Plotting with `ggplot2` - Exercise

Lachlan Deer Ulrich Bergmann

The goal of this set of exercises is to get more familarity with some of the common `ggplot` functions that we as economists use frequently. By the end of the exercise we should have a relatively nice looking figure that we would be happy to show someone.

The data we will use comes from the UN Human Development Report from 2011. We are interested in understanding the relationship between the Human Development Index (HDI) and the Corruption Perceptions Index (CPI).

First let us load the libraries we will need:

```r
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(ggplot2)
library(magrittr)
```

Let's get started!

Load Data and Clean variable names

1. Load the data from the file `data/EconomistData.csv`.

```r
df <- read_csv("data/EconomistData.csv")
```

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

```r
glimpse(df)
```

```
## Observations: 173
## Variables: 5
## $ Country  <chr> "Afghanistan", "Albania", "Algeria", "Angola", "Argentina"...
## $ HDI.Rank <dbl> 172, 70, 96, 148, 45, 86, 2, 19, 91, 53, 42, 146, 47, 65, ...
## $ HDI      <dbl> 0.398, 0.739, 0.698, 0.486, 0.797, 0.716, 0.929, 0.885, 0....
## $ CPI      <dbl> 1.5, 3.1, 2.9, 2.0, 3.0, 2.6, 8.8, 7.8, 2.4, 7.3, 5.1, 2.7...
## $ Region   <chr> "Asia Pacific", "East EU Cemt Asia", "MENA", "SSA", "Ameri...
```

2. Convert all columns names to snakecase (i.e. `my_variable`)

```
library(janitor)
```

```
##
## Attaching package: 'janitor'

## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

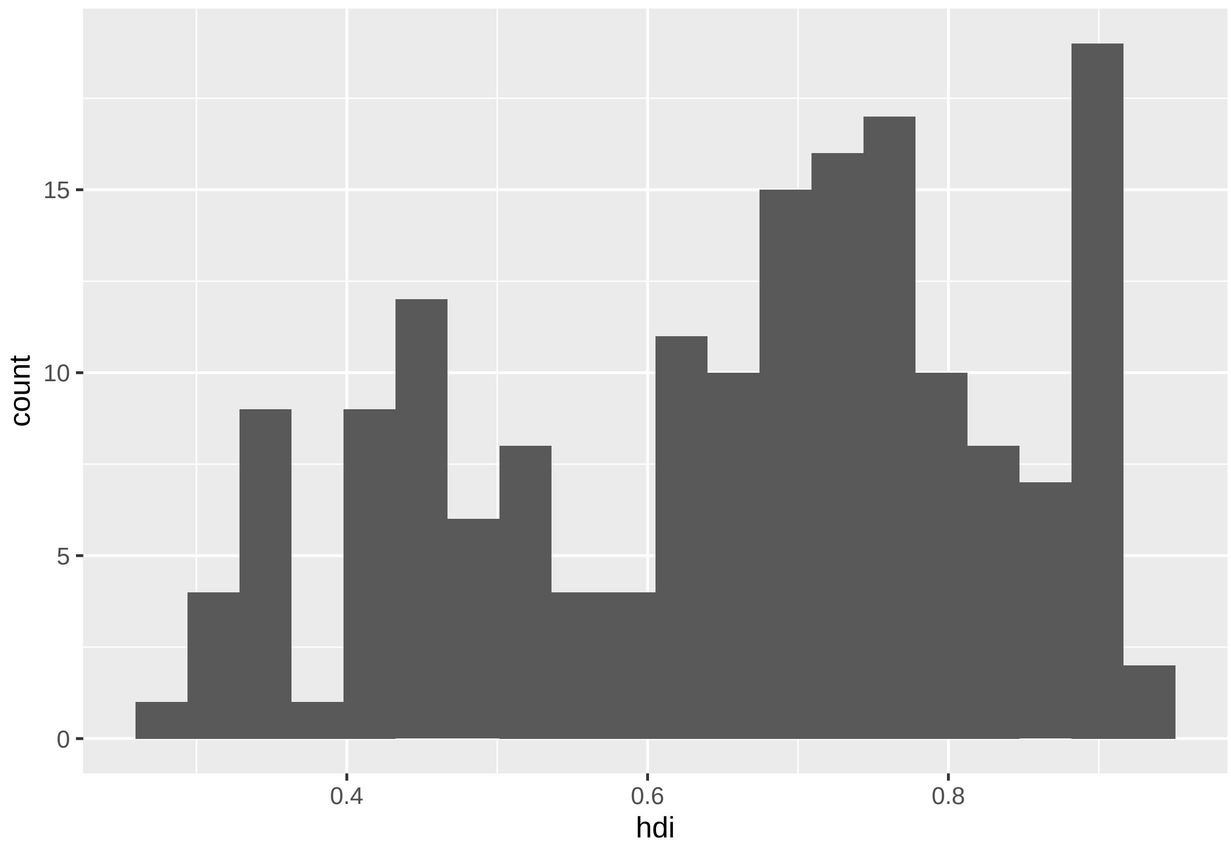
```
df %<>% clean_names("snake")
```

One Variable Graphs

First we work with some single variable plots.

1. Create a histogram of the human development index. Customize the number of bins to make the plot look nicer than the default.

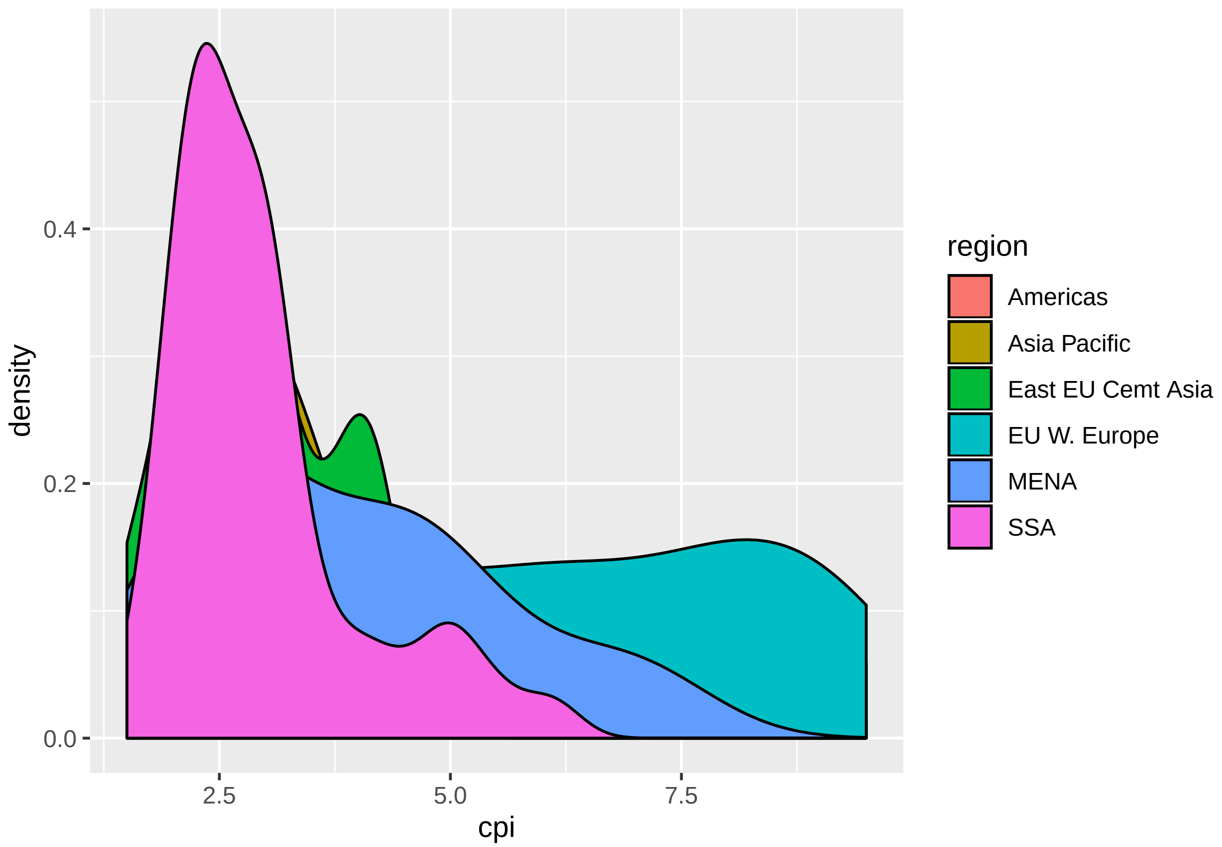
```
df %>%
  ggplot() +
  geom_histogram(aes(hdi),
                 bins = 20
                 )
```



2. Instead of a histogram, create a density plot of the HDI. Extend your plot by:

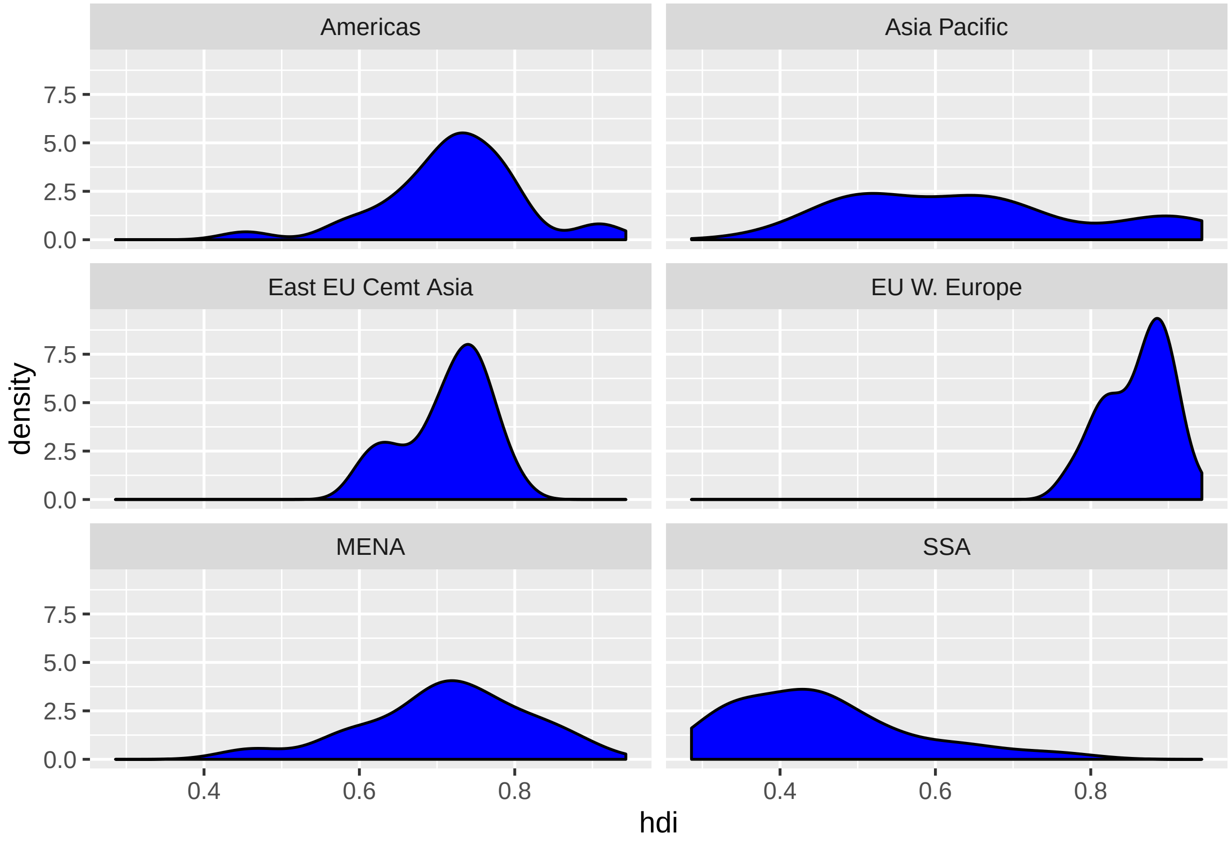
(a) In one graph plotting the densities by region.

