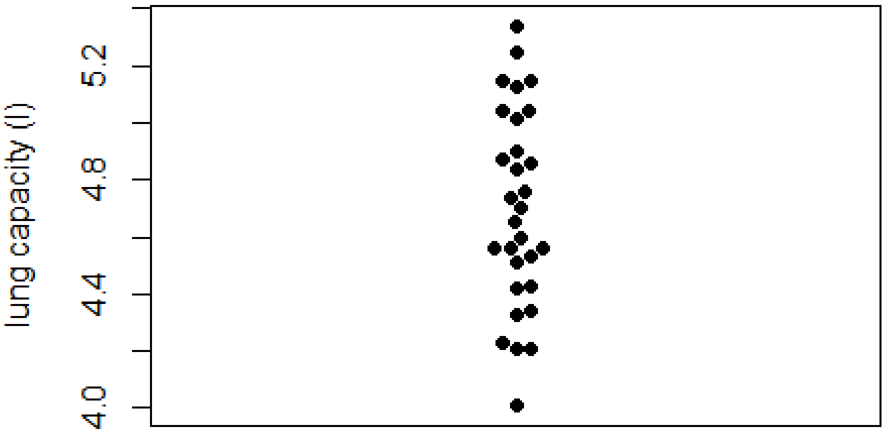
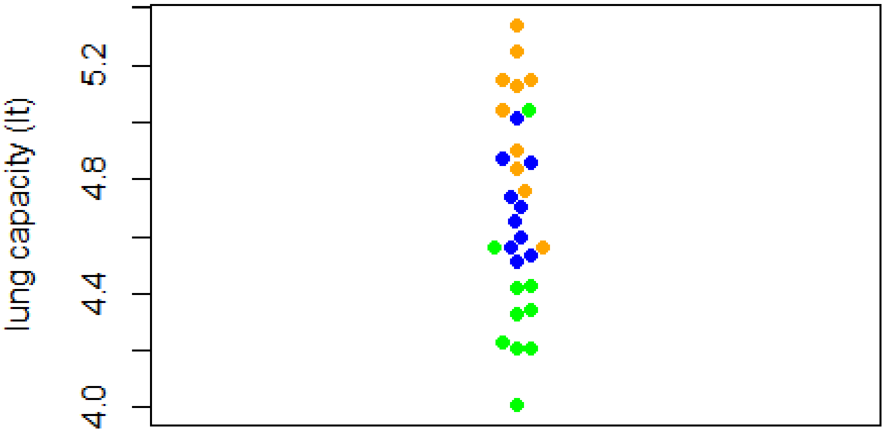
Session 1: Analysing differences between more than two groups

It’s all about variation

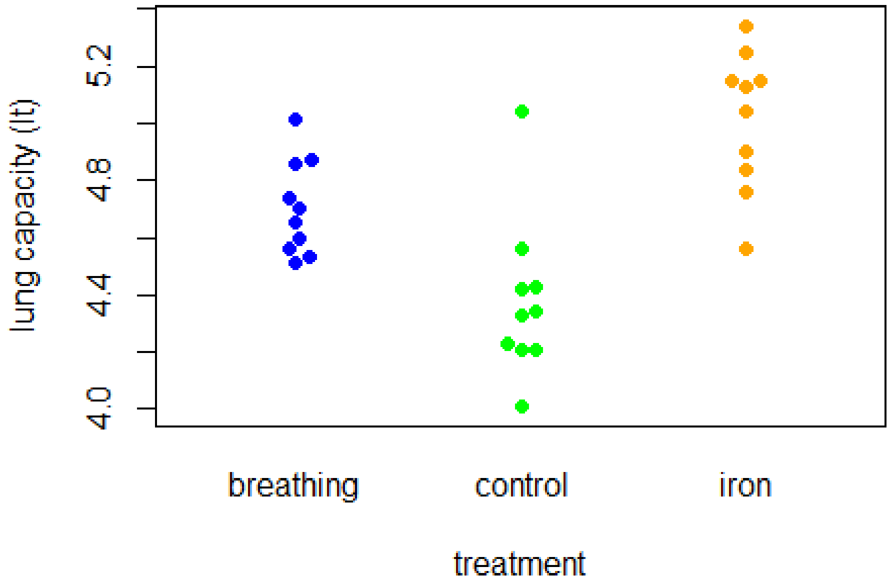
Statistics may seem like a scary topic, but at the end of the day, all that we are trying to do is to understand why things vary. If we take measurements of any response, our **dependent variable** or **DV** (such as length, weight, pH, number of species) from different replicate individuals or areas then there will be variation. Below is a plot of measurements of lung capacity (in litres) from a sample of male volunteers of similar age. We can see that there is quite a lot of variation between individuals.



