Lab 2

October 17, 2017

Intro

This second lab takes an applied project approach to teaching basic data manipulation with `dplyr` and graphing using `ggplot2`.

This lab was adapted from http://tutorials.iq.harvard.edu/R/Rgraphics/Rgraphics.html. It has been changed to conform with the `tidyverse` package and methods of data manipulation, and updated from the 2012 copy to reflect package changes.

Datasets are available on the DHS-OEDA GitHub page here: https://github.com/DHS-OEDA/r_training

Objective

The goal of this lab is to re-create this image from The Economist. This will demonstrate how to turn the default graphs produced in R into publication-quality images.

The problem is that many publication quality graphs are post-processed in programs like Adobe Illustrator or Adobe InDesign (in the case of plots in magazines/newspapers). This is an R Training, so let's just use R and some extra packages.

Getting started

Data

```
econ_data <- read_csv('data/lab_2/EconomistData.csv')
```

```
## Warning: Missing column names filled in: 'X1' [1]
```

```
## Parsed with column specification:
## cols(
##   X1 = col_integer(),
##   Country = col_character(),
##   HDI.Rank = col_integer(),
##   HDI = col_double(),
##   CPI = col_double(),
##   Region = col_character()
## )
```

```
head(econ_data)
```

```
## # A tibble: 6 x 6
##      X1     Country HDI.Rank   HDI   CPI            Region
##   <int>       <chr>    <int> <dbl> <dbl>             <chr>
## 1     1 Afghanistan      172 0.398   1.5      Asia Pacific
## 2     2     Albania       70 0.739   3.1 East EU Cemt Asia
## 3     3     Algeria       96 0.698   2.9              MENA
## 4     4      Angola      148 0.486   2.0               SSA
## 5     5   Argentina       45 0.797   3.0          Americas
```

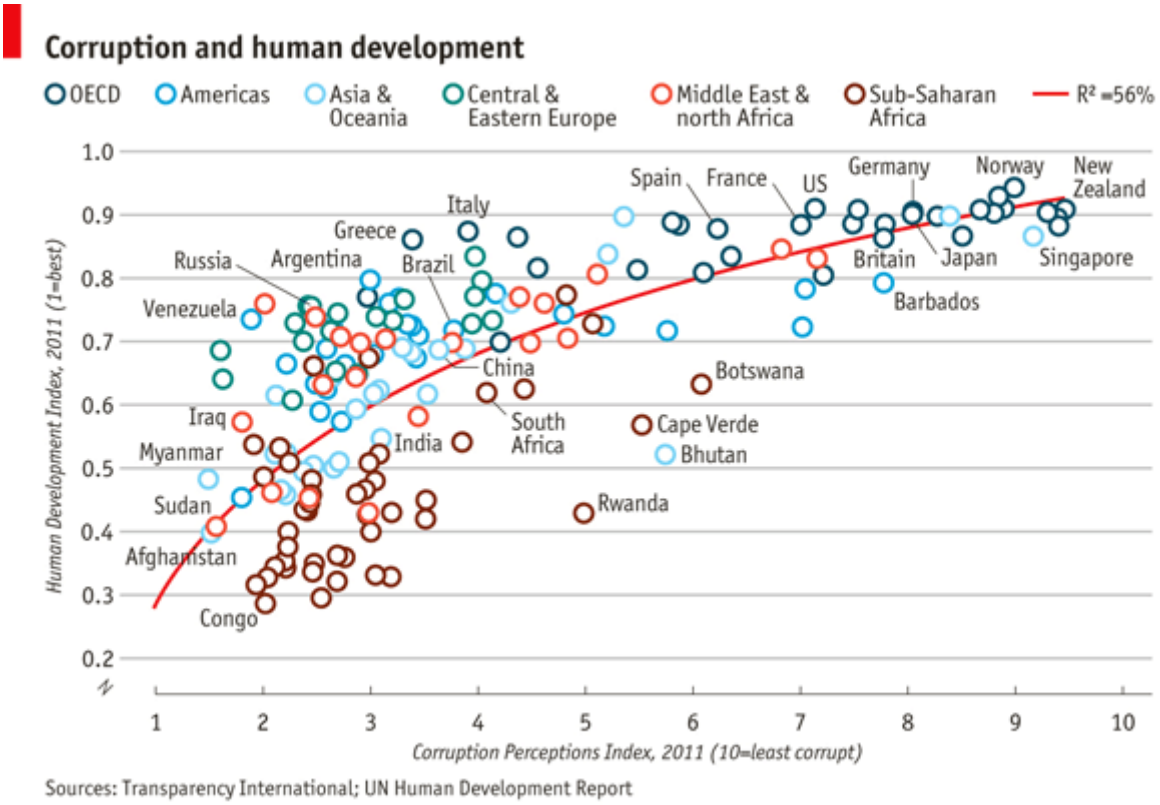

Corruption and human development
OECD
Americas
Asia & Oceania
Central & Eastern Europe
Middle East & north Africa
Sub-Saharan Africa
R² =56%
Human Development Index, 2011 (1=best)
Corruption Perceptions Index, 2011 (10=least corrupt)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
1
2
3
4
5
6
7
8
9
10
Spain
France
US
Germany
Norway
New Zealand
Italy
Greece
Russia
Argentina
Brazil
Britain
Japan
Singapore
Venezuela
Barbados
China
Botswana
Iraq
South Africa
Cape Verde
India
Bhutan
Myanmar
Sudan
Rwanda
Afghanistan
Congo
Sources: Transparency International; UN Human Development Report

Figure 1:

```
## 6     6     Armenia       86 0.716   2.6 East EU Cemt Asia
```

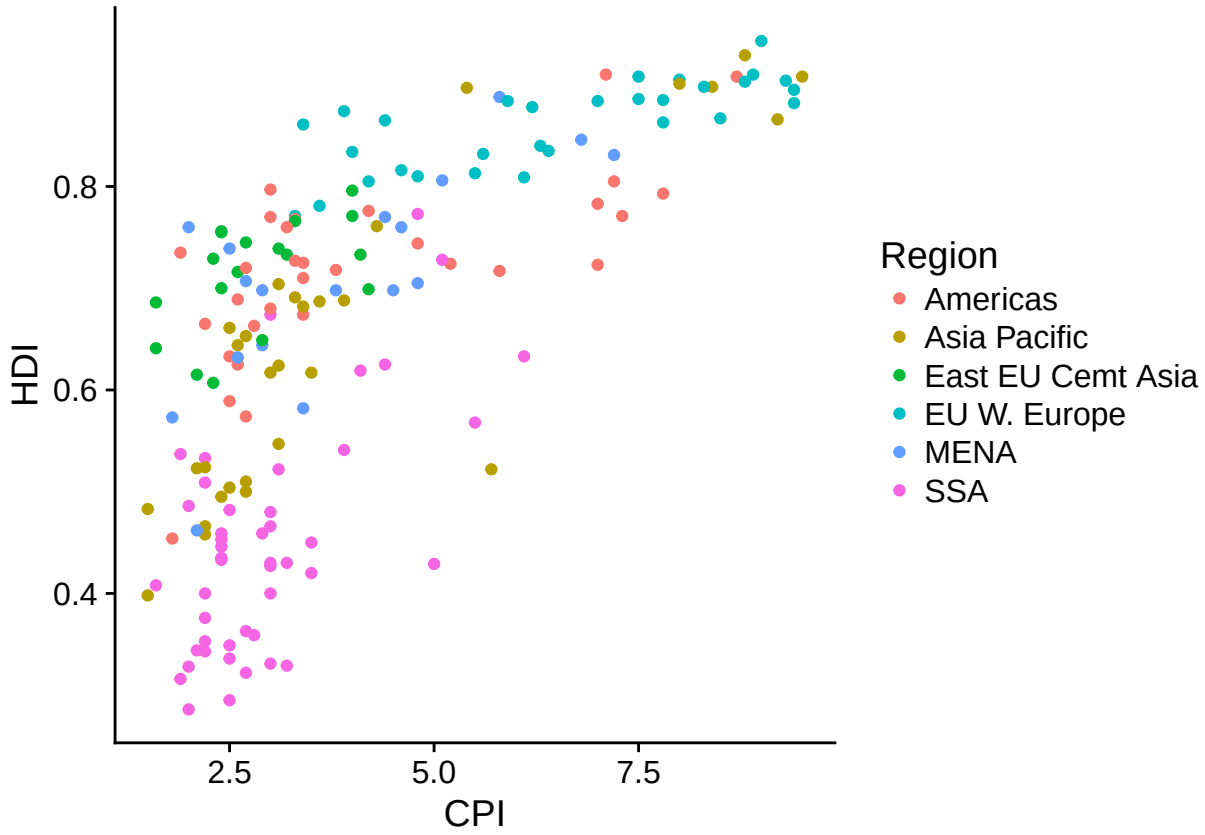
The original sources for these data are Transparency International and UN Human Development Reports.