```
df %>%
  ggplot(aes(cpi, fill = region)) +
  geom_density()
```



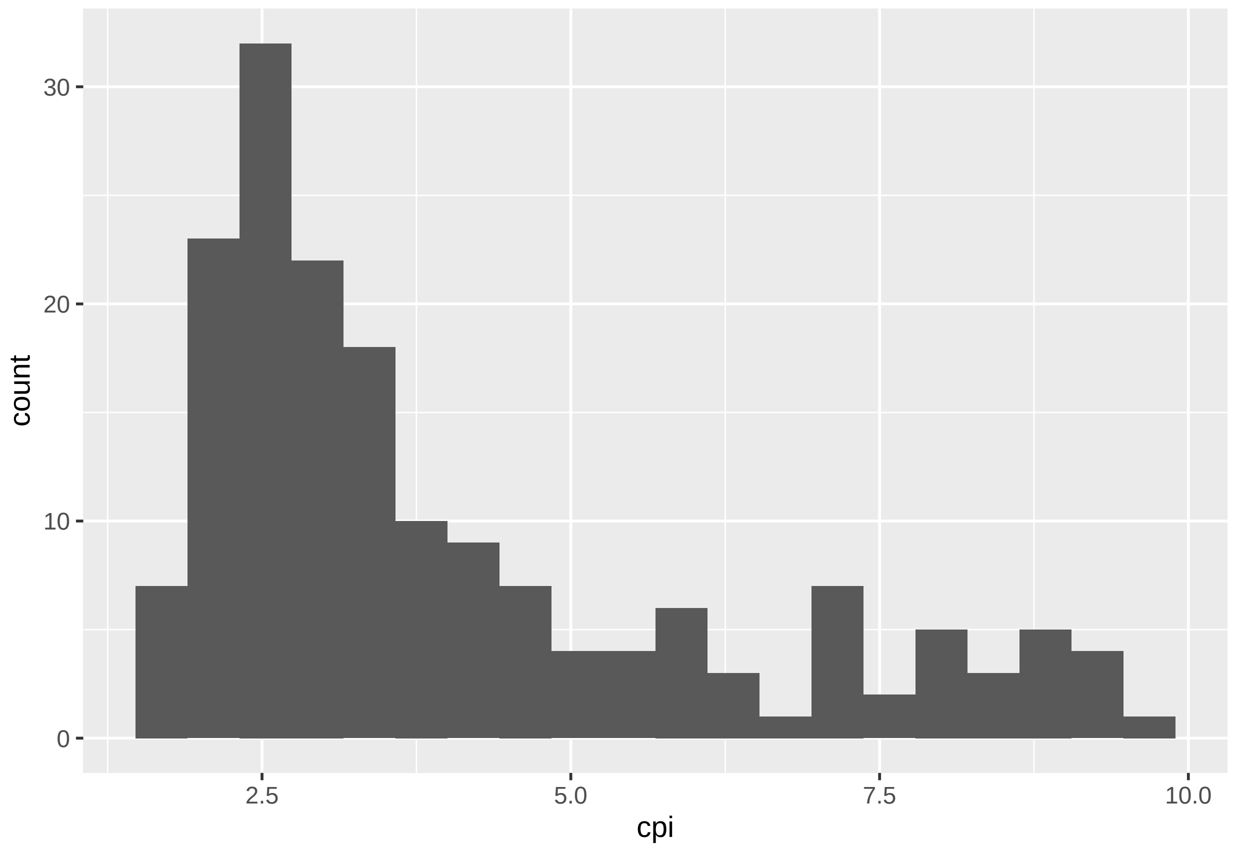
(b) Creating separate plots per region, with the area under the density to be coloured blue.

```
df %>%
  ggplot(aes(hdi)) +
  geom_density(fill = "blue") +
  facet_wrap(vars(region), ncol = 2)
```

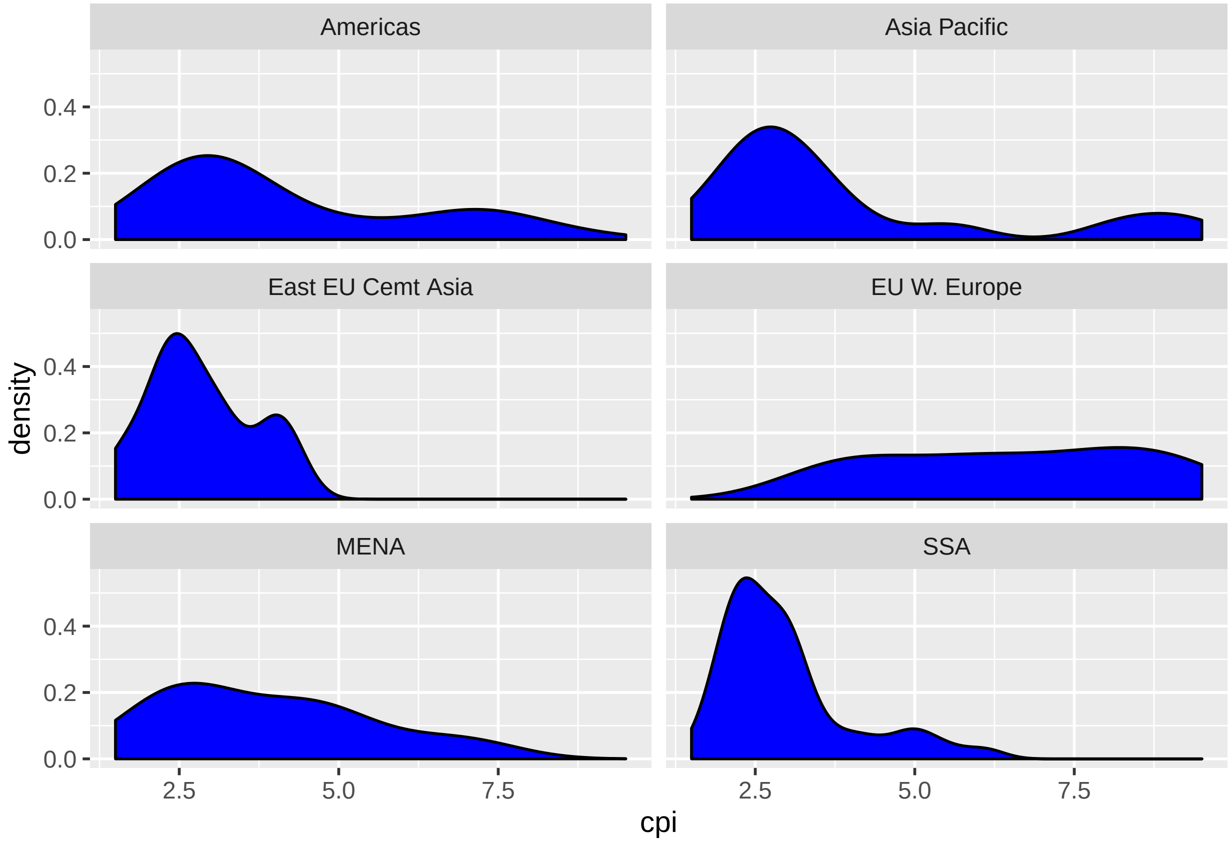


(c) Repeat (1) and (2) for the corruption perception index.

```
df %>%
  ggplot() +
  geom_histogram(aes(cpi),
                 bins = 20
  )
```