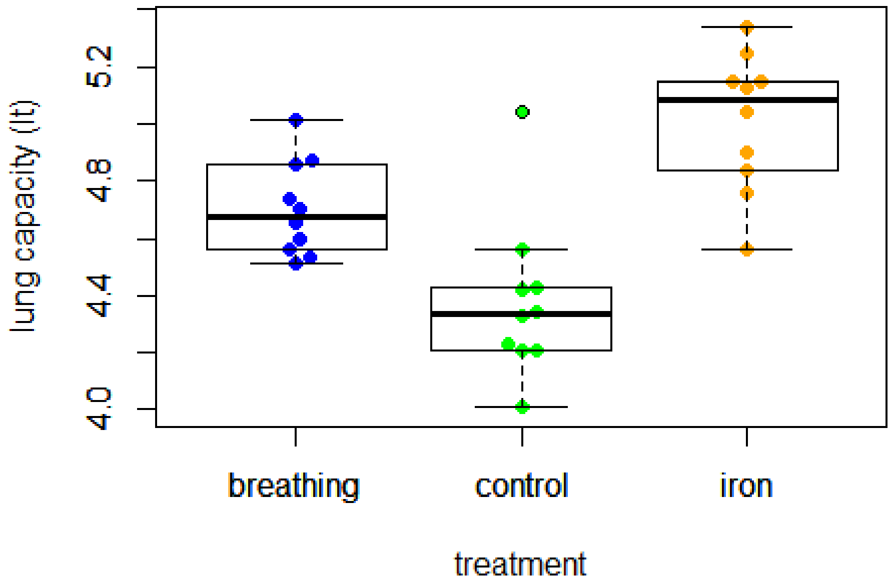
Our task it to try and understand what is causing, or influencing this variation. In order to do this, we try and determine any effect of our **predictor**, also known as the **independent variable** or **IV**. So in the above example, the data actually come from a study of the effect of different treatments (iron supplements, breathing exercises and a control group) on lung capacity. We could get an indication of any differences by colouring the points by treatment.



Based on the distribution of points of different colours, it certainly looks like there is an influence of treatment, but we do not know which is which. It would be more obvious if we separated them out into groups.



Seeing the individual points that make up each group helps to see how much variation there is within each group. We can combine this approach with the `boxplot` approach that you have used before to summarise variation in continuous variables between groups.



The example data shown above is a development of what we have done previously, when we went as far as looking at differences between two groups, through either the t-test or Wilcoxon rank-sum/signed-rank tests (*why would we use one or the other?*).

We can’t use these tests as we have three groups, so what next?

