Analysis of US Population using Data Analytics and Data Science Tools

Sharan Kumar Paratala Rajagopal
Senior Manager, Capgemini America Inc. Dallas, TX, USA

Abstract: The research document explains the various Data Science tools and techniques used to analyze US population data in order to make better decisions by using data analysis of the population data for investments in schools, hospitals, establishing new job training centers and adjusting the emergency services to the size and characteristics of the demographics of metropolitan and other areas, states, or the country as a whole. There are various benefits by analyzing the data which gives holistic view for government to invest in improvements which can be made in order to serve the people better.

Keywords: Data Analytics, Data Science tools, Python, SQL

1. Introduction

The United States collects and analyzes demographic data from the U.S. population. The U.S. Census Bureau provides annual estimates of the population size of each U.S. state and region. Various decisions will be made from this data by seeing where the best fit is to invest and helps in serving the community better.

There are multiple data science tools and techniques which can be used to clean the data in the required format, analyze it, represent in the required format and visually shown to make decisions. Will introduce to some of them in the next subsequent sections and will extract the raw data from the US census website.

2. Data Science Tools

This will be a multistep process from extraction to final representation of the data. The next following sections gives a detailed description of what tools can be used to perform the specific activities. In this research article we are going to use three main tools which helps in extraction, analyzing and representing the data.

- Python
- R
- SQL

3. Extract data using python

Python is the most powerful tool which can quickly extract data from the website using web scrapping. The advantages of python is due to the inbuilt libraries which can be used to perform the action quickly and effectively.

Let’s see how we can extract the web links from the HTML code of the “Current Estimates” web link. “Figure 1” shows the python code which can be used to extract the data from the US government website [1]. “href” provides the details of the link locator to another HTML page.

“Figure 2” provides details of relative links saved as absolute URL’s in the output file.

“Figure 3” shows that there are only unique links in the output file. And any duplicates will be removed. set() function provides the unique URL from the website.

To create the csv output file
filename = "output_unique_weblink.csv"
f = open(filename,"w")
header = "Output_Weblinks\n"
4. Linear regression analysis using R

Predicting the size of the population for the state which you currently live in for 2020 based on the current estimates.

The data extracted from the US census [1] website can be cleaned up and imported into the R Integrated development environment and run the commands to interpret the slope and intercept parameters accordingly. “Figure 4” provides the R command to verify the slope and intercept parameters.

```
> lm
Call:
  lm(formula = population ~ year, data = input_r)
Coefficients:
(Intercept)          year
  439087.6           49383
```

**Figure 4:** R command for intercepting parameters

“Figure 5” provides the details of the linear regression model summary.

```
> summary(lm)
Call:
  lm(formula = population ~ year, data = input_r)
Residuals:
   Min     1Q   Median     3Q    Max
-269817  -130.7  -3552.4  -51170.0 -10031.6  26768.7  20683.1
Coefficients: Estimate Std. Error t value Pr(>|t|)    
(Intercept)  439087.6  49383  1.76e+08 <2e-16 ***
year         -857358730.1  6368  1.32e-08  0.199 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 33700 on 5 degrees of freedom
Multiple R-squared: 0.9988, Adjusted R-squared: 0.9987
F-statistic: 4754 on 1 and 5 DF, p-value: 1.21e-08
```

**Figure 5:** Summary of the linear regression.

“Figure 6” plot for the linear regression.

```
> plot(lm)
Hit <Return> to see next plot:
```

```
> abline(lm)
```

**Figure 7:** Year vs Population linear line

Use the predict command to predict the population size for the year 2020 for state Texas based on the input value and linear regression model.

Finding the coefficient parameters

```
> coef(lm)
(Intercept)          year
  439087.6           49383
```

```
> summary(lm)
Call:
  lm(formula = population ~ year, data = input_r)
Residuals:
   Min     1Q   Median     3Q    Max
-901767  -130.7  -3552.4  -51170.0 -10031.6  26768.7  20683.1
Coefficients: Estimate Std. Error t value Pr(>|t|)    
(Intercept)  439087.6  49383  1.76e+08 <2e-16 ***
year         -857358730.1  6368  1.32e-08  0.199 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 33700 on 5 degrees of freedom
Multiple R-squared: 0.9988, Adjusted R-squared: 0.9987
F-statistic: 4754 on 1 and 5 DF, p-value: 1.21e-08
```

```
> coef(lm)
(Intercept)          year
  439087.6           49383
```

```
> lm$coef[1]
(Intercept)
  439087.6
```

```
> lm$coef[2]
  year
  49383
```

```
> predict(lm, list(year = 2020))
  1
29598308
```

Predict for list of values from 2017 to 2020.
Plot the linear regression line

> predict(lm, data.frame(year=c(2017, 2018, 2019, 2020)))

1 2 3 4
2826046 28720133 29159221 29598308

“Figure 8” provides the details of predicted values for state of Texas 2020 and the linear line is plotted against year vs population.

“Figure 9” shows the histogram of the predicted population for state of Texas in 2020.

> hist(predicted_output_r$population, breaks=seq(20000000, 30000000, 1000000), main='Histogram of the population estimates for Texas (2016 Dataset)', xlab='Population (Bin Size One Million)')

“Figure 10” shows the histogram for the population estimate for Texas state based on 2016 dataset chosen for the analysis.

“Figure 11” provides the detailed output based on the data loaded into the tables.

