HDS 0.2 Correlation and linear regression in practice

Matti Pirinen, University of Helsinki

28.12.2023

In this document, we study a relationship between two or more variables. First, we establish how to quantify the strength of a linear relationship between continuous variables and then we learn how to use the relationship to predict a value of an unknown variable given the values of the observed variables.

1. Correlation

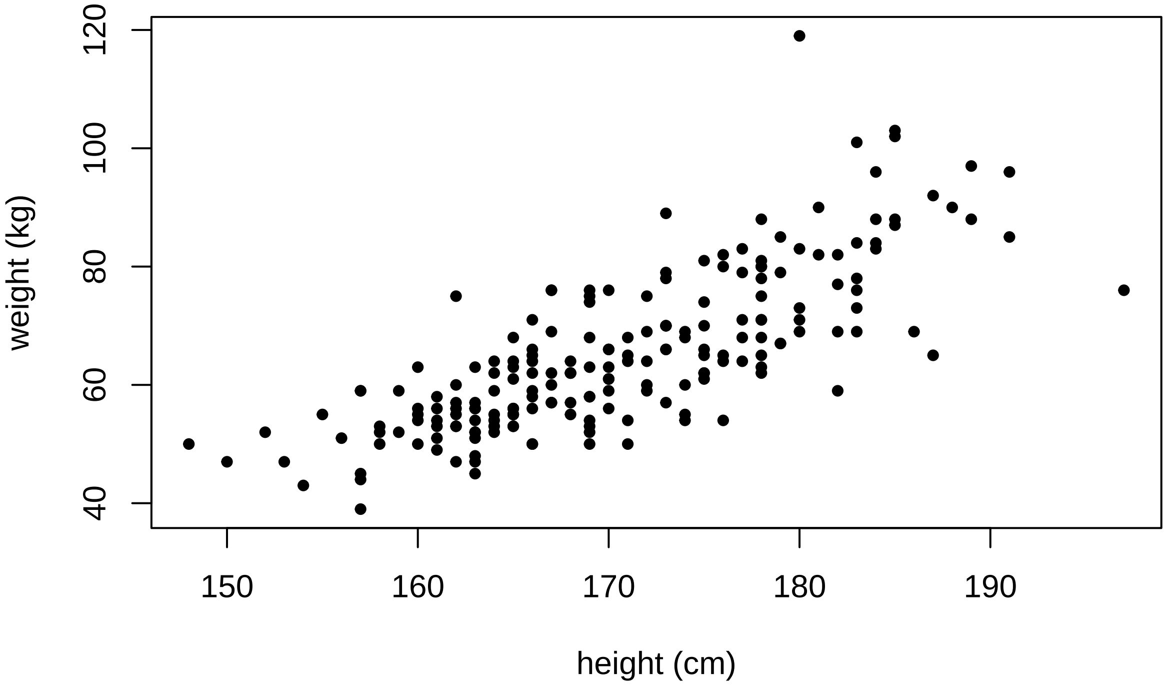
Let's read in a data set of heights and weights of 199 individuals (88 males and 111 females). This dataset originates from http://vincentarelbundock.github.io/Rdatasets/doc/carData/Davis.html.

```
y = read.table("https://www.mv.helsinki.fi/home/mjxpirin/HDS_course/material/Davis_height_weight.txt",
               as.is = TRUE, header = TRUE)
head(y)
```

```
##   X sex weight height repwt repht
## 1 1   M     77    182    77   180
## 2 2   F     58    161    51   159
## 3 3   F     53    161    54   158
## 4 4   M     68    177    70   175
## 5 5   F     59    157    59   155
## 6 6   M     76    170    76   165
```

The last two columns are self-reported values of weight and height. Let's plot weight against height.

```
plot(y$height, y$weight, pch = 20, ylab = "weight (kg)", xlab = "height (cm)")
```



Unsurprisingly, there is a clear pattern where taller individuals weigh more. To quantify the (linear part of the) relationship, we compute a Pearson's **correlation coefficient** between the variables. This happens in two parts: (1) compute the covariance between the variables and (2) scale the covariance by the variability of both variables to get a dimensionless correlation coefficient.

Covariance measures the amount of variation in the variables that is linearly shared between the variables. Technically, it is the expectation of the product of deviations from the means of the variables, and from a sample it is computed as

$$\mathrm{cov}(X,Y) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y}), \text{ where } \overline{x}, \overline{y} \text{ are the means of } x_i s \text{ and } y_i s, \text{resp.}$$

When both X and Y tend to be above their mean values simultaneously, then the covariance is positive, whereas the covariance is negative, if when X is above its mean, Y tends to be below its mean. Covariance of 0 says that there is no linear relationship between X and Y. Note that if we compute the covariance between the variable with itself, that is, $X = Y$ in the formula above, the result is simply the variance of that one variable. Thus, covariance is a generalization of the concept of variance to two variables.

Correlation coefficient results when covariance is normalized by the product of the standard deviations of the variables:

$$\mathrm{cor}(X,Y) = \frac{\mathrm{cov}(X,Y)}{\mathrm{SD}(X)\mathrm{SD}(Y)}.$$

Correlation is always between -1 and +1, and it denotes the strength of the linear relationship between the variables. If correlation is +1, then the values of X and Y are on a line that has a positive slope and if correlation is -1, then X and Y are on a line that has a negative slope. When correlation is 0, there is no linear association between the variables. (See Figures from Wikipedia.) Note that if correlation between X and Y is 1, it does not necessarily mean that $Y = X$, but only that there is a linear relationship of the form $Y = a + bX$ for some constants a and $b > 0$, and any such linear relationship leads to the correlation of 1.

Suppose we have an estimate $\widehat{r}$ of the correlation between X and Y, based on a sample of n observations ($n = 199$ in the example above). To do statistical inference with the correlation coefficient, we need to know the standard error of the estimate of the correlation. An approximation for Normally distributed data is:

$$\mathrm{SD}(\widehat{r}) \approx \sqrt{\frac{1-\widehat{r}^2}{n-2}}$$

This is accurate when correlation is not far from 0 (say $|r| < 0.5$), but becomes less accurate near ±1. For more accurate confidence intervals, one can use `r.con()` function from the `psych` package.

We expect a clear positive correlation between weight and height based on the Figure above. Let's see what it is.

```
n = nrow(y) #sample size
r = cor(y$height, y$weight) #correlation is symmetric cor(weight,height) = cor(height,weight)
se = sqrt((1-r^2)/(n-2))
library(psych) #DO FIRST: install.packages("psych")
ival = r.con(r, n, p = 0.95, twotailed = TRUE)
c(r = r, low95CI = ival[1], up95CI = ival[2],
  low95CI.norm.appr = r-1.96*se, up95CI.norm.appr = r+1.96*se)
```

```
##                 r           low95CI            up95CI low95CI.norm.appr
##         0.7707306         0.7074835         0.8217303         0.6817547
##  up95CI.norm.appr
##         0.8597065
```

In this case, `r.con()` gives 95CI as (0.707, 0.822) showing some difference from the less accurate normal approximation.

Exercises 1

1. Plot scatter plot of `weight` on x-axis and `repwt` on y-axis. Make a similar plot for `height` and `repht`. Do the reported values seem highly correlated with the measured values?
2. Compute correlation between `weight` and `repwt` (self-reported weight) and between `height` and `repht` (self-reported height). Note that you need to use `use = "complete.obs"` to get rid of NAs.
3. Compute a **correlation matrix** of the four variables `weight`, `height`, `repwt` and `repht` by giving `cor()` function those four columns of the data matrix as input. Note that you need to use `use = "pairwise"` in `cor()` to get rid of NA values.
4. Plot the curve $y = x^2$ in the range $x \in (-1, \dots, 1)$ using 1000 equally spaced values for x. Guess what is the correlation between x and y. Compute correlation.
5. Plot the curve $y = x^3$ in the range $x \in (-1, \dots, 1)$ using 1000 equally spaced values for x. Guess what is the correlation between x and y. Compute correlation.

2. Linear regression

Correlation coefficient r describes the strength of a linear relationship between two variables. Both variables are treated symmetrically in the definition of r. However, we may also want to utilize the linear relationship to predict the value of Y given that we know the value of X. For example, we may use our observed population above to make a statistical prediction of weights of individuals who are exactly 170 cm tall. From the Figure above we see that such individuals have weights roughly in range from 50 kg to 80 kg. To characterize the relationship more mathematically, we want to fit a line through the observed data that describes how Y depends on X. The **linear regression** model assumes that