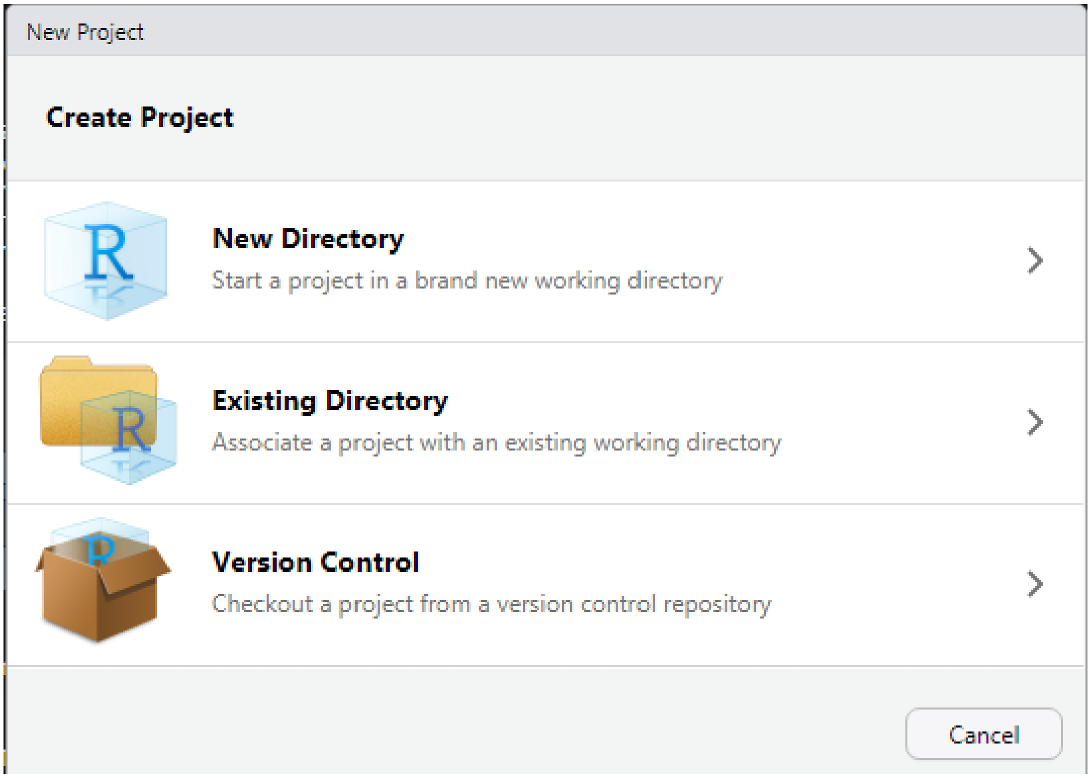
Going beyond two - ANOVA

The technique that we are going to use to determine whether there is a significant difference between more than two groups is known as **AN**alysis **O**f **VA**riance or **ANOVA** for short. As the name suggests, it analyses the variation in the DV, in this case in relation to a categorical IV or predictor (i.e. one that represents discrete groups that the data can be divided into).

Organisation

We will also use a new way of organising your R scripts, data etc. called ‘**Projects**’. These allow you to keep all the relevant files together and find them more easily. I would recommend that you have a project that contains all the information for the module; alternatively you can have anew project for each practical but doing it this way can make it hard to find things in the future unless you are very methodical in your naming and commenting in your code.

To start a new project, click on File > New Project. if you want to create the project in a new folder, choose the ‘New Directory’ option.