Exercise

These data consist of Human Development Index and Corruption Perception Index scores for several countries.

1. Create a scatter plot with `CPI` on the x axis and `HDI` on the y axis.
2. Map the color of the the points to `Region`.
3. Map the size of the points to `HDI.Rank`

```
# Create scatter plot with CPI on x axis and HDI on the y axis.
ggplot(data=econ_data, aes(x=CPI, y=HDI)) +
  geom_point(aes(color = Region))
```

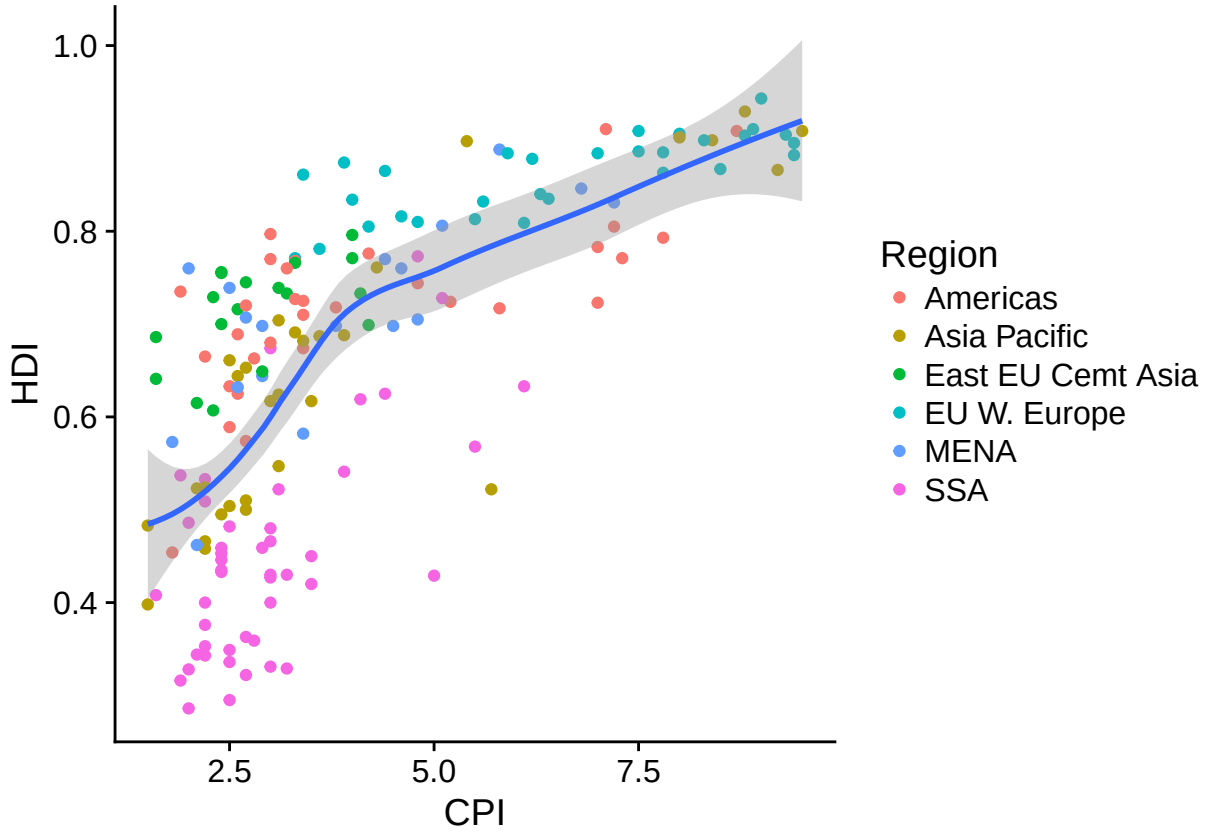


Exercise

Re-create the graph in the first exercise, but include a smoothing line (`geom_smooth`)

```
# Create scatter plot with CPI on x axis and HDI on the y axis.
ggplot(data=econ_data, aes(x=CPI, y=HDI)) +
  geom_point(aes(color = Region)) +
  geom_smooth()
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

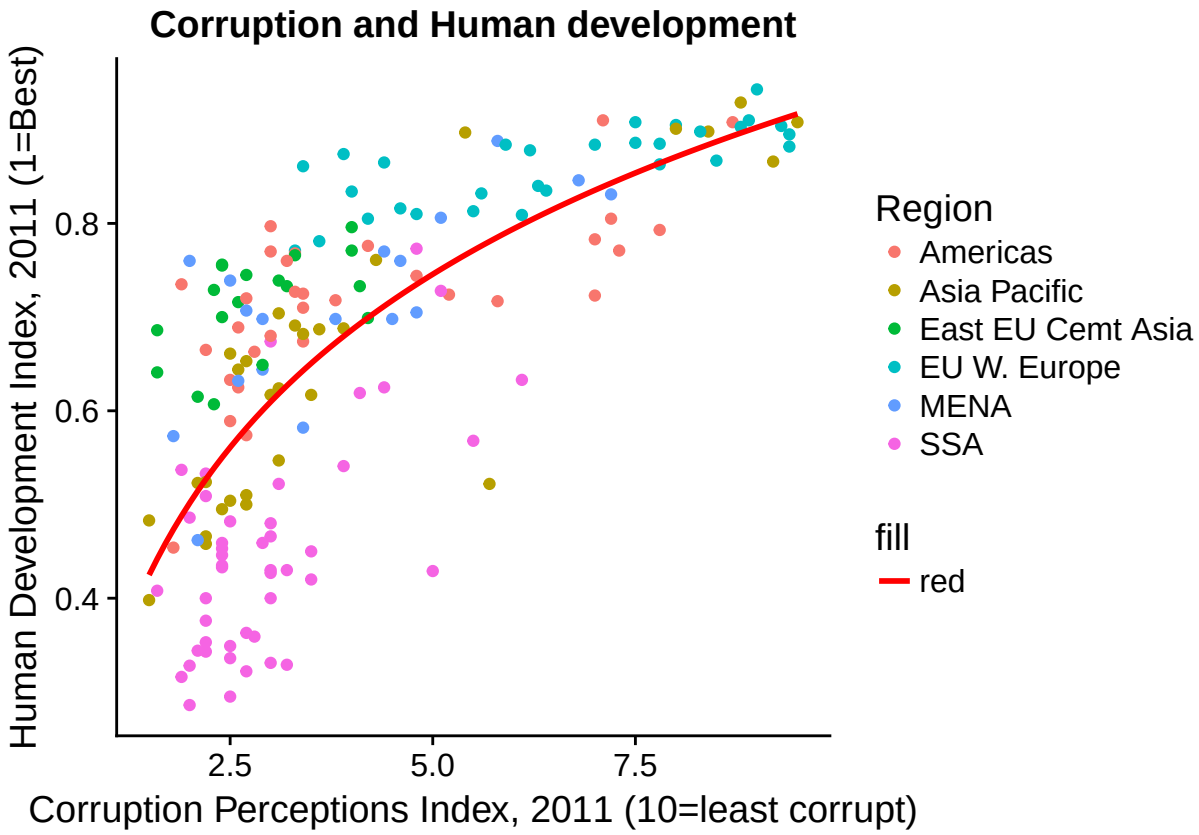


Exercise

Using the graph from the previous exercises, let's make some changes to aesthetics.

1. Rename the axis titles to their full name instead of abbreviations
2. Add a title to the plot
3. Tweak `geom_smooth()` to reflect the source graph.

