Project 1 Report

Irene Yang (Uniqname: yyichieh)

Motivation

My proposed project is focused on criminal incidence in the United States. Due to the social media threats of gunshots at the University of Michigan a few weeks ago, I started to think of a topic that related to crime.

In this project, I would try to find out which state may have the highest crime rate and to know the factors of the high crime rate. After reading some pieces of information about criminality in the United States, I suggested that the unemployment rate and educational level may be the two main factors that influence the criminal rate.

Data Sources

I used three datasets and there will be a brief introduction for them:

1. "United States crime rates by county"

   This data can be downloaded as a csv file. There are 24 columns and 3236 rows which cover the crime rate per 100000 people, county name with state code, population of the county, and the count of different kinds of crime acts. I decided to use the state code and recalculate the crime rate for each state using the population and the crime rate in each county. At last, I will get the state code with the crime rate for each state.

   The data is found on Kaggle: https://www.kaggle.com/mikejohnsonjr/united-states-crime-rates-by-county

2. "USA Unemployment"

   This data can be downloaded as a csv file. There are 93 columns and 3275 rows which cover county, state, labor force count, unemployment count, employment count, and unemployment rate from 2000 to 2020. It also covers the data that calculate the unemployment rate for each state, so I used this data with the year of 2000 for the analysis afterwards.

   The dataset is retrieved from Kaggle: https://www.kaggle.com/valbauman/student-engagement-online-learning-supplement?select=unemployment.csv

3. "USA Education Level"

   This data can be downloaded as a csv file. There are 48 columns and 3283 rows which covers county, state, count of adults less than high school

diploma, count of adults only have high school diploma, count of adults with 3 years' college experience, count of adults with four years college or higher, percentage of adults less than high school diploma, percentage of adults only have high school diploma, percentage of adults with 3 years' college experience, percentage of adults with four years college or higher for the following years: 1970, 1980, 1990, 2000, 2019. I used county, state, count of adults less than high school diploma for each county, percentage of adults less than high school diploma for each county with the year 2000 to get the data of low-level education rate. Then, I recalculated the education rate for each state.

The data is retrieved from Kaggle: https://www.kaggle.com/valbauman/student-engagement-online-learning-supplement?select=education.csv

Data Manipulation Methods

As mentioned in the data source part, the crime rate and the education rate need to be recalculated. I used MRJob to do this preprocess and clean the data.

As for the crime rate, I first get data of county, state, crime rate of each county per 100000 people, and population of each county. Following formula is the way I recalculate the crime rate for the state:

First, I need to find the crime population for each county:

$$crime\ population = \frac{all\ population\ \times\ crime\ rate}{100000}$$

After getting the crime population for each county, I used MRJob to aggregate the rows with the same state, and then, I will get the population of a state and crime population count of a state. To get the crime rate of each state, I use the formula as following:

$$crime\ rate\ (state) = \frac{crime\ population\ (state)}{all\ population(state)}$$

This process could get the crime rate of each state. Also, I cleaned the data in the mapper step, too. Here is my code:

```
class crime_rate(MRJob):

    def mapper(self, _, line):
        data_list = line.split(',')
        county=data_list[0][1:]
        state=data_list[1][1:3]
        crime_rate=data_list[2]
        population = data_list[22]
        if population!="FIPS_ST":
            crime_pop = int(population)*float(crime_rate)/100000
            yield(state, (crime_pop,int(population)))
    def reducer(self,key,counts):
        count = list(counts)
        crime_pop=0
        population=0
        for c in count:
            crime_pop+=c[0]
            population+=c[1]
        result = crime_pop/population
        yield(key, result)
```

Here is the sample that I get after the preprocessing:

crime_output

AK	**0.004737072956971900**
AL	0.0044423324302904900
AR	0.004639496271672080
AZ	0.003916473908971540

As for the education rate recalculation, the method is a little bit different from the crime rate because the dataset I got is not the same.

