

Project 1: Visualizations

Eli (Ilya) Bolotin

9/21/2019

Load libraries

```
if (!require("pacman")) install.packages("pacman")
pacman::p_load(ggplot2, dplyr, reshape, ggrepel)
```

Load and transform data

```
# Load data
world_stats <- read.csv("data.csv")

# Drop columns, rows
world_stats <- select(world_stats, -c(Series.Code, Country.Code))
world_stats <- world_stats[-c((nrow(world_stats)-5):nrow(world_stats)),]

# Rename year columns
cols_containing_x <- names(world_stats[, grepl("X", names(world_stats))])
new_names <- substring(cols_containing_x, 2, 5)
colnames(world_stats)[colnames(world_stats) %in% cols_containing_x] <- new_names

# Reshape data
md <- melt(world_stats, id=c("Country.Name", "Series.Name"))
world_stats <- cast(md, "Country.Name+variable~Series.Name")
world_stats <- world_stats[,-c(5,12:16)]

# Rename metrics
colnames(world_stats)[c(2,3,4,5,6,7,8,9,10,11)] <-
  c("year", "adolescent_fertility", "gdp", "life_expectancy", "mortality_rate", "net_migration", "prem", "pop_growth")

# Convert factors to numeric
as.numeric.factor <- function(x) {as.numeric(levels(x))[x]}
world_stats_numerics <- lapply(world_stats[,c(3:dim(world_stats)[2])], as.numeric.factor)
world_stats_numerics <- as.data.frame(world_stats_numerics)
world_stats <- cbind(world_stats[,c(1,2)], world_stats_numerics)

# Change units of GDP
world_stats$gdp <- world_stats$gdp/(1000000000)
```

Add corruption index to data

```
# Create corruption index lookup table
ci_df <- read.csv("cpi.csv", sep=";")
ci_df <- ci_df[,-c(1,2,4,7)]
ci_df <- ci_df[ci_df$time==2018,c(1,3)]
colnames(ci_df) <- c("Country.Name", "cpi")
ci_df$Country.Name <- as.character(ci_df$Country.Name)
```

Synchronize names between tables

```
# Get country names from world_stats df and corruption index df
countries_ws <- levels(droplevels(world_stats$Country.Name))
countries_ci <- ci_df$Country.Name

already_renamed = list()
idx = 0
'%ni%' <- Negate('%in%')

# Loop through world_stats df and synchronize differing country names
for(country in countries_ws) {
  j = agrep(country, countries_ci, ignore.case = FALSE, value = FALSE, max.distance = 0.1)
  if(length(j) == 1) {
    levels(world_stats$Country.Name)[levels(world_stats$Country.Name) == country] <-
      ↪ ci_df$Country.Name[j]
    already_renamed[idx] <- ci_df$Country.Name[j]
  }
  idx = idx + 1
}

# Loop through corruption_index df and synchronize differing country names
for(i in 1:nrow(ci_df)) {
  if(ci_df$Country.Name[i] %ni% already_renamed) {
    j = agrep(ci_df$Country.Name[i], countries_ws, ignore.case = FALSE, value = FALSE, max.distance
    ↪ = 0.1)
    if(length(j) == 1) {
      ci_df$Country.Name[i] <- countries_ws[j]
    }
  }
}
```

Join tables (economic and corruption data) and create categorical features

```
# Create joined table
world_stats <- full_join(world_stats, ci_df, by = "Country.Name")

# Update world_stats table with corruption level
world_stats <- world_stats %>% mutate(ci_rating = case_when(
  cpi >= 120 ~ "very corrupt",
  cpi < 120 & cpi >= 70 ~ "corrupt",
  cpi < 70 & cpi >= 50 ~ "somewhat corrupt",
  cpi < 50 & cpi >= 25 ~ "clean",
  cpi < 25 ~ "very clean"))

world_stats <- world_stats %>% mutate(mortality_cat = case_when(
  mortality_rate <= 15 ~ "low",
  mortality_rate > 15 & mortality_rate <= 30 ~ "medium",
  mortality_rate > 30 ~ "high"))

# Drop NAs
world_stats <- world_stats[which(!is.na(world_stats$gdp) & !is.na(world_stats$cpi) &
  ↪ !is.na(world_stats$mortality_cat)),]

# Create df for 2018
world_stats_2018 = world_stats[world_stats$year==2018,]
```