```
df %>%
  ggplot(aes(cpi)) +
  geom_density(fill = "blue") +
  facet_wrap(vars(region), ncol = 2)
```

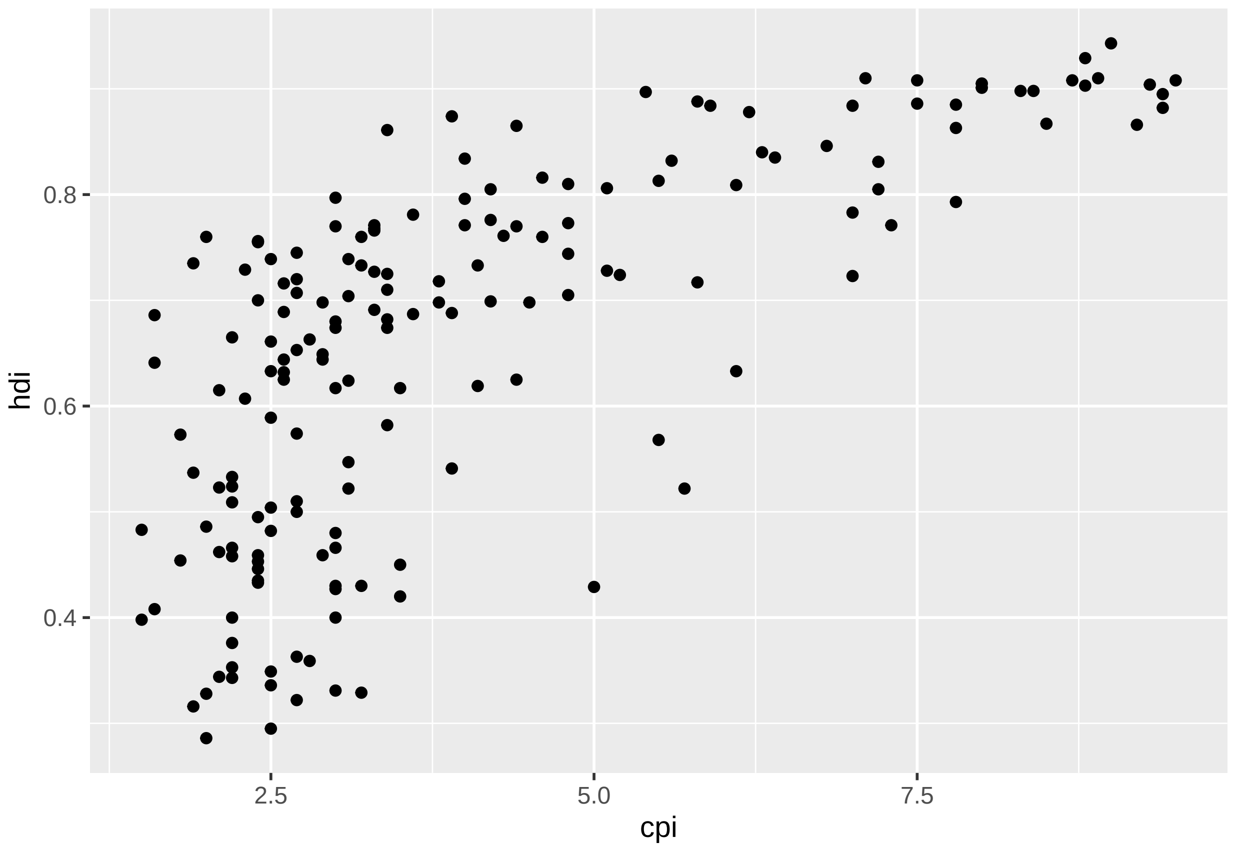


Two Variable Graphs

Now we are going to build up a ‘pretty’ graph that plots the corruption index (along the x-axis) against the human development index (along the y-axis).

1. Create the simple scatter plot

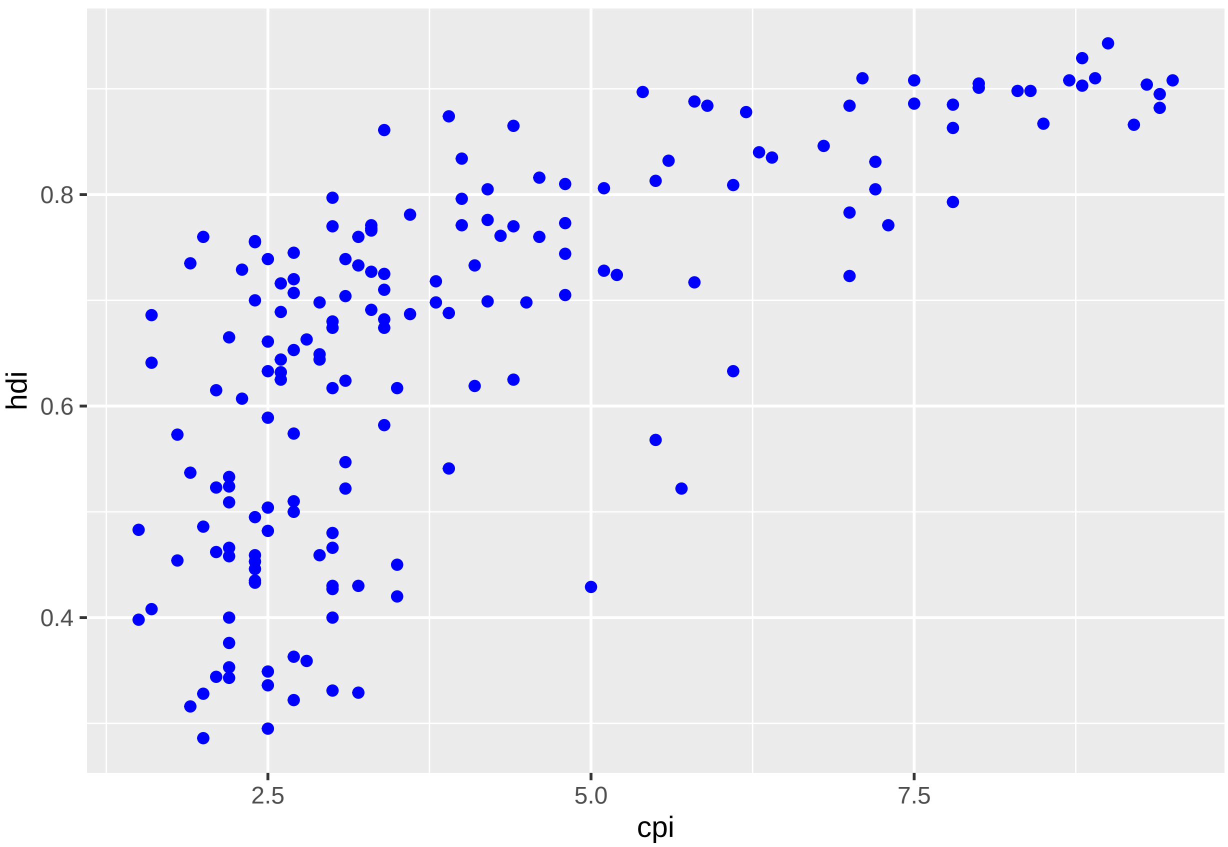
```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point()
```



2. Let's extend the plot in different ways. Modify the plot to (each point should be a different plot)

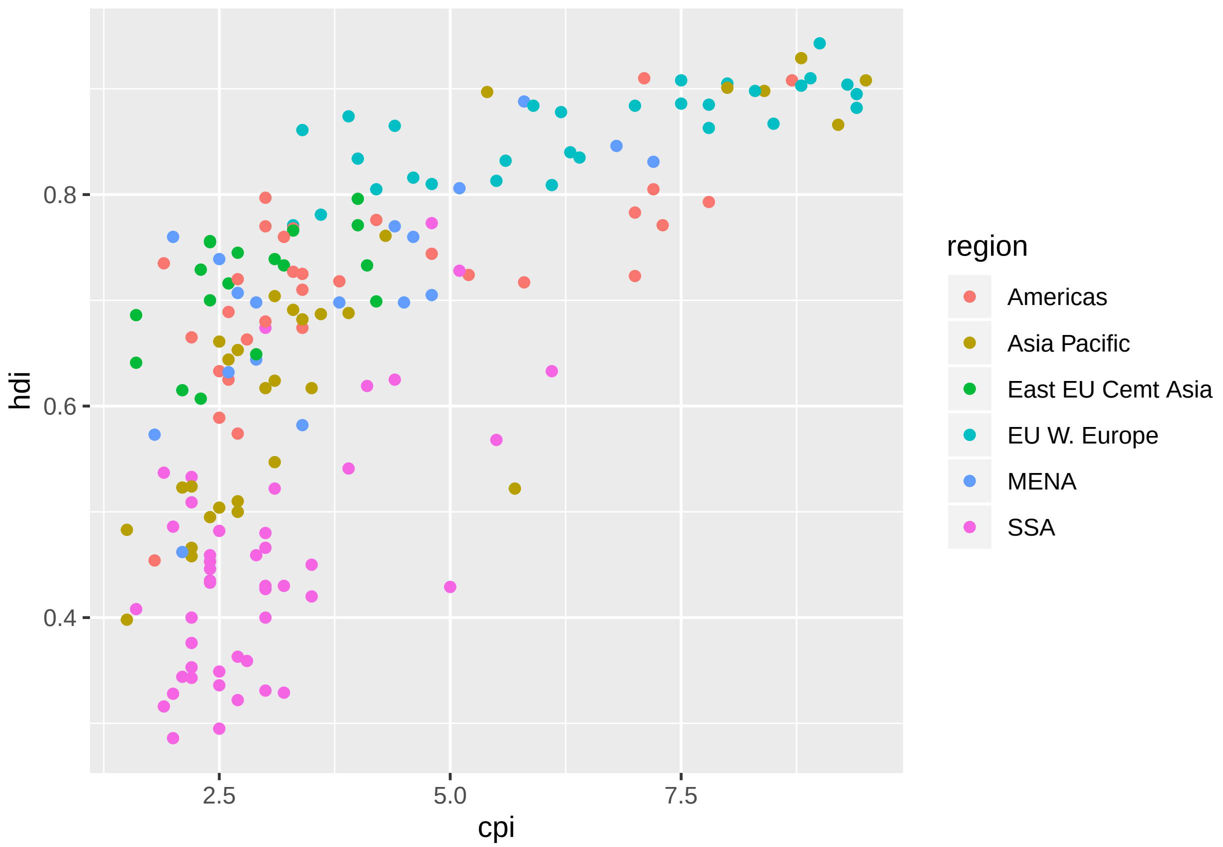
a. Make the points blue

```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(color = "blue")
```



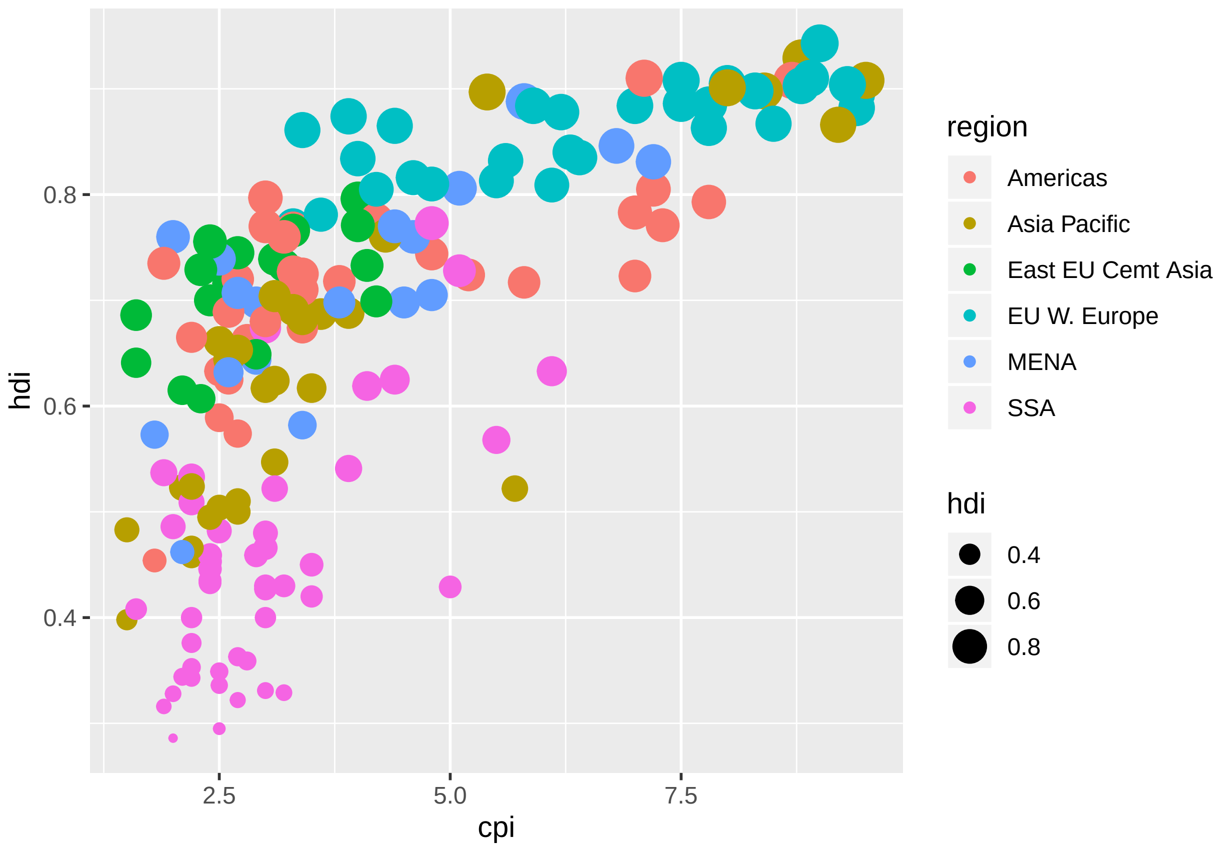
b. Color the points by region

```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region))
```



c. Color the points by region and make the size of the point vary by HDI.