I first transformed the csv file into txt file because I encountered a difficulty. The data has comma in the numbers (such as count of adults less than high school diploma in 1970 in Bibb County: 5,272), so I could not use comma to split the line. Therefore, I tried to transfer the data into txt file and use "\t" to be line split tool. As for the comma in the numbers, I replaced it with an empty string.

Then, I did some filters to get the data of state, count of adults less than high school diploma for each county (count less high school), percentage of adults only have high school diploma for each county with the year 2000 (percentage less high school). Here are the formulas that could lead us to the education rate for each state:

$$population\ per\ county = \frac{count\ less\ high\ school \times\ 100}{percentage\ less\ high\ school}$$

After getting the population for each county, I used MRJob to aggregate the rows with the same state, and then, I will get the population of a state and population less than high school of a state. This way, I can get the education rate for each state.

$$education\ rate\ (state) = \frac{less\ than\ high\ school\ population\ count(state)}{all\ population(state)}$$

Here is my code for the preprocessing:

```
class education(MRJob):

    def mapper(self, _, line):
        data_list=line.split("\t")
        state=data_list[1]
        ELevel_pop=data_list[32]
        ELevel=data_list[36]
        if ELevel!="" and ELevel!='"Percent of adults with less than a high school diploma, 2000"'\
            and ELevel_pop!="" and ELevel_pop!='"Less than a high school diploma, 2000"':
            ELevel=float(data_list[36])
            if len(ELevel_pop)>3:
                ELevel_pop=data_list[32][1:-1]
                ELevel_pop = ELevel_pop.replace(",","")
            else:
                ELevel_pop=data_list[32]
            ELevel_pop = int(ELevel_pop)
            #population = ELevel_pop*100/ELevel
            yield(state, (ELevel_pop,ELevel))

    def reducer(self, key, counts):
        count = list(counts)

        ELevel_pop=0
        pop=0
        for c in count:
            pop += ((c[0]*100)/c[1])
            ELevel_pop+=c[0]
        yield(key,ELevel_pop/pop)
```

Here is the sample that I get after the preprocessing:

edu_output

AK	**0.1168794742836800**
AL	0.24717905881024600
AR	0.24695906560527700
AZ	0.19016884591115000

Analysis and Visualization

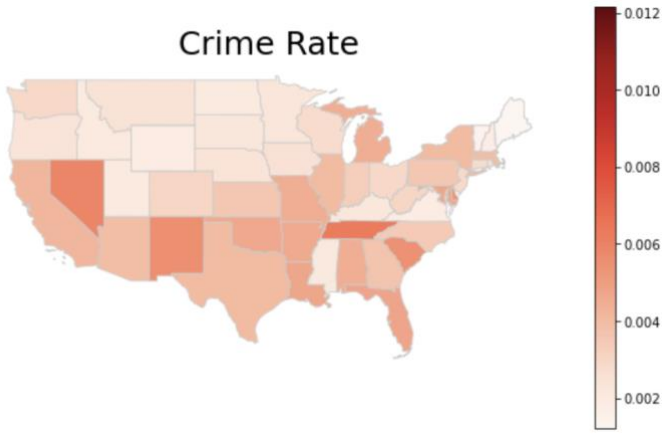
Question 1: Which state may have the highest crime rate?

I used PySpark to get the rank of the crime rate for each state and here is the result of the top 5 state crime rate:

```
+-----+--------------------+
|State|           crimeRate|
+-----+--------------------+
|   DC|0.012168013249999998|
|   TN|0.006329762816563472|
|   NV|0.005960282939488418|
|   NM|0.005570493426869...|
|   SC|0.005517882383068002|
```

I made a geoplot for each state with the colors for crime rate. First, I found the geodata for USA state plot. Then, I utilized geopandas and merge the geodata with crime rate data. At last, I drew the plot with darker color as higher crime rate in different states.

We can see that there are more dark colors in the south of the United States. It means that the higher crime rate is in the south.



