Project: Demographics

Introduction

For most of human history, fertility rates have remained high—well above the 2.1 children per woman needed for replacement. Families were large. This is because, in times where demanding agricultural work was the norm, so too was childhood mortality. The more children a family had, the more likely it was that one or more would survive. However, as education and technology have increased, an interesting demographic phenomenon has emerged. The growth of education and technology has coincided with a decline in family sizes, with fertility rates well below the replacement rate. For example, highly developed countries like Japan, the U.S., and Spain have fertility rates of 1.39, 1.84, and 1.29 children per woman, respectively. By contrast, less developed countries like Niger, Angola, and the DRC have fertility rates of 6.73, 5.76, and 5.56, respectively. That is an astonishing difference. This difference can be explained by what is known as demographic transition, which is a well-established theory explaining the shift in countries with low economic development and education from high birth and death rates to low birth and death rates as they become more developed (Bongaarts 2009).

The demographic transition consists of five stages. The first stage is where population growth is extremely slow, with simultaneously high birth rates and high mortality, and is where the world has been for most of history (Roser 2019). An interesting consequence of high birth rates and high mortality is that the age structure of countries in the early stages of demographic transition trends younger (Bongaarts 2009). In the second stage, the population begins to grow rapidly because birth rates remain high while mortality decreases as the population's health improves due to medical, educational, and economic development (Roser 2019). The decline in mortality goes hand-in-hand with an increase in life expectancy and often leads to an increase in labor force productivity (Choudhry and Elhorst 2010).

The third stage is when birth rates begin to fall. The population is still growing rapidly due to low mortality, but people are choosing to have fewer children (Roser 2019). There have been multiple studies into why this is. The simplest explanation for lower fertility rates is that, with lower infant mortality resulting from development, there simply is no need to have as many children as before (Rangathan, Swain, and Sumpter 2015). Having larger numbers of children in the previous demographic transition stages served as a sort of insurance policy that a family would be able to continue on despite high child mortality; however, with mortality decreasing and parents expecting the children they have to survive, the need to have so many children evaporates. Another study employs the quantity-quality tradeoff model for fertility, which is a model set up so that parents decide how to allocate their income between their own consumption and spending on children. The authors find that spending on each individual child is higher when there are fewer children, and therefore, the quality of each child is higher with fewer children, which could drive the decision to have fewer children in the midst of demographic transition (Lee and Mason 2010). There is also the theory of "status anxiety" (specific to developed countries), in which parents worry about their children's social mobility and, therefore, have fewer children (Reher 2011). Finally, the fourth stage of demographic transition is when rapid population growth stops and birth rates and mortality are both low (Roser 2019). What happens in the fifth stage of demographic transition is yet to be determined, as only very few countries have developed to this stage; thus, it is uncertain whether populations tend to rise, fall, or plateau. It will ultimately depend on the fertility rates of these highly developed countries.

The impacts of demographic transitions on economic growth have also been well studied. With the changing population age structure, there are sure to be effects on economic factors like education, healthcare, and social security resources (Bongaarts 2009). For example, as the age structure trends younger in the first and second stages of demographic transition, education resources become more stretched with higher demand, but as the age structure trends older in the remaining stages, education resources may become less stretched while healthcare and social security resources experience higher demand. Across the literature, it has been found that lower fertility rates can drive GDP growth because fewer young dependents increase the size of the labor force, but ultimately, whether demographic transition results in economic growth or shrinkage could depend on what stage of the transition a given country is in (Choudhry and Elhorst 2010, Lee and Mason 2010, and Reher 2011). For instance, economic growth generally increases in the third stage when fertility and mortality are low, but how growth changes in the fifth stage is unknown due to where most societies are in the transition at this point in history. This spurred our interest in examining the relationships between economic development variables and demographic variables for underdeveloped, developing, and developed countries. We want to see how our analysis lines up with the existing literature by exploring the following research questions:

References

Bongaarts, J., 2009, "Human population growth and the demographic transition", Philosophical Transactions of the Royal Society, B, 364: 2985-2990.

Choudhry, M. T., and J. P. Elhorst, 2010, "Demographic transition and economic growth in China, India and Pakistan", Economic Systems, 34: 218-236.

Lee, R., A. Mason, 2010, "Fertility, Human Capital, and Economic Growth over the Demographic Transition", European Journal of Population, 26: 159-182.

Ranganathan, S., R. B. Swain, and D. J. T. Sumpter, 2015, "The demographic transition and economic growth: implications for development policy", Palgrave Communications, 1:15033.

Reher, D. S., 2011, "Demographic Transition and Its Consequences: Economic and Social Implications of the Demographic Transition", Population and Development Review, 37: 11-34.

Roser, M., 2019, "Demographic transition: Why is rapid population growth a temporary Phenomenon?", Our World in Data, https://ourworldindata.org/demographictransition.

Research questions

How do development demographics like literacy, primary school enrollment, fertility rate, age dependency ratio, life expectancy, and rural population correlate with economic growth? Some specific questions to consider are:

- Based on the literature, the fertility rate seems to be the primary driver of economic growth across the demographic transition, where lower fertility correlates with increased growth. Do we find the same result in our data if we compare the fertility rate and GDP growth? Contrary to the literature, do any of our variables appear to be more correlated with GDP growth than fertility?

- Age dependency is also a factor that is heavily related to economic growth in the literature, where lower age dependency is correlated with economic growth. Does our data tell a similar story?

- Intuition makes us think that literacy, primary school enrollment, and rural population could also be highly correlated with economic growth, despite not being mentioned in the literature. Based on our data, could this hypothesis hold?

- Can we explain any of these trends by determining the level of development of each country? For example, does country A fall into the category of developed, developing, or underdeveloped? Does this categorization help us understand where it is in the demographic transition and why it may or may not be experiencing economic growth?

Data

The raw data for this project come from the World Bank, at https://databank.worldbank.org/source/world-development-indicators/.

It contains the following variables ...

Variable	**Definition**	**Units**
agedep	Age Dependency Ratio, young: The ratio of dependents (people younger than 15) to the working-age population (those ages 15-64)	Dependents/100 Working-Age Population
fert	Fertility Rate: The average number of children born to a woman over her lifetime.	Children/Woman
gdpg	GDP Growth Rate: The average annual rate of change of the GDP in a given economy over one year.	% of Previous Year's GDP
lifex	Life Expectancy: The average period of years that a person in a given country may expect to live.	Years
enroll	Primary School Enrollment: The number of children of official primary school age who are enrolled in primary education as a percentage of the total children of the official school age population.	% of School Age Population
litr	Literacy Rate: The proportion of the adult population aged 15 years and over which is literate, expressed as a percentage of the corresponding population.	% of Total Population
rural	Rural Population: People living in rural areas as defined by national statistical offices.	% of Total Population

Data discussion

As mentioned, the data comes from the World Bank Database. It is important to note that the data is taken as the average over the previous 10 years so as to avoid issues with time series. Each row is an individual country and each column is one variable. Another fact worthy of note is that some countries did not have complete data for each variable, especially developing countries. Thus, the average of those variables over the past 10 years removes the NaN values.

Data cleaning

Import pandas

[2]:
```python
import pandas as pd
```

Read in raw demographics data

[3]:
```python
df = pd.read_csv('data/Demographics.csv')
df.head()
```

[3]:
```
  Country Name Country Code    Time Time Code  \
0  Afghanistan          AFG  2013.0    YR2013
1  Afghanistan          AFG  2014.0    YR2014
2  Afghanistan          AFG  2015.0    YR2015
3  Afghanistan          AFG  2016.0    YR2016
4  Afghanistan          AFG  2017.0    YR2017

   Age dependency ratio, young (% of working-age population) [SP.POP.DPND.YG]  \
0                                          92.388046
1                                          90.015900
2                                          88.398202
3                                          87.405774
4                                          85.970407

  Fertility rate, total (births per woman) [SP.DYN.TFRT.IN]  \
0                                              5.696
1                                               5.56
2                                              5.405
3                                              5.262
4                                              5.129

  GDP growth (annual %) [NY.GDP.MKTP.KD.ZG]  \
0                          5.60074465808154
1                          2.72454336394854
2                          1.45131466009755
3                          2.26031420130452
4                          2.64700320195786

  Life expectancy at birth, total (years) [SP.DYN.LE00.IN]  \
0                                             62.417
1                                             62.545
2                                             62.659
3                                             63.136
4                                             63.016

  School enrollment, primary (% gross) [SE.PRM.ENRR]  \
0                                   107.695976257324
```

```
1                                  109.115516662598
2                                  106.182418823242
3                                  106.150283813477
4                                  106.129997253418

  Literacy rate, adult total (% of people ages 15 and above) [SE.ADT.LITR.ZS]  \
0                                                 ..
1                                                 ..
2                                                 ..
3                                                 ..
4                                                 ..

  Rural population (% of total population) [SP.RUR.TOTL.ZS]
0                                            75.627
1                                            75.413
2                                            75.197
3                                             74.98
4                                             74.75
```

Rename variables for ease of use

[4]:
```
df = df.rename(columns={"Country Name":'Country',
                        "Time":'Year',
                        "Age dependency ratio, young (% of working-age␣
↪population) [SP.POP.DPND.YG]":'agedep',
                        "Fertility rate, total (births per woman) [SP.DYN.TFRT.
↪IN]": 'fert',
                        "GDP growth (annual %) [NY.GDP.MKTP.KD.ZG]": 'gdpg',
                        "Life expectancy at birth, total (years) [SP.DYN.LE00.
↪IN]": 'lifex',
                        "School enrollment, primary (% gross) [SE.PRM.ENRR]":␣
↪'enroll',
                        "Literacy rate, adult total (% of people ages 15 and␣
↪above) [SE.ADT.LITR.ZS]": 'litr',
                        "Rural population (% of total population) [SP.RUR.TOTL.
↪ZS]": 'rural'})
df.head()
```

[4]:
```
       Country Country Code    Year Time Code     agedep   fert  \
0  Afghanistan          AFG  2013.0    YR2013  92.388046  5.696
1  Afghanistan          AFG  2014.0    YR2014  90.015900   5.56
2  Afghanistan          AFG  2015.0    YR2015  88.398202  5.405
3  Afghanistan          AFG  2016.0    YR2016  87.405774  5.262
4  Afghanistan          AFG  2017.0    YR2017  85.970407  5.129

              gdpg   lifex            enroll litr   rural
0  5.60074465808154  62.417  107.695976257324   ..  75.627
```

```
1  2.72454336394854  62.545  109.115516662598   ..  75.413
2  1.45131466009755  62.659  106.182418823242   ..  75.197
3  2.26031420130452  63.136  106.150283813477   ..   74.98
4  2.64700320195786  63.016  106.129997253418   ..   74.75
```

Filter out year and time code

[5]:
```
df = df.filter(['Country', 'agedep', 'fert', 'gdpg',
                'lifex', 'enroll', 'litr', 'rural'])
df.head()
```

