MTHM501 Working with data | Visualisation

Mark Kelson

Introduction

In this practical we will introduce graphics in `R`. One of the more popular methods of creating graphics in `R` is to use the ggplot language for graphics which is implemented in the `ggplot2` package. This way of creating graphs is flexible and modular, as you build a plot by layering different plot components.

Preliminaries

We need the following packages

- `ggplot2` - Package to implement the ggplot language for graphics in `R`.

Make sure that these packages are downloaded and installed in `R`. We use the `require()` function to load them into the `R` library.

```r
# Loading packages
require(ggplot2)
require(raster)
```

Data

To create plots, we will use the `mtcars` dataset available within `R`. This dataset consists of fuel consumption and other aspects of automobile design and performance for 32 cars from the 1973-74 Motor Trend US magazine. We load this dataset, using the `data()` function.

```r
# Loading mtcars dataset
data(mtcars)
```

Make sure you are familiar with the contents of this dataset before continuing on with the rest of this practical, by typing `?mtcars` into `R`.

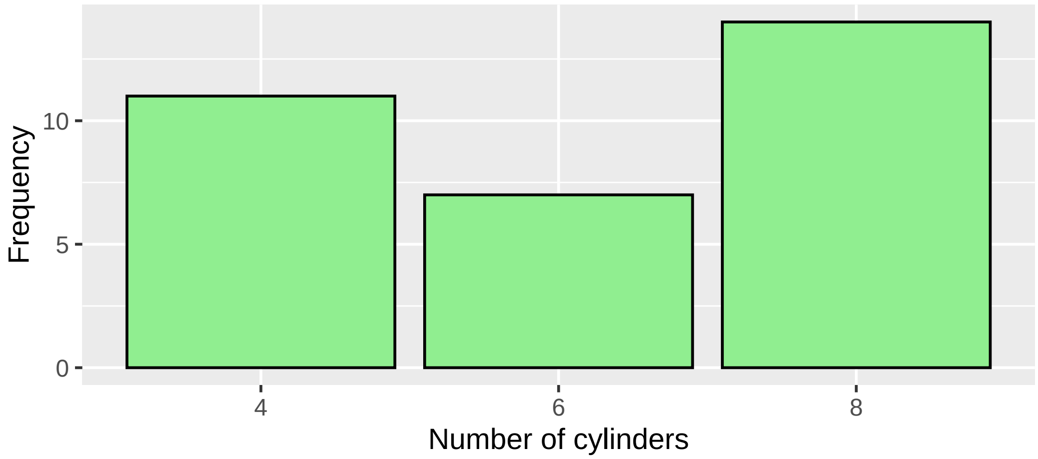
Visualisation

Barcharts

Bar charts can be used to display frequencies. We are interested in the number of cars that have 4, 6 and 8 cylinders being tested in the `mtcars` dataset. Let's display this information as a bar chart.

We use the `ggplot()` function to select the data that we want to display, then we use the `geom_bar()` to tell `R` we want to display the data as a bar chart.

```r
# Bar chart of cars by number of cylinders using ggplot
ggplot(mtcars, aes(x = factor(cyl))) + # ggplot with the desired data
  geom_bar(fill=’lightgreen’,colour=’black’) + # Specifying a bar chart
  labs(x="Number of cylinders", y="Frequency") # Axes labels
```

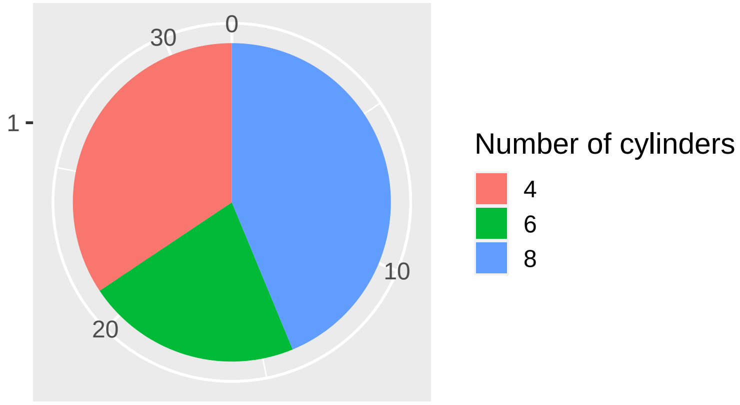


Pie Charts

Pie charts can be used to display variables where proportions are important. We are interested in the proportion of cars that have 4, 6 and 8 cylinders being tested in the `mtcars` dataset. Let’s display this information as a pie chart.