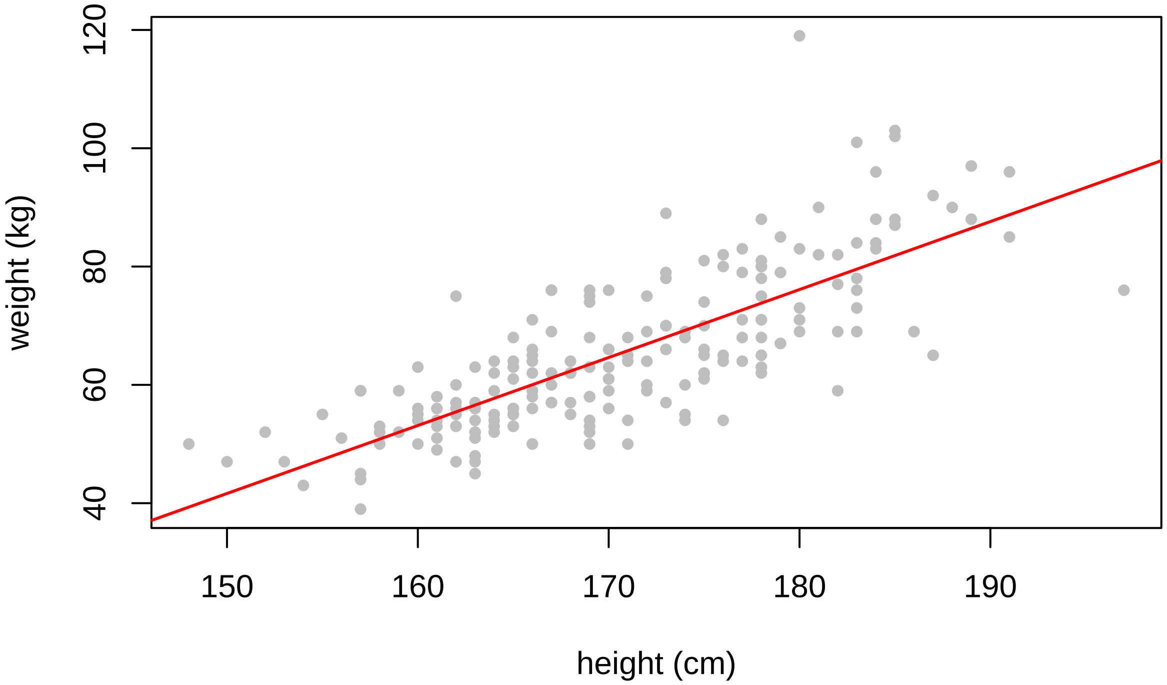
$$y_i = a + bx_i + \varepsilon_i,$$

where a (intercept, vakio) and b (slope, kulmakerroin) are model parameters that define the line and error term ε_i is the difference between the observed value y_i and the value predicted by the line $\varepsilon_i = y_i - (a + bx_i)$. Any pair of parameters a and b define one line (namely $y = a + bx$) in X-Y coordinates and among all those lines we will choose the one that fits best the observations as described next. We denote the estimates by $\widehat{a}$ and $\widehat{b}$ and the prediction for observation i by $\widehat{y_i} = \widehat{a} + \widehat{b}x_i$. The estimates are chosen by the least squares method that minimizes the residual sum of squares

$$\text{RSS} = \sum_{i=1}^{n}(y_i - \widehat{y_i})^2 = \sum_{i=1}^{n}(y_i - (\widehat{a} + \widehat{b}x_i))^2$$

In R, the linear model is estimated using `lm()` function where formula `y ~ x` says that "regress y on x". The regression line can be added to a figure by `abline()` function.

```
lm.1 = lm( weight ~ height, data = y)
plot(y$height, y$weight, pch = 20, ylab = "weight (kg)", xlab = "height (cm)", col="gray")
abline(lm.1, col="red", lwd=1.5)
```



```
summary(lm.1)
```

```
##
## Call:
## lm(formula = weight ~ height, data = y)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -19.650  -5.419  -0.576   4.857  42.887
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -130.74698   11.56271  -11.31   <2e-16 ***
## height         1.14922    0.06769   16.98   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 8.523 on 197 degrees of freedom
## Multiple R-squared:  0.594,  Adjusted R-squared:  0.592
## F-statistic: 288.3 on 1 and 197 DF,  p-value: < 2.2e-16
```

In output of the `summary`:

- Residuals are the observed vertical differences between the line and the observed weight value
- Coefficients show the least squares parameter estimates with their SEs and P-values (here highly significantly different from 0 with P-values < 2e-16). The slope tells that each cm in height corresponds to an average increase of 1.15 kg in weight.
- Residual standard error is an estimate for standard deviation of errors (often denoted by $\sigma : \text{Var}(\varepsilon) = \sigma^2$)
- R-squareds tell how large proportion of the variance in Y the line explains compared to the total variance in Y. Typically, we should be looking at Adjusted Rsquared as it is more reliable than the raw Rsquared if there are many predictors in the model.

In this case, the linear model on height explains almost 60% of the variation in weight and therefore about 40% of variation in weight is something that cannot be explained by (a linear effect of) height only. Note that in the case of simple linear regression with one predictor (here height), Rsquared is exactly the square of the correlation between X and Y.

```
cor(y$weight, y$height)^2
```

```
## [1] 0.5940256
```

```
summary(lm.1)$r.squared
```

```
## [1] 0.5940256
```

This explains why correlation coefficient is a measure of strength of a linear association between variabels: When $r = \pm 1$, then the linear regression model explains all variation in Y and hence the observations are on a single line in X-Y coordinates.

Let's see what columns we have in the `lm` object:

```
names(lm.1)
```

```
##  [1] "coefficients"  "residuals"     "effects"       "rank"
##  [5] "fitted.values" "assign"        "qr"            "df.residual"
##  [9] "xlevels"       "call"          "terms"         "model"
```

We could get both the fitted values (`lm.1$fitted.values`) as well as the residuals (`lm.1$residuals`) for each individual from the regression model object. For example, let's find for which individuals the predicted weight is more than 20kg off.

```
ii = which(abs(lm.1$residuals) > 20) #returns row indexes for which condition is TRUE
cbind(y[ii,2:4], lm.1$fitted.values[ii], lm.1$residuals[ii])
```

```
##     sex weight height lm.1$fitted.values[ii] lm.1$residuals[ii]
## 20    M    119    180               76.11303           42.88697
## 29    M    101    183               79.56070           21.43930
## 53    M    102    185               81.85914           20.14086
## 96    M    103    185               81.85914           21.14086
## 191   M     89    173               68.06848           20.93152
```

We notice two things. First, there is one outlier case where weight is 119 kg whereas it is predicted to be 76 kg based on the height of 180 cm. Second, all of the worst predictions happened for males. Indeed, it is unlikely that a single linear regression model would be good for both males and females.

Let’s add sex as another predictor to the model to allow for different mean weights in males and females. Since `sex` is of type “character” with two values (“M” and “F”), R automatically treats it as *factor* in the regression model. For factor variables, the regression model will get different baseline values for different levels of the factor (here males and females).

```
lm.2 = lm(weight ~ height + sex, data = y)
summary(lm.2)
```

```
##
## Call:
## lm(formula = weight ~ height + sex, data = y)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -20.131  -4.889  -0.404   5.205  41.490
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -76.63620   15.75543  -4.864 2.36e-06 ***
## height        0.81072    0.09555   8.485 5.24e-15 ***
## sexM          8.21621    1.71726   4.784 3.37e-06 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 8.086 on 196 degrees of freedom
## Multiple R-squared:  0.6365, Adjusted R-squared:  0.6328
## F-statistic: 171.6 on 2 and 196 DF,  p-value: < 2.2e-16
```

Now our model says that after we account for the difference in the average weight between males and females, each cm in height corresponds to +0.81 kg in height (it was +1.15 kg when sex wasn’t accounted for). Additionally, we see that for the same height, a male weighs on average 8.2 kg more than a female. This model is a better description of the data as the adjusted Rsquared has increased to 63% from 59% in model `lm.1`.

But are there differences in the height-weight slopes between the sexes? Let’s simply fit separate linear models in males and in females and check the slopes.

```
males = y$sex == "M"
lm.m = lm(weight ~ height, data = y[males,])
lm.f = lm(weight ~ height, data = y[!males,])
```

We see that the slope in females is

```r
b = summary(lm.f)$coeff[2,1] #coeff matrix has one row for each parameter
s = summary(lm.f)$coeff[2,2] #row 2 is for slope and row 1 for intercept
cbind(b=b, low95 = b-1.96*s, up95 =b+1.96*s)
```

```
##              b     low95      up95
## [1,] 0.6229425 0.4276806 0.8182044
```

And in males

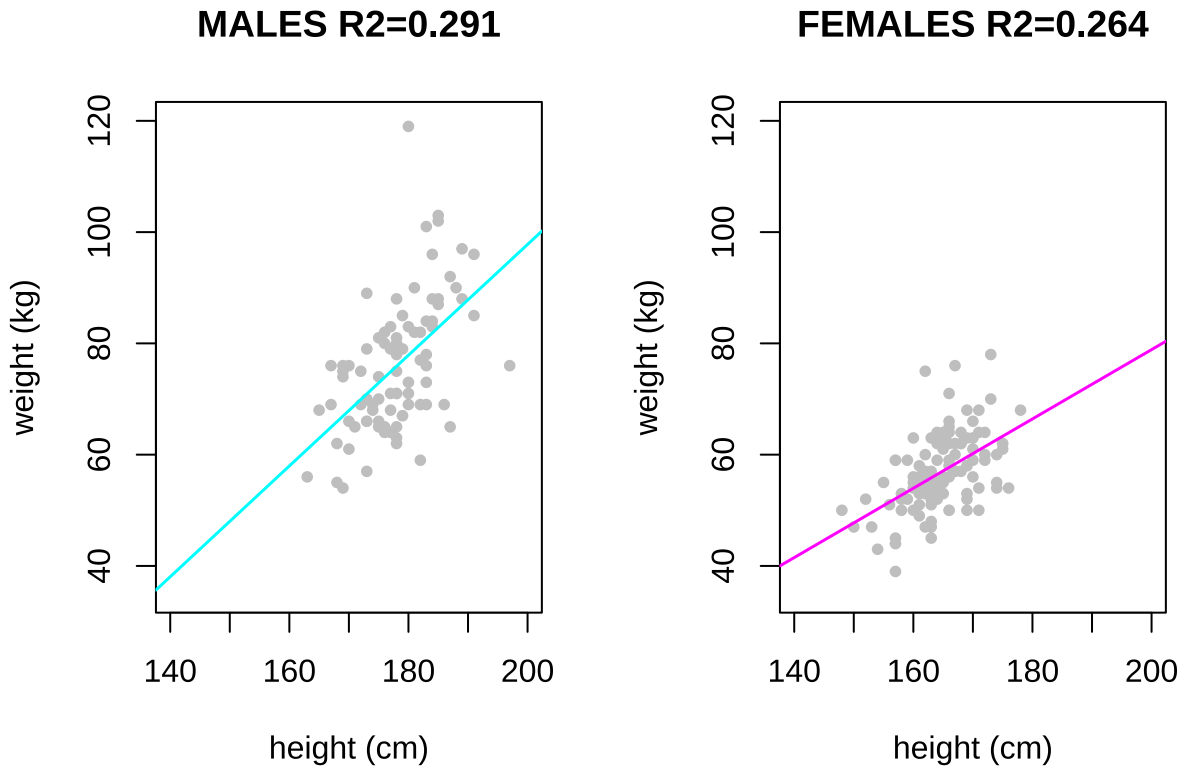
```r
b = summary(lm.m)$coeff[2,1]
s = summary(lm.m)$coeff[2,2]
cbind(b=b, low95 = b-1.96*s, up95 = b+1.96*s)
```

```
##              b     low95     up95
## [1,] 0.9955981 0.6670175 1.324179
```

