Lab 1

YOUR NAME

30 September 2021

Goals for today

For the first lab, we will reverse engineer some plots using a relatively small, historical dataset. This should help you practice the "grammar of graphics" as well as the process of downloading, completing, and knitting your assignments. This lab is based on the structure of the first assignment.

Load and look at the data

Don't worry if you cannot interpret the code above. If you can, excellent! All you need to know is that our data is now in the environment and saved as an object called `df` (for "dataframe").

Before we worry about visualizing the data, let's look at what it contains. Run the chunk below to view the column headings and first several rows of data.

```
head(df)  #show col names and several rows of data
```

```
## # A tibble: 6 x 8
##   Country         Year Proportion_Cons~ Proportion_Cons~ Proportion_Impo~ Tons_Imported
##   <chr>          <dbl>            <dbl>            <dbl>            <dbl>         <dbl>
## 1 United Kingdom  1893             0.02             100.               NA         25000
## 2 United Kingdom  1894             0                100                NA          7000
## 3 United Kingdom  1895             0.01             100.               NA         14000
## 4 United Kingdom  1896             0.01             100.               NA         16000
## 5 United Kingdom  1897             0.01             100.               NA          8000
## 6 United Kingdom  1898             0.01             100.               NA         10000
## # ... with 2 more variables: Tons_Produced <dbl>, Tons_Exported <dbl>
```

What are the variables in the data set?

YOUR ANSWER HERE

What does each row represent? (i.e. what is the *unit of observation*?)

YOUR ANSWER HERE

Notice that the first several rows of "Proportion_Imported_From_UK" are empty. Why might this data be missing? (hint: look at column 1)

YOUR ANSWER HERE

First plot

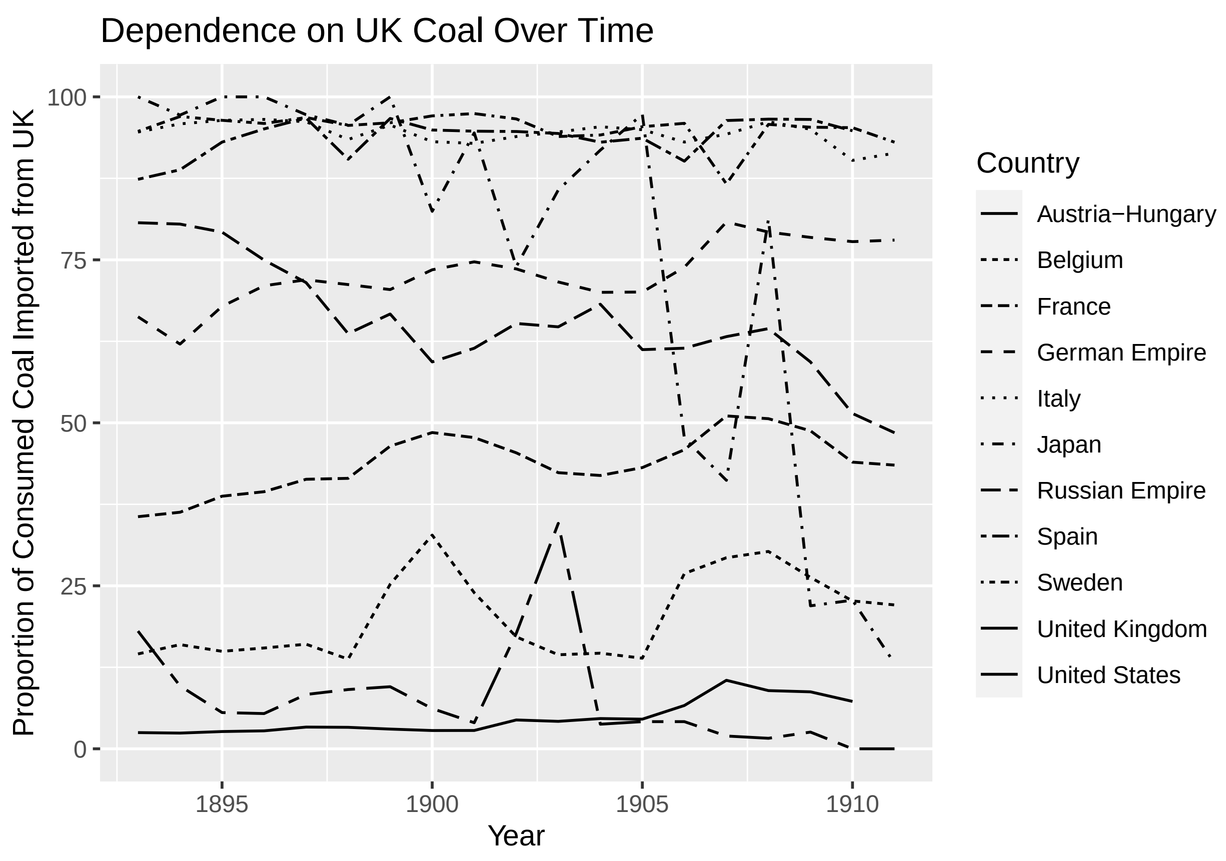
For the first plot, most of the code has been filled in for you. Finish the code to recreate this image.

