

Regression Models in Systems Biology with R

Part I: Simple Linear Regression

Uwe Menzel 2014

www.matstat.org

Outline

1. Simple Linear Regression

1. The statistics behind the output of "**lm**"

2. General Linear Model

1. Continuous and categorical variables mixed, "**lm**"
2. Interaction

3. Generalized Linear Model

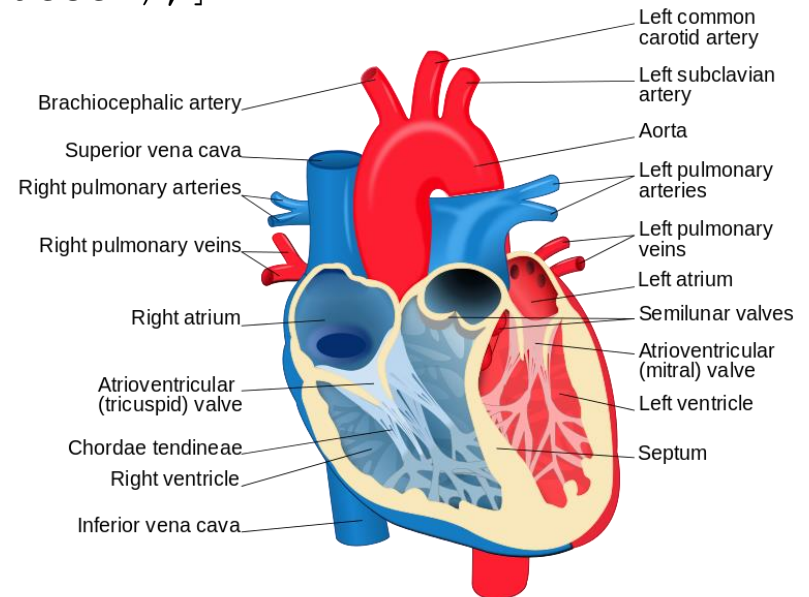
1. Logistic Regression - "**glm**"
2. Multinomial Regression - "**multinom**"

1. The Simple Linear Regression Model

- May not be useful in many cases
- ... but can be used to explain how regression works
- see [Reg_Models_Examples.R](#)

```
data(thuesen, package = "ISwR")
thuesen = thuesen[complete.cases(thuesen), ]
head(thuesen)
```

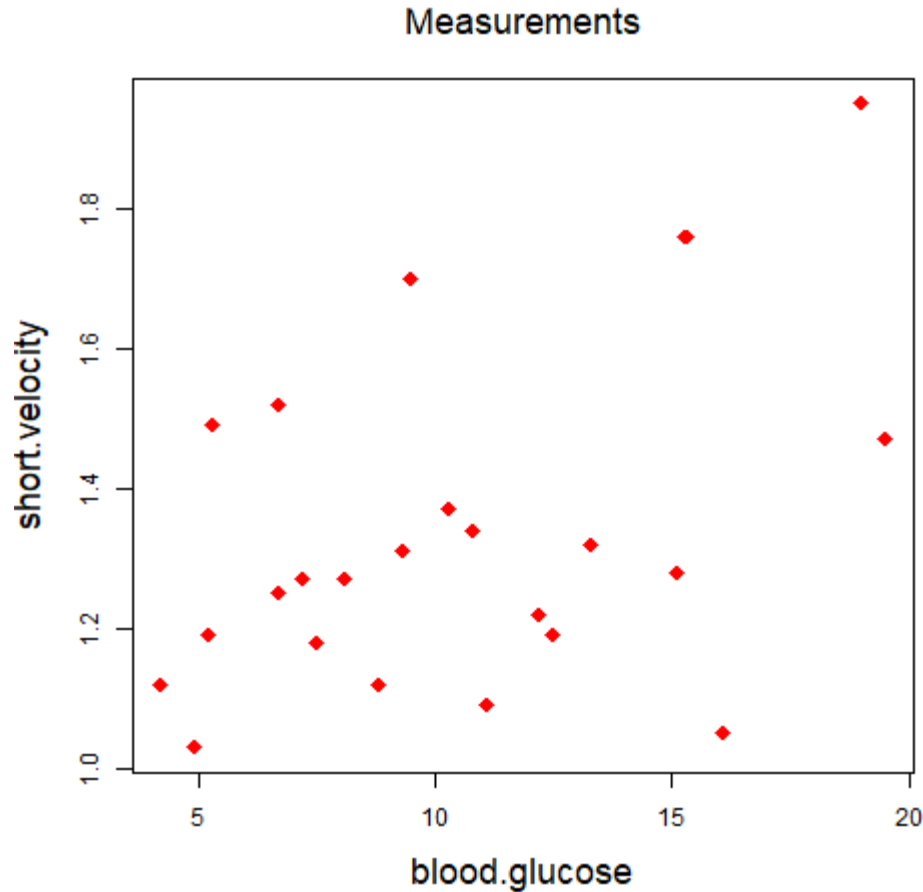
```
# blood.glucose short.velocity
# 1      15.3      1.76
# 2      10.8      1.34
# 3       8.1      1.27
# 4      19.5      1.47
# 5       7.2      1.27
# 6       5.3      1.49
# 7       9.3      1.31
# 8      11.1      1.09
```



```
plot(short.velocity ~ blood.glucose, data=thuesen,
main="Measurements", col="red", font.main=1)
```

The Simple Linear Regression Model

```
plot(short.velocity ~ blood.glucose, data=thuesen, main =  
"Measurements", col = "red", font.main = 1)
```



- Not thrillingly linear
- ... but let's give it a try



The Simple Linear Regression Model

- **Aim:** find functional relationship between blood glucose and velocity
- try with linear relationship:

$$y_i = \alpha + \beta x_i + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim N(0, \sigma)$$

signal + noise (= ε_i)
deterministic + probabilistic

- x : **independent variable**, predictor, regressor, explanatory variable, covariate, ...
- y : **dependent variable**, response, ...

We assume that the **noise is normally distributed** (Central Limit Theorem) with zero mean:

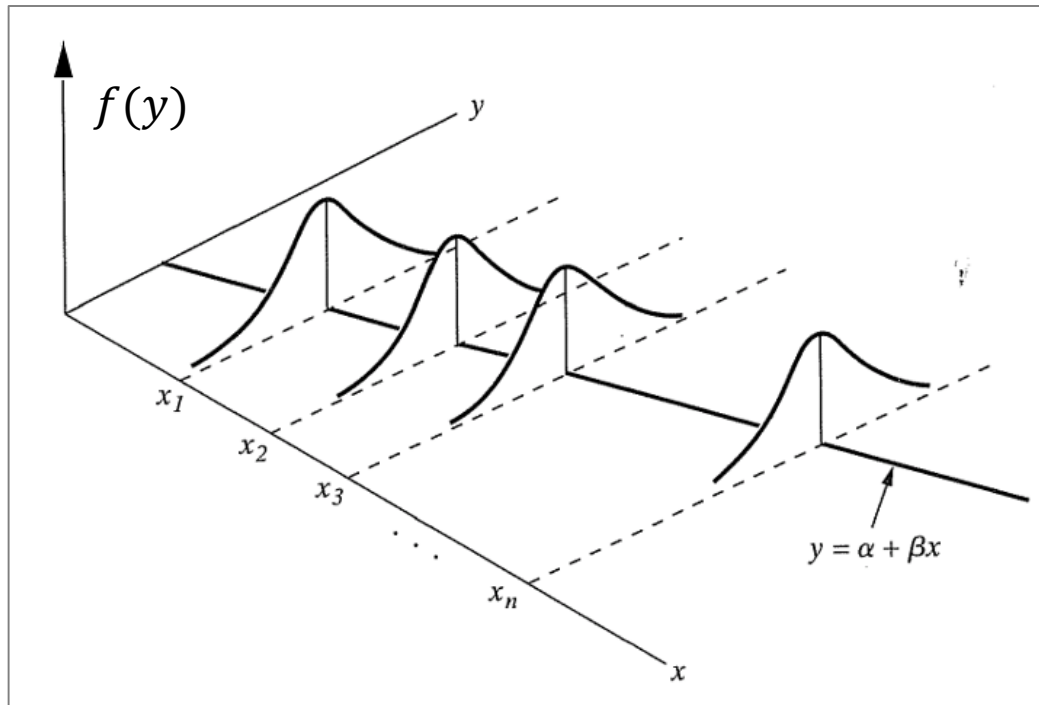
$$\varepsilon_i \sim N(0, \sigma) \quad \Rightarrow \quad Y_i \sim N(\alpha + \beta x_i, \sigma)$$

When the assumption is valid, each response variable Y_i is also normally distributed, with mean $\alpha + \beta \cdot x_i$ and standard deviation σ (However, α, β and σ still unknown $\rightarrow$ estimation)

The Simple Linear Regression Model

Assumptions:

- ε_i normally distributed
- ε_i independent
- equal variance assumption (**homoscedasticity**: same σ for all ε_i)
- x_i known - not random !

