Linear regression model and modelling of proportions

Josep L. Carrasco
Bioestadística. Departament de Fonaments Clínics
Universitat de Barcelona

Linear regression model

Introduction

We have seen that correlation analysis is useful to find out if two variables, say X and Y, are (linearly) related. Now, with the specification and estimation of a model we will have a tool to explain and summarize the way that Y **depends** on X.

Therefore, the aim of linear regression is to model the relationship between Y and X by a straight line.

$$Y = \alpha + \beta X$$

Usually Y is known as the dependent variable or outcome whereas X is identified as the predictor or covariable.

Notice that the model assumes that Y values depend on X values and not otherwise.

Usually the relationship between Y and X is not deterministic. That means that given a value of X it is not possible to know exactly the value of Y. There will be an error. This error is known as **random error**.

$$Y = \alpha + \beta X \ + \epsilon$$

It is commonly assumed that $\epsilon \sim N(0, \sigma_\epsilon)$

Example. Let's recover the example used in the correlation section where the association between the saturated fat intake (SF) and blood cholesterol levels (BC) was assessed. Remember that it was observed that the linear relation arose when the logarithm was applied to BC.

Next we have to decide which variable should be the outcome (Y) and which one the covariable (X). It makes sense to think that BC depends on SF, so that BC will be the outcome.

$$log(BC) = \alpha + \beta \cdot SF \ + \epsilon$$

Assumptions to validate

1) The relation between Y and X is linear.
2) The $Var(\epsilon) = \sigma_\epsilon^2$ is constant for all X.
3) The random error follows a Normal model with mean 0 and variance σ_ϵ^2.
4) Random errors are independent.
5) There are no extreme or influential values. These values have a large impact on the estimates, it means if they are removed the estimates get changed largely.

Interpretation and estimation of parameters

- α. It is known as **intercept** because it is the value of Y when $X = 0$.
- β. It is known as **slope**. It is the change in Y when X changes in one unit.

So that there are three parameters to estimate, α, β and σ_ϵ^2.

The most used estimation method to estimate α and β is the Ordinary Least Squares (OLS).

Let $\hat{Y}_i$ be the prediction for subject i using the regression model, i.e $\hat{Y}_i = \alpha + \beta X_i$ with $i = 1, \ldots, n$

The **residuals** (or errors) are defined as:

$$r_i = Y_i - \hat{Y}_i$$

the OLS approach minimizes the sum of square residuals Σr_i^2.

Predictive ability

One of the utilities of a regression model is to make predictions of Y from the values of X.

Therefore, it is necessary to evaluate the model's predictive ability to decide whether the model is valid to make predictions.

The predictive ability is assessed by the **determination coefficient** or R^2.

It is usually expressed as a percentage. Thus, the interpretation is **the percentage of variability of Y that is explained by X**.

If $R^2 = 1$ all the variability of Y is explained by X. There is no prediction error.

If $R^2 = 0$ no variability of Y is explained by X. The prediction error is maximum. This is the independence case.

Inference

The inference in a regression model will involve:

- Estimation of confidence intervals of the model parameters (α and β).
- Independence test based on the slope:

$$\begin{aligned} H_0: &\quad \beta = 0 \\ H_A: &\quad \beta \neq 0 \end{aligned}$$

Let's follow with the example. First we have to load the data.

```
coles=read.table("colesterol.txt",header=T)
```

The file contains three variables:

- *Pacient*. The subject's identification number.
- *Greixos*. Weekly saturated fat intake.
- *Colesterol*. Blood cholesterol level.

To estimate the model we will use the function *lm(formula,data)* with arguments:

- *formula*. The names of the outcome and the covariable separated by the symbol $\sim$.
- *data*. The data frame containing the variables in the model.

```
m1=lm(log(Colesterol)~Greixos,data=coles)
```

To show the estimates of the model run:

```
summary(m1)
```

```
Call:
lm(formula = log(Colesterol) ~ Greixos, data = coles)

Residuals:
     Min       1Q   Median       3Q      Max
-0.17526 -0.10409 -0.01826  0.09774  0.32920

Coefficients:
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 4.660430   0.094708   49.21  < 2e-16 ***
Greixos     0.023753   0.002296   10.35 5.27e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1343 on 18 degrees of freedom
Multiple R-squared:  0.8561,    Adjusted R-squared:  0.8481
F-statistic: 107.1 on 1 and 18 DF,  p-value: 5.268e-09
```

In the output, the section named "Coefficients" shows the estimates, standard errors and P-values linked to the hypothesis that the parameter is 0. The "Residual standard error" is the standard deviation of the random error. Therefore, the estimated model is:

$$Y = 4.66 + 0.024X + \epsilon$$

$$\epsilon \sim N(0, 0.134)$$

Additionally, the p-value related to the hypothesis of independence ($\beta = 0$) is very low (under 0.05) so the hypothesis is rejected.

Furthermore, the coefficient of determination is $R^2 = 85.61\%$, that means that above 85% of the variability of BC is explained by SF.

To build the 95% confidence intervals we can apply the following function:

```
confint(m1)
```

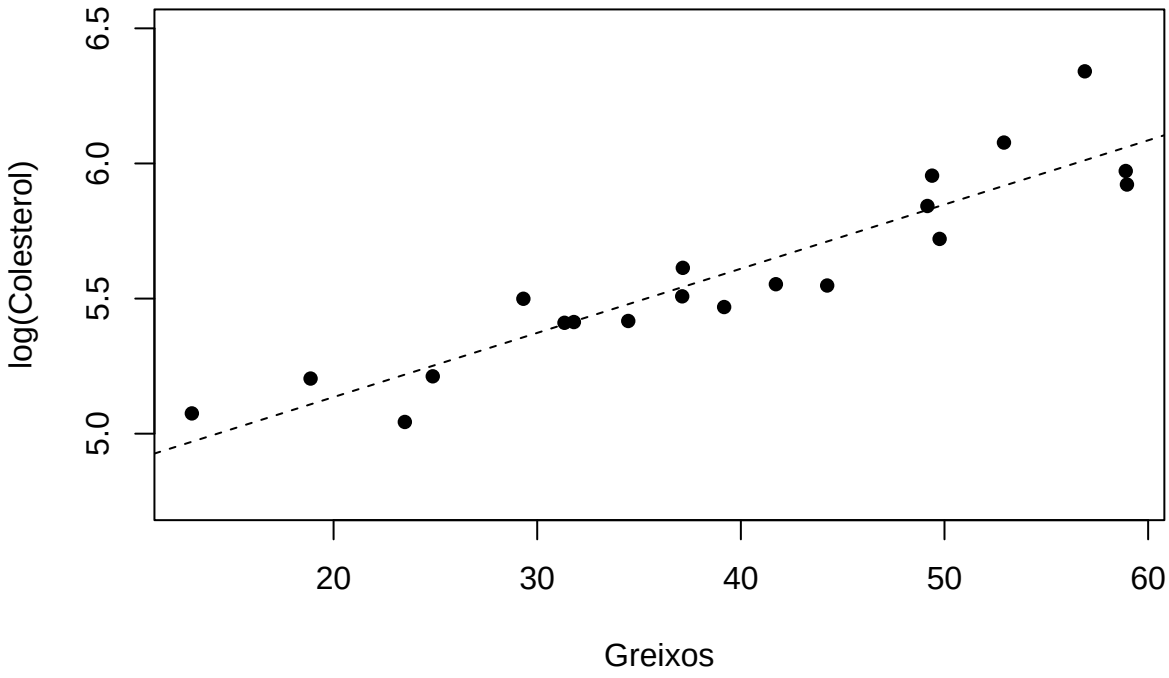
```
                2.5 %     97.5 %
(Intercept) 4.46145476 4.85940452
Greixos     0.01892994 0.02857545
```

Model validation

The validity of the model's assumptions we will be evaluated by inspection of some plots.

1) Plot both variables with the regression line.

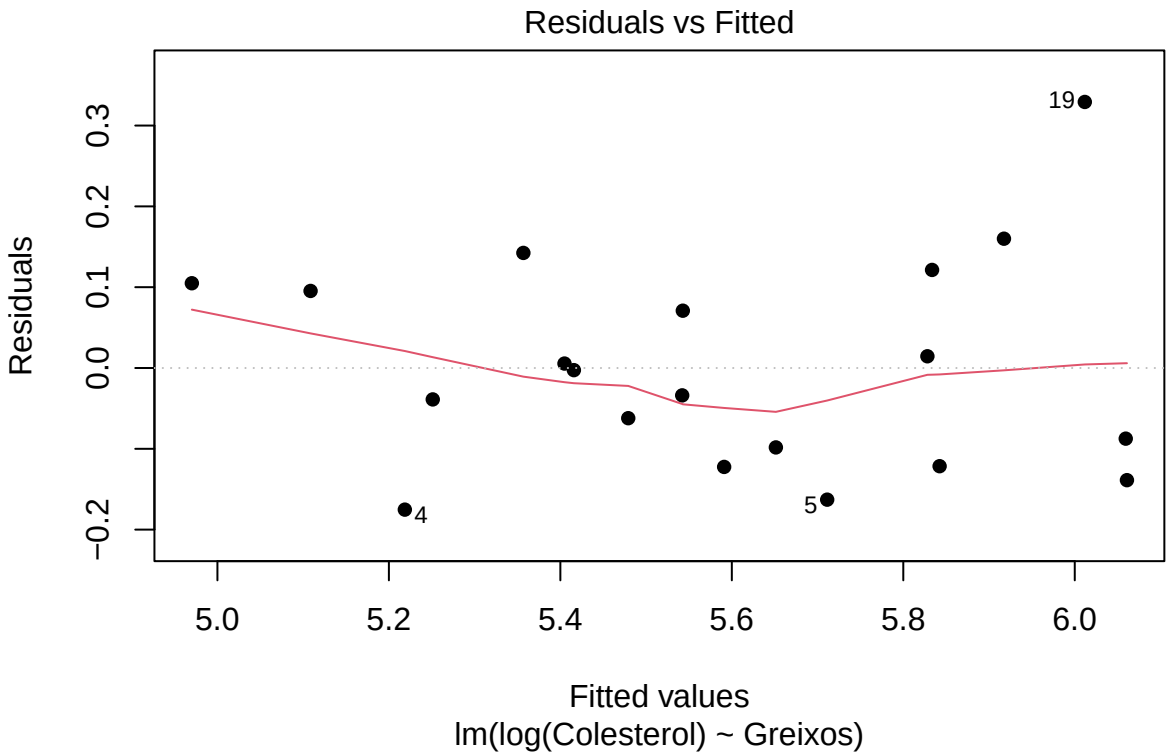
```
plot(log(Colesterol)~Greixos,data=coles,pch=16,ylim=c(4.75,6.5))
abline(m1,lty=2)
```



With this plot it is possible to check if the relation between Y and X is linear.

2) Plot of residuals and predictions.

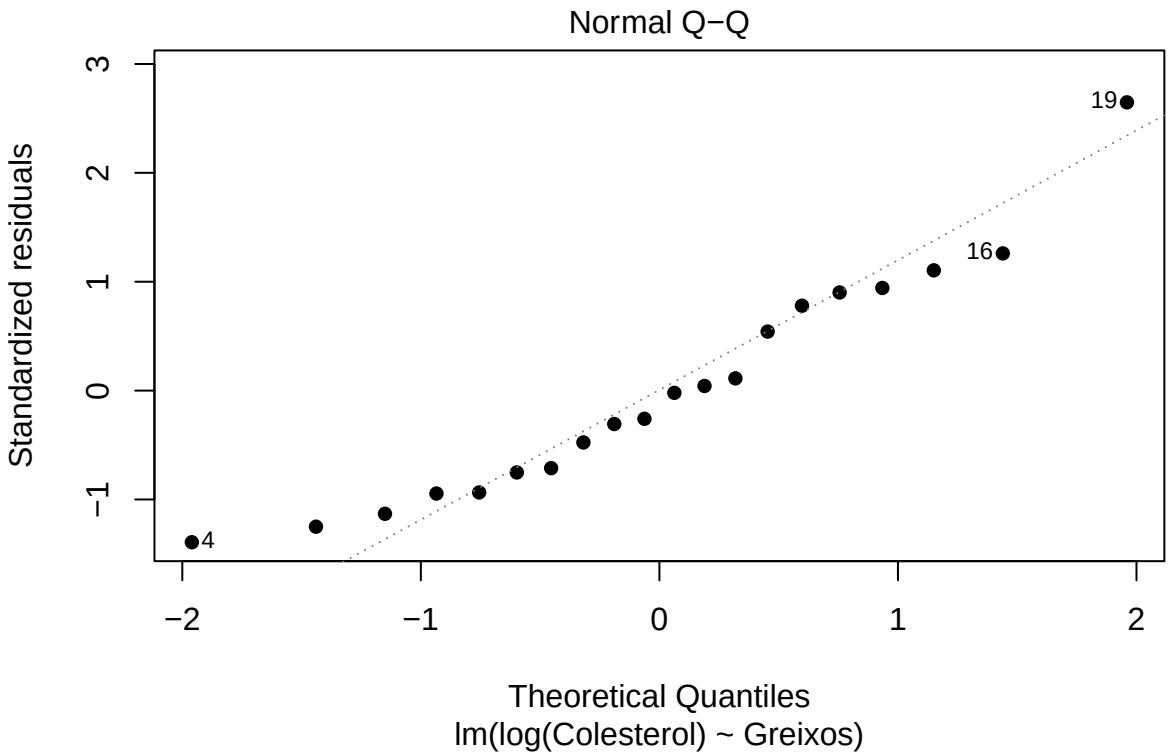
```
plot(m1,which=1,pch=16)
```



It is required that the residuals are placed around the 0 value. Additionally the residuals should not show any trend in relation to the predicted values (red line).

3) Q-Q plot of residuals

```
plot(m1,which=2,pch=16)
```



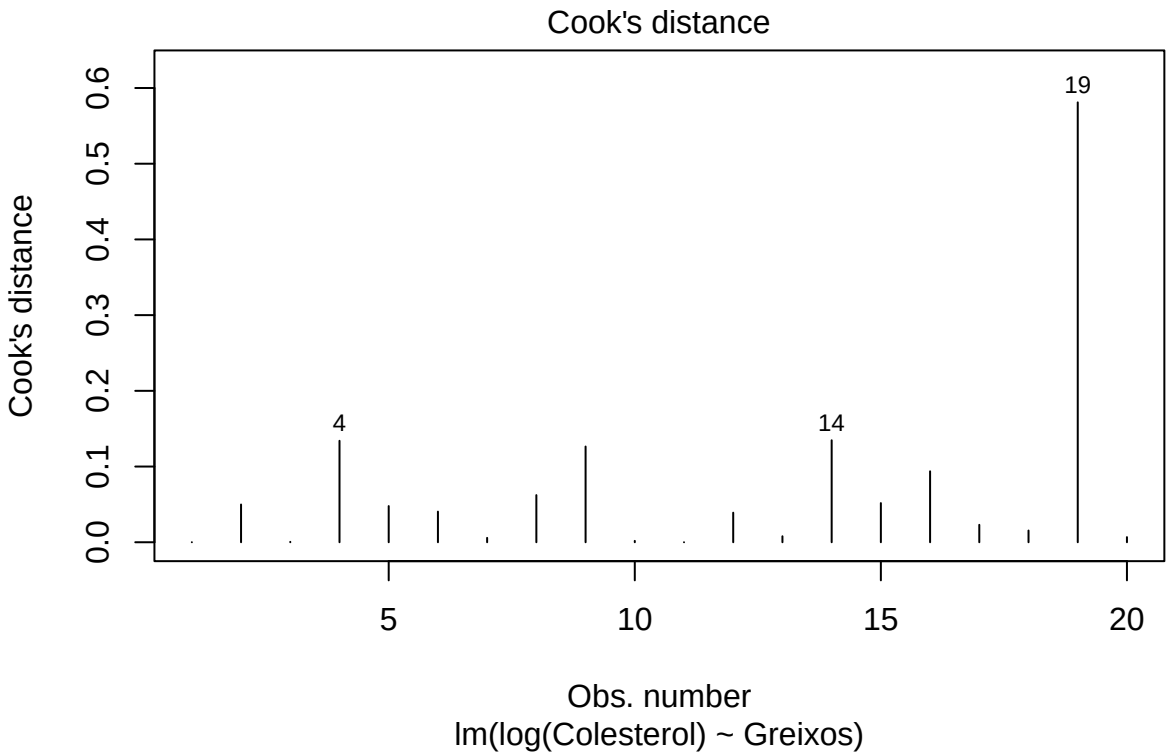
With this plot we can check the assumption of normality of the residuals. We could also apply the Shapiro-Wilks test to the residuals.

```
shapiro.test(resid(m1))
```

```
    Shapiro-Wilk normality test

data:  resid(m1)
W = 0.94343, p-value = 0.2782
```

4) Cook's distance

```
plot(m1,which=4,pch=16)
```



This plot allows checking the presence of influential data by computing the Cook's distance (a statistical distance). R always tags the three more extreme values, that means, the three values with larger distances. However it is our work to decide whether these distances are large enough to really be influential values. A rule of thumb is to consider a value as influential if the Cook's distance is larger than 1. In this case there is no Cook's distance larger than 1, therefore we conclude that there are no influential values.

Prediction

Once a regression model has been fitted it can be interesting to make predictions using hypothetical values for the independent variables.

Predictions are obtained by substituting X in the model by the target value. The estimated model in the example was:

$$Y = 4.66 + 0.024X + \epsilon$$

$$\epsilon \sim N\left(0, 0.134\right)$$

The predictions are the expectation of Y given a specific X:

$$E[Y|X] = 4.66 + 0.024X$$

For example, what is the blood cholesterol level prediction for subjects with a weekly saturated fat intake of 30?

$$E[Y|X = 30] = 4.66 + 0.024 \cdot 30$$

To make a prediction with R using the estimated model we have to proceed by generating a *data.frame* with the values of X. After that it is necessary to apply the function *predict*. Notice the name of the variable has to be exactly the same than that in the model.

```
new.df<-data.frame(Greixos=30)
predict(m1,newdata=new.df)
```

```
      1
5.37301
```

The next step is to give this prediction with an interval with some confidence. There are two options:

- **Confidence interval**. It estimates the confidence interval for the mean of Y given X. In the example the question would be: What is the mean of blood cholesterol in those subjects with a fat intake of 30?

```
new.df<-data.frame(Greixos=30)
predict(m1,newdata=new.df,interval="confidence")
```

```
      fit      lwr     upr
1 5.37301 5.296091 5.44993
```

- **Prediction interval**. It estimates the prediction interval for Y given X. In the example the question would be: What values do the blood cholesterol take in those subjects with a fat intake of 30?

```
new.df<-data.frame(Greixos=30)
predict(m1,newdata=new.df,interval="prediction")
```

```
      fit      lwr      upr
1 5.37301 5.080652 5.665369
```

Sample size

The function *pwr.f2.test(u,f2,power)* from *pwr* package is used to compute the sample size to reject the null hypothesis of $\beta = 0$ with a particular power. The arguments of the function are:

- *u*. Number of slopes in the model. In the simple linear regression case *u=1*.
- *f2*. Effect size. It is assumed that a value of 0.02 involves a small effect, 0.15 involves a medium effect, and a value of 0.35 is a large effect.
- *power*. Power of the test.

Let us compute the sample size necessary to detect a medium effect with a power of 80%.

```
library(pwr)
pwr.f2.test(u=1,f2=0.15,power=0.8)
```

```
     Multiple regression power calculation

              u = 1
              v = 52.315
             f2 = 0.15
      sig.level = 0.05
          power = 0.8
```

The result is 53 subjects.

Multiple linear regression

The outcome (Y) is usually the output of multiple factors rather than only one. The **multiple regression model** allows Y to be modeled by a set of covariables X_j, $j = 1, \ldots, k$. This procedure makes possible to:

1) Improve the prediction of Y.
2) Assess the relative weight of each X_j in the generation of Y.
3) Estimate the effect of a covariable given the remaining covariables.
4) Analyze potential interactions between covariables when generating the outcome. An interaction means that the effect of two covariables on the outcome is multiplicative rather than additive.

The model is:

$$Y = \beta_0 + \beta_1 \cdot X_1 + \ldots + \beta_k \cdot X_k = \boldsymbol{\beta X}$$

Where $\boldsymbol{\beta}$ is the vector of parameters and $\boldsymbol{X}$ is the matrix of covariables.

Example. A research study aimed to model the blood pressure in function of several variables. A sample of 20 subjects was collected and the following variables were measured: age, weight, body surface area, blood pressure, years from hypertension diagnosis, heart rate and a stress measure.

Estimation and inference

Let's load the data.

```
bp=read.table("bloodpress.txt",header=T,sep="\t")
```

The variables in the file are:

- *Pt.* Subject identifier.
- *BP.* Blood pressure.
- *Age.* Age in years.
- *Weight.* Weight in kg.
- *BSA* Body surface area in m^2.
- *Dur.* Years from hypertension diagnosis.

- *Pulse*. Heart rate.
- *Stress*. Stress score.

The function to estimate the multiple linear model is still **lm** but now more covariables are involved in the right side of the formula.

```
model.bp=lm(BP~Age+Weight+BSA+Dur+Pulse+Stress,data=bp)
summary(model.bp)
```

```
Call:
lm(formula = BP ~ Age + Weight + BSA + Dur + Pulse + Stress,
    data = bp)

Residuals:
     Min       1Q   Median       3Q      Max
-0.93213 -0.11314  0.03064  0.21834  0.48454

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -12.870476   2.556650  -5.034 0.000229 ***
Age           0.703259   0.049606  14.177 2.76e-09 ***
Weight        0.969920   0.063108  15.369 1.02e-09 ***
BSA           3.776491   1.580151   2.390 0.032694 *
Dur           0.068383   0.048441   1.412 0.181534
Pulse        -0.084485   0.051609  -1.637 0.125594
Stress        0.005572   0.003412   1.633 0.126491
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4072 on 13 degrees of freedom
Multiple R-squared:  0.9962,    Adjusted R-squared:  0.9944
F-statistic: 560.6 on 6 and 13 DF,  p-value: 6.395e-15
```

In some cases the null hypothesis about the parameter is 0 is not rejected (with a 5% type-I error rate). Specifically for Dur, Pulse and Stress. Let's fit the model taking out these variables.

```
model.bp2=lm(BP~Age+Weight+BSA,data=bp)
summary(model.bp2)
```

```
Call:
lm(formula = BP ~ Age + Weight + BSA, data = bp)
```

```
Residuals:
     Min       1Q   Median       3Q      Max
-0.75810 -0.24872  0.01925  0.29509  0.63030

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -13.66725    2.64664  -5.164 9.42e-05 ***
Age           0.70162    0.04396  15.961 3.00e-11 ***
Weight        0.90582    0.04899  18.490 3.20e-12 ***
BSA           4.62739    1.52107   3.042  0.00776 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.437 on 16 degrees of freedom
Multiple R-squared:  0.9945,    Adjusted R-squared:  0.9935
F-statistic: 971.9 on 3 and 16 DF,  p-value: < 2.2e-16
```

The R^2 is now 99.45% while that one from the former model was 99.62%. Thus, the loss of predictive ability has been very small and the new model is simpler because it is possible to make good predictions of blood pressure just using the age, weight and body surface area.

- **Interpretation of parameters**.

In the simple linear regression model the slope indicated the change in Y if X was changed in one unit. In the multiple linear regression the interpretation is similar but with a condition: the remaining variables must keep constant.

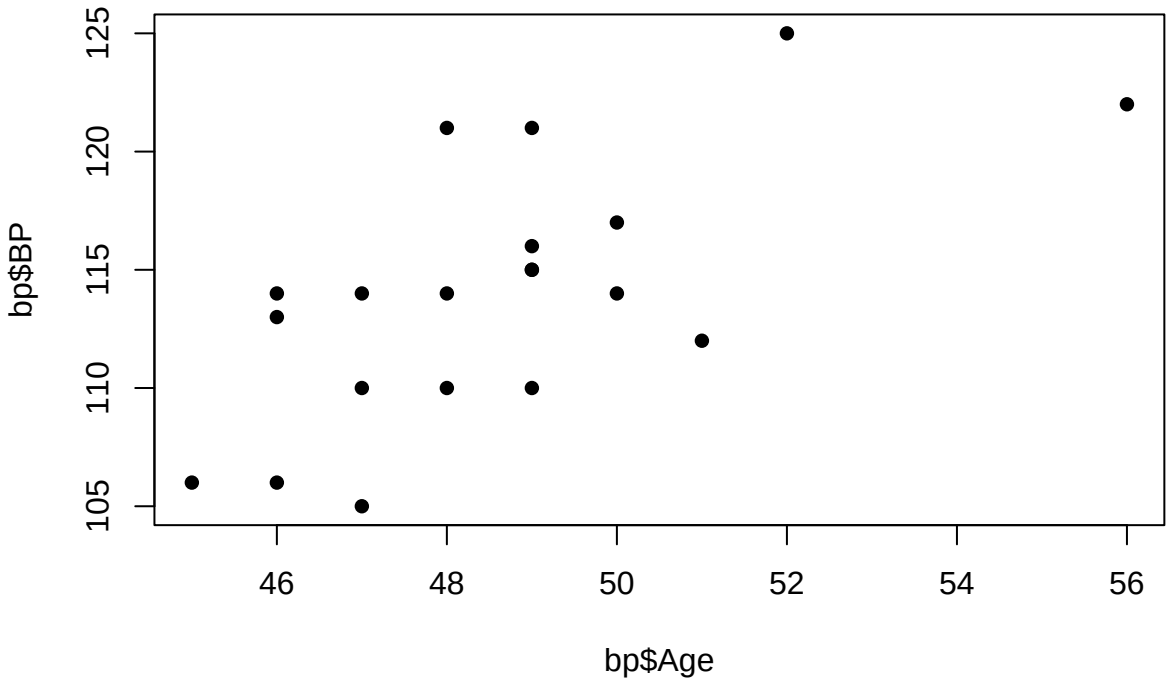
For example, the Age's slope is 0.702. This value involves that blood pressure increases, in average, in 0.702mmHg for every year increased **if the other covariables do not change**.

Model validation

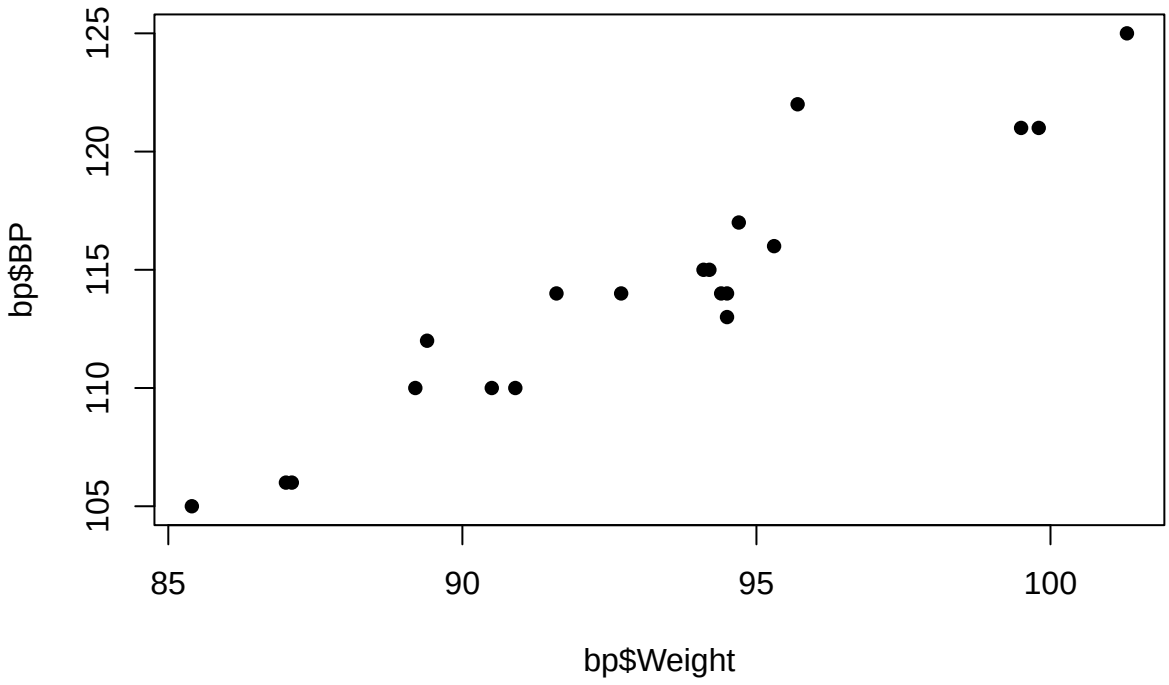
The validity of the model's assumptions will be evaluated by inspection of some plots.

1) Plot the outcome against every covariable to check the assumption of linearity.

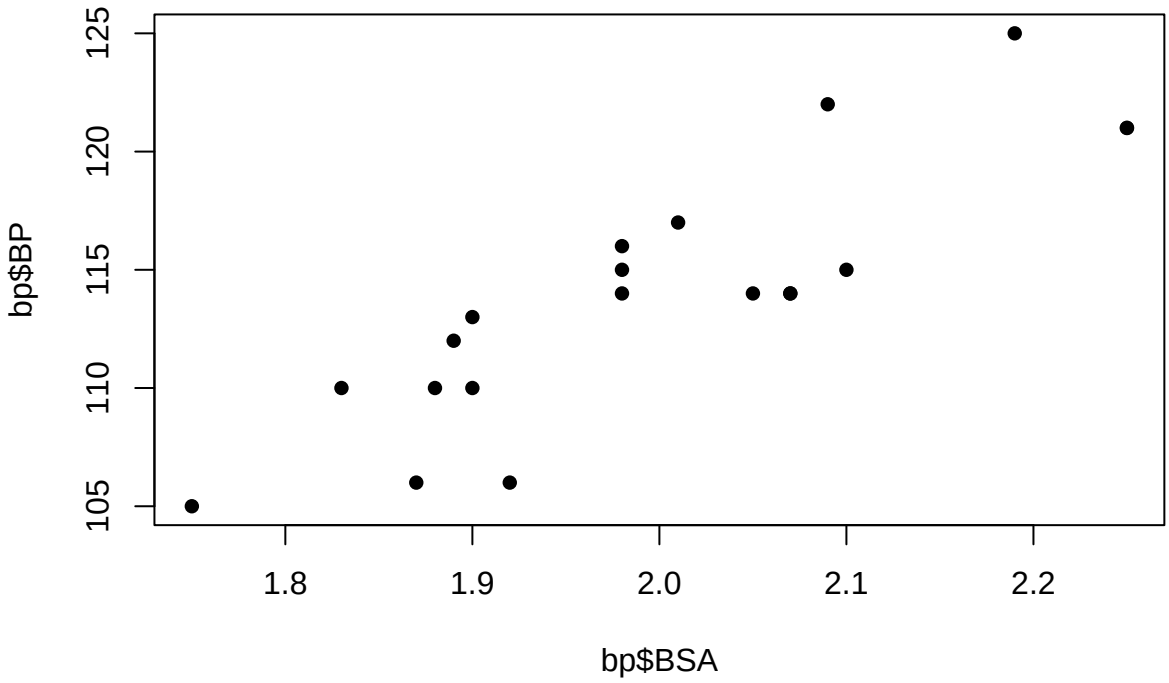
```
plot(bp$Age,bp$BP,pch=16)
```




```
plot(bp$Weight,bp$BP,pch=16)
```

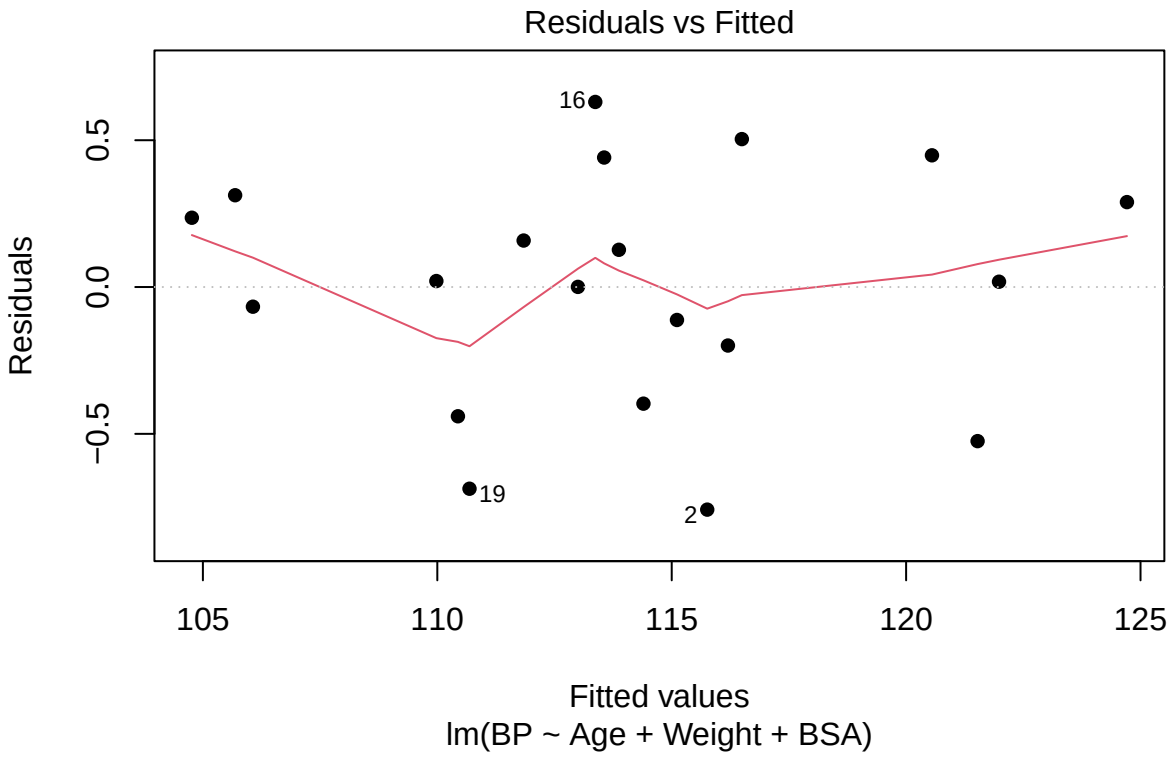



```
plot(bp$BSA,bp$BP,pch=16)
```



2) Plot of residuals and predictions.

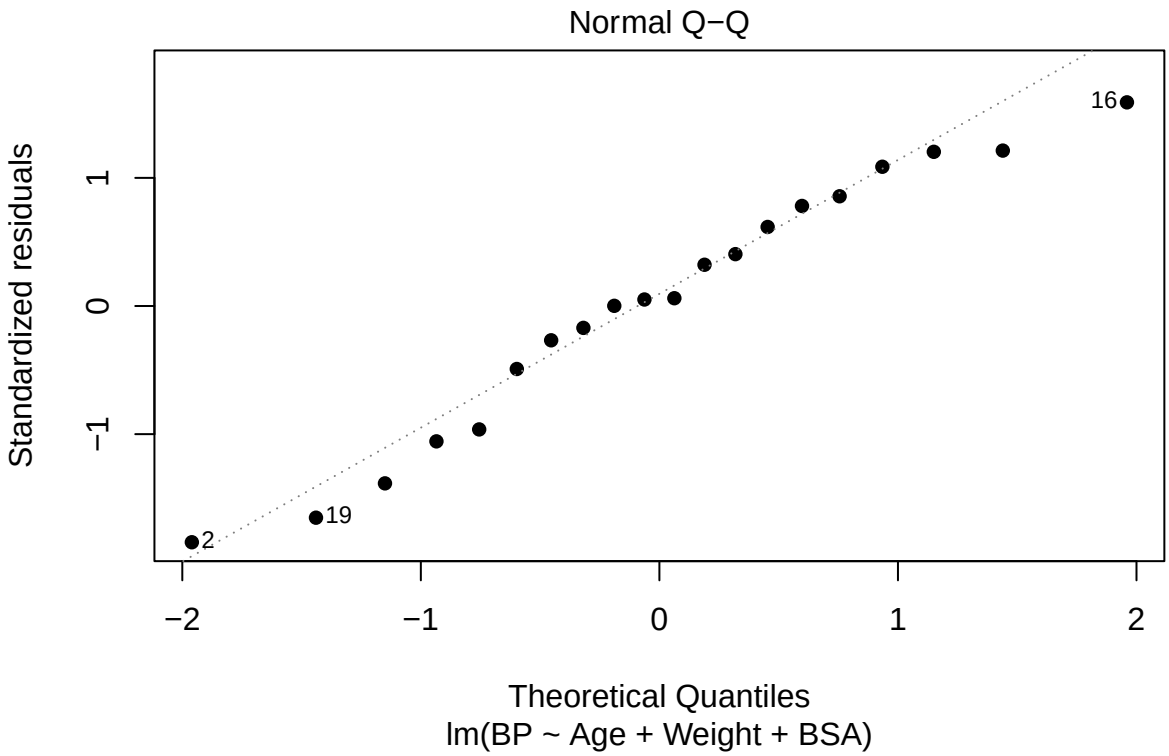
```
plot(model.bp2,which=1,pch=16)
```



It looks like the residuals are placed around the 0 value and do not show any trend in relation to the predicted values (red line).

3) Q-Q plot of residuals

```
plot(model.bp2,which=2,pch=16)
```



The points are quite aligned to the line. If the Shapiro-Wilks test is applied to the residuals.

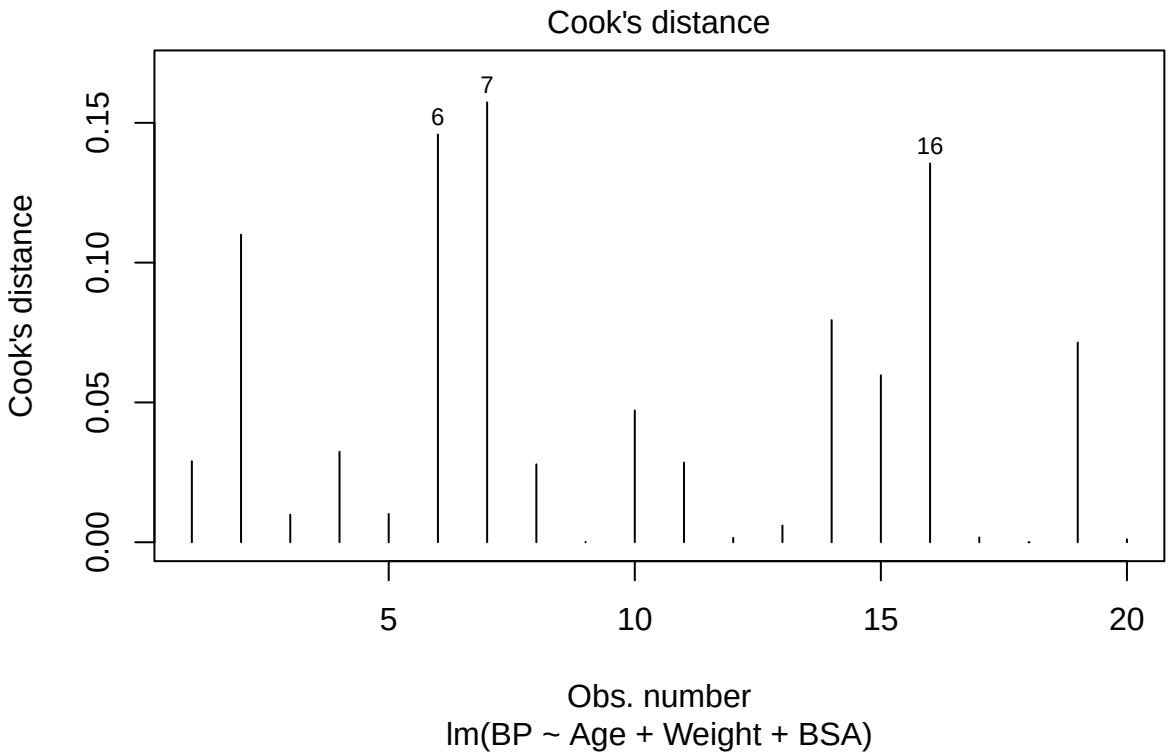
```
shapiro.test(resid(model.bp2))
```

```
    Shapiro-Wilk normality test

data:  resid(model.bp2)
W = 0.96183, p-value = 0.5809
```

It is acceptable to assume the normality of residuals.

4) Cook's distance

```
plot(model.bp2,which=4,pch=16)
```



R marks subjects 6,7 and 16 as those with a larger Cook's distance. However all of them have a Cook's distance lower than 1, so that the conclusion is there is no outlier or influential data.

Confusion and interaction

Confusion

Up to now we have given the same importance to all covariables. However, it would be possible that the research was focused on a concrete variable. For example, we are interested in finding out the relation between the blood pressure and the age. Nevertheless there could be other variables that are related to blood pressure and age. If these variables are not taken into account in the model the estimates of the model can be altered and the true relation between the outcome and the covariable can be masked.

Example. We are interested in evaluating the relation between the blood pressure and the age. A sample of 200 subjects, 100 men and 100 women, have been collected and their blood pressure and age recorded.

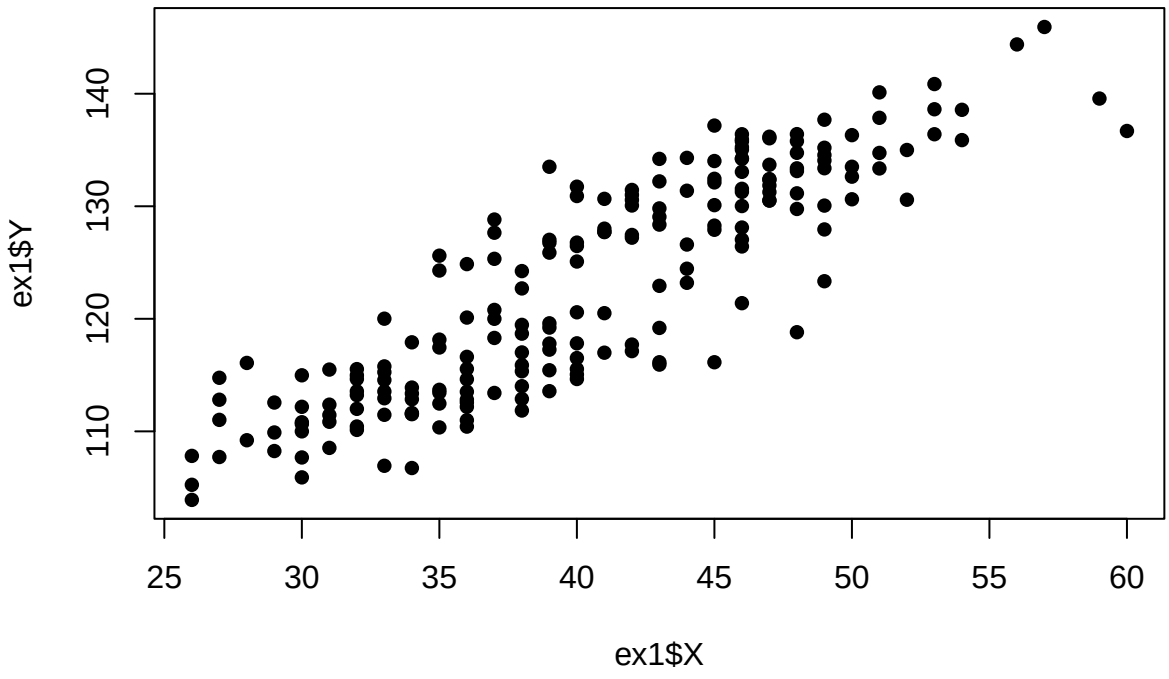
Let's read the data:

```
ex1<-read.table("example1.txt",header=T,sep="\t")
```

In the file Y stands for the blood pressure and X for the age.

Let's plot the two variables

```
plot(ex1$X,ex1$Y,pch=16)
```



The relation seems quite linear.

The model estimates are:

```
model1<-lm(Y~X,data=ex1)
summary(model1)
```

```
Call:
lm(formula = Y ~ X, data = ex1)

Residuals:
    Min      1Q  Median      3Q     Max
-13.160  -3.185   0.242   3.694  12.137
```

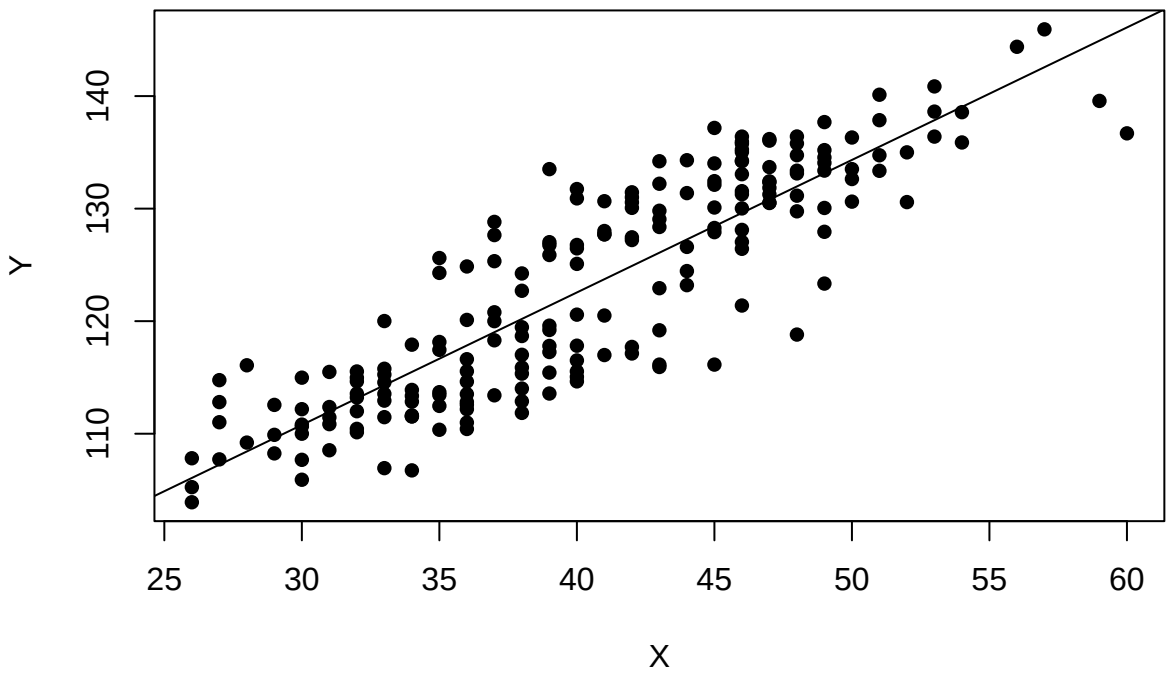
```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 75.45495    1.90741   39.56   <2e-16 ***
X            1.17739    0.04649   25.33   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.784 on 198 degrees of freedom
Multiple R-squared:  0.7641,    Adjusted R-squared:  0.7629
F-statistic: 641.5 on 1 and 198 DF,  p-value: < 2.2e-16
```

Therefore the estimate model is

$$Y = 75.45 + 1.18 \cdot X + \epsilon$$

$$\epsilon \sim N(0, 4.78)$$

```
plot(Y~X,data=ex1,pch=16)
abline(model1)
```



Notice that the slope estimate is 1.18 and the regression line really passes through the cloud of points.

Now, let's add the gender into the model using a *dummy* variable:

$$GenderW = \begin{cases} 1 & \text{if woman} \\ 0 & \text{if man} \end{cases}$$

```
model2<-lm(Y~X+Gender,data=ex1)
summary(model2)
```

```
Call:
lm(formula = Y ~ X + Gender, data = ex1)

Residuals:
   Min     1Q Median     3Q    Max
-7.942 -1.948 -0.284  2.201  7.165

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 91.71858    1.47302   62.27   <2e-16 ***
X            0.64021    0.04108   15.59   <2e-16 ***
GenderW     10.85549    0.59782   18.16   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.933 on 197 degrees of freedom
Multiple R-squared:  0.9118,    Adjusted R-squared:  0.9109
F-statistic:  1018 on 2 and 197 DF,  p-value: < 2.2e-16
```

The estimated model is

$$Y = 91.72 + 0.64 \cdot X + 10.86 \cdot GenderW + \epsilon$$

$$\epsilon \sim N(0, 2.93)$$

Actually that implies that there are two intercepts and one slope.

- Model for woman ($GenderW = 1$).

$$Y = 91.72 + 0.64 \cdot X + 10.86 \cdot 1 + \epsilon = 102.58 + 0.64 \cdot X + \epsilon$$

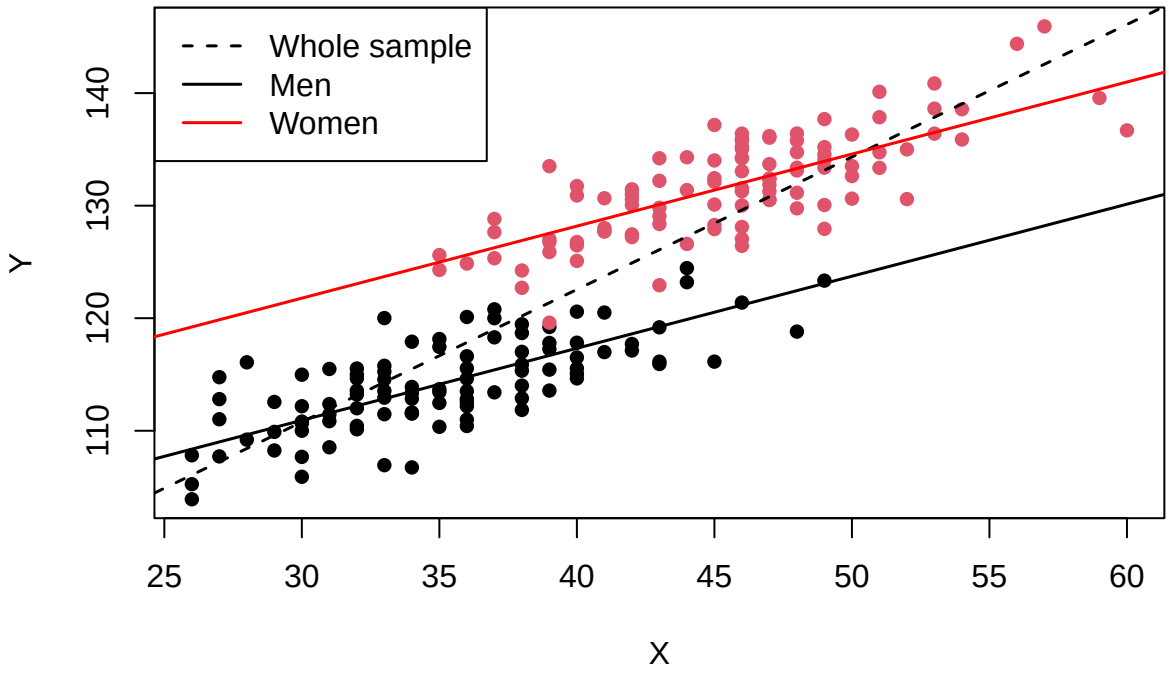
- Model for man ($GenderW = 0$).

$$Y = 91.72 + 0.64 \cdot X + 10.86 \cdot 0 + \epsilon = 91.72 + 0.64 \cdot X + \epsilon$$

The most important difference between the two models is the change in the slope. When the Gender is included in the model the slope gets down, it means the strength of the relation between the blood pressure and the age is lower when such a relation is controlled by gender. Actually, the second model's slope is known as **adjusted slope** or slope adjusted by gender.

Graphically,

```
plot(Y~X,data=ex1,pch=16,col=as.factor(ex1$Gender))
abline(model1,lty=2,lwd=1.5)
abline(a=coef(model2)[1],b=coef(model2)[2],lty=1,lwd=1.5)
abline(a=coef(model2)[1]+coef(model2)[3],b=coef(model2)[2],lwd=1.5,col="red")
legend("topleft",legend=c("Whole sample","Men","Women"),lty=c(2,1,1),
       lwd=rep(1.5,3),col=c("black","black","red"))
```



Interaction

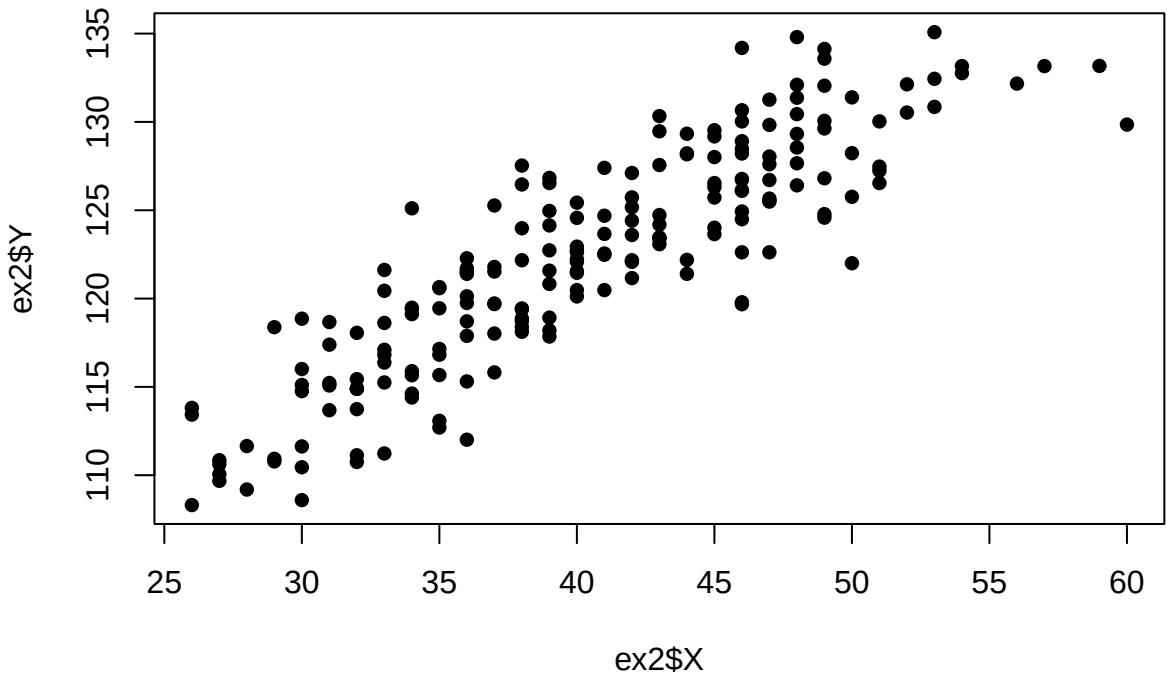
Interaction means that the effect of the variable on the outcome is modified by a third variable. Let's follow with the example of the blood pressure, age and gender. However, we are going to use a different dataset.

```
ex2<-read.table("example2.txt",header=T,sep="\t")
```

In the file Y stands for the blood pressure and X for the age.

Let's plot the two variables

```
plot(ex2$X,ex2$Y,pch=16)
```



The relation seems quite linear.

The model estimates are:

```
model1<-lm(Y~X,data=ex2)
summary(model1)
```

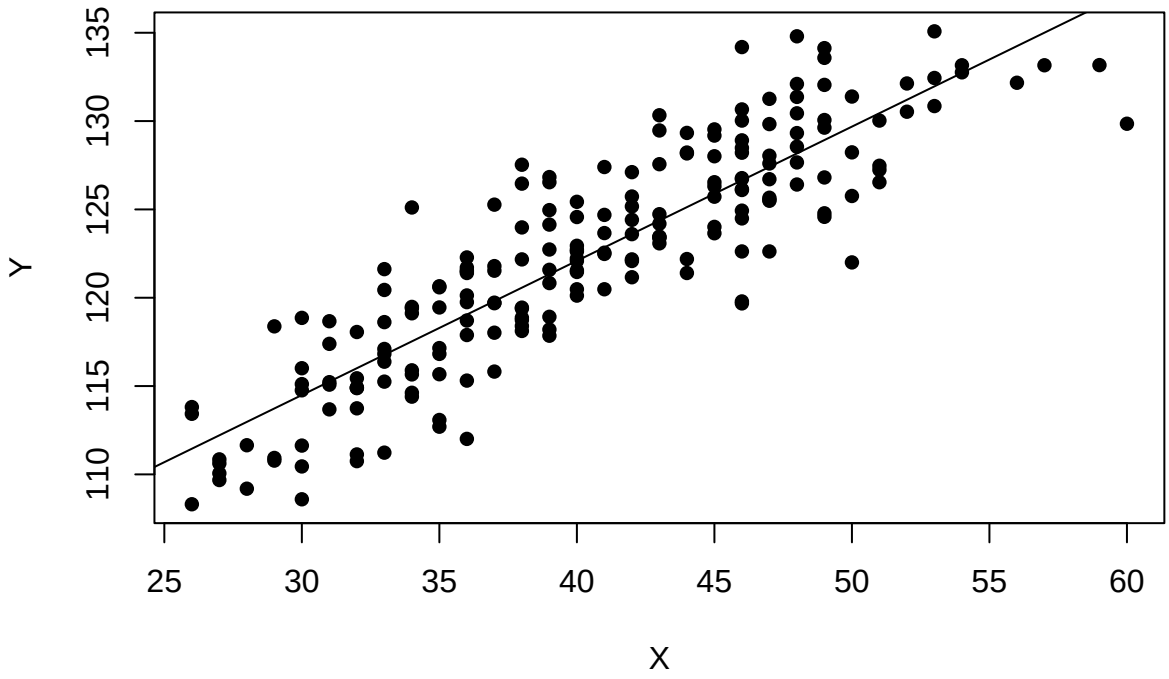
```
Call:
lm(formula = Y ~ X, data = ex2)

Residuals:
    Min      1Q  Median      3Q     Max
-7.6862 -1.8718 -0.1042  2.1221  7.5801

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 91.69788    1.20044   76.39   <2e-16 ***
X            0.75977    0.02926   25.97   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.011 on 198 degrees of freedom
Multiple R-squared:  0.773, Adjusted R-squared:  0.7719
F-statistic: 674.4 on 1 and 198 DF,  p-value: < 2.2e-16
```

Therefore the estimate model is

$$Y = 91.70 + 0.76 \cdot X + \epsilon$$

$$\epsilon \sim N(0, 3.01)$$

```
plot(Y~X,ex2,pch=16)
abline(model1)
```



Notice that the slope estimate is 0.76 and the regression line really passes through the cloud of points.

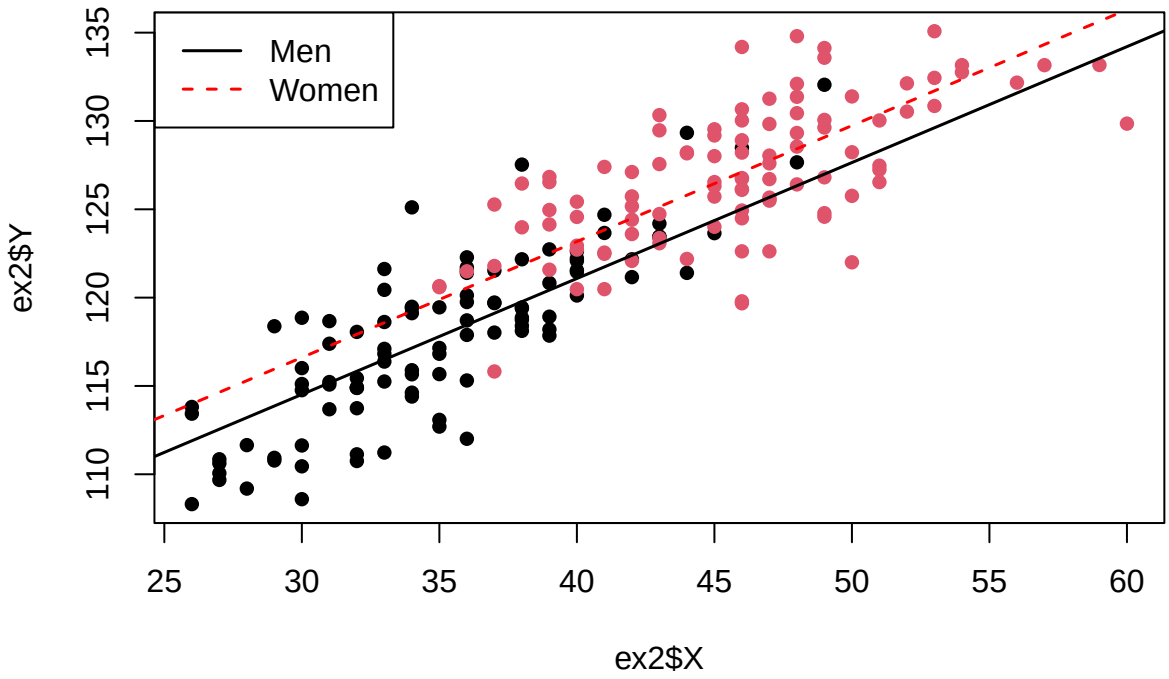
Now, let's add the gender into the model using a *dummy* variable as in the *confusion* case example:

```
model2<-lm(Y~X+Gender,ex2)
summary(model2)
```

```
Call:
lm(formula = Y ~ X + Gender, data = ex2)

Residuals:
   Min     1Q Median     3Q    Max
-7.736 -1.951 -0.086  2.028  7.965

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 94.83166    1.47075  64.479  < 2e-16 ***
X            0.65626    0.04102  16.000  < 2e-16 ***
GenderW      2.09171    0.59689   3.504 0.000567 ***
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.928 on 197 degrees of freedom
Multiple R-squared:  0.7863,    Adjusted R-squared:  0.7842
F-statistic: 362.5 on 2 and 197 DF,  p-value: < 2.2e-16
```

```
plot(ex2$X,ex2$Y,pch=16,col=as.factor(ex1$Gender))
abline(a=coef(model2)[1],b=coef(model2)[2],lty=1,lwd=1.5)
abline(a=coef(model2)[1]+coef(model2)[3],b=coef(model2)[2],lty=2,lwd=1.5,col="red")
legend("topleft",legend=c("Men","Women"),lty=c(1,2),
       lwd=rep(1.5,2),col=c("black","red"))
```



The estimated model is

$$Y = 94.83 + 0.66 \cdot X + 2.09 \cdot GenderW + \epsilon$$

$$\epsilon \sim N(0, 2.93)$$

Actually that implies that there are two intercepts and one slope.

- Model for woman ($GenderW = 1$).

$$Y = 94.83 + 0.66 \cdot X + 2.09 \cdot 1 + \epsilon = 96.92 + 0.66 \cdot X + \epsilon$$

- Model for man ($GenderW = 0$).

$$Y = 94.83 + 0.66 \cdot X + 2.09 \cdot 0 + \epsilon = 94.83 + 0.66 \cdot X + \epsilon$$

The slope has changed in relation to that from the first model. So we could conclude that gender is a confusing factor of the relation between blood pressure and age. However, it could happen that gender was not a merely confusing factor and it interacts in the relation between blood pressure and age. The model to fit the age-gender interaction effect is:

$$Y = \beta_0 + \beta_1 \cdot X + \beta_2 \cdot GenderW + \beta_3 \cdot X \cdot GenderW + \epsilon$$

Let's fit now a model with this interaction.

```
model3<-lm(Y~X+Gender+X*Gender,ex2)
summary(model3)
```

```
Call:
lm(formula = Y ~ X + Gender + X * Gender, data = ex2)

Residuals:
    Min      1Q  Median      3Q     Max
-7.3773 -1.9177 -0.2391  2.0035  8.1328

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 89.67536    2.01432  44.519  < 2e-16 ***
X            0.80300    0.05675  14.150  < 2e-16 ***
GenderW     13.73138    3.26032   4.212 3.86e-05 ***
X:GenderW   -0.28885    0.07962  -3.628 0.000365 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.842 on 196 degrees of freedom
Multiple R-squared:  0.7998,    Adjusted R-squared:  0.7967
F-statistic:   261 on 3 and 196 DF,  p-value: < 2.2e-16
```

Notice that the interaction effect is referred in the output as *X:GenderW*. Remember that the P-values in the last column are linked to the test that the parameter is 0. The interaction's p-value is lower than 5% therefore the null hypothesis is rejected and we can affirm that there is a significant interaction.

The estimated model is

$$Y = 89.68 + 0.8 \cdot X + 13.73 \cdot GenderW - 0.29 \cdot X \cdot GenderW + \epsilon$$

Actually that implies that there are two intercepts and one slope.

- Model for woman ($GenderW = 1$).

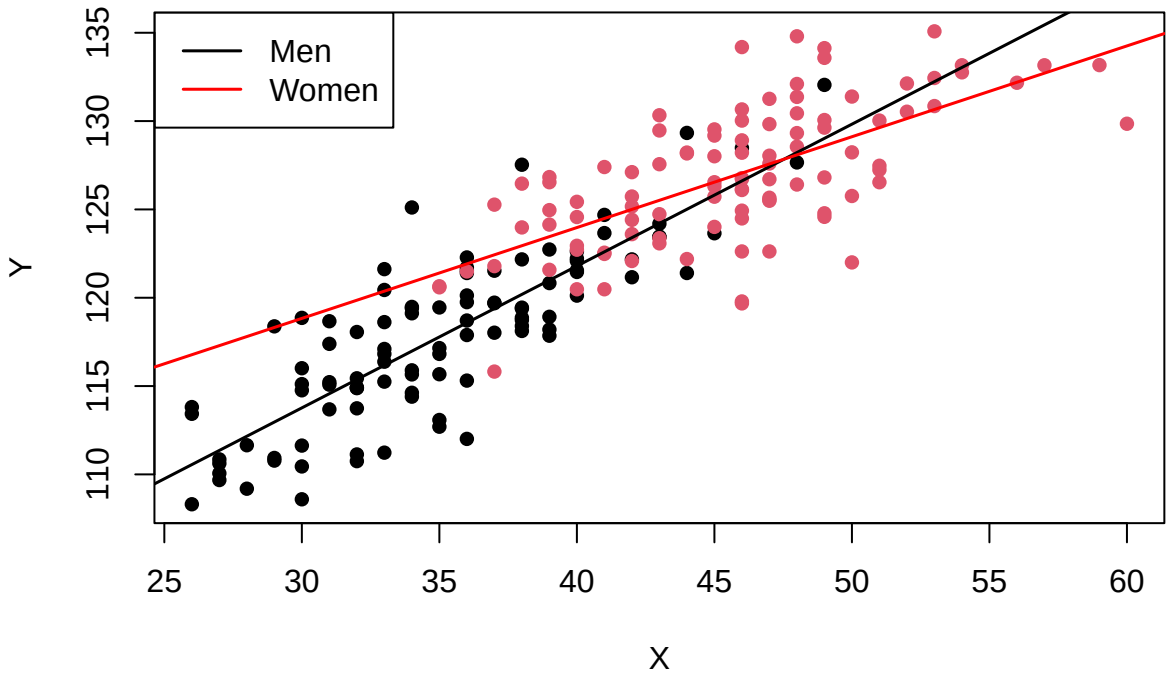
$$Y = 89.68 + 0.8 \cdot X + 13.73 \cdot 1 - 0.29 \cdot X \cdot 1 + \epsilon = 103.4 + 0.51 \cdot X + \epsilon$$

- Model for man ($GenderW = 0$).

$$Y = 89.68 + 0.8 \cdot X + 13.73 \cdot 0 - 0.28 \cdot X \cdot 0 + \epsilon = 89.68 + 0.8 \cdot X + \epsilon$$

Notice that the slopes are different, therefore the relation between blood pressure and age is different depending on the gender. Specifically the men's slope is steeper, in other words the relation between blood pressure and age is stronger in men. Graphically:

```
plot(Y~X,data=ex2,pch=16,col=as.factor(ex1$Gender))
abline(a=coef(model3)[1],b=coef(model3)[2],lwd=1.5)
abline(a=coef(model3)[1]+coef(model3)[3],b=coef(model3)[2]+coef(model3)[4],
       lwd=1.5,col="red")
legend("topleft",legend=c("Men","Women"),lwd=rep(1.5,3),col=c("black","red"))
```



Sample size

To compute the sample size we can use the same function as that in the simple linear regression.

For example, let us compute the sample size necessary to detect a medium effect with a power of 80% in a multiple regression with 4 covariates.

```
library(pwr)
pwr.f2.test(u=4,f2=0.15,power=0.8)
```

```
     Multiple regression power calculation

              u = 4
              v = 79.44992
             f2 = 0.15
      sig.level = 0.05
          power = 0.8
```

The result is 80 subjects.

Logistic regression

Introduction

The aim of the logistic regression is to model binary data.

Recall that the linear regression model has the following characteristics:

- The outcome Y is continuous.
- The covariates X can be either factors or quantitative variables.
- The relation between Y and X is linear.
- The probability distribution of Y (conditioned to X, i.e. the random error) is a Normal model.

So that, if the outcome is binary (success/failure) as "Positive response to treatment" the linear regression is not an appropriate approach because:

- The outcome Y is binary rather than continuous.
- The relation between Y and X is often not linear.
- The probability distribution of Y (conditioned to X, i.e. the random error) is no longer a Normal model. It is more suitable to use a Bernoulli model.

Therefore, to model binary data it is necessary to adapt the regression model to afford these new characteristics.

The model

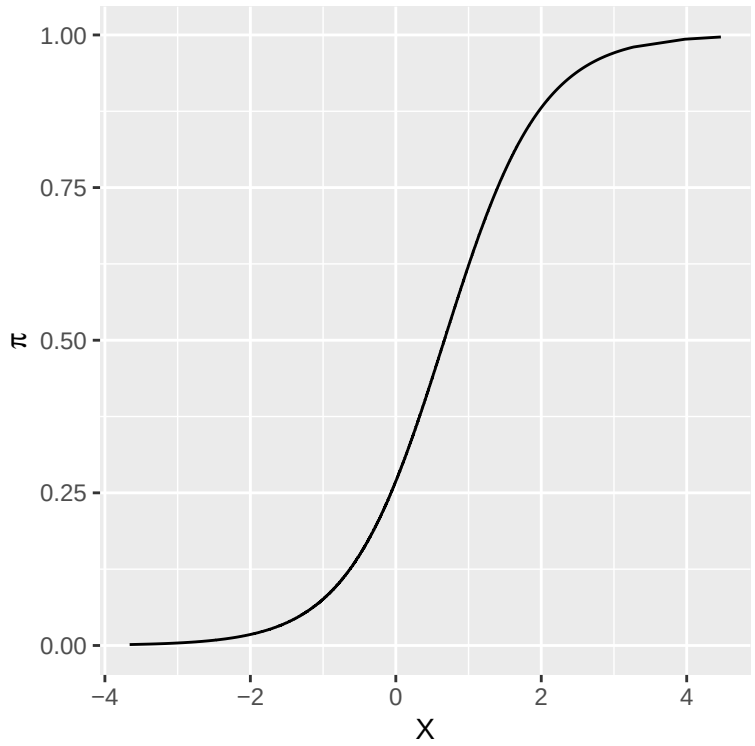
The logistic regression assumes the probability distribution of Y conditioned to X is a Bernoulli model with parameter π (the probability of success). Additionally it assumes that the relation between π and the covariates X is the *logit* function:

$$logit\left(\pi\right) = log\left(\frac{\pi}{1-\pi}\right) = \beta_0 + \beta_1 X$$

Or equally

$$\pi = \frac{exp\left(\beta_0 + \beta_1 X\right)}{1 + exp\left(\beta_0 + \beta_1 X\right)}$$

Graphically the *logit* function looks as this:



Notice that π increase slower when taking extreme values (close to 0 or 1) and faster with medium values (around 0.5). Furthermore it can not take values out of the range between 0 and 1.

Interpretation of the slope

Let's suppose that we are assessing the efficacy of a new treatment for a specific pathology. The aim is to evaluate if the probability of get cured is different depending on the treatment received (new or old).

Let's X be a binary covariate indicating if a subject has received the new treatment. X is codified as follows:

$$X = \begin{cases} 1 & \text{New treament} \\ 0 & \text{Old treatment} \end{cases}$$

The logit of π (probability of get cured) for subjects receiving the new treatment is:

$$log\left(\frac{\pi_{x=1}}{1-\pi_{x=1}}\right) = \beta_0 + \beta_1$$

On the other hand, the logit of π for subjects treated with the old treatment is

$$log\left(\frac{\pi_{x=0}}{1-\pi_{x=0}}\right) = \beta_0$$

Therefore, the β_1 parameter (slope) is equal to the difference of logits.

$$log\left(\frac{\pi_{x=1}}{1-\pi_{x=1}}\right) - log\left(\frac{\pi_{x=0}}{1-\pi_{x=0}}\right) = \beta_1$$

The difference of logits can be expressed as

$$\beta_1 = log\left(\frac{\pi_{x=1}}{1-\pi_{x=1}}\right) - log\left(\frac{\pi_{x=0}}{1-\pi_{x=0}}\right) = log\left(\frac{\frac{\pi_{x=1}}{1-\pi_{x=1}}}{\frac{\pi_{x=0}}{1-\pi_{x=0}}}\right) = log(OR)$$

where OR stand for the odds ratio. Thus, the parameter β_1 is the logarithm of the odds ratio (OR).

Remember that OR is a measure of association an it is interpreted as:

$$OR \begin{cases} < 1 & \text{Success is associated to } X = 0 \\ = 1 & \text{There is no association between success and X} \\ > 1 & \text{Success is associated to } X = 1 \end{cases}$$

If X was **quantitative** the slope would still be the logarithm of the odds ratio (OR), but the two values of X involved in the OR would be those just separated by one unit. Therefore the slope in this case is interpreted as the change on the association (OR) in front an increase of one unit in X.

Case example

Example. In a specific study, it was desired to evaluate the efficacy of two analgesic treatments (A and B) prescribed for patients with neuralgia. Sixty patients were recruited and randomly received one of the two treatments (A and B) or a placebo (P). Once the treatment was completed, the response was collected as *pain* (0) or *no pain* (1). Additional information that was collected is sex, age and time after the diagnosis of the disease.

Let's read the data.

```
neur<-read.table("neuralgia.txt",header=T)
```

The model

The aim is to model the probability of pain absence as a function of the treatment, gender, age and time from onset of the disease. Given that the outcome is binary a logistic model will be applied.

Firstly we have to bear in mind that the treatment variable has three levels (A,B and P). To include them in the model it will be necessary to use two **dummy** variables. The Placebo level will be defined as the baseline.

$$D_1 = \begin{cases} 1 & \text{if treatment=A} \\ 0 & \text{otherwise} \end{cases}$$

$$D_2 = \begin{cases} 1 & \text{if treatment=B} \\ 0 & \text{otherwise} \end{cases}$$

Thus the model with the treatment effect would be

$$logit\,(\pi) = \beta_0 + \beta_1 D_1 + \beta_2 D_2$$

Remember that *logit* is the logarithm of odds of π, and π is the probability of no pain.

Using this codification the dummy variables values for a subject from treatment A group are $(D_1 = 1, D_2 = 0)$. In case of treatment B are $(D_1 = 0, D_2 = 1)$. Finally the combination for Placebo group is $(D_1 = 0, D_2 = 0)$.

Thus, the logit for treatment A is

$$logit\,(\pi|Tr = A) = \beta_0 + \beta_1$$

The logit for treatment B is

$$logit\,(\pi|Tr = B) = \beta_0 + \beta_2$$

and that one of the Placebo group is

$$logit\,(\pi|Tr = P) = \beta_0$$

That implies β_1 and β_2 will be the logarithm of A/P and B/P odds ratios respectively. Furthermore, the logarithm of A/B odds ratio could be computed as $\beta_1 - \beta_2$.

Finally, the remaining variables are included in the model:

$$logit\,(\pi) = \beta_0 + \beta_1 D_1 + \beta_2 D_2 + \beta_3 Sex + \beta_4 Duration + \beta_5 Age$$

Estimating the model

In the process of estimating the logistic model, R will automatically build the *dummy* variables. However, we must verify that R really creates those *dummies* that we want.

First of all we should verify that the variable *Treatment* is a factor. However, in this case this verification is unnecessary since the variable is encoded alphanumerically, and you just have to do it when a qualitative variable is encoded with numbers.

Next question is about whether R is putting the *Placebo* level as the baseline category. We need to run the *contrasts* function.

```
neur$Treatment<-factor(neur$Treatment)
contrasts(neur$Treatment)
```

```
  B P
A 0 0
B 1 0
P 0 1
```

Every column is one of the *dummy* variables. So that R decided to assign the treatment A as the baseline. The reason of that is R use the alphabetical order to do that. We should use the *contr.treatment()* function to assign the *Placebo* as the baseline category.

```
contr.treatment(3,base=3)
```

```
  1 2
1 1 0
2 0 1
3 0 0
```

Notice that the value "3" in the function indicates that we have three levels, so R will built two dummies. The argument *base=3* means that the third category (in alphabetical order) must be selected as the baseline.

Finally it is necessary to assign the output of this function the the *contrasts* of treatment.

```
contrasts(neur$Treatment)<-contr.treatment(3,base=3)
contrasts(neur$Treatment)
```

```
  1 2
A 1 0
B 0 1
P 0 0
```

Now the Placebo level is the baseline category.

Let's estimate the model.

```
logres<-glm(Pain~Treatment+Sex+Duration+Age,data=neur,binomial)
summary(logres)
```

```
Call:
glm(formula = Pain ~ Treatment + Sex + Duration + Age, family = binomial,
    data = neur)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3153  -0.6151   0.1952   0.5904   2.7638

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) 17.406592   6.690857   2.602  0.00928 **
Treatment1   3.181690   1.016021   3.132  0.00174 **
Treatment2   3.708542   1.140577   3.251  0.00115 **
SexM        -1.832202   0.796206  -2.301  0.02138 *
Duration     0.005859   0.032992   0.178  0.85905
Age         -0.262093   0.097012  -2.702  0.00690 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 81.503  on 59  degrees of freedom
Residual deviance: 48.736  on 54  degrees of freedom
AIC: 60.736

Number of Fisher Scoring iterations: 5
```

The output table is similar to that one we saw in the linear regression. Under the columns *Estimate* and *Std. Error* appear the parameter estimates and their standard errors. The column *Pr (> | z |)* are the p-values related to the hypotheses testing of the parameters are equal to 0. It is observed that all the null hypotheses are rejected, unless the one referred to the duration of the disease.

We could be interested in making a global test on the all parameters of a variable instead of proceed parameter to parameter as in the table. This would be the case of the treatment variable where two parameters have been estimated. Thus, instead of evaluating whether $\beta_1 = 0$ and $\beta_2 = 0$ separately, we would test the global null hypothesis

$$H_0 : \beta_1 = \beta_2 = 0$$

The alternative hypothesis is that one parameter is different to 0 at least.

To perform this contrast we must execute

```
anova(logres,test="Chisq")
```

```
Analysis of Deviance Table

Model: binomial, link: logit

Response: Pain

Terms added sequentially (first to last)

          Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                         59     81.503
Treatment  2  14.0230        57     67.480 0.0009015 ***
Sex        1   7.5945        56     59.886 0.0058545 **
Duration   1   0.6731        55     59.213 0.4119823
Age        1  10.4769        54     48.736 0.0012088 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The P-value for *Treatment* is lower than 0.05, so the hypothesis that $\beta_1 = \beta_2 = 0$ is rejected.

Odds ratio estimation

As it has been stated before, the model slopes are the logarithm of the odds ratios. Therefore, we just need to apply the antilogarithm transformation to the model estimates to get the odds ratio estimates.

```
coef(logres)
```

```
 (Intercept)   Treatment1   Treatment2         SexM     Duration          Age
17.406592249  3.181689786  3.708542447 -1.832202124  0.005858679 -0.262093311
```

```
round(exp(coef(logres)),digits=3)
```

```
 (Intercept)   Treatment1   Treatment2         SexM     Duration          Age
36273291.215       24.087       40.794        0.160        1.006        0.769
```

To estimate the odds ratio confidence intervals we must run

```
confint(logres)
```

```
                 2.5 %      97.5 %
(Intercept)  5.67268114 32.45810193
Treatment1   1.38534008  5.45845778
Treatment2   1.72693142  6.29309634
SexM        -3.58627713 -0.38227954
Duration    -0.05768555  0.07338820
Age         -0.48146218 -0.09349231
```

```
round(exp(confint(logres)),digits=3)[-1,]
```

```
           2.5 %  97.5 %
Treatment1 3.996 234.735
Treatment2 5.623 540.825
SexM       0.028   0.682
Duration   0.944   1.076
Age        0.618   0.911
```

Let's put together the estimates and the confidence intervals.

```
myOR=cbind(coef(logres),confint(logres))
round(exp(myOR),digits=3)[-1,]
```

```
                  2.5 %  97.5 %
Treatment1 24.087 3.996 234.735
Treatment2 40.794 5.623 540.825
SexM        0.160 0.028   0.682
Duration    1.006 0.944   1.076
Age         0.769 0.618   0.911
```

The treatment odds ratios are greater than 1. That means the absence of pain is associated to the treatment instead of the placebo. In other words, the treatments work better than the placebo to relief the pain.

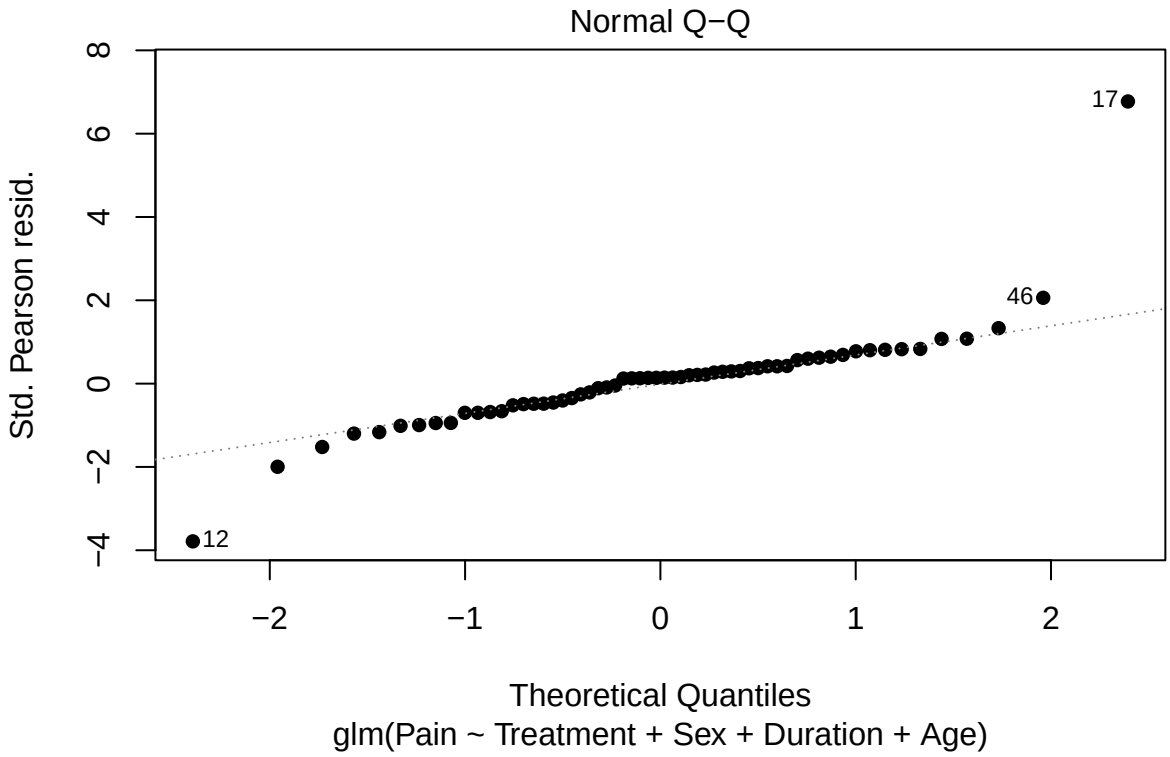
Additionally, treatment A is 24.1 times more associated to the absence of pain than the Placebo. This figure is 40.8 in case of treatment B.

Validation of the model

The validation will involve checking the normality of the residuals and the presence of outliers.

- Residuals Q-Q plot

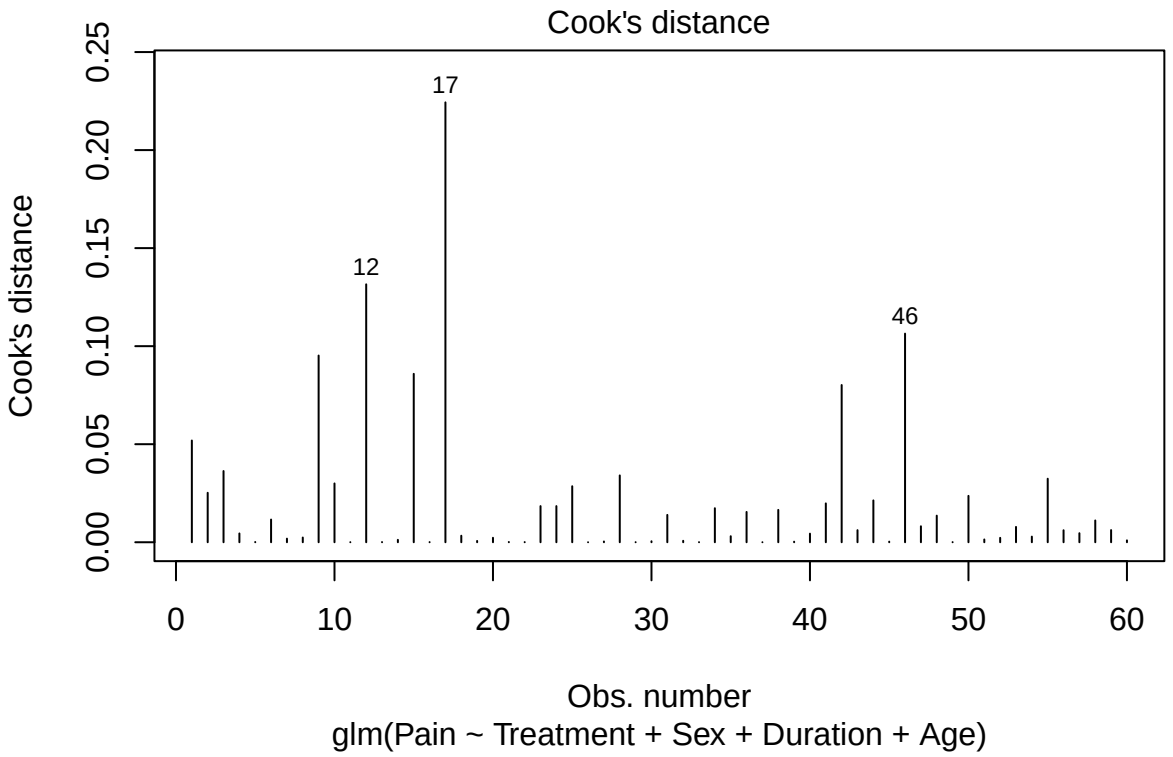
```
plot(logres,which=2,pch=16)
```



Residuals are well aligned on the concordance line except for the cases of subjects 12 and 17. These residuals could be influential values. Let us check that with the Cook's distance.

- Cook's distance

```
plot(logres,which=4,pch=16)
```



there is no distance greater than 1, so we conclude that there are no outliers (influential points).

Sample size

Let us use the function *wp.logistic(p0,p1,power)* from *WebPower* package to compute the sample size. Here, it will assumed that we wish to test one slope (beta) from a binary covariate (X) that takes values 0 and 1.

The arguments of the function are:

- *p0*. Probability of Y=1 when X=0.
- *p1*. Probability of Y=1 when X=1.
- *power*. Power of the test.

As example suppose we want to detect as significant with a power of 80% the following setting:

$$P\left(Y = 1|X = 0\right) = 0.25$$

$$P(Y = 1|X = 1) = 0.75$$

Notice that the implicit odds ratio here is:

$$Odds_{X=0} = \frac{0.25}{0.75} = \frac{1}{3}$$
$$Odds_{X=1} = \frac{0.75}{0.25} = 3$$

$$OR = \frac{Odds_{X=1}}{Odds_{X=0}} = 9$$

```
library(WebPower)
wp.logistic(p0=0.25,p1=0.75,power=0.8)
```

```
Power for logistic regression

      p0   p1     beta0    beta1        n alpha power
    0.25 0.75 -1.098612 2.197225 34.68299  0.05   0.8

URL: http://psychstat.org/logistic
```

The result is 35 subjects.