Chapter 9: Analysis of Variance (ANOVA)

Contents

1 One-way ANOVA – Introductory example

A t-test compares means of two independent groups ($H_0 : \mu_1 = \mu_2$). **One-way analysis of variance (one-way ANOVA) is a testing procedure to compare the means of multiple groups.**

Example *Pollution*
In some cities in the USA, they collect measurements about pollution. Next variables are available:

Variable	Description
city	Name of the city (located in USA)
regio	Region in USA: W: West N: North NO: North-East ZO: South-East C: Central
JanT	Average temperature in January (in Fahrenheit)
JulT	Average temperature in July (in Fahrenheit)
Hum	Relative humidity (in percentages)
Rain	Yearly amount of rain (in inches)
Mortality	Mortality, corrected for age
Educ	Median of the education level
density	Density
NW	Percentage of non-white

Import the data set *pollutie2.txt* as `pollution`.

```
pollution <- read.table(file=file.choose(), header=TRUE)
summary(pollution)
head(pollution, n = 15)
```

We will consider the pollution example and look at the average yearly amount of rain across different regions.

1.1 Descriptive statistics

```
install.packages("psych")
library(psych)
```

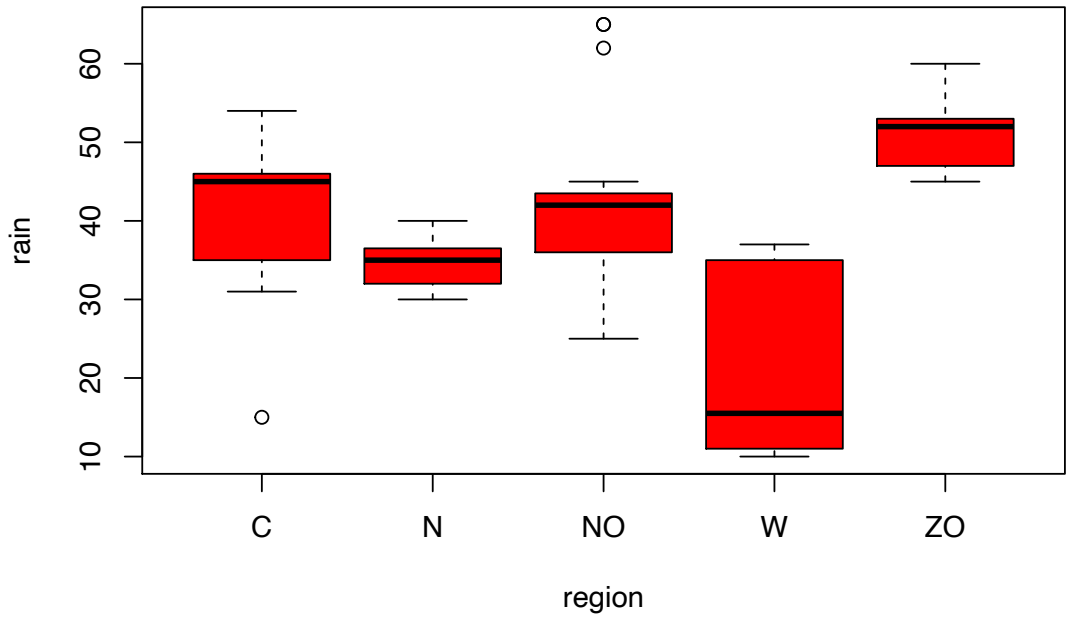
```
rain <- pollution$Rain
region <- pollution$regio

describe <- describeBy(rain, region, mat = TRUE)
describe.st <- subset(describe, select = c("group1", "n", "mean", "sd", "median", "min", "max"))
describe.st
```

```
##     group1  n     mean        sd median min max
## X11      C  9 40.55556 11.886033   45.0  15  54
## X12      N 15 34.80000  3.098387   35.0  30  40
## X13     NO 24 41.95833  9.993385   42.0  25  65
```

```
## X14      W  6 20.66667 12.209286   15.5  10  37
## X15     ZO  5 51.40000  5.856620   52.0  45  60
```

```
boxplot(rain ~ region, col = 2, names = levels(pollution$regio))
```



Remark:
Instead of using the `DescribeBy` function in the `psych` package, we can also use the `summarise` and `group_by` function from the `tidyverse` package.

Another way to obtain descriptive statistics:

```
library(tidyverse)
by_region <- group_by(pollution, regio)
summarise(by_region, Avgrain = mean(Rain))
```

```
## # A tibble: 5 x 2
##   regio Avgrain
##   <fct>   <dbl>
## 1 C        40.6
## 2 N        34.8
## 3 NO       42.0
## 4 W        20.7
## 5 ZO       51.4
```

1.2 Problem formulation

From the descriptive statistics analysis, we calculated the mean amount of rain for samples drawn from 5 different regions.

Region	Sample mean amount of rain	Population mean amount of rain
C	40.56	μ_1
N	34.80	μ_2
NO	41.96	μ_3
W	20.67	μ_4
ZO	51.40	μ_5

We want to evaluate the following questions:

1. Do the five regions have the same average amount of rain ($H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu_5$)?
2. Which groups of regions have the same average amount of rain (homogeneous groups)?

Since there are more than two independent groups, a t-test cannot be used to compare the means. Instead, one-way ANOVA will be used to assess these questions.

2 F-test for multiple means in one-way ANOVA

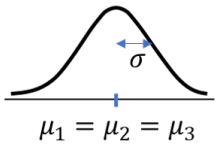
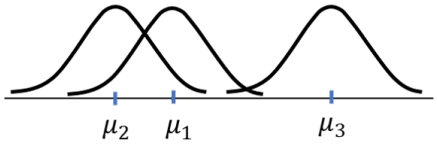
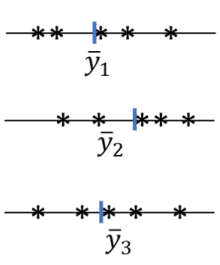
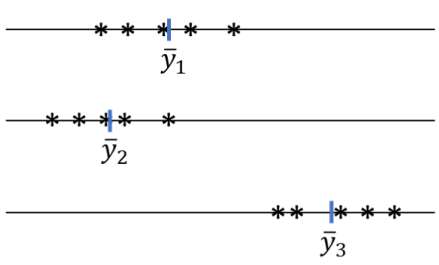
2.1 ANOVA – testing principle

The one-way ANOVA compares the means between different groups. If there are r number of groups, then the ANOVA tests the following **hypothesis** :

$H_0 : \mu_1 = \mu_2 = ... = \mu_r$
versus
H_1 : **the means are not all the same**

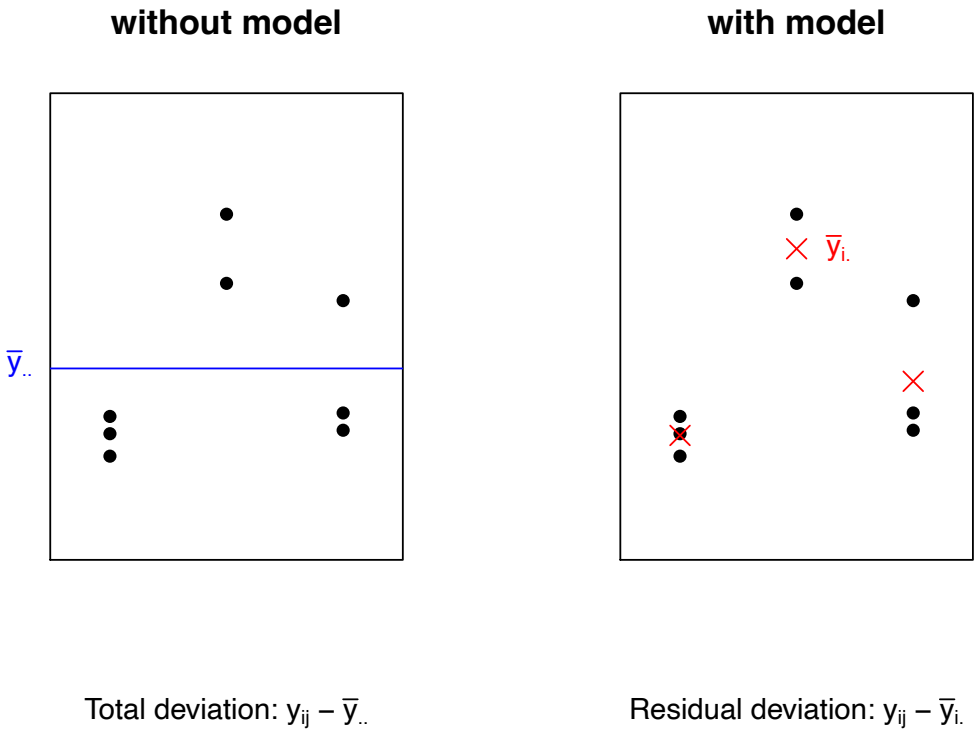
with μ_i the population group mean of group i.

Consider a situation where the mean of three different groups are compared. ANOVA is used to test the null hypothesis $\boldsymbol{H_0: \mu_1 = \mu_2 = \mu_3}$ against the alternative hypothesis that the means are not all the same.
We assume that the three groups correspond to 3 normally distributed populations with same variance σ^2

	H_0 true	H_0 not true
Population		
Sample From population 1: From population 2: From population 3:		

Testing principle: Reject H_0 if the variability of $\overline{y}_i$ is too big compared to the within-group variance σ^2.

Partitioning of the variances



Partitioning for observation ij:

$$\begin{gathered}(y_{ij} - \overline{y}_{\bullet\bullet}) = (y_{ij} - \overline{y}_{i\bullet}) + (\overline{y}_{i\bullet} - \overline{y}_{\bullet\bullet}) \\ (\text{deviation of observation } ij) = (\text{random error}) + (\text{effect from group } i)\end{gathered} \tag{1}$$

with

- y_{ij} the value of the observation ij
- $\overline{y}_{\bullet\bullet}$ the mean of all the observations of all the groups
- $\overline{y}_{i\bullet}$ the mean of all the observation in group i

Sum of squares (SS) decomposition:

$$\begin{gathered}\sum_{i=1}^{r}\sum_{j=1}^{n_j}(y_{ij} - \overline{y}_{\bullet\bullet})^2 = \sum_{i=1}^{r}\sum_{j=1}^{n_j}(y_{ij} - \overline{y}_{i\bullet})^2 + \sum_{i=1}^{r} n_i(\overline{y}_{i\bullet} - \overline{y}_{\bullet\bullet})^2 \\ (\text{SS total}) = (\text{SS residual}) + (\text{SS treatment}) \\ (\text{SS total}) = (\text{SS within}) + (\text{SS between})\end{gathered} \tag{2}$$

The corresponding degrees of freedom:

$$n - 1 = (n - r) + (r - 1)$$

with

- *SS total* = total sum of squares

- *SS between* = between group of squares, caused by the difference between the groups (treatment effect)
- *SS within* = within group of squares, caused by the variation within each group (residual part of the total SS).
- $n = \sum_i n_i$ = total number of observations
- r = number of groups

Mean sum of squares = sum of squares divided by the degrees of freedom:

