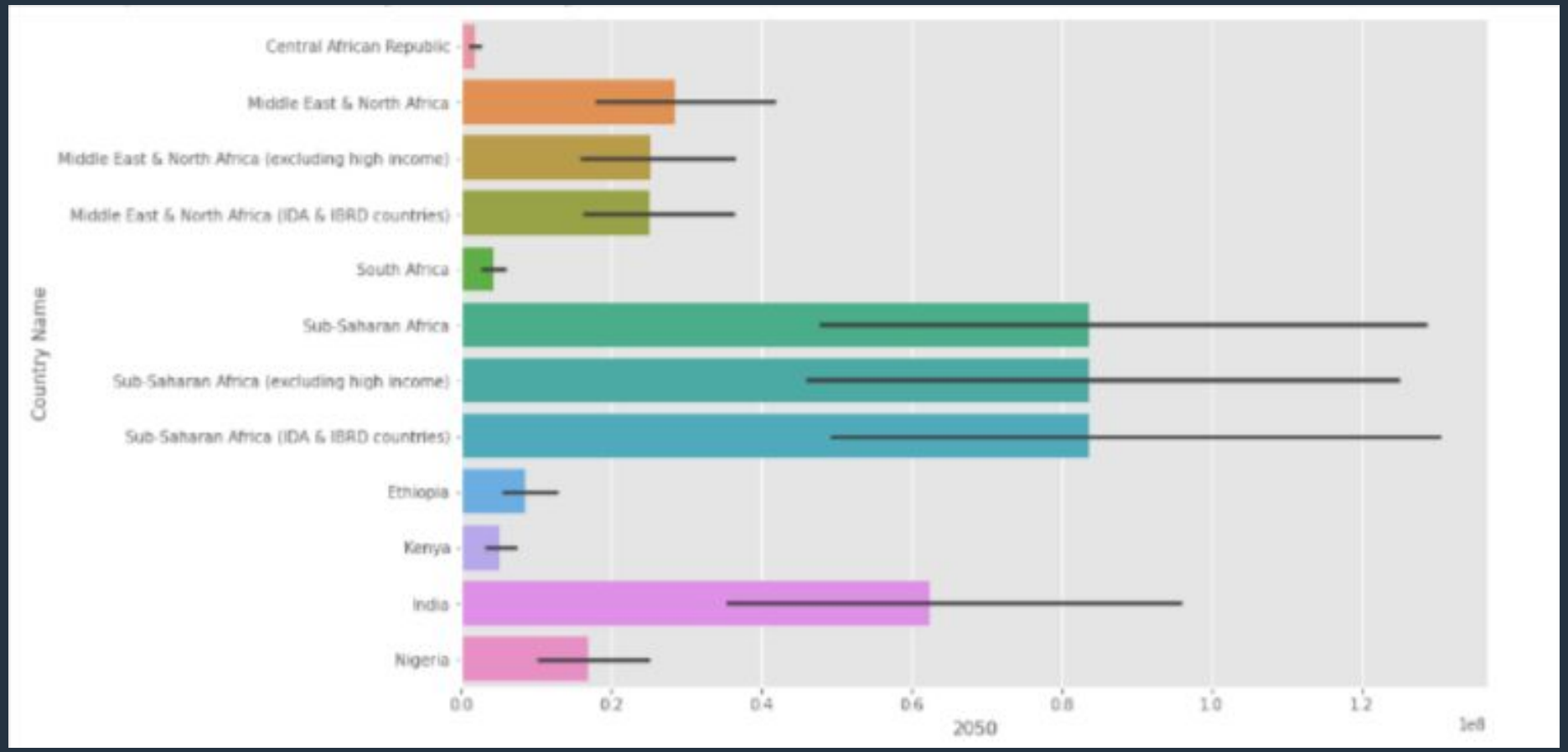
MALE POPULATION GROWTH ON THE BASICS OF THEIR AGE GROUPS:



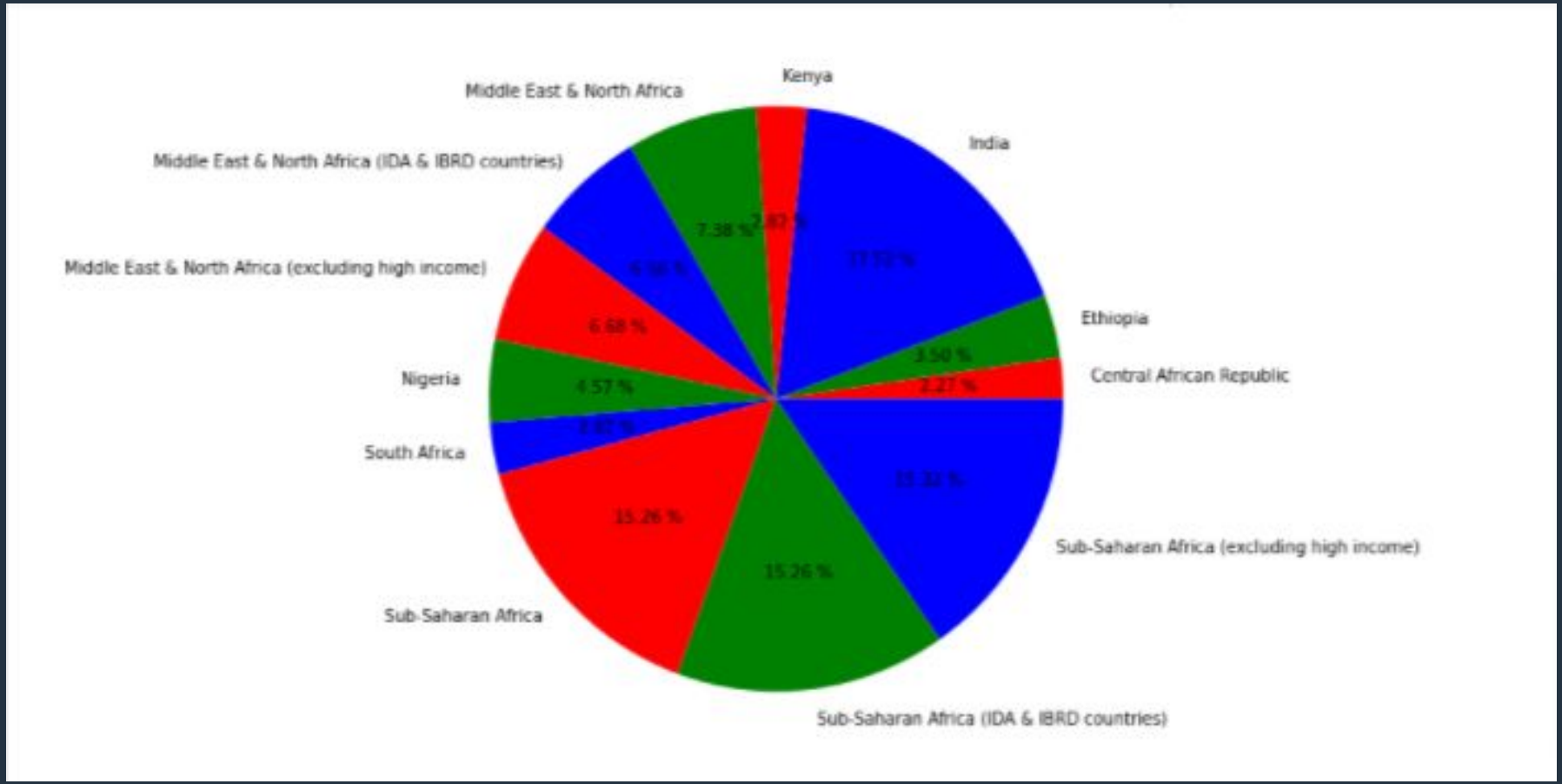
