**Starting R and opening a dataset**

**New R script**

* Remove previous variables: rm(list=ls())
* Set the working directory:  setwd(“file\_pathway”)
* Loading a dataset, labelling it ‘COPD’:  COPD <- read.csv("COPD\_student\_dataset.csv")

**Install a new package**

* install.packages(“Package name”)
* library(Package name)

**Viewing the dataset**

* Look at the whole dataset: View(COPD)
* Print the first few rows of your dataset: head(COPD)
* See how many rows and columns you have in your dataset: dim(COPD)
* Look at the different variables in the dataset: colnames(COPD)
* Look at all the values in a variable: print(variable)
* To visualise the structure of the data in a variable: str(variable)
* Look at a specific value (x) in a variable: variable[x]

**For continuous variables:**

* View number of values, missing values, mean and ranges using the describe() function from the ‘Hmisc’ package.

**For categorical variables:**

* View number of values, missing values, mean and ranges using the describe() function from the ‘Hmisc’ package OR Tabulate the data to view the number of values and their frequency using the CrossTable() function from the ‘gmodels package. To view missing values, type: sum(is.na(variable)).
* Viewing the categories and distribution of entries in a categorical variable: table(catvariable)
* You can add the argument exclude = NULL in the function parentheses to include missing values in the output.

**Running a linear regression**

The basic format is:

* modelname <- lm(outcome~predictor, data = dataframe)
* Viewing the regression model output: summary(modelname)
* Viewing the model 95% confidence intervals: confint(modelname)

**Drawing a Q-Q plot, constant variance plot, and other diagnostic plots**

* Calculate predicted values: predict(modelname)
* Calculate residuals: residuals(modelname)
* Set a plotting format of 4 graphs: par(mfrow=c(2,2))
* View the 4 resulting plots: plot(modelname)

**Create a histogram**

* The basic format is: hist(variablename)
* If you are getting a variable from the dataset, the $ sign allows R to locate this variable. E.g. COPD$MWT1Best
* To change the title of the histogram, use the command: main = “histogram title”
* Don’t forget quotation marks when using text!
* To change the x or y axes labels, use the commands: xlab = “x axis label” or ylab = “y axis label”
* Don’t forget quotation marks using text!
* To change the number of bins displayed, use the command main = to specify the number of bins you want to see.
* To look at specific values in your variable, you can use the subset() function, using the basic code subset(dataframe, variable > 15) if you want to see values over 15 for that variable. You can add additive rules by including ‘|’, e.g. subset(dataframe, variable1 > 15 | variable2 < 5)

**Summary statistics**

* Basic summary statistics (incl. minimum, medium, maximum, 1st and 3rd quartiles, and number of blank cells): summary(variablename)
* List of summary statistics, including the basic summary() outcome, standard deviation, range, and inter-quartile range:
* list(summary(variablename), sd(variablename, na.rm = TRUE), range(variablename, na.rm = TRUE), IQR(variablename, na.rm = TRUE))
* Note that the na.rm = TRUE command tells R to remove NA values. Without this, an error message will be displayed.

**Correlation**

* Scatterplot of two variables: plot(x, y)
* Correlation coefficient: cor(x, y)
* The default method is Pearson, but you can change this to Spearman by adding method = “spearman” in your parentheses. You need to remove missing values, otherwise you will get an error message. To do this, add use = “complete.obs” in your parentheses.
* Correlation test: cor.test(x, y)
* The default method is also Pearson here. You also need to remove missing values to avoid an error message.

**Creating a correlation matrix:**

* Create a vector with the variables to include in the matrix,  e.g. data <- COPD[,c(“AGE”, “PackHistory”, “FEV1”)]
* Create the correlation matrix vector, assigning correlation coefficients of the different variables to it,  e.g. cor\_matrix <- cor(data)
* View the matrix: e.g. cor\_matrix to view the output and round(cor\_matrix,2) to round this output to 2 decimal points.
* Visualising correlation between variables, i.e. correlation plot: pairs(~ variable1 + variable2 + variable3, data = dataframe)

**Multiple linear regression**

The basic format is:

* modelname <- lm(outcome~predictor1 +  predictor2, data = dataframe)
* Viewing the regression model output: summary(modelname)
* Viewing the model 95% confidence intervals: confint(modelname)
* Examining the VIF using the imcdiagF() function from the ‘mctest’ package.

**Regression with categorical variables**

2 ways to do this:

* Check what the variable is saved as, change it to a factor variable if it is not saved as such.
* Check what the variable has been saved as using the class() function
* If it is not saved as a factor, can it using the factor() command, in the following format: variable <- factor(variable)
* Run the regression as normal
* Include factor() before the variable in the regression model. E.g. modelname <- lm(outcome~predictor1 + factor(predictor2),  data = dataframe)

**Changing the reference category of a variable:**

* Use the relevel() function in the following format: variable <- relevel(variable, ref = newreflevel) with the newreflevel  being the new reference level, written either as a numeric (1, 2, 3, …) or a character (in which case it needs to be written within apostrophes – “MILD”, “SEVERE”, …)

**Changing data type for a variable**

* Check what the variable has been as using the class() function.
* Changing data type:
* To numeric: as.numeric()
* To character: as.character()
* To factor: factor() or as.factor()
* To integer: as.integer()

**Creating a new variable on R: e.g. variable ‘comorbid’**

comorbid is a variable you were asked to create. This variable was to be binary, and indicated the presence of at least one comorbidity (‘1’) or complete absence of comorbidities (‘0’) based on the responses to the variables: Diabetes, muscular, hypertension, AtrialFib, and IHD.

* Check that all variables are saved as the correct datatype.
* Create an empty vector of the correct length.
* Here, comorbid will be the same length as the other variables, so:
* comorbid <- length(COPD$Diabetes)
* Assign values to this vector.
* Here, we want comorbid = 1 when Diabetes OR muscular OR hypertension OR AtrialFib OR IHD = 1. So: comorbid[COPD$Diabetes == 1 | COPD$muscular == 1 | COPD$hypertension == 1 | COPD$AtrialFib == 1 | COPD$IHD == 1] <- 1
* This will assign 1 to the values meeting the set conditions, and NAs to those that are not meeting those conditions.
* We also want comorbid = 0 when ALL above variables = 0. So: comorbid[is.na(cormorbid)] <- 0
* Convert this variable to a factor.
* Optional: add the variable to the dataset, using the following command COPD$comorbid <- cormorbid

**Regression with interaction effect**

* Use the same format as a multiple linear regression, but include both terms, i.e.
* modelname <- lm(outcome~predictor1 +  predictor2 + (predictor1 \* predictor2),
* data = dataframe)
* Interpretation of the interaction effect can be simplified using the prediction() function from the ‘prediction’ package.