It seems like the slope in females might be considerably smaller than in males, but since the 95%CIs are quite wide, this cannot be concluded reliably from these data alone.

Let's see how much the regression model explains within each sex.

```r
par(mfrow = c(1,2))
plot(y[males,"height"], y[males,"weight"], pch = 20, ylim = c(35, 120), xlim = c(140, 200),
     ylab = "weight (kg)", xlab = "height (cm)", col="gray")
abline(lm.m, col = "cyan", lwd = 1.5)
title(paste0("MALES R2=",signif(summary(lm.m)$r.squared,3)))
plot(y[!males,"height"], y[!males,"weight"], pch = 20, ylim = c(35, 120), xlim = c(140, 200),
     ylab = "weight (kg)", xlab = "height (cm)", col = "gray")
abline(lm.f, col = "magenta", lwd = 1.5)
title(paste0("FEMALES R2=", signif(summary(lm.f)$r.squared,3)))
```



We see a dramatic decrease in variance explained after we have adjusted the analyses for sex. This is because there is a strong difference in both height and weight distributions between males and females, and therefore a large part of the explanatory power of the orginal model was the effect of sex. After sex is accounted for, height explains about 25-30% of variation in weight. This shows how important the additional variables, also called covariates, can be for the interpretation of the regression model parameters. We talk more about these effects of *covariate adjustment* later in this document.

Exercises 2

1. Generate 10 values for x from interval $(-1, \ldots, 1)$ and generate corresponding values for y using `rnorm(10,x,sigma)` where `sigma` takes sequentially values of 0.1, 1 and 10. For each value of `sigma`, plot the points on X-Y coordinates, fit the linear model and add the fitted line to the picture. Use `summary()` command to see whether there is a statsitically significant linear relationship between X and Y in each data set.

2. Make a scatter plot of `weight` on x-axis and `repwt` on y-axis. Fit a linear model regressing `repwt` on `weight`. Add the regression line to the Figure. What is the slope of the model? What is the value of `repwt` for individual 19 and what is the fitted value of `repwt` based on the model for individual 19? What is the residual error for individual 19?

3. Predictions from the regression model

Let's use `predict()` function to predict a weight for two new individuals who are 160 cm and 180 cm tall. We need to give the input data in the same kind of data.frame that was used in the original model fit with `lm()` function. The uncertainty of prediction can be asked on the level of individual value `interval = "prediction"` or as a confidence interval for the population average at the given parameters `interval = "confidence"`. Typically, we are interested in predicting new observations, and we use `"prediction"`. Then the uncertainty is larger than when predicting the population mean with the `"confidence"` option.

```
data.in = data.frame(height = c(160,180) )
predict(lm.f, newdata = data.in, interval = "prediction") #for females of 160 and 180 cm
```

```
##        fit      lwr      upr
## 1 53.96238 42.10332 65.82143
## 2 66.42123 54.21885 78.62361
```

```
predict(lm.m, newdata = data.in, interval = "pred") #for males of 160 and 180 cm
```

```
##        fit      lwr      upr
## 1 57.96565 36.95574 78.97555
## 2 77.87761 57.73251 98.02271
```

We see that a female of 180 cm are predicted at 66 kg with 95% interval (54,79) whereas male of same height at 78kg (58,98).

Exercises 3.

1. What is the predicted weight of a male who is 170 cm tall? What is the predicted population average weight of all males who are 170 cm tall? Give prediction intervals in both cases.
2. Fit a linear model regressing `height` on `weight` in females. What is the predicted height of a woman who is 170cm tall?

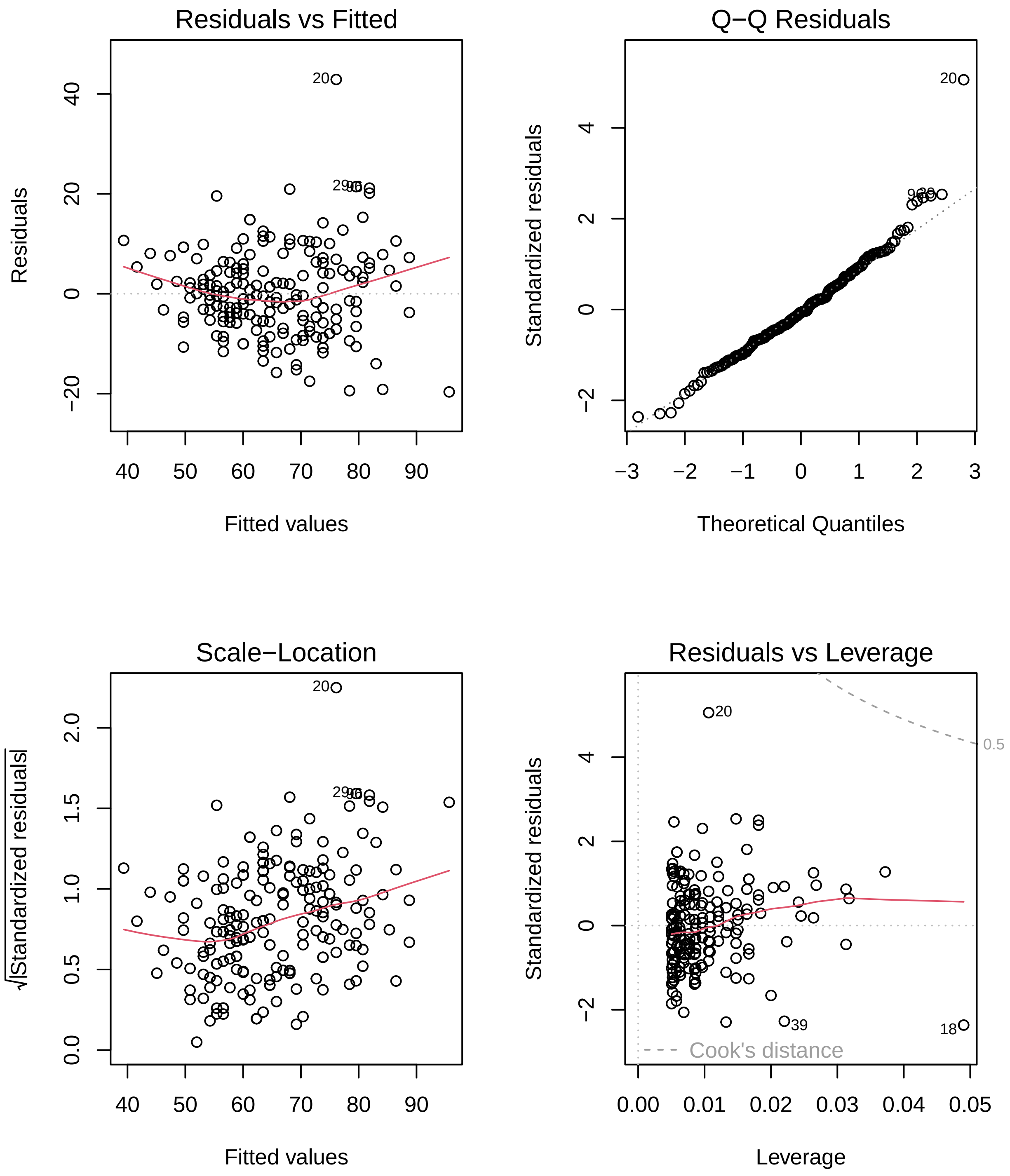
4. Assessing model fit

How do we know whether a linear model is an approriate model for our data set? First we should consider this conceptually: Is there a reason to think that a linear relationship between the variables is appropriate for the research question we have in mind? If answer is positive, then we should ask how is the linear model fitting the data and are the modling assumptions valid? The main assumptions for linear model are:

1. The relationship between Y and X is linear.
2. Errors of Y have constant variance for all values of X.
3. Errors of Y for different units/individuals are independent of each other.
4. Errors have Normal distribution (for SE and P-value to be valid).