[5]:
```
       Country     agedep   fert              gdpg   lifex            enroll  \
0  Afghanistan  92.388046  5.696  5.60074465808154  62.417  107.695976257324
1  Afghanistan  90.015900   5.56  2.72454336394854  62.545  109.115516662598
2  Afghanistan  88.398202  5.405  1.45131466009755  62.659  106.182418823242
3  Afghanistan  87.405774  5.262  2.26031420130452  63.136  106.150283813477
4  Afghanistan  85.970407  5.129  2.64700320195786  63.016  106.129997253418

  litr   rural
0   ..  75.627
1   ..  75.413
2   ..  75.197
3   ..   74.98
4   ..   74.75
```

Make a copy of current data frame

[6]:
```
df2 = df.copy()
```

Convert the Country variable in the dataframe to a series called Country

[7]:
```
Country = df['Country']
Country.head()
```

[7]:
```
0    Afghanistan
1    Afghanistan
2    Afghanistan
3    Afghanistan
4    Afghanistan
Name: Country, dtype: object
```

[8]:
```
type(Country)
```

[8]:
```
pandas.core.series.Series
```

Change all non-numeric objects to NaN

[9]:
```
for col in  df2.columns[1:]:
    df2[col] = pd.to_numeric(df2[col], errors='coerce')
```

[10]:
```
df2.head()
```

[10]:

	Country	agedep	fert	gdpg	lifex	enroll	litr	rural
0	Afghanistan	92.388046	5.696	5.600745	62.417	107.695976	NaN	75.627
1	Afghanistan	90.015900	5.560	2.724543	62.545	109.115517	NaN	75.413
2	Afghanistan	88.398202	5.405	1.451315	62.659	106.182419	NaN	75.197
3	Afghanistan	87.405774	5.262	2.260314	63.136	106.150284	NaN	74.980
4	Afghanistan	85.970407	5.129	2.647003	63.016	106.129997	NaN	74.750

Count non-missing observations for each variable

[11]:
```
df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1958 entries, 0 to 1957
Data columns (total 8 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   Country  1955 non-null   object
 1   agedep   1953 non-null   float64
 2   fert     1892 non-null   float64
 3   gdpg     1867 non-null   float64
 4   lifex    1886 non-null   float64
 5   enroll   1494 non-null   float64
 6   litr     309 non-null    float64
 7   rural    1935 non-null   float64
dtypes: float64(7), object(1)
memory usage: 122.5+ KB
```

Calculate the mean of all non-missing values by country

[12]:
```
df = df2.groupby('Country').mean()
df.head()
```

[12]:

Country	agedep	fert	gdpg	lifex	enroll
Afghanistan	85.939664	5.146333	-0.362928	62.775111	107.580319
Albania	25.747409	1.518889	2.647364	78.333222	107.653550
Algeria	45.839689	2.993889	1.700000	75.576889	109.901002
American Samoa	46.999079	NaN	-0.096902	NaN	NaN
Andorra	20.279530	NaN	0.572704	NaN	89.001746

Country	litr	rural
Afghanistan	37.266041	74.708667
Albania	NaN	40.702556
Algeria	81.407837	28.003556
American Samoa	NaN	12.792889

```
Andorra              NaN  11.828778
```

Drop na's

[13]:
```
df = df.dropna()
df.head()
```

[13]:
```
                agedep      fert      gdpg      lifex      enroll       litr  \
Country
Afghanistan  85.939664  5.146333 -0.362928  62.775111  107.580319  37.266041
Algeria      45.839689  2.993889  1.700000  75.576889  109.901002  81.407837
Angola       87.853405  5.612556  0.170649  61.252222  109.244814  66.030113
Armenia      29.046853  1.598889  3.233333  74.051000   99.460075  99.756363
Aruba        26.537023  1.734111  2.447594  75.678778  117.817928  97.989998

                 rural
Country
Afghanistan  74.708667
Algeria      28.003556
Angola       35.206333
Armenia      36.815778
Aruba        56.650667
```

[14]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 123 entries, Afghanistan to Zimbabwe
Data columns (total 7 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   agedep  123 non-null    float64
 1   fert    123 non-null    float64
 2   gdpg    123 non-null    float64
 3   lifex   123 non-null    float64
 4   enroll  123 non-null    float64
 5   litr    123 non-null    float64
 6   rural   123 non-null    float64
dtypes: float64(7)
memory usage: 7.7+ KB
```

We have 123 countries left to work with, and here is our clean data

[15]:
```
df
```

[15]:
```
                       agedep      fert      gdpg      lifex      enroll  \
Country
Afghanistan         85.939664  5.146333 -0.362928  62.775111  107.580319
Algeria             45.839689  2.993889  1.700000  75.576889  109.901002
```

```
Angola              87.853405  5.612556  0.170649  61.252222  109.244814
Armenia             29.046853  1.598889  3.233333  74.051000   99.460075
Aruba               26.537023  1.734111  2.447594  75.678778  117.817928
...                       ...       ...       ...        ...         ...
Venezuela, RB       44.201793  2.296556 -1.275646  72.093889   98.563437
Viet Nam            33.369352  1.944889  5.871594  74.052444  111.475478
West Bank and Gaza  71.026831  3.814667  1.869919  74.261556   95.364152
Zambia              83.675509  4.634778  3.027344  61.604111  100.721525
Zimbabwe            77.194439  3.738444  1.223848  59.999222   99.378456

                         litr      rural
Country
Afghanistan         37.266041  74.708667
Algeria             81.407837  28.003556
Angola              66.030113  35.206333
Armenia             99.756363  36.815778
Aruba               97.989998  56.650667
...                       ...        ...
Venezuela, RB       96.866154  11.796556
Viet Nam            95.753868  64.776111
West Bank and Gaza  96.710934  24.089778
Zambia              87.500000  57.002000
Zimbabwe            88.693420  67.662222

[123 rows x 7 columns]
```

Data analysis

Create histograms of key variables, and scatterplots to illustrate relationships between them

[16]:
```
import warnings # remove annyoing user warnings #
warnings.filterwarnings("ignore", category=FutureWarning)
```

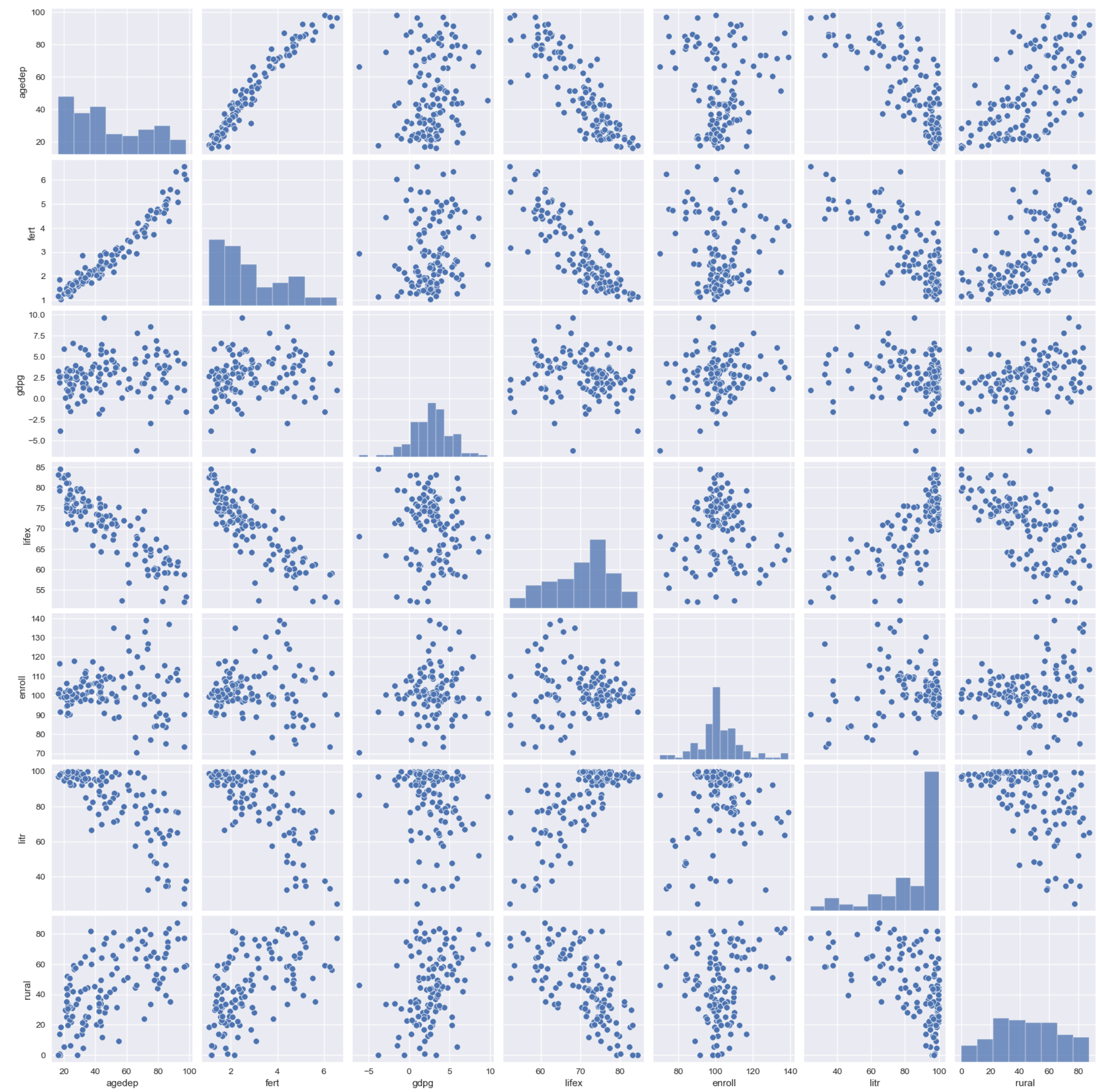
Import matplotlib, seaborn, numpy, and pandas

[17]:
```
%matplotlib inline
import seaborn;seaborn.set()
import matplotlib.pyplot as plt
plt.style.use('seaborn-v0_8')
import numpy as np
import pandas as pd
```

Create pairplot to get a quick overview of our variables' relationships

[18]:
```
import seaborn as sns
sns.pairplot(df)
```