```
ggplot(data=econ_data, aes(x=CPI, y=HDI)) +
  geom_point(aes(color = Region)) +
    geom_smooth(aes(fill="red"),method = "lm",formula = y~log(x), se=F, color="red") +
ggtitle('Corruption and Human development') +
  scale_x_continuous(name = "Corruption Perceptions Index, 2011 (10=least corrupt)") +
  scale_y_continuous(name = "Human Development Index, 2011 (1=Best)")
```



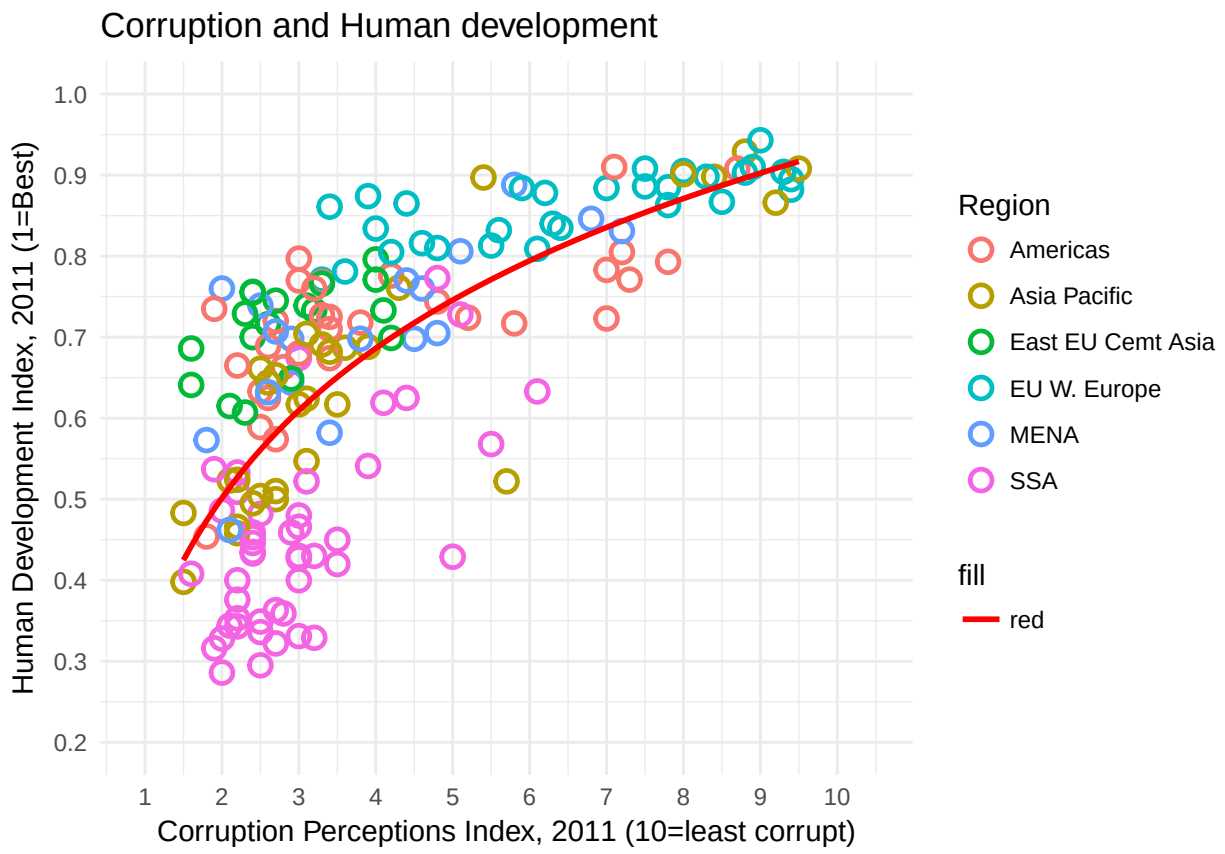
Exercise

Using the graph from the previous exercises, let's make more changes.

1. Change the geom_point() icons.
2. Make axis label limits similar to the original graph.

3. Change the theme to `theme_minimal()` from the `ggthemes` package.

```
ggplot(data=econ_data, aes(x=CPI, y=HDI)) +
  geom_point(aes(color = Region),
             shape = 1,size=3, fill=NA, stroke=1.25) + # change the symbols to reflect original graph
    geom_smooth(aes(fill="red"),method = "lm",formula = y~log(x), se=F, color="red") +
ggtitle('Corruption and Human development') +
  scale_x_continuous(name = "Corruption Perceptions Index, 2011 (10=least corrupt)",
                     limits = c(.9, 10.5),
                     breaks = 1:10) +
  scale_y_continuous(name = "Human Development Index, 2011 (1=Best)",
                     limits = c(.2, 1.0),
                     breaks = seq(.2, 1, by=0.1)) +
  theme_minimal() # change theme
```



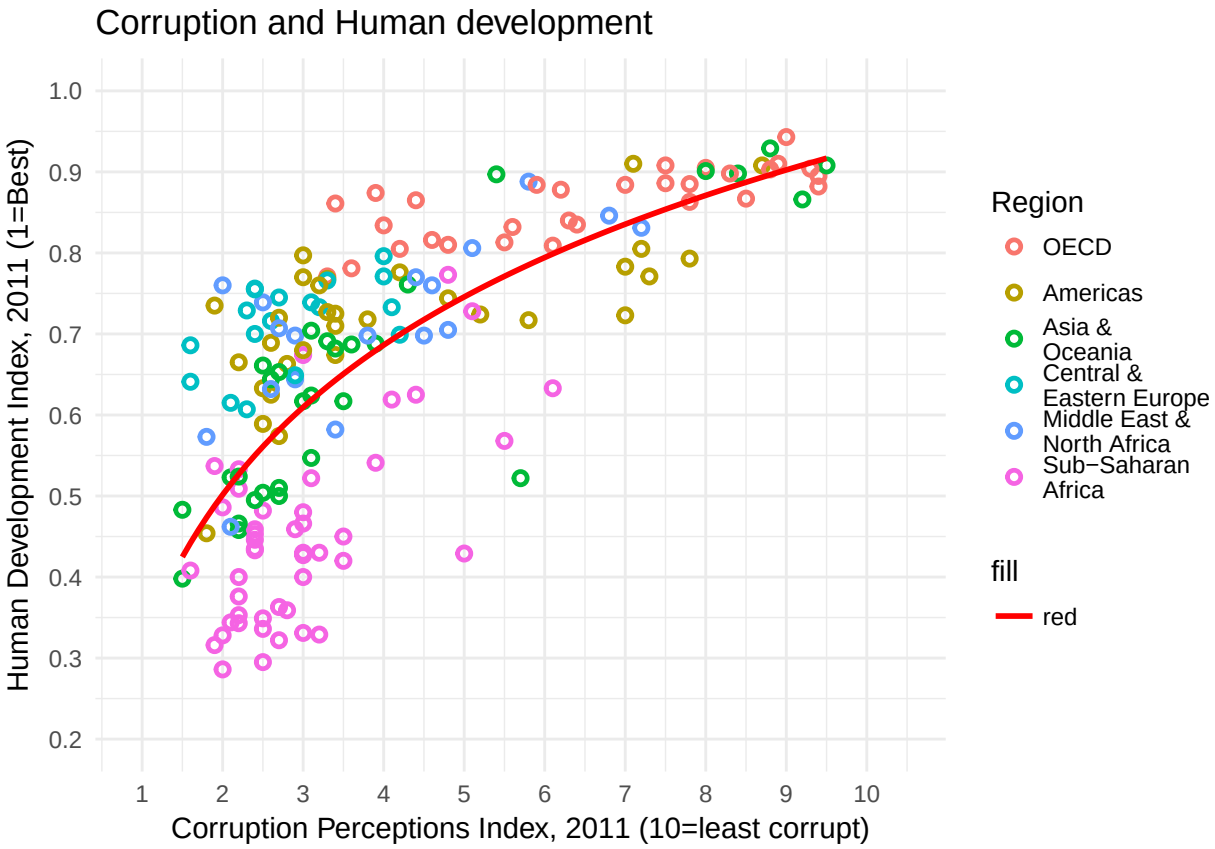
Exercise

1. Change the legend names for the regions.