MS total = $\frac{\text{SS total}}{n-1}$
MS between = $\frac{\text{SS between}}{r-1}$
MS within = $\frac{\text{SS within}}{n-r}$

Test statistic used to test $H_0 : \mu_1 = \mu_2 = ... = \mu_r$ versus H_1**: Not all population averages are the same** is the F-statistic:
$F = \frac{\text{MS between}}{\text{MS within}} \approx F_{r-1,n-r}$

Conclusion: reject the null hypothesis if F is too big (small p-value)

2.2 Example

Example *Pollution*
What we have just discussed will now be applied to the amount of rain (which is a variable from the data set `pollution`).

```
glm1 <- lm(Rain  ~ regio, data = pollution)
summary(glm1)
```

```
##
## Call:
## lm(formula = Rain ~ regio, data = pollution)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.5556  -4.6000   0.0417   2.7431  23.0417
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   40.556      3.024  13.411  < 2e-16 ***
## regioN        -5.756      3.825  -1.505 0.138226
## regioNO        1.403      3.546   0.396 0.693954
## regioW       -19.889      4.781  -4.160 0.000115 ***
## regioZO       10.844      5.060   2.143 0.036623 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.072 on 54 degrees of freedom
## Multiple R-squared:  0.4239, Adjusted R-squared:  0.3813
## F-statistic: 9.935 on 4 and 54 DF,  p-value: 4.245e-06
```

The average amount of region of the 5 different regions are compared with a **F-test in one-way ANOVA**.
$p - value$ (one-sided) $= 0.000004 < 0.05$

Conclusion: reject H_0. The average amount of rain in these five regions are not the same. The factor `regio` has an effect on the amount of rain. However, we do not know where the differences are.

3 One-way ANOVA as a linear model

One-way ANOVA can be seen as a linear model with a continuous response variable and a categorical explanatory variable (with multiple levels).
One-way ANOVA test $H_0 : \mu_1 = \mu_2 = ... = \mu_r$ versus H_1**: Not all the means are the same**.

3.1 Use of treatment coding

We will demonstrate the treatment coding by using the an example.

Example *Pollution*
We know, from the descriptive statistics performed earlier, the average amount of rain for the different regions:

Regions	C	N	NO	W	ZO
Average	40.56	34.80	41.96	20.67	51.40

These numbers can be checked by the using the code

Interpretation of the parameter estimates of the output of `summary(glm1)`.

Treatment coding

$$
\begin{aligned}
&Rain = \alpha + \beta_1 D_1 + \beta_2 D_2 + \beta_3 D_3 + \beta_4 D_4 + \varepsilon \\
&\text{with:} \\
&\quad D_1 = 1 \text{ if } region = N \\
&\qquad = 0 \text{ else} \\
&\quad D_2 = 1 \text{ if } region = N0 \\
&\qquad = 0 \text{ else} \\
&\quad D_3 = 1 \text{ if } region = W \\
&\qquad = 0 \text{ else} \\
&\quad D_4 = 1 \text{ if } region = ZO \\
&\qquad = 0 \text{ else}
\end{aligned}
\tag{3}
$$

The expected values for the different regions:

$$
\begin{aligned}
&E(Rain_N) = \alpha + \beta_1 \\
&E(Rain_{NO}) = \alpha + \beta_2 \\
&E(Rain_W) = \alpha + \beta_3 \\
&E(Rain_{ZO}) = \alpha + \beta_4 \\
&E(Rain_C) = \alpha
\end{aligned}
\tag{4}
$$

- α = the average amount of rain in C
- β_i = difference in average amount of rain in region i compared to C

$\rightarrow$ Hence, region C can be considered as the reference category.

The output of `summary(glm1)` estimates these parameters:
$\hat{\alpha} = 40.56$
$\hat{\beta}_1 = -5.76$
$\hat{\beta}_2 = 1.40$
$\hat{\beta}_3 = -19.89$
$\hat{\beta}_4 = 10.84$

Looking back to the means given in the descriptive statistics, we see that the estimated value corresponds to the observed means. For instance for the region N, we have $\hat{\alpha} + \hat{\beta}_1 = 40.56 - 5.76 = 34.80$.

In this model, the null hypothesis $H_0 : \mu_1 = \mu_2 = \mu_3 = \mu_4 = \mu_5$ is equivalent to $H_0 : \beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$.

In R we can ask for the design matrix (only part of the output is given here)

```
model.matrix(glm1)
```

```
##   (Intercept) regioN regioNO regioW regioZO
## 1           1      1       0      0       0
## 2           1      0       1      0       0
## 3           1      0       1      0       0
## 4           1      0       0      0       1
```

3.2 Use of sum coding

When running a linear regression, R will use the *treatment coding* by default. The type of coding can be changed to ***sum coding*** using (in the function `lm`) the argument `contrasts = list(Name_variable = "contr.sum")` and replacing `Name_variable` by the name of the variable being coded.

Example *Pollution*

```
glm2 <- lm(Rain ~ regio, data = pollution, contrasts = list(regio = "contr.sum"))
summary(glm2)
```

```
##
## Call:
## lm(formula = Rain ~ regio, data = pollution, contrasts = list(regio = "contr.sum"))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.5556  -4.6000   0.0417   2.7431  23.0417
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   37.876      1.389  27.268  < 2e-16 ***
## regio1         2.679      2.723   0.984   0.3295
## regio2        -3.076      2.285  -1.346   0.1839
## regio3         4.082      1.997   2.044   0.0458 *
## regio4       -17.209      3.187  -5.399 1.53e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.072 on 54 degrees of freedom
## Multiple R-squared:  0.4239, Adjusted R-squared:  0.3813
## F-statistic: 9.935 on 4 and 54 DF,  p-value: 4.245e-06
```

Regions	C	N	NO	W	ZO
Average	40.56	34.80	41.96	20.67	51.40

Interpretation of the parameter estimates of the output of `summary(glm2)`:

Sum coding

$$
\begin{aligned}
&Rain = \alpha + \beta_1 D_1 + \beta_2 D_2 + \beta_3 D_3 + \beta_4 D_4 + \varepsilon \\
&\text{with:} \\
&\quad D_1 = 1 \text{ if } region = C \\
&\qquad = -1 \text{ if } region = ZO \\
&\qquad = 0 \text{ else} \\
&\quad D_2 = 1 \text{ if } region = N \\
&\qquad = -1 \text{ if } region = ZO \\
&\qquad = 0 \text{ else} \\
&\quad D_3 = 1 \text{ if } region = NO \\
&\qquad = -1 \text{ if } region = ZO \\
&\qquad = 0 \text{ else} \\
&\quad D_4 = 1 \text{ if } region = W \\
&\qquad = -1 \text{ if } region = ZO \\
&\qquad = 0 \text{ else}
\end{aligned}
\tag{5}
$$

The expectation values for the different regions:

$$
\begin{aligned}
&E(Rain_C) = \alpha + \beta_1 \\
&E(Rain_N) = \alpha + \beta_2 \\
&E(Rain_{NO}) = \alpha + \beta_3 \\
&E(Rain_W) = \alpha + \beta_4 \\
&E(Rain_{ZO}) = \alpha - \beta_1 - \beta_2 - \beta_3 - \beta_4
\end{aligned}
\tag{6}
$$

- α = the average of the group averages
- β_i = difference in average amount of rain in region i compared to global average

The output of `summary(glm1)` estimates these parameters:
$\hat{\alpha} = 37.88$
$\hat{\beta}_1 = 2.68$
$\hat{\beta}_2 = -3.08$
$\hat{\beta}_3 = 4.08$
$\hat{\beta}_4 = -17.21$

The expected amount of rain in the region ZO is:
$E(Rain_{ZO}) = 37.88 + 2.68 \cdot (-1) - 3.08 \cdot (-1) + 4.08 \cdot (-1) - 17.21 \cdot (-1) = 51.4$

The model matrix can be returned by using `model.matrix(glm2)`.

```
head(model.matrix(glm2))
```

```
##   (Intercept) regio1 regio2 regio3 regio4
## 1           1      0      1      0      0
## 2           1      0      0      1      0
## 3           1      0      0      1      0
## 4           1     -1     -1     -1     -1
## 5           1      0      0      1      0
## 6           1     -1     -1     -1     -1
```

4 Model diagnostics

4.1 Assumptions made

ANOVA make the following **assumptions**:

- **Assumption of normality**: Each group sample is drawn from a normally distributed population
- **Assumption of homogeneity of variance**: Different samples have the same variance irrespective if they come from the same population or not
- **Assumption of independence**: The observations between groups should be independent and the observations within each group must be independent.

The first two assumptions in symbolic notation:
$Y_{1j} \sim N(\mu_1, \sigma^2)$
$Y_{2j} \sim N(\mu_2, \sigma^2)$
...
$Y_{rj} \sim N(\mu_r, \sigma^2)$

Besides, it is assumed that the observations are sampled randomly. The presence of outliers can also cause problems and should therefore be check.

4.2 Checking assumptions

Check the model conditions:

1. Independent groups?
   → According to the design of the experiment.
2. Constant within-group variance?
   → Test $H_0 : \sigma_1^2 = \sigma_2^2 = ... = \sigma_r^2$
   - Visual check of the boxplot
   - Test for identical variances in a number of groups (e.g., Levene test and Bartlett test)
3. Normal distribution in the groups?
   → Shapiro-Wilk test per group
   Or
   → Shapiro-Wilk test of the within-group residuals + histogram of within-group residuals
4. Presence of influential observations
   → Cook's distance

4.3 Check assumption of homogeneity of variance

$H_0 : \sigma_1^2 = \sigma_2^2 = ... = \sigma_r^2$
versus
H_1 : **not all** σ_i^2 **are equal** $(i = 1, 2, ..., r)$

We can use the Levene's test to test the homogeneity of variance assumption.

In R, this test can be performed by the function `leveneTest` from the package `car`.

```
library(car)
leveneTest(rain ~ region)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  4  1.6305 0.1799
##       54
```

We can assume homogeneity of variances since $p - value > 0.05$.

Remark 1:
If the Levene's test is rejected, be aware that there exists some robustness. If the variances are not too unequal, we can still make use of the ANOVA F-test.
Rule of thumb: $\frac{\max(\sigma_1,\sigma_2,\ldots,\sigma_r)}{\min(\sigma_1,\sigma_2,\ldots,\sigma_r)} \leq 5$

Remark 2:
In case there is no homogeneity of variances, a modification of the F-test can be used:

```
oneway.test(Rain ~ regio, data = pollution, var.equal = FALSE)
```

```
##
##  One-way analysis of means (not assuming equal variances)
##
## data:  Rain and regio
## F = 12.07, num df = 4.000, denom df = 14.202, p-value = 0.0001751
```

4.4 Check assumption of normality

We assume
$Y_{1j} \sim N(\mu_1, \sigma^2) \quad \Rightarrow \quad Y_{1j} - \mu_1 \sim N(0, \sigma^2)$
$Y_{2j} \sim N(\mu_2, \sigma^2) \quad \Rightarrow \quad Y_{2j} - \mu_2 \sim N(0, \sigma^2)$
...
$Y_{rj} \sim N(\mu_r, \sigma^2) \quad \Rightarrow \quad Y_{rj} - \mu_r \sim N(0, \sigma^2)$

Hence, we can check normality for the within-group residuals!