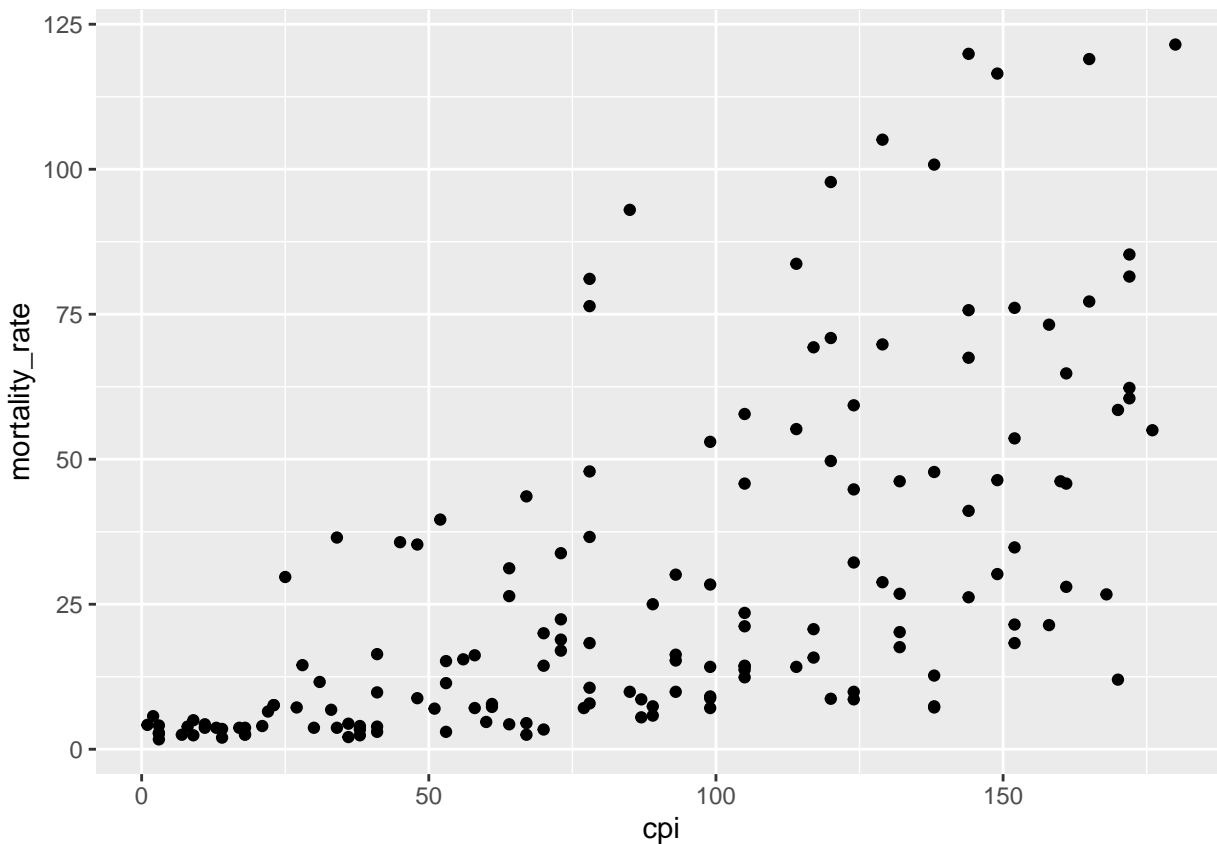
```
world_stats_2018$ci_rating <- factor(world_stats_2018$ci_rating, levels = c("very clean", "clean",  
  ↪ "somewhat corrupt", "corrupt", "very corrupt"))
```

Visualization 1

In the first visualization, mortality rate is plotted against corruption to get a sense of correlation between the two variables.

Code:

```
ggplot(data = world_stats_2018)+  
  geom_point(aes(x=cpi, y=mortality_rate)) +  
  scale_x_continuous(labels = scales::comma)
```