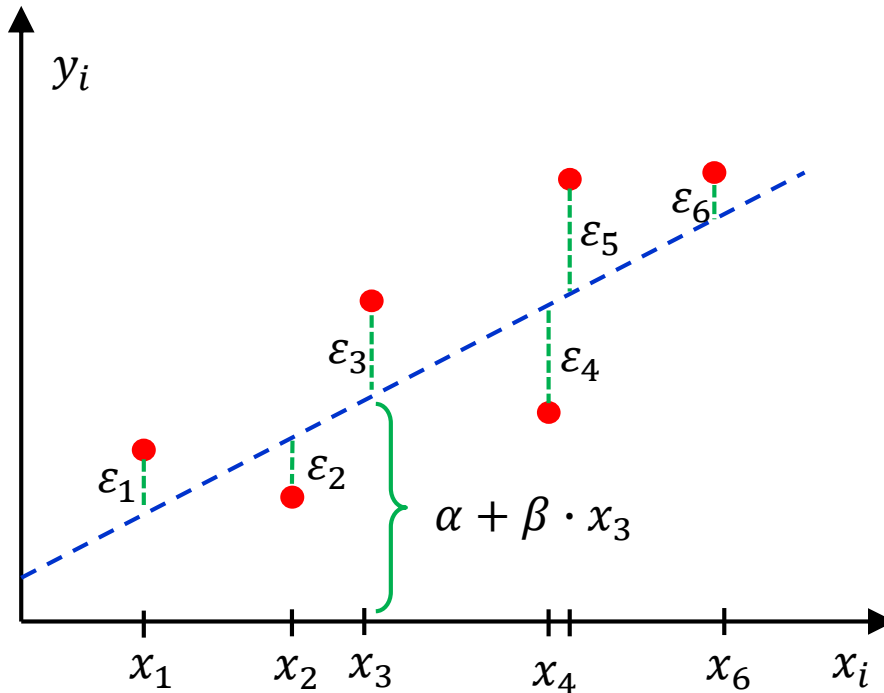


$$\varepsilon_i \sim N(0, \sigma) \quad \Rightarrow$$
$$Y_i \sim N(\alpha + \beta x_i, \sigma)$$

Picture Source: Wackerly et al.,
ISBN 0-534-37741-6

Estimation of α , β , and σ

Least Squares Method



● measurements
--- regression line

Minimize the sum of the squared errors (between estimated curve and measured data points)!

$$y_i = \alpha + \beta x_i + \varepsilon_i$$

$$y_i^* = \alpha^* + \beta^* x_i$$

$$\varepsilon_i \sim N(0, \sigma)$$

$$\varepsilon_i = y_i - y_i^*$$

$$SS_{res} = \sum_{i=1}^n \varepsilon_i^2 = \sum_{i=1}^n (y_i - (\alpha^* + \beta^* x_i))^2 \Rightarrow \text{Minimum}$$

That's essentially the **idea of least squares** !

Least Squares Method*

Find those α and β that make SS_{res} as small as possible (partial derivations must be zero):

$$\frac{\partial SS_{res}}{\partial \alpha^*} = -2 \sum_{i=1}^n (y_i - \alpha^* - \beta^* x_i) = 0$$

$$\frac{\partial SS_{res}}{\partial \beta^*} = -2 \sum_{i=1}^n x_i (y_i - \alpha^* - \beta^* x_i) = 0$$

Two algebraic equations for two unknowns α and β . We find for α and β (see Appendix):

$$\beta^* = \frac{S_{xy}}{S_{xx}} \quad \text{and} \quad \alpha^* = \bar{y} - \beta^* \bar{x}$$

$$S_{xy} = \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) \quad S_{xx} = \sum_{i=1}^n (x_i - \bar{x})^2$$

These quantities can be completely calculated from the sample (from the y_i and x_i). **Note:** α^* and β^* are **random variables** (depend on sample: new sample $\rightarrow$ new $\varepsilon_i \rightarrow y_i \rightarrow S_{xy} \rightarrow \alpha^* \rightarrow \dots$

An estimator for the standard deviation of the noise (σ)

We found the parameters α and β that minimize the residual Sum of Squares:

$$SS_{res} = \sum_{i=1}^n \varepsilon_i^2 = \sum_{i=1}^n (y_i - (\alpha^* + \beta^* x_i))^2 \Rightarrow \text{Minimum}$$

The minimum value is then: (insert the calculated values of α and β into the expression for SS_{res}):

$$\begin{aligned} SS_{res}^0 &= SS_{res}(\alpha^*, \beta^*) = \sum_{i=1}^n (y_i - \alpha^* - \beta^* x_i)^2 \\ &= \sum_{i=1}^n [y_i - \bar{y} + \beta^* \cdot \bar{x} - \beta^* x_i]^2 && \text{using (see Appendix):} \\ & && \bar{y} = \alpha^* + \beta^* \cdot \bar{x} \\ &= \sum_{i=1}^n [(y_i - \bar{y}) - \beta^*(x_i - \bar{x})]^2 \\ &= \dots \\ &= S_{yy} - S_{xy}^2 / S_{xx} \end{aligned}$$

This minimum can be used to estimate the variance of the error terms $\varepsilon_i \rightarrow$

An estimator for the standard deviation of the noise (σ)

$$S^2 = SS_{res}^0 / (n - 2) \quad \text{unbiased estimator for } \sigma^2$$

$$S = \sqrt{SS_{res}^0 / (n - 2)} \quad \text{unbiased estimator for } \sigma$$

Absence of bias can be confirmed by just calculating the expectation value of the above expression for S^2 (calculation not shown here):

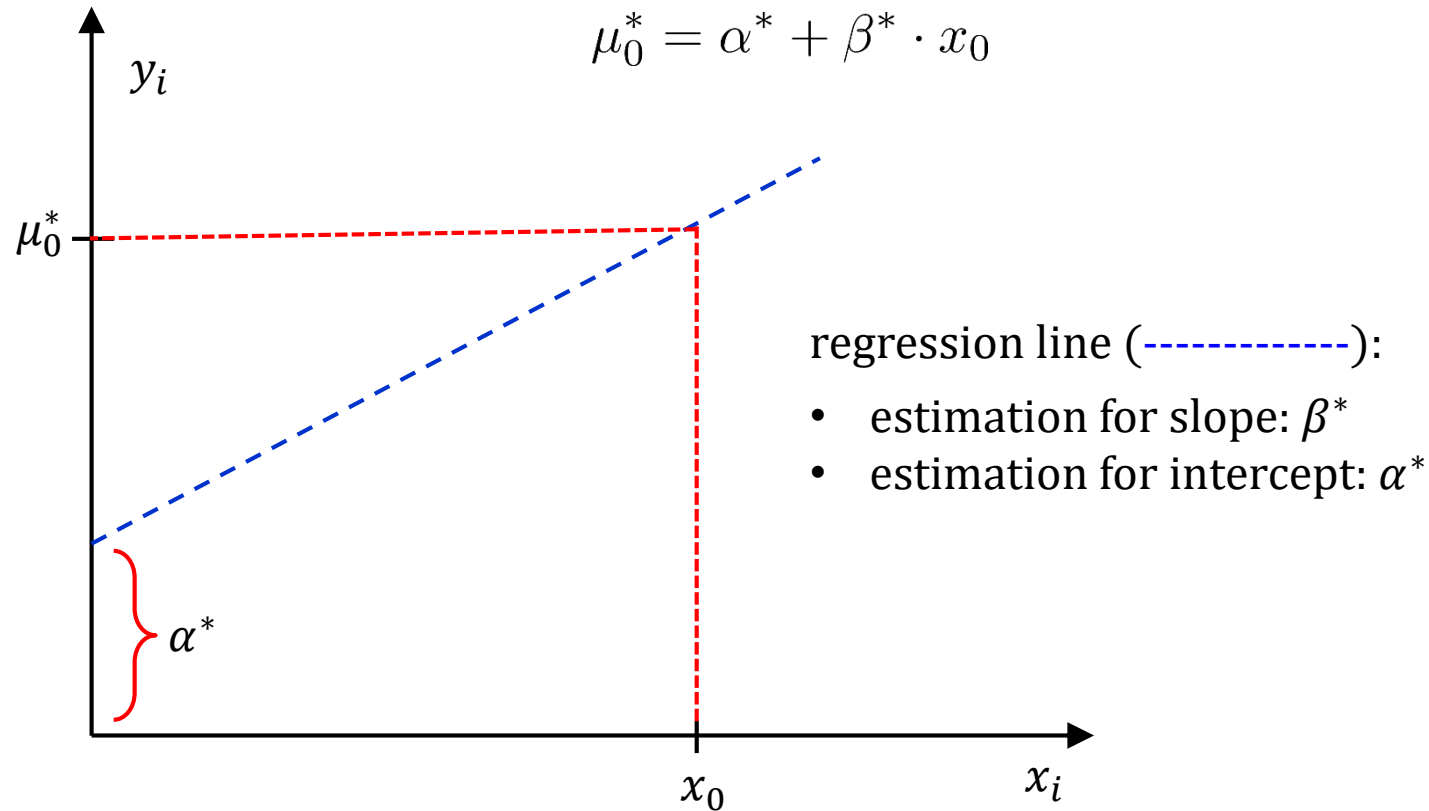
$$E(S^2) = \sigma^2$$

This formula means that S^2 is an **unbiased estimator** for σ^2 :

- if many samples were taken,
- ... and many S^2 were calculated,
- ... their mean would come close to the true σ^2
- ... without systematic error
- the more samples are taken, the closer comes S^2 to σ^2

Calculation of the regression line

Now, an estimation of the response is available for **arbitrary** predictors x_0 :



- x_0 can be located between measured values x_i ($\rightarrow$ prediction)
- α^* and β^* are random variables $\rightarrow \mu_0^*$ is a random variable (changes with each new sample).