To do this we create a bar chart as before similar to the one above but we add an extra option to say we want a pie chart.

```r
# Pie chart of cars by number of cylinders using ggplot
ggplot(mtcars, aes(x = factor(1), fill=factor(cyl))) + # ggplot with the desired data
  geom_bar(width = 1) + # A bar chart
  coord_polar(theta = "y") + # Specifying a pie chart
  labs(x="",y="",fill=’Number of cylinders’) # Blank Axes labels
```

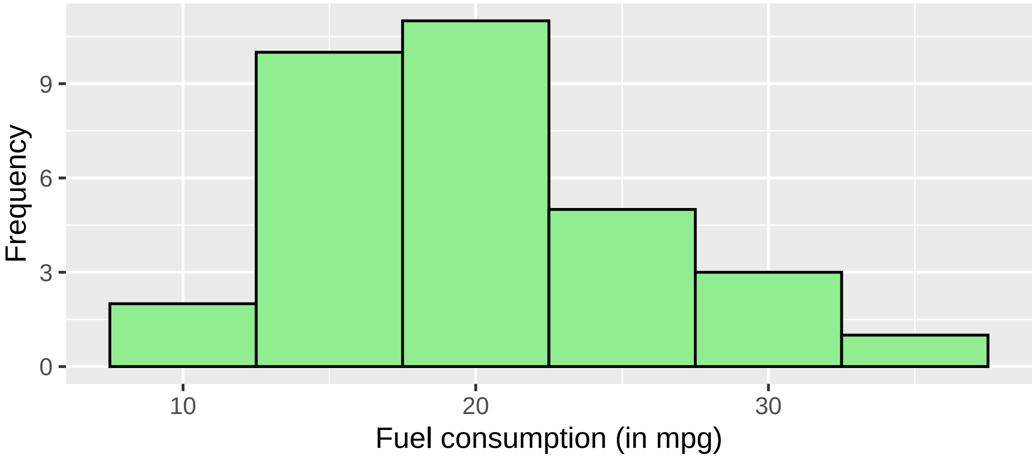


Histograms

Histograms can be used to display frequencies of quantitative variable using frequencies. We are interested in the distribution of fuel consumption (in miles per gallon) of cars tested in the `mtcars` dataset. Let’s display this information as a histogram.

We use the `ggplot()` function to select the data that we want to display, then we use the `geom_histogram()` to tell `R` we want to display the data as a scatter plot.

```
# Histogram of fuel consumption using ggplot
ggplot(mtcars, aes(x=mpg)) +  # ggplot with the desired data
  geom_histogram(binwidth=5, fill='lightgreen',colour='black') + # Specifying bar chart
  labs(x="Fuel consumption (in mpg)", y="Frequency") # Axes labels
```



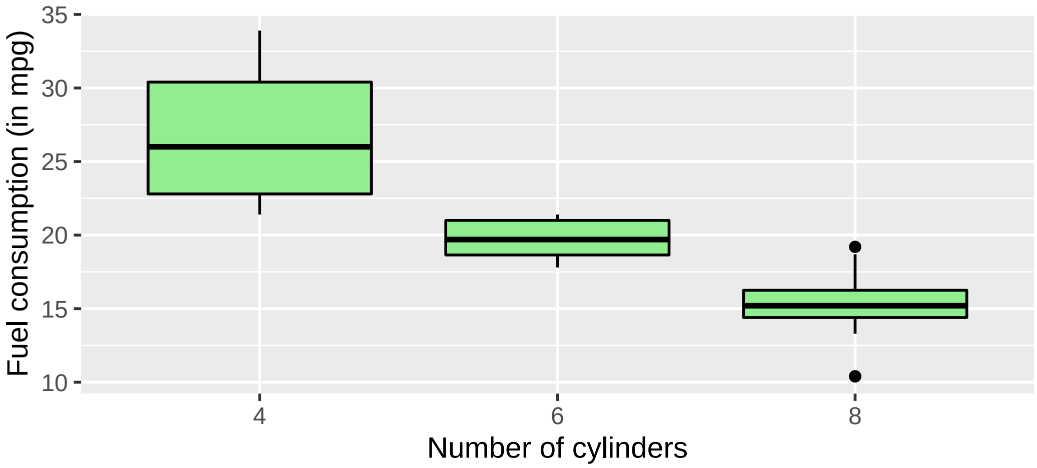
Box Plots

Box plots can be used to display median and variability in data. The central line is drawn at the median, and the box extends from the lower quartile to the upper quartile. The lines that extend from the box indicate three interquartile ranges, with any data outside this range shown using dots.