Visualization 2

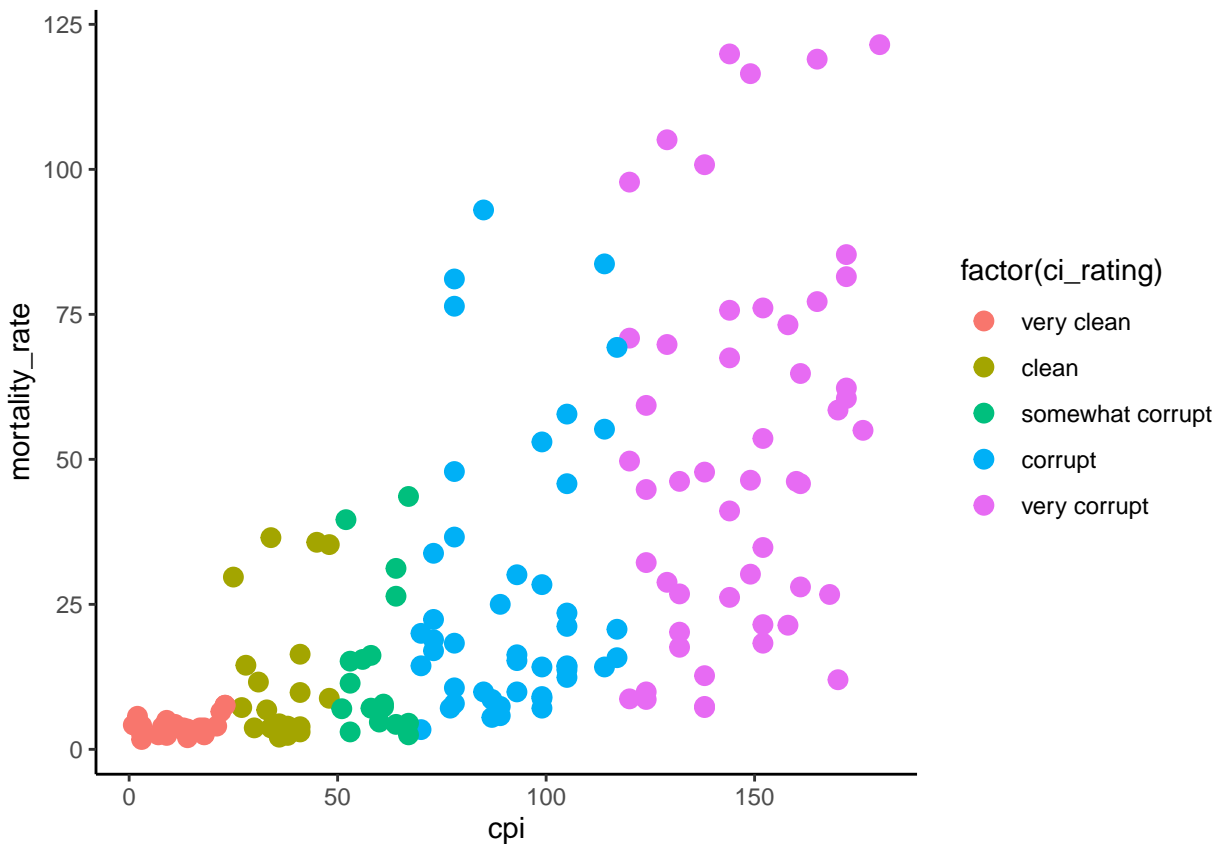
In the second visualiation:

- Color is introduced to segment the corruption rating of different countries
- Point size is increased for readability
- The graph grid is removed and axes cleaned up

Code:

```
p <- ggplot(data = world_stats_2018, aes(x=cpi, y=mortality_rate))  
p + geom_point(aes(colour = factor(ci_rating)), size = 3) +
```

```
theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
panel.grid.minor = element_blank(), axis.line = element_line(colour = "black"))
```



Visualization 3

The final visualization:

- Adds title and subtitle
- Adds X and Y-axis titles
- Adjusts size of X and Y-axis labels
- Adds size dimension to points, indicating GDP of country
- Adds annotations of countries
- Adds red color scheme indicating level of mortality (darker=higher mortality)
- Lightened XY axes

Code:

```
p <- ggplot(data = world_stats_2018, aes(x=cpi, y=mortality_rate))
p + geom_point(aes(colour = factor(ci_rating), size = gdp)) +
  scale_size("GDP", breaks=c(5000,10000,15000,20000), labels=c("$5t", "$10t", "$15t", "$20t")) +
  scale_x_continuous(breaks=seq(0,190,25), limits=c(0,190)) +
  scale_colour_brewer(palette = "Reds",
    direction=1,
    name = "Corruption Ranking",
    labels = c("Very clean", "Clean", "Somewhat corrupt", "Corrupt", "Very
    ↪ corrupt")) +
  geom_text_repel(
    data = subset(world_stats_2018, Country.Name %in% c("United States of
    ↪ America", "Canada", "China", "Brazil", "Portugal", "Italy", "India", "Denmark", "Russian
    ↪ Federation", "Angola", "Afghanistan")),
```

```

aes(label=Country.Name),
nudge_y = 55,
direction = "y",
angle = 0,
color = "brown4",
segment.color = "grey88",
segment.size = .2,
segment.alpha = .8,
size = 3,
point.padding = .25,
arrow = arrow(length=unit(0.025,"npc"), type="closed"),
vjust = 0) +
theme_bw() + theme(panel.border = element_blank(),
                    panel.grid.major = element_blank(),
                    panel.grid.minor = element_blank(),
                    axis.line = element_line(colour = "black", size=.05, linetype="solid")) +
labs(title="Global Infant Mortality vs Corruption",
     subtitle = "Data collected from World Bank and Corruption Perceptions Index",
     x = "Corruption Ranking",
     y = "Infant mortality Rate (in 1000 live births)") +
theme(plot.title = element_text(size=18),
      plot.subtitle = element_text(size=10),
      axis.text.x = element_text(size=9),
      axis.text.y = element_text(size=9),
      axis.title.x = element_text(size=8),
      axis.title.y = element_text(size=8))

```

Global Infant Mortality vs Corruption

Data collected from World Bank and Corruption Perceptions Index