Calculating the regression line

see [Reg_Models_Examples.R](#)



```
x = thuesen$blood.glucose
y = thuesen$short.velocity

Sxx = sum((x - mean(x))^2)           # 429.7043
Sxy = sum((x - mean(x))*(y - mean(y))) # 9.437391
Syy = sum((y - mean(y))^2)           # 1.193365

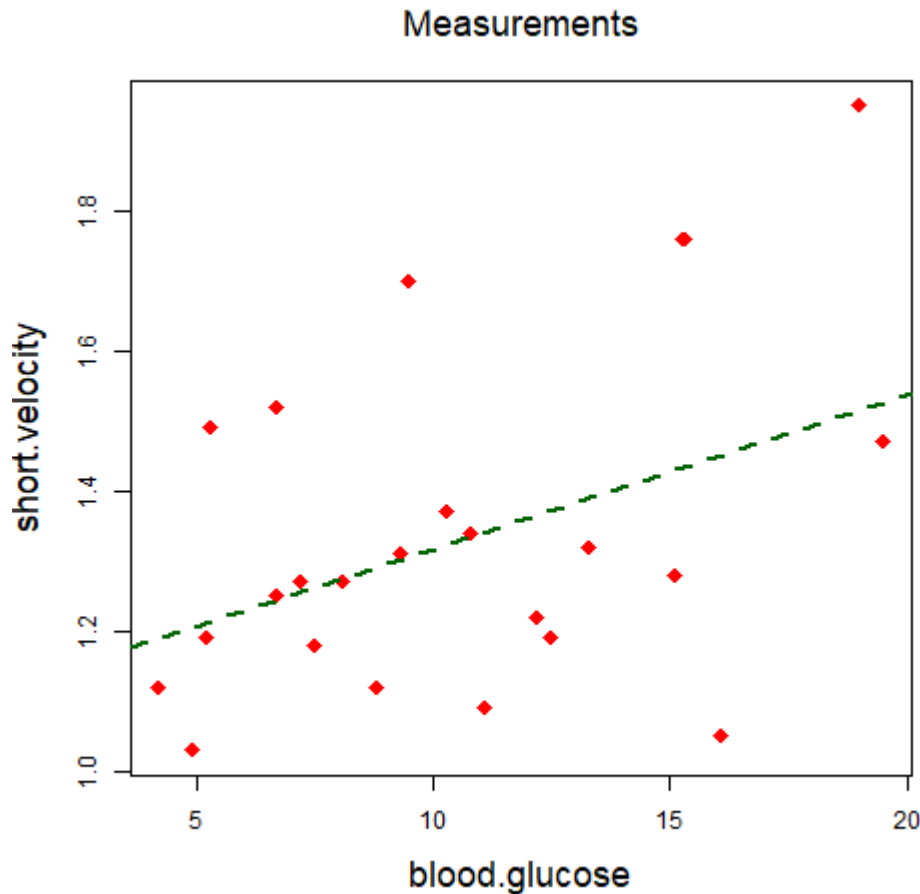
# estimated slope of regression line:
beta.star = Sxy/Sxx                  # 0.02196252
# estimated intercept of regression line:
alpha.star = mean(y) - beta.star*mean(x) # 1.097815

plot(short.velocity ~ blood.glucose, data = thuesen, main =
"Measurements and Regression Line", col="red", font.main=1)

abline(a=alpha.star, b=beta.star, col="darkgreen", lty=2, lwd=2)
```

How reliable is the regression line?

see [Reg_Models_Examples.R](#)

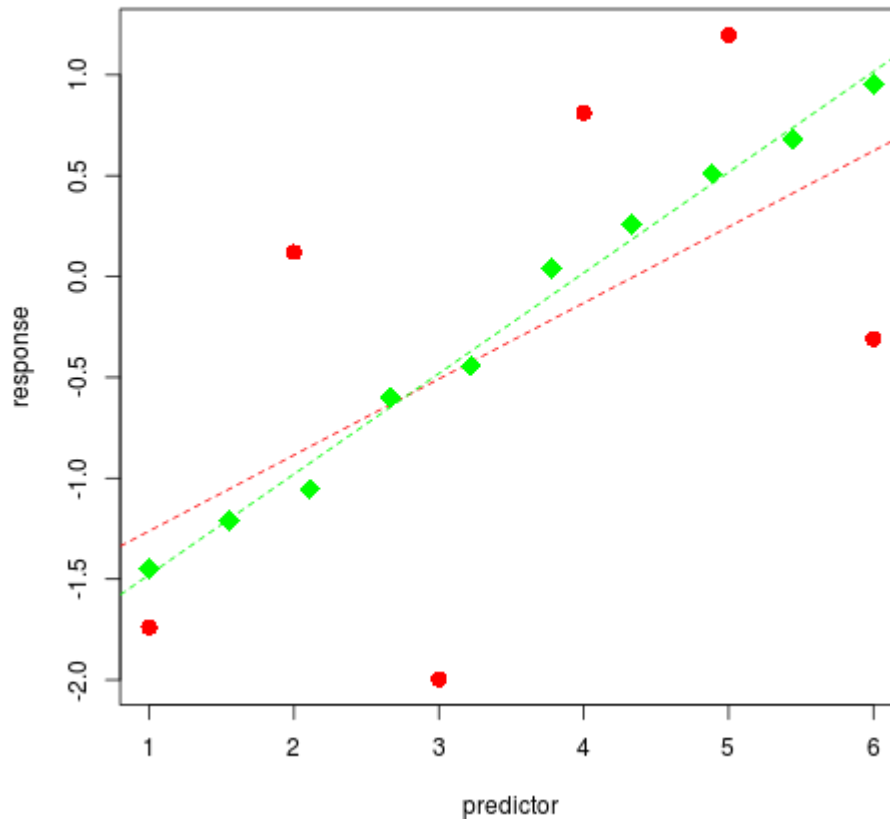


Questions:

- How sure can we be about the obtained result ?
- If we took a new sample from the same population: how much would the new regression line deviate from this one ?
- when taking a new sample: could it also be possible to get a regression line with slope zero, indicating that x and y are not related ?
- Is the result "significant"?

How reliable is the regression line?

- Exclude the possibility that the result is spurious, just emerging from the current sample
- **Important:** if the slope is "significantly" different from 0, we have some support that y really depends on x



- **Red points:** high variance in the data, slope depends very much on particular values, changing few values could change the slope essentially
- **Green points:** low variance, slope seems to be fairly justified
- **For situations not so extreme:**
- A **test** is needed to show if a data set yields a "justified" slope (a slope which is significantly different from zero)

Sampling Distribution (of the estimated parameters)

- Slope and intercept estimated from a sample deviate from the true but unknown values (limited sample size, random noise).
- new sample $\rightarrow$ new y_i 's $\rightarrow$ new values for slope and intercept
- If we measure many sets of y_i and calculate slope and intercept for each of this sets, we observe a **distribution** of estimated slopes and intercepts.
- This is called **sampling distribution**.

```
betai <- numeric(1000)
```

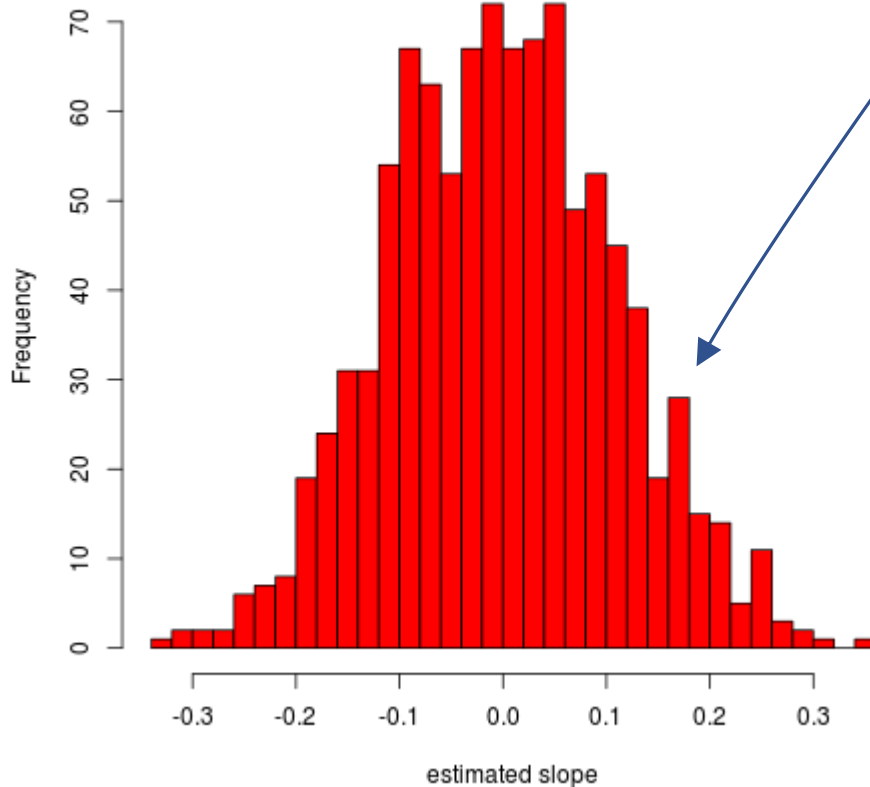


```
for (i in 1:1000) {  
  xi = seq(1,10,len=10)          # not random  
  yi = 1 + rnorm(10, mean=0, sd=1) # y not depending on x !  
  Sxx = sum((xi - mean(xi))^2)  
  Sxy = sum((xi - mean(xi))*(yi - mean(yi)))  
  betai[i] = Sxy/Sxx             # estimated slope for sample i  
}
```

```
hist(betai, breaks=25, main="Distribution of 1000 estimators  
of the slope", font.main=1, col="red", xlab="estimated  
slope")
```

Sampling Distribution for the estimated slope

Distribution of 1000 estimators of the slope



- **Example:** 30 samples yield a slope between 0.16 and 0.18
- ... but the true slope is zero (y does not depend on x)
- The estimated slopes inferred from some samples could make us believe the slope has some non-zero value, leading us to the false conclusion that y actually depends on x.
- **Conclusion:** we have to **test** if our estimations are **significant**

Probability distributions of the estimates

What probability distributions do α^* , β^* and μ_0^* have?

$$\beta^* = \frac{S_{xy}}{S_{xx}} \quad \text{and} \quad \alpha^* = \bar{y} - \beta^* \bar{x}$$

$$S_{xy} = \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) \quad S_{xx} = \sum_{i=1}^n (x_i - \bar{x})^2$$

With some calculus ☺ (see Appendix), we get:

$$\beta^* = \sum_{i=1}^n c_i \cdot y_i \quad \text{with} \quad c_i = \frac{x_i - \bar{x}}{S_{xx}}$$
$$\alpha^* = \sum_{i=1}^n d_i \cdot y_i \quad \text{with} \quad d_i = \frac{1}{n} - c_i \bar{x}$$

- α^* and β^* are linear combinations of the y_i
- because the y_i are normally distributed, the α^* and β^* are as well