1. What are the residuals in our example?

```
by_region <- group_by(pollution, regio)
summarise(by_region, n = n(), mean = mean(Rain), sd = sd(Rain), median = median(Rain),
          min = min(Rain), max = max(Rain))
```

```
## # A tibble: 5 x 7
##   regio     n  mean    sd median   min   max
##   <fct> <int> <dbl> <dbl>  <dbl> <int> <int>
## 1 C         9  40.6 11.9    45      15    54
## 2 N        15  34.8  3.10   35      30    40
## 3 NO       24  42.0  9.99   42      25    65
## 4 W         6  20.7 12.2    15.5    10    37
## 5 ZO        5  51.4  5.86   52      45    60
```

Look at the first 5 observations of the pollution data set

```
head(pollution, n = 5)
```

```
##        city regio JanT JulT Hum Rain Mortality Educ density   NW
## 1     Akron     N   27   71  59   36    921.87 11.4    3243  8.8
## 2    Albany    NO   23   72  57   35    997.87 11.0    4281  3.5
## 3 Allentown    NO   29   74  54   44    962.35  9.8    4260  0.8
## 4   Atlanta    ZO   45   79  56   47    982.29 11.1    3125 27.1
## 5 Baltimore    NO   35   77  55   43   1071.29  9.6    6441 24.4
```

Compute the corresponding residuals by yourself

City	Residual
Akron	
Albany	
Allentown	

City	Residual
Atlanta	
Baltimore	

Compare your results with the residuals obtained in R

```
glm1 <- lm(Rain  ~ regio, data = pollution)
combine <- data.frame(city = pollution$city, region = pollution$regio,
                      residuals = glm1$residuals)
head(combine, n = 5)
```

```
##        city region residuals
## 1     Akron      N  1.200000
## 2    Albany     NO -6.958333
## 3 Allentown     NO  2.041667
## 4   Atlanta     ZO -4.400000
## 5 Baltimore     NO  1.041667
```

2. How to test normality of these residuals in R?

Test normality of the within-group residuals:

```
shapiro.test(glm1$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  glm1$residuals
## W = 0.94388, p-value = 0.008825
```

```
hist(glm1$residuals)
```

Remark:

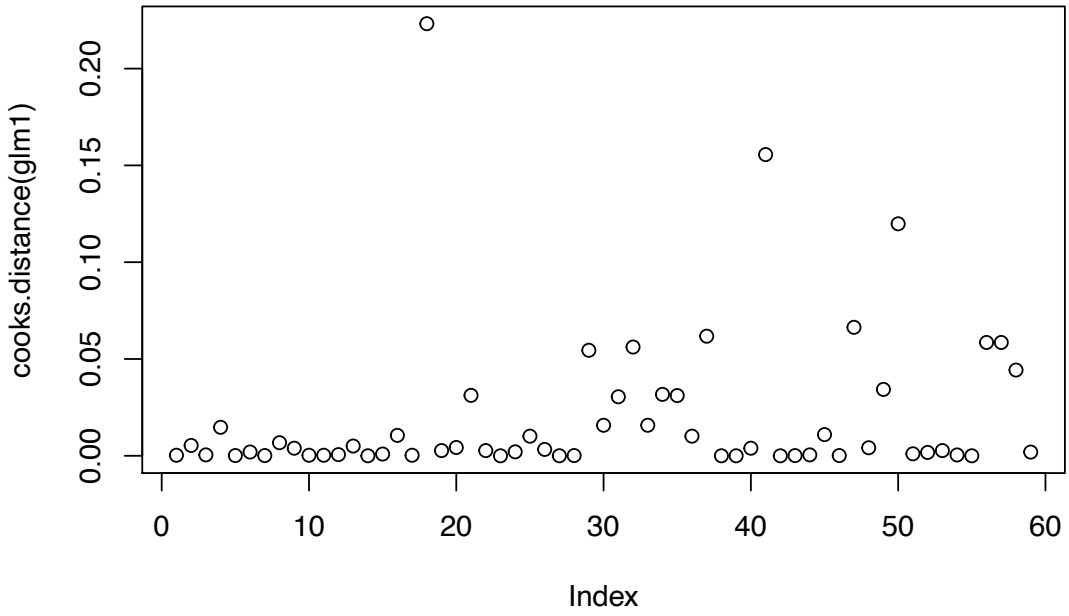
- When normality is rejected by the Shapiro-Wilk test, we still can interpret the ANOVA F statistic as long as the residuals are symmetric distributed (seen from the histogram).
- In case of asymmetric distribution of residuals, we can transform the response variable Y.
- In case of asymmetry, we can always use the non-parametric version: the Kruskal-Wallis test.

4.5 Checking for influential observations

As in regression analysis, we use the **Cook's distance** to check for influential observations.

We plot the Cook's distance versus observation number

```
plot(cooks.distance(glm1))
```



There are no influential observations.

5 Pairwise comparisons of treatment effects

5.1 Questions post-hoc

We rejected the null hypothesis $H_0 : \mu_N = \mu_{NO} = \mu_W = \mu_{ZO} = \mu_C$ with μ_i the average amount of rain in region i. The rejection was made based on the results of a F-test in one-way ANOVA. (see earlier).

We now raise some questions post-hoc:

1. Is there a pair of regions with the same average amount of rain?
2. Test all pairs of regions on the same average amount of rain.

If you want to do multiple comparisons in R, you have to first use the function `aov` and store the results in an object.

```
poll.aov1 <- aov(Rain ~ regio, data = pollution)
summary(poll.aov1)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## regio        4   3270   817.6   9.935 4.24e-06 ***
## Residuals   54   4444    82.3
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

5.2 Planned comparison of means in ANOVA

When performing an ANOVA test and a significant result is obtained, then the null hypothesis of equal means is rejected. However, an ANOVA test cannot tell which group differs. To address this problem, the **Least Significant Difference** (LSD-method) method can be used to test a planned comparison (between two groups). In the LSD test, the mean of one group is compared to the mean of another group. The LSD test is basically a t-test for two means in ANOVA. The main difference with a regular t-test is that the standard deviation estimate is based on the observations of all the groups. In a regular t-test, only the observations from the two groups under consideration are used to estimate the standard deviation.

Consider r groups with population means $\mu_1, \mu_2, ..., \mu_r$. Assume an ANOVA test is performed and suggest rejection of the null hypothesis, that is, the difference between the group means is significant. We now want to check whether the mean of group 1 and group 2 are significantly different from each other. This is checked with a LSD test.

Test problem:
$H_0 : \mu_1 = \mu_2$ (group 1 and group 2 have the same mean)
$H_1 : \mu_1 \neq \mu_2$ (group 1 and group 2 have a different mean)

Test statistic:
$t = \frac{\overline{X}_1 - \overline{X}_2}{\sqrt{s^2\left(\frac{1}{n_1}\frac{1}{n_2}\right)}} \sim t_{n-r}$ (distribution under H_0)
with $s^2 = MS\ within = \frac{SS\ within}{n-r} = \frac{\sum_{i=1}^{r}\sum_{j=1}^{n_j}(Y_{ij}-\overline{Y}_{i\bullet})^2}{n-r}$ the pooled estimator of the variance.

Conclusion:
Reject the null hypothesis and accept a difference in effect between the two groups if the test statistic t is too big or too small (two-sided $p - value$).

This test method is also known as the LSD-method (Least Significant Difference method).

To compare two groups in ANOVA:

- In case there is homogeneity of variances, use the LSD method in ANOVA (with $n - r$ degrees of freedom, see above).
- In case there is no homogeneity of variances, use the two-sample t-test for independent groups (with $n_1 + n_2 - 2$ degrees of freedom).

The usual t-test for two groups estimates the within-group variance σ^2 based on the pooled sample variance of the two groups. For each pair of groups, one uses a different variance estimator, which is not logical because all groups have the same variance. The t-test above for two groups in ANOVA (thus the LSD test) uses information from all groups – the pooled sample variance based on the r groups – to estimate σ^2. Thus this test will make less wrong decisions. On the other hand, if there are indications that not all groups have the same variance, one should prefer the usual t-test for two groups.

Example *Pollution* in R
We want to test whether the mean amount of rain in region C and region NO are the same or not.

Statement hypothesis:
$H_0 : \mu_{NO} = \mu_C$ versus $H_1 : \mu_{NO} \neq \mu_C$

```
pairwise.t.test(rain, region, p.adjust.method = "none")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  rain and region
##
##    C       N       NO      W
## N  0.13823 -       -       -
```