We are interested in the distribution of fuel consumption (in miles per gallon) of cars with 4, 6 and 8 cylinders in the `mtcars` dataset separately. Let's display this information as a boxplot.

We use the `ggplot()` function to select the data that we want to display, then we use the `geom_boxplot()` to tell `R` we want to display the data as a box plot.

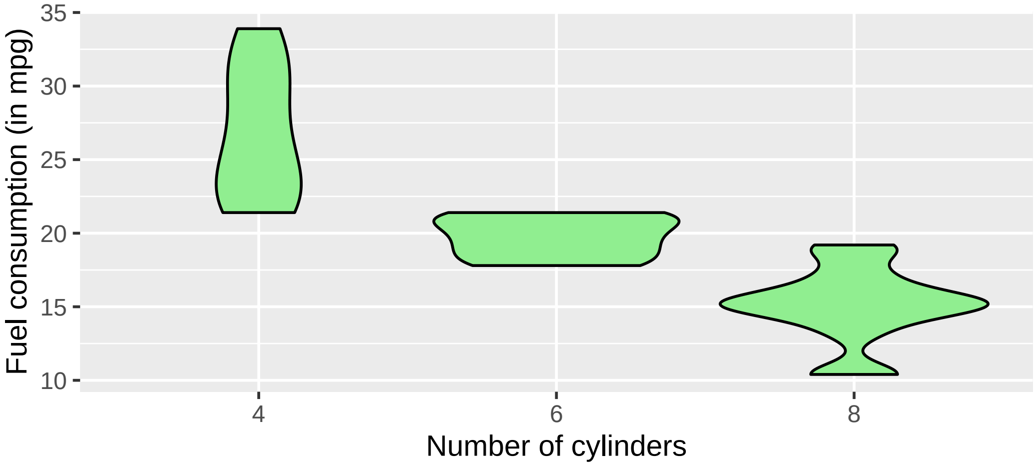
```
# Boxplot of fuel consumption using ggplot
ggplot(mtcars, aes(x=factor(cyl),y=mpg)) + # ggplot with the desired data
  geom_boxplot(fill='lightgreen',colour='black') + # Specifying boxplot
  labs(x="Number of cylinders",y="Fuel consumption (in mpg)") # Axes labels
```



Violin plots

We might want to incorporate more information about the distribution of the underlying subsets of data (rather than just depicting a big block indicating where the middle 50% of data lies like in a boxplot). Violin plots are one solution.

```
# Violin plot of fuel consumption using ggplot
ggplot(mtcars, aes(x=factor(cyl),y=mpg)) + # ggplot with the desired data
  geom_violin(fill='lightgreen',colour='black') + # Specifying boxplot
  labs(x="Number of cylinders",y="Fuel consumption (in mpg)") # Axes labels
```

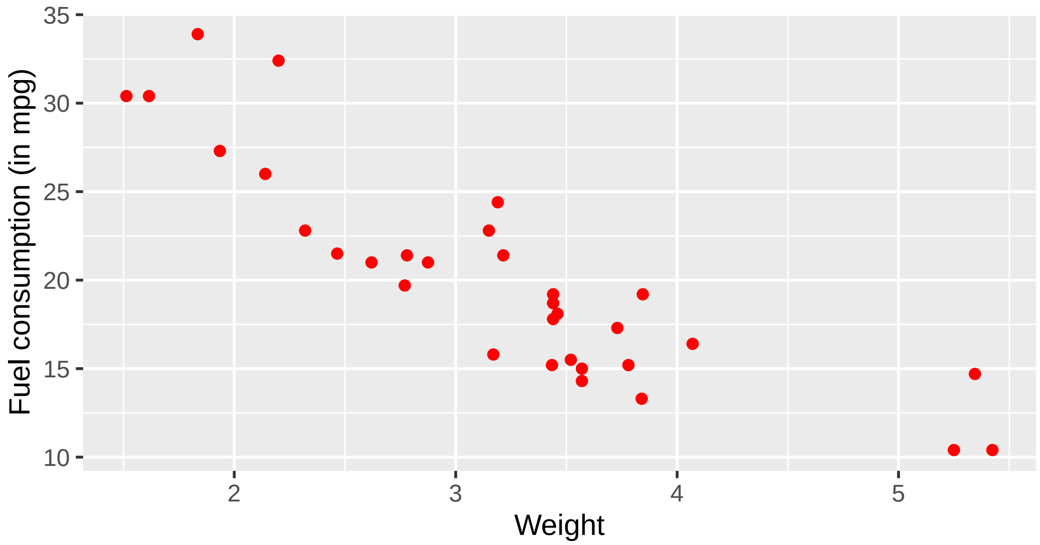


Scatter plots

Scatter plots can be used to display pairs of values of two quantitative variables, often to test for a correlation or association of the variables. We are interested in seeing the relationship between weight and fuel consumption (in miles per gallon) of cars in the `mtcars` dataset. Let's display this information as a scatter plot.

