pop_prediction

February 4, 2022

1 Applied Machine Learning (CS 5394) Lab #1

Names: Jiayi Zhu, Alex Glover, Yuanchun Zhao

In this notebook, we will be using 60 years of historical data to predict the world population in 2122.

Business Case: Our team is attempting to develop a prediction model that will be able to estimate the world population for many years to come, specifically the year 2122 for this assignment. Population prediction is an important field that has been relied on to provide approximate but useful information about the future. The target market for these prediction models are policymakers, researchers, government officials, program planners, ect. Our model's intended use includes but is not limited to helping forecast future demographic characteristics, future economy and economic trends, and the demands of the future population. Also, knowing the size of the population can be very useful when making decisions that will affect our world's resources. For example, with an increase in population also comes an increase in the demand of housing. A good population estimate could help accurately plan land usage and housing for a given year.

Dataset: The dataset we will use in this notebook is a population dataset from the World Bank. It includes total population of the world for the last 60 years as well as total population for the time range broken down by country.

[45]:
```
# import libraries
import pandas as pd
import numpy as np
from sklearn.linear_model import LinearRegression
import re
import json
```

[46]:
```
import warnings
warnings.filterwarnings("ignore")
```

[47]:
```
# load in data from csv file
# the dataset we will be using is from the World Bank (https://data.worldbank.
 ↪org/indicator/SP.POP.TOTL?most_recent_year_desc=true)

url = 'https://raw.githubusercontent.com/fluffyjiayi/applied_ml/main/
 ↪populations.csv'
df = pd.read_csv(url, index_col=False)
```

```
df.head()
```

[47]:

	Country Name	Country Code	…	2019	2020
0	Aruba	ABW	…	106310.0	106766.0
1	Africa Eastern and Southern	AFE	…	660046272.0	677243299.0
2	Afghanistan	AFG	…	38041757.0	38928341.0
3	Africa Western and Central	AFW	…	446911598.0	458803476.0
4	Angola	AGO	…	31825299.0	32866268.0

[5 rows x 65 columns]

[48]:
```
# drop columns that aren't helpful

world_df = df.loc[df['Country Name']=='World']
world_df.drop(['Country Name','Country Code', 'Indicator Name', 'Indicator Code'],axis=1,inplace=True)

# reformat data for linear regression
world_df = world_df.transpose()
world_df.dropna(inplace=True)
world_df=world_df.reset_index().rename(columns={259:'population','index':'year'})
world_df.head()
```

[48]:

	year	population
0	1960	3.032156e+09
1	1961	3.071596e+09
2	1962	3.124561e+09
3	1963	3.189656e+09
4	1964	3.255146e+09

[49]:
```
world_df.describe()
```

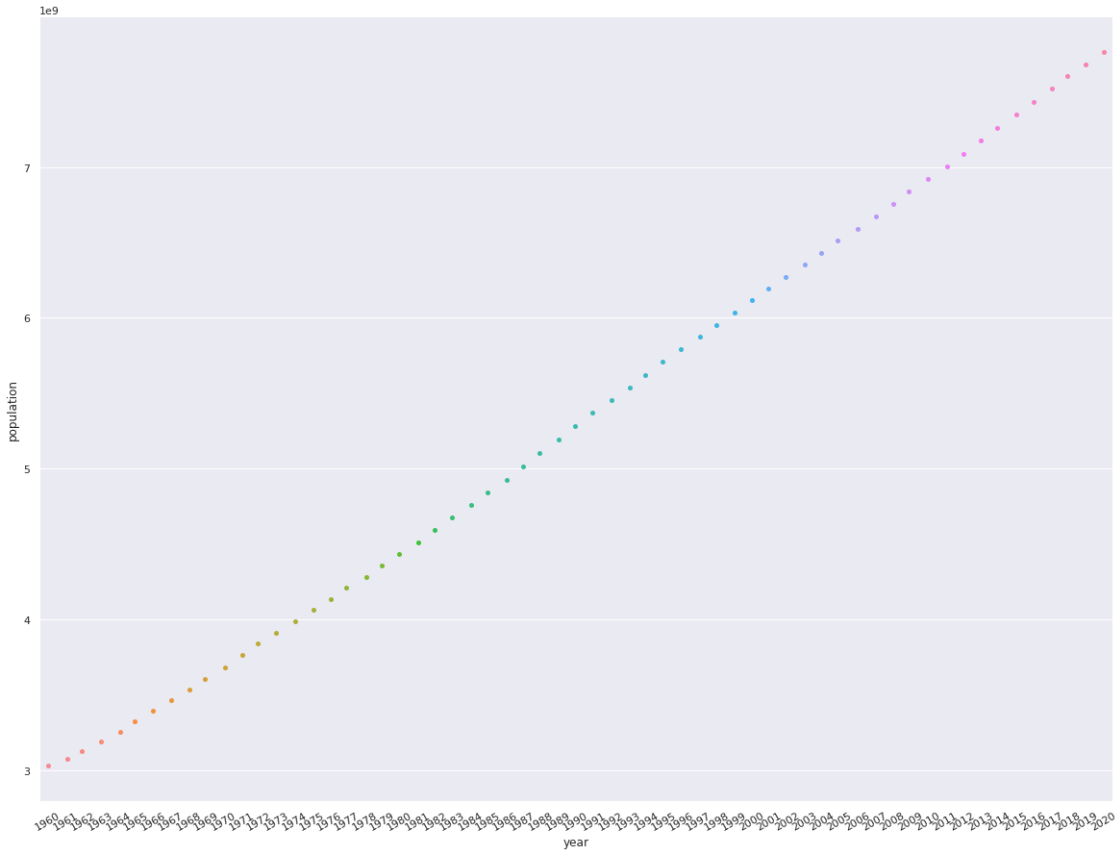
[49]:

	population
count	6.100000e+01
mean	5.301718e+09
std	1.433209e+09
min	3.032156e+09
25%	4.062507e+09
50%	5.280063e+09
75%	6.511725e+09
max	7.761620e+09

[50]:
```
import matplotlib.pyplot as plt
import seaborn as sns

# visualize world population data
```

```
sns.set(palette="muted",color_codes=True)
plt.figure(figsize=(20,15))
sns.
 ↪stripplot(x=world_df['year'],y=world_df['population'],data=world_df,jitter=True)
plt.xticks(rotation=30)
```

[50]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
        17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
        34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50,
        51, 52, 53, 54, 55, 56, 57, 58, 59, 60]),
 <a list of 61 Text major ticklabel objects>)
```



Prediction Method: Linear Regression

Linear regression is a simple algorithm that will assume a linear relationship between population and year. It will fit a linear equation to observed data and we can use that equation to predict future years.

We decided to use linear regression to predict the population because it is a method that is easy to understand and that is compatible with the data we have. We ran linear regression on our historical world population and found that the population projected for 2122 was around 15.9 billion, more than doubling between 2020 and 2122. Given that world population growth rate is on the decline,

we figured that this prediction was not reasonable, so we started looking into options to make our projection more realistic.

We decided that instead of using linear regression to predict total world population, we would use it to predict growth rate instead. This should help us capture the trend of slowing population growth in recent years. Using our projected growth rates, we calculated total population for each future year until we reached year 2122 where the population is projected to be about 9.2 billion.

After taking a closer look at our projected growth rates, we found that annual growth rate started to become negative after year 2080. While some studies show that population in certain countries will begin contracting in the next 100 years, we do not foresee the entire world population declining within that time. Instead, we expect growth to level out, so in order to reflect this, we made one more adjustment to our method: when the projected growth rates started to become negative, we set them to the last positive rate as though growth became constant after that point. When we used this method, we got an estimate of about 10.7 billion by 2122. We think this is our best estimation.

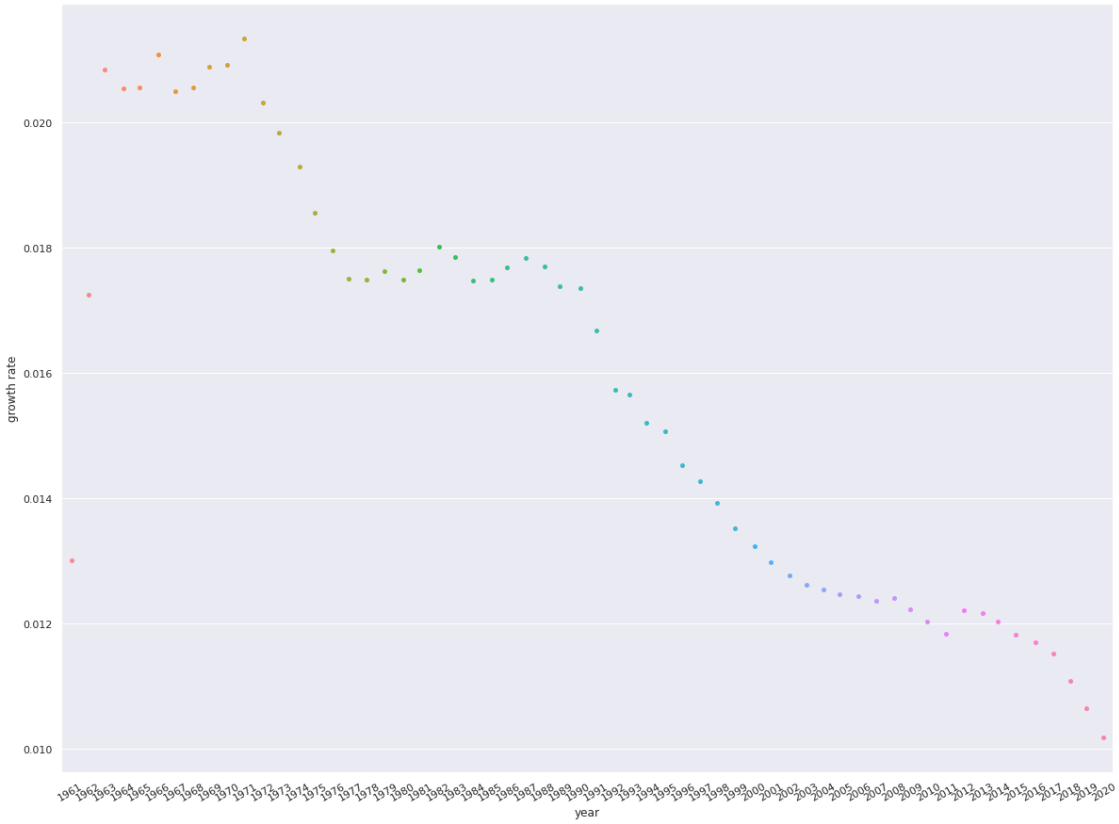
[51]:
```
# calculate historical growth rates
growth_rates = world_df['population'].pct_change()
frames = [world_df['year'], growth_rates]
final = pd.concat(frames, axis=1)
final.dropna(inplace=True)
final = final.rename(columns={'year': 'year', 'population': 'growth rate'})
final.head()
```

[51]:

	year	growth rate
1	1961	0.013007
2	1962	0.017243
3	1963	0.020833
4	1964	0.020532
5	1965	0.020552

[52]:
```
# visualize historical growth rates

