Correlation and regression

14/11/2023

Null Hypothesis Significance Testing (NHST)

Think back to our previous questions:

1. Do men and women differ in terms of their fear of crime?

2. Are people who have been a victim of crime more fearful of crime?

The basis of NHST is to phrase these questions as:

If there is only one population, how likely is it that our two samples have values this different from each other?

Performing *t*-tests in R

The tilde (~) symbol in R usually means "modelled by"

`FoC ~ victim_crime` means `FoC modelled by victim_crime`.

`data = crime` tells R to look in the `crime` data frame for the data.

`paired = FALSE` tells R that this is an *independent samples* test.

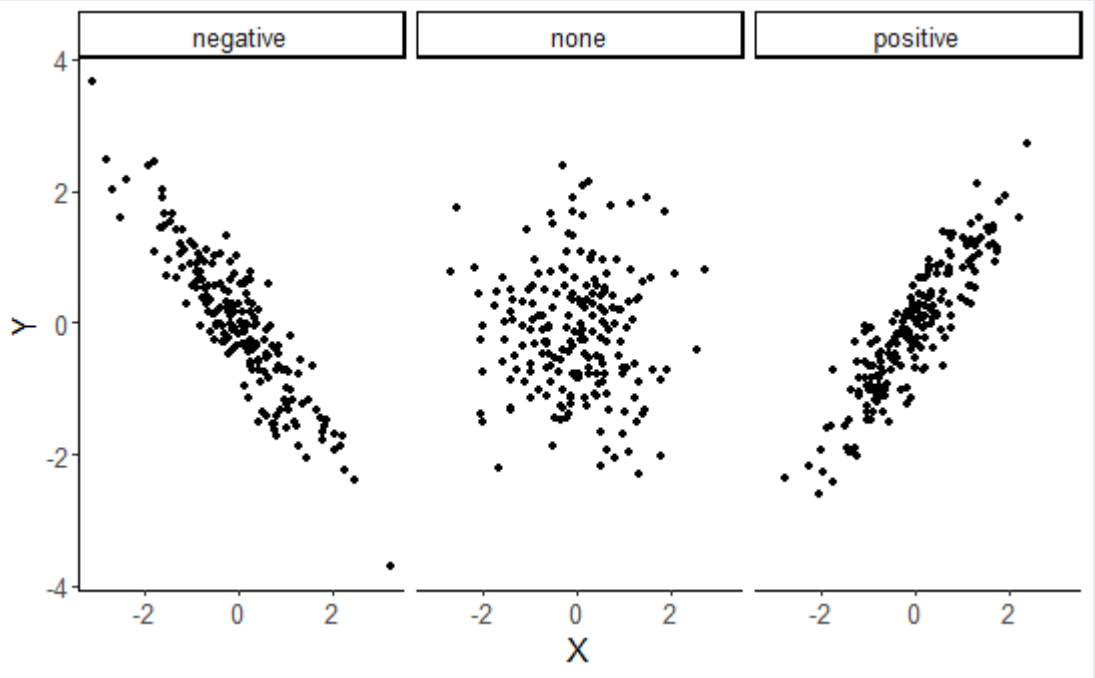
```
t.test(FoC ~ victim_crime,
       data = crime,
       paired = FALSE)
```

```
##
##     Welch Two Sample t-test
##
## data:  FoC by victim_crime
## t = 0.45309, df = 197.48, p-value = 0.651
## alternative hypothesis: true difference in means
## 95 percent confidence interval:
##  -0.1873001  0.2990388
## sample estimates:
##  mean in group no mean in group yes
##          2.463636           2.407767
```

Correlation and statistical relationships

Correlation

Correlation measures the strength and direction of an association between two continuous variables.



Correlation

How related are the variables in the Fear of Crime dataset?

```
head(crime)
```

```
## # A tibble: 6 × 15
##   Particip…¹ sex     age victi…²     H     E     X     A     C     O    SA    TA
##   <chr>      <chr> <dbl> <chr>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 R_01TjXgC… male     55 yes       3.7   3     3.4   3.9   3.2   3.6  1.15  1.55
## 2 R_0dN5YeU… fema…    20 no        2.5   3.1   2.5   2.4   2.2   3.1  2.05  2.95
## 3 R_0DPiPYW… male     57 yes       2.6   3.1   3.3   3.1   4.3   2.8  2     2.6
## 4 R_0f7bSsH… male     19 no        3.5   1.8   3.3   3.4   2.1   2.7  1.55  2.1
## 5 R_0rov2Ro… fema…    20 no        3.3   3.4   3.9   3.2   2.8   3.9  1.3   1.8
## 6 R_0wioqGE… fema…    20 no        2.6   2.6   3     2.6   2.9   3.4  2.55  1.5
## # … with 3 more variables: OHQ <dbl>, FoC <dbl>, Foc2 <dbl>, and abbreviated
## #   variable names ¹Participant, ²victim_crime
```

Correlation

Let's look at the relationship between Emotionality (*E*) and Fear of Crime (*FoC*).

```
ggplot(crime,
       aes(x = E,
           y = FoC)) +
  geom_jitter() +
  theme_classic(base_size = 20) +
  labs(x = "Emotionality",
       y = "Fear of Crime")
```