```
## NO 0.69395 0.02000 -       -
## W  0.00011 0.00214 3.9e-06 -
## ZO 0.03662 0.00082 0.03887 7.5e-07
##
## P value adjustment method: none
```

Conclusion:
There is no significant difference in the average amount of rain between the region NO and C.

5.3 Multiple comparisons

5.3.1 The problem of multiple comparisons

A planned test versus multiple test
If the groups, one wants to compare, are determined before one looks at the data, then the LSD test is suitable.

Risk on test differences which in reality are not present
If one compares each pair of means using the LSD test with significance level $\alpha = 0.05$, there is a *high probability that one falsely reports differences.*
(remember: $\alpha = P(reject\ H_0 | H_0\ true)$)

- Even with a small number of groups, many pairs are possible, i.e.

$$\binom{r}{2} = \frac{r(r-1)}{2} \tag{7}$$

- A t-test for two means ($\mu_1 = \mu_2$ versus $\mu_1 \neq \mu_2$) with significance level $\alpha = 0.05$ will report a false difference in 5% of applications (or 1 on 20 samples) on populations with the same mean. (e.g. a clinical study placebo/medication: a medicine without effect will be concluded as effective in 5% of tests, due to the coincidental structure of the sample.)

Example: ANOVA with $r = 10$ groups, gives $\frac{10 \cdot 9}{2} = 45$ pairs. If there are no population differences, and one tests each pair with a LSD test on significance level $\alpha = 0.05$, one can expect that in these samples, $5\% \cdot 45 = 2$ pairs will be considered different, while in the populations they are not. (Roughly, because the successive tests are not statistical independent experiments.)

Problem: find a procedure that, if there are no differences, using multiple comparisons for all pairs, keeps the overall probability of falsely reporting at least one difference, less than a certain chosen significance level α.

5.3.2 An overview of multiple comparisons procedures

1. **Tukey HSD (honestly significant differences) test for multiple pairs: all pairwise comparisons**
   - Test similarity of pairs: $H_0 : \mu_i = \mu_j$ (all pairs are equal)
     Reports the differences pairwise $\mu_i \neq \mu_j$
   - Advantage: The overall significance level is exactly α
2. **Bonferroni method for multiple tests**
   - Test similarity of pairs: $H_0 : \mu_i = \mu_j$
   - Per pair t-test with reduced significance level: $\alpha^* = \frac{\alpha}{\text{number of pairs}}$
   - Disadvantage: overall significance level $\leq \alpha$, even $< \alpha$ (test is conservative for H_0)
   - Advantage: you can use it with a small number of groups
3. **Scheffé multiple contrasts test: all linear contrasts**
   - $H_0 : c_1\mu_1 + c_2\mu_2 + ... + c_r\mu_r = 0$ (given that $\sum_{i=1}^{r} c_i = 0$) (all linear contrasts 0)
   - Disadvantage: overall significance level $\leq \alpha$ (test sometimes conservative for H_0)
4. **Holm's step-wise correction for multiple tests**

- Test similarity of pairs: $H_0 : \mu_i = \mu_j$
- The Holm adjustment sequentially compares the lowest $p - value$ with a type I error rate that is reduced for each consecutive test. This method is generally considered superior to the Bonferroni adjustment.

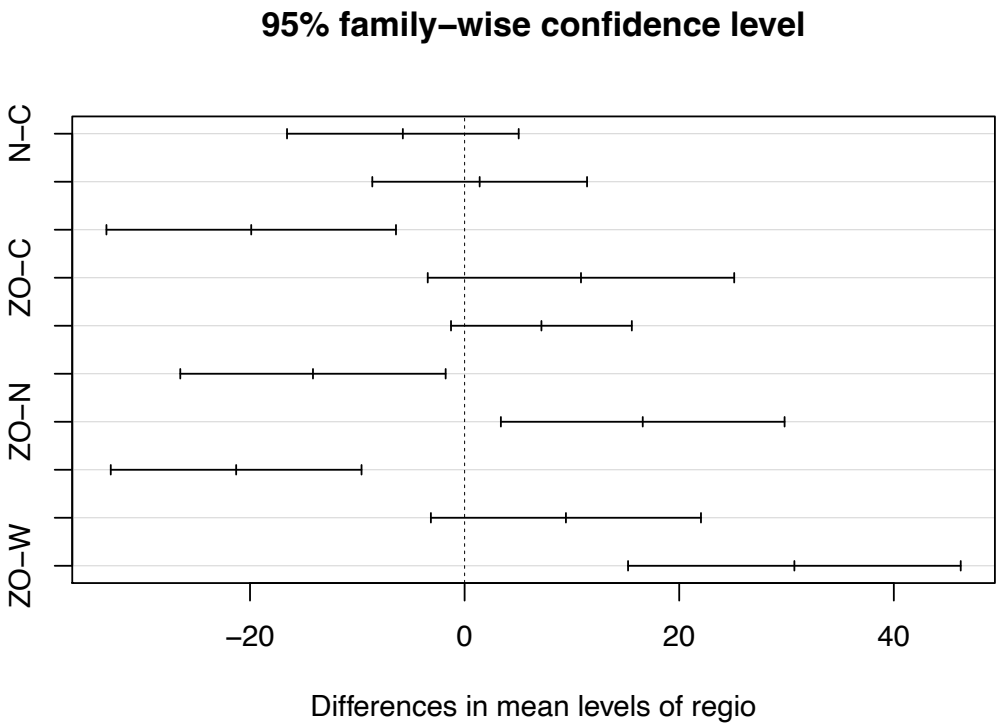
In R

Tukey HSD test

```
diffs <- TukeyHSD(poll.aov1, whih = "region", conf.level = 0.95)
diffs
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Rain ~ regio, data = pollution)
##
## $regio
##             diff        lwr       upr     p adj
## N-C    -5.755556 -16.550086  5.038975 0.5639861
## NO-C    1.402778  -8.604020 11.409575 0.9946762
## W-C   -19.888889 -33.382052 -6.395726 0.0010501
## ZO-C   10.844444  -3.435377 25.124266 0.2172902
## NO-N    7.158333  -1.268144 15.584811 0.1316844
## W-N   -14.133333 -26.500021 -1.766645 0.0174879
## ZO-N   16.600000   3.379454 29.820546 0.0070724
## W-NO  -21.291667 -32.977089 -9.606245 0.0000371
## ZO-NO   9.441667  -3.143918 22.027251 0.2278970
## ZO-W   30.733333  15.230869 46.235797 0.0000073
```

```
plot(diffs)
```



If the value 0 is not included in the 95 % CI then there is a significant difference between the average values of the two groups.

The adjusted $p - value$ returned by the `aov` function depends on assumptions of the residuals. For the $p - value$ to be correct, these residuals need to be independent, normally distributed, and have constant variance. In the following section, we see a non-parametric function that does not require the normality assumption.

Remark 1:
You can use the `scheffe.test` function (from package `agricolae`) for Scheffé multiple comparisons and the function `pairwise.t.test` for the Bonferroni multiple comparisons. There exist also a HSD function which is similar to Tukey.

```
library(agricolae)
```

```
poll.aov1 <- aov(Rain ~ regio, data = pollution)
```

another HSD test

```
HSD.test(poll.aov1, "regio", group = FALSE)$comparison
```

```
##         difference pvalue signif.        LCL        UCL
## C - N     5.755556 0.5640          -5.038975  16.550086
## C - NO   -1.402778 0.9947         -11.409575   8.604020
## C - W    19.888889 0.0011       **   6.395726  33.382052
## C - ZO  -10.844444 0.2173         -25.124266   3.435377
## N - NO   -7.158333 0.1317         -15.584811   1.268144
## N - W    14.133333 0.0175        *   1.766645  26.500021
## N - ZO  -16.600000 0.0071       ** -29.820546  -3.379454
```

```
## NO - W   21.291667 0.0000     ***   9.606245  32.977089
## NO - ZO  -9.441667 0.2279         -22.027251   3.143918
## W - ZO  -30.733333 0.0000     *** -46.235797 -15.230869
```

Scheffé multiple contrasts test

```
scheffe.test(poll.aov1, "regio", group = FALSE)$comparison
```

```
##         Difference pvalue sig         LCL        UCL
## C - N     5.755556 0.6883      -6.4436228  17.954734
## C - NO   -1.402778 0.9970     -12.7117188   9.906163
## C - W    19.888889 0.0042  **   4.6399160  35.137862
## C - ZO  -10.844444 0.3439     -26.9824405   5.293552
## N - NO   -7.158333 0.2344     -16.6813139   2.364647
## N - W    14.133333 0.0461   *   0.1574188  28.109248
## N - ZO  -16.600000 0.0215   * -31.5408811  -1.659119
## NO - W   21.291667 0.0002 ***   8.0856687  34.497665
## NO - ZO  -9.441667 0.3565     -23.6649617   4.781628
## W - ZO  -30.733333 0.0000 *** -48.2530694 -13.213597
```

Bonferroni method

```
pairwise.t.test(rain, region, p.adj="bonferroni")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  rain and region
##
##    C      N      NO      W
## N  1.0000 -      -       -
## NO 1.0000 0.2000 -       -
## W  0.0011 0.0214 3.9e-05 -
## ZO 0.3662 0.0082 0.3887  7.5e-06
##
## P value adjustment method: bonferroni
```

Remark 2: Homogeneous groups

Scheffé, homogeneous groups

```
Hgroups.scheffe <- scheffe.test(poll.aov1, "regio", group = TRUE)
Hgroups.scheffe$groups
```

```
##        Rain groups
## ZO 51.40000      a
## NO 41.95833     ab
## C  40.55556     ab
## N  34.80000      b
## W  20.66667      c
```

Remark 3: Holm's approach
Illustration of the Holm's approach

1. We compute the non-adjusted $p-values$ for every hypothesis (In this example: 10 different pairs so 10 $p-values$)

```
pairwise.t.test(rain, region, p.adj = "none")
```

```
##
```

```
##  Pairwise comparisons using t tests with pooled SD
##
## data:  rain and region
##
##    C       N       NO      W
## N  0.13823 -       -       -
## NO 0.69395 0.02000 -       -
## W  0.00011 0.00214 3.9e-06 -
## ZO 0.03662 0.00082 0.03887 7.5e-07
##
## P value adjustment method: none
```

2. Compare the smallest $p-value$ with $\frac{0.05}{10} = 0.005$
   This is $7 \cdot 10^{-7} < 0.005$. Hence this null hypothesis is rejected. There is a significant difference in average rain between ZO and W.
3. Compare the second smallest $p-value$ to $\frac{0.05}{9} = 0.0056$.
   This is $3.9 \cdot 10^{-6} < 0.0056$. Hence this null hypothesis is rejected. There is a significant difference in average rain between W and NO.
4. Compare the third smallest $p-value$ to $\frac{0.05}{8} = 0.0063$.
   This is $0.00011 < 0.0063$. Hence this null hypothesis is rejected. There is a significant difference in average rain between W and N.
5. Compare the fourth smallest $p-value$ to $\frac{0.05}{7} = 0.0071$
   This is $0.0008 < 0.0071$. Hence the corresponding null hypothesis is rejected.
6. Compare the fifth smallest $p-value$ to $\frac{0.05}{6} = 0.0083$
   This is $0.002 < 0.0083$. Hence the corresponding null hypothesis is rejected.
7. Compare the sixth smallest $p-value$ to $\frac{0.05}{5} = 0.01$
   This is $0.02 > 0.01$. Hence the corresponding null hypothesis is not rejected. As soon as that happens, you stop, and therefore, also fail to reject the remaining hypothesis

Holm's approach in R

```
pairwise.t.test(rain, region, p.adj = "holm")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  rain and region
##
##    C       N       NO      W
## N  0.27645 -       -       -
## NO 0.69395 0.10001 -       -
## W  0.00092 0.01283 3.5e-05 -
## ZO 0.14649 0.00577 0.14649 7.5e-06
##
## P value adjustment method: holm
```

Remark 4: General remark about the use of the `aov` function in R
Only factors can be used in ANOVA. The `aov` function really needs the explanatory variables to be a factor. An error is returned when an explanatory variable $var1$ is not a factor. This can be solved by making variable $var1$ as a factor $var1.f$ (use function `as.factor()`).