There arefour standard plots that we can use to assess these assumptions and they are made by calling `plot()` on the regression model object. Let's first check how our first model without an adjustment for `sex` behaves in these diagnostic plots:

```
par(mfrow = c(2,2)) #plot diagnostics to 2x2 area
plot(lm.1)
```

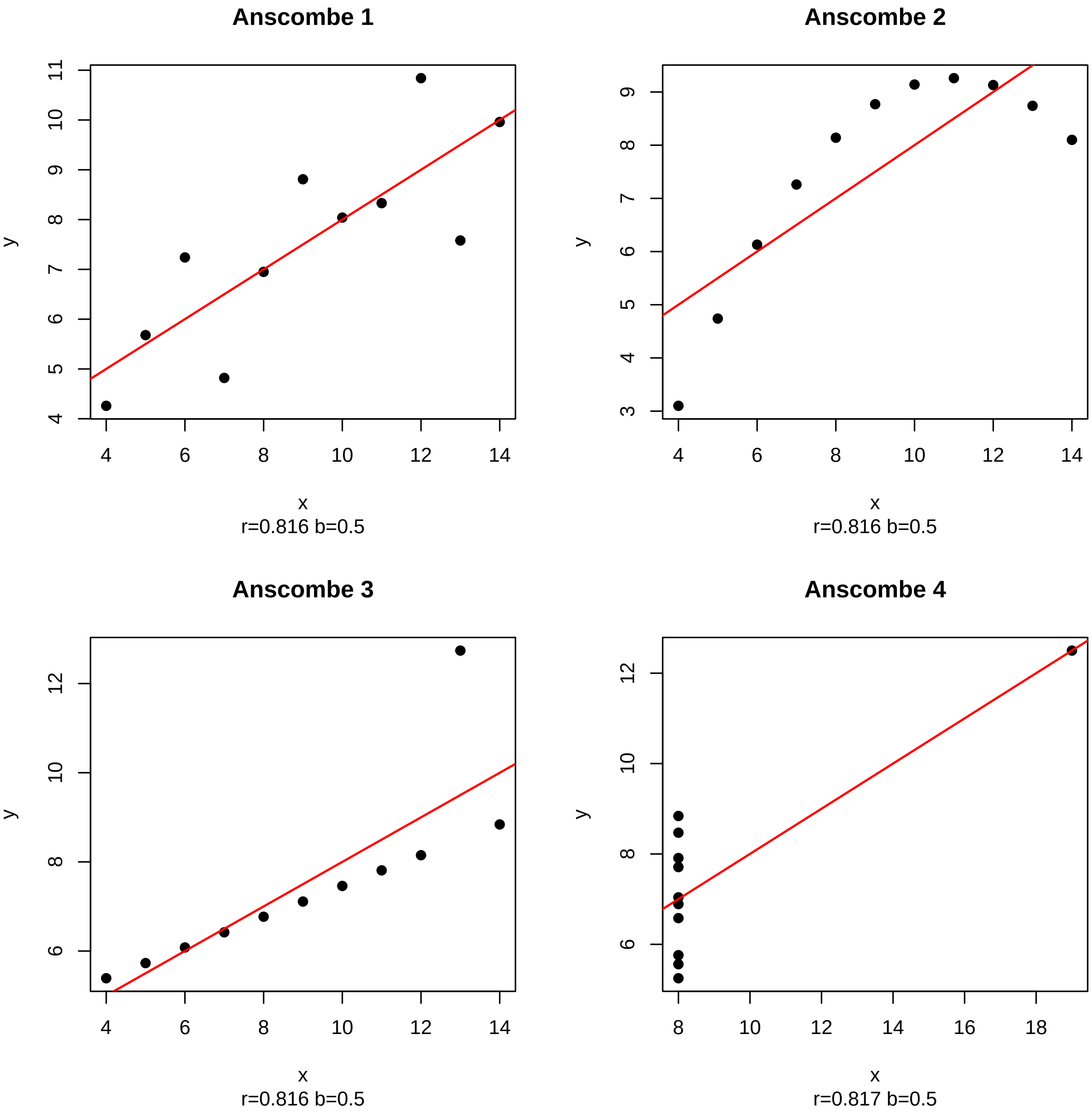


To interpret these plot, read https://data.library.virginia.edu/diagnostic-plots/. For our test data, the diagnostics look quite good, except that the observation 20 seems a bit outlying from the rest. However, the lower right plot shows that it does not have a large influence on the fitted line (as explained in the link above) and therefore we are not that concerned about it.

Anscombe’s quartet

Francis Anscombe generated four data sets of 11 observations that have similar summary statistics (means, variances and correlations) but look very different when plotted. His point was to emphasize the importance of plotting the data whenever possible. Here we use these four sets to assess how a linear model fits different data sets. Let’s start by plotting the data sets and fitting the linear model and computing correlations and slopes.

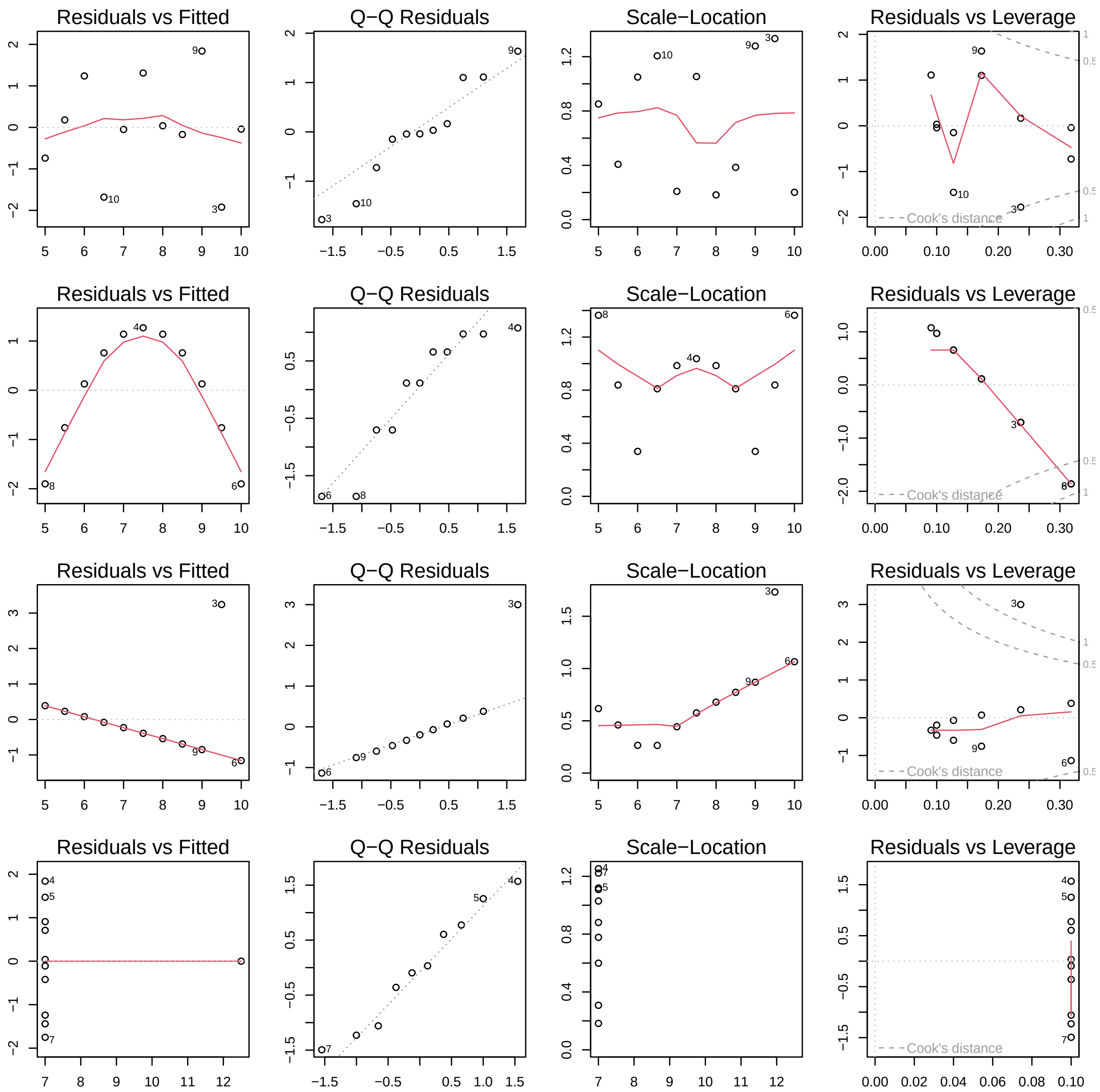
```r
par(mfrow = c(2,2))
lm.ans = list() #this is empty list where linear models are collected
for(ii in 1:4){
  x = anscombe[,ii] #x-coordinates for Anscombe's data set ii
  y = anscombe[,4+ii] #y-coordinates
  lm.ans[[ii]] = lm(y ~ x)
  plot(x,y, main=paste("Anscombe",ii), pch = 19, xlab = "x", ylab = "y",
       sub = paste0("r=",signif(cor(x,y),3)," b=",signif(coefficients(lm.ans[[ii]])[2],3)))
  abline(lm.ans[[ii]], col = "red", lwd = 1.5)
}
```



We see that the data sets are identical with respect to x-y correlation and the linear model slope, but we also see that the data sets are very different in other respects, and, in particular, the linear model does not seem appropriate for the data sets 2, 3 and 4. Let's see the diagnostic plots for these linear models.

```r
par(mfrow = c(4,4))
par(mar = c(3,3,2,1))
for(ii in 1:4){plot(lm.ans[[ii]])}
```

```
## Warning: not plotting observations with leverage one:
##   8
```



Question: Find problematic patterns from the diagnostic plots on rows 2, 3 and 4, which tell that the linear model is not approriate for those data sets.