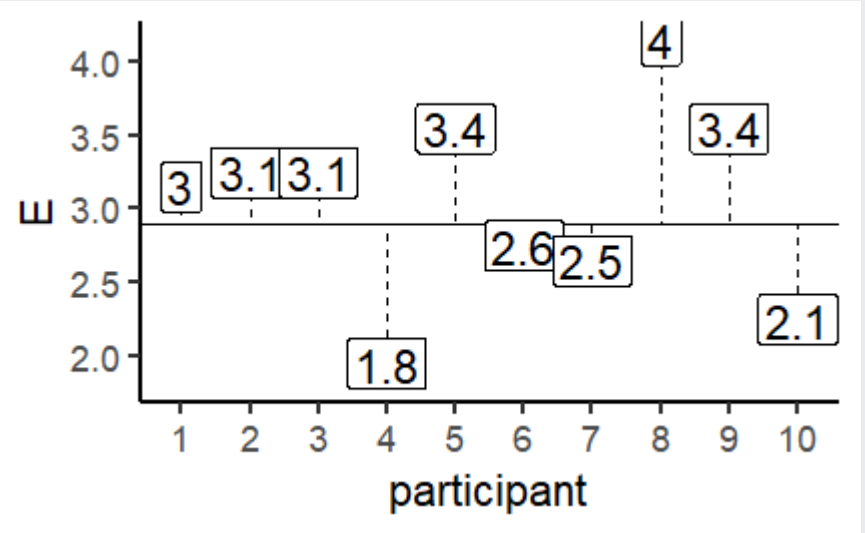
Correlation and covariance

How do we quantify the relationship between these variables?

We need to look at how much they *vary together*.

The plot shows the Emotionality values of the first ten participants.

The line across the middle is the mean of those values - **2.9**.



The mean and the variance

As you can see, the values don't lie directly on the mean, but are spread around it.

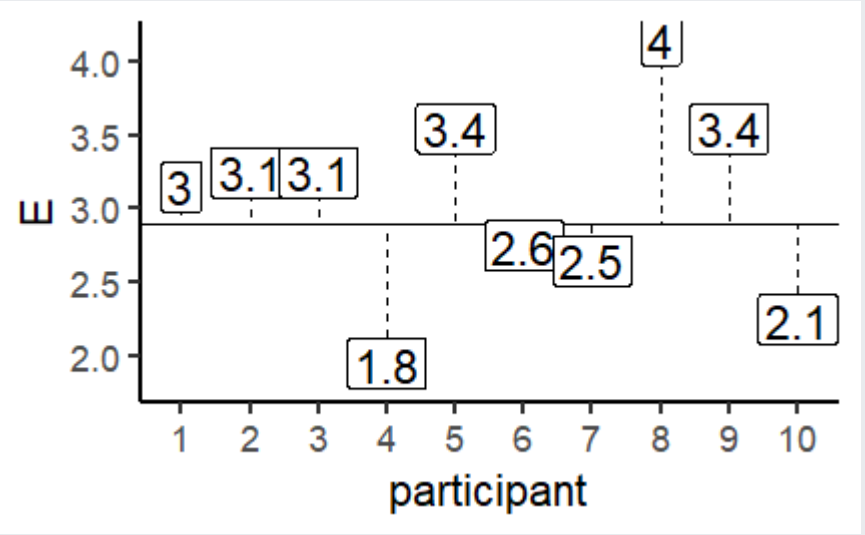
To quantify how much the values vary from the mean, we can calculate the *variance*.

Here's the scary looking formula for the variance:

$$\sigma^2 = \frac{\sum(x - \bar{x})^2}{N - 1}$$

And here's the not-so-scary R function:

```
var(x)
```

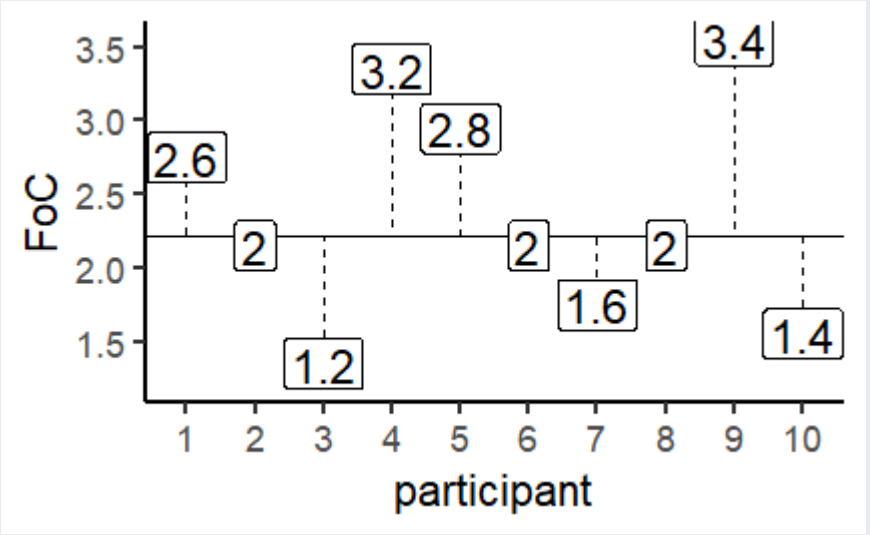


Correlation and covariance

Now let's look at the same plot for Fear of Crime (FoC).

Again, these points and labels are individual ratings of Fear of Crime.

The line across the middle shows the mean, which is **2.22**.



Correlation and covariance

Now let's look at these previous two plots as differences from their respective means.

What we want to now is to what extent the values *vary together*. I.e. as one goes up, does the other?