Question 2: Do education rate and the crime rate have a positive correlation?

First, I utilized PySpark to merge the crime rate and education rate data. I encountered some difficulties. I uploaded the csv data from MRJob and when I wanted to use it, the data had only one column ("AL" 0.004…), but they supposed to be in two separated columns. I learned the way to separate the column into two columns. And I also removed the quotation mark before and after the state string. If I did not do so, it could not merge well. Another difficulty was that there were some spaces in the state column, I did not know the reason. I used "trim" function to get rid of the spaces.

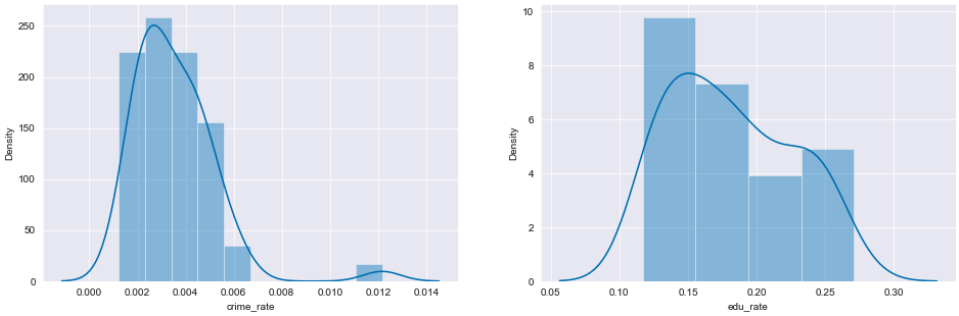
```
education = sqlContext.read.csv("edu_output.csv", header=False)
education.registerTempTable('education')
#education.printSchema()
spiltCol = f.split(education['_c0'],'\t')
education=education.withColumn('State', spiltCol.getItem(0)).withColumn('edu_rate',spiltCol.getItem(1))
education = education.withColumn('State', regexp_replace('State', '"', ''))
education.registerTempTable('education')
```

After merging crime rate and education rate data, I got data like this:

```
AK      0.004737072956971895    0.1168794742836801
AL      0.0044423324302904875   0.24717905881024624
AR      0.004639496271672083    0.24695906560527683
AZ      0.0039316473908971536   0.19016884591114996
CA      0.004189888789361677    0.2319776301713925
```

I then drew the plot below to see if there are outliers in the data. And I found an outlier in crime rate data.

```
sns.set_style("darkgrid")
plt.figure(figsize=(16,5))
plt.subplot(1,2,1)
sns.distplot(edu_crime['crime_rate'])
plt.subplot(1,2,2)
sns.distplot(edu_crime['edu_rate'])
plt.show()
```



To double check, if that data is outside mean ± three standard deviations, then it is an outlier for sure.

```
print("Highest allowed",edu_crime['crime_rate'].mean() + 3*edu_crime['crime_rate'].std())
print("Lowest allowed",edu_crime['crime_rate'].mean() - 3*edu_crime['crime_rate'].std())
```

