# HW5_wgeither

Warren Geither

11/2/2020

## Problem 3

How many data points were there in the complete dataset? In your cleaned dataset? - original: 886,930 X 70 - clean: 4,825,021 X 6

```
# load data
#bank_data <- fread("Edstats_csv/EdStatsData.csv", header=TRUE)
#saveRDS(bank_data, "bank_data_raw.RDS")
bank_df <- readRDS("bank_data_raw.RDS")

# create column names
col_names <- c("Country Name"
                ,"Country Code"
                ,"Indicator Name"
                ,"Indicator Code"
                , seq(1977, 2020, by = 1)
                , seq(2025, 2100, by = 5))

# get columns that need fixin
years_columns <- bank_df[,5:64]

# create a vector of the year columns
years <- as.character(c(seq(1977, 2020, by = 1)
                        , seq(2025, 2100, by = 5)))

# consolidate those values into 1 column
years_fixed <- years_columns %>% gather(key="Year"
                              , value=years)

# remove na values
test_df <- years_fixed[complete.cases(years_fixed), ]

# bind together with previous df
fixed_df <- cbind(bank_df[,1:4], test_df)
```

```
## Warning in as.data.table.list(x, keep.rownames = keep.rownames, check.names =
## check.names, : Item 1 has 886930 rows but longest item has 4825021; recycled
## with remainder.
```

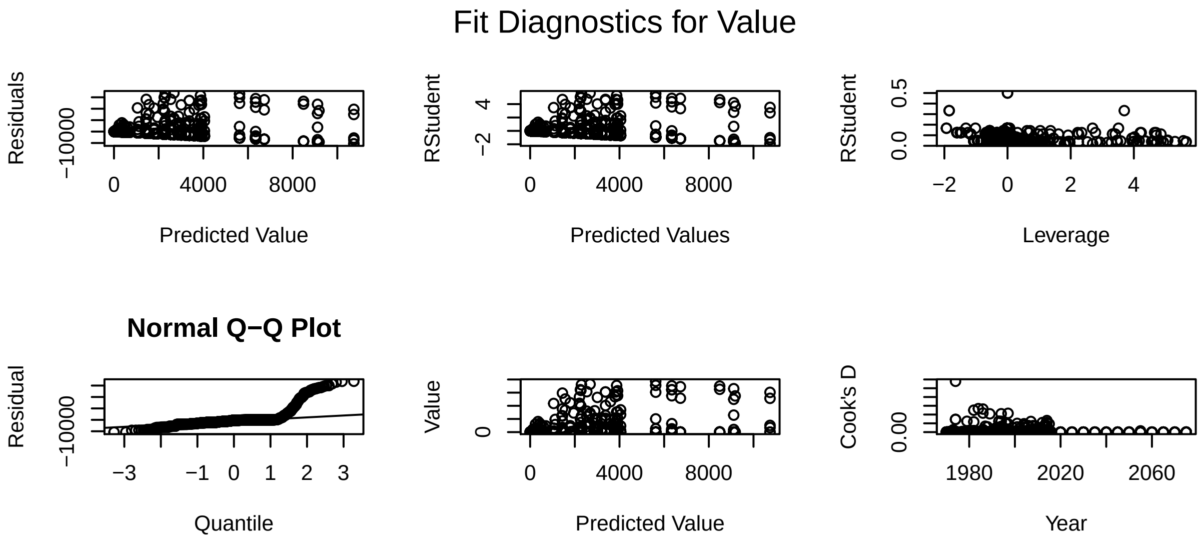
```r
# fix column names
colnames(fixed_df) <- c("Country_Name"
                        , "Country_Code"
                        , "Indicator_Name"
                        , "Indicator_Code"
                        , "Year"
                        , "Value")

# pick 2 countries and an indicator
small_df <- fixed_df %>% filter(Country_Name == "India" | Country_Name == "France") %>%
                         filter(Indicator_Code == "LO.LLECE.MAT6.1.MA")

# apply summary to each country
sum_table <- tapply(small_df$Value, small_df$Country_Code, summary)

# couldnt get knitr to work with array so making values from sum_table into df
sum_table1 <- data.frame(Min = c(0,20.3), Q1 = c(0.5,66.6)
                         , Median = c(5.3,58312)
                         , Mean = c(91594.9, 441691.4)
                         , Q3 = c(39.6, 173272.0)
                         , Max = c(549509.0,1976786.0))

# fix rownames
rownames(sum_table1) <- c("France", "India")

# print pretty table
knitr::kable(sum_table1)
```

| | Min | Q1 | Median | Mean | Q3 | Max |
|---|---|---|---|---|---|---|
| France | 0.0 | 0.5 | 5.3 | 91594.9 | 39.6 | 549509 |
| India | 20.3 | 66.6 | 58312.0 | 441691.4 | 173272.0 | 1976786 |

## Problem 4

```r
# look at more data
test_df <- fixed_df %>% filter(Indicator_Code == "LO.LLECE.MAT6.1.MA")

# remove outliers
testtest_df <- test_df %>% filter(Value < 40000)

# model
lmfit <- lm(Value~Year, data = testtest_df)

# studentized residuals
studentized <- rstandard(lmfit)

# calculate leverage
leverage <- lm.influence(lmfit)$hat

# set plot matrix
```