sns.set(palette="muted",color_codes=True)
plt.figure(figsize=(20,15))
sns.stripplot(x=final['year'],y=final['growth rate'],data=final,jitter=True)
plt.xticks(rotation=30)
```

[52]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
        17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
        34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50,
        51, 52, 53, 54, 55, 56, 57, 58, 59]),
 <a list of 60 Text major ticklabel objects>)
```



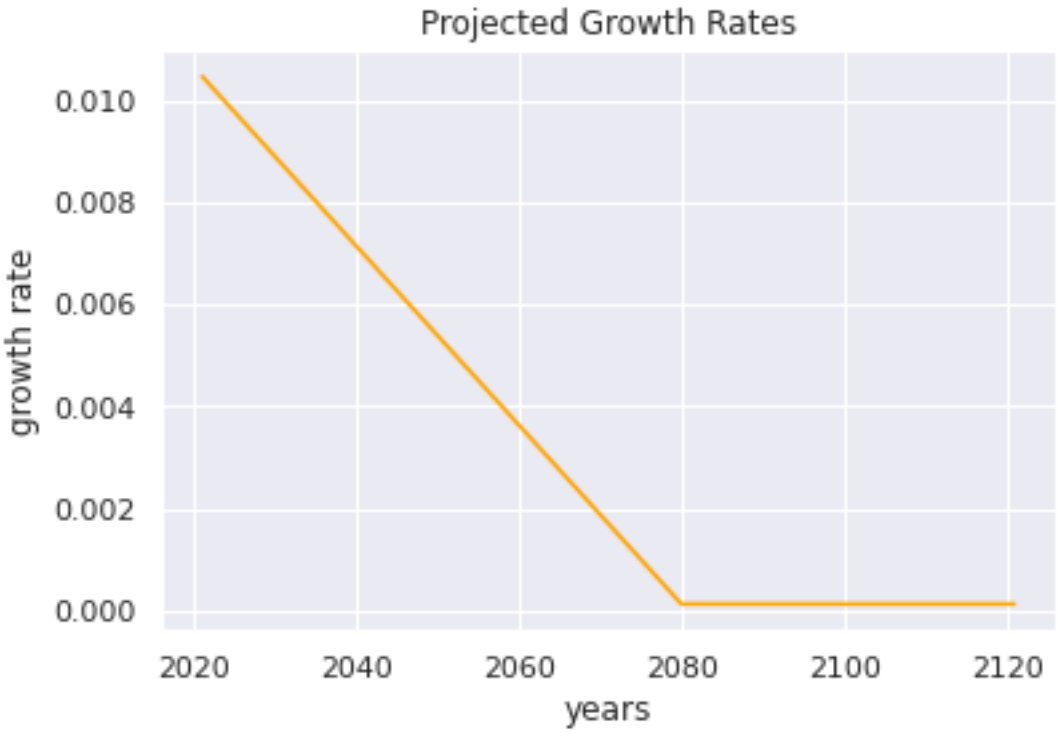
[53]:
```
# use linear regression to predict 2122 population

x = final.iloc[:, 0].values.reshape(-1, 1)
y = final.iloc[:, 1].values.reshape(-1, 1)
model = LinearRegression().fit(x, y)

projected_rates = []
projected_populations = []
total_population = world_df.iat[60,1] # retrieve most recent population we have
previous = 0

for x in range(2021, 2122):
  y_pred = model.predict([[x]])
  if y_pred < 0:
    y_pred = previous
  else:
    previous = y_pred
  projected_rates.append(y_pred)
  total_population = total_population * (1 + y_pred)
  projected_populations.append(total_population)
```

```
print('The projected world population for 2122 is', total_population)
```

```
The projected world population for 2122 is [[1.0713417e+10]]
```