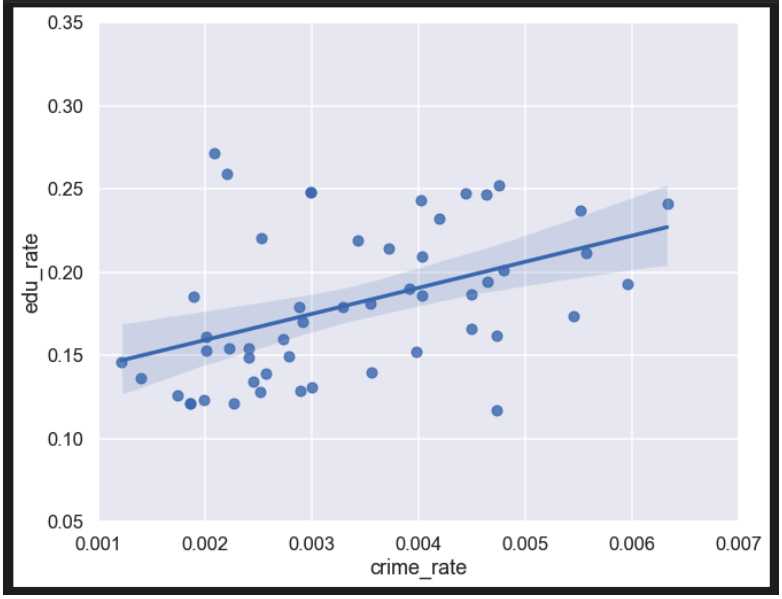
```
Highest allowed 0.008769676699524442
Lowest allowed -0.0016986922295327837
```

```
edu_crime[(edu_crime['crime_rate'] > 0.008769676699524442) | (edu_crime['crime_rate'] < -0.0016986922295327837)]
```

	state	crime_rate	edu_rate
7	DC	0.012168	0.222

I deleted the outlier before plotting. Here is the plot:

```
sns.set_style("darkgrid")
sns.set_context("talk", font_scale=1)
f, ax = plt.subplots(figsize=(10, 8))
sns.regplot(x='crime_rate', y='edu_rate', data=new_edu_crime, ax=ax);
plt.ylim(0.05, 0.35)
plt.xlim(0.001,0.007)
plt.show()
```



We can see that there is a positive correlation between education rate and crime rate. It means that if there are more people having less than high school diploma, there may be a higher crime rate. To see if this is statistically significant, I used Spearman correlation to get the p-value and correlation. The p-value is under 0.05, so it is statistically significant.

```
scipy.stats.spearmanr(x, y)
```

```
SpearmanrResult(correlation=0.4869364754098361, pvalue=0.00025158456760481223)
```

Question 3: Is unemployment rate correlated with the crime rate?

After the merging the education rate table and the crime rate table, I knew the way to deal with the same problems: separating one column into two columns, deleting the quotation marks, and trimming the string before joining table.

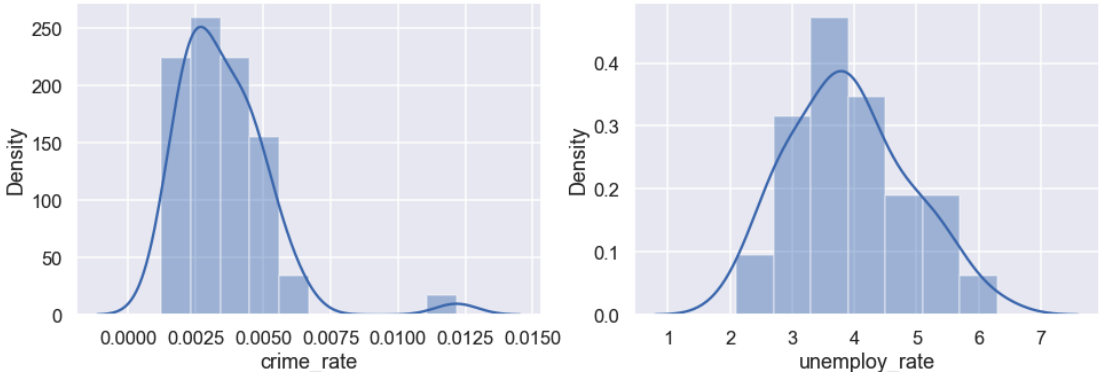
However, unemployment table was different from the table above. When I tried to join the crime table with the unemployment table, I saw that the raw data has included the unemployment rate for each state in the bottom of the data. That means I do not need to recalculate the unemployment rate again for each state. I utilized a column called FIPS_Code to filter the state data with the unemployment rate. Then, I joined the unemployment rate table with the crime rate table.

```
unemploy = sqlContext.sql("SELECT FIPS_Code, State, Unemployment_rate_2000 as unemploy_rate \
                          from unemployment where FIPS_Code like '%000' and FIPS_Code!='00000'")
unemploy.registerTempTable("unemploy")
```