WORLD POPULATION BY SELECTED REGIONS IN 1960:



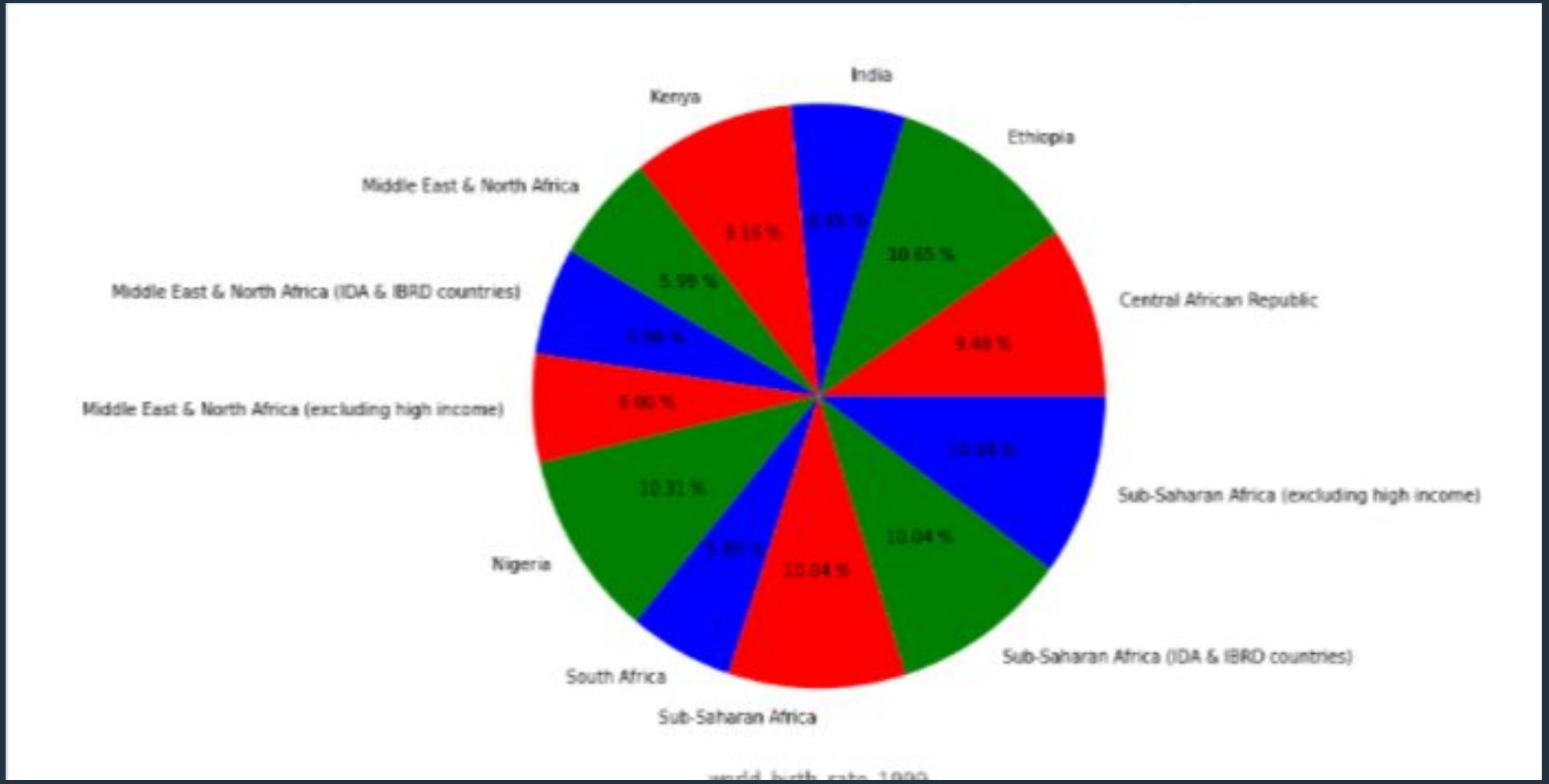
WORLD POPULATION BY SELECTED REGIONS IN 2050:



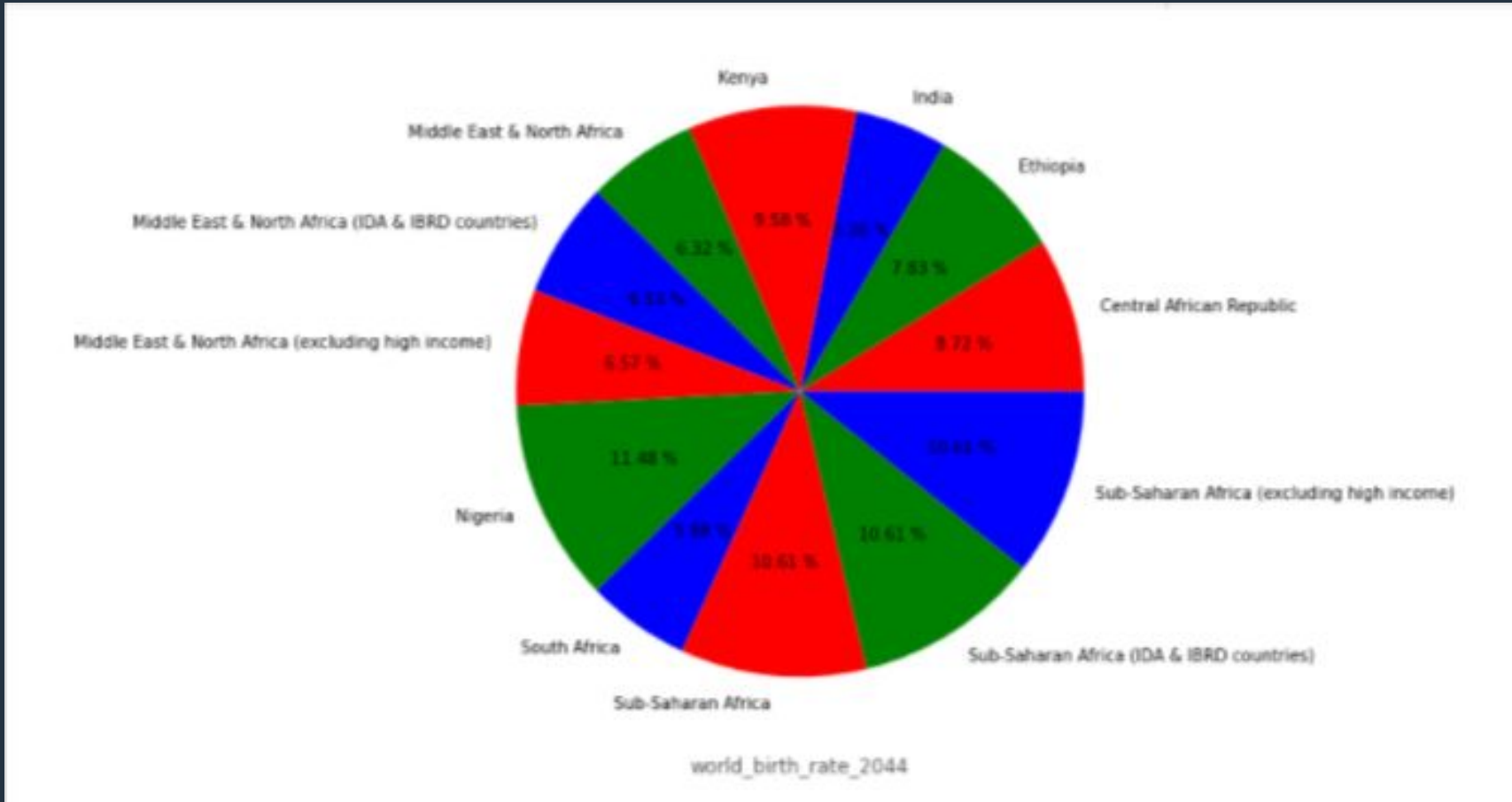