```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region, size = hdi ))
```

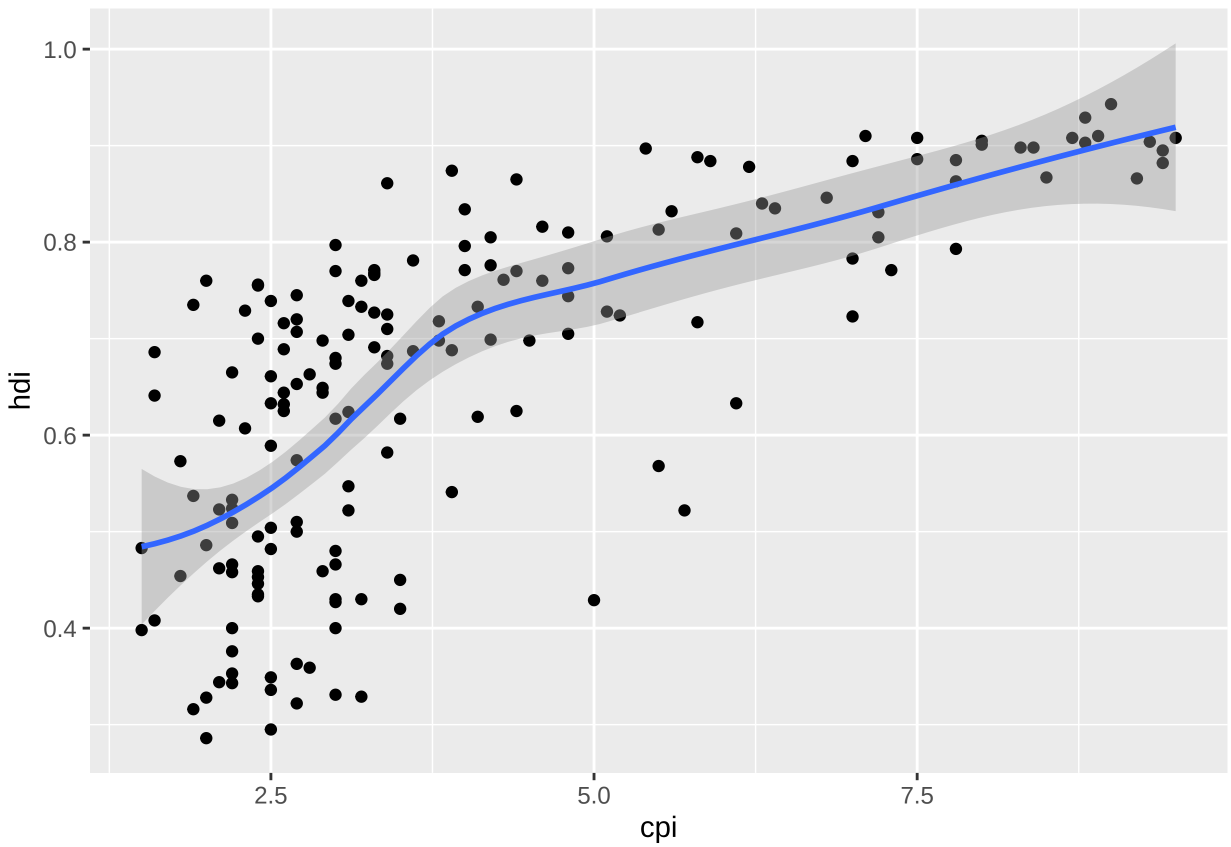


3. Let's extend the plot in (1) by adding some summary functions to it.

a. Add a loess smoother

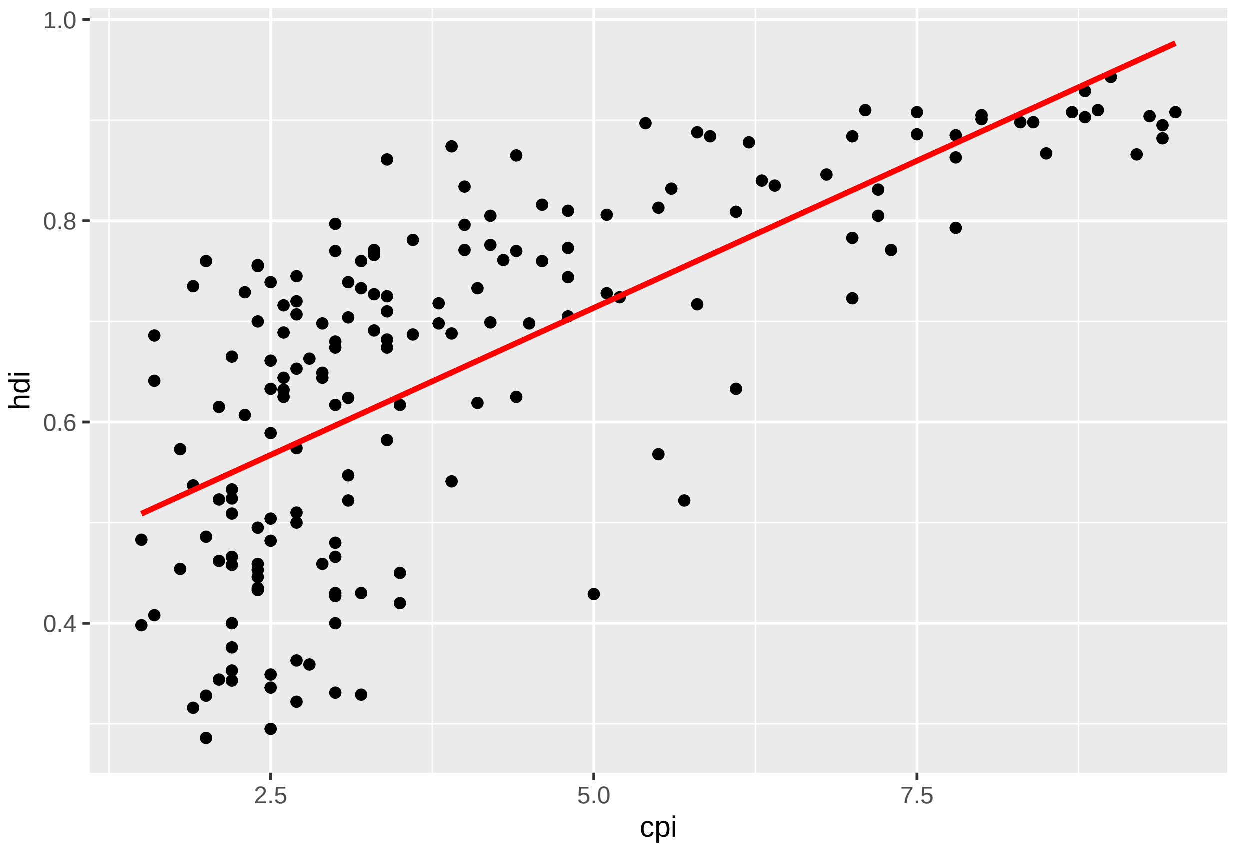
```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point() +
  geom_smooth()
```

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



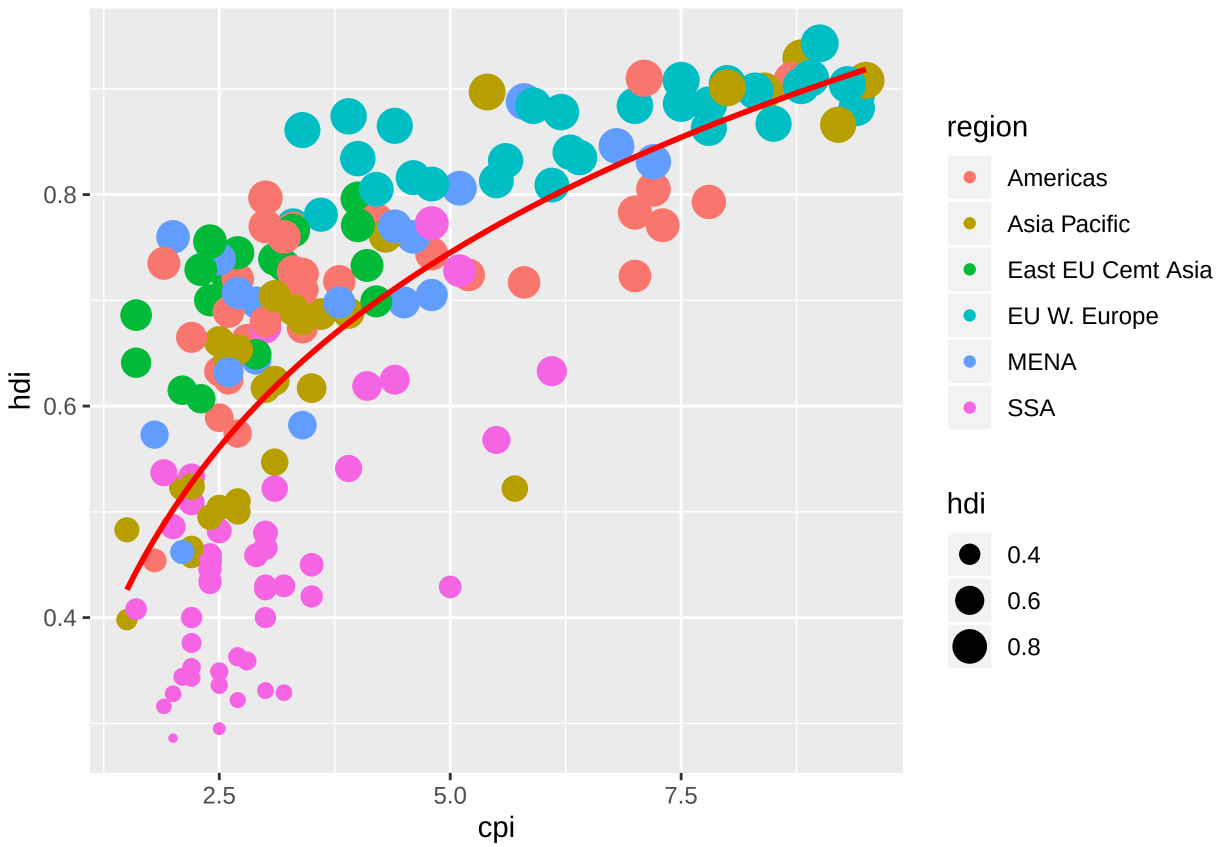
b. Add a linear smoother, without the confidence interval. Color the line red.