```
# change the labels for the regions
# note that the order here is important, since factors are ordered

econ_data$Region <- factor(econ_data$Region, # change the variable class from character to factor
                           levels = c("EU W. Europe", # the original "levels" or unique values
                                      "Americas",
                                      "Asia Pacific",
                                      "East EU Cemt Asia",
                                      "MENA",
                                      "SSA"),
                           labels = c("OECD", # the new labels
                                      "Americas",
                                      "Asia &\nOceania", # /n is the "newline" syntax
                                      "Central &\nEastern Europe",
                                      "Middle East &\nNorth Africa",
                                      "Sub-Saharan\nAfrica"))
```

```
ggplot(data=econ_data, aes(x=CPI, y=HDI)) +
  geom_point(aes(color = Region),
             shape = 1, fill=NA, stroke=1.25) + # change the symbols to reflect original graph
  geom_smooth(aes(fill="red"),method = "lm",formula = y~log(x), se=F, color="red") +
ggtitle('Corruption and Human development') +
```

```
  scale_x_continuous(name = "Corruption Perceptions Index, 2011 (10=least corrupt)",
                     limits = c(.9, 10.5),
                     breaks = 1:10) +
  scale_y_continuous(name = "Human Development Index, 2011 (1=Best)",
                     limits = c(.2, 1.0),
                     breaks = seq(.2, 1, by=0.1)) +
  theme_minimal() # change theme
```



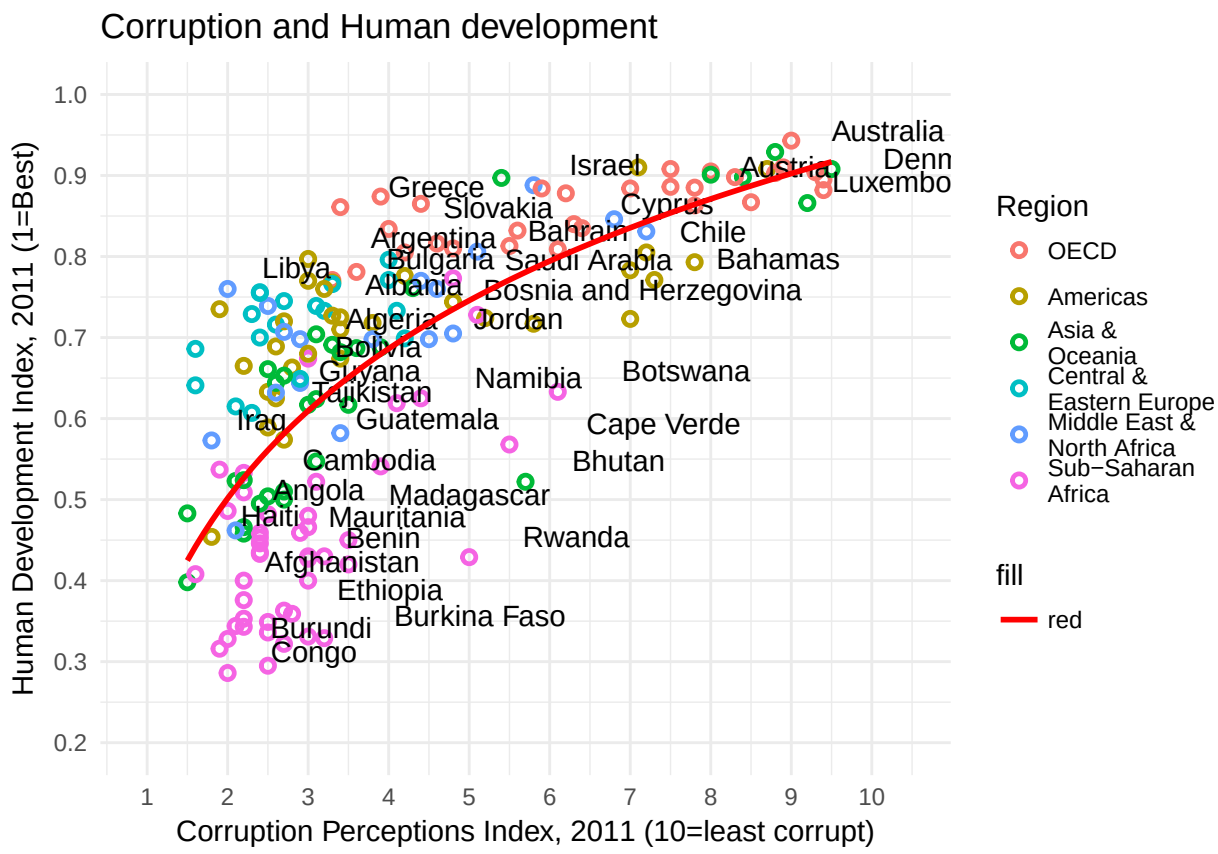
Exercise

1. Label the points as in the original image.

Let's try our luck at labeling the points. The default `ggplot2` commands are `geom_text()` for simple text and `geom_label()` if you want a pretty little box around the labels. `vjust` and `hjust` are vertical and horizontal adjustments that can take values of -1 to 1. They are sometimes annoying to work with, but have persisted in `ggplot` for a long time out of necessity.

```
ggplot(data=econ_data, aes(x=CPI, y=HDI)) +
  geom_point(aes(color = Region),
             shape = 1, fill=NA, stroke=1.25) + # change the symbols to reflect original graph
  geom_text(aes(label = Country), check_overlap = T, hjust=-.5, vjust=-.5) +
  geom_smooth(aes(fill="red"),method = "lm",formula = y~log(x), se=F, color="red") +
ggtitle('Corruption and Human development') +
  scale_x_continuous(name = "Corruption Perceptions Index, 2011 (10=least corrupt)",
                     limits = c(.9, 10.5),
                     breaks = 1:10) +
  scale_y_continuous(name = "Human Development Index, 2011 (1=Best)",
                     limits = c(.2, 1.0),
```

```
                    breaks = seq(.2, 1, by=0.1)) +
  theme_minimal() # change theme
```



Wow, geom_text really made it messy. Maybe we can clean that up?

Recall the original graph. It seems the author(s) were labeling only select countries. Likely, they used vector graphics software like Adobe Illustrator to place select labels after the fact. This is common in publication images. But maybe we can use R to do something similar?

Unfortunately, this seems to be a manual task, which R users typically abhor.