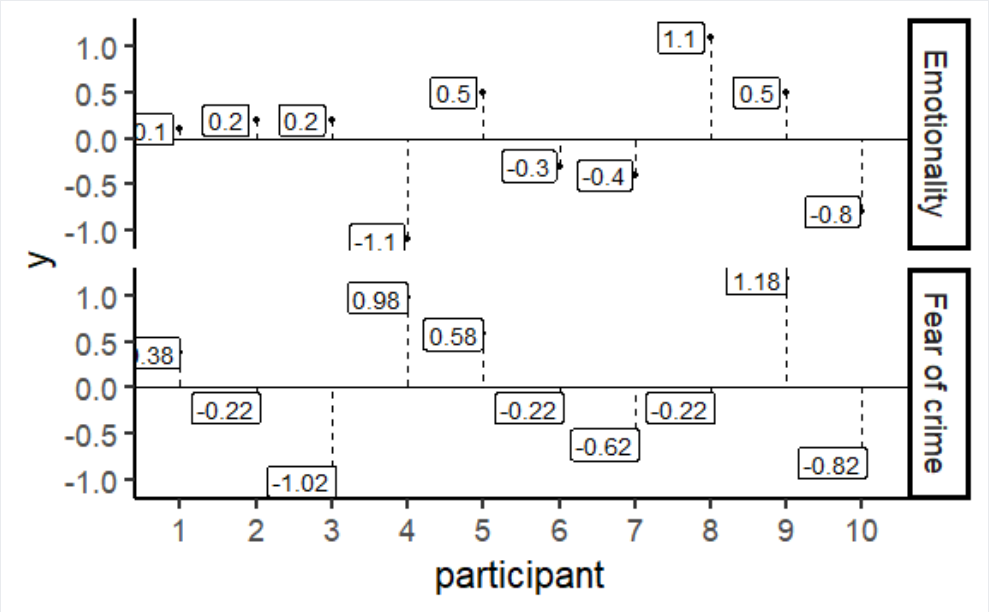
This is *covariance*.

Here's the scary formula:

$$cov(x, y) = \frac{\sum((x - \bar{x})(y - \bar{y}))}{N - 1}$$

Here's the not-so-scary R function:

```
cov(x, y)
```



Correlation and covariance

Covariance gives us a measure of how much two variables vary together.

But the numbers it gives us can be hard to interpret when the variables are on very different scales.

So we rescale the covariance using the standard deviations of each variable.

$$corr(x, y) = r = \frac{cov(x, y)}{\sigma^x \sigma^y}$$

This gives us the *correlation coefficient*, or *r*.

```
cor(crime$E, crime$FoC)
```

```
## [1] 0.369891
```

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

Pearson's product-moment correlation

The **cor.test()** function can be used to test the *significance* of a correlation.

```
cor.test(crime$E, crime$FoC,
         method = "pearson")
```

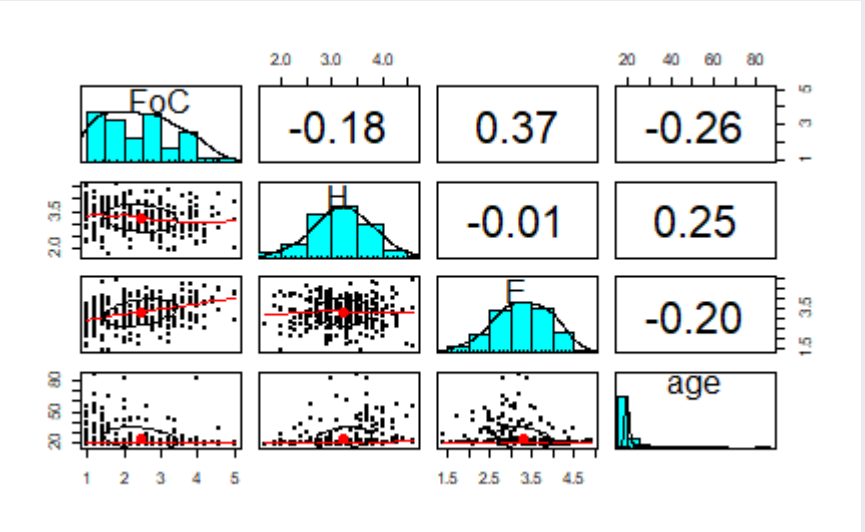
```
##
##     Pearson's product-moment correlation
##
## data:  crime$E and crime$FoC
## t = 6.8843, df = 299, p-value = 3.421e-11
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.2680476 0.4635586
## sample estimates:
##      cor
## 0.369891
```

Running multiple correlation using the psych package

Unfortunately, `cor.test()` won't work on multiple variables at once!

A nice way to run multiple correlations at once, and get significance values, is using the `pairs.panels()` function from the `psych` package.

```
crime_corr <- select(crime, FoC, H, E, age)
```



Running multiple correlation using the apaTables package

You can use 'apa.cor.table' function from the apaTables package to create a nicely formatted APA table showing the means, standard deviation, and correlation with confidence intervals.

```
crime_corr2 <- select(crime, FoC, H, E, X,
```

```
apa.cor.table (crime_corr2, filename = "Cor
```