```r
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point() +
  geom_smooth(se= FALSE, method = "lm", color = "red")
```



c. Add the line `y ~ x + log(x)`, without the confidence interval. Color the line red.

```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region, size = hdi )) +
  geom_smooth(se= FALSE,
              method = "lm",
              formula = y ~ x + log(x),
              color = "red")
```



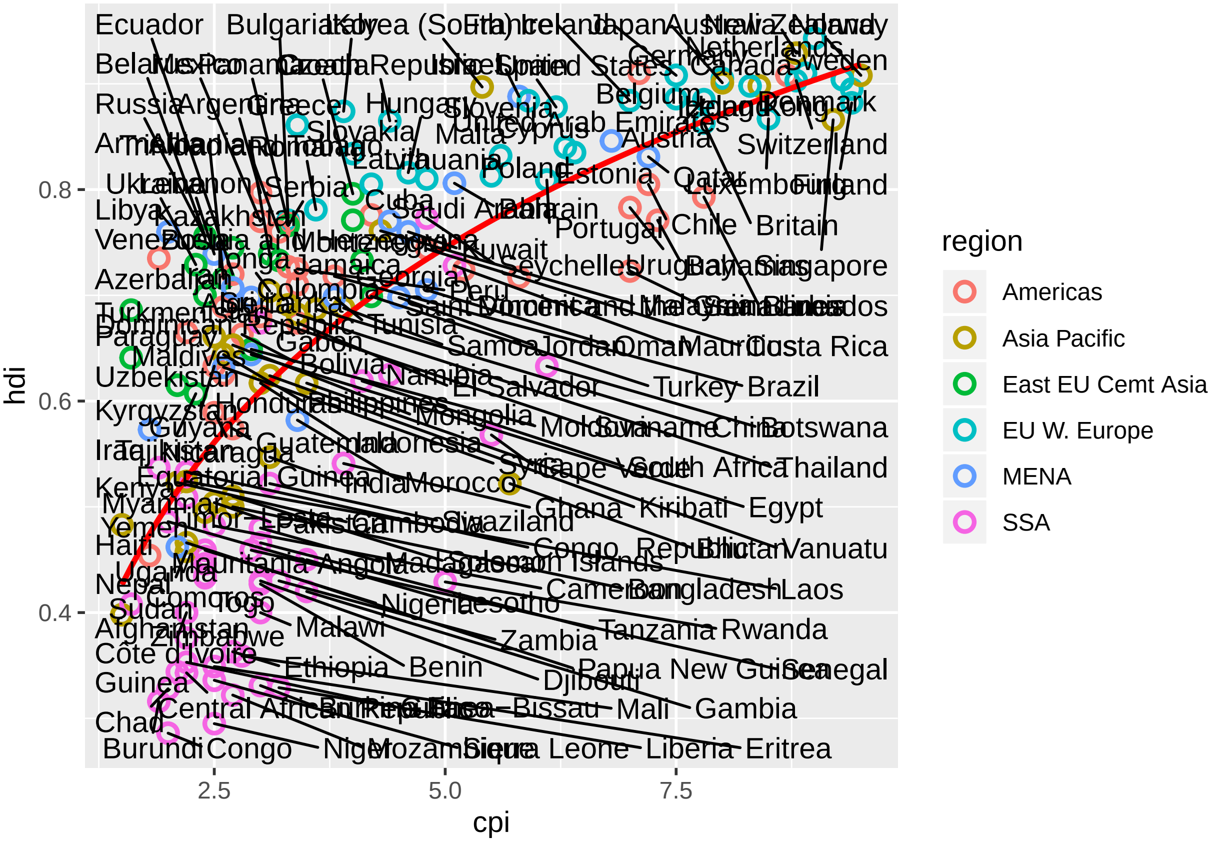
4. Now we will add the country names to the plot from (1).

For this we will need the package `ggrepel` because it makes this process easier. Install the package `ggrepel`. Use the function `geom_text_repel`

```
library(ggrepel)
```

a. Use `geom_text()` to add country names to our plot.

```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region),
             shape = 1, size = 2.5, stroke = 1.25) +
  geom_smooth(se= FALSE,
              method = "lm",
              formula = y ~ x + log(x),
              color = "red")  +
  geom_text_repel(aes(label = country))
```

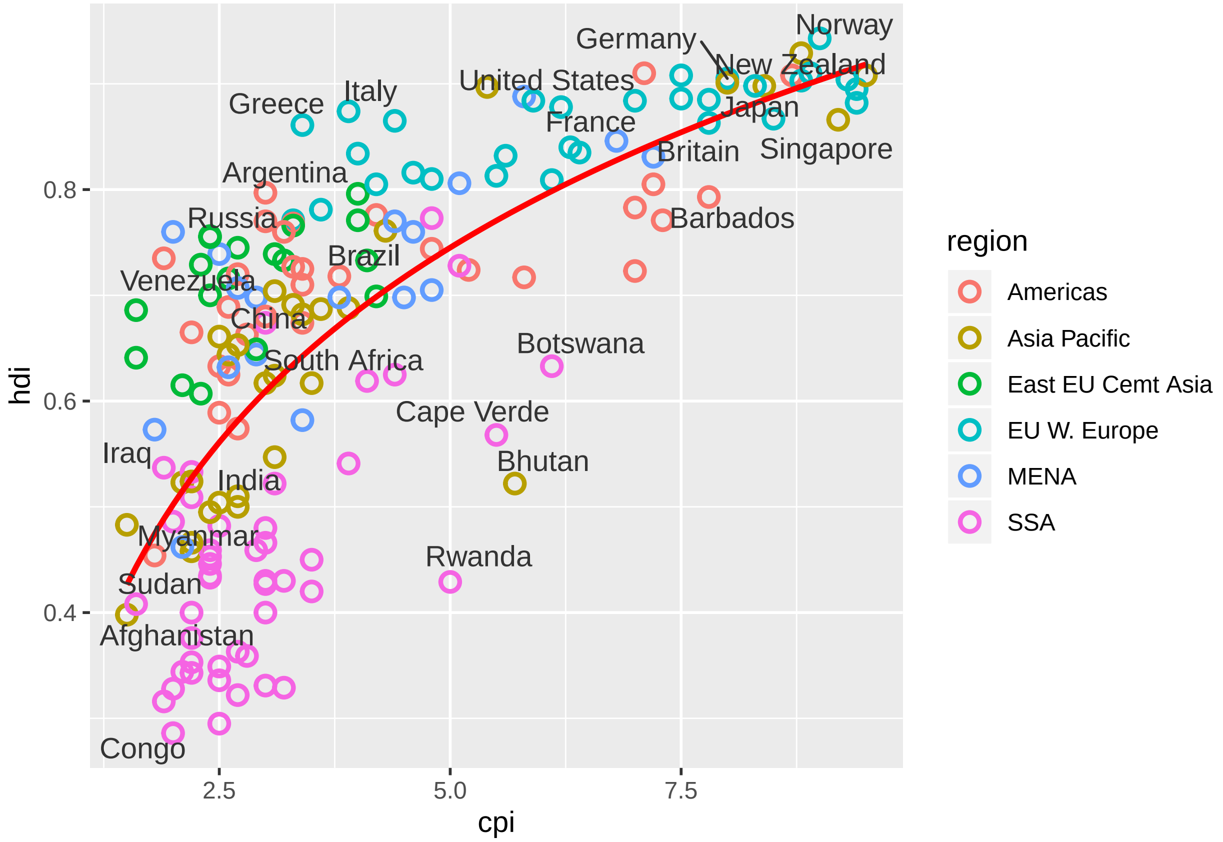


b. We might not want *all* the points labelled. Create the vector