```
pointsToLabel <- c("Russia", "Venezuela", "Iraq", "Myanmar", "Sudan",
                   "Afghanistan", "Congo", "Greece", "Argentina", "Brazil",
                   "India", "Italy", "China", "South Africa", "Spane",
                   "Botswana", "Cape Verde", "Bhutan", "Rwanda", "France",
                   "United States", "Germany", "Britain", "Barbados", "Norway", "Japan",
                   "New Zealand", "Singapore")
```

We could use `geom_text()` from `ggplot2` but as you can see the labels overlap, and if you read the function documentation you will see there is no way to add lines to some points.

```
ggplot(data=econ_data, aes(x=CPI, y=HDI)) +
  geom_point(aes(color = Region),
             shape = 1, fill=NA, stroke=1.25) + # change the symbols to reflect original graph
# our new code snippet
 geom_text(aes(label = Country),
            color = "gray20",
            data = subset(econ_data, Country %in% pointsToLabel)) +
```

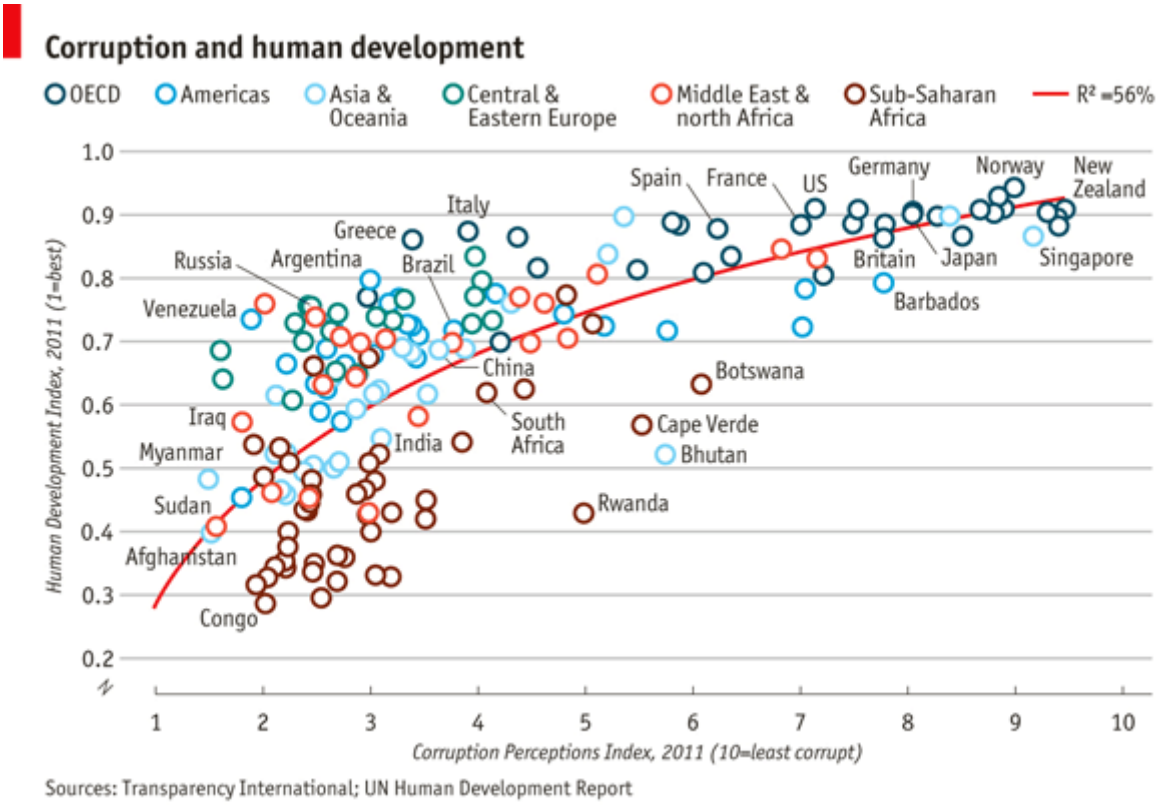
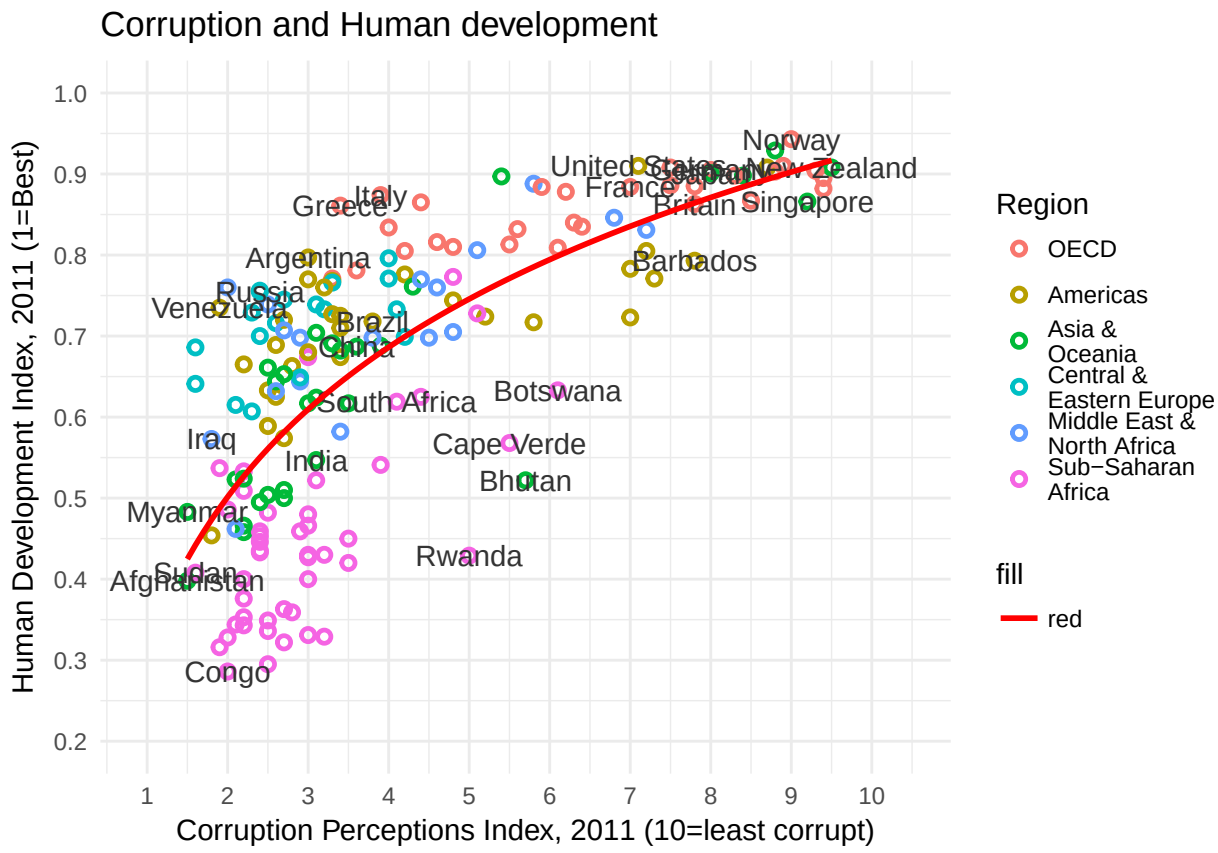

Corruption and human development
OECD
Americas
Asia & Oceania
Central & Eastern Europe
Middle East & north Africa
Sub-Saharan Africa
R² =56%
Human Development Index, 2011 (1=best)
Corruption Perceptions Index, 2011 (10=least corrupt)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
1
2
3
4
5
6
7
8
9
10
Spain
France
US
Germany
Norway
New Zealand
Italy
Greece
Russia
Argentina
Brazil
Britain
Japan
Singapore
Venezuela
Barbados
China
Botswana
Iraq
South Africa
Cape Verde
India
Bhutan
Myanmar
Sudan
Rwanda
Afghanistan
Congo
Sources: Transparency International; UN Human Development Report

Figure 2:

```
# back to the old graph code
  geom_smooth(aes(fill="red"),method = "lm",formula = y~log(x), se=F, color="red") +
ggtitle('Corruption and Human development') +
  scale_x_continuous(name = "Corruption Perceptions Index, 2011 (10=least corrupt)",
                     limits = c(.9, 10.5),
                     breaks = 1:10) +
  scale_y_continuous(name = "Human Development Index, 2011 (1=Best)",
                     limits = c(.2, 1.0),
                     breaks = seq(.2, 1, by=0.1)) +
  theme_minimal() # change theme
```



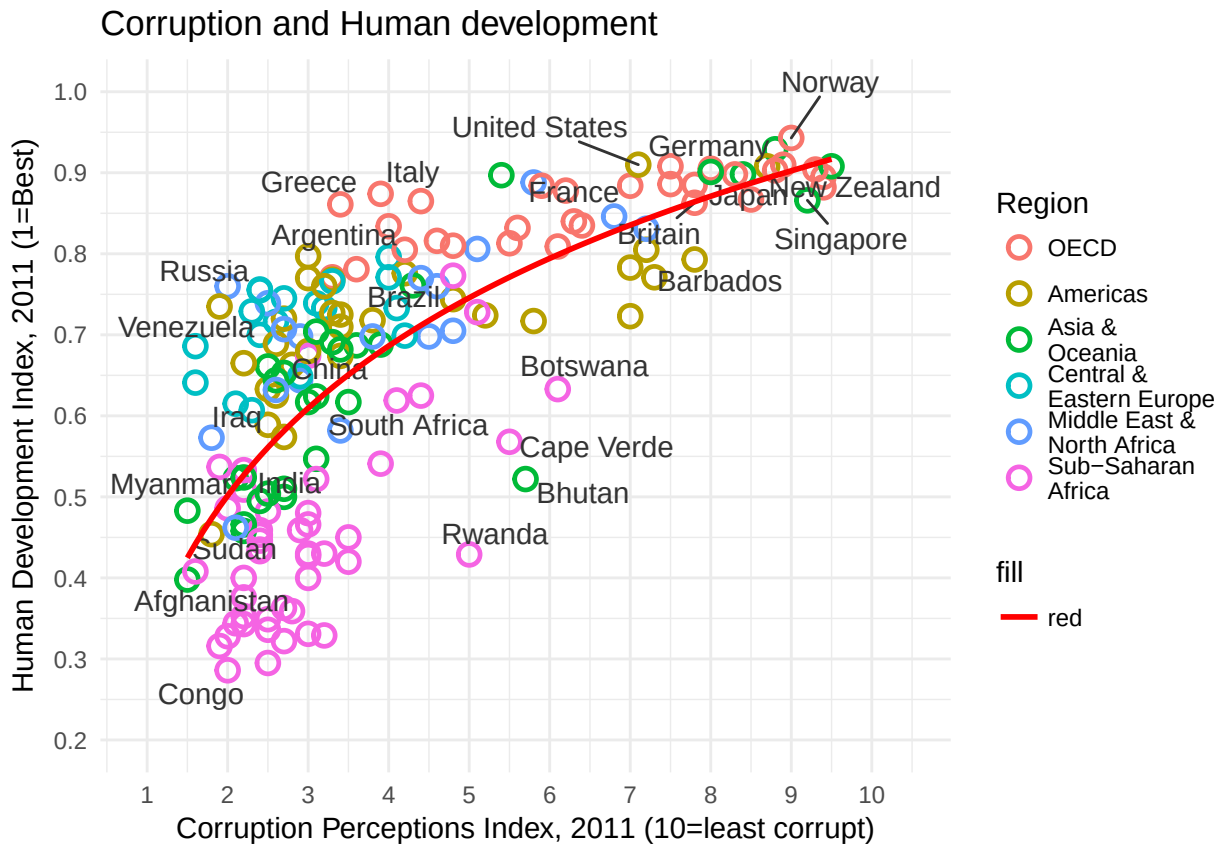
Better, but not perfect. But, since this is R, there probably is a package built to handle just this problem.

Hint: There is.

Enter `ggrepel` and a better `geom_text()` called `geom_text_repel()`

```
ggplot(data=econ_data, aes(x=CPI, y=HDI)) +
  geom_point(aes(color = Region),
             shape = 1, size=3, fill=NA, stroke=1.25) + # change the symbols to reflect original graph
# our new code snippet ----------------------------------------------------
geom_text_repel(aes(label = Country),
            color = "gray20",
            data = subset(econ_data, Country %in% pointsToLabel),
            force = 10) +
# back to the old graph code----------------------------------------------------
  geom_smooth(aes(fill="red"),method = "lm",formula = y~log(x), se=F, color="red") +
```

```
ggtitle('Corruption and Human development') +
  scale_x_continuous(name = "Corruption Perceptions Index, 2011 (10=least corrupt)",
                     limits = c(.9, 10.5),
                     breaks = 1:10) +
  scale_y_continuous(name = "Human Development Index, 2011 (1=Best)",
                     limits = c(.2, 1.0),
                     breaks = seq(.2, 1, by=0.1)) +
  theme_minimal() # change theme
```



Exercise

1. Add our R^2 value to the legend
2. Adjust color scale to match source image

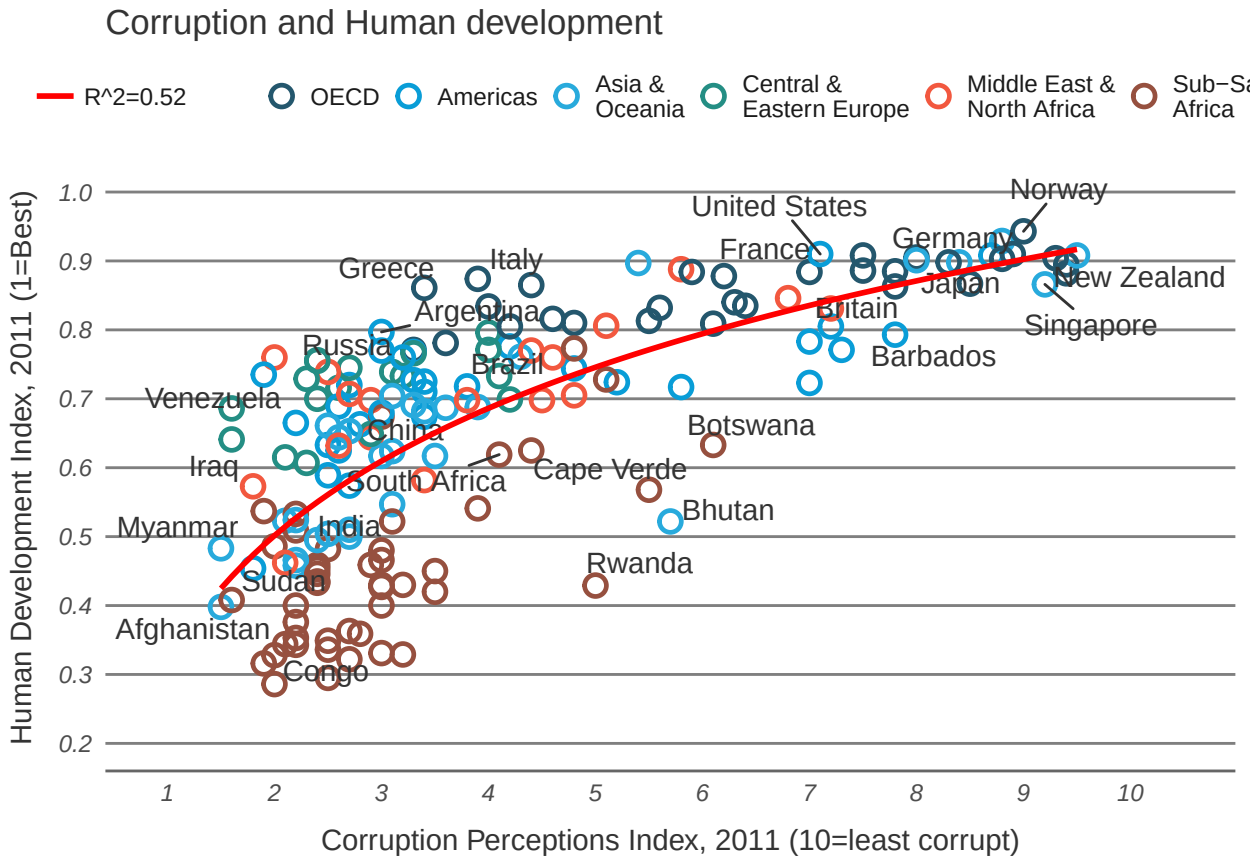
Now to just add our R^2 value to the legend, change the color scale, and add the source note. *Warning: R^2 legend is going to be a bit hacky. I wish there were a better solution but haven't found one yet.*

The colors in R can have names, or be identified by hexadecimal values. I did the dirty work for you here, but note that there is a great shiny app "Addin" for R called `colourpicker` that will help do this for you. Thanks Dean Attali.

