STAT 5014 Homework 5

Jingbin Xu

10/23/2020

Problem 3

1. How many data points were there in the complete dataset?

There are 886930 rows in the complete dataset with 70 variables.

```
## -- Attaching packages ------------------------------------------------------------------------ tidyver
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts --------------------------------------------------------------------------- tidyverse_con
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
## Warning: Missing column names filled in: 'X70' [70]
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   `Country Name` = col_character(),
##   `Country Code` = col_character(),
##   `Indicator Name` = col_character(),
##   `Indicator Code` = col_character(),
##   `2015` = col_logical(),
##   `2016` = col_logical(),
##   `2017` = col_logical(),
##   `2020` = col_logical(),
##   `2025` = col_logical(),
##   `2030` = col_logical(),
##   `2035` = col_logical(),
##   `2040` = col_logical(),
##   `2045` = col_logical(),
##   `2050` = col_logical(),
##   `2055` = col_logical(),
##   `2060` = col_logical(),
##   `2065` = col_logical(),
##   `2070` = col_logical(),
##   `2075` = col_logical(),
##   `2080` = col_logical()
##   # ... with 5 more columns
## )
```

```
## See spec(...) for full column specifications.
```

```
## Warning: 876291 parsing failures.
##  row  col           expected           actual                          file
## 2484 2015 1/0/T/F/TRUE/FALSE 62.4392794473723 'Edstats_csv/EdstatsData.csv'
## 4908 2015 1/0/T/F/TRUE/FALSE 21057820874350.2 'Edstats_csv/EdstatsData.csv'
## 4908 2016 1/0/T/F/TRUE/FALSE 21923168354725.3 'Edstats_csv/EdstatsData.csv'
## 4909 2015 1/0/T/F/TRUE/FALSE 21766948388560.2 'Edstats_csv/EdstatsData.csv'
## 4909 2016 1/0/T/F/TRUE/FALSE 22480427869996.2 'Edstats_csv/EdstatsData.csv'
## .... .... .................. ................ .............................
## See problems(...) for more details.
```

```
# compute the dimension of the raw dataset
dim(dt) # 886930 rows with 70 columns
```

```
## [1] 886930     70
```

2. In your clean data?

There are 4060128 data points in the clean dataset.

```
# remove the irrelevant column
delete_column <- c(3:4,50:70)
# retrieve the clean dataset
dt.clean <- dt[, -delete_column]
# head(dt.clean)
# remove the missing values in the complete dataset
df <- gather(dt.clean, key = "Year", value = "Value", 3:47, na.rm = T)
# rename the columns
colnames(df) <- c("name", "code", "year", "value")
```