After merging, I got the data like this:

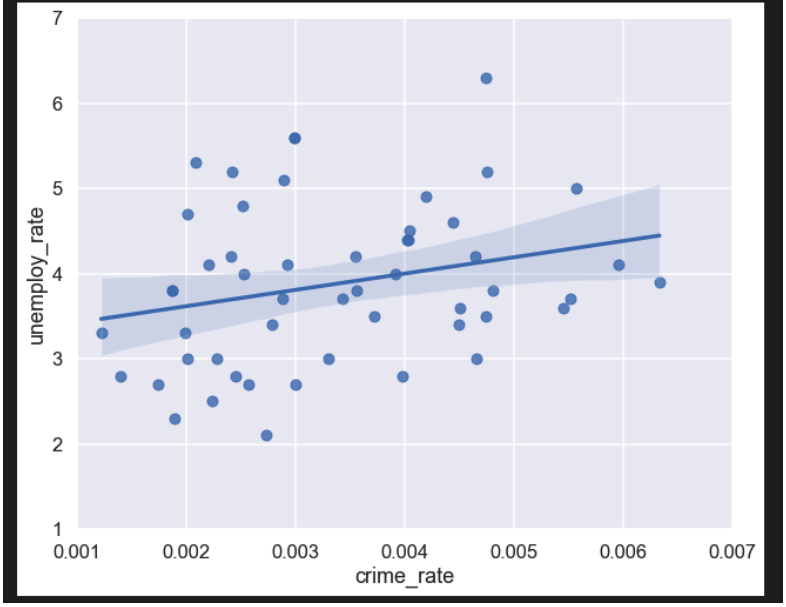
AL	0.0044423324302904875	4.6
AK	0.004737072956971895	6.3
AZ	0.003916473908971536	4.0
AR	0.004639496271672083	4.2
CA	0.004189888789361677	4.9

I used the same technique to see if there is an outlier:



Then I created a new table without the outlier and drew the plot:

```
sns.set_style("darkgrid")
sns.set_context("talk", font_scale=1)
f, ax = plt.subplots(figsize=(10, 8))
sns.regplot(x='crime_rate', y='unemploy_rate', data=new_unemploy_crime, ax=ax);
plt.ylim(1, 7)
plt.xlim(0.001,0.007)
plt.show()
```



This time, the regression line is flatter than the previous one, but it is still a positive regression line. To find out if it is statistically significant, I ran the Spearman correlation to get the p-value and correlation. The p-value is under 0.05, so it is statistically significant.

```
scipy.stats.spearmanr(x, y)
```

```
SpearmanrResult(correlation=0.29058601275483237, pvalue=0.03663244553009995)
```

Conclusion:

The southern United States may have higher crime rate. And low-level of education may be one of the factors that influences the crime rate because it is positive correlated with crime rate. Also, the unemployment rate can be another factor that slightly effects crime rate.

However, there are still many factors that will impact crime rates. It is not a one-on-one relationship, so there is still a lot of room for improvement of this project. But I have learned a lot to use the tools like MRJob and PySpark and do a real analysis with those tools. It was fun to do this project!

Challenges

I have encountered many challenges while preprocessing the data. I have mentioned all the challenges before. The first challenge was that I realized I need to recalculate the education rate and crime rate again which I had never think of while writing the proposal. I could not aggregate all the rate and divided by the number of the county for each state because the rate is related to the population of each county. And the population of each county is not the same. Therefore, I did the recalculation to solve this challenge.

The next trouble was that the output from MRJob became a same column when I uploaded into the PySpark table. I have tried many ways to separate the columns and finally succeeded. Then, I found out I need to remove the quotation mark and the spaces in the column in order to make the state columns look the same. I searched for many ways and looked for answer in the documentation of PySpark. Eventually, I got the result that I desired.