5. Example analysis: Social factors in 1998

Let's study some social factors from around the world using a UN data sets from 1998. For description of variables see http://vincentarelbundock.github.io/Rdatasets/doc/carData/UN98.html.

```
y = read.csv("https://www.mv.helsinki.fi/home/mjxpirin/HDS_course/material/UN98.csv",
             as.is = TRUE)
head(y) #show some first rows
```

```
##                   X region  tfr contraception educationMale educationFemale
## 1    Afghanistan   Asia 6.90            NA            NA              NA
## 2        Albania Europe 2.60            NA            NA              NA
## 3        Algeria Africa 3.81            52          11.1             9.9
## 4 American.Samoa   Asia   NA            NA            NA              NA
## 5        Andorra Europe   NA            NA            NA              NA
## 6         Angola Africa 6.69            NA            NA              NA
##   lifeMale lifeFemale infantMortality GDPperCapita economicActivityMale
## 1     45.0       46.0             154         2848                 87.5
## 2     68.0       74.0              32          863                   NA
## 3     67.5       70.3              44         1531                 76.4
## 4     68.0       73.0              11           NA                 58.8
## 5       NA         NA              NA           NA                   NA
## 6     44.9       48.1             124          355                   NA
##   economicActivityFemale illiteracyMale illiteracyFemale
## 1                    7.2         52.800            85.00
## 2                     NA             NA               NA
## 3                    7.8         26.100            51.00
## 4                   42.4          0.264             0.36
## 5                     NA             NA               NA
## 6                     NA             NA               NA
```

Let's give these variables shorter names to avoid writing long names below:

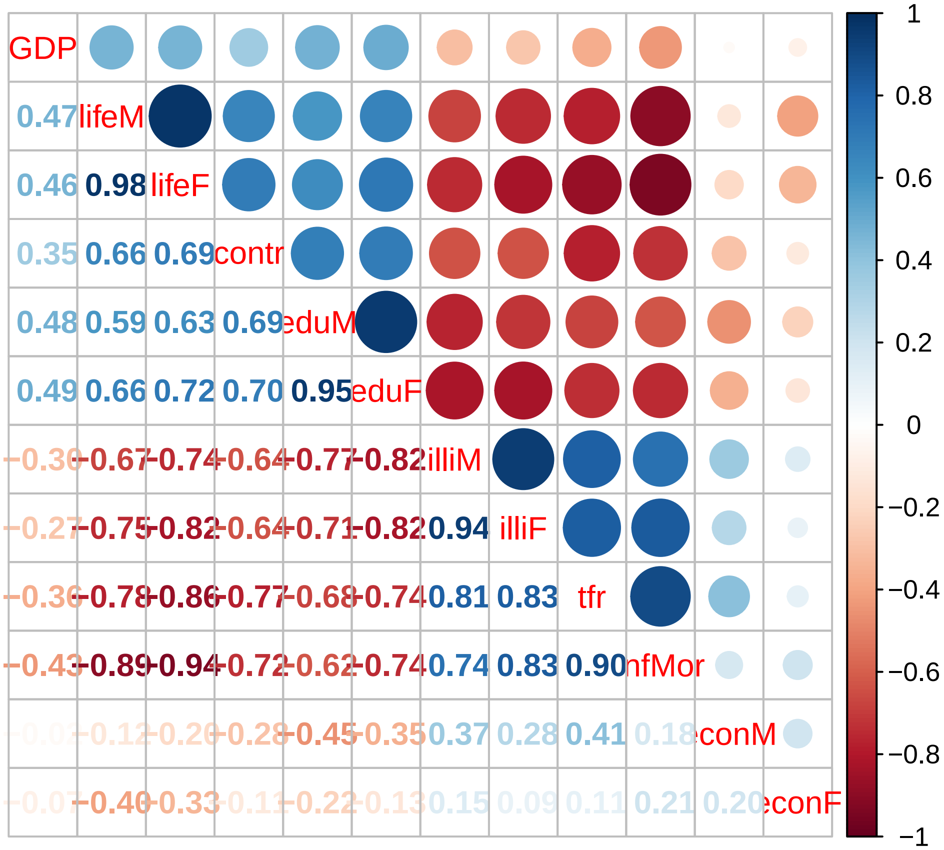
```r
colnames(y) = c("country","region","tfr","contr","eduM","eduF","lifeM",
                "lifeF","infMor","GDP","econM","econF","illiM","illiF")
```

Let's plot the pairwise correlations using `corrplot`

```r
#install.packages("corrplot")
library(corrplot)
```

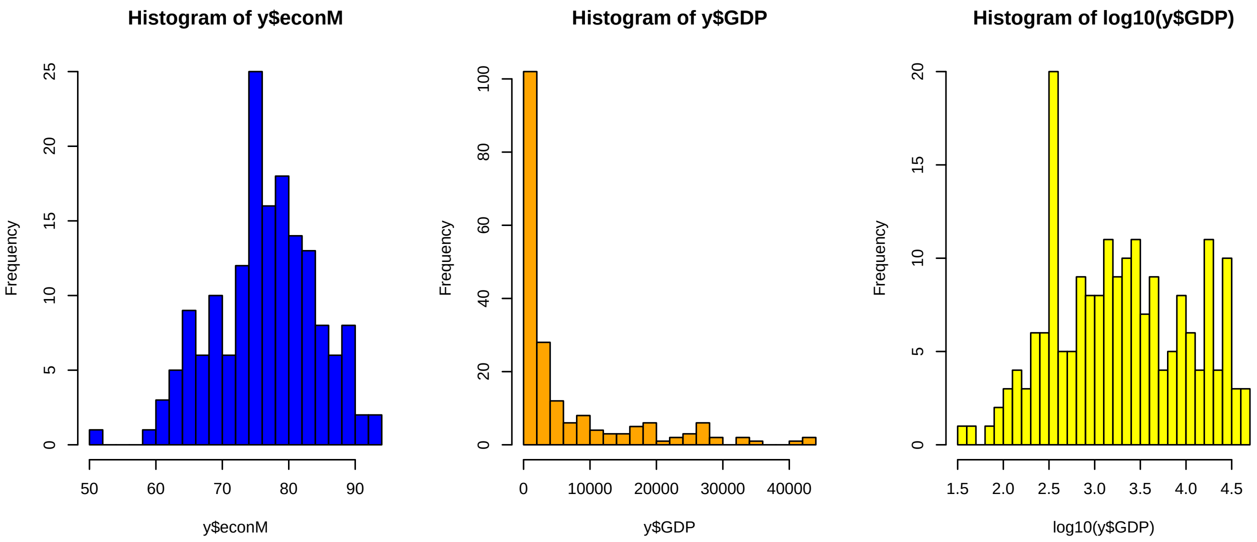
```
## corrplot 0.92 loaded
```

```r
x = as.matrix(y[,3:14]) #make a matrix of numeric columns for cor()
corr = cor(x, use = "complete") #"complete" means remove NAs
corrplot.mixed(corr, order="hclust")
```



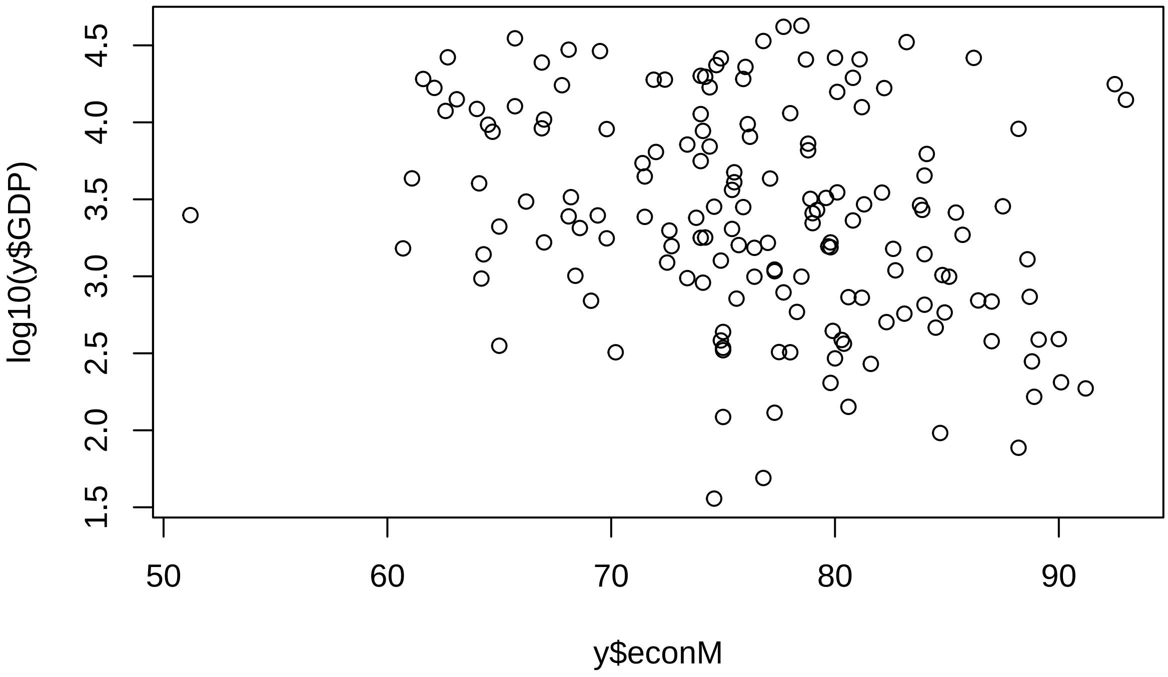
We see two clear blocks of highly correlated variables. First block is about positive things like life expectancy, education and availability of contraception and the second about negative things like illiteracy, infant mortality and high fertility rate. There are also some surprising correlations, or rather surprising lack of correlation, between, for example, GDP and economic activity. Let's have a more careful look. Let's first check the distributions of these two variables.

```
par(mfrow = c(1,3))
hist(y$econM, col = "blue", breaks = 25)
hist(y$GDP, col = "orange", breaks = 25)
hist(log10(y$GDP), col = "yellow", breaks = 25)
```