PERCENTAGES OF THE REGIONAL POPULATION IN 2022



BIRTH RATE AMONG REGIONS IN 1999:



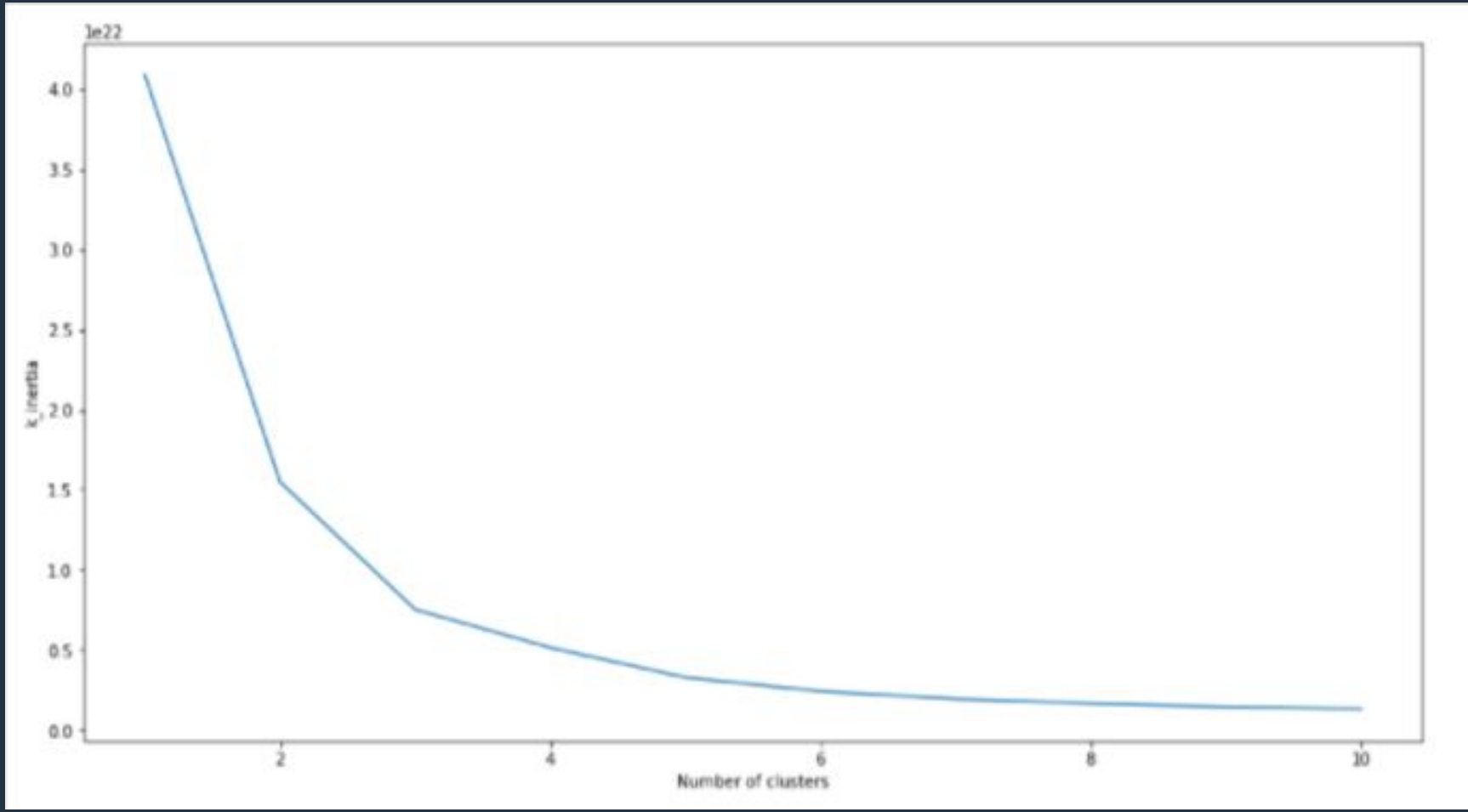
BIRTH RATE AMONG REGIONS IN 1999:



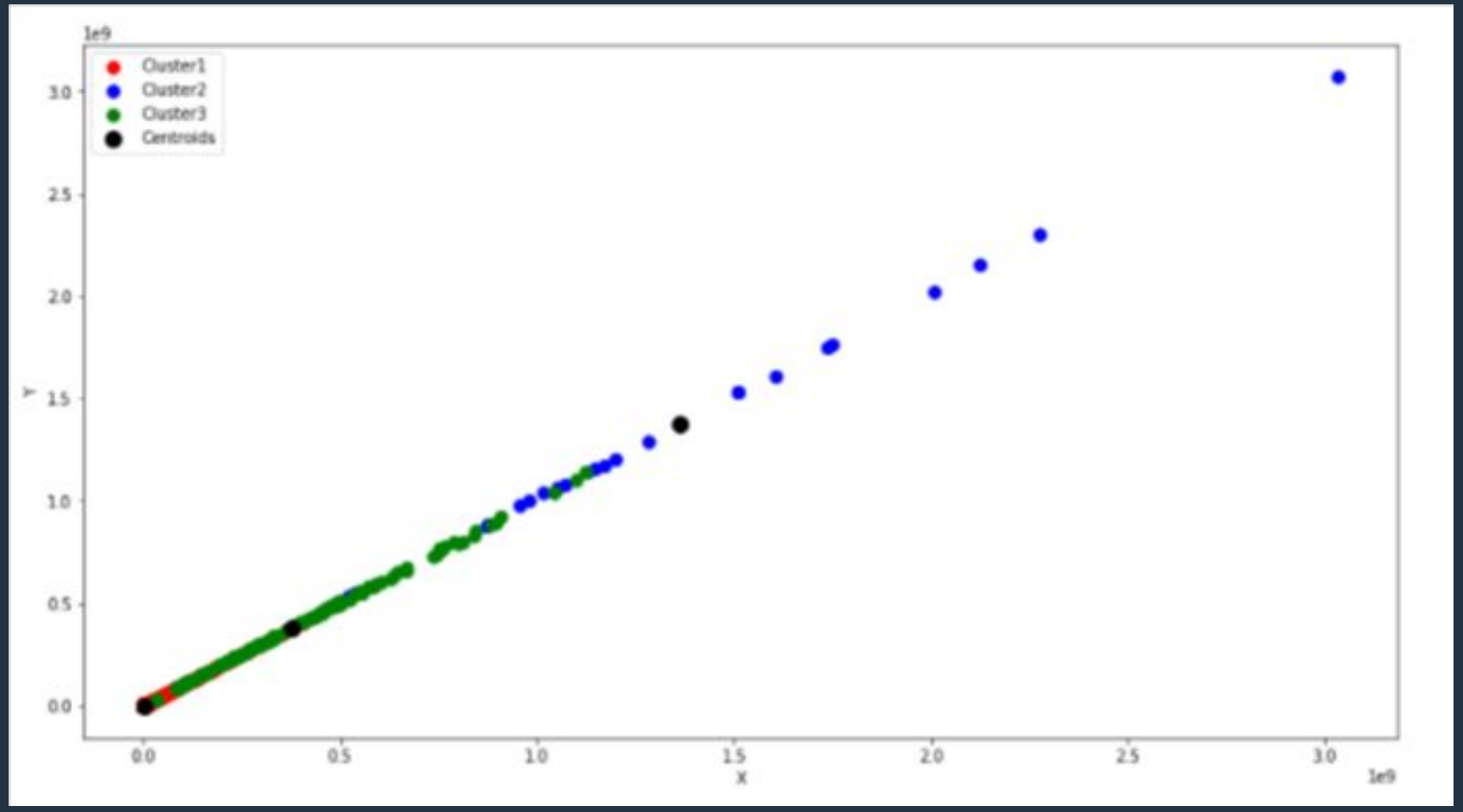
MODEL:

```
from sklearn.cluster import KMeans    #Importing KMeans from sklearn and
k_inertia = []                        # Creating a list to store the kmeans.inertia_
for k in range(1,11):
    kmeans = KMeans(n_clusters=k, init='k-means++')
    kmeans.fit(X1)
    k_inertia.append(kmeans.inertia_)
```

ELBOW METHOD:



CLUSTERS:


1e9
Cluster1
Cluster2
Cluster3
Centroids
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Y
0.0
0.5
1.0
1.5
2.0
2.5
3.0
X
1e9

thank you!