```
[18]: <seaborn.axisgrid.PairGrid at 0x1331d9610>
```



This pairplot shows us that there are quite a few interesting and unexpected relationships between our variables of interest. For instance, there is a very defined positive relationship between the age dependency ratio and the fertility rate. There is also a clear negative relationship between life expectancy and fertility. Interestingly, the strongest relationship to GDP growth appears to be with the rural population, where the two are positively correlated (one might expect the opposite result). Now that we have a brief overview of the relationships between variables, we can dive into specific relationships and try to make sense of them. Let's start by examining the relationships between some of the demographic variables, and we'll finish the data analysis by discussing how the demographic variables relate to GDP growth.

Age dependency and fertility rate share almost the same relationship with each of the variables, as they are highly positively correlated with each other. It makes sense that the age dependency ratio of young dependents and the fertility rate would be positively correlated. As more children are produced, the ratio of dependents should increase; thus, we see the positive relationship between the two variables. There is a definite negative relationship between the fertility rate/age dependency ratio and life expectancy. This relationship is also quite intuitive and follows the story behind the demographic transition, where societies begin having fewer children (and thus, fewer dependents) as life expectancy increases. We can also see that higher literacy (i.e., higher development, which is associated with stage three of the demographic transition) correlates with a lower fertility rate/age dependency ratio. There is a positive relationship between the fertility rate/age dependency ratio and the rural population. Again, this follows the demographic transition story, where higher levels of development correlate with lower levels of fertility. If we think of rural population as a proxy for development, where higher rural populations are indicative of lower development, then it makes sense that higher rural populations are correlated with higher fertility. Finally, we thought there would be a clearer relationship between fertility/dependency and primary school enrollment, thinking that primary school enrollment and literacy would be more heavily related; however, neither of these relationships is strong in either direction.

As we discussed earlier, life expectancy has a definite negative relationship with the age dependency ratio and the fertility rate. Below is a graph of the relationship between life expectancy and the age dependency ratio, where we examine where certain countries lie on the graph:

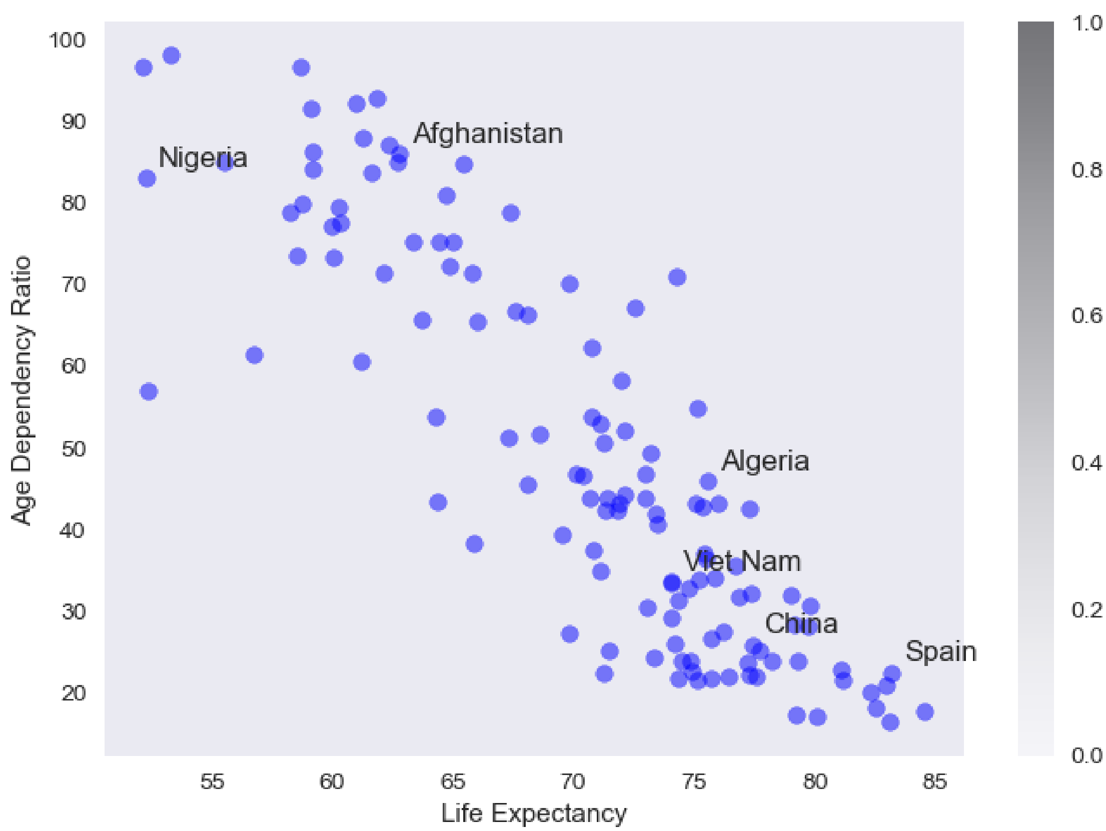
Plot relationship between life expectancy and age dependency ratio

[19]:
```
# Initialize variable names for easy graphing
GDPgrowth = df['gdpg']
Fertility = df['fert']
LifeExp = df['lifex']
Rural = df['rural']
Dependence = df['agedep']
Enrollment = df['enroll']
Literacy = df['litr']
```

[20]:
```
import matplotlib.pyplot as plt

# Assuming you have lists of country names corresponding to LifeExp and␣
↪Dependence
countries_to_label = ['Afghanistan', 'Viet Nam', 'Nigeria', 'Spain', 'Algeria',␣
↪'China']

fig = plt.figure()

# This line is necessary to avoid a pesky warning message
plt.rcParams['axes.grid'] = False

# Color the scatter plot dots differently by ROE
plt.scatter(LifeExp, Dependence,
            c='blue', cmap='viridis', alpha=0.5)
plt.colorbar()
```

```
# Label the axes
plt.xlabel('Life Expectancy')
plt.ylabel('Age Dependency Ratio')


for country in countries_to_label:
    plt.annotate(country,(df['lifex'][country],df['agedep'][country]),
    xytext=(5,5), textcoords='offset points')

plt.show()
```

```
/var/folders/5_/_j189m713gbb9x4bf6hnxg800000gn/T/ipykernel_2582/3465784530.py:12
: UserWarning: No data for colormapping provided via 'c'. Parameters 'cmap' will
be ignored
  plt.scatter(LifeExp, Dependence,
```



Again, there is not a clear relationship between primary school enrollment and life expectancy; however, it is clear that life expectancy has a very strongly positive relationship with literacy and a strongly negative relationship with the rural population. As we associate higher-developed countries with higher levels of both literacy and life expectancy, it makes perfect sense that the two variables are positively correlated. Further, it makes sense that countries with larger rural populations (i.e., potentially less developed in terms of healthcare) would have lower life expectancies.

As we've found from examining the pairplot, primary school enrollment does not seem to have a very strong relationship with any of our other variables; however, literacy has clear negative relationships with the fertility rate, the age dependency ratio, and the rural population and a strongly positive correlation with life expectancy. All of these relationships are explored in the above discussions. Interestingly, there is clear bunching on the far right-hand side of the literacy plots, which, in tandem with the histogram, tells us that most of the countries in our dataset are highly literate.

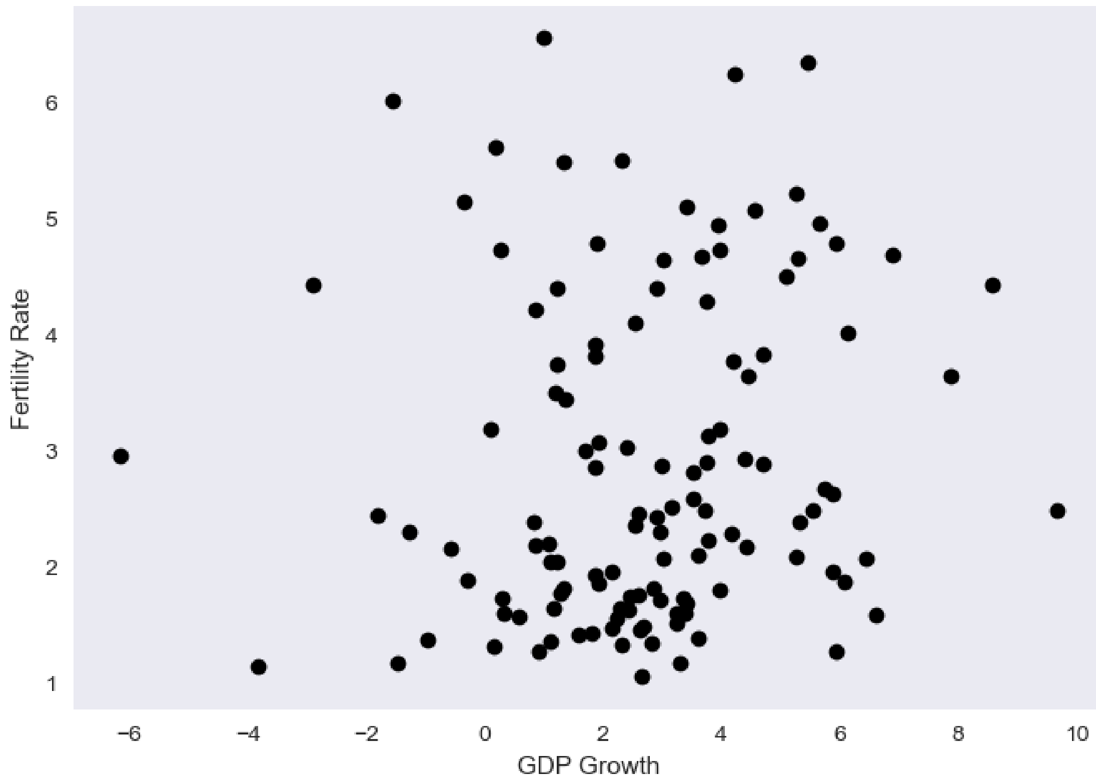
Finally, the rural population variable has clear positive correlations with the age dependency ratio and the fertility rate, negative relationships with life expectancy and literacy, and no clear relationship with school enrollment, as discussed before.

Now, we can look specifically at the relationships between our demographic variables and GDP growth:

[21]:
```
# Initialize variable names for easy graphing
GDPgrowth = df['gdpg']
Fertility = df['fert']
LifeExp = df['lifex']
Rural = df['rural']
Dependence = df['agedep']
Enrollment = df['enroll']
Literacy = df['litr']
```

Plot GDP growth against fertility rate

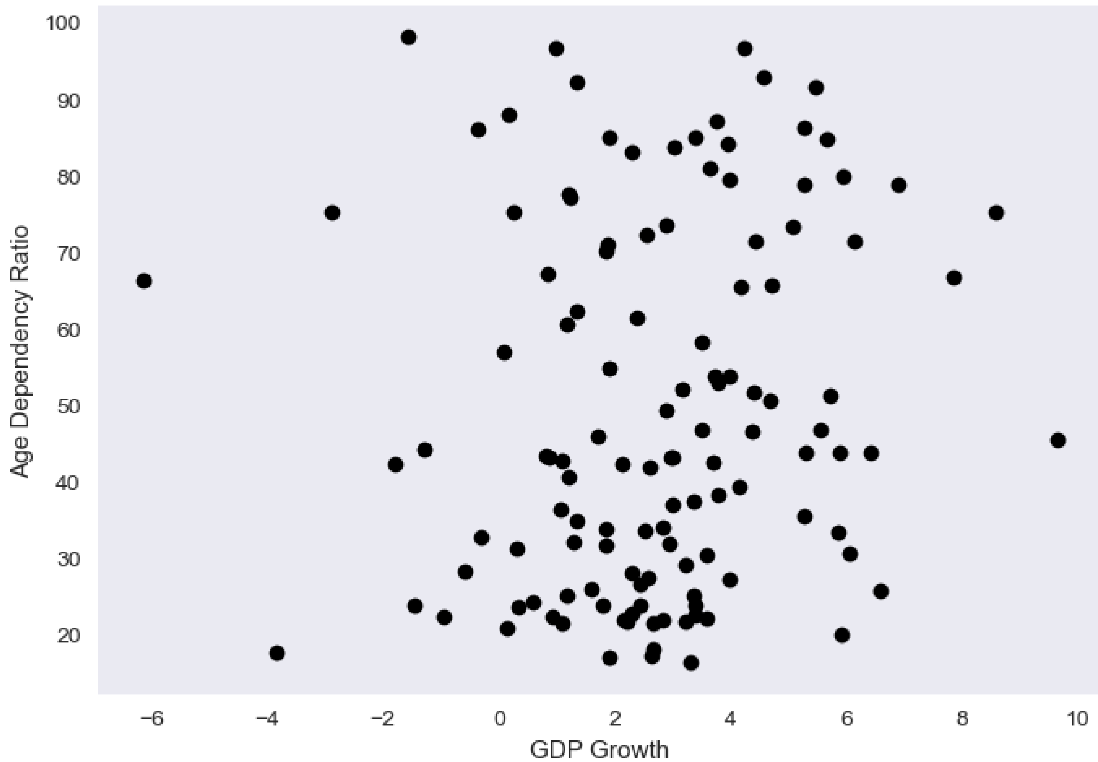
[22]:
```
# Start a new figure
fig = plt.figure()