```
points_to_label <- 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")
```

Now adjust the code in (a) to only label these points

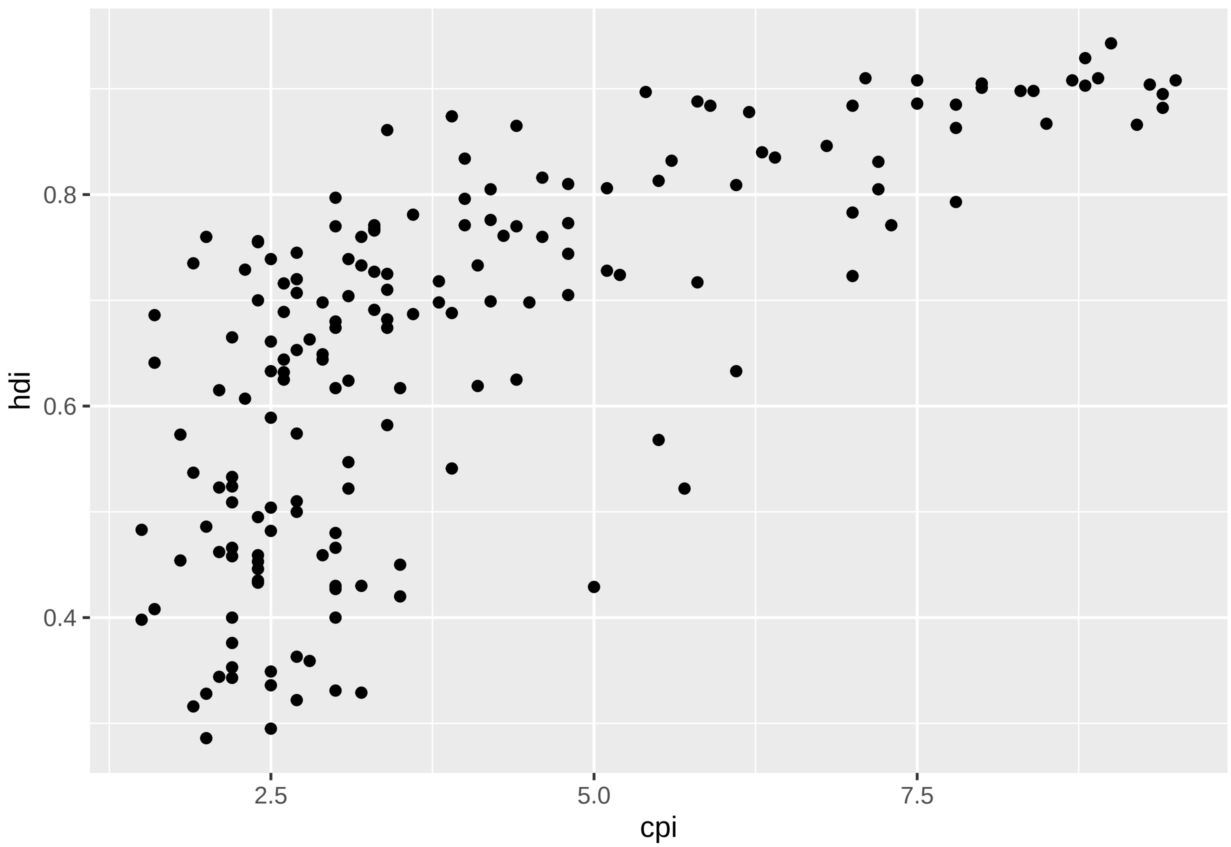
```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region),
             shape = 1, size = 2.5, stroke = 1.25) +
  geom_smooth(se= FALSE,
              method = "lm",
              formula = y ~ x + log(x),
              color = "red")  +
  geom_text_repel(aes(label = country),
                color = "gray20",
                data = filter(df, country %in% points_to_label),
                force = 10)
```



5. Now let's combine what we learned above, and from the class notes to build up a presentable notes. Proceed as follows:

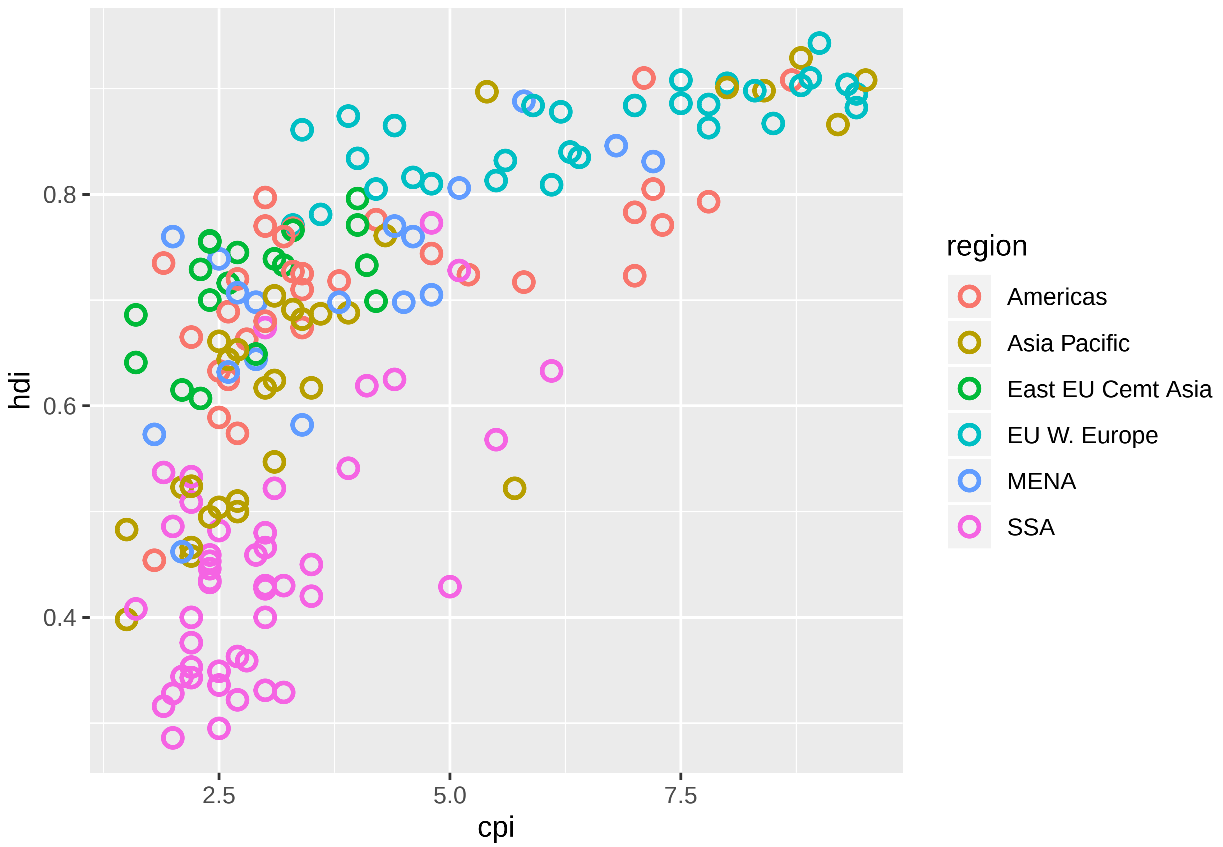
a. Create the simple scatter plot

```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point()
```



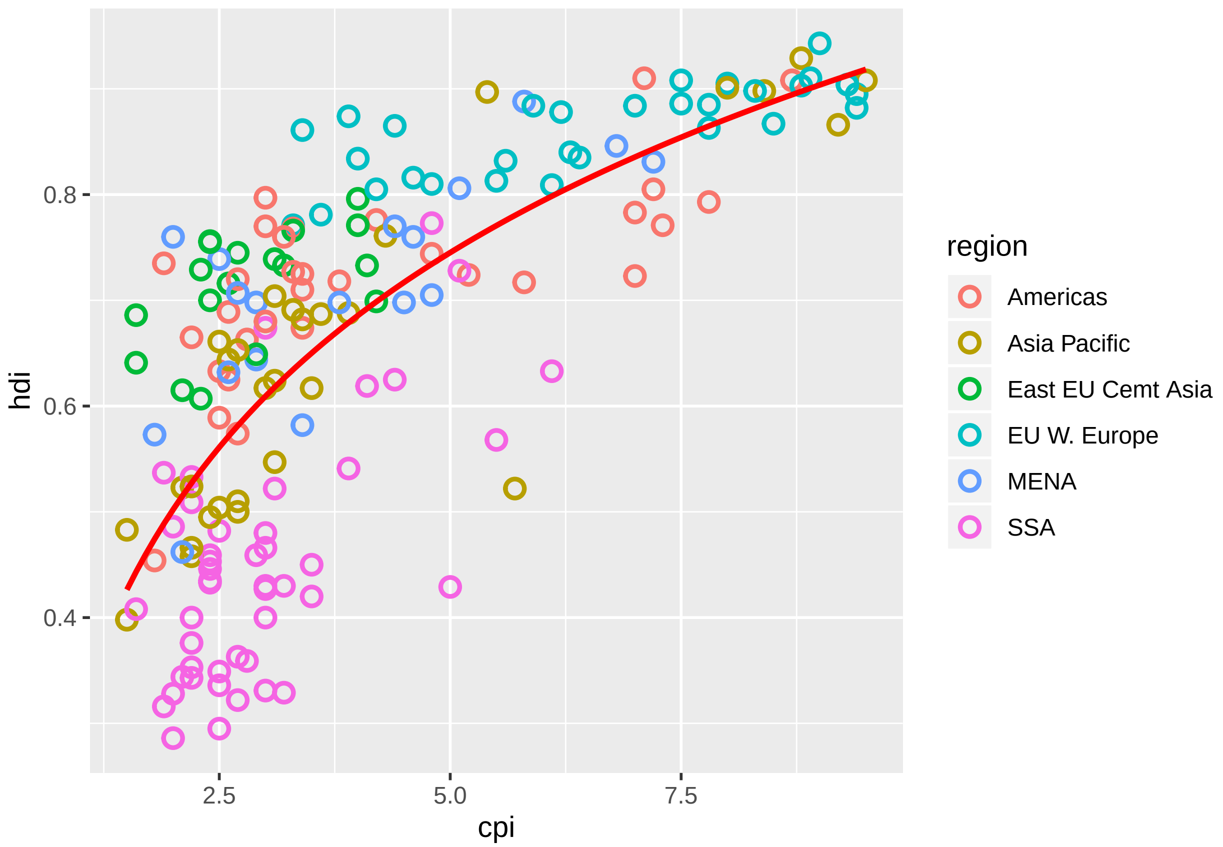
b. Make the points hollow, and colored by region. Adjust the size of the dots to make them easier to see.

```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region),
             shape = 1, size = 2.5, stroke = 1.25)
```



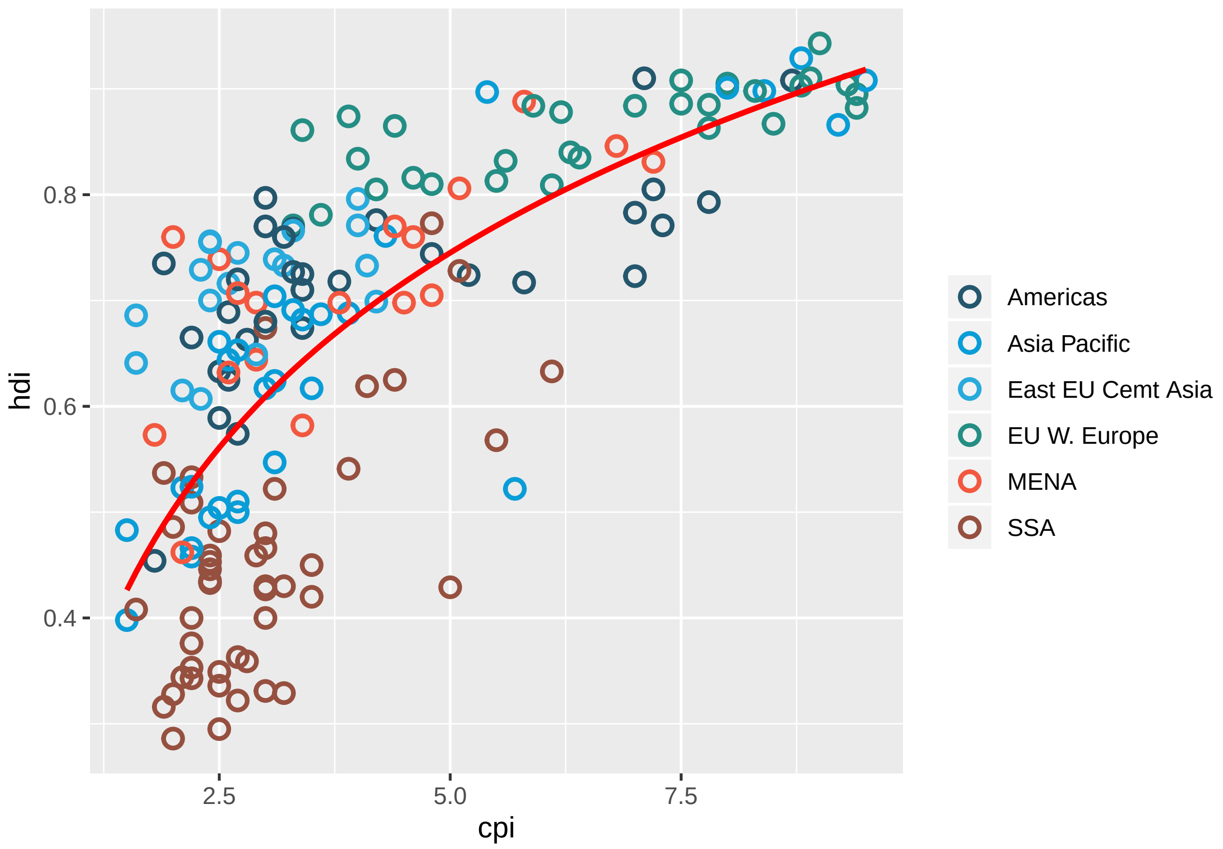
c. Add the line `y ~ x + log(x)`, without the confidence interval. Color the line red.