Probability distributions of the estimates

The estimates α^* , β^* , μ_0^* are normally distributed. Expectation values, variances and standard deviations can be calculated:

$$E(\beta^*) = \dots = \beta \quad \text{unbiased estimator (see Appendix for derivation)}$$

$$V(\beta^*) = \dots = \frac{\sigma^2}{S_{xx}} \quad \text{consistent, because} \quad S_{xx} = \sum_{i=1}^n (x_i - \bar{x})^2 \rightarrow \infty \quad \text{for} \quad n \rightarrow \infty$$

$$E(\mu_0^*) = E(\alpha^* + \beta^* \cdot x_0) = E\left[\sum_{i=1}^n (d_i + c_i \cdot x_0) \cdot Y_i\right] = \dots = \alpha + \beta \cdot x_0 = \mu_0$$

$$\begin{aligned} V(\mu_0^*) &= V(\alpha^* + \beta^* \cdot x_0) = V\left[\sum_{i=1}^n \left(\frac{1}{n} + c_i \cdot (x_0 - \bar{x})\right) \cdot Y_i\right] = \dots \\ &\dots = \sigma^2 \cdot \left(\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}}\right) \quad \text{using} \quad d_i = \frac{1}{n} - c_i \cdot \bar{x} \end{aligned}$$

Expectation and variance for α^* follow from the corresponding expressions for μ_0^* by setting $x_0 = 0$. Note that the variance of μ_0^* is big when x_0 is far from $\bar{x}$.

Probability distributions of the estimates

$$\beta^* \sim N \left(\beta, \frac{\sigma}{\sqrt{S_{xx}}} \right)$$

distribution for the slope estimation (if $\varepsilon_i \sim N$)

$$\alpha^* \sim N \left(\alpha, \sigma \sqrt{\frac{1}{n} + \frac{\bar{x}^2}{S_{xx}}} \right)$$

distribution for the intercept estimation (if $\varepsilon_i \sim N$)

$$\mu_0^* \sim N \left(\mu_0, \sigma \sqrt{\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{S_{xx}}} \right)$$

distribution for the points on the regression line (if $\varepsilon_i \sim N$)
($x_0 = 0 \rightarrow$ distribution for α)

$$\mu_0 = \alpha + \beta \cdot x_0$$

All estimators are **unbiased** and **consistent**.

Finding a Pivot variable for testing if $\beta = 0$

To establish a test, we need a **pivot variable**:

1. includes the parameter of interest (parameter the hypothesis is about, e.g. β)
2. all other quantities must be known (calculated from the sample)
3. the probability distribution of the pivot variable must be known

$$\frac{\beta^* - \beta}{se(\beta^*)} \sim t(n - 2)$$

pivot variable, t -distributed with $(n - 2)$ degrees of freedom

$$se(\beta^*) = S / \sqrt{S_{xx}}$$

standard error for estimator of the slope

$$S = \sqrt{SS_{res}^0 / (n - 2)}$$

$$SS_{res}^0 = S_{yy} - S_{xy}^2 / S_{xx}$$

Minimum residual deviance
(see below)

Finding the Pivot variable (*)

$$\beta^* \sim N\left(\beta, \frac{\sigma}{\sqrt{S_{xx}}}\right)$$

distribution for the slope
estimation (if $\varepsilon_i \sim N$)

$$Z = \frac{\beta^* - \beta}{\sigma / \sqrt{S_{xx}}} \sim N(0, 1)$$

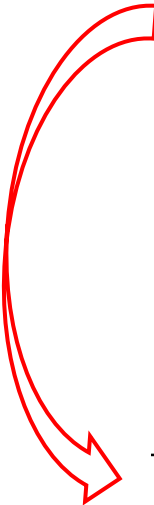
standard normal, but not a
pivot (because σ is unknown)

$$SS_{res}^0 = \sum_{i=1}^n \varepsilon_i^2$$

sum of squares for residuals

$$\frac{SS_{res}^0}{\sigma^2} = \sum_{i=1}^n \left(\frac{\varepsilon_i}{\sigma}\right)^2 \sim \chi^2(n-2)$$

sum of squares of independent
 $N(0,1)$ -variables is χ^2 -distributed


$$\frac{Z}{\sqrt{\frac{\chi^2(n)}{n}}} \sim t(n) \quad \text{generally valid for all } n \quad \Rightarrow \quad \frac{Z}{\sqrt{\frac{\chi^2(n-2)}{n-2}}} \sim t(n-2)$$

$$\frac{Z}{\sqrt{\frac{SS_{res}^0}{\sigma^2(n-2)}}} \sim t(n-2)$$

now, insert the $N(0, 1)$ -distributed
variable (above) $\rightarrow$

Finding the pivot variable (*)

$$Z = \frac{\beta^* - \beta}{\sigma / \sqrt{S_{xx}}} \sim N(0, 1)$$


$$\frac{\frac{\beta^* - \beta}{\sigma / \sqrt{S_{xx}}}}{\sqrt{\frac{SS_{res}^0}{\sigma^2(n-2)}}} \sim t(n-2)$$

the unknown σ can be cancelled !

$$\frac{\beta^* - \beta}{\sqrt{\frac{SS_{res}^0}{S_{xx}(n-2)}}} \sim t(n-2)$$

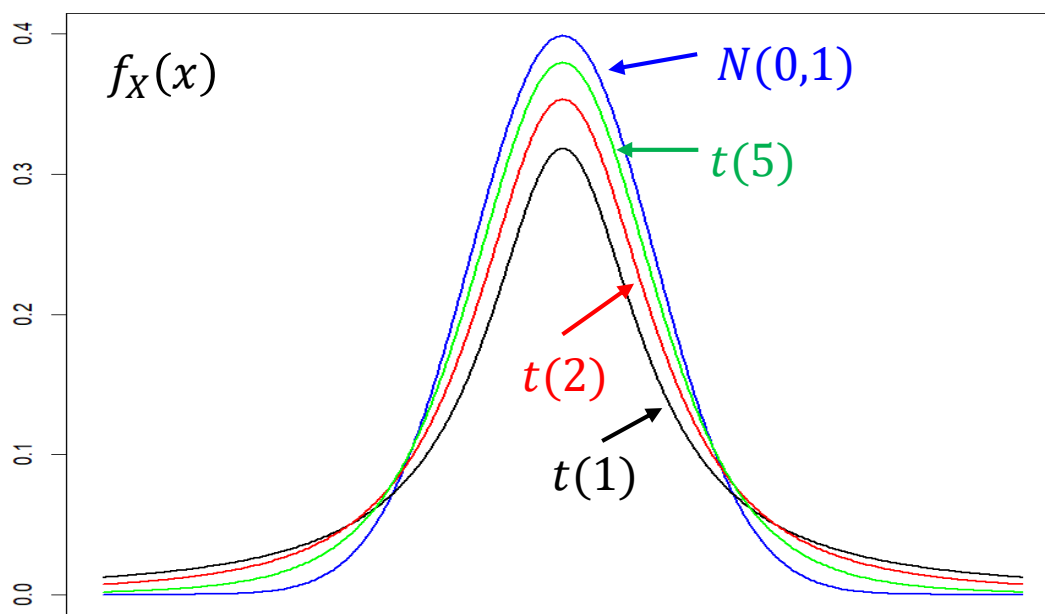
$$S = \sqrt{SS_{res}^0 / (n-2)} \quad \text{estimation for } \sigma$$

$$\frac{\beta^* - \beta}{S / \sqrt{S_{xx}}} \sim t(n-2)$$

Pivot!: everything known from sample except for β (which we want to test)

Pivot variable if true σ is unknown*

- Replacing the true (but unknown) σ by it's estimate S
- probability distribution shifts therefore from N to the somewhat broader t -distribution. This reflects the increase of uncertainty.



$$\frac{\beta^* - \beta}{\sigma / \sqrt{S_{xx}}} \sim N \quad (\text{narrow})$$

$$\frac{\beta^* - \beta}{S / \sqrt{S_{xx}}} \sim t \quad (\text{broad})$$

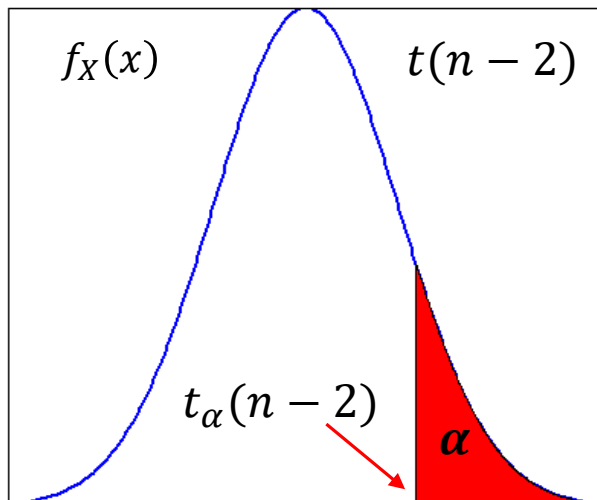
$$f(t) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi} \Gamma(\frac{\nu}{2})} \left(1 + \frac{t^2}{\nu}\right)^{-\left(\frac{\nu+1}{2}\right)}$$

Probability density function
 ν = degrees of freedom

Testing the hypothesis $\beta = 0$

$H_0: \beta = 0$ t-statistic (just a number)

$$\frac{\beta^* - 0}{S/\sqrt{S_{xx}}} \sim t(n - 2)$$



H_0 **not** rejected if
t is here

H_0 rejected if
t is here

α : significance level
(area exxegerated in the plot)

We do not reject H_0 if the value of the t-statistic calculated from the sample lies well within the distribution expected under H_0 .

The plot corresponds to a one-sided test. For a two-sided test, an area of $\alpha/2$ must be tagged in both tails.