New Project

Create Project

New Directory

Start a project in a brand new working directory

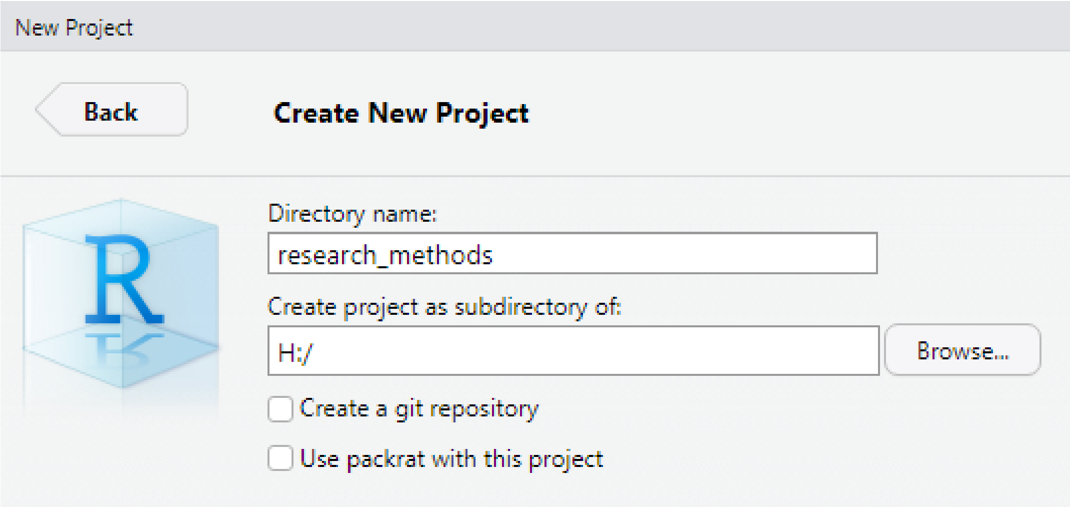
Existing Directory

Associate a project with an existing working directory

Version Control

Checkout a project from a version control repository

In the next screen choose ‘New Project’. Give the new folder a name, making sure that it is being created in the right place and then click Create Project.



Directory name: research_methods

Create project as subdirectory of: H:/

□ Create a git repository

□ Use packrat with this project

If you want to create the project in an existing folder then choose the ‘Existing Directory’ option and navigate to the folder you want to be working in.

This automatically sets the Working Directory to the specified folder, which helps avoid the annoying error messages that you get when you are not where you thought you were! Now that you have your project open, add a new R script file (File > New File > R Script or the button on the top left).

This script will be your library of code for this module. You can search it for specific terms etc. if you need to find things. In order to help yourself **MAKE SURE THAT YOU ADD COMMENTS TO REMIND YOU WANT YOU ARE DOING!!** It is recommended that you have a separate section in your script for each session. A really useful way to do this is to press *Ctrl+Shift+R*. Enter the name of the section that you want to create and then click on OK. It then creates a new section heading. At the bottom of the Script window you will see the name of the section that you are in. You can click on this and quickly navigate to any named section.

On Moodle, find and download the `lung.csv` file from Moodle and save it into the folder that you just created for your project. Once you have done this, you should see it in the **Files** list on the lower right-hand side.

In your script file, type the following to load the `beeswarm package and also to load the data file into an object called `lung`.

```r
# load the beeswarm library (for plotting values)
library(beeswarm)
# load the data file into the 'lung ' object
lung <- read.csv("lung.csv", stringsAsFactors = TRUE)
# check out the structure of the data
str(lung)
```

Run the commands that you have typed in.

Fancy figures

Beeswarm plots are a way of plotting the data that allow you to see all the variation in individual values (and hence can look a bit like a swarm of bees), unlike a boxplot or something similar, which summarises the variation in the values in a particular group. Plotting using `beeswarm` is much the same as plotting with the previous commands that you have used. To plot a continuous variable by a categorical (or grouping) variable you would write something along the lines of:

```
beeswarm(continuous~categorical, data = data)
```

We can also add in other details, such as axis labels (`xlab` and `ylab`), symbol types (`pch`) and colours (`col`), just as previously. So to recreate the last two plots that are shown above, try the following:

```
# first just the beeswarm plot
beeswarm(lung_capacity~treatment, data=lung, xlab="treatment", ylab="lung
capacity (lt)", pch=19, col=c("Blue", "Green", "Orange")
```

To add a boxplot over the top of this, we can use the `boxplot` command that we have used before, but add it to the previous plot using the add=TRUE command, and give it a colour that is transparent, so that you can see the points below.

```
beeswarm(lung_capacity~treatment, data=lung, xlab="treatment", ylab="lung
capacity (lt)", pch=19, col=c("Blue", "Green", "Orange"))
boxplot(lung$lung_capacity~lung$treatment, col="#00000000", add=TRUE)
```

Nice work! So we have made some fancy graphs of the variation in lung capacity between different treatment types, but now we need to follow up with some actual analysis of the differences.

lm - your new best friend!