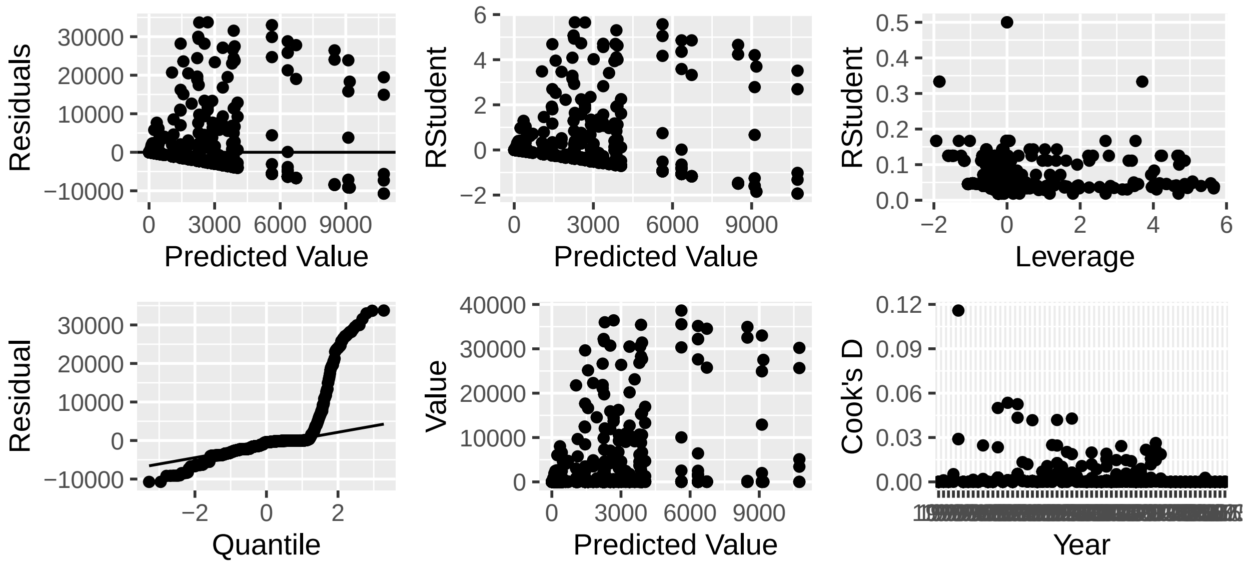
```r
par(mfrow = c(3,3))

# residual plot
plot(x = fitted(lmfit)
     , y = residuals(lmfit)
     , xlab = "Predicted Value"
     , ylab = "Residuals")

# studenttized residuals
plot(x = fitted(lmfit)
     , y = studentized
     , xlab = "Predicted Values"
     , ylab = "RStudent")

# studentized residuals vs leverage
plot(studentized
     , leverage
     , xlab = "Leverage"
     , ylab = "RStudent")

# residual QQ plot
qqnorm(lmfit$res
       , xlab = "Quantile"
       , ylab = "Residual")
qqline(lmfit$res)

# weight vs prediced
plot(x = fitted(lmfit)
     , y = testtest_df$Value
     , xlab = "Predicted Value"
     , ylab = "Value")

# cooks distance
plot(x = testtest_df$Year
     , y = cooks.distance(lmfit)
     , xlab = "Year"
     , ylab = "Cook's D")

mtext("Fit Diagnostics for Value", side = 3, line = -2, outer = TRUE)
```



## Problem 5

```r
library(ggpubr)
# residual plot
g1 <- ggplot(testtest_df
       , aes(x = fitted(lmfit), y = residuals(lmfit))) +
  geom_point() +
  geom_hline(yintercept=0) +
  xlab("Predicted Value") +
  ylab("Residuals")

# studenttized residuals
g2 <- ggplot(testtest_df
       , aes(x = fitted(lmfit), y = studentized)) +
  geom_point() +
  xlab("Predicted Value") +
  ylab("RStudent")

# studentized residuals vs leverage
g3 <- ggplot(testtest_df
       , aes(x = studentized, y = leverage)) +
  geom_point() +
  xlab("Leverage") +
  ylab("RStudent")
```

```r
# residual QQ plot
g4 <- ggplot(lmfit, aes(sample=residuals(lmfit)))+
  stat_qq() +
  stat_qq_line() +
  xlab("Quantile") +
  ylab("Residual")

# weight vs predicted
g5 <- ggplot(testtest_df
       , aes(x = fitted(lmfit), y = Value)) +
  geom_point() +
  xlab("Predicted Value") +
  ylab("Value")

# cooks distance
g6 <- ggplot(testtest_df
       , aes(x = Year, y = cooks.distance(lmfit))) +
  geom_point() +
  xlab("Year") +
  ylab("Cook's D")

# arrange on same page
ggarrange(g1, g2, g3, g4, g5, g6,
          ncol = 3, nrow = 3)
```



Recreate the plot in problem 3 using ggplot2 functions. Note: there are many extension libraries for ggplot, you will probably find an extension to the ggplot2 functionality will do exactly what you want.

## Problem 6

Finish this homework by pushing your changes to your repo.

**Only submit the .Rmd and .pdf solution files. Names should be formatted HW5_lastname_firstname.Rmd and HW5_lastname_firstname.pdf**