Final Project

Patkhedkar Ninad

2020-11-13

How to import and clean my data?

I have 3 datasets. Lets check and clean up dataset one by one

Dataset file - jeee16t03.csv

First We will simplify the labels as labels contains spaces and some names are too long. We am replacing spaces with underscores and changing to lower case. We will also rename 2 columns - state_and_type_of_government change to "state" - population_2016_thousands to "population_k"

```
library(knitr)
library(dplyr)
library(janitor)

options("width"=200)
options(scipen=999)  # turn-off scientific notation like 1e+48
setwd("/cloud/project/completed/final_project")

df_jee03 <- read.csv("jeee16t03.csv")
df_jee03 <- df_jee03 %>% clean_names() %>%
              rename(state = state_and_type_of_government) %>%
              rename(population_k = population_2016_thousands)
```

We will also just focusing only on records from each state. There are some records at other levels like local county govt, muncipalty and no population is provided for such records. So We will dropping all such records keeping only State govt level records.

```
df_jee03 <- df_jee03 %>% filter(population_k != "-")
df_jee03 <- df_jee03[-1,]
head(df_jee03[,1:5],10)
```

```
##                    state population_k total_direct_expenditure total_justice_system_amount total_justi
## 2          Alabama (AL)         4865                 45277563                     2335599
## 3           Alaska (AK)          742                 15808697                      962214
## 4          Arizona (AZ)         6945                 58975013                     4929687
## 5         Arkansas (AR)         2990                 27299957                     1507133
## 6       California (CA)        39209                532948138                    41714177
## 7         Colorado (CO)         5541                 57293994                     3940585
## 8      Connecticut (CT)         3579                 45649898                     2748059
## 9         Delaware (DE)          949                 11413711                      864358
## 10 District of Columbia          687                 16593661                      870775
## 11         Florida (FL)        20630                167229459                    14463341
```

First record in dataset is total of all taxes. We will drop that record. All state entries are followed by 2 letter abbreviation like Virginia (VA). We will trim these 2 letters abbreviations. Will also trim and leading and trailing spaces.

```
df_jee03$state <- sub("\\(.*", "",df_jee03$state )
trim <- function (x) gsub("^\\s+|\\s+$", "", x)
df_jee03$state <- trim(df_jee03$state)
head(df_jee03[,1:5],10)
```

```
##                   state population_k total_direct_expenditure total_justice_system_amount total_justi
## 2               Alabama         4865                 45277563                     2335599
## 3                Alaska          742                 15808697                      962214
## 4               Arizona         6945                 58975013                     4929687
## 5              Arkansas         2990                 27299957                     1507133
## 6            California        39209                532948138                    41714177
## 7              Colorado         5541                 57293994                     3940585
## 8           Connecticut         3579                 45649898                     2748059
## 9              Delaware          949                 11413711                      864358
## 10 District of Columbia          687                 16593661                      870775
## 11              Florida        20630                167229459                    14463341
```

Dataset file - jeee16t08.csv

Again will simplify columns names by changing to lower case and replacing spaces with underscores. We will change the column name "population_2016" to "popuation" as all data is of year 2016.

I will also drop firest record as it talks about Total of all states. I am focused on individual state data.

```
df_jee08 <- read.csv("jeee16t08.csv")
df_jee08 <-df_jee08 %>% clean_names() %>%
              rename(population = population_2016)

df_jee08 <-filter(df_jee08, state != "Total")
head(df_jee08[,1:5])
```

```
##        state population total_justice_system_pc police_protection_pc judicial_and_legal_pc
## 1    Alabama    4864745                  480.11               257.21                 74.43
## 2     Alaska     741504                 1297.65               499.27                342.55
## 3    Arizona    6945452                  709.77               325.62                141.59
## 4   Arkansas    2990410                  503.99               231.09                 73.68
## 5 California   39209127                 1063.89               448.11                221.27
## 6   Colorado    5540921                  711.18               338.09                136.11
```

Dataset file - jeee16t11.csv

Again will simplify columns names by changing to lower case and replacing spaces with underscores. We will drop first record as its for Total of all states. We will concentrate on statewise data.

```
df_jee11 <- read.csv("jeee16t11.csv")
df_jee11 <- df_jee11 %>% clean_names() %>% filter(state != "Total")
head(df_jee11[,1:5])
```

```
##        state tjs_total_employees tjs_full_time_employees tjs_full_time_equivalent tjs_march_payrolls
## 1    Alabama                9134                    8580                     8903              37020
## 2     Alaska                4360                    4228                     4287              27090
```

```
## 3    Arizona                 14079                   13952                   14009             56542
## 4   Arkansas                  8453                    8292                    8372             29185
## 5 California                 75822                   73341                   74779            550815
## 6   Colorado                 13878                   13317                   13770             67854
```

What does the final data set look like?

We will consolidate all datasets by joining together on "state" field.

```r
df_consolidated <- inner_join(df_jee08, df_jee03, by = "state")
df_consolidated <- inner_join(df_consolidated, df_jee11, by = "state")
```

We got 2 population fields in consolidated dataset.

population_k - represents population of state in thousands … basically round figure(k - stands for 1000) population - represents actual count of population

We will keep field which represents population in thousands as its easy to follow for analysis. We will drop other population field.