```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region),
             shape = 1, size = 2.5, stroke = 1.25) +
  geom_smooth(se= FALSE,
              method = "lm",
              formula = y ~ x + log(x),
              color = "red")
```



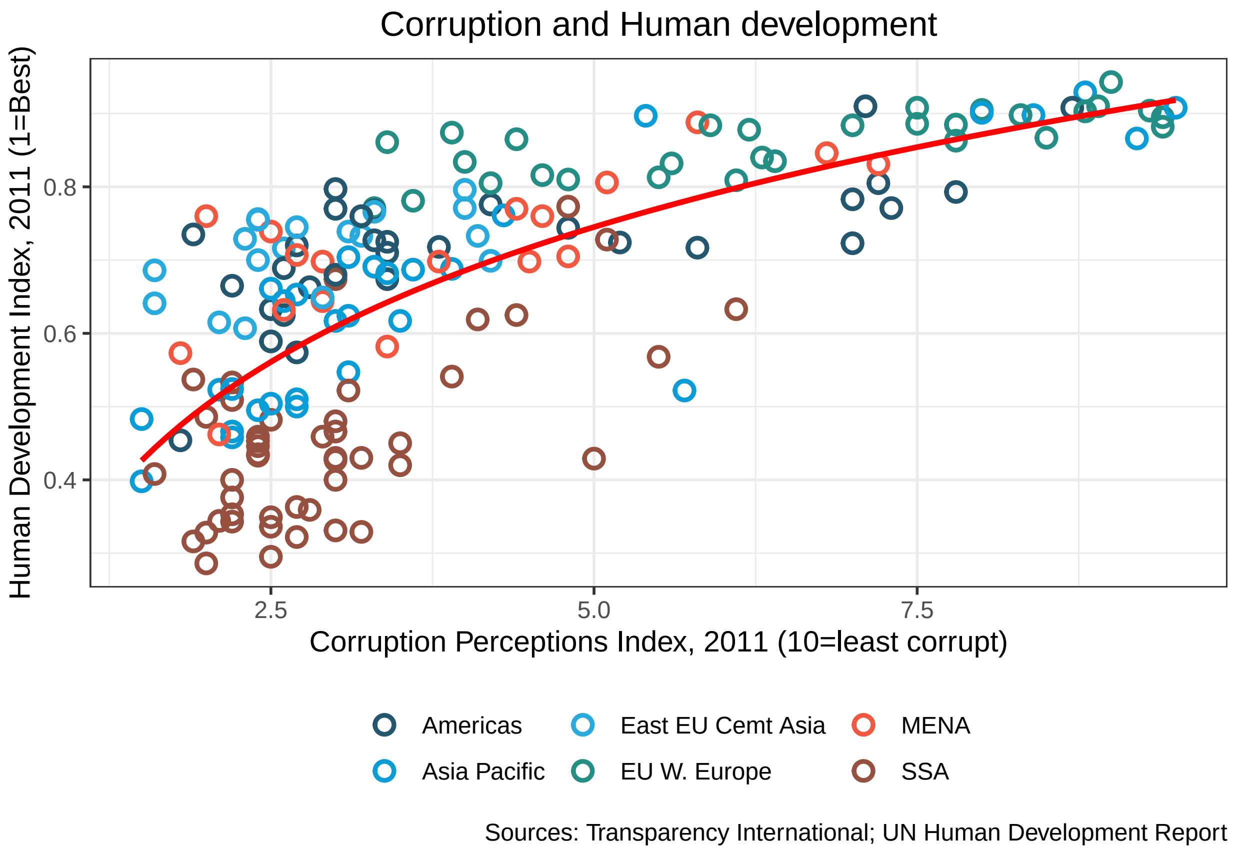
d. Change the color of the dots to be less ugly. I used `scale_color_manual()` but you don't need to.

```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region),
             shape = 1, size = 2.5, stroke = 1.25) +
  geom_smooth(se= FALSE,
              method = "lm",
              formula = y ~ x + log(x),
              color = "red") +
 scale_color_manual(name = "",
                    values = c("#24576D",
                               "#099DD7",
                               "#28AADC",
                               "#248E84",
                               "#F2583F",
                               "#96503F"))
```



e. Add meaningful x and y labels. And a title (which is centered). Can you add a note near the bottom of the figure to say that the data comes from "Transparency International and UN Human Development Report"?

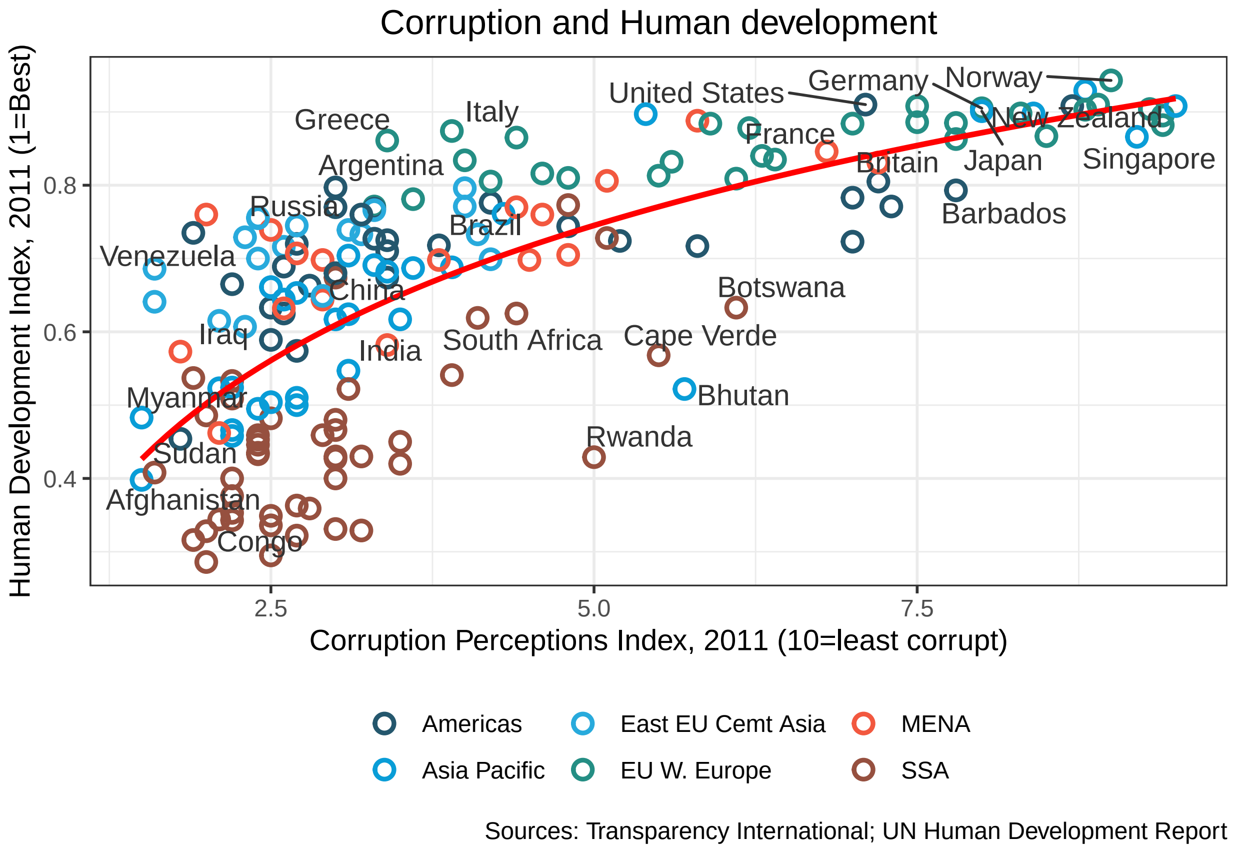
```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region),
             shape = 1, size = 2.5, stroke = 1.25) +
  geom_smooth(se= FALSE,
              method = "lm",
              formula = y ~ x + log(x),
              color = "red") +
 scale_color_manual(name = "",
                     values = c("#24576D",
                                "#099DD7",
                                "#28AADC",
                                "#248E84",
                                "#F2583F",
                                "#96503F")
                    ) +
  theme_bw() +
  theme(legend.position = "bottom",
        plot.title = element_text(hjust = 0.5)) +
  xlab("Corruption Perceptions Index, 2011 (10=least corrupt)") +
  ylab("Human Development Index, 2011 (1=Best)") +
  ggtitle("Corruption and Human development", subtitle = waiver()) +
   labs(caption="Sources: Transparency International; UN Human Development Report")
```



Sources: Transparency International; UN Human Development Report

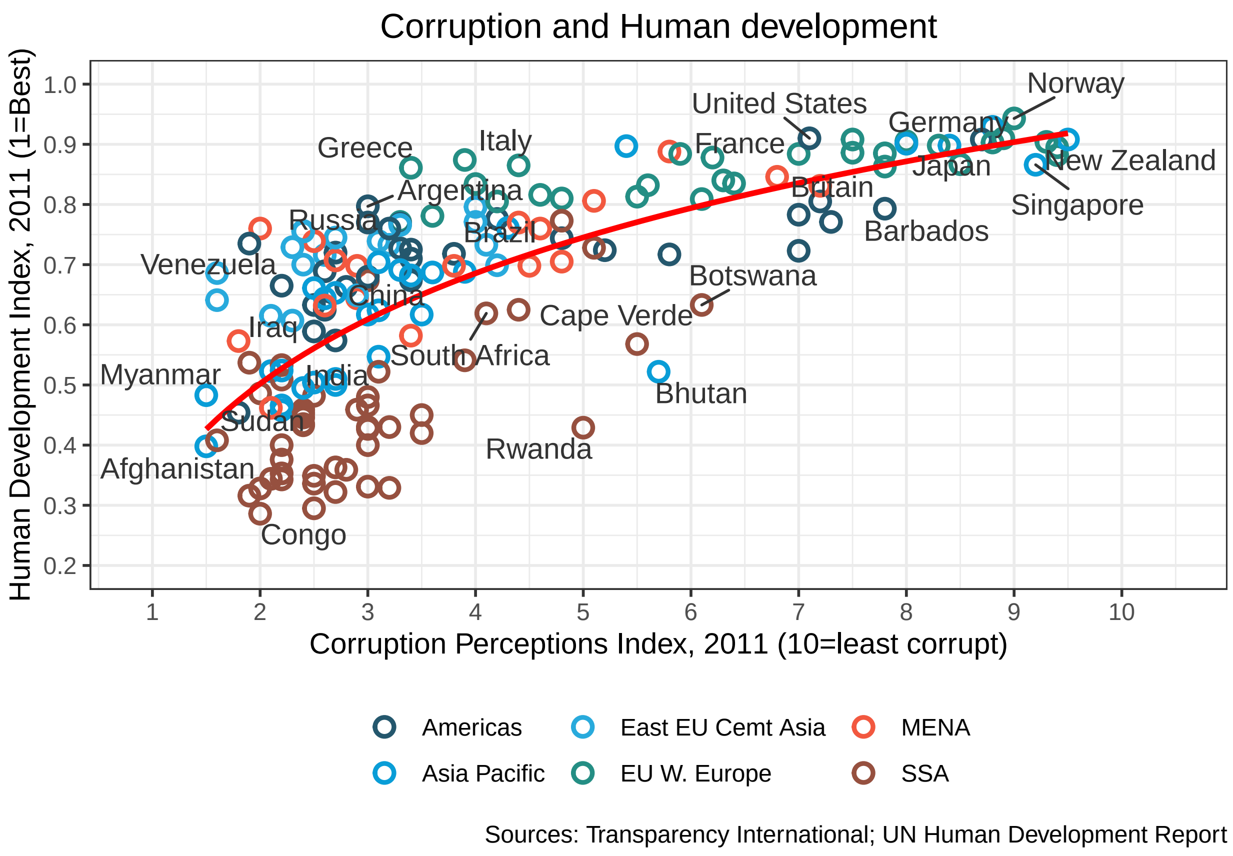
f. Label the points from `points_to_label` in 4b.