Table 1

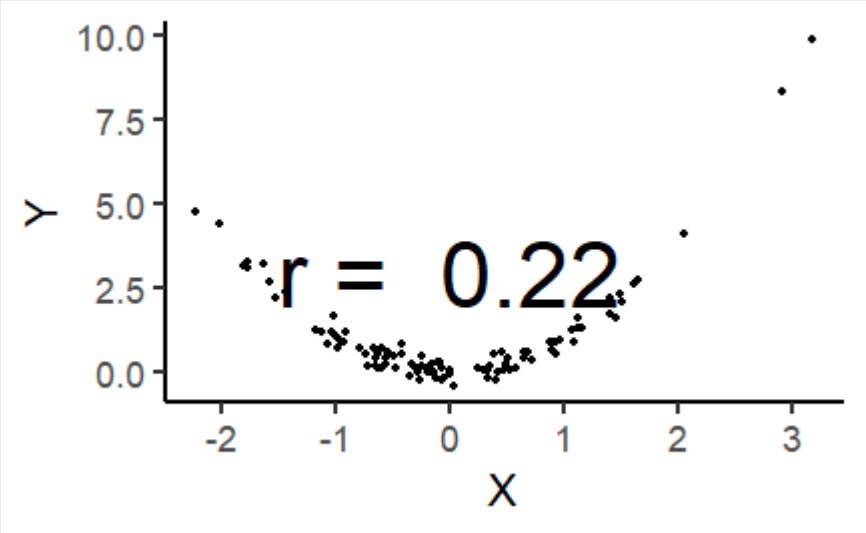
Means, standard deviations, and correlations with confidence intervals

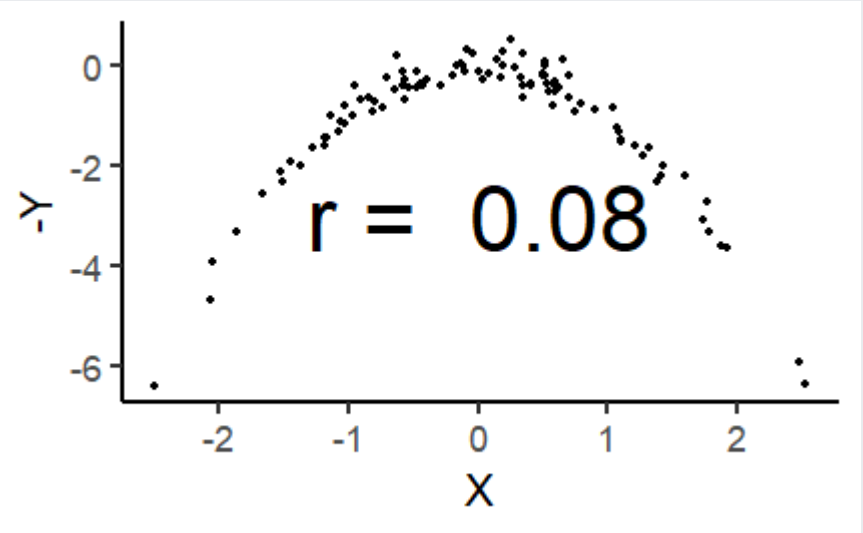
Variable	*M*	*SD*	1	2	3	4	5
1. H	3.22	0.56					
2. E	3.28	0.67	-.01 [-.12, .11]				
3. X	3.15	0.61	.24** [.13, .34]	-.09 [-.20, .02]			
4. A	3.10	0.68	.33** [.22, .42]	.12* [.01, .23]	.26** [.15, .36]		
5. C	3.47	0.59	.17** [.06, .28]	.23** [.12, .34]	.27** [.16, .37]	.09 [-.02, .20]	
6. O	3.42	0.62	.11 [-.00, .22]	-.09 [-.20, .02]	.03 [-.09, .14]	.03 [-.09, .14]	.13* [.02, .24]

Note. *M* and *SD* are used to represent mean and standard deviation, respectively. Values in square brackets indicate the 95% confidence interval for each correlation. The confidence interval is a plausible range of population correlations that could have caused the sample correlation (Cumming, 2014). * indicates $p < .05$. ** indicates $p < .01$.

Curved or non-linear relationships

If your data look like this:

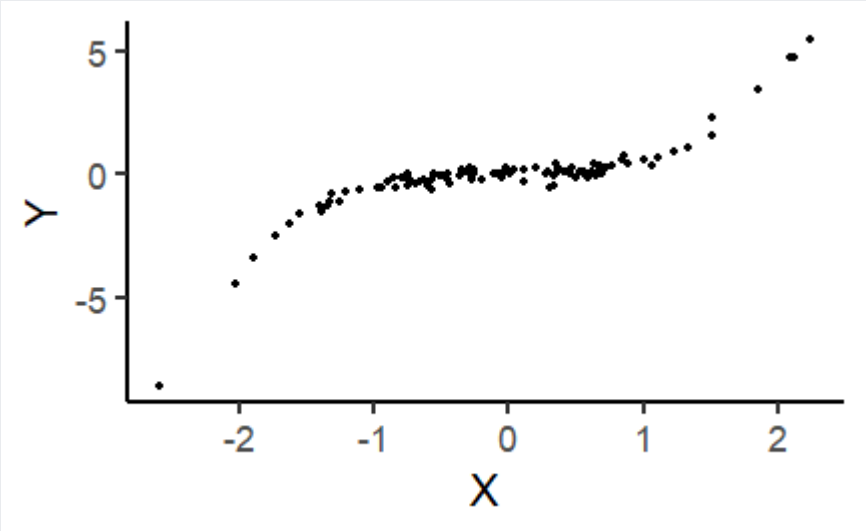




...forget about correlation.

Curved or non-linear relationships

...but if your data look like this:



...there is some hope!

Spearman's rank correlation is used to measure *monotonicity*, and is the non-parametric equivalent to Pearson's correlation.