We use the `ggplot()` function to select the data that we want to display, then we use the `geom_point()` to tell `R` we want to display the data as a scatter plot.

```
# Scatter plot of cars weight by fuel consumption using ggplot
ggplot(data = mtcars, aes(x=wt,y=mpg)) + # ggplot with the desired data
  geom_point(colour='red') + # Specifying a scatter plot
  labs(x="Weight", y="Fuel consumption (in mpg)") # Axes labels
```



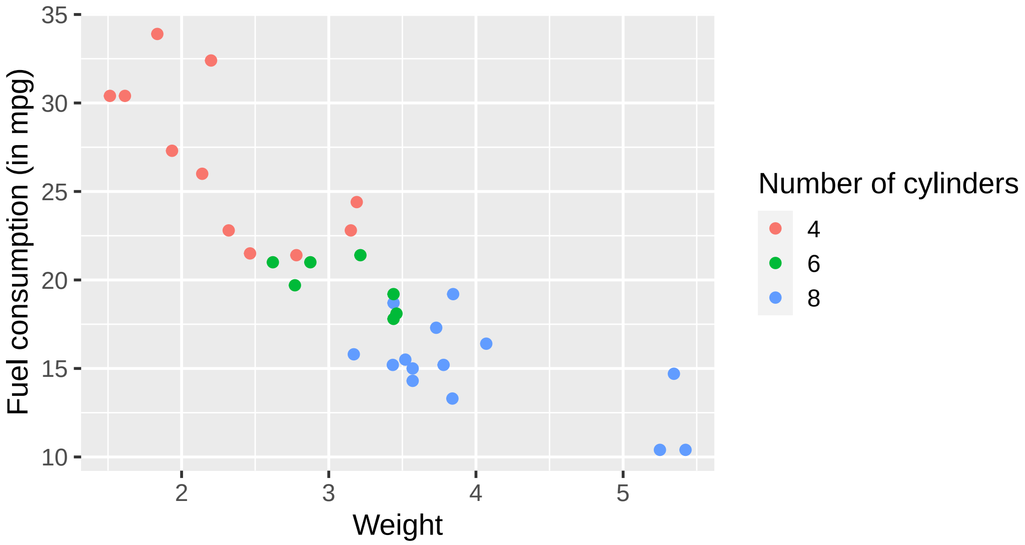
We can see that there is a negative relationship between weight and fuel consumption (in miles per gallon), so as weight increases, the fuel consumption decreasess.

Customising your plot

Adding colour

We created a scatter plot of weight against fuel consumption (in miles per gallon) and saw that there is a negative relationship between weight and fuel consumption. We are interested in seeing if this relationship is different between cars with 4, 6, and 8 cylinders. We can colour points in a scatter plot by a variable in our dataset by using the `colour` option in the aesthetics of the `ggplot()` function.

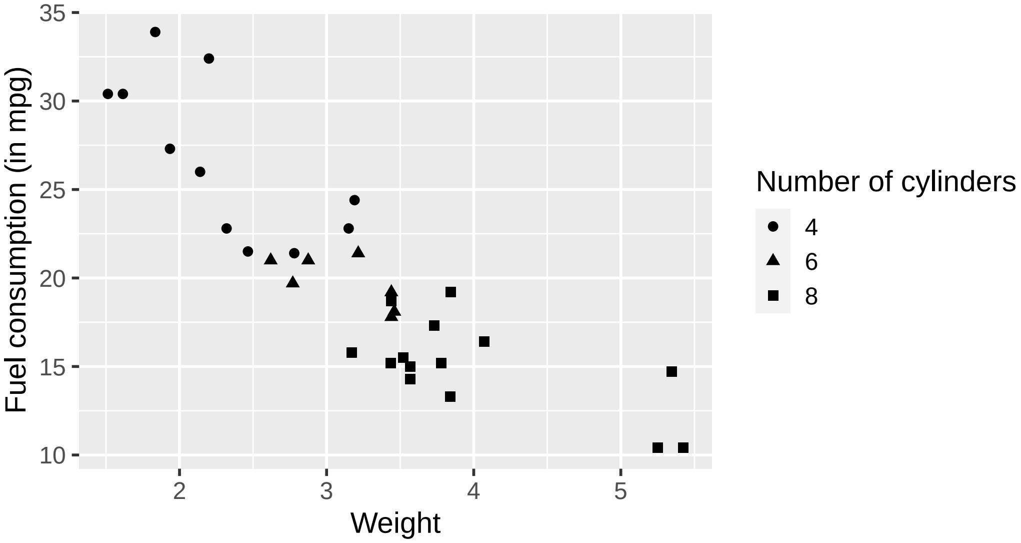
```r
# Scatter plot of cars weight by fuel consumption using ggplot
ggplot(data = mtcars, aes(x=wt,y=mpg,colour=factor(cyl))) +
  # ggplot with the desired data
  geom_point() + # Specifying that we want it to be a scatter plot
  labs(x="Weight", y="Fuel consumption (in mpg)",colour=’Number of cylinders’) # Axes labels
```