Then we will check the details of all fields in our dataframe.

```r
df_consolidated <- df_consolidated %>% select(-c(population))
str(df_consolidated)
```

```
## 'data.frame':    50 obs. of  40 variables:
##  $ state                                 : chr  "Alabama" "Alaska" "Arizona" "Arkansas" ...
##  $ total_justice_system_pc               : num  480 1298 710 504 1064 ...
##  $ police_protection_pc                  : num  257 499 326 231 448 ...
##  $ judicial_and_legal_pc                 : num  74.4 342.6 141.6 73.7 221.3 ...
##  $ corrections_pc                        : num  148 456 243 199 395 ...
##  $ total_justice_system_employment       : num  55.4 77.2 66.1 68.4 59.9 ...
##  $ police_protection_total_employment    : num  29.1 25.7 28.3 29.5 25.6 ...
##  $ police_protection_sworn_only_employment: num  23 15.6 20.4 22.1 18.3 ...
##  $ judicial_and_legal_employment         : num  9.73 20.08 15.91 11.52 11.31 ...
##  $ corrections_employment                : num  16.6 31.4 21.9 27.4 22.9 ...
##  $ population_k                          : chr  "4865" "742" "6945" "2990" ...
##  $ total_direct_expenditure              : num  45277563 15808697 58975013 27299957 532948138 ...
##  $ total_justice_system_amount           : int  2335599 962214 4929687 1507133 41714177 3940585 274
##  $ total_justice_system_percent          : num  5.2 6.1 8.4 5.5 7.8 6.9 6 7.6 8.6 7.1 ...
##  $ police_protection_amount              : int  1251270 370209 2261558 691059 17570133 1873320 1236
##  $ police_protection_percent             : num  53.6 38.5 45.9 45.9 42.1 47.5 45 40.3 54.3 46.4 ...
##  $ judician_and_legal_amount             : int  362060 254000 983419 220343 8675761 754162 826903 2
##  $ judicial_and_legal_percent            : num  15.5 26.4 19.9 14.6 20.8 19.1 30.1 24.1 16.4 20.8 .
##  $ corrections_amount                    : int  722269 338005 1684710 595731 15468283 1313103 68415
##  $ corrections_percent                   : num  30.9 35.1 34.2 39.5 37.1 33.3 24.9 35.7 29.4 32.8 .
##  $ tjs_total_employees                   : int  9134 4360 14079 8453 75822 13878 14197 5879 48022 2
##  $ tjs_full_time_employees               : int  8580 4228 13952 8292 73341 13317 13574 5778 46826 2
##  $ tjs_full_time_equivalent              : int  8903 4287 14009 8372 74779 13770 13725 5843 47381 2
##  $ tjs_march_payrolls                    : int  37020 27090 56542 29185 550815 67854 77831 28123 16
##  $ tjs_average_earnings                  : int  4177 6357 4028 3465 7394 4984 5640 4836 3584 3221 .
##  $ pp_total_employees                    : int  1303 683 1963 1223 11444 1274 2142 1100 4410 2625 .
##  $ pp_full_time_employees                : int  1284 640 1919 1207 11176 1257 1905 1088 4098 2559 .
##  $ pp_full_time_equivalent               : int  1291 650 1932 1214 11216 1265 1938 1095 4206 2593 .
##  $ pp_march_payrolls                     : int  5148 4290 9973 4697 94064 7559 14229 7277 16616 102
##  $ pp_average_earnings                   : chr  "3987" "6649" "5163" "3877" ...
```

```
## $ jl_total_employees                 : int  3167 1406 2416 1667 6569 5191 6221 1835 19872 3571
## $ jl_full_time_employees             : int  2855 1366 2346 1528 6127 4702 5904 1786 19181 3481
## $ jl_full_time_equivalent            : int  3048 1382 2383 1599 6329 5096 5988 1818 19544 3521
## $ jl_march_payrolls                  : int  14292 9242 11654 6464 44374 28588 29355 8543 81600
## $ jl_average_earnings                : int  4777 6709 4868 3949 7061 5816 4826 4730 4190 4559 .
## $ c_total_employees                  : int  4664 2271 9700 5563 57809 7413 5834 2944 23740 16587
## $ c_full_time_employees              : int  4441 2222 9687 5557 56038 7358 5765 2904 23547 15792
## $ c_full_time_equivalent             : int  4564 2255 9694 5559 57234 7409 5799 2930 23631 16184
## $ c_march_payrolls                   : int  17580 13558 34915 18024 412377 31707 34247 12303 715
## $ c_average_earnings                 : int  3847 6056 3600 3242 7230 4281 5897 4216 3028 2807 .
```

Questions for future steps.

Considering the questions we want to find answer for, its required to identify correct variables. Currently there are 41 variables after joining the datasets.

I have identfied below variables which we would use. However based on how analysis goes, we may need to add or drop some variables.