ANOVA is one of a series of techniques that fall under a general heading of **linear models**. These are genrally performed using the `lm` command, or developments of this, so we will be doing a lot of `lm` in the coming sessions! In the case we are analysing, the analysis would be called a **one-way ANOVA** as we are only grouping data into one set of treatments. We will go into more complex option in a future session!

ANOVA is determining whether there is a significant difference between the **mean** values of each group. It makes some assumptions about the data, just like a t-test (which assumes normally distributed data). We need to check these as part of the analysis, but in this case we do it *after* the analysis has been performed.

The commands that you need to use are very similar to those for a t-test. We are looking for differences in lung capacity in relation to the treatment used. As for most of what you do in R, you want to store the outcome in an object, so you would write this as follows:

```
# do the analysis
# call it what you want - l1 is just a shorthand for lung 1
# just call it something that you can remember what it is!
l1 <- lm(lung_capacity ~ treatment, data=lung)
```

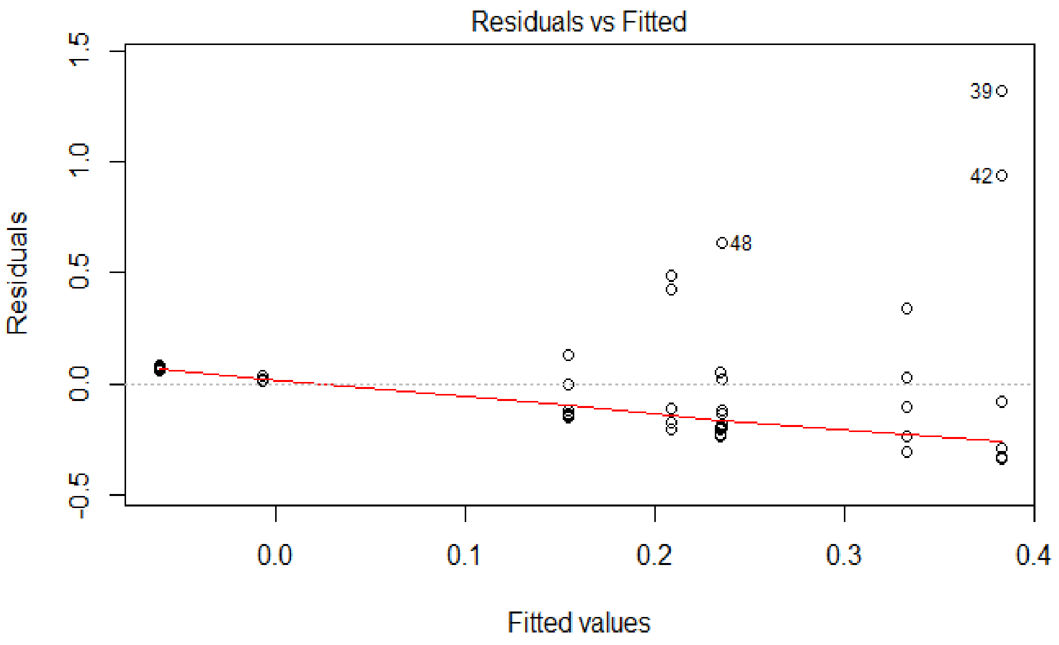
The results are stored in the `l1` object, so you don’t see anything for now.

Checking those assumptions

First, ANOVA assumes that the variance (i.e. the variability of the values in each group) is similar in each of the groups. We can assess this visually by plotting what is known as a fits vs residuals plot.