```
# get our R^2 value
mR2 <- summary(lm(HDI ~ log(CPI), data=econ_data))$r.squared
mR2 <- round(mR2, 2)
```

```
# the plot ----------------------------------------------------------
ggplot(data=econ_data, aes(x=CPI, y=HDI)) +
  geom_point(aes(color = Region),
             shape = 1, size=3, fill=NA, stroke=1.25) + # change the symbols to reflect original graph
  # text labels for selected countries ------------------------------
geom_text_repel(aes(label = Country),
            color = "gray20",
            data = subset(econ_data, Country %in% pointsToLabel),
            force = 10) +
  # our regression line ---------------------------------------------
  geom_smooth(aes(fill="red"),method = "lm",formula = y~log(x), se=F, color="red") +
  # title and axis scales -------------------------------------------
ggtitle('Corruption and Human development') +
  scale_x_continuous(name = "Corruption Perceptions Index, 2011 (10=least corrupt)",
                     limits = c(.9, 10.5),
                     breaks = 1:10) +
  scale_y_continuous(name = "Human Development Index, 2011 (1=Best)",
                     limits = c(.2, 1.0),
                     breaks = seq(.2, 1, by=0.1)) +
  # our regression line labels --------------------------------------
  scale_fill_manual(name = "MyR^2",
                    values = c("red"),
                    labels = c(paste0("R^2=", mR2))) +



  # new color scale -------------------------------------------------
  scale_color_manual(name = "",
                     values = c("#24576D",
                                "#099DD7",
                                "#28AADC",
                                "#248E84",
                                "#F2583F",
                                "#96503F"))+
  # end new color scale ---------------------------------------------
  theme_minimal() + # change theme
  # theme tweaks ----------------------------------------------------
  theme(text = element_text(color = "gray20"),
        legend.position = "top", # position the legend in the upper left
        legend.direction = "horizontal",
        legend.justification = c(0.05,0), # anchor point for legend.position.
        legend.text = element_text(size = 8.5, color = "gray10"),
        legend.title = element_blank(),
        axis.text = element_text(face = "italic"),
        axis.title.x = element_text(vjust = -1), # move title away from axis
        axis.title.y = element_text(vjust = 2), # move away for axis
        axis.ticks.y = element_blank(), # element_blank() is how we remove elements
        axis.line = element_line(color = "gray40", size = 0.5),
        axis.line.y = element_blank(),
        panel.grid.major = element_line(color = "gray50", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank()
```

```
        ) + guides(colour = guide_legend(nrow = 1),
                   fill = guide_legend(nrow = 1))
```



Exercise

1. Add source note

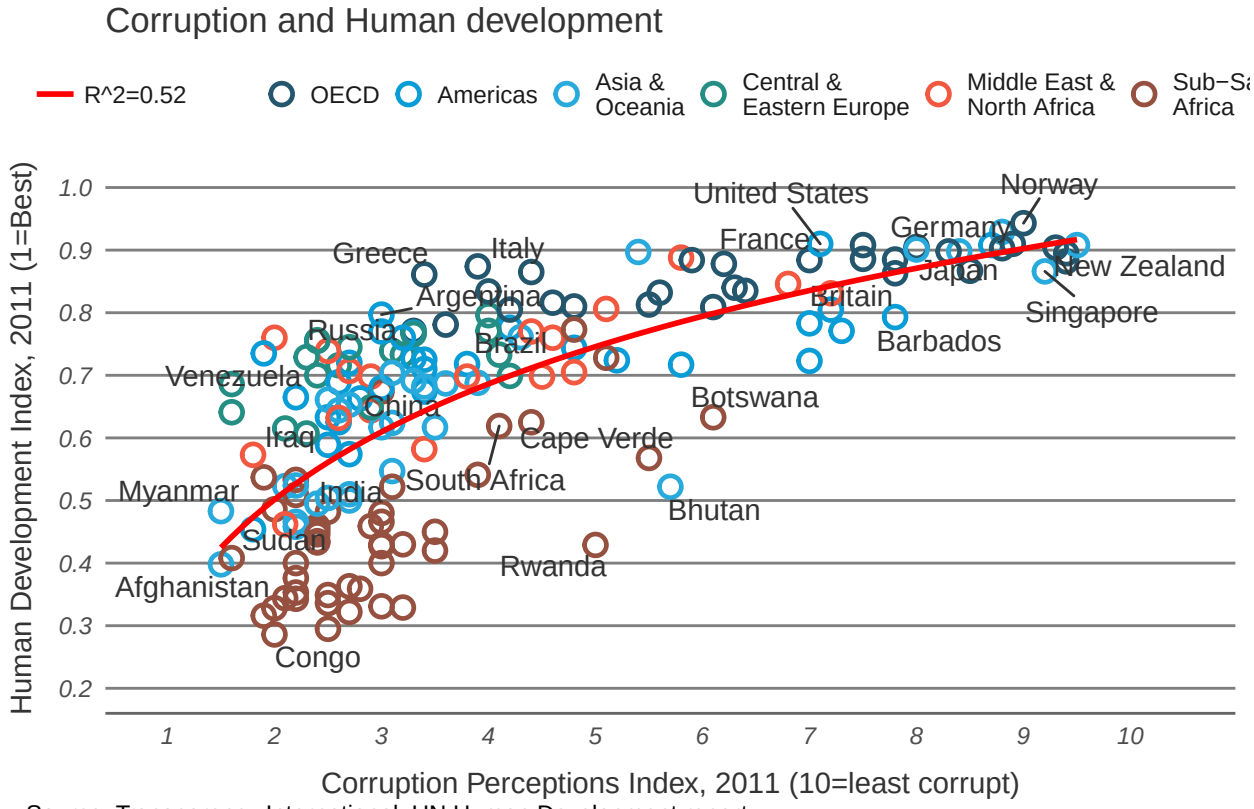
Now we just need to add the source note. Again, many times things like this are done in post-processing using image software like Adobe Illustrator. But I'm bull-headed and like using R as much as possible, so we're going to barrel on through.

Another limitation of `ggplot2` is that there are not easy methods for adding annotations *outside* of the plot area. After some searching the package `cowplot` came to my attention that will allow for annotations outside the plot area.