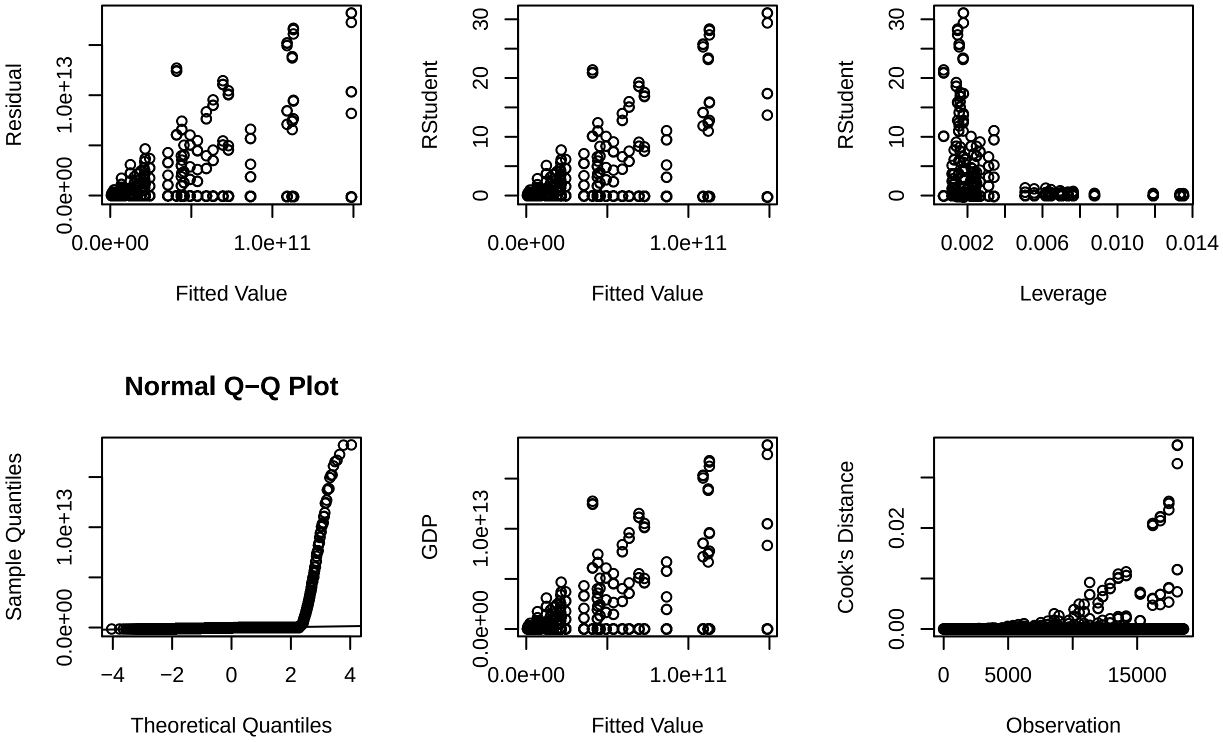
3. Creat a summary table of indicators for comparison of 2 countries.

```
# compute the summary statistics for country code ARB and EAS
t <- df %>%
  filter(get('code') %in% c("ARB", "EAS") ) %>%
  group_by(code) %>%
  summarise("Mean" = mean(value), "Standard Deviation" = sd(value), .groups = "drop")
# table the ARB and EAS for display
knitr::kable(t[1:2,])
```

code	Mean	Standard Deviation
ARB	28530367997	2.958890e+11
EAS	223312839937	1.928374e+12