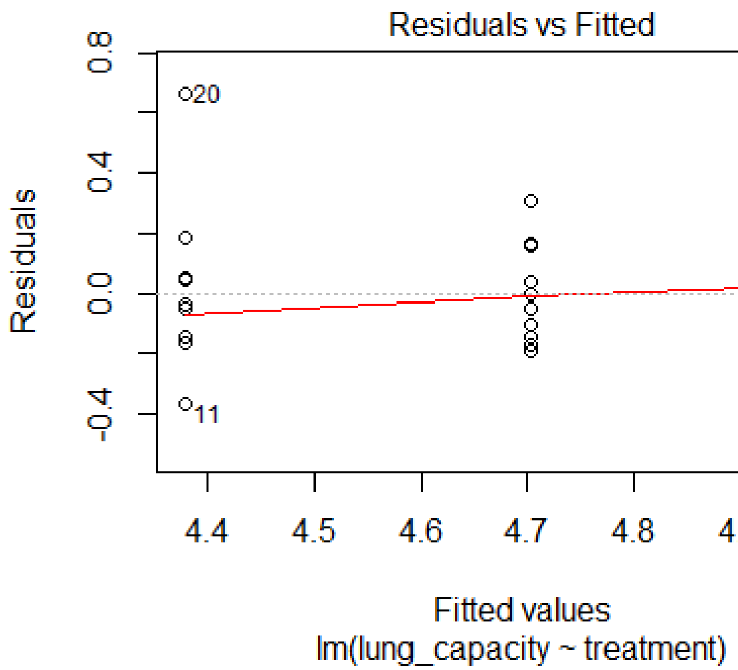
This plot shows how the difference between the observed and predicted values (the residuals) varies as the fitted values (i.e. the predicted responses) increase. How the points are distributed on the graph gives you information on whether the variances of the groups are equal.

If there is no consistent pattern of change in the values as you move from the left to the right of the plot then you are OK to assume that the variances are similar. If however the values form a definite pattern, generally either increasing or decreasing along the x axis, although you can get more complex patterns , then you have a problem. The most common problem by far is that the variance tends to increase with the fitted values i.e. larger values of the response also are more variable.



To get the required plot use:

```
plot(l1, which = 1)
```



The plot of the residuals vs the fitted values looks OK - no consistent change in the spread of values above and below the ‘0’ from the left to the right.

We can also check that the variances are equal more formally, through **Levene’s test** as follows:

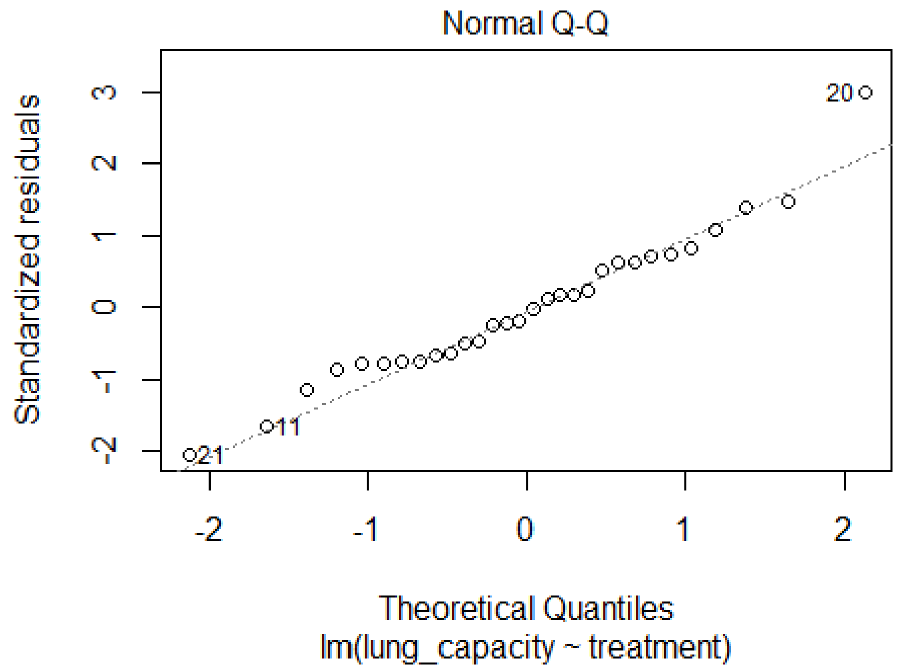
```
# first we want to load the 'car' library as well
library(car)

## Warning: package 'car' was built under R version 3.6.3

## Loading required package: carData

# the command tests the variance in each group of lung capacity
leveneTest(lung_capacity ~ treatment, data=lung)

## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  2  0.3863 0.6833
##       27
```