We see that activity is roughly symmetric and Normally distributed but GDP is an example of highly skewed distribution with a strong tail to right. This is typical to variables describing wealth or, more generally, variables that take only non-negative values. Often logaríthm of such a variable is a much better input variable to linear regression as it is often more symmetric and Normally distributed (as is the case here). Thus we will use log10 of GDP.

```
plot(y$econM, log10(y$GDP))
```

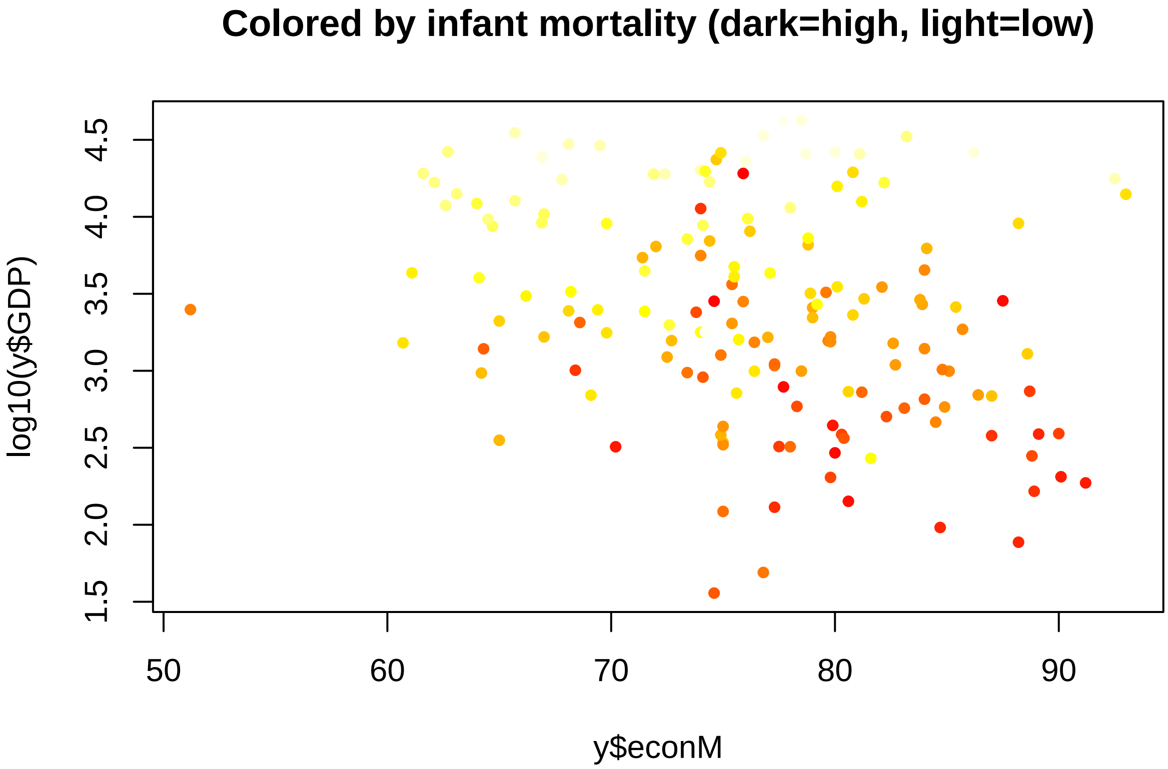


```
y[which(y$econM > 90),] #print countries with over 90% Males "economically active"
```

```
##                     country region  tfr contr eduM eduF lifeM lifeF infMor   GDP
## 29                  Burundi Africa 6.28     9  5.1  4.0  45.5  48.8    114   205
## 35                     Chad Africa 5.51    NA   NA   NA  46.3  49.3    115   187
## 153                   Qatar   Asia 3.77    32 10.6 11.6  70.0  75.4     17 14013
## 194 United.Arab.Emirates   Asia 3.46    NA  9.8 10.3  73.9  76.5     15 17690
##     econM econF illiM illiF
## 29   90.1  90.6  50.7  77.5
## 35   91.2  25.3  37.9  65.3
## 153  93.0  27.5  20.8  20.1
## 194  92.5  24.2  21.1  20.2
```

We have poor countries like Burundi and Chad and rich countries like Qatar and United Arab Emirates that both have high proportion of economic activity. One clear factor separating these is infant mortality. Let's look the same plot but color points by infant mortality.

```
plot(y$econM, log10(y$GDP), pch = 20, col = heat.colors(nrow(y))[nrow(y)-rank(y$infMor)],
     main = "Colored by infant mortality (dark=high, light=low)")
```



Based on this we can guess what linear model says when we predict GDP by both Economic activity and Infant mortality:

```
lm.0 = lm(log10(GDP) ~ econM + infMor, data = y)
summary(lm.0)
```

```
##
## Call:
## lm(formula = log10(GDP) ~ econM + infMor, data = y)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.49103 -0.29882 -0.04389  0.31470  1.75052
##
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  4.112476   0.417481   9.851   <2e-16 ***
## econM       -0.002795   0.005644  -0.495    0.621    
## infMor      -0.014051   0.001251 -11.236   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.496 on 155 degrees of freedom
##   (49 observations deleted due to missingness)
## Multiple R-squared:    0.5,  Adjusted R-squared:  0.4936 
## F-statistic: 77.51 on 2 and 155 DF,  p-value: < 2.2e-16
```

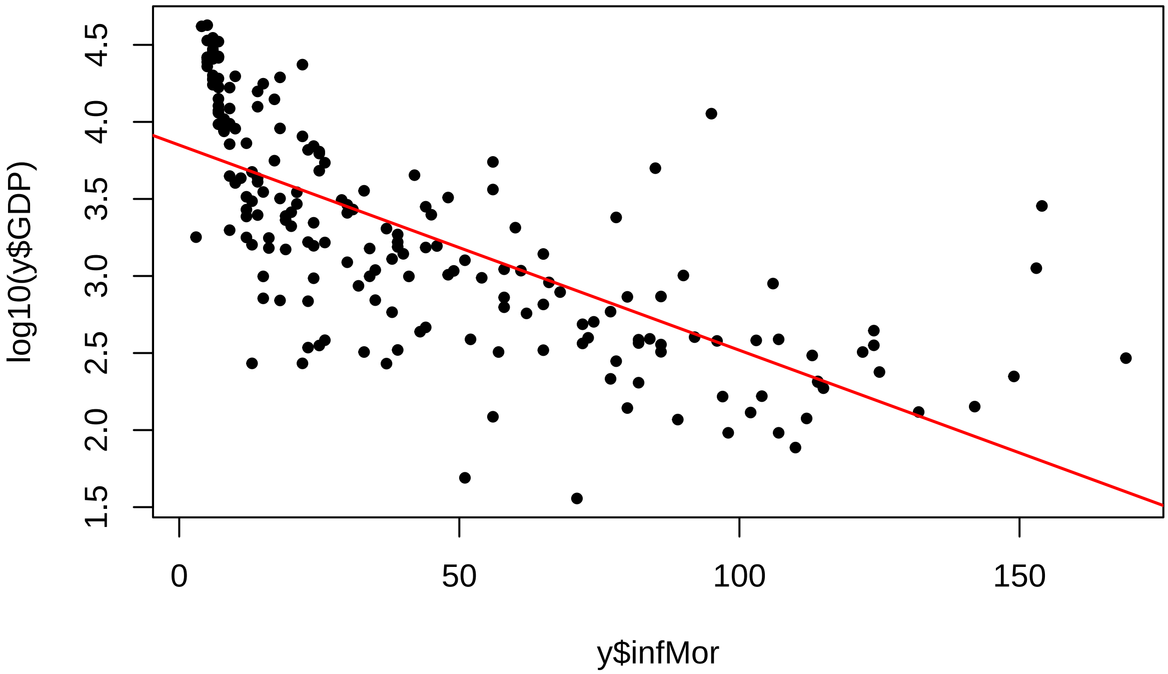
Economic activity is not important predictor whereas Infant mortality is.