Using plotting character symbols

Some journals restrict the use of colour in plots. Instead of adding colour to a scatter plot, we can change the style of the points in a scatter plot by a variable in our dataset. We do this using the `shape` option in the aesthetics of the `ggplot()` function.

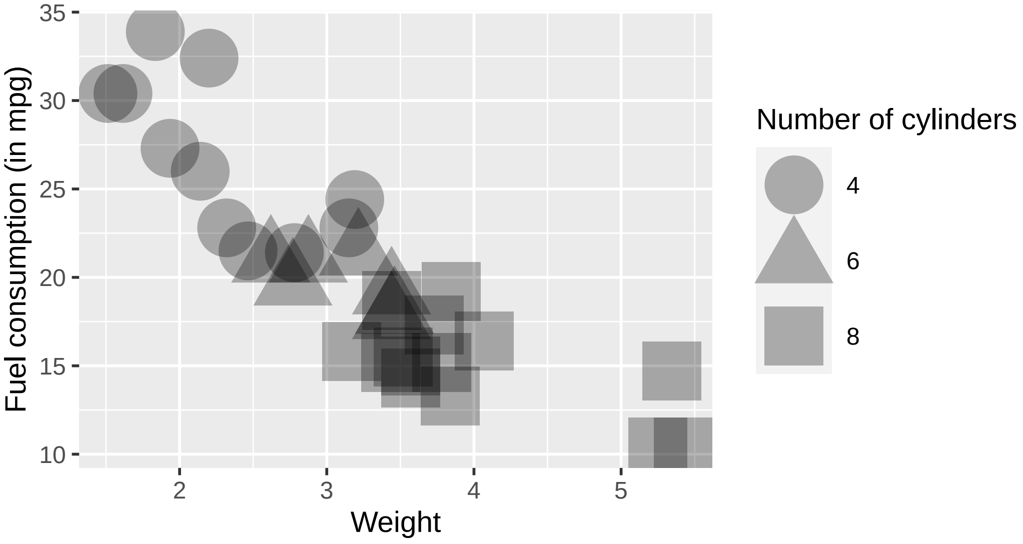
```
# Scatter plot of cars weight by fuel consumption using ggplot
ggplot(data = mtcars, aes(x=wt,y=mpg,shape=factor(cyl))) +
  # ggplot with the desired data
  geom_point(aes(x=wt,y=mpg)) + # Specifying that we want it to be a scatter plot
  labs(x="Weight", y="Fuel consumption (in mpg)",shape='Number of cylinders') # Axes labels
```



Altering point size and transparency

We can alter the size and transparency of points in plots using 'cex' (for size, the default is 1) and 'alpha' to set the transparency (0=fully transparent and 1 = fully opaque). In the figure below we have dramatically increased the point size so that some overlap. Adjusting the transparency helps us see all of the points.

```
# Scatter plot of cars weight by fuel consumption using ggplot
ggplot(data = mtcars, aes(x=wt,y=mpg,shape=factor(cyl))) +
  # ggplot with the desired data
  geom_point(aes(x=wt,y=mpg),cex=10,alpha=0.3) + # Specifying that we want it to be a scatter plot
  labs(x="Weight", y="Fuel consumption (in mpg)",shape='Number of cylinders') # Axes labels
```