The important bit of the output is the `Pr(>F)` part, which is the probability value. Here the p-value is > 0.05, which indicates that the difference in variance between the groups is not significant. **This is what you want - in this case a non-significant p-value is good!**

That is the first assumption checked. The second assumption is to do with normality, but in this case in terms of the normality of what are known as the **residual** values rather than the actual data, which is what we have checked previously.

Residual values are essentially the variation in the data that is not related to the treatment groups. There is another plot that we can use for this.

```
plot(l1, which=2)
```



The residuals should fall along the diagonal line. They look reasonably normal but we can also check this more formally using the Shapiro-Wilk test that we used in a similar way for the t-test.

```
shapiro.test(l1$residuals)

##
##  Shapiro-Wilk normality test
##
## data:  l1$residuals
## W = 0.96486, p-value = 0.4096
```

Again, the key value here is the p-value. As for the equality of variance test, what you want is a non-significant p-value i.e. p > 0.05). 0.4096 is much larger than 0.05, so we can be satisifed that the residuals are normal and we are safe interpreting the outcome of the ANOVA.

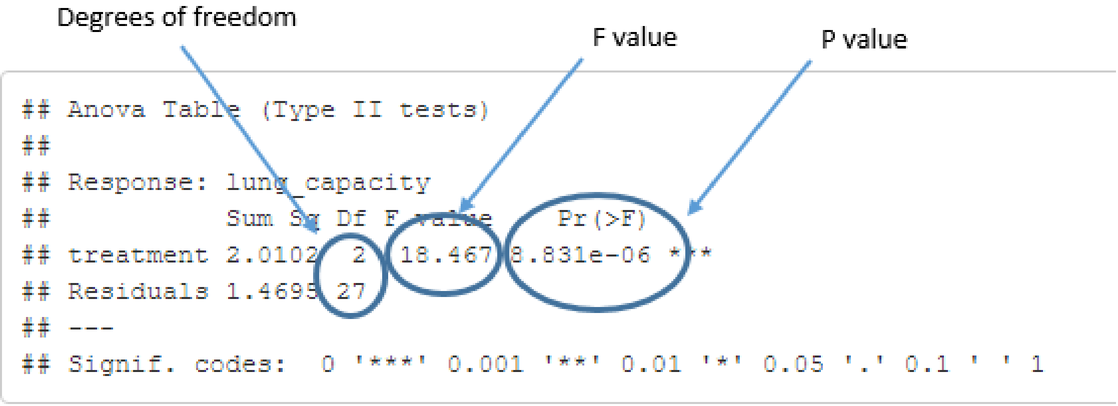
So what is the result?

Assumptions have now been checked, so in order to determine whether there is a significant difference between the mean values we do the following:

```
# NOTE THE CAPITAL 'A' BELOW
# also this needs the 'car' package to be loaded, which we did previously
# but remember to do this
Anova(l1)
```

```
## Anova Table (Type II tests)
##
## Response: lung_capacity
##           Sum Sq Df F value    Pr(>F)
## treatment 2.0102  2  18.467 8.831e-06 ***
## Residuals 1.4695 27
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This is what is known as an ANOVA table. You need to find your way around this to extract and interpret the different parts as shown below.



```
## Anova Table (Type II tests)
##
## Response: lung_capacity
##           Sum Sq Df F value    Pr(>F)
## treatment 2.0102  2  18.467 8.831e-06 ***
## Residuals 1.4695 27
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p value indicates whether there is a significant difference between the mean values of each group. The value here is shown in scientific notation i.e. it is 8.31 x 10^{-6} (0.00000831). This is indicative of a highly significant difference.