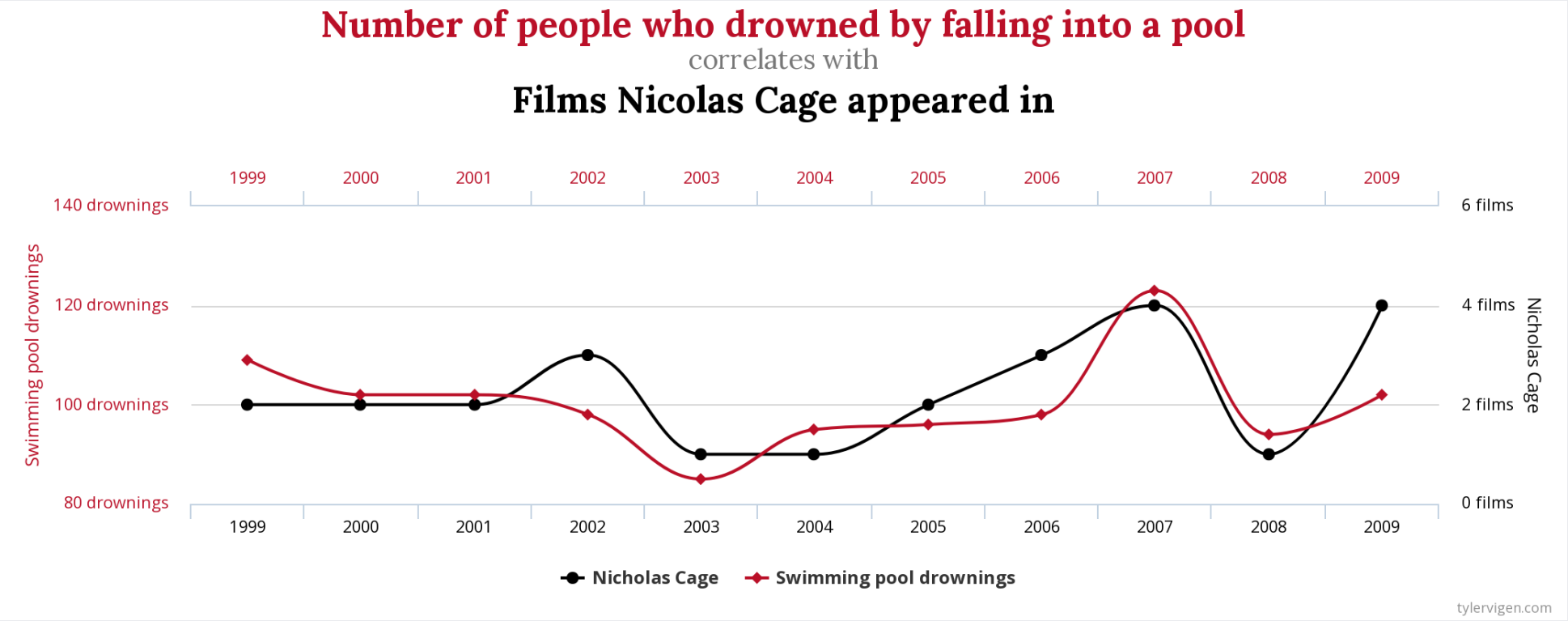
The data is converted to ranks, and then correlated.

```
cor.test(X, Y,
         method = "spearman")
```

```
##
##     Spearman's rank correlation rho
##
## data:  X and Y
## S = 22610, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to
## sample estimates:
##       rho
## 0.8643264
```

Correlation is not causation

https://www.spuriouscorrelations.com



Reporting a correlation

Reporting a correlation is pretty straightforward. Typically, the degrees of freedom or number of observations should also be given, e.g. $r(299) = .37$, $p < .001$, or $r = .37$, $p < .001$, $N = 301$.

With degrees of freedom in bracket, "There was a significant weak positive correlation between emotionality and fear of crime, $r(299) = .37$, $p < .001$." OR With number of observations written out, "There was a significant weak positive correlation between emotionality and fear of crime, $r = .37$, $p < .001$, $N = 301$."

It is best to also specify which type of correlation you used (e.g. Pearson's or Spearman's); the strength of the relationship (i.e., weak or strong) and a scatterplot showing the relationship should almost always be shown.

Note that *r* is considered a measure of *effect size*. An *r* of .1 is considered a small effect, while an *r* of .8 is considered a large effect.

Linear regression

Correlation, regression and prediction

Correlation quantifies the *strength* and *direction* of an association between two continuous variables.

But what if we want to *predict* the values of one variable from those of another?

For example, as Emotionality increases, *how much* does Fear of Crime change?

```
ggplot(crime,
       aes(x = E, y = FoC)) +
  geom_jitter() +
  stat_smooth(method = "lm", se = FALSE) +
  theme_classic(base_size = 22) +
  labs(x = "Emotionality",
       y = "Fear of crime")
```

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

Linear regression

The line added to this scatterplot is the *line of best fit*.

It's the straight line that gets closest to going through all of the points on the plot.

But how do we work out where the line should be?

The line of best fit

The line represents the predicted value of **y** at each value of **x**.

The prediction is made using the following formula:

$y = a + bX$

a is the *intercept* - the point where the line would cross the y-axis when the value of the x-axis is 0.

y is the dependent or outcome variable we are predicting.

b is the *slope* - the steepness and direction of the line.

The *line of best fit* can be found by adjusting the *intercept* and *slope* to minimise the *sum of squared residuals*.

Line of best fit demo

Fear of crime predicted by emotionality

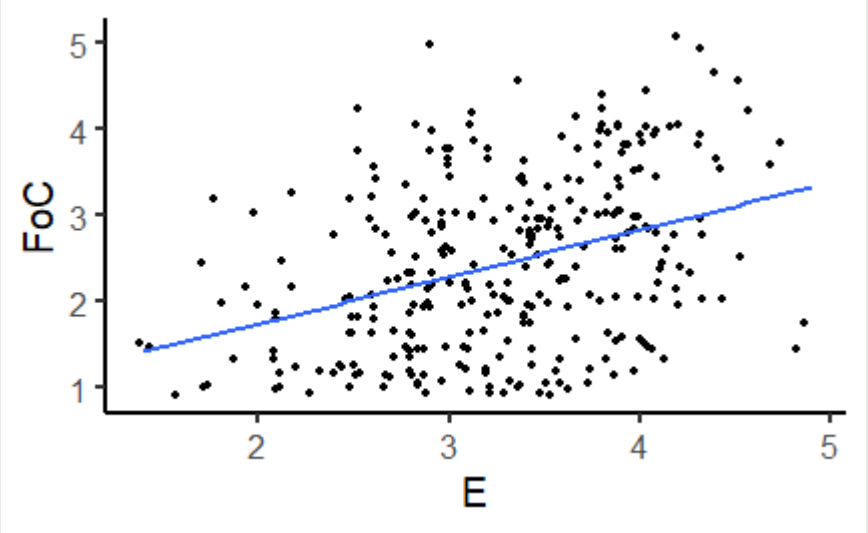
Let's try using the `lm()` function to predict Fear of Crime (*FoC*) from Emotionality (*E*).

```
foc_by_E <- lm(FoC ~ E, data = crime)
foc_by_E
```

```
##
## Call:
## lm(formula = FoC ~ E, data = crime)
##
## Coefficients:
## (Intercept)            E
##      0.6492       0.5475
```

These are the *intercept* and *slope* of the regression line on the right.

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



Is this a good model of Fear of crime?

```
summary(foc_by_E)
```

```
##
## Call:
## lm(formula = FoC ~ E, data = crime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.87698 -0.72952 -0.03902  0.70844  2.76319
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.64918    0.26621   2.439   0.0153 *
## E            0.54746    0.07952   6.884 3.42e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9278 on 299 degrees of freedom
## Multiple R-squared:  0.1368,    Adjusted R-squared:  0.1339
## F-statistic: 47.39 on 1 and 299 DF,  p-value: 3.421e-11
```

Fear of crime predicted by emotionality

Let's focus on the coefficients.

```
summary(foc_by_E)$coefficients
```

```
##               Estimate Std. Error  t value     Pr(>|t|)
## (Intercept) 0.6491774 0.26621482 2.438547 1.532835e-02
## E           0.5474598 0.07952319 6.884279 3.421376e-11
```

Estimate is the *coefficient* of each predictor; Std. Error is an estimate of the accuracy of that coefficient.

The significance of each predictor is tested using a t-test; *t value* is the t statistic, and the *Pr(>|t|)* column is the p-value.

Thus, *Emotionality* is a significant predictor of *Fear of Crime*.

Since its coefficient (0.547) is positive, Fear of Crime increases as Emotionality increases.

Fear of crime predicted by emotionality

Again, the regression line is described by the formula $y = a + bX$. So we can fill that out with our model coefficients as follows:

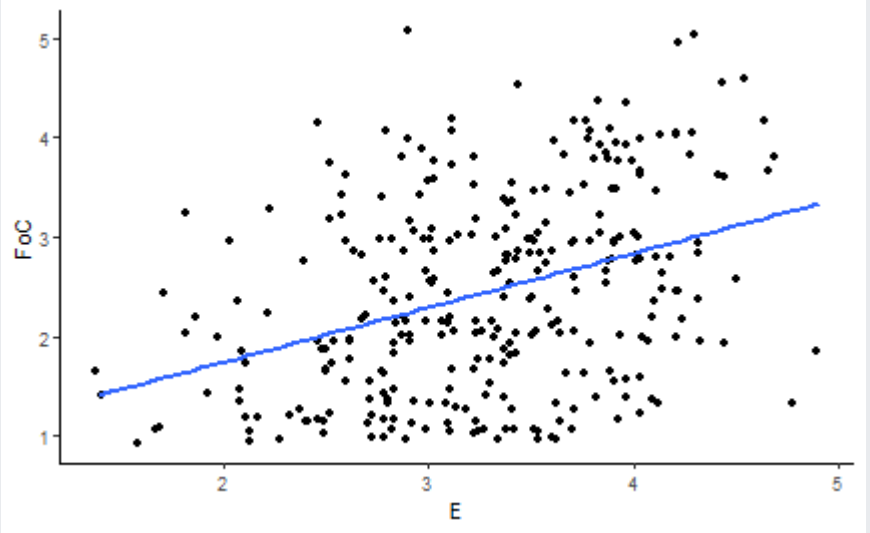
Fear of crime = 0.65 + 0.55 * X

X is the value of the *predictor*.

The *intercept* is now the value of y when the value of the predictor is *zero*.

The coefficient for the predictor is the amount that y increases for each 1 unit increase in the predictor.

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



Assessing model significance

Is this a good model?

```
summary(foc_by_E)
```

```
##
## Call:
## lm(formula = FoC ~ E, data = crime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.87698 -0.72952 -0.03902  0.70844  2.76319
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.64918    0.26621   2.439   0.0153 *
## E            0.54746    0.07952   6.884 3.42e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9278 on 299 degrees of freedom
## Multiple R-squared:  0.1368,    Adjusted R-squared:  0.1339
## F-statistic: 47.39 on 1 and 299 DF,  p-value: 3.421e-11
```

The mean as a model

First, let's create a linear model that simply finds the *mean* using the `lm()` function.

```
intercept_only <- lm(FoC ~ 1, data = crime)
intercept_only
```

```
##
## Call:
## lm(formula = FoC ~ 1, data = crime)
##
## Coefficients:
## (Intercept)
##       2.445
```

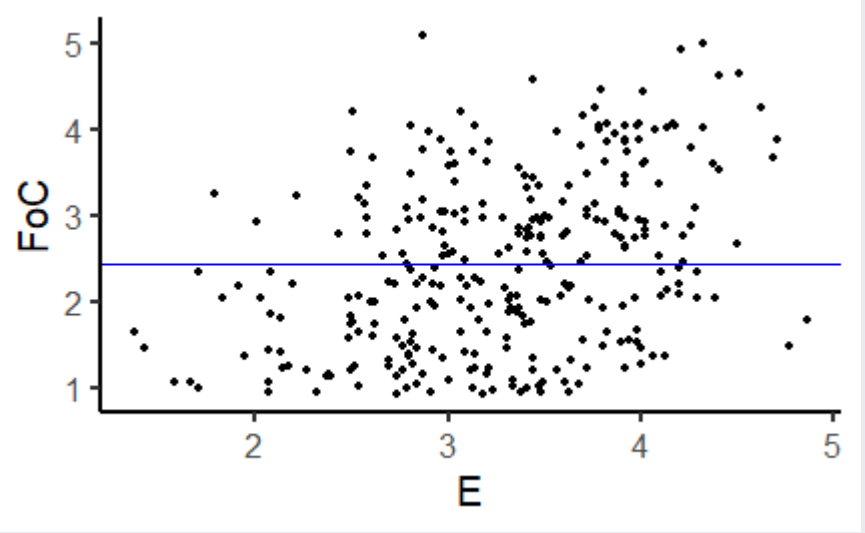
Here the Intercept is equal to the *mean* of FoC.

```
mean(crime$FoC)
```

```
## [1] 2.444518
```

In the formula $y = a + bX$, *a* is the *Intercept*.

So our prediction for the value of *y* is $y = 2.44$.



The mean as a model

```
summary(intercept_only)
```

```
##
## Call:
## lm(formula = FoC ~ 1, data = crime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.44452 -0.84452 -0.04452  0.55548  2.55548
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.44452    0.05746   42.54   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9969 on 300 degrees of freedom
```

Model comparison

We can compare models using the `anova()` function.

```
anova(intercept_only, foc_by_E)
```

```
## Analysis of Variance Table
##
## Model 1: FoC ~ 1
## Model 2: FoC ~ E
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    300 298.16
## 2    299 257.37  1    40.795 47.393 3.421e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(foc_by_E)$fstatistic
```

```
##    value    numdf    dendf
##  47.3933   1.0000 299.0000
```

At least it's better than the mean!

```
summary(foc_by_E)
```

```
##
## Call:
## lm(formula = FoC ~ E, data = crime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.87698 -0.72952 -0.03902  0.70844  2.76319
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.64918    0.26621   2.439   0.0153 *
## E            0.54746    0.07952   6.884 3.42e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9278 on 299 degrees of freedom
## Multiple R-squared:  0.1368,    Adjusted R-squared:  0.1339
## F-statistic: 47.39 on 1 and 299 DF,  p-value: 3.421e-11
```

Assessing model fit

How much Y does X explain?

```
summary(foc_by_E)
```

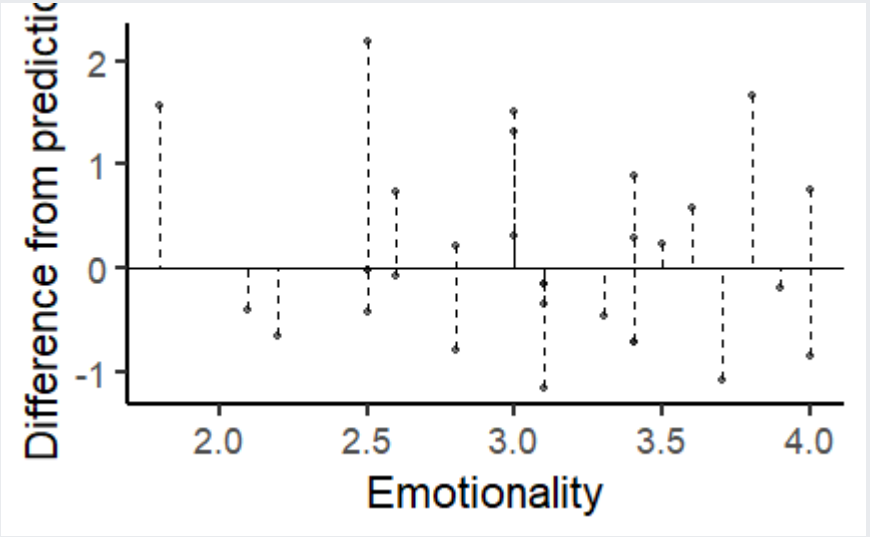
```
##
## Call:
## lm(formula = FoC ~ E, data = crime)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.87698 -0.72952 -0.03902  0.70844  2.76319
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.64918    0.26621   2.439   0.0153 *
## E            0.54746    0.07952   6.884 3.42e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9278 on 299 degrees of freedom
## Multiple R-squared:  0.1368,    Adjusted R-squared:  0.1339
## F-statistic: 47.39 on 1 and 299 DF,  p-value: 3.421e-11
```

Model fit

R-squared (R^2) is a measure of model fit. Specifically, it's the proportion of *explained* variance in the data.