Incomplete code:

```
plot_1 <- ggplot(data = df,
                 mapping = aes(x = Year,
                               y = ______,
                               linetype = ____)) +
  geom_line() +
  labs(x = "Year",
       y = "Proportion of Consumed Coal Imported from UK",
       ____ = "Dependence on UK Coal Over Time")

plot_1
```

```r
plot_1 <- ggplot(data = df,
                 mapping = aes(x = Year,
                               y = Proportion_Imported_From_UK,
                               linetype = Country)) +
  geom_line() +
  labs(x = "Year",
       y = "Proportion of Consumed Coal Imported from UK",
       title = "Dependence on UK Coal Over Time")

plot_1
```

```
## Warning: Removed 21 row(s) containing missing values (geom_path).
```



Second plot

For the second plot, more of the code is missing. See if you can recreate the image. (hint, the line being fitted is a linear model)

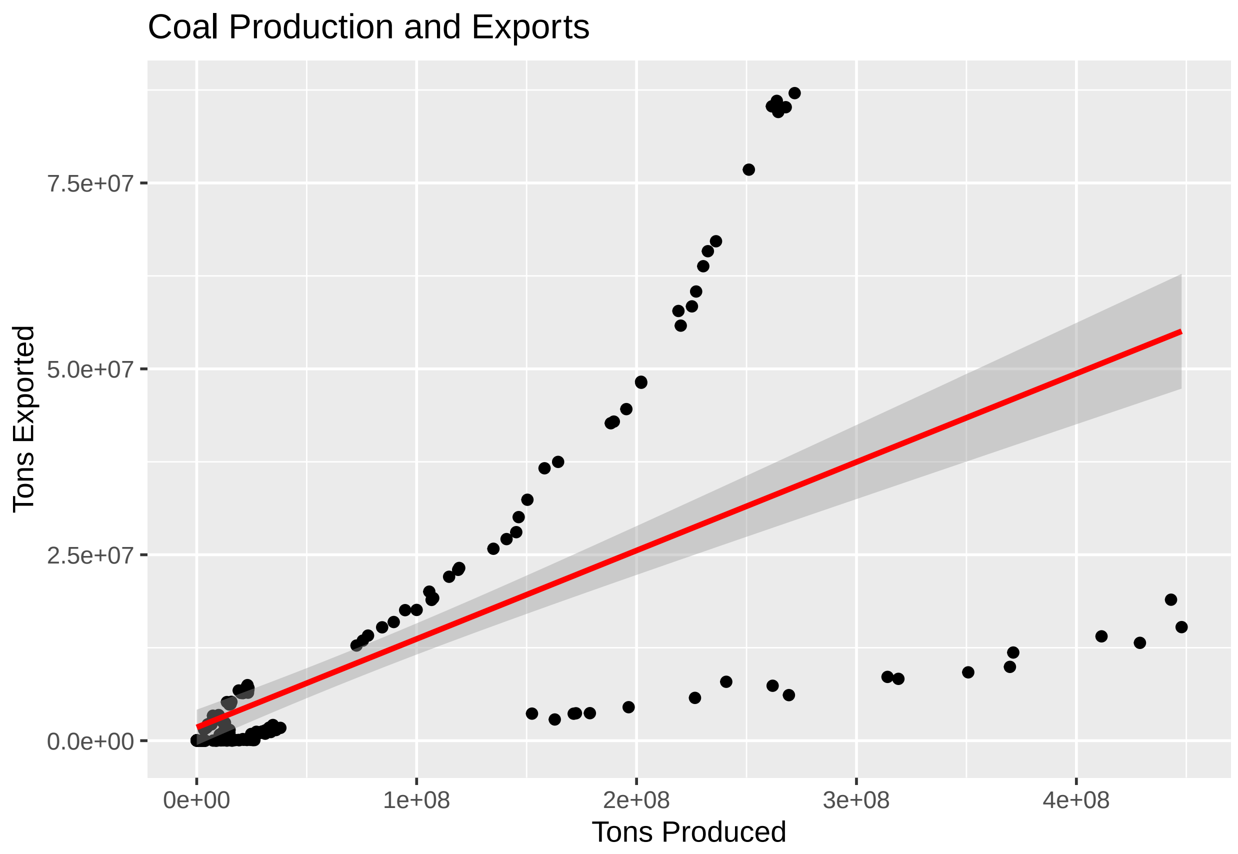
Incomplete code:

```
plot_2 <- ggplot(data = df,
                 mapping = aes(x = Tons_Produced,
                               y = Tons_Exported)) +
  geom_point() +
  geom_smooth(method = "lm",
              color = "red") +
  labs(x = "Tons Produced",
       y = "Tons Exported",
       title = "Coal Production and Exports")

plot_2
```

```
plot_2 <- ggplot(data = df,
                 mapping = aes(x = Tons_Produced,
                               y = Tons_Exported)) +
  geom_point() +
  geom_smooth(method = "lm",
              color = "red") +
      labs(x = "Tons Produced",
           y = "Tons Exported",
           title = "Coal Production and Exports")

    plot_2
```

```
## Warning: Removed 2 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 2 rows containing missing values (geom_point).
```



What relationship does this plot represent?

YOUR ANSWER HERE

Does the overall trend (as represented by the line) fit your expectations?

YOUR ANSWER HERE

Why do you think there is a cluster of points in the bottom right corner?

YOUR ANSWER HERE

What about the points in the lower left?