In order to summarise the outcome in the results section of a report etc. you also need to include the other information. So this would be as follows in this case:

"There was a highly significant difference in the mean lung capacity between the treatment groups (one-way ANOVA, F = 18.47, df = 2, 27, $p < 0.001$)."

In general if the p-value is less than 0.05 and greater than 0.0001 then you would give the actual value. Smaller values than this can just be given as $p < 0.0001$.

But where are the differences?

The ANOVA tells us that there is a significant difference between the mean values for the treatments. if we were just comparing two treatments, then we could easily say which was significantly higher or lower. As there are three here (and you can have more than three) we need to determine which are significantly different. This is done using what is known as a **post-hoc test**. There are a large number of these, but we will use the **Tukey test** (also known as Tukey's Honestly Significant Difference test).

This is performed as follows:

```
TukeyHSD(aov(l1))

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = l1)
##
## $treatment
##                     diff         lwr         upr     p adj
## control-breathing -0.325 -0.58368547 -0.06631453 0.0116575
## iron-breathing     0.309  0.05031453  0.56768547 0.0168090
## iron-control       0.634  0.37531453  0.89268547 0.0000051
```

The table of values shows you the significance of the difference between each pair of means being compared. So the first line compares the `control` group with the `breathing exercises` group. The significance of the difference between the means is 0.0012 (rounded up), so these are significantly different.

In this case all means are significantly different from each other. If any of the p-values are > 0.05 then that pair of means are not significantly different.

Less than normal

ANOVA works if you have continuous data values with equal variances in each group and normally distributed residuals. This is not always the case; sometimes (more often than we would like!) we have data that do not meet the assumptions, or are not continuous (e.g. rank data like scores from 1-10). In these cases we need an alternative.

That alternative is known as the Kruskal-Wallis test. This the non-parametric equivalent of the one-way ANOVA, which can be used if the assumptions of ANOVA are not met.

Download the `mtt_assay.csv` file from Moodle and import it into the `mtt` object. The data are derived from a study of effects of different organophosphate pesticides on the activity of human T-cells, as measured by the MTT assay. The data represent absorbance of cultures measured at 450nm following exposure for one week.

The data have similar variances in the different groups (run a Levene's test if you want), but the residuals are not normally distributed (again you can run the analysis usin `lm` if you want to see). Therefore we need to do use a Kruskal-Wallis test instead. The code is as follows:

```
kruskal.test(absorbance~pesticide, data=mtt)

##
##  Kruskal-Wallis rank sum test
##
## data:  absorbance by pesticide
## Kruskal-Wallis chi-squared = 8.9515, df = 2, p-value = 0.01138
```

The output indicates that there is a significant difference, this time in the **MEDIAN** values, rather than the means, as this is what the test is looking for differences in.

As for the ANOVA, we have established that there is a significant difference in the effect of the pesticides considered, but we do not know where the differences are. We need a post-hoc test. In this case, the test is known as Dunn's test. We can load the packages and run the test as follows:

```
library(dunn.test)
dunn.test(mtt$absorbance, mtt$pesticide, kw=FALSE, method="bh")

##
##                             Comparison of x by group
##                               (Benjamini-Hochberg)
## Col Mean-|
## Row Mean |   Chloropy   Glyphosa
## ---------+----------------------
## Glyphosa |  -2.442474
##          |    0.0109*
##          |
## Pendimet |  -2.717682  -0.275208
##          |    0.0099*     0.3916
##
## alpha = 0.05
## Reject Ho if p <= alpha/2
```

The layout is slightly different to the Tukey test for ANOVA, but the same information is there. For each combination of medians, the important information is the p-value, which is the **lower** of each pair of values. So in this case there is no significant difference in the median values for Glyphosate and Pendimethalin (p = 0.3916) but for the other combinations there are significant differences.

If you want to gain more plotting experience, then you could produce similar plots to those for the lung data.

Make sure that you have written sufficient comments in your script to be able to go back to it in future sessions - you will need to be able to interpret your commands for other examples and for the tests in due course.