Calculating the p-value belonging to the slope

$$\frac{\beta^* - 0}{S/\sqrt{S_{xx}}} \sim t(n-2) \quad \text{hypothesis } \beta = 0$$


```
n = length(x)
df = n-2                                # 23                degrees of freedom
SSres0 = Syy - Sxy^2/Sxx                # 0.9860963          residual deviance
s = sqrt(SSres0/df)                    # 0.2166956          estimated sigma
se.beta = s/sqrt(Sxx)                  # 0.01045358         standard error
t = beta.star / se.beta                 # 2.100957           t-statistic
p.value = 2*pt(t, df=df, lower.tail=FALSE) # 0.0468             two-sided test
```

For SSres0, see page 9

Simple Regression using "lm"



see [Reg_Models_Examples.R](#)

```
lm.out = lm(short.velocity ~ blood.glucose, data = thuesen)
lm.out

# Call:
# lm(formula = short.velocity ~ blood.glucose, data = thuesen)
#
# Coefficients:
# (Intercept) blood.glucose
# 1.09781 0.02196
```

These are the same results for slope and intercept as obtained above (page 12)

More lm-output by using summary



```
summary(lm.out)
```

```
# Call:
# lm(formula = short.velocity ~ blood.glucose, data = thuesen)
#
# Residuals:
#      Min       1Q   Median       3Q      Max
# -0.40141 -0.14760 -0.02202  0.03001  0.43490
#
# Coefficients:
#              Estimate Std. Error t value Pr(>|t|)
# (Intercept)   1.09781    0.11748   9.345 6.26e-09 ***
# blood.glucose  0.02196    0.01045   2.101 0.0479 *
# ---
# Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
# Residual standard error: 0.2167 on 21 degrees of freedom
# Multiple R-squared:  0.1737, Adjusted R-squared:  0.1343
# F-statistic: 4.414 on 1 and 21 DF, p-value: 0.0479
```

Suggestion: let's try to understand the output of `summary(lm.out)`:

Residuals

```
# Residuals:  
# Min 1Q Median 3Q Max  
# -0.40141 -0.14760 -0.02202 0.03001 0.43490
```

$\varepsilon_i = y_i^* - y_i$ Difference between measured and fitted points

```
residuals = resid(lm.out) # another extractor function  
summary(residuals)       # the usual summary command
```

```
#      Min.      1st Qu.  Median      Mean  3rd Qu.  Max.  
# -0.40140 -0.14760 -0.02202 0.00000  0.03001  0.43490  
# mean=0 as expected (normal eqns.)
```

Possibility to (roughly) check the assumptions:

- median close to zero
 - skewness (N is symmetrical)
- $\varepsilon_i \sim N(0, \sigma)$

Estimated coefficients, their standard errors, t-statistic and p-values

```
# Coefficients:
#               Estimate Std. Error t value Pr(>|t|)
# (Intercept)   1.09781   0.11748   9.345   6.26e-09 ***
# blood.glucose 0.02196   0.01045   2.101   0.0479 *
```

$$se(\beta^*) = \frac{S}{\sqrt{S_{xx}}} \quad S = \sqrt{\frac{SS_{res}^0}{n-2}} \quad SS_{res}^0 = S_{yy} - S_{xy}^2/S_{xx}$$

```
SSreg0 = Syy - Sxy^2/Sxx
s = sqrt(SSreg0/(length(x)-2))
se.beta = s/sqrt(Sxx) # 0.01045358
```

$$t = \frac{\beta^*}{se(\beta^*)} \quad (\text{page 15, for } \beta = 0)$$

```
t = beta.star / se.beta # 2.100957
p.value = 2*pt(t, df=length(x)-2, lower.tail=FALSE) # 0.04789591
```

Significance codes

```
# Coefficients:
#               Estimate Std. Error t value Pr(>|t|)
# (Intercept)    1.09781    0.11748   9.345 6.26e-09 ***
# blood.glucose  0.02196    0.01045   2.101 0.0479  *
# ---
# Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1
```

Residual standard error

```
# Residual standard error: 0.2167 on 21 degrees of freedom
# Multiple R-squared: 0.1737, Adjusted R-squared: 0.1343
# F-statistic: 4.414 on 1 and 21 DF, p-value: 0.0479
```

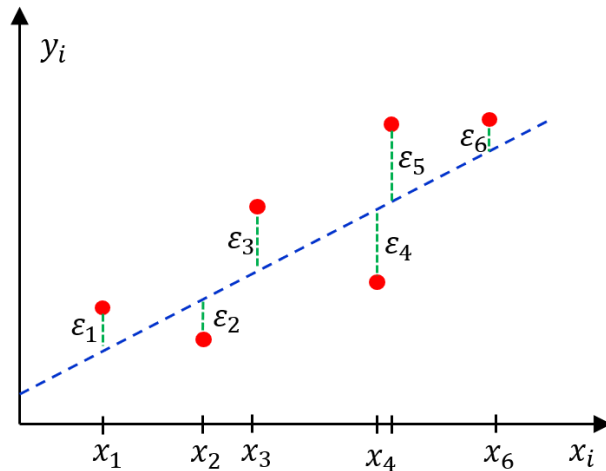
$$S = \sqrt{\frac{SS_{res}^0}{n-2}} \quad SS_{res}^0 = S_{yy} - S_{xy}^2/S_{xx}$$

```
s = sqrt(SSreg0/(n-2)) # 0.2166956 estimation for sigma
```

Coefficient of Determination

("How good is the fit?")

Residual standard error: 0.2167 on 21 degrees of freedom
Multiple R-squared: 0.1737, Adjusted R-squared: 0.1343
F-statistic: 4.414 on 1 and 21 DF, p-value: 0.0479



The y -values vary for two reasons:

1. because they change with x and
2. because of the noise ε_i

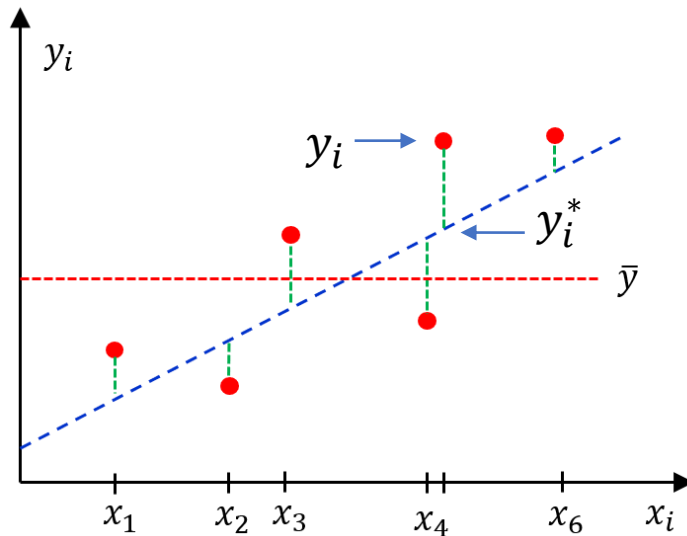
Want to find out:

- How much of the total variance of y can be explained by the linkage to x (i.e. by the signal) and
- ... how much must be considered as random noise (left unexplained)?
- The lower the random noise compared to the signal, the better the fit.

Coefficient of Determination

- Subdividing the variation -

Subdivide the total variation SS_{tot} into two parts:



Sums of squares:

$$SS_{tot} = \sum_i (y_i - \bar{y})^2$$

$$SS_{reg} = \sum_i (y_i^* - \bar{y})^2$$

$$SS_{res} = \sum_i (y_i - y_i^*)^2$$

$$SS_{tot} = SS_{reg} + SS_{res}$$

$\bar{y}$: grand mean ; y_i : measured points ; y_i^* : estimated points

SS_{tot} : total Sum of Squares, regression + random noise

SS_{reg} : Sum of Squares for regression, describing the change by linkage to x

SS_{res} : Sum of Squares for residuals, describing random noise

Coefficient of Determination

- Definition -

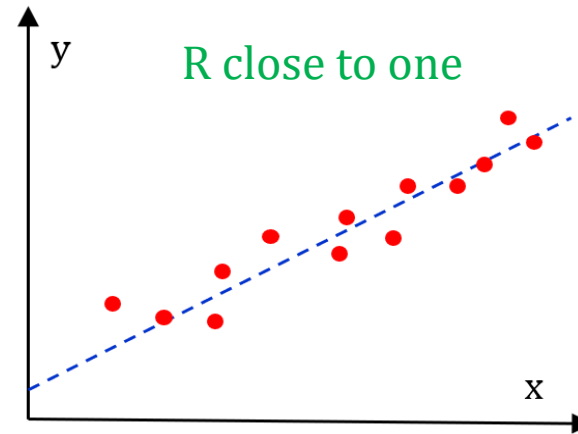
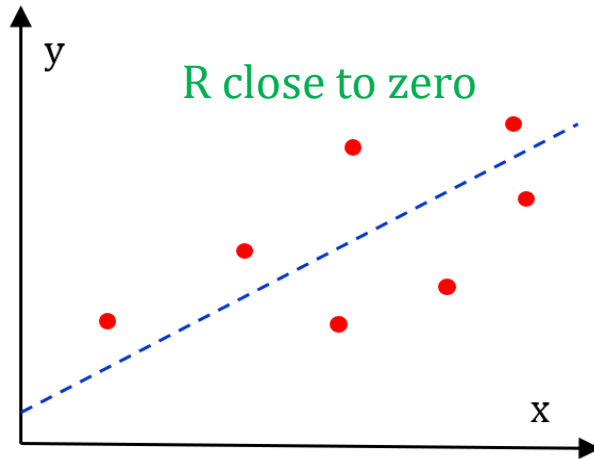
```
# Residual standard error: 0.2167 on 21 degrees of freedom
# Multiple R-squared: 0.1737, Adjusted R-squared: 0.1343
# F-statistic: 4.414 on 1 and 21 DF, p-value: 0.0479
```

$$R^2 = \frac{SS_{reg}}{SS_{tot}} = 1 - \frac{SS_{res}}{SS_{tot}} = \frac{S_{xy}^2}{S_{xx} \cdot S_{yy}} \quad \text{Definition}$$