YOUR ANSWER HERE

Third Plot

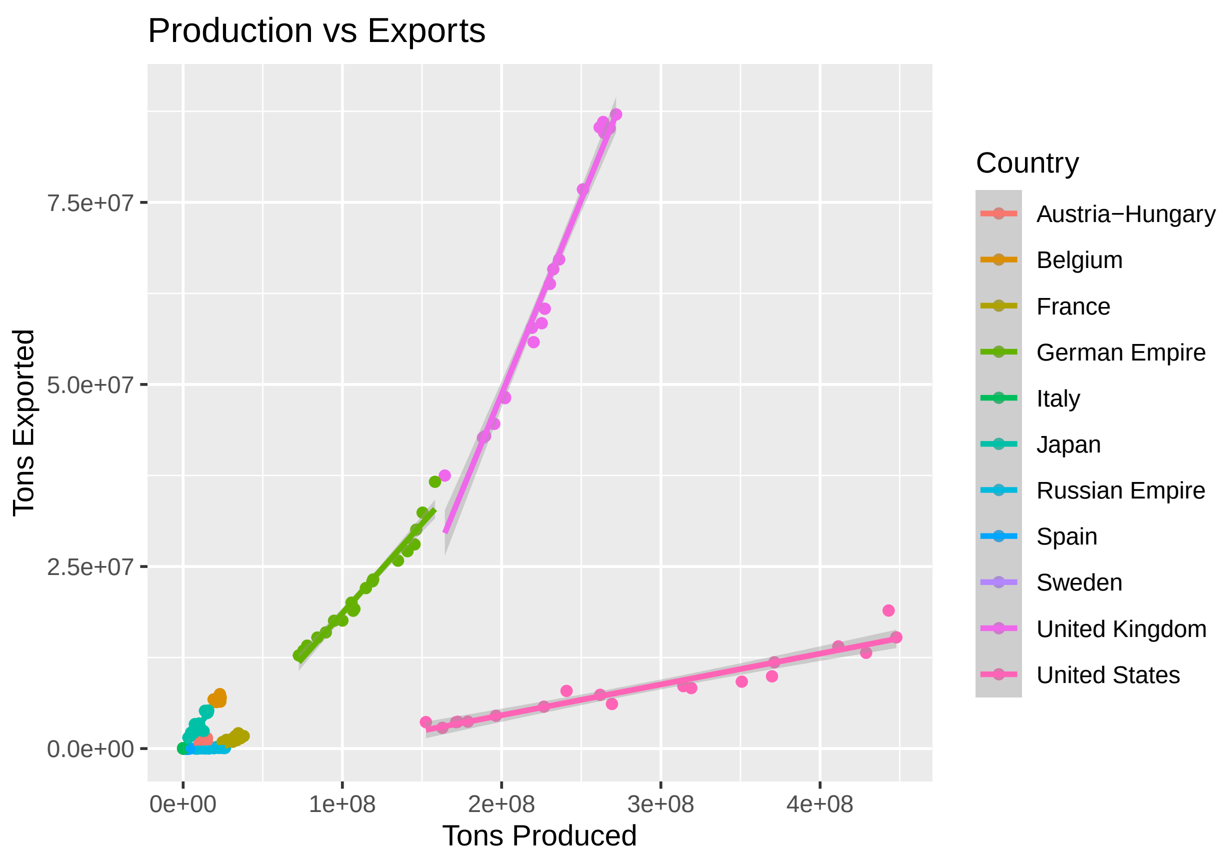
Since we suspect that the clustering is the result of different countries exporting different proportions of the coal they produce, we can alter the code above slightly to color the points by country and fit a separate line for each country. Complete the code to produce this plot.

Incomplete code:

```
plot_3 <- ggplot(data = df,
                      mapping = aes(x = Tons_Produced,
                                    y = Tons_Exported,
                                    color = Country)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(x = "Tons Produced",
  y = "Tons Exported",
  title = "Production vs Exports")

plot_3
```

```r
plot_3 <- ggplot(data = df,
                      mapping = aes(x = Tons_Produced,
                                    y = Tons_Exported,
                                    color = Country)) +
  geom_point() +
  geom_smooth(method = "lm") +
  labs(x = "Tons Produced",
  y = "Tons Exported",
  title = "Production vs Exports")

plot_3
```

```
## Warning: Removed 2 rows containing non-finite values (stat_smooth).

## Warning: Removed 2 rows containing missing values (geom_point).
```



What issues do you see with this plot?

YOUR ANSWER HERE

How might we address this concern with a different visualization?

YOUR ANSWER HERE

Fourth Plot

None of the code is provided for you. Think about what you can use from the previous plot and what has changed.

```
#YOUR CODE HERE
```

While colors can be a powerful visualization tool, line type, point shape, and other visual elements can make your work more accessible to people who are colorblind. Alternatively, there are packages outside the tidyverse (e.g. `ggthemes`*) which contain colorblind friendly palettes or allow you to manually define a palette using several different color reference systems.*