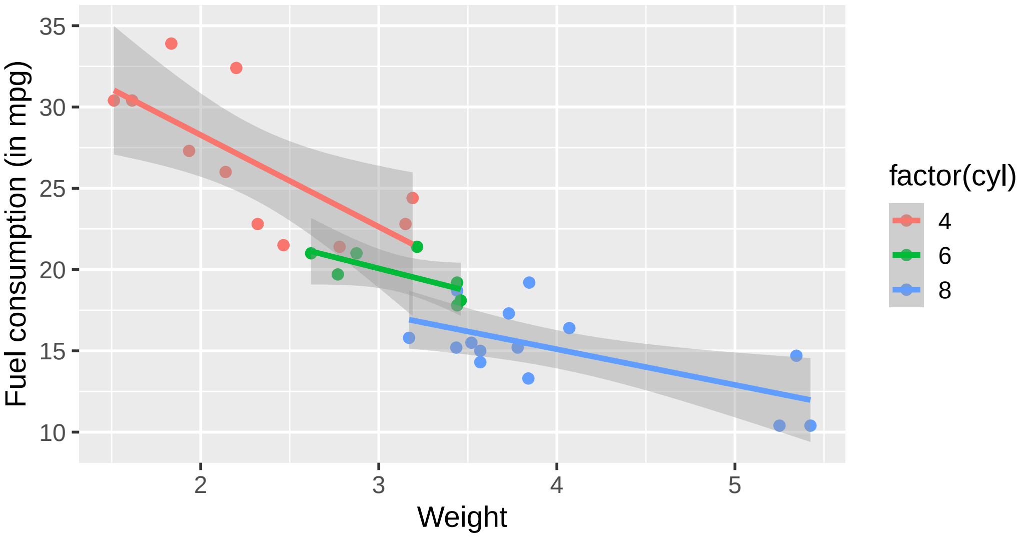


Adding trend lines

We see that while there is a relationship between weight and fuel consumption, it differs between 4, 6, and 8 cylinder cars.

We can use a function called 'geom_smooth' to provide an indication of the linear relationship between the x and y variables is.

```
# Scatter plot of cars weight by fuel consumption using ggplot
ggplot(data = mtcars, aes(x=wt,y=mpg,colour=factor(cyl))) +
  # ggplot with the desired data
  geom_point(aes(x=wt,y=mpg)) + # Specifying that we want it to be a scatter plot
  geom_smooth(method="lm") + # Indicating we want to add a linear trend to the plot
  labs(x="Weight", y="Fuel consumption (in mpg)",shape='Number of cylinders') # Axes labels
```

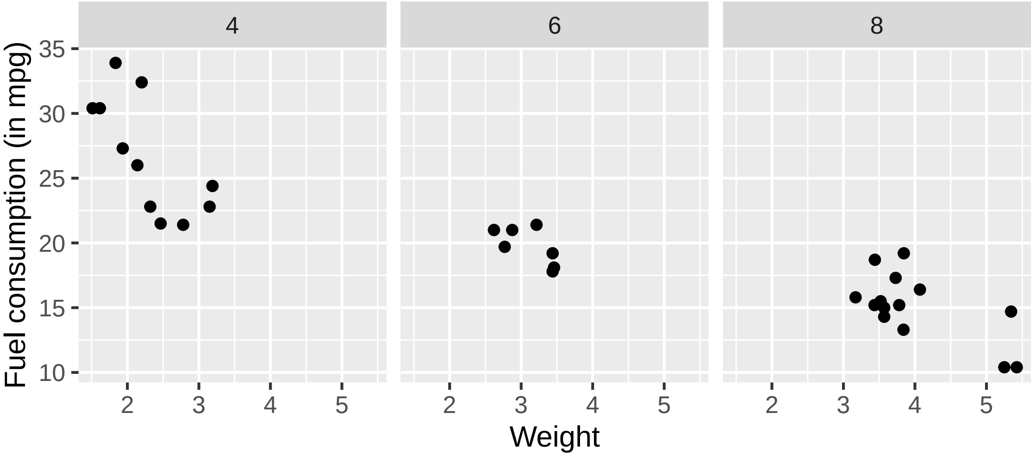


Facetting

There may be times where you cannot distinguish patterns or relationships by simply plotting the entire dataset in one plot. It may be useful when trying to find these patterns if different the data from different groups are displayed in separate panels.

We can do this in a straightforward fashion using ggplot2 package, but not using base R graphics. We created a scatter plot of weight against fuel consumption (in miles per gallon) and saw that there is a negative relationship between weight and fuel consumption. We want to see if the pattern differs between cars with 4, 6 and 8 cylinders. We add a call to the facet_grid() function to split the plot into different facets (or panels)

```
# Scatter plot of cars weight by fuel consumption using ggplot
ggplot(data = mtcars, aes(x=wt,y=mpg)) + # ggplot with the desired data
  geom_point(aes(x=wt,y=mpg)) + # Specifying that we want it to be a scatter plot
  labs(x="Weight", y="Fuel consumption (in mpg)") + # Axes labels
  facet_grid(. ~ cyl) # Facet split by columns
```



We see that while there is a relationship between weight and fuel consumption, it differs between 4, 6, and 8 cylinder cars.