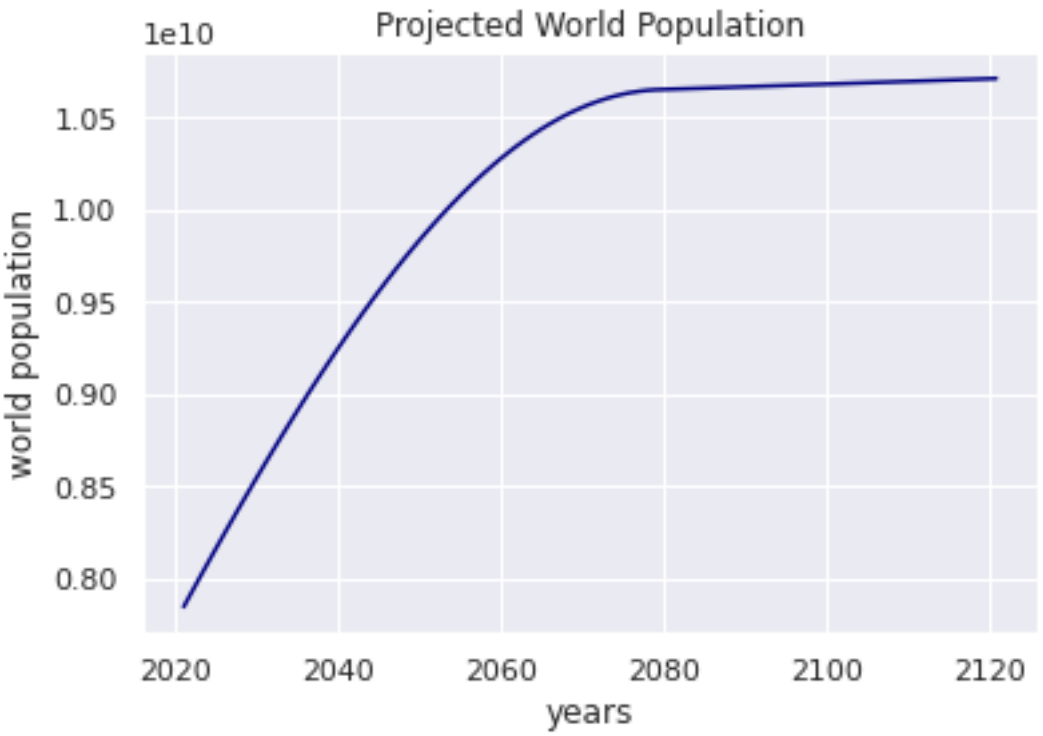
[56]:
```
# visualize projections
projected_rates = np.array(projected_rates)
projected_rates = np.squeeze(projected_rates, axis=1)
future_years = np.arange(start=2021, stop=2122, step=1)
plt.plot(future_years, projected_rates, color='orange')
plt.title('Projected Growth Rates')
plt.xlabel('years')
plt.ylabel('growth rate')
```

[56]:
```
Text(0, 0.5, 'growth rate')
```



[58]:
```
projected_populations = np.array(projected_populations)
projected_populations = np.squeeze(projected_populations, axis=1)
plt.plot(future_years, projected_populations, color='navy')
plt.title('Projected World Population')
plt.xlabel('years')
plt.ylabel('world population')
```

[58]:
```
Text(0, 0.5, 'world population')
```


Projected World Population
1e10
1.05
1.00
0.95
0.90
0.85
0.80
world population
2020
2040
2060
2080
2100
2120
years