```
# load cowplot
library(cowplot)
# save our plot to a variable (note: we could do this for each step, but for class purposes I didn't)
p <- ggplot(data=econ_data, aes(x=CPI, y=HDI)) +
  geom_point(aes(color = Region),
             shape = 1, size=3, fill=NA, stroke=1.25) + # change the symbols to reflect original graph
  # text labels for selected countries ------------------------------
geom_text_repel(aes(label = Country),
            color = "gray20",
            data = subset(econ_data, Country %in% pointsToLabel),
            force = 10) +
```

```
  # our regression line -------------------------------------------
  geom_smooth(aes(fill="red"),method = "lm",formula = y~log(x), se=F, color="red") +
  # title and axis scales -----------------------------------------
ggtitle('Corruption and Human development') +
  scale_x_continuous(name = "Corruption Perceptions Index, 2011 (10=least corrupt)",
                     limits = c(.9, 10.5),
                     breaks = 1:10) +
  scale_y_continuous(name = "Human Development Index, 2011 (1=Best)",
                     limits = c(.2, 1.0),
                     breaks = seq(.2, 1, by=0.1)) +
  # our regression line labels ------------------------------------
  scale_fill_manual(name = "MyR^2",
                    values = c("red"),
                    labels = c(paste0("R^2=", mR2))) +




  # new color scale -----------------------------------------------
  scale_color_manual(name = "",
                     values = c("#24576D",
                                "#099DD7",
                                "#28AADC",
                                "#248E84",
                                "#F2583F",
                                "#96503F"))+
  # end new color scale -------------------------------------------
  theme_minimal() + # change theme
  # theme tweaks --------------------------------------------------
  theme(text = element_text(color = "gray20"),
        legend.position = "top", # position the legend in the upper left
        legend.direction = "horizontal",
        legend.justification = c(0.05,0), # anchor point for legend.position.
        legend.text = element_text(size = 8.5, color = "gray10"),
        legend.title = element_blank(),
        axis.text = element_text(face = "italic"),
        axis.title.x = element_text(vjust = -1), # move title away from axis
        axis.title.y = element_text(vjust = 2), # move away for axis
        axis.ticks.y = element_blank(), # element_blank() is how we remove elements
        axis.line = element_line(color = "gray40", size = 0.5),
        axis.line.y = element_blank(),
        panel.grid.major = element_line(color = "gray50", size = 0.5),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank()
        ) + guides(colour = guide_legend(nrow = 1),
                   fill = guide_legend(nrow = 1))

# add our source note with cowplot's add_sub() function
p2 <- # we're saving our results to a new plot to not overwrite our last one
  add_sub(p, # p = our saved plot from above
          "Source: Transparency International; UN Human Development report",
          x=-0.07,
          hjust = 0,
```

```
          fontface = "plain",
          size = 9)

# cowplot requires the function ggdraw to draw the object now
# this is because cowplot takes our plot and turns it into a table with a plot in the middle
# and extra annotations, etc. where we want them
ggdraw(p2)
```



Source: Transparency International; UN Human Development report

Wrap-up

Compared to the original, we're pretty darn close. Congrats. This wasn't as easy of a tutorial as it first seemed, but it does show you the strengths and limitations of `ggplot2` and ultimately will be a good resource if you start using `ggplot2` for publication-quality images.

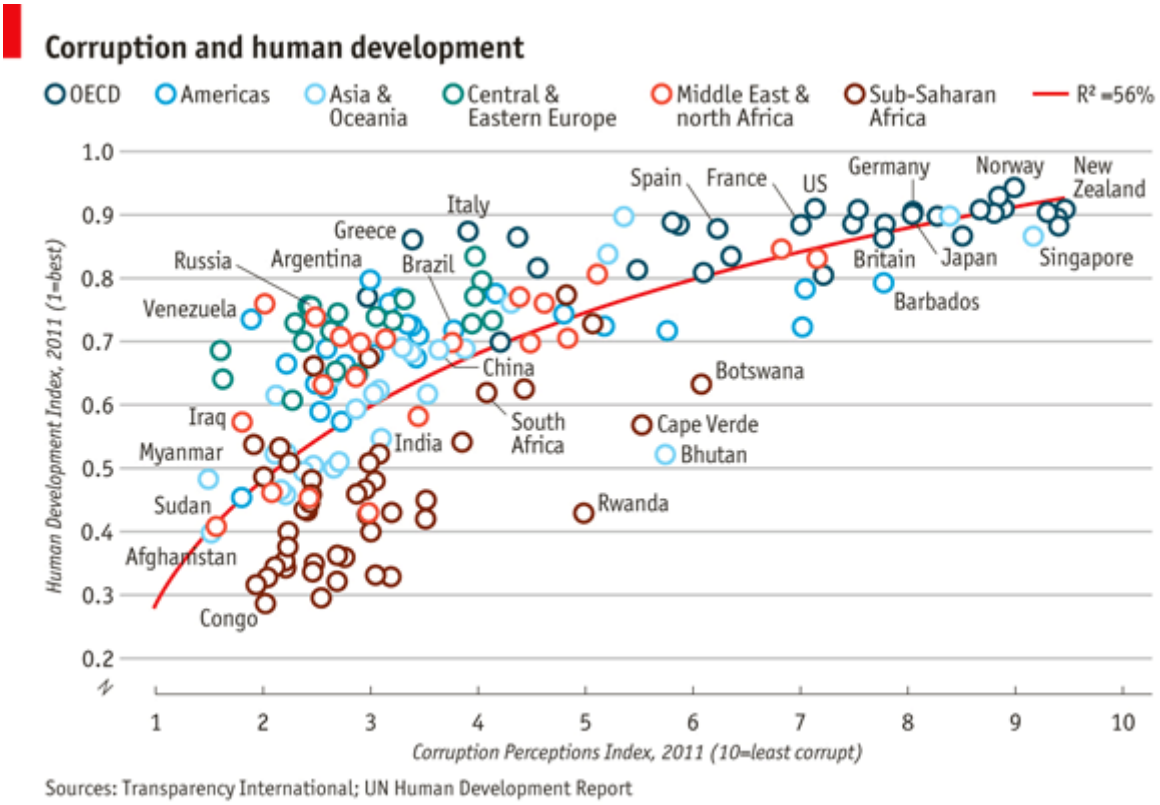

Corruption and human development
OECD
Americas
Asia & Oceania
Central & Eastern Europe
Middle East & north Africa
Sub-Saharan Africa
R² =56%
Human Development Index, 2011 (1=best)
Corruption Perceptions Index, 2011 (10=least corrupt)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
1
2
3
4
5
6
7
8
9
10
Spain
France
US
Germany
Norway
New Zealand
Italy
Greece
Russia
Argentina
Brazil
Britain
Japan
Singapore
Venezuela
Barbados
China
Botswana
Iraq
South Africa
Cape Verde
India
Bhutan
Myanmar
Sudan
Rwanda
Afghanistan
Congo
Sources: Transparency International; UN Human Development Report

Figure 3: