Relationships and Regression

Jan Rovny

Relationships

One of the most interesting questions we have about phenomena out in the real world concerns their connections. Is A related to B? If one achieves higher level of education, what impact is it likely to have on her income?

Covariance and Correlation

The simplest way of assessing relationships is by considering how different variables change (or vary) together, that is to say, how do they covary or (in a mathematically related assessment) correlate.

Let's first consider a simple example of madeup data:

```r
a<-c(1,2,3,4,5,6,7,8,9,10) #vector *a* with 10 values
b<-c(10,7,5,4,6,3,4,2,3,1) #vector b with 10 values
```

You should be able to note that as a increases, b tends to decrease, though it is not 'perfect'. Can we describe the relationship between a and b, and assess its strength?

To do so, we can turn to two useful measures: covariance and correlation. These measures address the extent to which the values of two variables 'change together', in other words, to what extent they covary or correlate.

Covariance is measured like this:

$Cov(XY) = S_{(XY)} = \frac{\sum (x_i - \bar{X})(y_i - \bar{Y})}{N-1}$

Correlation is measured like this:

$Corr(XY) = r_{(XY)} = \frac{\sum (\frac{x_i - \bar{X}}{S_X})(\frac{y_i - \bar{Y}}{S_Y})}{N-1}$

Generally speaking, correlation is a preferred measure because it 'standardizes' each variable by dividing its variance by its standard deviation. Consequently, correlation ranges from -1 to 1, where -1 means that the two variables are perfectly negatively correlated, 1 means that the variables are perfectly (positively) correlated, and 0 means that there is no association between the two variables.

In R we can assess this simply:

```r
cov(a,b)
```

```
## [1] -7.055556
```

```r
cor(a,b)
```

```
## [1] -0.8843145
```

Let us now turn to real data, looking at a data subset from the ESS dataset:

```r
library(rio)
ESS<-import("https://jan-rovny.squarespace.com/s/ESS_FR.dta")
```

Let's consider some relationships, for example one between education and support for European integration:

```r
cov(ESS$eduyrs,ESS$euftf, use="complete.obs")
```

```
## [1] 1.485939
```

```
cor(ESS$eduyrs,ESS$euftf, use="complete.obs")
```

```
## [1] 0.1451533
```

Regression

Let us now consider views on European integration as a function of one's education. We will turn to model views on the EU, which we will call y as a function of educaiton, which we will call x.

We will describe the relationship as $y_i = \alpha + \beta x_i + \epsilon$

Here α and β are regression coefficients, describing the relationship between x and y, and ϵ is a so-called *error term*, which captures all the deviations (or variance) that our model fails to explain.

R allows us to evaluate the values of α and β:

```
model<-lm(ESS$euftf~ESS$eduyrs)
summary(model)
```

```
##
## Call:
## lm(formula = ESS$euftf ~ ESS$eduyrs)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.6062 -1.8541  0.0485  1.9512  5.9245
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.68614    0.20945  17.599  < 2e-16 ***
## ESS$eduyrs   0.09733    0.01552   6.272 4.42e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.593 on 1828 degrees of freedom
##   (87 observations deleted due to missingness)
## Multiple R-squared:  0.02107,    Adjusted R-squared:  0.02053
## F-statistic: 39.34 on 1 and 1828 DF,  p-value: 4.425e-10
```

If you write in LaTeX, the following produces the results table in TeXcode, make it easy to produce pretty tables just by copy-pasting:

```
library(xtable)
xtable(model)
```

```
## % latex table generated in R 3.3.2 by xtable 1.8-2 package
## % Sat Jul  1 16:09:52 2017
## \begin{table}[ht]
## \centering
## \begin{tabular}{rrrrr}
##   \hline
##  & Estimate & Std. Error & t value & Pr($>$$|$t$|$) \\
##   \hline
## (Intercept) & 3.6861 & 0.2095 & 17.60 & 0.0000 \\
##   ESS\$eduyrs & 0.0973 & 0.0155 & 6.27 & 0.0000 \\
##    \hline
```

```
## \end{tabular}
## \end{table}
```

The helpful aspect of R's object-based language is that everything we do becomes an object. The above regression model has just become another object that we can call up and work with. For example:

```r
model$coefficients # vector containing all the coefficients of model
model$coefficients[1] #calls up the first item (intercept) of the coefficient vector
model$coefficients[2] #calls up the second item (the coefficient on ESS$relig) of the coefficient vecto
predict(model) #produces a vector of fitted values (yhat) for each observation
model$residuals # is a vector containing all of the residuals of the model
summary(model)$r.squared  # prduces a scalar with the R^2 of the model
vcov(model) #produces the variance-covariance matrix of the estimator
```

In a similar way, we can construct a matrix containing the key information from our model. We can then use the individual items in the matrix:

```r
stat.coef  <- summary(model)$coefficients #construct a matrix of model results
stat.coef[,1]    # 1st column: coefficients
```

```
## (Intercept)   ESS$eduyrs
##  3.68614470   0.09733358
```

```r
stat.coef[,2]     # 2nd column: se for each coef
```

```
## (Intercept)   ESS$eduyrs
##  0.20945280   0.01551757
```

```r
stat.coef[,3]     # 3rd column: t-value for each coef
```

```
## (Intercept)   ESS$eduyrs
##    17.598928    6.272476
```

```r
stat.coef[,4]     # 4th column: p-value for each coef
```

```
##  (Intercept)     ESS$eduyrs
## 3.628791e-64 4.424607e-10
```

Visualization

It is nice to visualize what we are doing. We can plot the values of x and y and draw a line that is mathematically expressed in the equation: $y_i = \alpha + \beta x_i$. We call this the regression line, or line of best fit.

```r
plot(ESS$eduyrs, ESS$euftf, main='Predicting EU support', ylab='EU support', xlab='Years of Education')
 #create a scatter plot with title and axis labels
abline(lm(ESS$euftf~ESS$eduyrs)) #draw the regression line
```

Predicting EU support
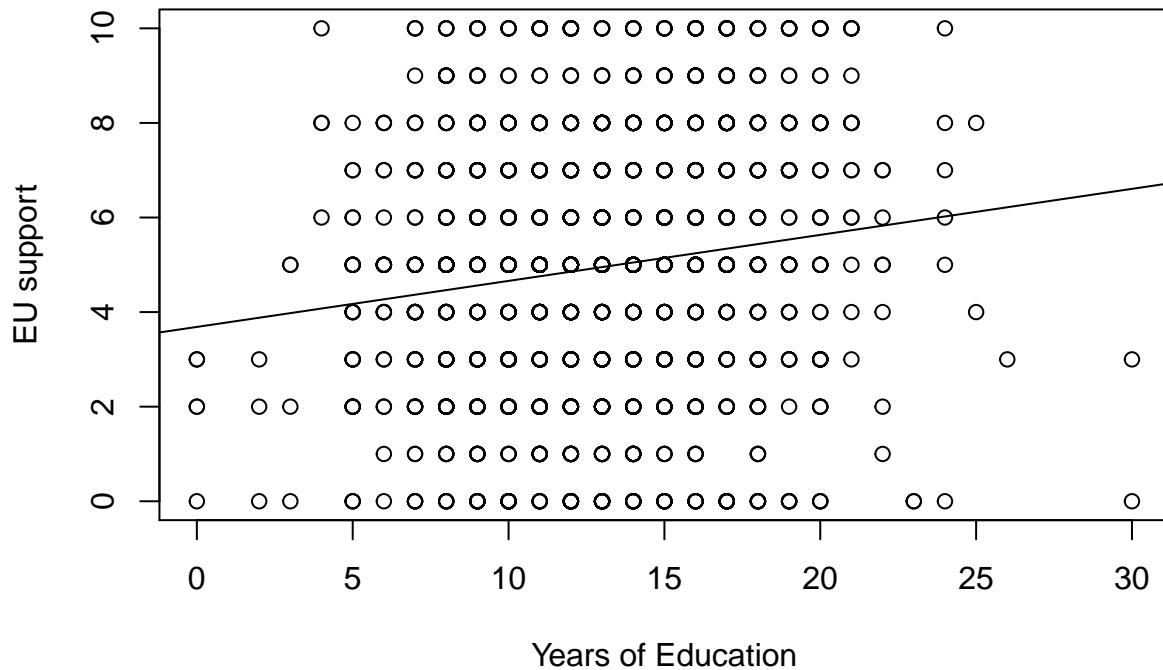
We may now wish to consider the level of EU support at higher levels (above the mean) of education:

```
summary(ESS$euftf[ESS$eduyrs>mean(ESS$eduyrs, na.rm=T)])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   3.000   5.000   5.232   7.000  10.000      34
```

In the next step, we may be interested in seeing the confidence intervals of our results:
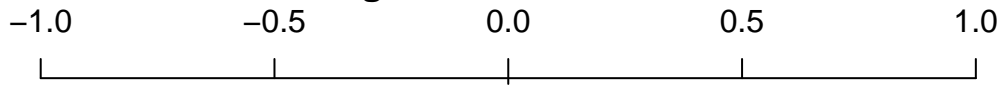
```
confint(model)
```

```
##                    2.5 %    97.5 %
## (Intercept) 3.27535276 4.0969366
## ESS$eduyrs  0.06689955 0.1277676
```

Finally, we may wish to create a so-called *coefficient plot* which shows the magnitude of the coefficients:

```
library(arm) # load library needed
```

```
coefplot(model, xlim=c(-1,1), intercept=F) #plots coefficients on range from -1 to 1, excluding the int
```

Regression Estimates

−1.0   −0.5   0.0   0.5   1.0

GS$eduyrs