```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region),
             shape = 1, size = 2.5, stroke = 1.25) +
  geom_smooth(se= FALSE,
              method = "lm",
              formula = y ~ x + log(x),
              color = "red") +
 scale_color_manual(name = "",
                    values = c("#24576D",
                               "#099DD7",
                               "#28AADC",
                               "#248E84",
                               "#F2583F",
                               "#96503F")
                   ) +
    geom_text_repel(aes(label = country),
               color = "gray20",
                 data = filter(df, country %in% points_to_label),
                 force = 10) +
  theme_bw() +
  theme(legend.position = "bottom",
        plot.title = element_text(hjust = 0.5)) +
  xlab("Corruption Perceptions Index, 2011 (10=least corrupt)") +
  ylab("Human Development Index, 2011 (1=Best)") +
  ggtitle("Corruption and Human development", subtitle = waiver()) +
   labs(caption="Sources: Transparency International; UN Human Development Report")
```



g. Adjust the x and y axes to have a better range, and set of axis ticks. You are free to choose what you like.

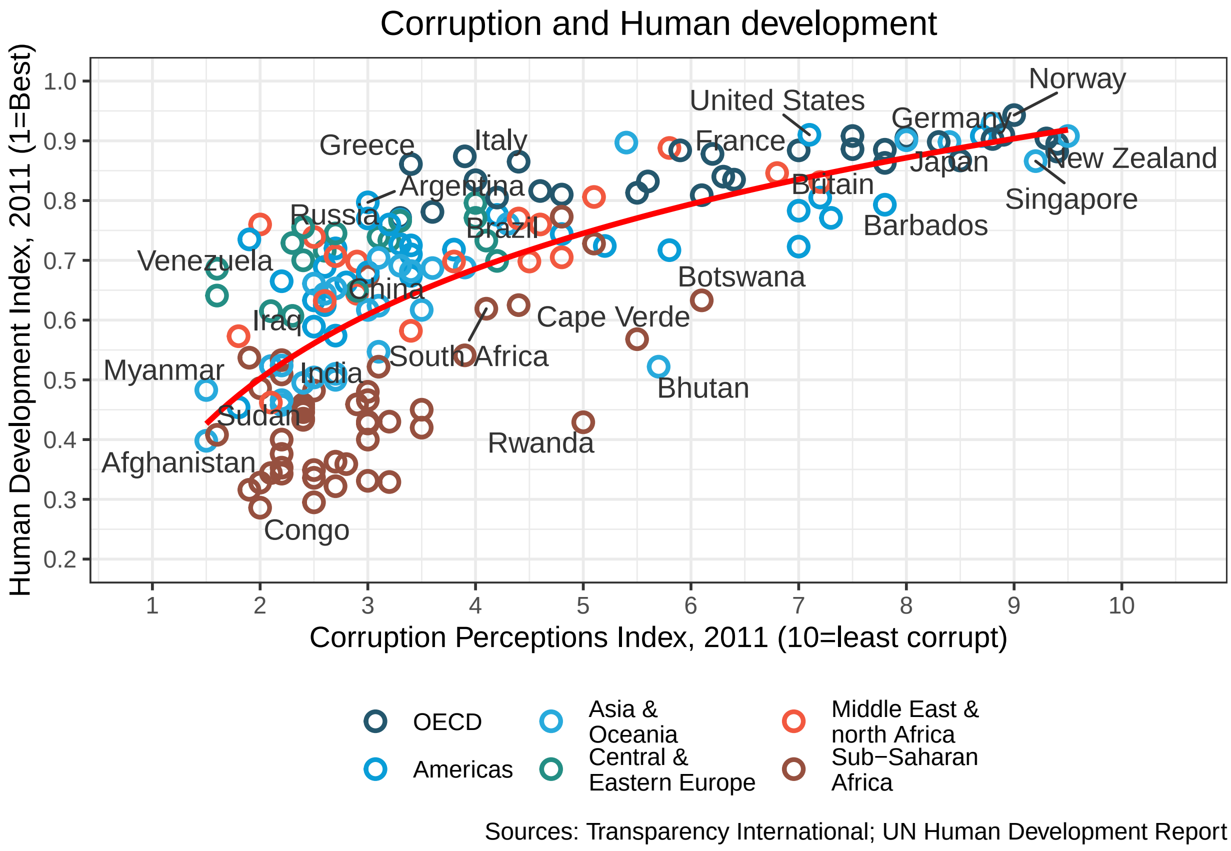
```
df %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region),
             shape = 1, size = 2.5, stroke = 1.25) +
  geom_smooth(se= FALSE,
              method = "lm",
              formula = y ~ x + log(x),
              color = "red") +
 scale_color_manual(name = "",
                    values = c("#24576D",
                               "#099DD7",
                                "#28AADC",
                                "#248E84",
                                "#F2583F",
                                "#96503F")
                    ) +
    geom_text_repel(aes(label = country),
                color = "gray20",
                data = filter(df, country %in% points_to_label),
                force = 10) +
   scale_x_continuous(
    limits = c(.9, 10.5),
    breaks = 1:10) +
  scale_y_continuous(
    limits = c(0.2, 1.0),
    breaks = seq(0.2, 1.0, by = 0.1)
  ) +
  theme_bw() +
  theme(legend.position = "bottom",
        plot.title = element_text(hjust = 0.5)) +
  xlab("Corruption Perceptions Index, 2011 (10=least corrupt)") +
  ylab("Human Development Index, 2011 (1=Best)") +
  ggtitle("Corruption and Human development", subtitle = waiver()) +
   labs(caption="Sources: Transparency International; UN Human Development Report")
```



h. Move the legend to the bottom of the plot. Adjust the legend names so that they are easier to read and more meaningful. The easiest way to do this is to use `dplyr` to recode the region variable as a factor, and give it appropriate labels. Using the help menu for `factor` should help you here.

```r
df2 <- df %>%
  mutate(region = factor(region,
                     levels = c("EU W. Europe",
                                "Americas",
                                "Asia Pacific",
                                "East EU Cemt Asia",
                                "MENA",
                                "SSA"),
                     labels = c("OECD",
                                "Americas",
                                "Asia &\nOceania",
                                "Central &\nEastern Europe",
                                "Middle East &\nnorth Africa",
                                "Sub-Saharan\nAfrica")
                     )
        )

df2 %>%
  ggplot(aes(x = cpi, y = hdi)) +
  geom_point(aes(color = region),
             shape = 1, size = 2.5, stroke = 1.25) +
  geom_smooth(se= FALSE,
              method = "lm",
              formula = y ~ x + log(x),
              color = "red")  +
  geom_text_repel(aes(label = country),
                  color = "gray20",
                  data = filter(df, country %in% points_to_label),
                  force = 10) +
  scale_color_manual(name = "",
                     values = c("#24576D",
                                "#099DD7",
                                "#28AADC",
                                "#248E84",
                                "#F2583F",
                                "#96503F")) +
  theme_bw() +
  theme(legend.position = "bottom",
        plot.title = element_text(hjust = 0.5)) +
  xlab("Corruption Perceptions Index, 2011 (10=least corrupt)") +
  ylab("Human Development Index, 2011 (1=Best)") +
  ggtitle("Corruption and Human development", subtitle = waiver()) +
  scale_x_continuous(
    limits = c(.9, 10.5),
    breaks = 1:10) +
  scale_y_continuous(
    limits = c(0.2, 1.0),
    breaks = seq(0.2, 1.0, by = 0.1)
) +
labs(caption="Sources: Transparency International; UN Human Development Report")
```


Corruption and Human development
Human Development Index, 2011 (1=Best)
Corruption Perceptions Index, 2011 (10=least corrupt)
Norway
United States
Germany
Greece
Italy
France
Japan
New Zealand
Argentina
Britain
Singapore
Russia
Brazil
Barbados
Venezuela
China
Botswana
Iraq
Cape Verde
Myanmar
India
South Africa
Bhutan
Sudan
Rwanda
Afghanistan
Congo
OECD
Americas
Asia & Oceania
Central & Eastern Europe
Middle East & north Africa
Sub−Saharan Africa
Sources: Transparency International; UN Human Development Report