# Plot GDP growth against fertility
plt.plot(GDPgrowth, Fertility, 'o', color='black');

# Label the axes
plt.xlabel('GDP Growth');
plt.ylabel('Fertility Rate');
```



This scatter plot represents the relationship between fertility rate and GDP growth. The relationship between fertility and GDP growth is weakly positive. Based on the literature, it is surprising that we are seeing a positive relationship between these two variables. The literature says that generally, lower fertility rates are associated with more economic growth due to fewer dependents and a larger labor force. This counterintuitive result could stem from the fact that we don't necessarily know at what stage of the demographic transition most of the countries in our dataset are in. Higher fertility could lead to higher economic growth if a country is in the first or second stage, while it could lead to lower growth if the country is in the third or fourth stage. We can explore this further by clustering the countries into development categories in the machine learning section.

Plot GDP Growth against age dependency ratio

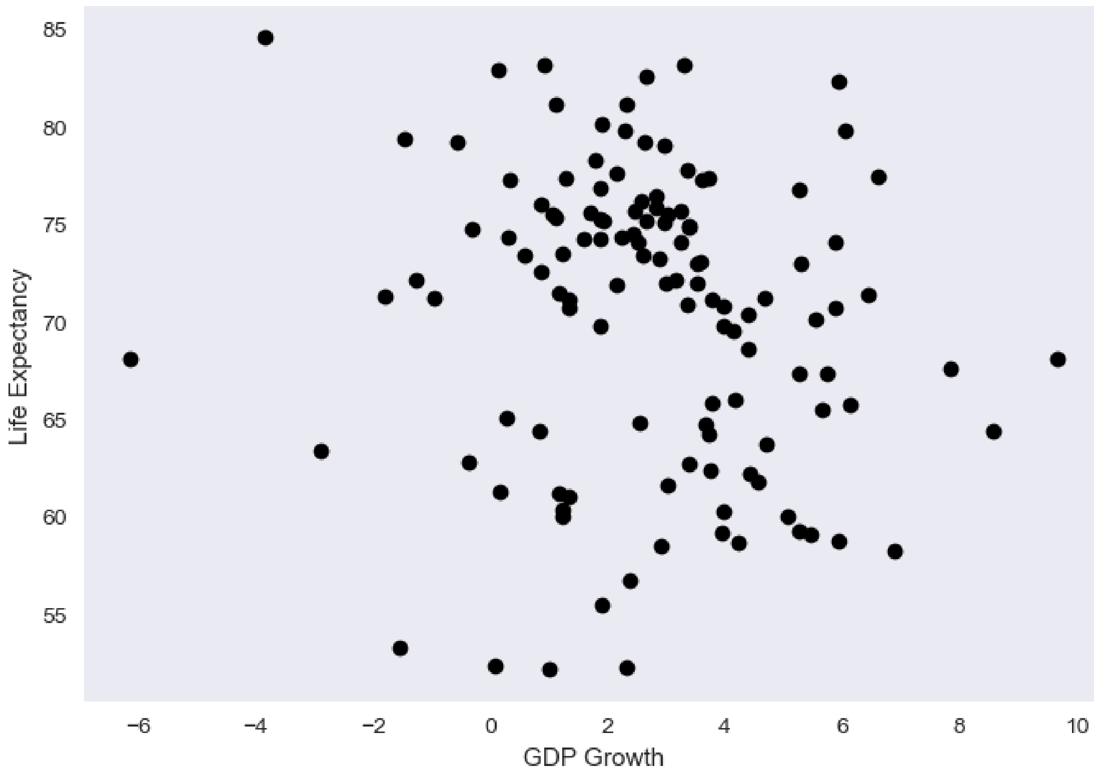
[23]:
```
# Start a new figure
fig = plt.figure()

# Plot GDP Growth against age dependency ratio
plt.plot(GDPgrowth, Dependence, 'o', color='black');

# Label the axes
plt.xlabel('GDP Growth');
plt.ylabel('Age Dependency Ratio');
```



This scatter plot represents the relationship between the age dependency ratio and GDP growth. We expected these to be negatively correlated, with more dependents leading to smaller GDP growth. Again, this plot is counterintuitive to the literature, but we could be seeing this difference because of the lack of knowledge about the stage of the demographic transition these countries are experiencing. While this result does not immediately make sense, it is at least consistent with the relationship between the fertility rate and GDP growth.

Plot GDP growth against life expectancy

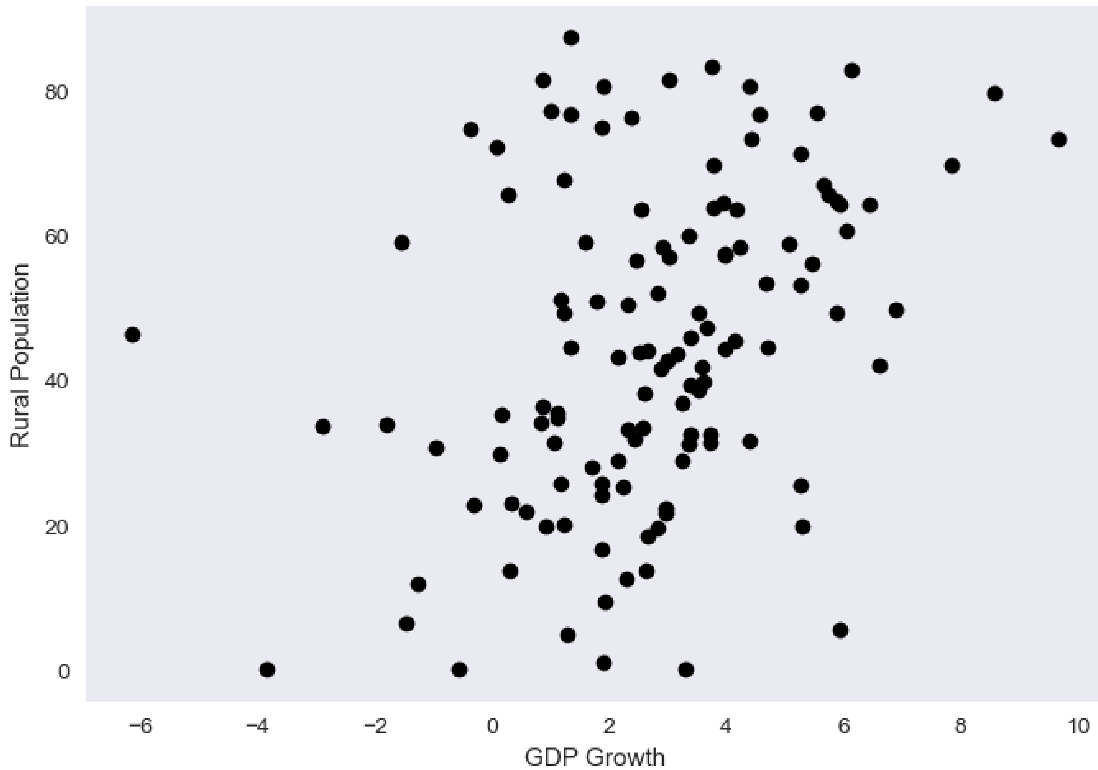
[24]:
```
# Start a new figure
fig = plt.figure()

# Plot GDP growth against life expectancy
plt.plot(GDPgrowth, LifeExp, 'o', color='black');

# Label the axes
plt.xlabel('GDP Growth');
plt.ylabel('Life Expectancy');
```



This scatter plot represents the relationship between life expectancy and GDP growth. We assumed these would be positively correlated, which they are up to a certain point, where the points turn back, making a sort of backward "c" shape. This structural break actually makes sense, though, because up to a certain point, higher life expectancy means more people in the labor force and thus more GDP growth; however, once the life expectancy exceeds a certain age, the elderly essentially become dependents that must be provided for because they can no longer work, and so GDP growth decreases.

Plot GDP growth against rural population

[25]:
```
# Start a new figure
fig = plt.figure()