Let's do linear model using Infant mortality alone as predictor.

```
lm.1 = lm(log10(GDP) ~ infMor, data = y)
summary(lm.1)
```

```
##
## Call:
## lm(formula = log10(GDP) ~ infMor, data = y)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.48033 -0.30369 -0.04039  0.34797  1.65554 
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  3.8496233  0.0551925   69.75   <2e-16 ***
## infMor      -0.0133157  0.0009465  -14.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5057 on 191 degrees of freedom
##   (14 observations deleted due to missingness)
## Multiple R-squared:  0.5089, Adjusted R-squared:  0.5063 
## F-statistic: 197.9 on 1 and 191 DF,  p-value: < 2.2e-16
```

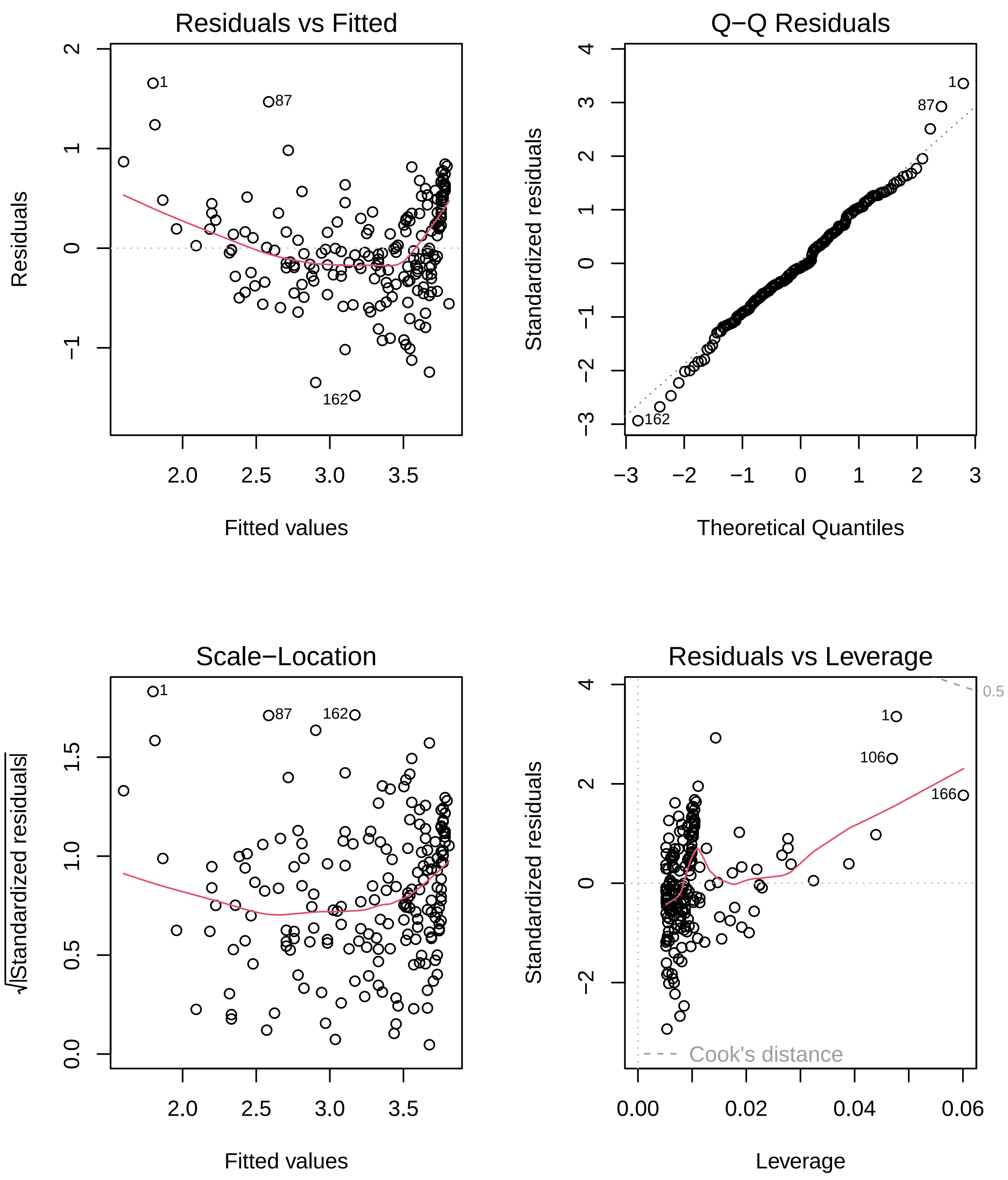
```
plot(y$infMor, log10(y$GDP), pch = 20)
abline(lm.1, col = "red", lwd = 1.5)
```



We see that log10 of GDP is decreasing by 0.014 for each unit of infant deaths (per 1000 infants). In other words, when infant mortality increases one per mille, GDP drops multiplicatively by a factor of 10^(-0.014051)=0.9681642, i.e., it drops about 3%. This model alone explains about 50% of variation in GDP.

Does this model fit the data well? It looks like there is tendency of errors being positive at the ends, which would suggest adding second order term in the model. Let's make diagnostic plots.

```
par(mfrow = c(2,2))
plot(lm.1)
```



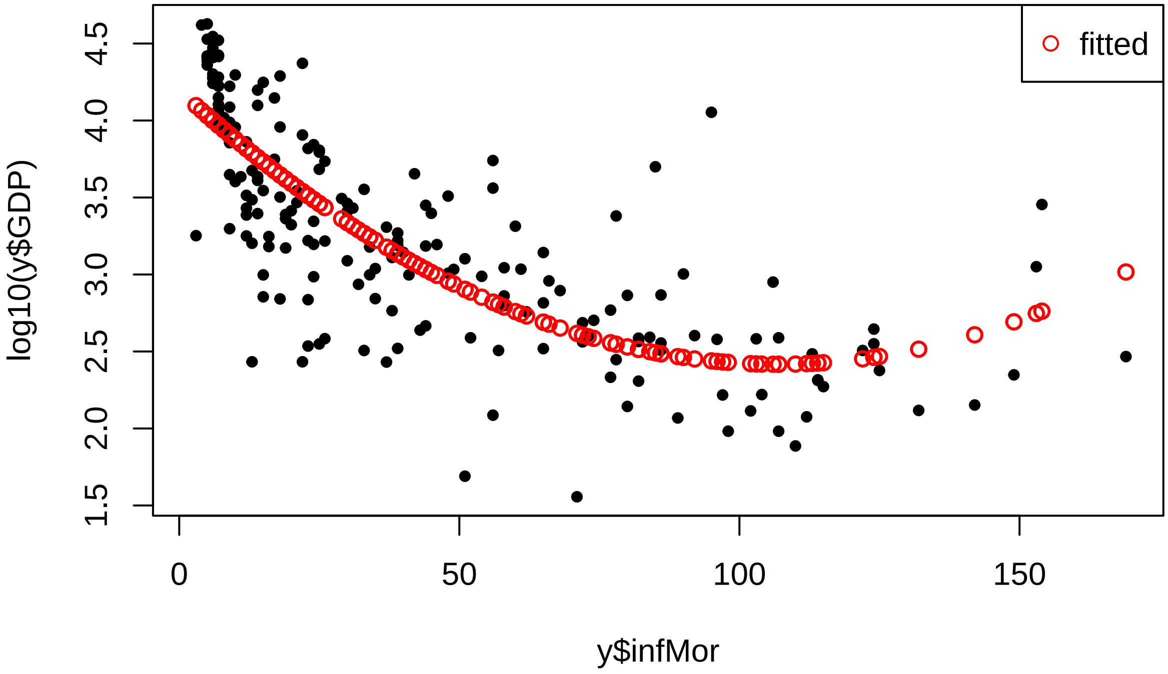
No other clear problems except that residuals have a shape of a prabola in the first plot. Let's add quadratic term whence our model becomes `log10(GDP) ~ InfMor + InfMor^2`. The quadratic term needs to be input through `I()` notation as below.

```
lm.2 = lm(log10(GDP) ~ infMor + I(infMor^2), data = y)
summary(lm.2)
```

```
##
## Call:
```

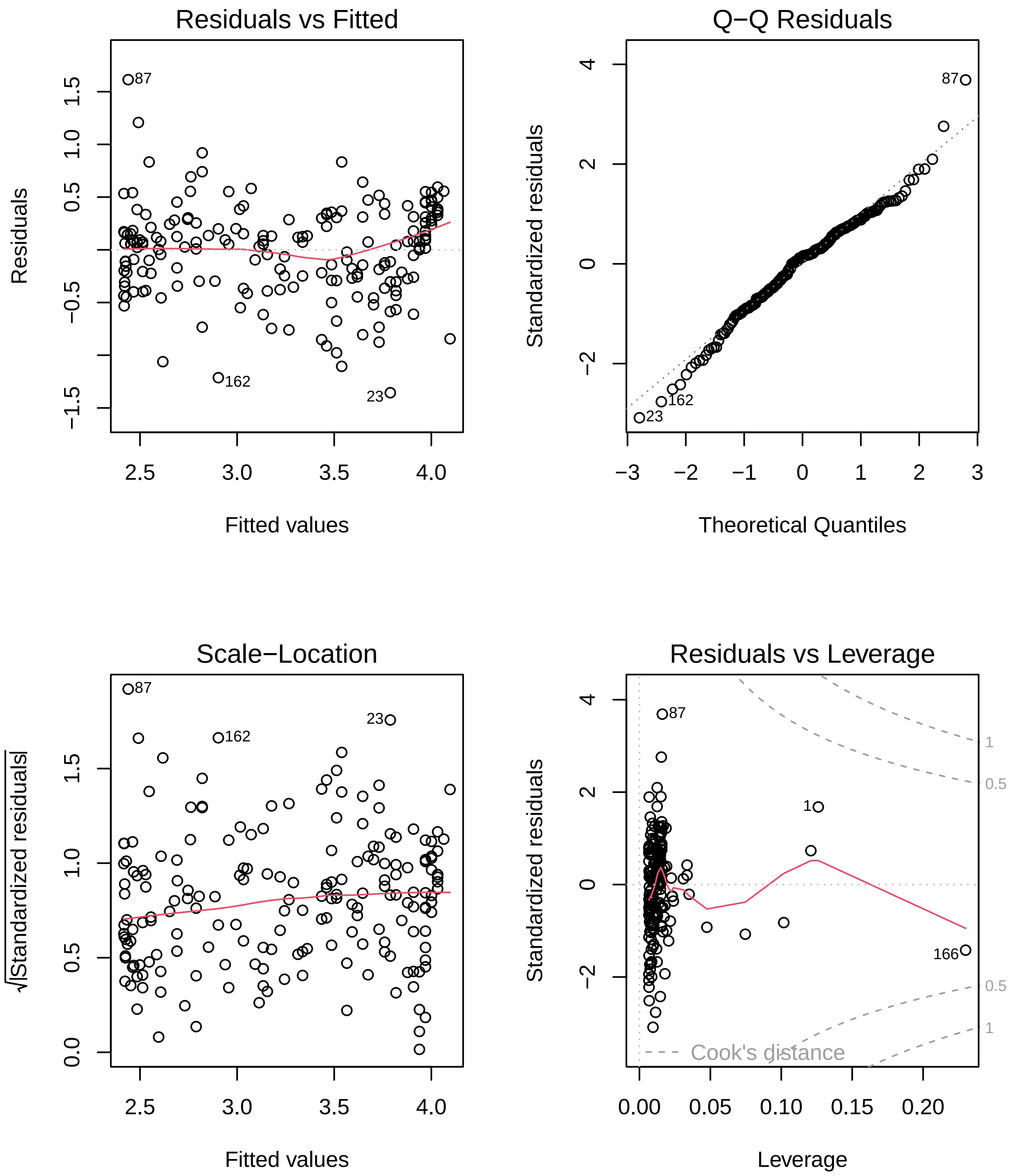
```
## lm(formula = log10(GDP) ~ infMor + I(infMor^2), data = y)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.35590 -0.27433  0.06152  0.30104  1.61427
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.195e+00  6.546e-02   64.08  < 2e-16 ***
## infMor      -3.326e-02  2.690e-03  -12.36  < 2e-16 ***
## I(infMor^2)  1.555e-04  1.996e-05    7.79 4.23e-13 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.4414 on 190 degrees of freedom
##   (14 observations deleted due to missingness)
## Multiple R-squared:  0.6278, Adjusted R-squared:  0.6239
## F-statistic: 160.2 on 2 and 190 DF,  p-value: < 2.2e-16
```

```
plot(y$infMor, log10(y$GDP), pch = 20)
#To plot the fitted values from lm.2, we need the correct x-coordinates, and these we get as:
included = setdiff(1:nrow(y), as.numeric(lm.2$na.action)) #these indexes of y were incl'd in regression
points(y$infMor[included], lm.2$fitted.values, col = "red", lwd = 1.5)
legend("topright", pch = 1, legend = c("fitted"), col = "red")
```