```
var1.f <- as.factor(var1)
```

6 Extensions

6.1 The Kruskal-Wallis test

The **Kruskal-Wallis test** is a **non-parametric version of one-way analysis of variance**. The assumption underlying this test is that the measurements come from a continuous distribution, but not necessarily a normal distribution. The test is based on an analysis of variance using the ranks of the data values, not the data values.

Statement of hypothesis:
H_0 : **The location parameters of the distribution of X are the same in each group.**
versus
H_1 : **The location parameters differ in at least one group.**

Test statistic:

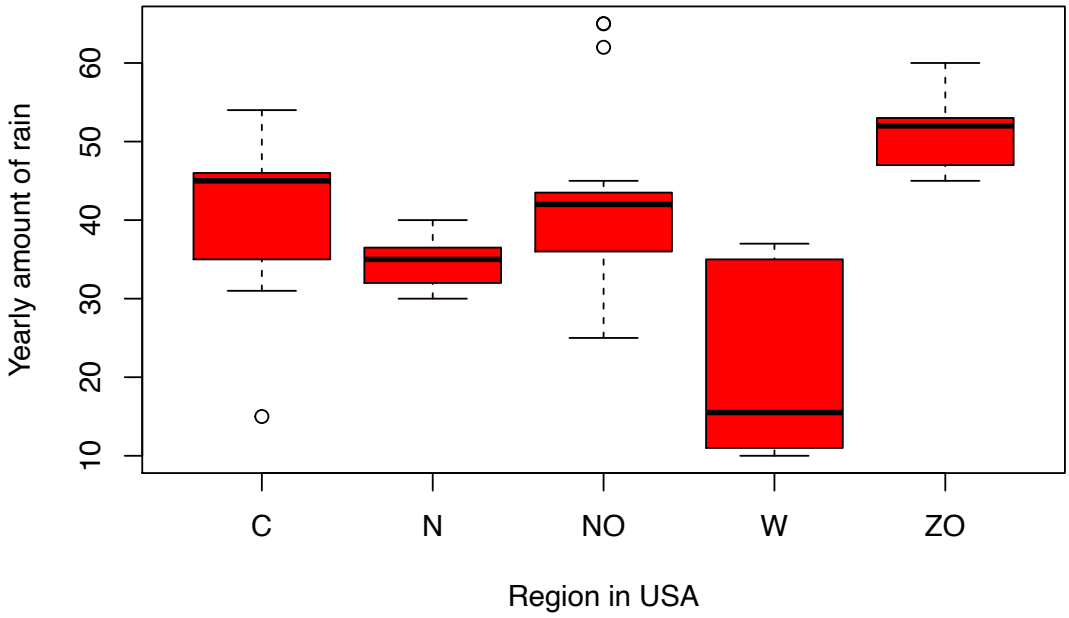
1. We sort our data from small to large values for the variable *Rain*.

```
subpol <- pollution[,c("Rain", "regio")]
library("doBy")
sortsubpol <- orderBy( ~ Rain, data = subpol)
head(sortsubpol)
```

```
##    Rain regio
## 47   10     W
## 29   11     W
## 49   13     W
## 18   15     C
## 48   18     W
## 34   25    NO
```

2. We associate ranks to the observations. The smallest observation gets rank 1, the largest one gets rank 59. We then compute the total number of ranks within each region.

Region	number of observations	Total number of ranks	Average rank
C	9	329	36.5
N	15	304	20.3
NO	24	818.5	34.1
W	6	57.5	9.6
ZO	5	261	52.2



3. The Kruskal-Wallis statistic is based on the ranks
   Compute the Kruskal-Wallis statistic in R:

```
kruskal.test(rain ~ region)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  rain by region
## Kruskal-Wallis chi-squared = 24.397, df = 4, p-value = 6.649e-05
```

Conclusion:
Since $p - value < 0.05$, we reject H_0. The location parameters are not the same in all the regions. In at least one region, there is a shift in location parameter.

4. Multiple comparisons
   When you install the R package `pgirmess`, you have the possibility to ask for multiple comparisons:

```
library(pgirmess)
kruskalmc(rain, region)
```

```
## Multiple comparison test after Kruskal-Wallis
## p.value: 0.05
## Comparisons
##         obs.dif critical.dif difference
## C-N   16.288889     20.32813      FALSE
## C-NO   2.451389     18.84468      FALSE
## C-W   26.972222     25.41016       TRUE
## C-ZO  15.644444     26.89159      FALSE
```

```
## N-NO  13.837500     15.86864      FALSE
## N-W   10.683333     23.28880      FALSE
## N-ZO  31.933333     24.89677       TRUE
## NO-W  24.520833     22.00584       TRUE
## NO-ZO 18.095833     23.70102      FALSE
## W-ZO  42.616667     29.19405       TRUE
```

7 Two-way ANOVA

7.1 Introduction

Consider an experiment involving two fixed-effect factors.

Example *Diet*

24 men, each weighing about 20 kg too much, are accidentally spread over 12 treatments (3 levels of jogging, 4 levels of diet). There are two men for each treatment. Each man used the same number of calories per day, the diet differs only in the amounts of protein, fat and carbohydrates. At the end of the experiment, the men are weighted again and their losses are calculated. At the end of the experiment, the men are weighted again and their losses are calculated.

Weight loss		**Diet**			
		Normal	Protein	Fat	Carbohydrates
Jogging	0 km	8.5 11.5	15.5 16.5	8.5 7.5	15.5 13.5
	1 km	14 16	20 23	13 11	21 18
	2 km	24.5 19.5	27 24	22 27	24.5 27.5

Import the data set *diet.txt* as `diet_df` in R.

```
head(diet_df)
```

```
##   LOSS JOGGING    DIET
## 1  8.5     0km  normal
## 2 11.5     0km  normal
## 3 15.5     0km protein
## 4 16.5     0km protein
## 5  8.5     0km     fat
## 6  7.5     0km     fat
```

Descriptive statistics

We can ask for some description statistics for the treatment:

```
names(diet_df)
```

```
## [1] "LOSS"    "JOGGING" "DIET"
```

```
by_diet_jogging <- group_by(diet_df, DIET, JOGGING)
summarise(by_diet_jogging, Avgloss = mean(LOSS),SDloss=sd(LOSS),number=n())
```

```
## # A tibble: 12 x 5
## # Groups:   DIET [4]
##    DIET    JOGGING Avgloss SDloss number
##    <fct>   <fct>     <dbl>  <dbl>  <int>
##  1 carbo   0km        14.5  1.41       2
##  2 carbo   1km        19.5  2.12       2
##  3 carbo   2km        26    2.12       2
##  4 fat     0km         8    0.707      2
##  5 fat     1km        12    1.41       2
##  6 fat     2km        24.5  3.54       2
##  7 normal  0km        10    2.12       2
##  8 normal  1km        15    1.41       2
##  9 normal  2km        22    3.54       2
## 10 protein 0km        16    0.707      2
## 11 protein 1km        21.5  2.12       2
## 12 protein 2km        25.5  2.12       2
```

If we want to see the descriptive statistics for jogging and diet separately:

- descriptive statistics by diet

```
by_diet <- group_by(diet_df, DIET)
summarise(by_diet, Avgloss = mean(LOSS),SDloss=sd(LOSS),number=n())
```

```
## # A tibble: 4 x 4
##   DIET    Avgloss SDloss number
##   <fct>     <dbl>  <dbl>  <int>
## 1 carbo      20     5.37      6
## 2 fat        14.8   7.89      6
## 3 normal     15.7   5.73      6
## 4 protein    21     4.48      6
```

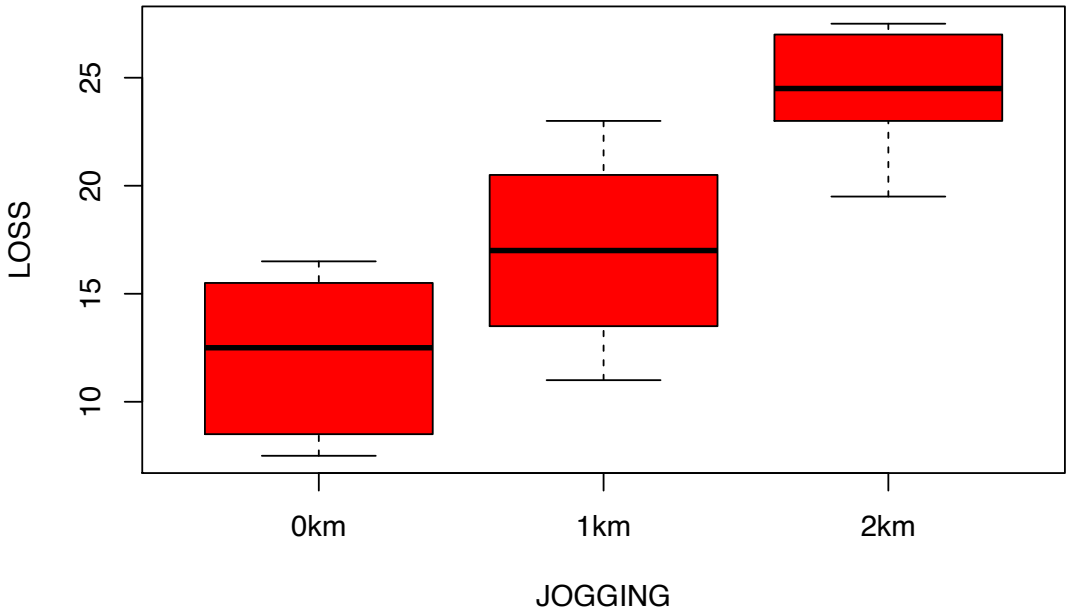
- descriptive statistics by jogging

```
by_jogging <- group_by(diet_df,  JOGGING)
summarise(by_jogging, Avgloss = mean(LOSS),SDloss=sd(LOSS),number=n())
```

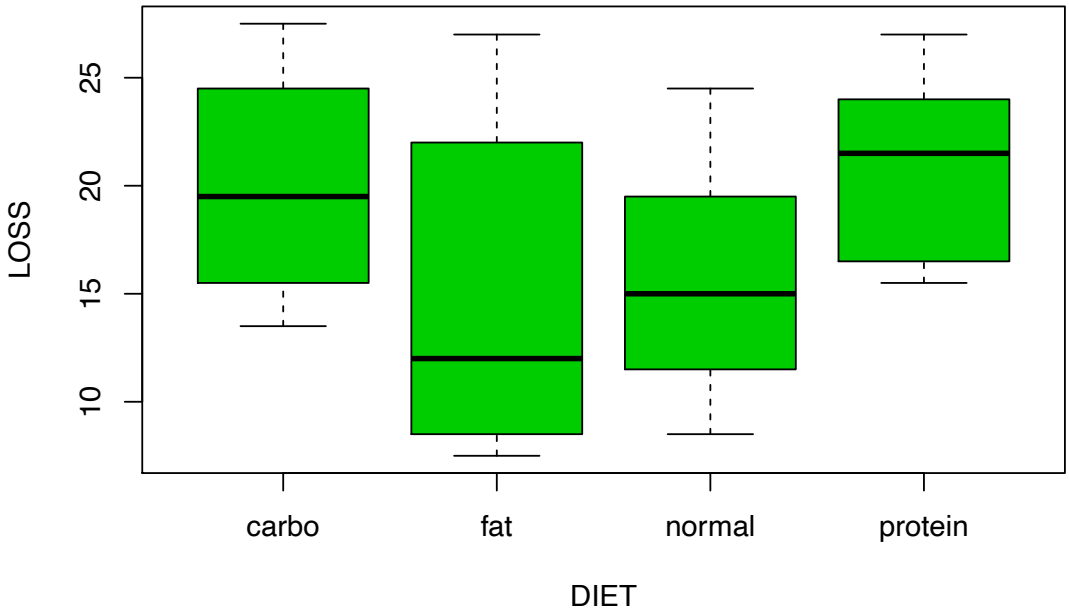
```
## # A tibble: 3 x 4
##   JOGGING Avgloss SDloss number
##   <fct>     <dbl>  <dbl>  <int>
## 1 0km        12.1   3.62      8
## 2 1km        17     4.21      8
## 3 2km        24.5   2.75      8
```

Some univariate box plots