# Plot GDP Growth against rural population
plt.plot(GDPgrowth, Rural, 'o', color='black');

# Label the axes
plt.xlabel('GDP Growth');
plt.ylabel('Rural Population');
```

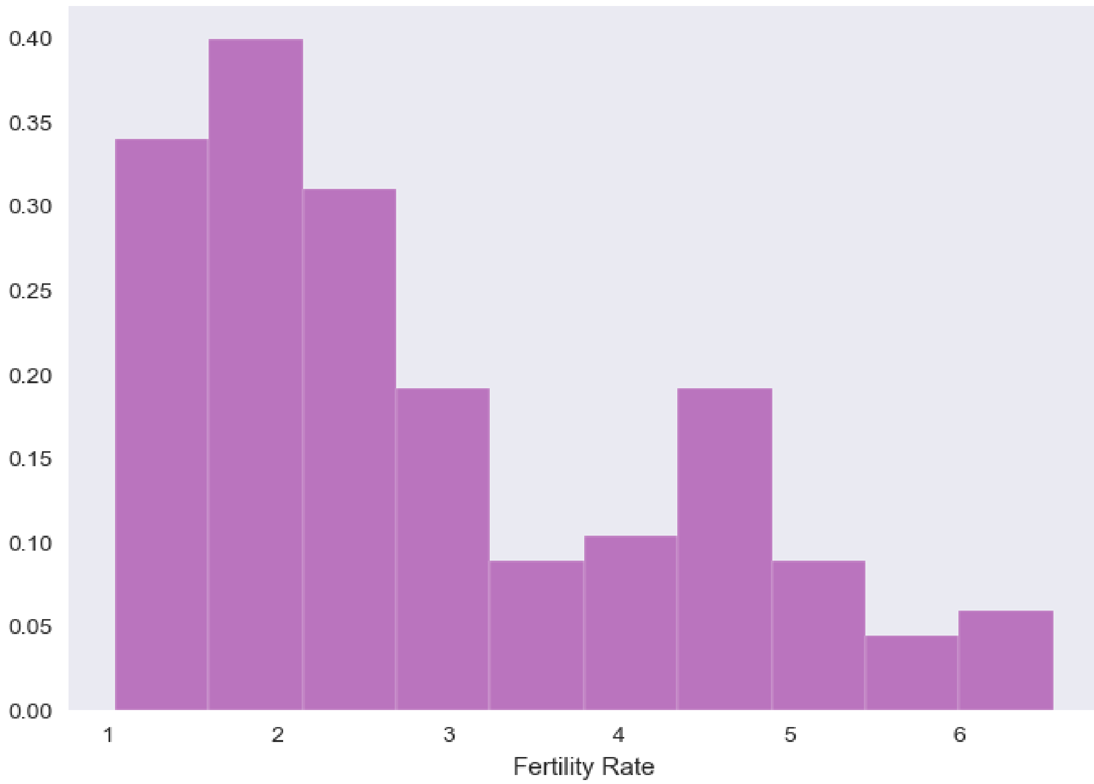


This scatter plot represents the relationship between the rural population and GDP growth. We expected these to be negatively correlated, with a higher rural population being associated with underdeveloped countries. This graph seems to show the opposite, with the two variables being positively correlated. This could be because we failed to account for how much agriculture impacts a country's GDP growth, where some more rural countries have much more agricultural industries contributing to GDP.

Now, let's revisit the histograms for each variable in more detail to understand country trends.

Plot histogram of fertility rate

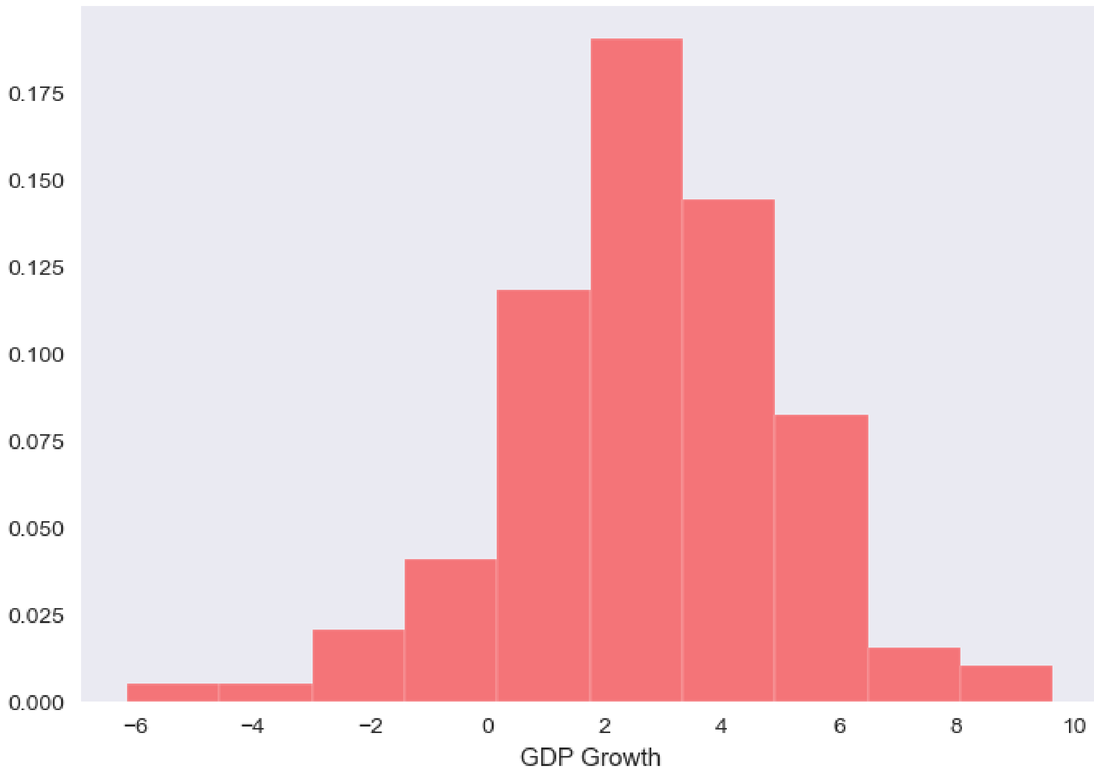
[26]:
```
fig = plt.figure()

plt.hist(Fertility, bins=10, density=True,
         color='darkmagenta', alpha=0.5);
plt.xlabel('Fertility Rate');
```



This histogram represents the frequency of each fertility rate value in the data. This shows that more than half of the countries in our data set have an average fertility rate of just under 2 children per woman.

Plot histogram of GDP growth

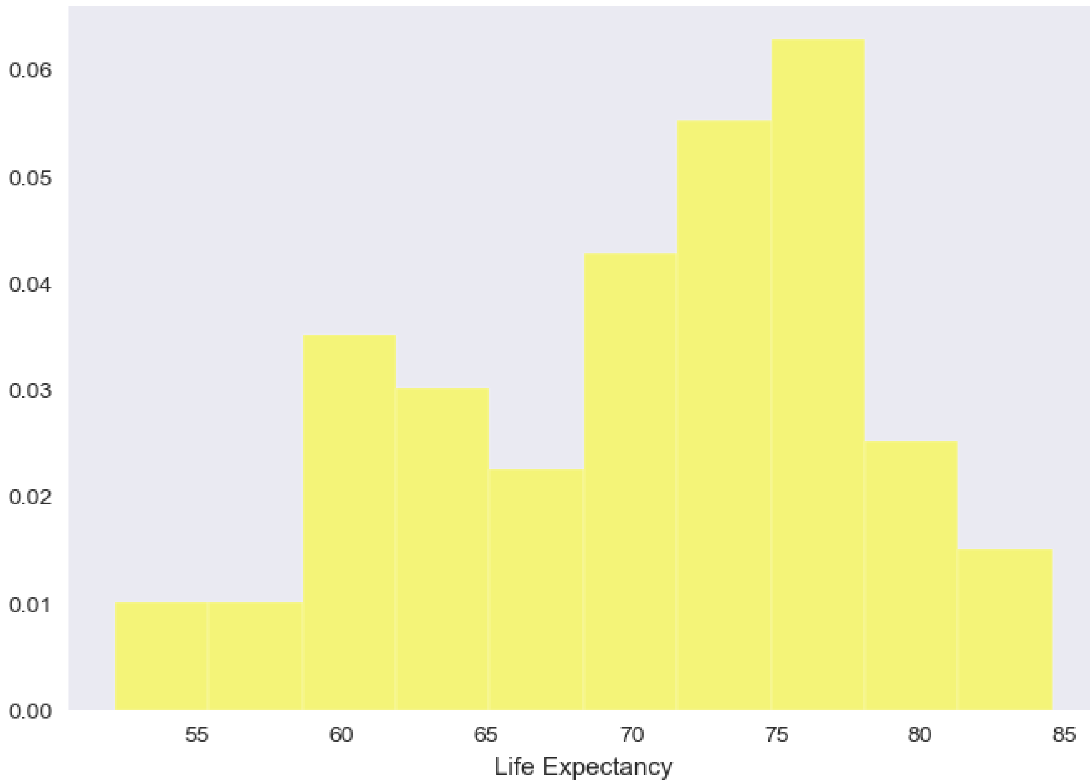
[27]:
```
fig = plt.figure()

plt.hist(GDPgrowth, bins=10, density=True,
         color='red', alpha=0.5);
plt.xlabel('GDP Growth');
```



This histogram represents the frequency of each GDP growth value in the data. This shows that most of the countries in our dataset have GDP growth of about 3%. Very rarely do countries see GDP growth of more than 6% or less than 1%.

Plot histogram of life expectancy

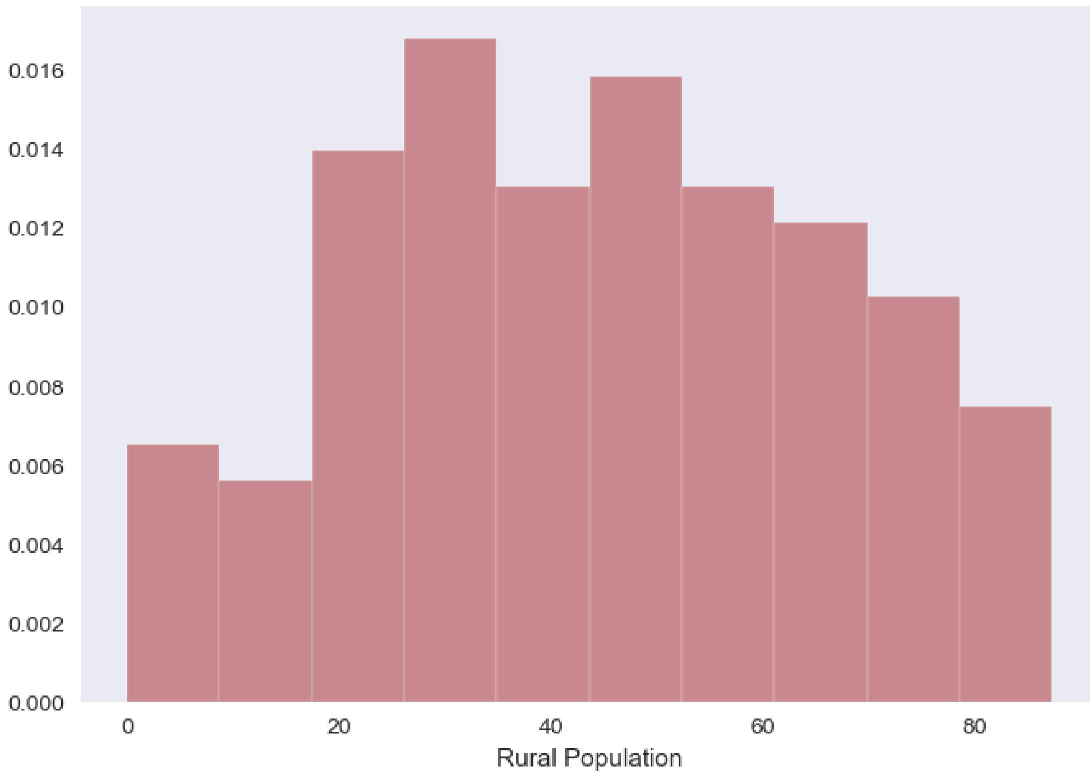
[28]:
```
fig = plt.figure()