- state
- population_k
- total_direct_expenditure
- police_protection_amount
- total_justice_system_amount
- total_justice_system_pc
- tjs_total_employees
- tjs_full_time_equivalent
- tjs_average_earnings

What information is not self-evident?

There is no crime rate related data in datasets. It can be assumed that Police protection functions cost more in states having large metro areas with high crime rate.But its not clear if civil services expense are also high in such states. We will try to establish correlation between spending on police protection and civil services.

What are different ways you could look at this data?

I plan to perform linear regression and correlation analysis to find some variables which may have impact expenses.We will also explore clustering based on police costs.

How do you plan to slice and dice the data?

Yes. Dataset is already created by joining two datsets. We may need to further derive employment related variable by combining full time and part time data.

How could you summarize your data to answer key questions?

```r
library(skimr)
#skim(df_consolidated)
summary(df_consolidated)
```

```
##     state           total_justice_system_pc police_protection_pc judicial_and_legal_pc corrections_pc
##  Length:50          Min.   : 450.1          Min.   :160.3        Min.   : 72.35        Min.   :141.9
##  Class :character   1st Qu.: 556.4          1st Qu.:258.9        1st Qu.:109.11        1st Qu.:177.0
##  Mode  :character   Median : 662.1          Median :292.1        Median :130.77        Median :207.1
##                     Mean   : 678.5          Mean   :311.0        Mean   :140.81        Mean   :226.7
##                     3rd Qu.: 738.7          3rd Qu.:347.4        3rd Qu.:158.88        3rd Qu.:251.9
##                     Max.   :1297.7          Max.   :505.2        Max.   :342.55        Max.   :455.8
##  police_protection_sworn_only_employment judicial_and_legal_employment corrections_employment popula
##  Min.   :13.68                           Min.   : 7.29                 Min.   :13.60          Length
##  1st Qu.:17.54                           1st Qu.:10.78                 1st Qu.:17.52          Class
##  Median :20.94                           Median :12.38                 Median :20.01          Mode
##  Mean   :20.87                           Mean   :13.21                 Mean   :21.10
##  3rd Qu.:22.70                           3rd Qu.:15.39                 3rd Qu.:24.61
##  Max.   :38.35                           Max.   :23.58                 Max.   :33.34
##  police_protection_amount police_protection_percent judician_and_legal_amount judicial_and_legal_perc
##  Min.   :  188210         Min.   :31.70             Min.   :  80521          Min.   :13.00
##  1st Qu.:  458832         1st Qu.:40.98             1st Qu.: 259228          1st Qu.:18.02
##  Median : 1244134         Median :46.20             Median : 566624          Median :19.95
##  Mean   : 2172335         Mean   :46.12             Mean   : 922489          Mean   :20.48
##  3rd Qu.: 2460940         3rd Qu.:49.17             3rd Qu.:1022504          3rd Qu.:22.85
##  Max.   :17570133         Max.   :60.50             Max.   :8675761          Max.   :30.30
##  tjs_full_time_equivalent tjs_march_payrolls tjs_average_earnings pp_total_employees pp_full_time_emp
##  Min.   : 1778            Min.   :  7912     Min.   :3221         Min.   :    0.0    Min.   :    0.0
##  1st Qu.: 5070            1st Qu.: 24260     1st Qu.:4016         1st Qu.:  777.8    1st Qu.:  756.2
##  Median : 9422            Median : 39736     Median :4766         Median : 1438.0    Median : 1331.5
##  Mean   :14355            Mean   : 72745     Mean   :4793         Mean   : 2065.6    Mean   : 2007.2
##  3rd Qu.:18544            3rd Qu.: 81625     3rd Qu.:5366         3rd Qu.: 2597.2    3rd Qu.: 2530.0
##  Max.   :74779            Max.   :550815     Max.   :7394         Max.   :11444.0    Max.   :11176.0
##  jl_full_time_employees jl_full_time_equivalent jl_march_payrolls jl_average_earnings c_total_employe
##  Min.   :  470          Min.   :  470           Min.   :  2594    Min.   :3492        Min.   :  799
##  1st Qu.: 1214          1st Qu.: 1228           1st Qu.:  6668    1st Qu.:4831        1st Qu.: 2851
##  Median : 2278          Median : 2323           Median : 12991    Median :5444        Median : 5449
##  Mean   : 3415          Mean   : 3504           Mean   : 19825    Mean   :5691        Mean   : 8895
##  3rd Qu.: 3710          3rd Qu.: 3844           3rd Qu.: 21413    3rd Qu.:6236        3rd Qu.:12116
##  Max.   :19181          Max.   :19544           Max.   :140023    Max.   :9481        Max.   :57809
```

For some reason, skim output is not showing up in R Markdown. However summary data is good enough.

What types of plots and tables will help you to illustrate the findings to your questions?

Scatter plots, Histograms and Cluster plots will help with findings on questions.

Do you plan on incorporating any machine learning techniques to answer your research questions? Explain.

Yes, I plan to use k clustering for classification of states on some metrics like salaries etc.