```
boxplot(LOSS ~ JOGGING, col=2, data=diet_df)
```




```
boxplot(LOSS ~ DIET, col=3, data=diet_df)
```

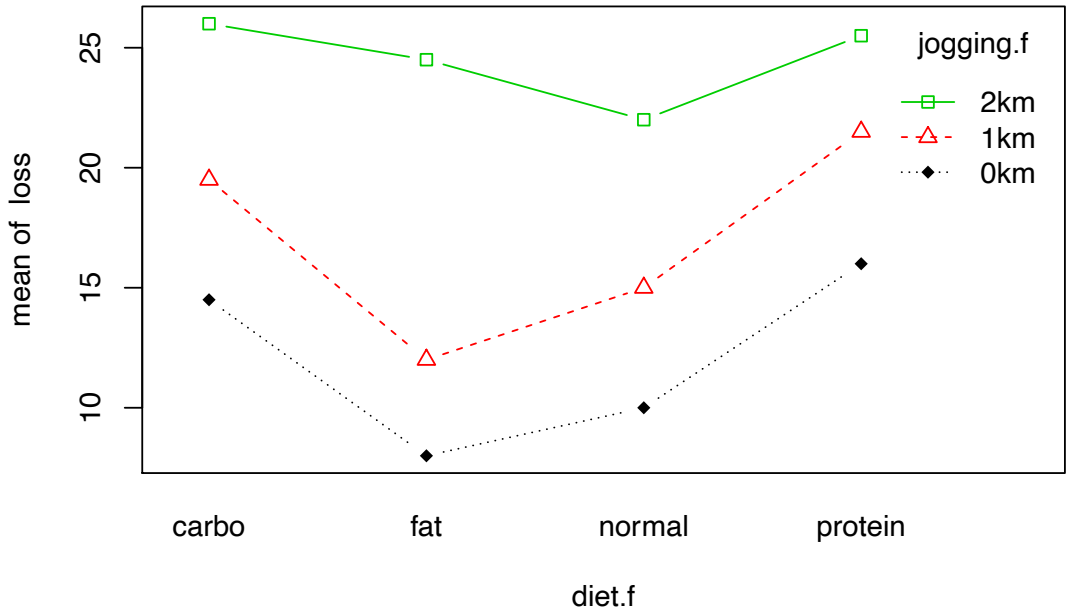


When we only consider one factor at a time, we miss the 'joint' effect. Such a joint effect is called **interaction**. We can also ask for one interaction plot with the function `interaction.plot` from the package `stats`. This function plots the mean (or other summary) of the response for two-way combinations of factors, thereby illustrating possible interactions.

```
?interaction.plot
```

Visualization of the mean values for every combination of the factors `diet` and `jogging`

```
diet.f <- as.factor(diet_df$DIET)
jogging.f <- as.factor(diet_df$JOGGING)
loss <- diet_df$LOSS
interaction.plot(diet.f, jogging.f, loss, type = "b", pch = c(18, 24, 22), col = c(1, 2, 3))
```



7.2 Two-way ANOVA model

Regression model	
Model without interaction:	$Y = \alpha + \beta_1 x_1 + \beta_2 x_2$
Model with interaction:	$Y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \beta_{12}\,(x_1 \cdot x_2)$

ANOVA model	
Main effects model:	$Y_{ij} = \mu_{\bullet\bullet} + \alpha_i + \beta_j$
Model with interaction:	$Y_{ij} = \mu_{\bullet\bullet} + \alpha_i + \beta_j + (\alpha\beta)_{ij}$

In the two-way ANOVA model with interaction:
$Y_{ijk} = \mu_{\bullet\bullet} + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \varepsilon_{ijk}$
with $i = 1, 2, ..., I; j = 1, 2, ..., J$ and $k = 1, 2, ..., K$

Example *Diet*

The weight loss for an individual can be written as the sum of

- The overall mean loss ($\mu_{\bullet\bullet}$)
- A part depending on the jogging level (α_i)
- A part depending on the diet type (β_i)
- A part depending on the interaction between jogging and diet type ($(\alpha\beta)_{ij}$)
- An error term ε_{ijk} which we assume to be independent and normally distributed ($\varepsilon_{ijk} \sim N(0, \sigma^2)$)

7.3 Strategy for the analysis of two-way ANOVA studies

Step 1: Test whether the interaction is significant
$\xrightarrow[\text{Yes}]{}$ If yes, go to step 2A
$\xrightarrow[\text{No}]{}$ If no, go to step 2B

Step 2A: The interaction term is significant

- Check the diagnostics.
- Use pairwise comparisons on interaction effect.

Step 2B: Only main effects are important. Drop the interaction term and refit the model.

- Check the diagnostics.
- Use pairwise comparisons on the main effects.

7.3.1 Example in R

Example *Diet* in R

We always use **sum coding** when working with two-way ANOVA.

```
diet.aov1 <- aov(LOSS ~ JOGGING + DIET + JOGGING*DIET, data = diet_df,
                 contrasts = list(JOGGING = "contr.sum", DIET = "contr.sum"))
summary(diet.aov1)
```

```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## JOGGING       2  621.7  310.87  68.450 2.74e-07 ***
## DIET          3  170.5   56.82  12.511 0.000528 ***
## JOGGING:DIET  6   43.9    7.32   1.612 0.226638
## Residuals    12   54.5    4.54
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We have an F-test for each effect in our model.

Step 1: Check whether the interaction term is significant.
The interaction term is not significant ($p - value = 0.23$).
This means that the curves in the `interaction.plot` are parallel.

Step 2B: Hence we can drop the interaction term from our model and rerun the model.
Drop interaction term and refit model:

```
diet.aov2 <- aov(LOSS ~ JOGGING + DIET, data = diet_df,
                 contrasts = list(JOGGING = "contr.sum", DIET = "contr.sum"))
summary(diet.aov2)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## JOGGING      2  621.7  310.87   56.86 1.66e-08 ***
## DIET         3  170.5   56.82   10.39  0.00034 ***
## Residuals   18   98.4    5.47
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We now have an additive model with 2 significant main factors: $JOGGING$ and $DIET$.

Remark:
Since ANOVA can be seen as a linear model, we can also use the `lm` function in R.

Here we give the results for the full model (comparable with the previous results).

```
diet.lm <- lm(LOSS ~ JOGGING + DIET + JOGGING * DIET, data = diet_df,
              contrasts = list(JOGGING ="contr.sum", DIET = "contr.sum"))
# To see the ANOVA table
Anova(diet.lm, type = "III") # function from package 'car'
```

```
## Anova Table (Type III tests)
##
## Response: LOSS
##              Sum Sq Df   F value    Pr(>F)
## (Intercept)  7668.4  1 1688.4495 2.795e-14 ***
## JOGGING       621.7  2   68.4495 2.740e-07 ***
## DIET          170.5  3   12.5107 0.0005277 ***
## JOGGING:DIET   43.9  6    1.6116 0.2266382
## Residuals      54.5 12
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

7.4 Diagnostics

7.4.1 Checking homogeneity of variances

When the interaction term is significant, then we have to check
$H_0 : \sigma^2_{11} = \sigma^2_{12} = ...\sigma^2_{IJ}$

Since the interaction term is not significant (in the *Diet* example), it suffices to check homogeneity of variances for $JOGGING$ and $DIET$ separately.

- **For** $JOGGING$**:** $H_0 : \sigma^2_{0km} = \sigma^2_{1km} = \sigma^2_{2km}$
  Test homogeneity of variances for $JOGGING$:

```
leveneTest(LOSS ~ JOGGING, data = diet_df) # This is a function from the package 'car'
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  2  1.6545 0.2151
##       21
```

- **For** $DIET$**:** $H_0 : \sigma^2_{normal} = \sigma^2_{protein} = \sigma^2_{fat} = \sigma^2_{carbo}$
  Test homogeneity of variances for $DIET$:

```
leveneTest(LOSS ~ DIET, data = diet_df)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  3  0.3822 0.7669
##       20
```

Remark:

1. In case Levene's test is rejected, we can also use the rule of thumb:
   $\frac{\max(\sigma^2_1,\sigma^2_2,...,\sigma^2_r)}{\min(\sigma^2_1,\sigma^2_2,...,\sigma^2_r)} \leq 5$. In case the rule of thumb is satisfied, an ANOVA F-test can be used.

```
diet_df %>% group_by(JOGGING, DIET) %>%
  summarise(mean = mean(LOSS, na.rm = T), var = var(LOSS, na.rm = T),
            n = n()) %>% select(DIET, JOGGING, mean, var, n)
```

```
## # A tibble: 12 x 5
## # Groups:   JOGGING [3]
##    DIET    JOGGING  mean   var     n
```

```
##    <fct>   <fct>   <dbl> <dbl> <int>
##  1 carbo   0km      14.5   2       2
##  2 fat     0km       8     0.5     2
##  3 normal  0km      10     4.5     2
##  4 protein 0km      16     0.5     2
##  5 carbo   1km      19.5   4.5     2
##  6 fat     1km      12     2       2
##  7 normal  1km      15     2       2
##  8 protein 1km      21.5   4.5     2
##  9 carbo   2km      26     4.5     2
## 10 fat     2km      24.5  12.5     2
## 11 normal  2km      22    12.5     2
## 12 protein 2km      25.5   4.5     2
```

2. If there is an indication of unequal variances (based on the above tests), you can use heteroscedastic consistent covariance matrices.
   (Here, this is not the case. We just illustrate here how to work.)

```
# Robust analysis
Anova(diet.aov2, type = "III", white.adjust = "hc3")
```

```
## Analysis of Deviance Table (Type III tests)
##
## Response: LOSS
##             Df         F    Pr(>F)
## (Intercept)  1 1051.8855 < 2.2e-16 ***
## JOGGING      2   42.2050 1.601e-07 ***
## DIET         3    7.9432  0.001395 **
## Residuals   18
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

This robust analysis also indicates that $JOGGING$ and $DIET$ are significant.

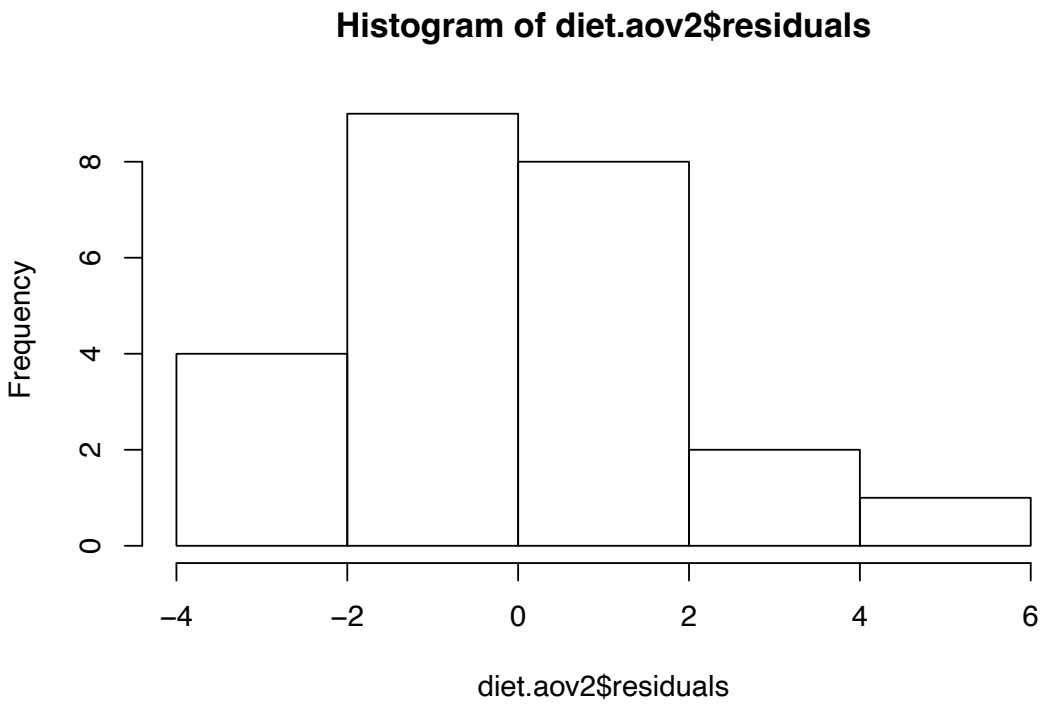
7.4.2 Checking normality of residuals

Test normality of the within-cell residuals