plt.hist(LifeExp, bins=10, density=True,
         color='yellow', alpha=0.5);
plt.xlabel('Life Expectancy');
```



This histogram represents the frequency of each life expectancy value in the data. It has two peaks, telling us that the majority of countries have a life expectancy of either ~60 or 75 years of age.

Plot histogram of rural population

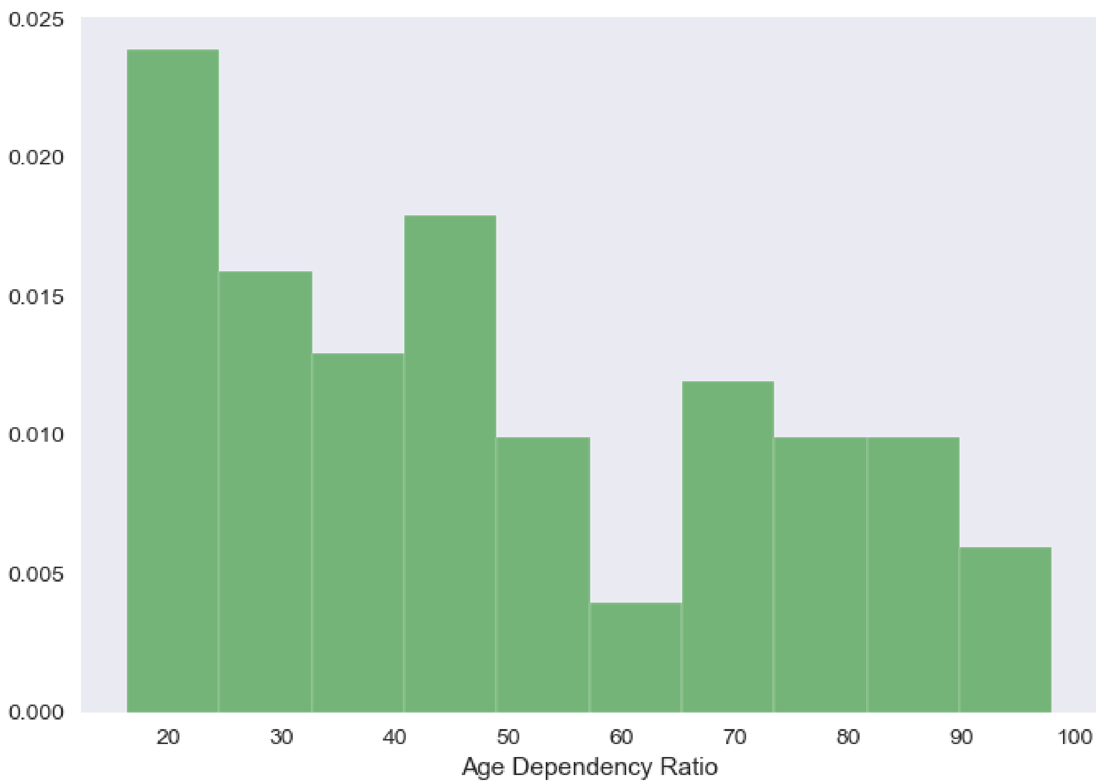
[29]:
```
fig = plt.figure()

plt.hist(Rural, bins=10, density=True,
         color='brown', alpha=0.5);
plt.xlabel('Rural Population');
```



This histogram represents the frequency of each rural population value in the data. The pattern of this histogram is much more difficult to interpret. It has multiple peaks at varying levels of rural population, but does have the highest peaks at ~35% and ~50% of the population being rural.

Plot histogram of age dependency ratio

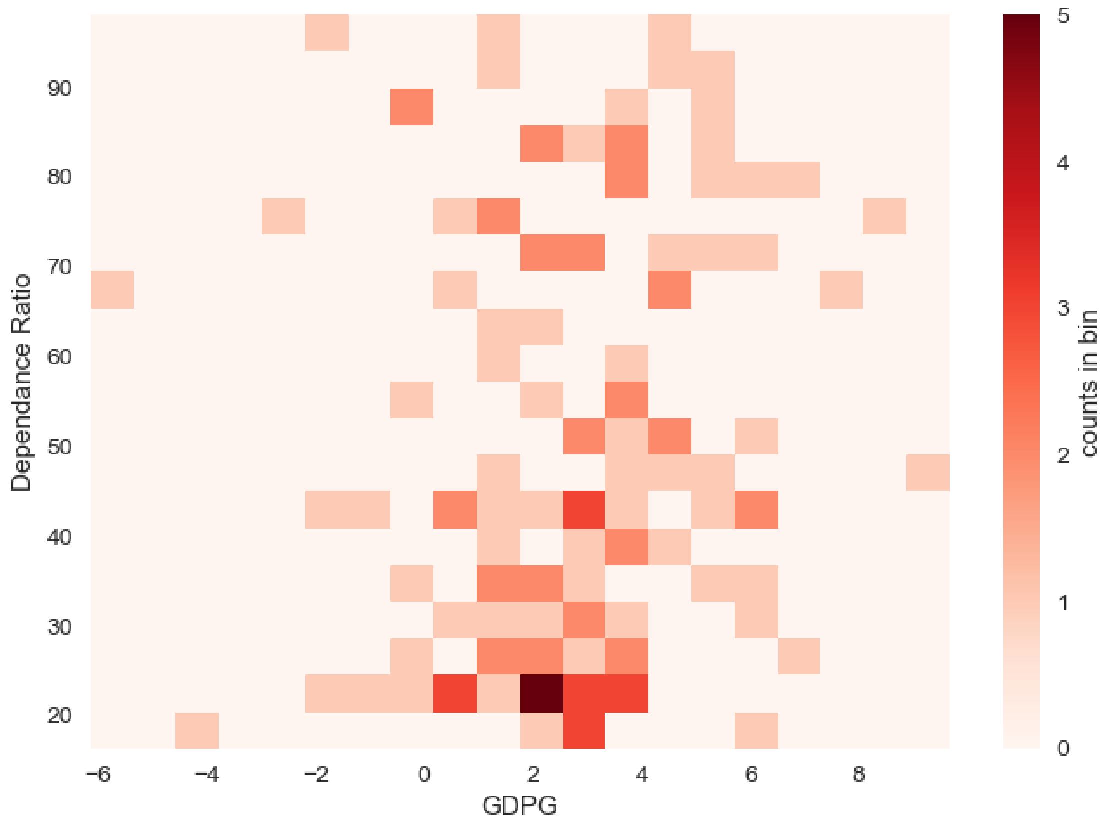
[30]:
```
fig = plt.figure()

plt.hist(Dependence, bins=10, density=True,
         color='green', alpha=0.5);
plt.xlabel('Age Dependency Ratio');
```



This histogram represents the frequency of each age dependency ratio value in the data. This data has one strong peak at around 25% of young dependents but, otherwise, has a fairly evenly distributed pattern.

Plot two-dimensional histogram for GDP growth and age dependency ratio

[31]:
```python
# Start a new figure
fig = plt.figure()

# This line is necessary to avoid a pesky warning message
plt.rcParams['axes.grid'] = False

# Generate a two-dimensional histogram of the log-
# salary data against log-sales data
plt.hist2d(GDPgrowth, Dependence, bins=20, cmap='Reds');

# Add a color bar legend
cb = plt.colorbar()
cb.set_label('counts in bin')

plt.xlabel('GDPG');
plt.ylabel('Dependance Ratio');
```



This is an attempt at creating a two dimensional histogram for GDP growth and age dependency ratio. This 2D histogram illustrates the density of the related data by grouping x and y points into bins. These graphs are useful to combat over-plotting and are better for large data sets that may overlap or hide patterns. The graph (above) shows a high density between 0 and 4 for the GDP growth rate and between 20 and 30 for the dependence ratio. This may suggest that stagnant or average growth rates are correlated with a low dependence ratio.

Plot histogram of age dependency ratio with the population split into older and younger

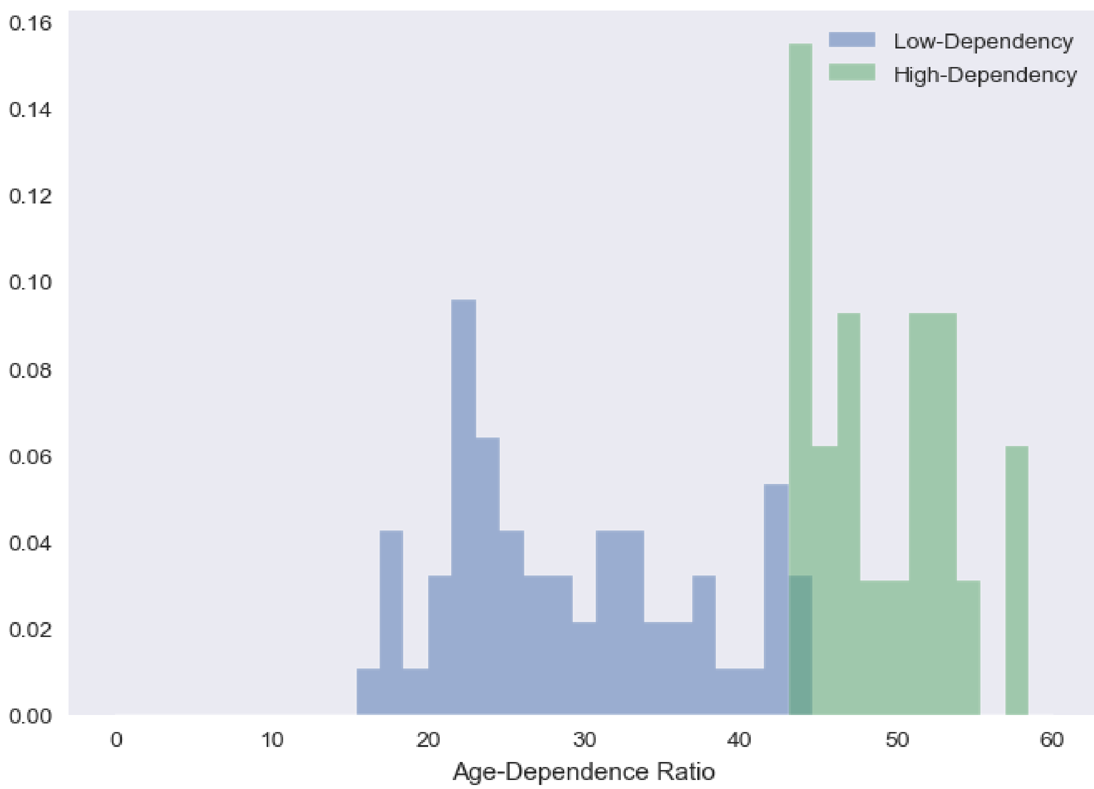
[32]:
```
# Initialize a new category variable called 'Age' to an empty string for all␣
 ↪observations
df['Dependency'] = ''

# Turn the agedep variable into a series
Dependance = df['agedep']

# For those observations for which the CEO salary is below the median, replace␣
 ↪the empty string with "Low" in the 'Age' column
df.loc[df['agedep'] < df['agedep'].median(), 'Dependency'] = "Low"
```

```
# For those observations for which the CEO salary is equal to or above the␣
↪median, replace the empty string with "High" in the 'Age' column
df.loc[df['agedep'] >= df['agedep'].median(), 'Dependency'] = "High"

# Tabulate the resulting values of the new 'Age' variable
df['Dependency'].value_counts()
```

[32]:
```
Dependency
High    62
Low     61
Name: count, dtype: int64
```

[33]:
```
# Turn the category variable 'Age' into a series
Age = df['Dependency']  # Assuming 'df' is your DataFrame

# Create a dictionary of options to be used
# for all histograms to be overlayed
options = dict(histtype='stepfilled', alpha=0.5,
               density=True, bins=np.linspace(0, 60, 40))

# Start a new figure
fig = plt.figure()

# Plot a histogram of agedep values only for observations where Age is 'Low'
plt.hist(df['agedep'][Age == 'Low'], label='Low-Dependency', **options)

# Overlay a histogram of agedep values only for observations where Age is 'High'
plt.hist(df['agedep'][Age == 'High'], label='High-Dependency', **options)