- R^2 is between 0 and 1
- $R^2 = 1$ is perfect!

```
SS.tot = sum((y - mean(y))^2)      # 1.1933
SS.reg = sum((y.hat - mean(y))^2)  # 0.2072 deviance by regression
SS.res = sum(resid(lm.out)^2)      # 0.9861 deviance by noise  $\varepsilon_i$ 
SS.tot / (SS.reg + SS.res)         # 1 ok
R.squared = SS.reg/SS.tot          # 0.17368 as in summary above
R.squared = 1 - SS.res/SS.tot      # 0.17368 same
R.squared = Sxy^2/(Sxx*Syy)        # 0.17368 same
```

How well did the fit work?



$$R^2 = \frac{S_{xy}^2}{S_{xx} \cdot S_{yy}} = 1 - \frac{\sum (y_i - y_i^*)^2}{\sum (y_i - \bar{y})^2}$$

Coefficient of determination

poor fit $\rightarrow 0 \leq R^2 \leq 1 \leftarrow$ good fit

$$R^2 = r_{xy}^2 \text{ (for simple linear regression)} \quad r_{xy} = \frac{c_{xy}}{s_x \cdot s_y} \quad \begin{array}{l} \text{correlation coefficient} \\ c_{xy} = \text{covariance} \end{array}$$

R^2 and the Pearson correlation coefficient*

$$R^2 = \frac{S_{xy}^2}{S_{xx} \cdot S_{yy}} \quad \text{Coefficient of determination}$$

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} = \frac{S_{xy}}{\sqrt{S_{xx}S_{yy}}} \quad \text{Pearson correlation}$$

Coefficient of determination and Pearson correlation coefficient are the same. This is valid for simple linear regression, i.e. if we have one independent variable only (see below for regression with multiple independent variables).

```
r = Sxy/sqrt(Sxx*Sy) # 0.4167
cor(blood.glucose, short.velocity, method="pearson") # 0.4167
r^2 # 0.1736844 = R^2
R.squared # 0.1736844 same!
```

see [Reg_Models_Examples.R](#)

Adjusted Coefficient of Determination*

```
# Residual standard error: 0.2167 on 21 degrees of freedom  
# Multiple R-squared: 0.1737, Adjusted R-squared: 0.1343  
# F-statistic: 4.414 on 1 and 21 DF, p-value: 0.0479
```

$$R_{adj}^2 = 1 - (1 - R^2) \frac{n - 1}{n - p - 1}$$

p is the number of independent variables used in the model

```
n = length(x) # 25  
p = 1 # number of independent variables  
R2.adjusted = 1 - (1 - R.squared) * (n - 1) / (n - p - 1) # 0.134336 ok
```

- **Adjusted R-squared:** compares explanatory power of regression models with different numbers of predictors ($x_1, x_2, x_3, \dots$)
- **In general:** using more predictors, a better fit can **always** be achieved.
- The usual R^2 **always** increases when new predictors are added.
- The adjusted R^2 increases only if the new term improves the model significantly.

The F-Test for Analyzing the Variance*

```
# Residual standard error: 0.2167 on 21 degrees of freedom
# Multiple R-squared: 0.1737, Adjusted R-squared: 0.1343
# F-statistic: 4.414 on 1 and 21 DF, p-value: 0.0479
```

Test: $H_0: \beta = 0$

- If the null hypothesis is rejected: the predictor “explains” the response
- If not: no evidence for a linkage between x and y

$$\frac{\frac{\chi^2(n)}{n}}{\frac{\chi^2(m)}{m}} \sim F(m, n)$$

- This relationship is generally valid.
- $F(m, n)$: **F-distribution**
- m : numerator degrees of freedom
- n : denominator degrees of freedom

The F-Test for Analyzing the Variance*

$$\frac{1}{\sigma^2} SS_{reg} = \frac{1}{\sigma^2} \sum_i (y_i^* - \bar{y})^2 = \frac{\beta^{*2}}{\sigma^2} \sum_i (x_i - \bar{x})^2$$

because $y_i^* = \alpha^* + \beta^* \cdot x_i$ and $\bar{y} = \alpha^* + \beta^* \cdot \bar{x} \rightarrow y_i^* - \bar{y} = \beta^* \cdot (x_i - \bar{x})$

$$\frac{1}{\sigma^2} SS_{reg} = \frac{\beta^{*2}}{\sigma^2} S_{xx} \quad \text{because} \quad S_{xx} = \sum_{i=1}^n (x_i - \bar{x})^2$$

$$\frac{1}{\sigma^2} SS_{reg} = \left(\frac{\beta^*}{\sigma / \sqrt{S_{xx}}} \right)^2$$

remember: $\sigma / \sqrt{S_{xx}}$ is the standard deviation of β^*
and β is the mean of β^* (page 19)

We assume $H_0: \beta = 0$, so that $E(\beta^*) = 0$. It follows that SS_{reg}/σ^2 is a square of a normally distributed variable, i.e. it is χ^2 - distributed with one degree of freedom.

$$\frac{1}{\sigma^2} SS_{reg} \sim \chi^2(1)$$

Here, we estimated 2 parameters: α, β . A general expression for p parameters is

$$\frac{1}{\sigma^2} SS_{reg} \sim \chi^2(p - 1)$$

The F-Test for Analyzing the Variance*

$$\frac{1}{\sigma^2} SS_{res} = \frac{1}{\sigma^2} \sum_i (y_i - y_i^*)^2 = \frac{1}{\sigma^2} \sum_i \varepsilon_i^2 = \sum_i \left(\frac{\varepsilon_i}{\sigma} \right)^2$$

Remember that $\varepsilon_i \sim N(0, \sigma)$, so that each term of the sum is a square of a normally distributed variable. It follows that

$$\frac{1}{\sigma^2} SS_{res} \sim \chi^2(n - 2)$$

because we estimated 2 parameters: α, β , and therefore lose two degrees of freedom. A general expression for p parameters is

$$\frac{1}{\sigma^2} SS_{res} \sim \chi^2(n - p)$$

The F-Test for Analyzing the Variance*

$$\frac{\frac{\chi^2(n)}{n}}{\frac{\chi^2(m)}{m}} \sim F(m, n)$$

- This relationship is generally valid.
- $F(m, n)$: **F-distribution**

We have seen that $\frac{1}{\sigma^2} SS_{reg} \sim \chi^2(p - 1)$

and $\frac{1}{\sigma^2} SS_{res} \sim \chi^2(n - p)$

Therefore, we have

$$\frac{\frac{1}{\sigma^2} \frac{SS_{reg}}{p-1}}{\frac{1}{\sigma^2} \frac{SS_{res}}{n-p}} \sim F(p - 1, n - p) \quad (\text{unknown } \sigma \text{ cancels}) \quad \frac{\frac{\chi^2(p-1)}{(p-1)}}{\frac{\chi^2(n-p)}{(n-p)}}$$

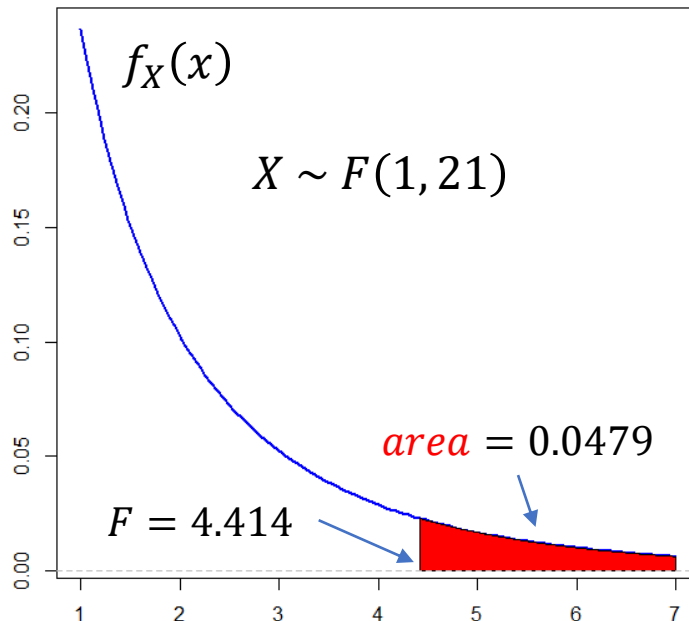
$$F = \frac{\frac{SS_{reg}}{p-1}}{\frac{SS_{res}}{n-p}} \sim F(p - 1, n - p) \quad \dots \text{if } H_0 \text{ is true, i.e. } \beta = 0$$

If we want evidence for a functional relationship between x and y , we have to **reject the H_0** .

The F-Test for Analyzing the Variance*

```
# Residual standard error: 0.2167 on 21 degrees of freedom
# Multiple R-squared: 0.1737, Adjusted R-squared: 0.1343
# F-statistic: 4.414 on 1 and 21 DF, p-value: 0.0479
```

$$F = \frac{\frac{SS_{reg}}{p-1}}{\frac{SS_{res}}{n-p}} \sim F(p-1, n-p) \quad \begin{array}{l} p : \text{number of parameters,} \\ \text{here: } p = 2 (\alpha, \beta) \end{array}$$



- $H_0: \beta = 0$
- F gets big if the regression variation is big compared to the residual variation
- If F gets big, we reject the null hypothesis, i.e. we consider β as significantly different from zero.