Problem 4

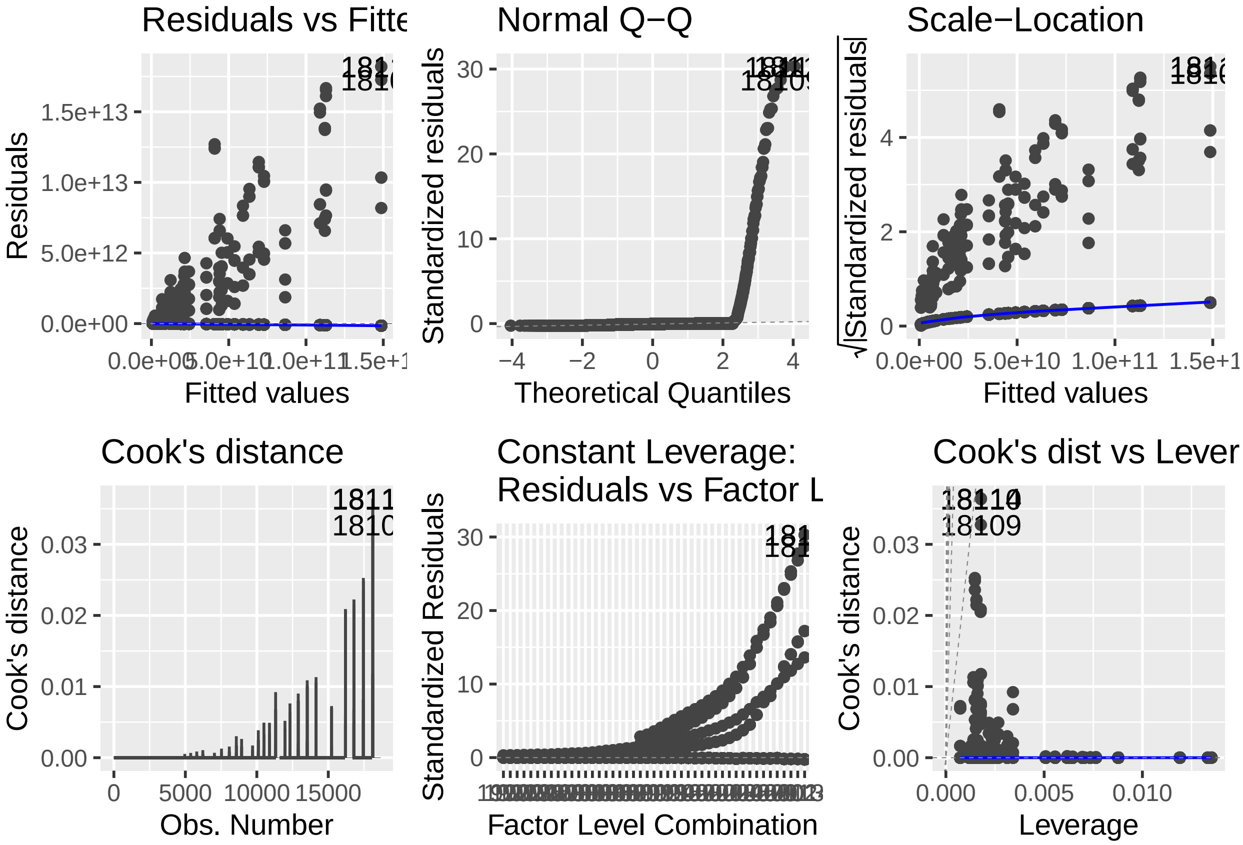
```
# select the CHN gdp
CHN <- df %>%
  filter(get('code') == "CHN")
# fit the linear regression model on CHN gdp
lmfit <- lm(value ~ year, data = CHN)
# plot the plots with 2 rows and 3 columns
par(mfrow = c(2, 3))
# residual vs fitted value
plot(lmfit$fitted.values, lmfit$residuals, xlab = "Fitted Value", ylab = "Residual")
# R studentized residual vs fitted value
plot(lmfit$fitted.values, studres(lmfit), xlab = "Fitted Value", ylab = "RStudent")
# leverage point vs r studentized residual
plot(hatvalues(lmfit), studres(lmfit), xlab = "Leverage", ylab = "RStudent")
qqnorm(lmfit$residuals)
qqline(lmfit$residuals)
# fitted value with the CHN gdp value
plot(lmfit$fitted.values, CHN$value, xlab = "Fitted Value", ylab = "GDP")
# cooks distance with the observations
plot(cooks.distance(lmfit), ylab = "Cook's Distance", xlab = "Observation")
```



Problem 5

```
# using plackage of ggplot extension to plot the aboved plots.
autoplot(lmfit, which = 1:6, nrow = 2)
```

```
## Warning: `arrange_()` is deprecated as of dplyr 0.7.0.
## Please use `arrange()` instead.
## See vignette('programming') for more help
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```


Residuals vs Fitte
Residuals
Fitted values
0.0e+00
5.0e+12
1.0e+13
1.5e+13
0.0e+05.0e+101.0e+111.5e+1
181
181(
Normal Q−Q
Standard residuals
Theoretical Quantiles
0
10
20
30
−4
−2
0
2
4
1811
18109
Scale−Location
√|Standardized residuals|
Fitted values
0
2
4
0.0e+00
5.0e+10
1.0e+11
1.5e+1
181
181
Cook's distance
Cook's distance
Obs. Number
0.00
0.01
0.02
0.03
0
5000
10000
15000
1811
1810
Constant Leverage:
Residuals vs Factor L
Standardized Residuals
Factor Level Combination
0
10
20
30
181
181
Cook's dist vs Lever
Cook's distance
Leverage
0.00
0.01
0.02
0.03
0.000
0.005
0.010
18110
18109