We previously looked at the *variance* around the *mean*.

After linear regression, we look at how much reality differs from the model predictions - the *residual error*.

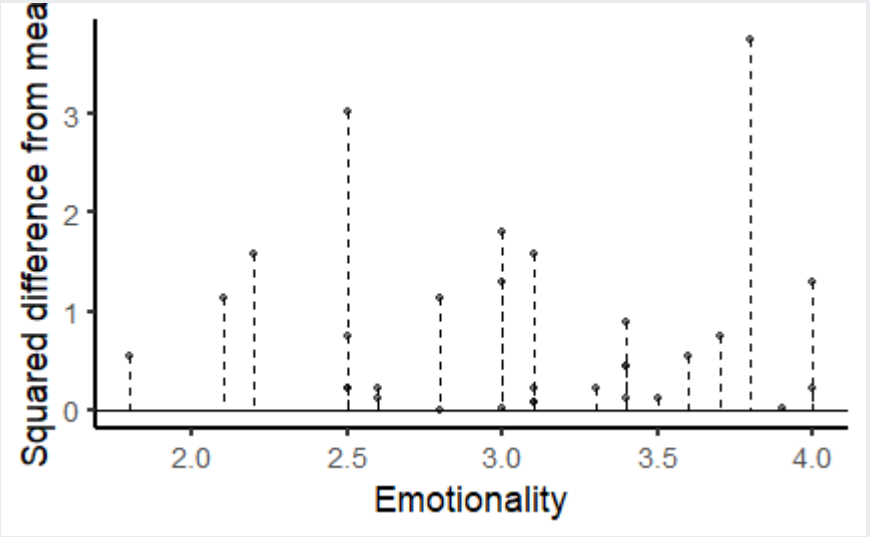


Model fit

To work out how well our model fits, we first need to know how much *total variation* there is in the data.

For that, we sum the squared differences of the values of the dependent variable y from the mean of the dependent variable $\bar{y}$ - the *total sum of squares*, SS_t:

$$SS_t = \sum (y - \bar{y})^2$$



Squared differences

Why square the differences?

1. Negative values become positive.
2. Values that are further away from the mean often get even further away.

This prevents "errors" from cancelling out, and effectively penalises values that are far away from the mean.

Model fit

We then calculate the sum of the squared differences of the values of the dependent variable (y) from the model predictions - the sum of the squared residuals, SS_r:

$$SS_r = \sum (y - \hat{y})^2$$

Model fit

Finally, we calculate *model sum of squares* - SS_m - as the difference between the *total sum of squares* and the *residual sum of squares*. This tells us, roughly, how much better our model is than just using the *mean*:

$$SS_m = SS_t - SS_r$$

R-squared (R^2) can then be calculated by dividing the model sum of squares by the total sum of squares:

$$R^2 = \frac{SS_m}{SS_t}$$

This yields the *percentage of variance explained by the model*.

This is a long-winded way of saying: Higher R^2 means more explained variance, and thus, a better fitting model.

Model fit

Thankfully, R does all these calculations for us!

```
summary(foc_by_E)$r.squared
```

```
## [1] 0.1368193
```

Our simple regression model of the effect of Emotionality on Fear of Crime explained ~ 14% of the variance.

What's left?

1. Other variables?

2. Measurement error?

Reporting simple regression

Example of reporting a simple regression model

"Simple linear regression was used to investigate the relationship between emotionality and fear of crime. A significant regression equation was found that explained 14% of the variance, R^2 = .14, *F*(1, 299) = 47.39, *p* < .001. Fear of crime increased significantly with increases in Emotionality, b = 0.55, *t*(299) = 6.884, *p* < .001."

Nicely formatted tables using sjPlot

```
library(sjPlot)
tab_model(foc_by_E,
          show.std = TRUE,
          title = "Table 1: Linear regression model",
          pred.labels = c("Intercept", "Emotionality"),
          dv.labels = "Fear of Crime")
```

Table 1: Linear regression model

	Fear of Crime				
Predictors	*Estimates*	*std. Beta*	*CI*	*standardized CI*	*p*
Intercept	0.65	0.00	0.13 – 1.17	-0.11 – 0.11	**0.015**
Emotionality	0.55	0.37	0.39 – 0.70	0.26 – 0.48	**<0.001**
Observations	301				
R^2 / R^2 adjusted	0.137 / 0.134				

Next week

Next week we'll continue with **regression**, looking at multiple predictors.

We'll also begin with **one-way ANOVA** for comparison of multiple means.

Reading

Chapter 10 - Comparing Several Means - ANOVA (GLM 1)

Additional support

Maths & Stats Help (AKA MASH) are a service offered by the University, based over in the library.

They offer support to both undergraduate and postgraduate students. You'll find their website at

https://guides.library.lincoln.ac.uk/mash

Note that while their website is mostly about other software, they do support R!

Or join the MS Teams group, use the discussion board, or drop me an email!