Line Plots

To create plots, we will use the mtcars dataset available within R. The economics dataset contains information about the US economy across time. We load this dataset, using the data() function.

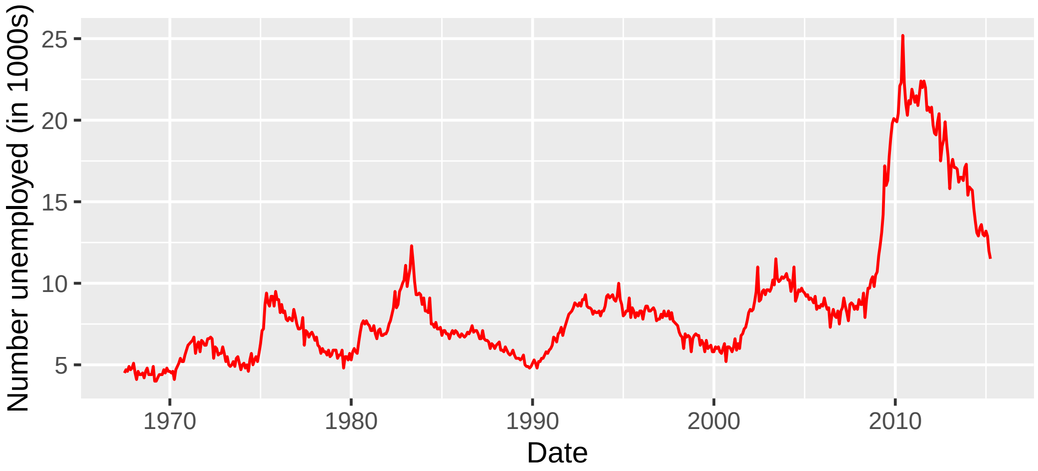
```
# Loading economics dataset
data(economics)
```

Make sure you are familiar with the contents of this dataset before continuing on with the rest of this practical, by typing ?economics into R.

Line plots can be used to show values of one or more variables measured over time, connected by a line. We are interested in seeing the number of people unemployed changing over time. The dataset economics in R contains this information. Let's display this information as a line plot.

We use the ggplot() function to select the data that we want to display, then we use the geom_line() to tell R we want to display the data as a line plot.

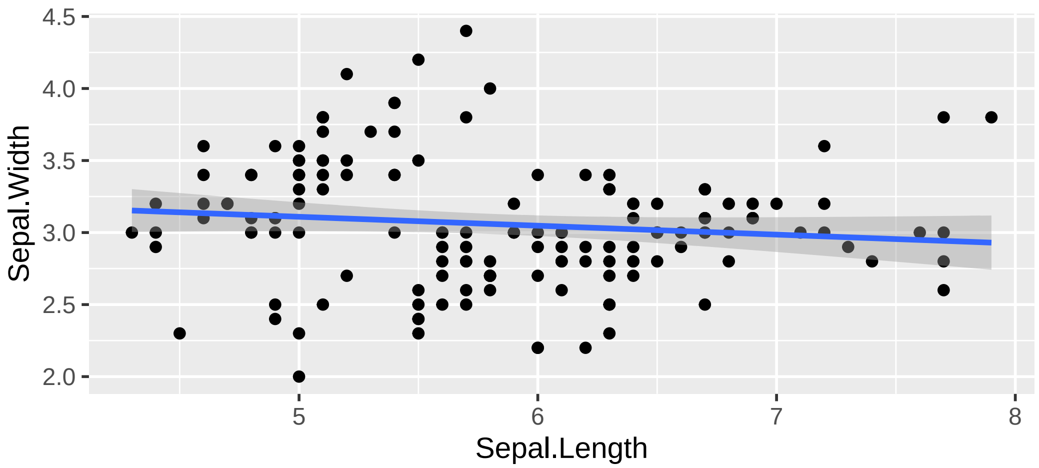
```
# Line plot of unemployment using ggplot2
ggplot(data=economics, aes(x=date,y=uempmed)) + # ggplot with the desired data
  geom_line(colour='red') + # Specifying a bar chart
  labs(x='Date',y='Number unemployed (in 1000s)') # Axes labels
```