# Add a legend
plt.legend()

plt.xlabel('Age-Dependence Ratio')
```

[33]:
```
Text(0.5, 0, 'Age-Dependence Ratio')
```



This graph categorizes our data by separating those with a high dependency ratio from those with a low ratio. Calculated based on the median of the dependency ratio, this graph illustrates the dichotomy between the high and low ratios. However, it should be noted that this does not necessarily infer age, as countries with a plethora of babies also register as highly dependent.

Possible Machine Learning Models

1. One possible machine learning technique that can be used in this case is **meanshift clustering**. Our goal is to to try to cluster observations (countries) by development level. Each of the features in the features matrix is a development indicator. As an example of unsupervised learning, it would be interesting to use meanshift clustering to identify categories of development and compare them to traditional categories.

2. However, because we in theory know how many clusterst to look for (developed, developing, and undeveloped), **k-means clustering** could be another machine learning model to use in labeling development levels.

3. For another analysis, **Gaussian Naïve Bayes Classifiers** could be used to classify countries into different categories of demographic transition. Countries with a higher age dependency ratio should be further from the demographic transiton than others.

4. To support each of these methods, and to provide meaningful interpretations, **principal component analysis** can narrow down the important variables of the dataset.

Machine Learning

[34]:
```
import warnings # remove annyoing user warnings #
warnings.filterwarnings("ignore", category=UserWarning)
```

Choose a class of model by importing the appropriate estimator class from Scikit-Learn

Format: import model from ScikitLearn view model hyperparameters

[35]:
```
from sklearn.decomposition import PCA
PCA?
```

[36]:
```
from sklearn.cluster import KMeans
KMeans?
```

[37]:
```
from sklearn.cluster import MeanShift
MeanShift?
```

[38]:
```
from sklearn.naive_bayes import GaussianNB
GaussianNB?
```

Choose model hyperparameters by instantiating this class with desired values

[39]:
```
pca = PCA(n_components=2)
```

[40]:
```
KMmodel = KMeans(n_clusters=3) # instantiate kmeans with 1 cluster per level of␣
 ↪development #
```

[41]:
```
MSmodel = MeanShift() # instantiate meanshift model #
```

[42]:
```
GNBmodel = GaussianNB() # instantiate naive bayes #
```

Arrange data into features matrix and target vector.

[43]:
```
names = ['low','medium','high'] # set bins and bin names for discretizing #
bins = [0, 30, 60, np.inf]
```

[44]:
```
df['agedep_cat'] = pd.cut(df['agedep'], bins, labels = names) # discretize age␣
 ↪dependency ratio for SVC #
df.head()
```

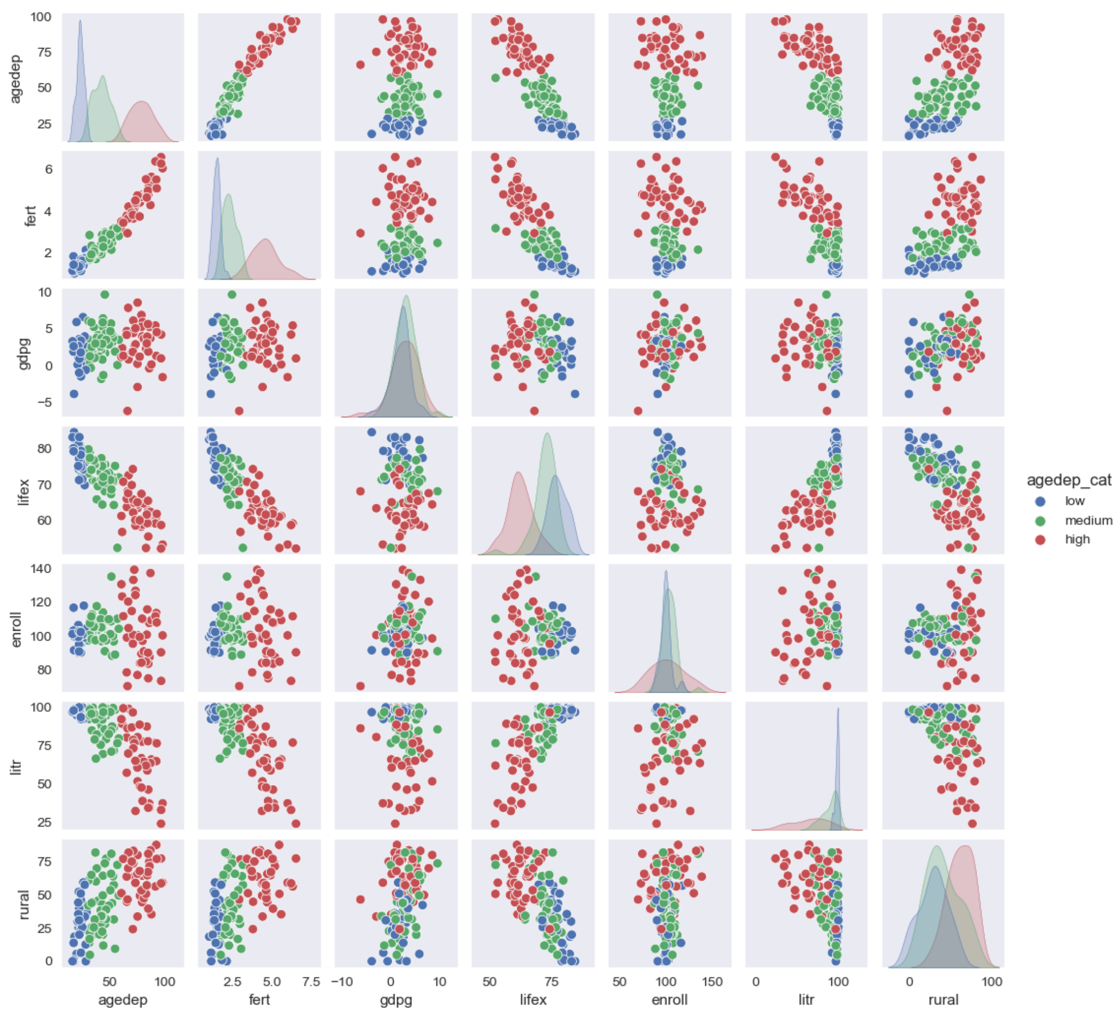
[44]:
```
                agedep      fert      gdpg      lifex      enroll       litr  \
Country
Afghanistan  85.939664  5.146333 -0.362928  62.775111  107.580319  37.266041
Algeria      45.839689  2.993889  1.700000  75.576889  109.901002  81.407837
Angola       87.853405  5.612556  0.170649  61.252222  109.244814  66.030113
Armenia      29.046853  1.598889  3.233333  74.051000   99.460075  99.756363
Aruba        26.537023  1.734111  2.447594  75.678778  117.817928  97.989998

                 rural Dependency agedep_cat
```

```
Country
Afghanistan  74.708667       High       high
Algeria      28.003556       High     medium
Angola       35.206333       High       high
Armenia      36.815778        Low        low
Aruba        56.650667        Low        low
```

[45]:
```python
y = df[['agedep']] # target vector #
y_discrete = df[['agedep_cat']]
X = df[['fert','gdpg','lifex','enroll','litr','rural']] # features matrix #
```

[46]:
```python
sns.pairplot(df,hue='agedep_cat',height=1.5)
```

[46]: <seaborn.axisgrid.PairGrid at 0x147977650>



The three age dependency categories (low, medium, and high) are most easily distinguishable for the age dependency variable and for the fertility variable. Given that age dependency and fertility are so highly correlated, this makes sense. We can now explicitly see that low age dependency is correlated with slightly lower GDP growth (GDP is potentially growing more slowly as the population is reaching its potential and having fewer children), higher life expectancy, higher literacy, and lower rural populations. Countries with low age dependency are more than likely to be countries in the later stages of the demographic transition (i.e., more developed), while countries with high age dependency are likely to be in the early stages of the demographic transition (i.e., less developed), either before or during the period of rapid population growth .

Age dependency categories are much less differentiable for our other variables. For example, our GDP growth, literacy, and rural population variables show a lot of mixing of the low, medium, and high age dependency categories. The low and medium categories are particularly jumbled; however, one important note to make is that, despite some overlapping, the high age dependency category is generally quite well separated from the other two. This means that high age dependency countries should be easier for us to distinguish from low and medium dependency countries, which is quite helpful for telling our demographic transition story.

[47]:
```
print(" y shape: ",y.shape,"\n","y_discrete shape: ",y_discrete.shape,"\n","X␣
↪shape: ",X.shape) # check the data #
```

```
 y shape:  (123, 1)
 y_discrete shape:  (123, 1)
 X shape:  (123, 6)
```

[48]:
```
y.head() # check target vector
```

[48]:
```
                agedep
Country
Afghanistan  85.939664
Algeria      45.839689
Angola       87.853405
Armenia      29.046853
Aruba        26.537023
```

[49]:
```
y_discrete.head() # discrete target vector #
```

[49]:
```
            agedep_cat
Country
Afghanistan       high
Algeria         medium
Angola            high
Armenia            low
Aruba              low
```

[50]:
```
X.head() # check features matrix #
```

[50]:
```
                 fert       gdpg       lifex        enroll        litr       rural
Country
```

```
Afghanistan  5.146333 -0.362928  62.775111  107.580319  37.266041  74.708667
Algeria      2.993889  1.700000  75.576889  109.901002  81.407837  28.003556
Angola       5.612556  0.170649  61.252222  109.244814  66.030113  35.206333
Armenia      1.598889  3.233333  74.051000   99.460075  99.756363  36.815778
Aruba        1.734111  2.447594  75.678778  117.817928  97.989998  56.650667
```

Split data into *training* set and *testing* set

[51]:
```
from sklearn.model_selection import train_test_split

train_test_split?
```

[52]:
```
Xtest, Xtrain, ytest, ytrain, y_disctest, y_disctrain = ␣
↪train_test_split(X,y,y_discrete,random_state=1,test_size=0.5,shuffle=True)
```

[53]:
```
Xtrain.head()
```

[53]:
```
                fert      gdpg      lifex      enroll        litr      rural
Country
Italy       1.313333  0.145629  82.927642  101.660625   99.349098  29.847333
Ukraine     1.361778 -0.958311  71.226206   90.752796  100.000000  30.722889
Mozambique  4.937667  3.948685  59.212778  115.599219   58.824680  64.529222
Tanzania    4.962889  5.653959  65.444667   89.997944   77.887230  66.937778
Hungary     1.510000  3.234005  75.688618   99.141894   99.099998  28.921889
```

[54]:
```
y_disctrain.head()
```

[54]:
```
           agedep_cat
Country
Italy             low
Ukraine           low
Mozambique       high
Tanzania         high
Hungary           low
```

Fit the model to the training data using the `fit()` method of the model instance

[55]:
```
PCAfit = pca.fit(X) # run PCA to identify "most important" features #KMfit =␣
↪KMmodel.fit(Xtrain,ytrain)
```

[56]:
```
KMfit = KMmodel.fit(Xtrain,ytrain) # K-means #
```

[57]:
```
MSfit = MSmodel.fit(Xtrain,ytrain)
```

[58]:
```
GNBfit = GNBmodel.fit(Xtrain,y_disctrain) # naive bayes #
```

Apply the model to the test data using the `predict()` method of the model instance

[59]:
```
y_KMpredict = KMmodel.predict(Xtest)
```

[60]:
```
y_MSpredict = MSmodel.predict(Xtest)
```

[61]:
```
y_GNBpredict = GNBmodel.predict(Xtest)
```

Assessing Model Accuracy

[62]:
```
from sklearn.metrics import accuracy_score as acscore
```

[63]:
```
kmlabels = {'low': 0, 'medium': 2, 'high': 1}
mslabels = {'low': 2, 'medium': 1, 'high': 0}
y_disctestnumkm = y_disctest.replace(kmlabels)
y_disctestnumms = y_disctest.replace(mslabels)
```

[64]:
```
print(' K-Means Accuracy: ',acscore(y_disctestnumkm,y_KMpredict),'\n',
     'Meanshift Accuracy: ',acscore(y_disctestnumms,y_MSpredict),'\n',
     'Gaussian Naïve Bayes Accuracy: ',acscore(y_disctest,y_GNBpredict))
```