<table>
<thead>
<tr>
<th>AREA</th>
<th>YEAR</th>
<th>POPULATION_DIFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alabama</td>
<td>2010</td>
<td>-321</td>
</tr>
<tr>
<td>Alaska</td>
<td>2010</td>
<td>-10</td>
</tr>
<tr>
<td>Arizona</td>
<td>2010</td>
<td>-14</td>
</tr>
<tr>
<td>Arkansas</td>
<td>2010</td>
<td>-99</td>
</tr>
<tr>
<td>California</td>
<td>2010</td>
<td>-134</td>
</tr>
<tr>
<td>Colorado</td>
<td>2010</td>
<td>-390</td>
</tr>
<tr>
<td>Connecticut</td>
<td>2010</td>
<td>-182</td>
</tr>
<tr>
<td>Delaware</td>
<td>2010</td>
<td>-35</td>
</tr>
<tr>
<td>District of Columbia</td>
<td>2010</td>
<td>-57</td>
</tr>
<tr>
<td>Florida</td>
<td>2010</td>
<td>792</td>
</tr>
<tr>
<td>Georgia</td>
<td>2010</td>
<td>-67</td>
</tr>
<tr>
<td>Hawaii</td>
<td>2010</td>
<td>35</td>
</tr>
<tr>
<td>Idaho</td>
<td>2010</td>
<td>-24</td>
</tr>
</tbody>
</table>

We can create multiple SQL queries to join the data set to get the output in required format. One of the query where we can find useful information is to find the differences between the states that exceed 10,000 individuals.
“Figure 12” shows the result of the SQL query and shows the states where there is individual > 10000 population.

```
SELECT nst_est2015.area, nst_est2015.year, ROUND((nst_est2016.population - nst_est2015.population) / 10000) AS DIFFERENCE
FROM nst_est2015 INNER JOIN
      nst_est2016
ON nst_est2015.area = nst_est2016.area
AND nst_est2015.year = nst_est2016.year
WHERE nst_est2015.population - nst_est2015.population > 10000
```

Now we can see one more variation in the query to provide the data in a readable data format which helps for analysis.

```
SELECT x.area, MAX(x.year, 2010, x.population) AS year_2010, MAX(x.year, 2011, x.population) AS year_2011,
       MAX(x.year, 2012, x.population) AS year_2012, MAX(x.year, 2013, x.population) AS year_2013,
       MAX(x.year, 2014, x.population) AS year_2014, MAX(x.year, 2015, x.population) AS year_2015
FROM (SELECT x.year, x.population
      FROM /etc/section_1.x)
      GROUP BY x.area
      ORDER BY x.year
```

“Figure 13” shows tabular format of data representation using SQL query. Year is transposed to column and state as rows which shows the population accordingly.

```
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Arizona</td>
<td>-10960</td>
<td>-111500</td>
<td>-112600</td>
<td>-123900</td>
<td>-130600</td>
<td>-134800</td>
</tr>
<tr>
<td>California</td>
<td>-23200</td>
<td>-49000</td>
<td>-79000</td>
<td>-111500</td>
<td>-123900</td>
<td>-130600</td>
</tr>
<tr>
<td>Florida</td>
<td>-12400</td>
<td>-16800</td>
<td>-26400</td>
<td>-30300</td>
<td>-36900</td>
<td>-39600</td>
</tr>
<tr>
<td>Georgia</td>
<td>-13500</td>
<td>-14600</td>
<td>-20900</td>
<td>-25900</td>
<td>-28600</td>
<td>-30900</td>
</tr>
<tr>
<td>Illinois</td>
<td>-19100</td>
<td>-14600</td>
<td>-20900</td>
<td>-25900</td>
<td>-28600</td>
<td>-30900</td>
</tr>
<tr>
<td>Montana</td>
<td>-21400</td>
<td>-10200</td>
<td>-12000</td>
<td>-15500</td>
<td>-18600</td>
<td>-21200</td>
</tr>
<tr>
<td>Massachusetts</td>
<td>-19000</td>
<td>-13000</td>
<td>-16000</td>
<td>-18600</td>
<td>-21200</td>
<td>-23800</td>
</tr>
<tr>
<td>New Jersey</td>
<td>-17500</td>
<td>-30300</td>
<td>-36900</td>
<td>-43600</td>
<td>-49600</td>
<td>-55600</td>
</tr>
<tr>
<td>New York</td>
<td>-18100</td>
<td>-27100</td>
<td>-34300</td>
<td>-39500</td>
<td>-44500</td>
<td>-50500</td>
</tr>
<tr>
<td>Pennsylvania</td>
<td>-13700</td>
<td>-15400</td>
<td>-17100</td>
<td>-18600</td>
<td>-20200</td>
<td>-21800</td>
</tr>
<tr>
<td>Texas</td>
<td>-13700</td>
<td>-15400</td>
<td>-17100</td>
<td>-18600</td>
<td>-20200</td>
<td>-21800</td>
</tr>
<tr>
<td>Virginia</td>
<td>-18100</td>
<td>-27100</td>
<td>-34300</td>
<td>-39500</td>
<td>-44500</td>
<td>-50500</td>
</tr>
<tr>
<td>Washington</td>
<td>-13700</td>
<td>-15400</td>
<td>-17100</td>
<td>-18600</td>
<td>-20200</td>
<td>-21800</td>
</tr>
</tbody>
</table>
```

5. Result of Analysis

Data representation and data analysis can be performed using various tools and techniques [4]. The research paper shows that the most effective tools to use for data analysis includes and not limited to R, Python and SQL. By using various commands and inbuilt libraries plots can be constructed for easy readability of the huge data set and gives the picture in a faster and better way to identify the most influential factors in a data set.

References


Author Profile

Sharan Kumar Paratala Rajagopal is a Senior Manager with Capgemini America, Inc. having 14+ years of design, development and architecture experience. He is specialized in Java/J2EE, Integration methodologies, Guidewire Product, Data Analytics, AI and Cloud technologies. He has vast domain experience in Public Services, Hospitality and Property & Casualty Insurance. Also contributed multiple technical articles to major Dev communities.