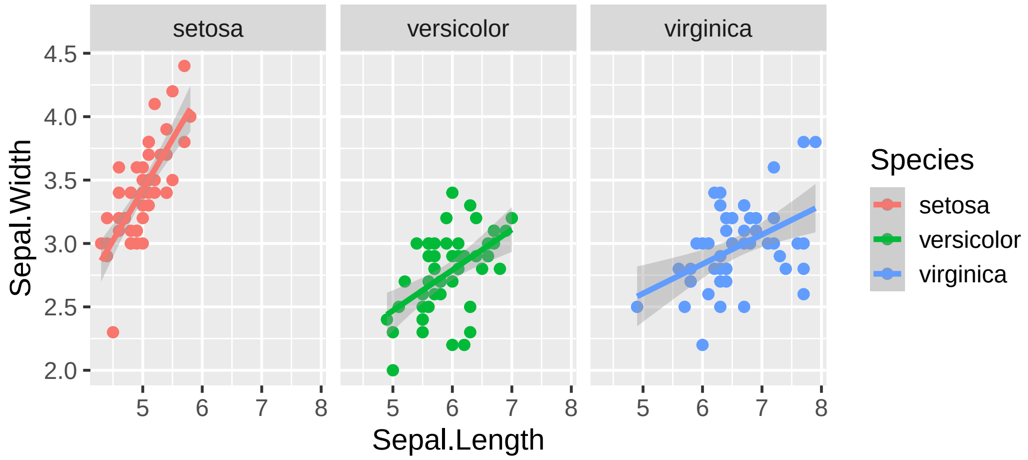


Exercises

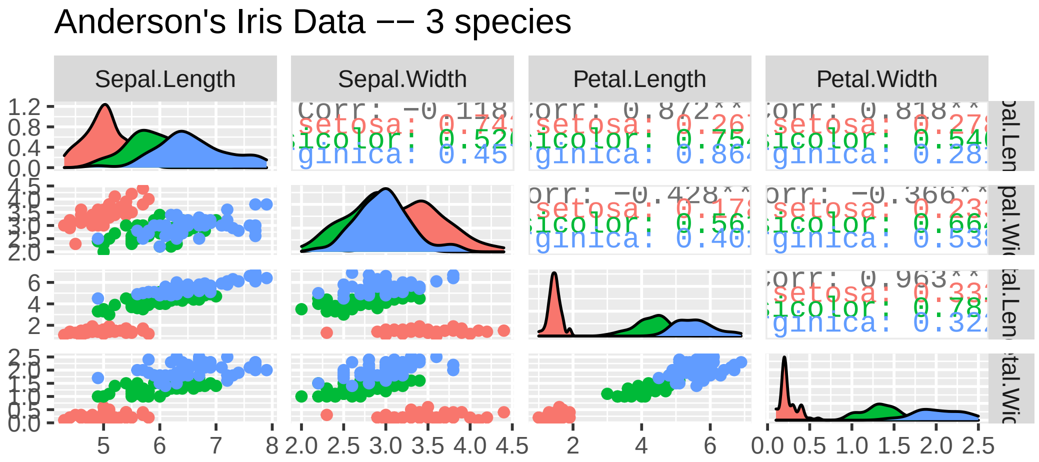
1. Recreate the plots below of the iris data we explored earlier in the course?
2. Change the axis labels and give the figure a title (use 'xlab', 'ylab' and 'ggtitle')?
3. Choose the colours manually (use 'scale_color_manual')?

(Optional exercise: Try loading the 'wesanderson' package and use 'scale_color_manual(values=wes_palette(n=3, name="GrandBudapest1"))' (help is available here http://www.sthda.com/english/wiki/ggplot2-colors-how-to-change-colors-automatically-and-manually))





4. Now we can try extending our knowledge. Use the ggpairs function in the GGally library to produce this plot. Feel free to use Google for help if you need it.


Anderson's Iris Data -- 3 species
Sepal.Length
Sepal.Width
Petal.Length
Petal.Width
Corr: -0.118
setosa: 0.743
icolor: 0.526
ginica: 0.457
orr: 0.872***
setosa: 0.267
icolor: 0.754
ginica: 0.864
orr: 0.818***
setosa: 0.278
icolor: 0.546
ginica: 0.281
orr: -0.428***
setosa: 0.178
icolor: 0.561
ginica: 0.401
orr: -0.366***
setosa: 0.233
icolor: 0.664
ginica: 0.538
orr: 0.963***
setosa: 0.332
icolor: 0.787
ginica: 0.322
pal.Len
pal.Wid
tal.Len
tal.Wid