```
 K-Means Accuracy:  0.4098360655737705
 Meanshift Accuracy:  0.19672131147540983
 Gaussian Naïve Bayes Accuracy:  0.8524590163934426
```

An interesting note to make here is that, having tried three different models on our data, Gaussian Naive Bayes has the highest accuracy, while K-means and meanshift clustering are accurate less than 50% of the time.

Show components and explained variance for graphing purposes and examining feature relationships

[65]:
```
print(pca.components_)
```

```
[[ 0.04011392  0.02832663 -0.23326151  0.04044072 -0.58427338  0.77470427]
 [ 0.03011451 -0.03253331 -0.0951231  -0.49203408 -0.68838969 -0.52250205]]
```

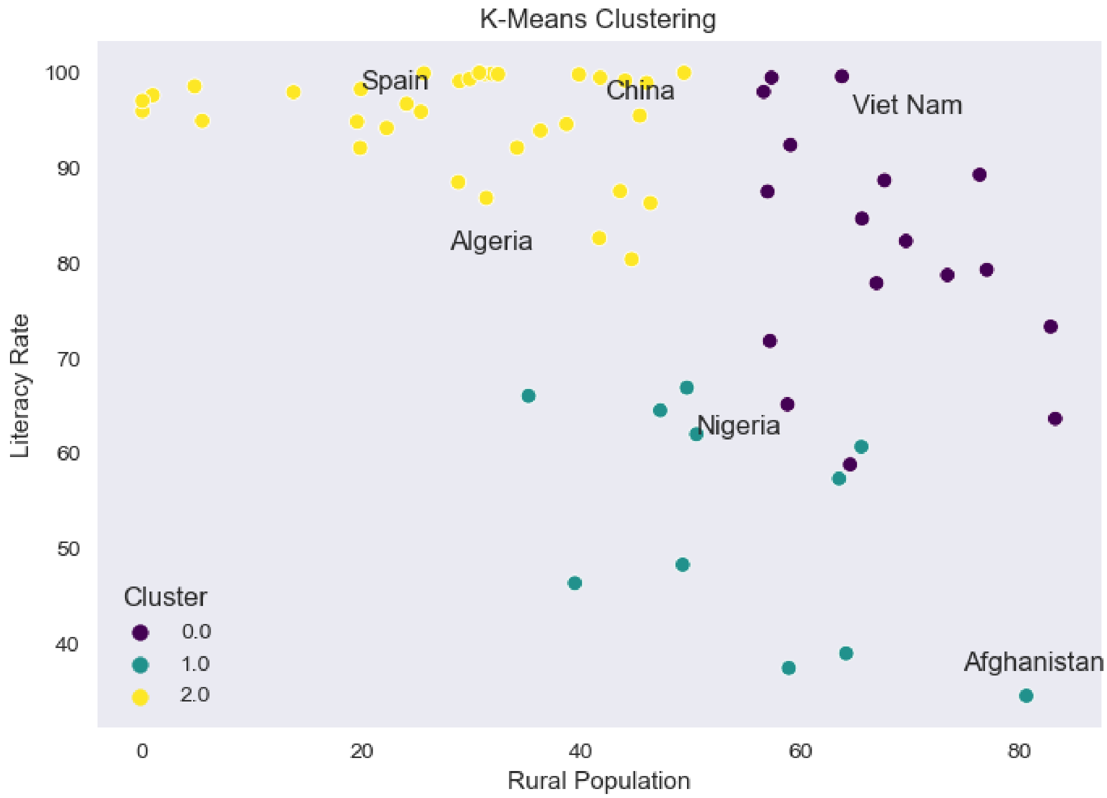
[66]:
```
print(pca.explained_variance_ratio_)
```

```
[0.65566068 0.21232424]
```

Machine Learning Graphs

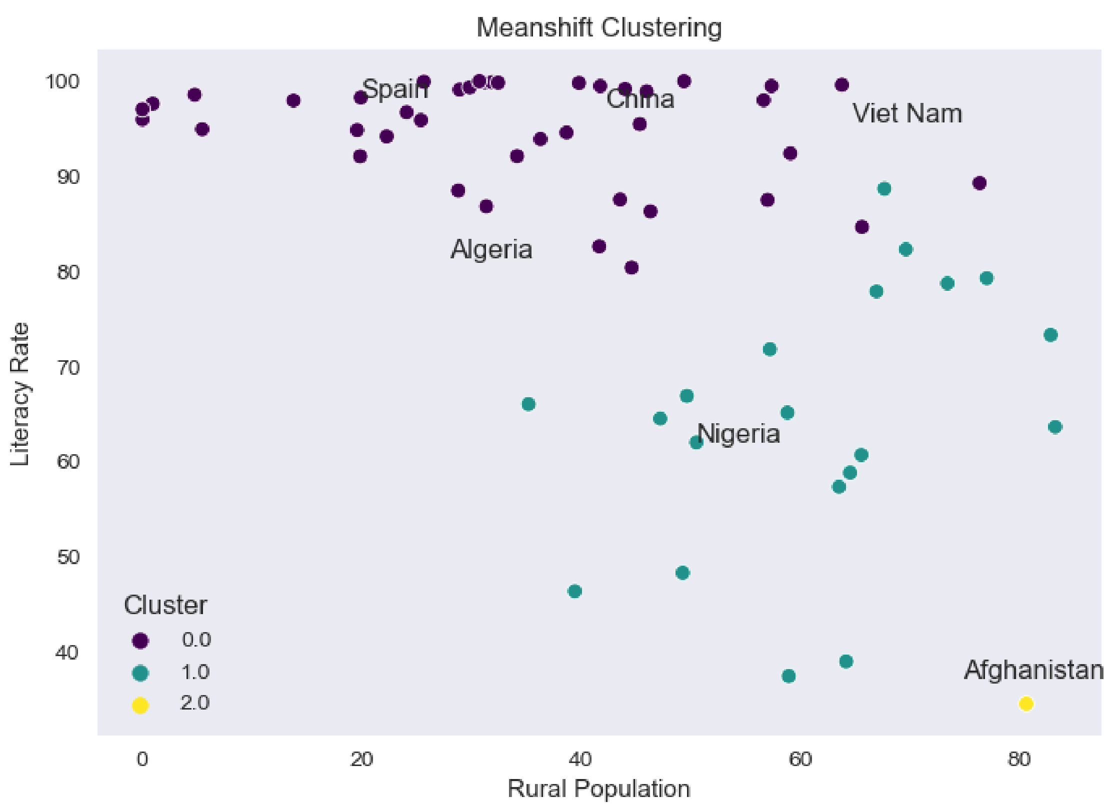
[84]:
```
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd

X_copy = X.copy()

kmclust = pd.DataFrame(data=KMfit.labels_,columns=['Cluster'],index=Xtrain.index)
KMdf = pd.concat([X_copy,kmclust],axis=1)
sns.scatterplot(x='rural',y='litr',hue='Cluster',data=KMdf,palette='viridis')
plt.title('K-Means Clustering')
```

```
plt.xlabel('Rural Population')
plt.ylabel('Literacy Rate')
#for country in Xtrain.index:
#    plt.annotate(country,xy=(Xtrain.loc[country,'rural'],Xtrain.
 ↪loc[country,'litr']))
# The above 'for' statement labels all countries, the one below cleans the plot␣
 ↪and labels only a few countries for viewing and interpretability
for country in countries_to_label:
    plt.annotate(country,(X['rural'][country],X['litr'][country]))
plt.show()
```



Here, we apply the K-means clustering model to our data, plotting rural population against literacy rate. We can see three relatively distinct clusters, where we might assume that the yellow cluster with higher literacy and lower rural populations are countries that are further along in the demographic transition and, therefore, could be called "developed" countries. The turquoise cluster groups countries with a lower literacy rate and a generally higher rural population, which is indicative of "developing" countries. And finally, the purple cluster shows a group of countries with low literacy and similar rural populations as the turquoise cluster. We might refer to these countries as "underdeveloped." It will be interesting to examine how these clusters differ using meanshift clustering.

[85]:
```
msclust = pd.DataFrame(data=MSfit.labels_,columns=['Cluster'],index=Xtrain.index)
MSdf = pd.concat([X_copy,msclust],axis=1)
sns.
 ↪scatterplot(x='rural',y='litr',hue='Cluster',data=MSdf,palette='viridis',marker='o')
plt.title('Meanshift Clustering')
plt.xlabel('Rural Population')
plt.ylabel('Literacy Rate')
#for country in Xtrain.index:
#    plt.annotate(country,(X['rural'][country],X['litr'][country]))
# Attempt to label only a few countries given the messiness of all of the␣
 ↪countries being printed.
for country in countries_to_label:
    plt.annotate(country,(X['rural'][country],X['litr'][country]))
plt.show()
```



Looking at the same relationship between literacy and rural population for each of our countries, meanshift clustering comes up with very different clusters compared to K-means clustering. It has grouped many of the countries that were previously in separate clusters into two larger clusters. The clustering here seems more driven by literacy than rural population, whereas previously, the clustering seemed to have been driven by literacy and rural population equally. Given that the accuracy of the K-means clustering is higher than the meanshift for our data and outlined parameters, we are inclined to trust the clustering of the previous plot more than this one.

Conclusion

To summarize, we pulled demographic data from the World Bank to conduct an analysis of the demographic transition across countries and investigated how demographic factors can inform us of the level of development or stage of the demographic transition for a given country. We cleaned the data, then carried out data analysis, where we found very high positive correlations between the age dependency ratio and fertility rate, and that these two variables are positively correlated with the rural population. We also found that these two variables are strongly negatively correlated with life expectancy and literacy. Next, we tried a few machine learning methods, finding that Gaussian Naive Bayes most accurately predicted our data. We used K-means and meanshift clustering to cluster our countries into developed, developing, and underdeveloped categories, finding that K-means did a better job of clustering than meanshift.

Ultimately, our data tells an interesting story of demographic transition across the world. Some potential further investigation could involve taking development classification data and assessing how our clusters line up with real-world classifications.