This quadratic term clearly improves model fit (R2 increases from 50% to 62%) and fitted values take into account the curved pattern in data. Let’s see diagnostic plots.

```
par(mfrow=c(2,2))
plot(lm.2)
```



Look good.

We have found very strong (inverse) association between Infant mortality and GDP. Can we now say that Infant mortality is **the cause** of differences in GDP? The answer is **NO**. Causal statements can never be made from regression analysis alone! **Correlation is not causation** is an important fact to remember in all fields of data analysis! This is because there can be some third factor Z, that could be causally influencing both X and Y, and hence the effect of Z could cause the observed correlation between X and Y even though neither X nor Y was causally influencing the other. Such a third variable Z is called a **confounder** when it is *confounding* the association between X and Y. A common statistics textbook example is the positive correlation between ice-cream sales and deaths by drowning. Is ice-cream causing drowning? No, it is sun and hot temperature that increase ice-cream sales and also increase activity on water, which then also leads to more accidents on water, and consequently deaths by drowning. So weather is confounding the association between ice-cream sales and drowning. Correlation is not causation.

Since we can never measure all variables in observational studies, it remains very difficult to end up in causal inference using observational data as additional, unmeasured confounders could influence the correlation estimates. Luckily, in clinical medicine and experimental science, there is an important method to overcome the problem of unmeasured variables: **randomized controlled trial (RCT)** (link). The situation is more challenging for observational population-level epidemiology or social sciences, for example, and there causal statements are difficult to justify.

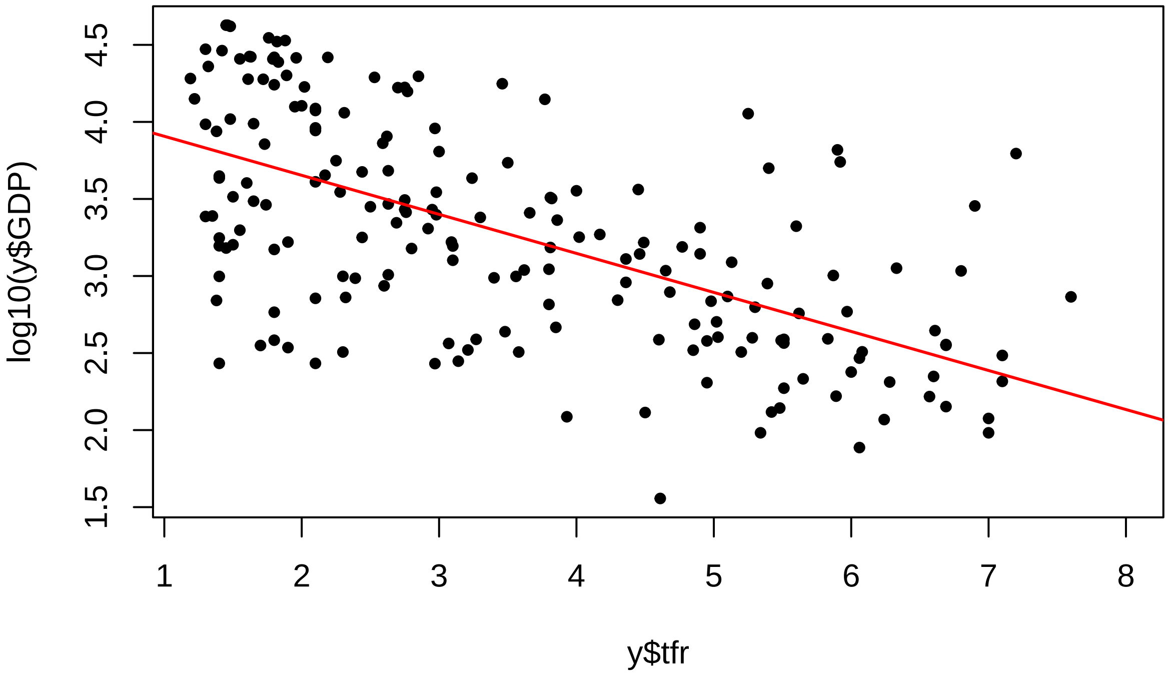
6. Multiple regression with social factors

Let's think a bit more about what happens when we include multiple predictors in the same linear regression model. Above we established that Infant Mortality was a strong predictor of GDP. What about Total Fertility Rate (tfr)?

```
lm.3 = lm(log10(GDP) ~ tfr, data = y)
summary(lm.3)
```

```
##
## Call:
## lm(formula = log10(GDP) ~ tfr, data = y)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.4358 -0.3703 -0.0134  0.4197  1.4586
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.15995    0.09179   45.32   <2e-16 ***
## tfr         -0.25333    0.02343  -10.81   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5647 on 188 degrees of freedom
##   (17 observations deleted due to missingness)
## Multiple R-squared:  0.3834, Adjusted R-squared:  0.3801
## F-statistic: 116.9 on 1 and 188 DF,  p-value: < 2.2e-16
```

```
plot(y$tfr, log10(y$GDP), pch = 20)
abline(lm.3, col = "red", lwd = 1.5)
```



A very clear association where higher fertility associates with decreasing GDP. The model explains about 38% of the variation. Let's then include both Infant Mortality and tfr in the same model. This is an example of multiple regression model.

```
lm.4 = lm(log10(GDP) ~ tfr + infMor, data = y)
summary(lm.4)
```

```
##
## Call:
## lm(formula = log10(GDP) ~ tfr + infMor, data = y)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.35185 -0.30586 -0.04307  0.34557  1.62657
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.887919   0.088177  44.092  < 2e-16 ***
## tfr         -0.021036   0.037179  -0.566    0.572
## infMor      -0.012434   0.001677  -7.416 4.14e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4958 on 186 degrees of freedom
##   (18 observations deleted due to missingness)
## Multiple R-squared:  0.5206, Adjusted R-squared:  0.5154
## F-statistic:   101 on 2 and 186 DF,  p-value: < 2.2e-16
```

In this model, only `infMor` seems important (effect is many SEs away from zero as denoted by small P-value) whereas the association with `tfr` has disappeared (effect dropped from -0.25 to -0.02 and P-value went from <2e-16 to 0.57). This is possible because `infMor` and `tfr` are correlated and therefore their effect estimates in multiple regression model can be different from their effects in simple regression models where only one of them is included at once. In these data it seems that the effect that `tfr` explained when it was alone in the model, can be completely explained by a correlated variable `InfMor`. Multiple regression has here produced new insights over pairwise correlations.

We say that the effect size of `tfr` from multiple regression model "has been adjusted for" `infMor` whereas the effect of `tfr` from the simple regression model was not so adjusted. The interpretation for the effect estimates from multiple regression is that they tell what happens to the outcome variable (here log10(GDP)) per a unit change in the focal predictor (here `tfr`), when other predictors (here `infMor`) are kept constant. Thus, here we have learned that if we fix the level of infant mortality, then the fertility rate does not anymore noticeably affect GDP.

Note that, in general, when several highly correlated variables are simultaneously included in a multiple regression model, then the standard errors of individual parameters may become large and it is difficult to interpret the parameters separately from the other parameters with a meaningful precision. In particular, individual P-values of predictors tend to become large but they cannot be used for inferring whether individual predictors are important or not for the total model fit!