$F = 4.414$ is the observation

The F-Test for Analyzing the Variance*

```
# Residual standard error: 0.2167 on 21 degrees of freedom
# Multiple R-squared: 0.1737, Adjusted R-squared: 0.1343
# F-statistic: 4.414 on 1 and 21 DF, p-value: 0.0479
```

$$F = \frac{\frac{SS_{reg}}{p-1}}{\frac{SS_{res}}{n-p}} \sim F(p-1, n-p) \quad \begin{array}{l} p : \text{number of parameters,} \\ \text{here: } p = 2 (\alpha, \beta) \end{array}$$

```
num.params = 2 # number of regression parameters
n           # 23 , number of data points
df.reg = num.params - 1 # 1 df for regression variation
df.res = n - num.params # 21 df for residual variation
F = (SS.reg/df.reg) / (SS.res/df.res) # 4.414018
p.value = pf(F, df.reg, df.res, lower.tail = FALSE) # 0.04789591
```

Simple linear regression ($p = 2$): same p-value for F-test as for t-test:

```
# Coefficients:
#               Estimate Std. Error t value Pr(>|t|)
# (Intercept)   1.09781    0.11748   9.345 6.26e-09 ***
# blood.glucose 0.02196    0.01045   2.101 0.0479 *
```

The F-Test for Analyzing the Variance*

Why is the p-value the same as for t-test?

$$\frac{\frac{\chi^2(n)}{n}}{\frac{\chi^2(m)}{m}} \sim F(m, n) \quad \text{F-distribution, general}$$

$$\frac{\chi^2(1)}{\frac{\chi^2(m)}{m}} \sim F(1, n) \quad \text{for numerator } df = 1$$

$\chi^2(1)$ is a square of a standard-normal variable
 $n = 1$ (simple linear regression)

If we take the square-root of this, we get:

$$\frac{Z}{\sqrt{\frac{\chi^2(m)}{m}}} \sim t(m) \quad \text{t-distributed} \quad F = t^2 \text{ for simple linear regression}$$

remember that $\frac{Z}{\sqrt{\frac{\chi^2(m)}{m}}} \sim t(m)$ is generally valid for all m

```
F = (SS.reg/df.reg) / (SS.res/df.res) # 4.414018
t = beta.star / se.beta # 2.100957
t^2 # 4.414018 same as F
```

The F-Test is what ANOVA does

- In general, ANOVA compares two variances
- With regard to regression, ANOVA performs the same analysis as above
- ANOVA “knows” what to do with an `lm` object

```
anova(lm.out) # apply ANOVA to lm object
# Analysis of Variance Table
# Response: short.velocity
#
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
# blood.glucose	1	0.20727	0.207269	4.414	0.0479 *
# Residuals	21	0.98610	0.046957		

- These are the **same** numbers as obtained above.
- Later we will see that ANOVA can also be used to compare the performance of two regression models (by comparing the residual variances of both models).

Checking the Assumptions of the Model

Assumptions:

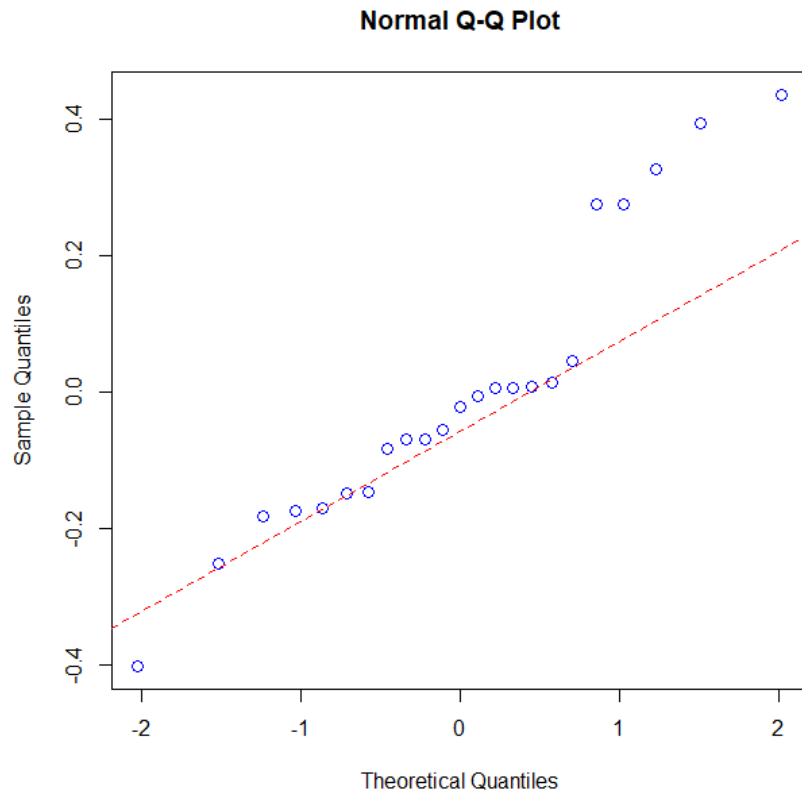
- Residuals ε_i normally distributed
- Equal variance assumption (homoscedasticity): same σ for all ε_i
- Residuals ε_i independent

$$\varepsilon_i \sim N(0, \sigma)$$

Normality of the Residuals

Create Quantile-quantile (QQ-) plot with the residuals:

```
qqnorm(resid(lm.out), col = "blue", cex = 1.3) # QQ-plot  
qqline(resid(lm.out), col = "red", lty = 2)
```



A **QQ-plot** compares empirical quantiles (obtained from the data) with the quantiles deriving from some theoretical distribution. The function `qqnorm` compares empirical quantiles with the quantiles of the normal distribution. The data do not conflict with the assumed distribution if the data points roughly follow the line created by `qqline`. Deviations occur often at both ends, indicating deviations in the tails of the distribution.

Normality of the Residuals

R provides also a number of functions to **test for normality**. The null hypothesis of the tests is that the data are normally distributed. If the null cannot be rejected, we have no evidence that the data deviate from a normal distribution, i.e. we may accept that the data is normally distributed. Note that failure of H_0 -rejection does **not proof** that the data is normally distributed. There might simply be too few data to give the test enough power. Some functions are:

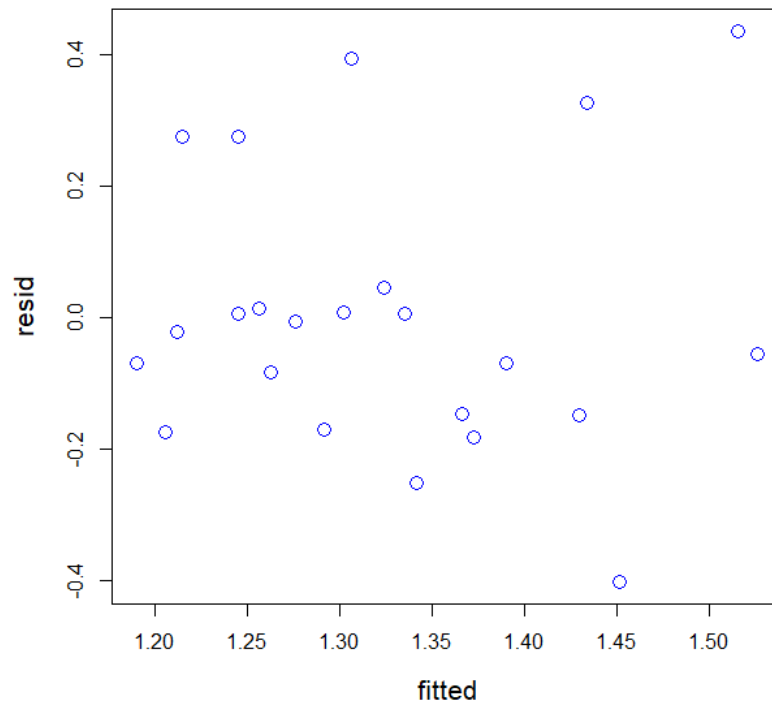
- **Shapiro-Wilks** test
- **Kolmogorov-Smirnov** test
- **Anderson-Darling** test

```
res = residuals(lm.out)
shapiro.test(res) # p-value = 0.08173
# ks.test needs notional mean and sd, use estimate
SSres0 = Syy - Sxy^2/Sxx ; s = sqrt(SSres0/(n-2))
ks.test(res, "pnorm", mean = 0, sd = s) # p-value = 0.2134
library(nortest) # for ad.test
ad.test(res) # p-value = 0.02418
```

Equal variance assumption

Plot residuals vs. fitted values:

```
fitted = fitted(lm.out)
resid  = residuals(lm.out)
plot(fitted, resid, col = "blue", cex = 1.5, cex.lab = 1.3)
```

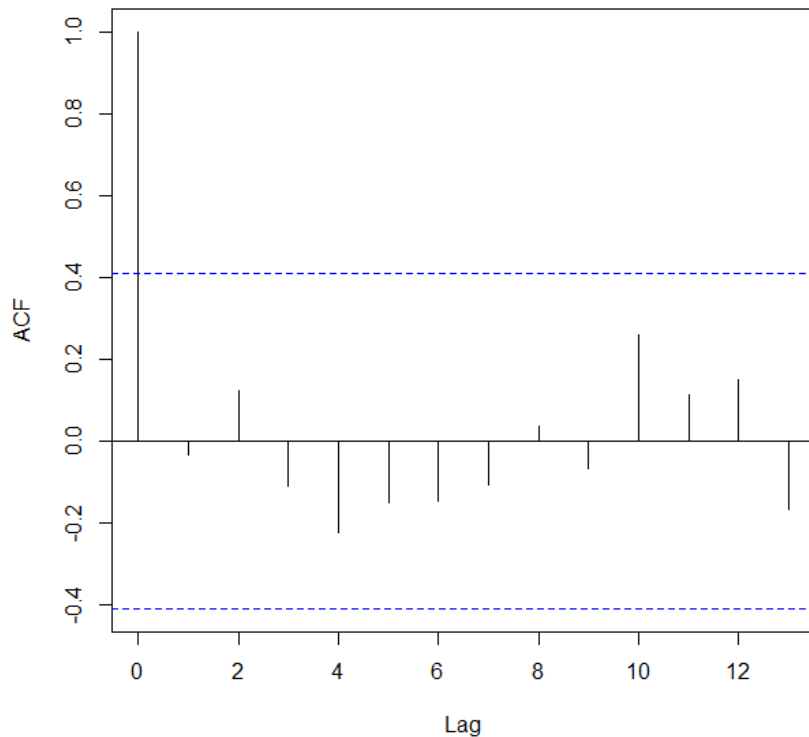


- plot should not show any structure (shape of a cone etc ...)
- `plot(lm.out)` shows more diagnostic plots

Independence of the Residuals

Plot auto-correlation of the residuals:

```
auto.cor = acf(resid(lm.out))  
plot(auto.cor, main = "")
```



Autocorrelation is the correlation of a signal with a delayed copy of itself as a function of the delay (Wikipedia). A high value of autocorrelation for some lag indicates that there is some repeated pattern in the data. Because the residuals are assumed to be independent we do **not** expect such a pattern, but expect that the autocorrelation low, i.e. within the confidence limits (-----) for all lags, except for lag 0.

Independence of the Residuals

Independence can also be **tested** in R:

- Box-Pierce or Ljung-Box test
- H_0 : independence

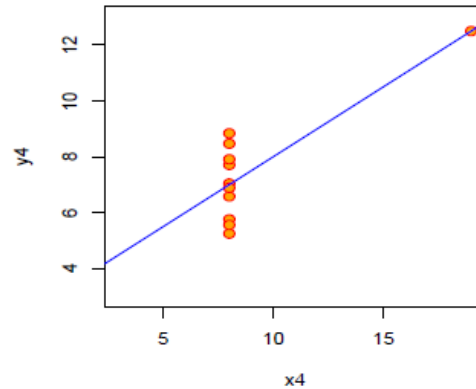
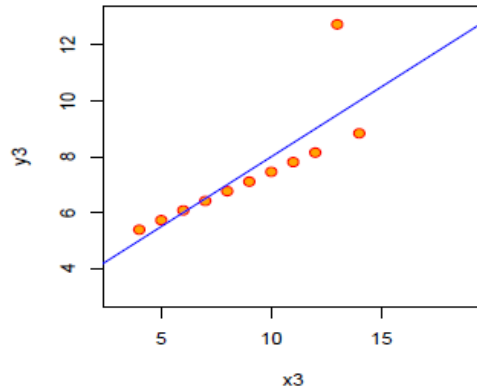
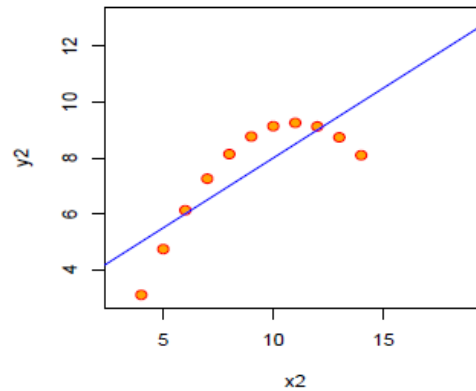
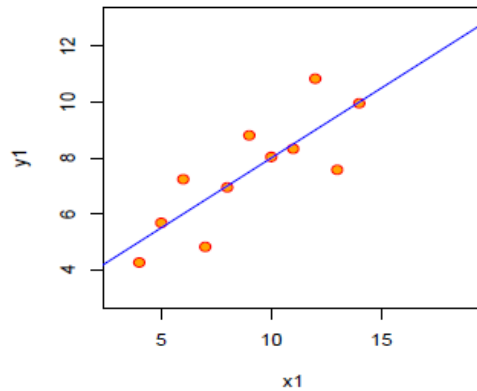
```
res = residuals(lm.out)
Box.test(res, lag = 1, type = "Box-Pierce") # p-value = 0.8718
Box.test(res, lag = 2, type = "Box-Pierce") # p-value = 0.8314
Box.test(res, lag = 1, type = "Ljung-Box")  # p-value = 0.8634
```

All p-values are ≥ 0.05 . We can **not** reject the null hypothesis of independence (on significance level 0.05). We can (pending further notice!) accept that the residuals are independent. (But keep at the back of your mind the possibility of insufficient power of the test).

`plot(lm.out)` → diagnostic plots

Plotting is necessary!

Anscombe's 4 Regression data sets



All four datasets yield the same regression line!

outlier and influential observations

Identifying Influential Observations

Identification of outliers:

- o `library(car) ; outlierTest(x)`
- o `library(mvoutlier) ; pcout(x)`

Identification of "overly" influential observations:

```
influence.measures(lm.out)
```

#	dfb.1_	dfb.bld.	dffit	cov.r	cook.d	hat	inf
# 1	-0.242084	0.413389	0.54996	0.949	1.40e-01	0.1000	
# 2	0.001439	0.000483	0.00492	1.153	1.27e-05	0.0439	
# 3	-0.004942	0.002987	-0.00642	1.167	2.16e-05	0.0555	
# 4	0.108248	-0.146100	-0.16166	1.433	1.37e-02	0.2373	*
# 5	0.015030	-0.010388	0.01755	1.181	1.62e-04	0.0669	
# 6	0.443579	-0.354659	0.46590	1.027	1.04e-01	0.1034	
# 7	0.004796	-0.001941	0.00805	1.156	3.40e-05	0.0462	

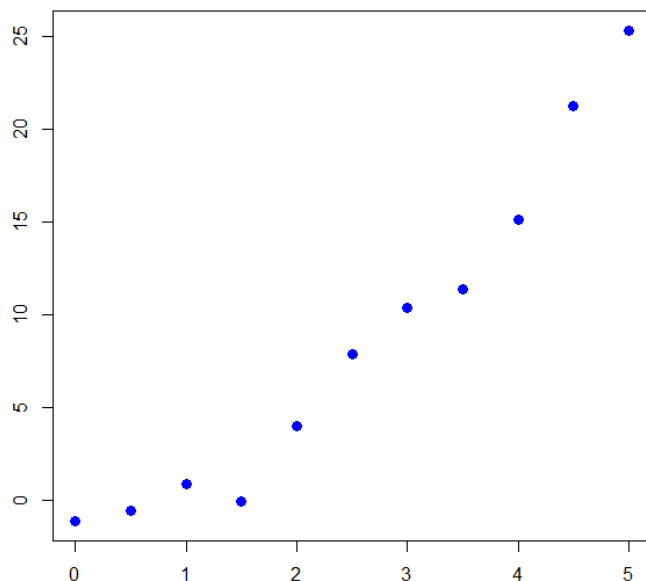
removing observation 4 would significantly change the fitted regression model → check this observation!

Non-linear terms of the independent variable

$$y_i = \alpha + \beta \underline{x_i^2} + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim N(0, \sigma)$$

see [Reg_Models_Examples.R](#)

```
x1 = seq(0, 5, len = 11)
y1 = x1^2 + rnorm(length(x1), mean = 0, sd = 1) # x2 + noise
plot(x1, y1, col = "blue", pch = 19, xlab = "", ylab = "")
```



Can we fit such a curve using
a linear model ?



Non-linear terms of the independent variable

$$y_i = \alpha + \beta x_i^2 + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim N(0, \sigma)$$

- This is still a linear model w.r.t. the coefficients α and β !
- We can cheat a little bit: replace the x_i^2 by new variable

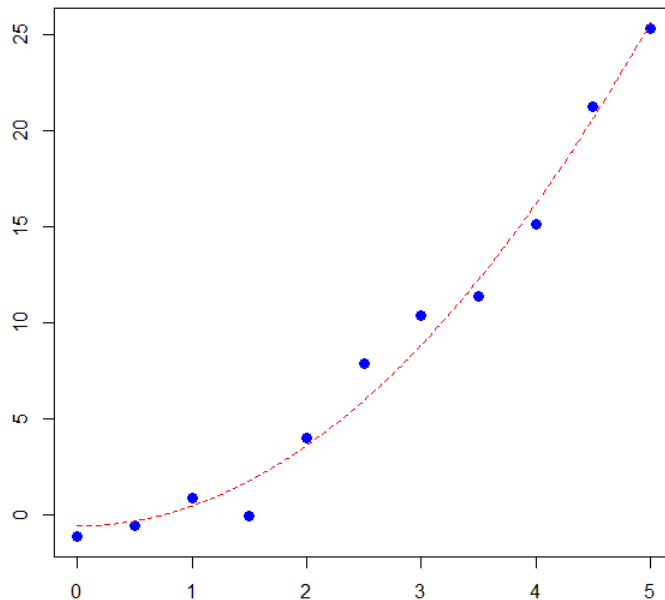
If we **substitute** $x' = x^2$, we get back to the usual form:

$$y_i = \alpha + \beta x'_i + \varepsilon_i$$

```
data = data.frame(xsq = x1^2, y = y1) # new xsq
head(data)
#      xsq      y
# 1 0.00  1.2329697
# 2 0.25 -0.5919289
# 3 1.00  2.6289114
# 4 2.25  1.7347630
# 5 4.00  3.6582234
```

Non-linear terms of the independent variable

```
lm2 = lm(y ~ xsq, data=data)           #  $y = a + b \cdot x^2 + e$   
coef(lm2)                             # 0.4511422 0.9832360  
a = coef(lm2)[1]                       # intercept  
b = coef(lm2)[2]                       # slope  
xt = seq(0, 5, len=401)  
yt = a + b*xt^2                         # regression line  
lines(xt, yt, col = "red", lty = 2, lwd = 1) # not too bad
```



works with all kinds of
functions $f(x_i)$

Non-linear terms of the independent variable

$$y_i = \alpha + \beta x_i^2 + \varepsilon_i \quad \text{with} \quad \varepsilon_i \sim N(0, \sigma)$$

In R, it is also possible to include non-linear terms **directly**:

```
data3 = data.frame(x = x1, y = y1) # the original data
plot(y ~ x, data = data3)
lm3 = lm(y ~ I(x^2), data = data3) # W-R-notation
```

The symbol $I(.)$ is important! (Wilkinson-Rogers notation)

More functions in R:

```
y ~ poly(x, ...)      # Polynom fitting
lm(log(y) ~ x)         # Regression on transformed data
res = model(y ~ x)
library(MASS)
boxcox(res)             # Find the power of y that improves the fit
```