```
shapiro.test(diet.aov2$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  diet.aov2$residuals
## W = 0.97129, p-value = 0.699
```

```
hist(diet.aov2$residuals)
```

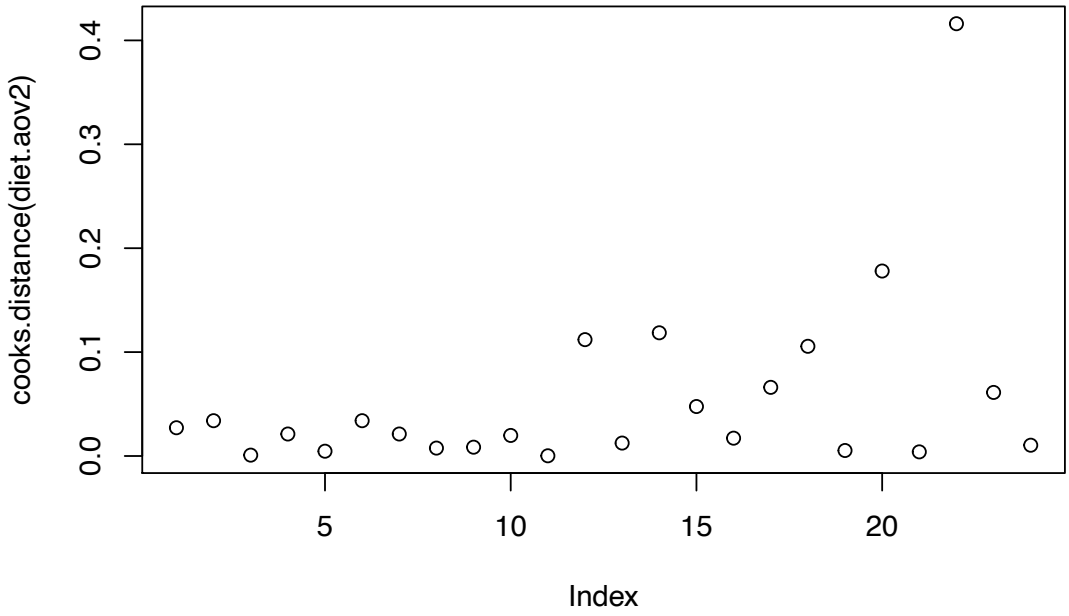


The residuals seem to be normally distributed

7.4.3 Influential observations

We check the presence of influential observations by using Cook's distance.

```
plot(cooks.distance(diet.aov2))
```




```
diet_df[cooks.distance(diet.aov2)>0.3,]
```

```
##    LOSS JOGGING DIET
## 22   27     2km  fat
```

Observation number 22 has a large Cook's distance compared to the rest. We'll repeat the analysis for the data while deleting that point.

```
diet_df_small <- diet_df[-22,] # Delete observation 22

loss_small <- diet_df_small$LOSS
jogging_small <- diet_df_small$JOGGING
diet_small <- diet_df_small$DIET
```

Two-way ANOVA

```
diet.aov1_small <- aov(loss_small ~ jogging_small + diet_small + jogging_small * diet_small,
                       contrasts = list(jogging_small = "contr.sum", diet_small = "contr.sum"))
summary(diet.aov1_small)
```

```
##                          Df Sum Sq Mean Sq F value   Pr(>F)
## jogging_small             2  542.0  271.00  70.977 5.16e-07 ***
## diet_small                3  204.3   68.09  17.832 0.000156 ***
## jogging_small:diet_small  6   15.5    2.58   0.675 0.672896
## Residuals                11   42.0    3.82
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Two-way ANOVA without interaction

```
diet.aov2_small <- aov(loss_small ~ jogging_small + diet_small,
                       contrasts = list(jogging_small = "contr.sum", diet_small = "contr.sum"))
summary(diet.aov2_small)
```

```
##                Df Sum Sq Mean Sq F value   Pr(>F)
## jogging_small   2  542.0  271.00   80.17 2.21e-09 ***
## diet_small      3  204.3   68.09   20.14 7.79e-06 ***
## Residuals      17   57.5    3.38
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We obtain similar results as compared to before removing observation 22. We keep observation 22 in the data and continue the analysis.

7.5 Multiple comparisons for the main effects (in case interaction is not significant)

In our example, the interaction effect is not significant. The two main effects *JOGGING* and *DIET* are significant. We are going to use multiple comparisons techniques (Tukey, Scheffé, Bonferroni and Holm) on the main effects.

Multiple comparisons on the main effects:

```
library(agricolae)
library(sp)
diet.aov2 <- aov(LOSS ~ JOGGING + DIET, data = diet_df,
                 contrasts = list(JOGGING = "contr.sum", DIET = "contr.sum"))
```

- Tukey HSD test for *JOGGING*

```
out <-  HSD.test(diet.aov2, "JOGGING", group = FALSE)
out$means
```

```
##       LOSS      std r  Min  Max   Q25  Q50   Q75
## 0km 12.125 3.622844 8  7.5 16.5  8.50 12.5 15.50
## 1km 17.000 4.208834 8 11.0 23.0 13.75 17.0 20.25
## 2km 24.500 2.751623 8 19.5 27.5 23.50 24.5 27.00
```

```
out$comparison
```

```
##           difference pvalue signif.        LCL       UCL
## 0km - 1km     -4.875 0.0016      **  -7.858847 -1.891153
## 0km - 2km    -12.375 0.0000     *** -15.358847 -9.391153
## 1km - 2km     -7.500 0.0000     *** -10.483847 -4.516153
```

- Tukey HSD test for *DIET*

```
out1 <- HSD.test(diet.aov2, "DIET", group = FALSE)
out1$means
```

```
##             LOSS      std r  Min  Max    Q25  Q50    Q75
## carbo   20.00000 5.366563 6 13.5 27.5 16.125 19.5 23.625
## fat     14.83333 7.890923 6  7.5 27.0  9.125 12.0 19.750
## normal  15.66667 5.732946 6  8.5 24.5 12.125 15.0 18.625
## protein 21.00000 4.483302 6 15.5 27.0 17.375 21.5 23.750
```

```
out1$comparison
```

```
##                 difference pvalue signif.        LCL       UCL
```

```
## carbo - fat       5.1666667 0.0062      **  1.3511428  8.982190
## carbo - normal    4.3333333 0.0229       *  0.5178095  8.148857
## carbo - protein  -1.0000000 0.8794         -4.8155238  2.815524
## fat - normal     -0.8333333 0.9252         -4.6488572  2.982190
## fat - protein    -6.1666667 0.0012      ** -9.9821905 -2.351143
## normal - protein -5.3333333 0.0047      ** -9.1488572 -1.517810
```

Remark:

- The majority of the R manuals suggest the Tukey HSD to ask for multiple comparisons test. But when writing a report, try to be systematic by using the same method for multiple comparisons everywhere.
- If we have a model with a significant interaction, before using the `HSD.test()` function, you first need to create a new variable that is equal to interaction. The, apply an additive ANOVA model on three variables and after that apply the Tukey HSD test (or another test). (This will be discussed later).

7.6 Two-way ANOVA when cells have unequal sample size

7.6.1 What is an unbalanced design?

If you have a two-way ANOVA with **unequal numbers of entries per cell**, then the main effects and the interaction effect are no longer independent of each other. This type of design where the sample sizes for the different treatment combinations are not all equal is called an **"unbalanced design"**.

- With a *balanced design*, you have the following decomposition of the Total Sum of Squares:
  $SSTO = SSA + SSB + SSAB + SSE$
- In an *unbalanced design* this equation does not hold anymore.

Hence, the general recommendation for an unbalanced design is to use the regression approach:

- Use *Type III SS* to check the significance of the effects of the model.
- For the interpretation, use the least square averages instead of the sample averages.

7.6.2 Illustrative example

Example *Training*

The data *training.txt* contains information about children that were assigned to different training methods ($Method$) and that were separated for some period of time (Sep_Period). The results of the test is registered in *score*.

	Length of separation period		
Method	**20 minutes**	**40 minutes**	**60 minutes**
No Training	26 23 28 19 18	30 25 27 36	6 11 17 10 14 19
Training	15 24 25 16 22 21	24 29 23 26 27 21	31 29 35 38 34 30

Import the data set *training.txt* as `training` in R.

```
training <- read.table(file=file.choose(), header=TRUE)
```

```
head(training)
```

```
##        Method Sep_Period score
## 1 No_Training     20_min    26
## 2 No_Training     20_min    23
## 3 No_Training     20_min    28
## 4 No_Training     20_min    19
## 5 No_Training     20_min    18
## 6 No_Training     40_min    30
```

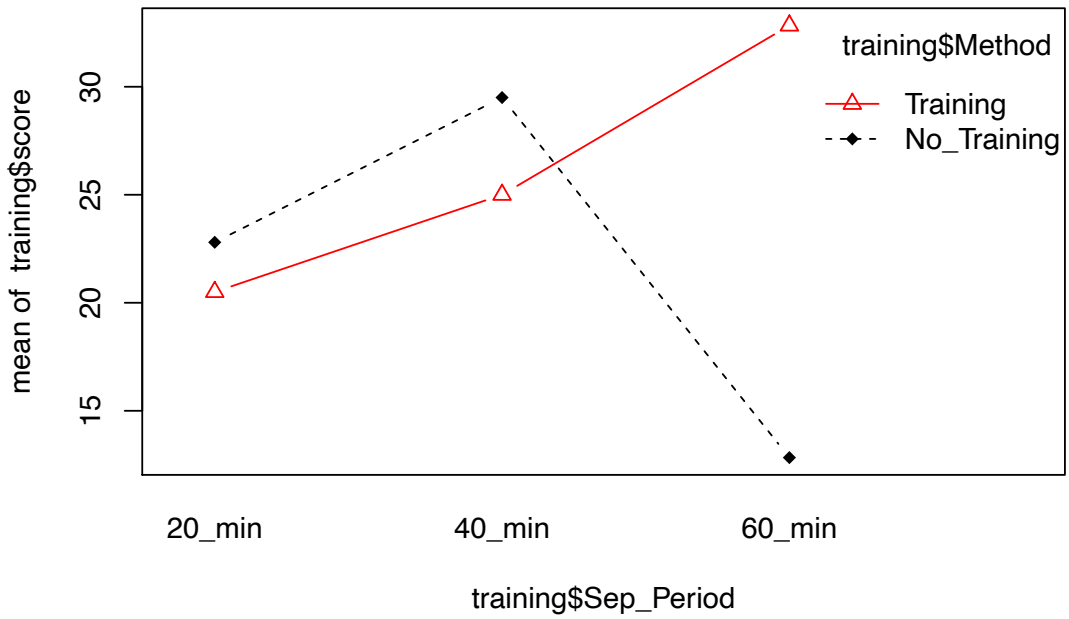
Descriptive statistics:

```r
by_Method_SepPeriod <- group_by(training, Method, Sep_Period)
summarise(by_Method_SepPeriod, Avg = mean(score), SD = sd(score), number = n())
```

```
## # A tibble: 6 x 5
## # Groups:   Method [2]
##   Method      Sep_Period   Avg    SD number
##   <fct>       <fct>      <dbl> <dbl>  <int>
## 1 No_Training 20_min      22.8  4.32      5
## 2 No_Training 40_min      29.5  4.80      4
## 3 No_Training 60_min      12.8  4.79      6
## 4 Training    20_min      20.5  4.14      6
## 5 Training    40_min      25    2.90      6
## 6 Training    60_min      32.8  3.43      6
```

Visualization:

```r
library(stats)
interaction.plot(training$Sep_Period, training$Method, training$score, type="b",
                 pch = c(18, 24, 22), col = c(1, 2, 3))
```



7.6.3 ANOVA table

For an unbalanced design, we use a regression approach. To obtain a *Type III SS ANOVA*, we have to do the following:
We are going to apply function `Anova()` from the `library(car)` in combination with function `lm()` for the linear models.

- Set for `lm()` the contrast from `contr.treatment` to `contr.sum`.

- Specify for `Anova()` the value of `type = "III"`.

```
# ANOVA table in case of unbalanced design: use of lm function
training.lm <- lm(score ~ Method + Sep_Period + Method*Sep_Period, data = training,
                  contrasts = list(Method = "contr.sum", Sep_Period = "contr.sum"))
Anova(training.lm, type = "III")
```

```
## Anova Table (Type III tests)
##
## Response: score
##                     Sum Sq Df   F value    Pr(>F)
## (Intercept)       18432.3  1 1118.4453 < 2.2e-16 ***
## Method              156.0  1    9.4681  0.004753 **
## Sep_Period          175.8  2    5.3333  0.011168 *
## Method:Sep_Period  1036.3  2   31.4404  8.89e-08 ***
## Residuals           445.0 27
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Step 1: Check whether the interaction is significant
We can see that the interaction is significant ($p - value < 0.05$). It means that we are not allowed to interpret the main effects.

Step 2A:

- Check the diagnostics.
- Use pairwise comparisons on interaction effect.

Both will be treated in the following two sections.

7.6.4 Diagnostics

1. Check assumption of *normality*
   Test normality of the within-cell residuals

```
shapiro.test(training.lm$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  training.lm$residuals
## W = 0.96223, p-value = 0.2985
```

```
hist(training.lm$residuals)
```

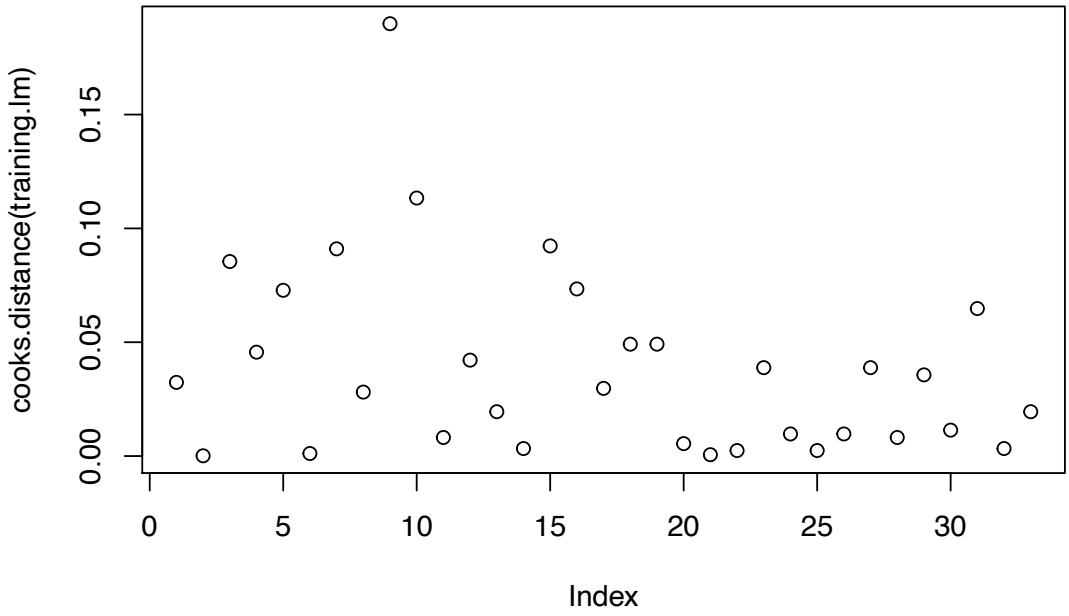
2. Check assumption of *homogeneity of variances*

```
leveneTest(score ~ Method*Sep_Period, data = training)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  5  0.3678 0.8661
##       27
```

3. Check *influential observations*
   Plotting Cook's distance

```
plot(cooks.distance(training.lm))
```



7.6.5 Pairwise comparisons of treatment effects

Since the interaction term is significant, we are interested in the pairwise comparisons of the interaction effect ($Method * Sep\_Period$).

Method:
Create one new variable which is the interaction term. Refit a one-way ANOVA with as single variable this interaction term. Then you can use Tukey and other MC (multiple comparison) methods on this new variable.

```
## Method 1: Create a new variable with the interaction term
# Initialize the new variable
method_sep_period <- character(length(training$score))
# Store values in new variable
for (i in 1: length(training$score))
{method_sep_period[i] <- paste(substring(training$Method[i], 1, 4),
                               training$Sep_Period[i], sep="")}
# Create a new data frame
new_df <- data.frame(score = training$score, method_sep_period)
head(new_df, n = 10)
```

```
##    score method_sep_period
## 1     26        No_T20_min
## 2     23        No_T20_min
## 3     28        No_T20_min
## 4     19        No_T20_min
## 5     18        No_T20_min
## 6     30        No_T40_min
```

```
## 7     25        No_T40_min
## 8     27        No_T40_min
## 9     36        No_T40_min
## 10     6        No_T60_min
```

Descriptive statistics

```
describe <- describeBy(new_df$score, new_df$method_sep_period, mat = TRUE)
describe.st <- subset(describe, select=c("group1", "n", "mean", "sd", "median", "min", "max"))
describe.st
```

```
##         group1 n     mean       sd median min max
## X11 No_T20_min 5 22.80000 4.324350   23.0  18  28
## X12 No_T40_min 4 29.50000 4.795832   28.5  25  36
## X13 No_T60_min 6 12.83333 4.792355   12.5   6  19
## X14 Trai20_min 6 20.50000 4.135215   21.5  15  25
## X15 Trai40_min 6 25.00000 2.898275   25.0  21  29
## X16 Trai60_min 6 32.83333 3.430258   32.5  29  38
```

```
# Apply one-way ANOVA on this new data frame
new_df.aov <- aov(score ~ method_sep_period, new_df,
                  contrasts = list(method_sep_period = "contr.sum"))
summary(new_df.aov)
```

```
##                   Df Sum Sq Mean Sq F value   Pr(>F)
## method_sep_period  5   1419  283.78   17.22 1.17e-07 ***
## Residuals         27    445   16.48
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Ask for Tukey HSD test
out2 <- HSD.test(new_df.aov, "method_sep_period", group = FALSE)
round(out2$means, 2)
```

```
##            score  std r Min Max   Q25  Q50   Q75
## No_T20_min 22.80 4.32 5  18  28 19.00 23.0 26.00
## No_T40_min 29.50 4.80 4  25  36 26.50 28.5 31.50
## No_T60_min 12.83 4.79 6   6  19 10.25 12.5 16.25
## Trai20_min 20.50 4.14 6  15  25 17.25 21.5 23.50
## Trai40_min 25.00 2.90 6  21  29 23.25 25.0 26.75
## Trai60_min 32.83 3.43 6  29  38 30.25 32.5 34.75
```

```
out2$comparison
```

```
##                           difference pvalue signif.         LCL         UCL
## No_T20_min - No_T40_min   -6.700000 0.1718         -15.0436290   1.6436290
## No_T20_min - No_T60_min    9.966667 0.0046      **   2.4351153  17.4982181
## No_T20_min - Trai20_min    2.300000 0.9336          -5.2315514   9.8315514
## No_T20_min - Trai40_min   -2.200000 0.9444          -9.7315514   5.3315514
## No_T20_min - Trai60_min  -10.033333 0.0043      ** -17.5648847  -2.5017819
## No_T40_min - No_T60_min   16.666667 0.0000     ***   8.6380059  24.6953274
## No_T40_min - Trai20_min    9.000000 0.0213       *   0.9713392  17.0286608
## No_T40_min - Trai40_min    4.500000 0.5328          -3.5286608  12.5286608
## No_T40_min - Trai60_min   -3.333333 0.7972         -11.3619941   4.6953274
## No_T60_min - Trai20_min   -7.666667 0.0313       * -14.8477191  -0.4856142
## No_T60_min - Trai40_min  -12.166667 0.0002     *** -19.3477191  -4.9856142
## No_T60_min - Trai60_min  -20.000000 0.0000     *** -27.1810525 -12.8189475
## Trai20_min - Trai40_min   -4.500000 0.4123         -11.6810525   2.6810525
```

```
## Trai20_min - Trai60_min -12.333333 0.0002     *** -19.5143858  -5.1522809
## Trai40_min - Trai60_min  -7.833333 0.0265       * -15.0143858  -0.6522809
```

8 Experimental design

8.1 Observational study versus designed experiment

- In an **experiment** investigators apply treatments to experimental units (people, animals, plots of land, etc.) and then proceed to observe the effect of the treatments on the experimental units.
- In an **observational study**, investigators observe subjects and measure variables of interest without assigning treatments to the subjects. The treatment that each subject receives is determined beyond the control of the investigator.

Example *Smoking*
Suppose we want to study the effect of smoking on the lung capacity in women.

Experiment

- Find 100 women age 30 who do not currently smoke.
- Randomly assign 50 of the 100 women to the smoking treatment and the other 50 to the no-smoking treatment.
- Those in the smoking group smoke a pack a day for 10 years while those in the control group remain smoke free for 10 years.
- Measure lung capacity for each of the 100 women.
- Analyze, interpret and draw conclusions from data.

Observational study

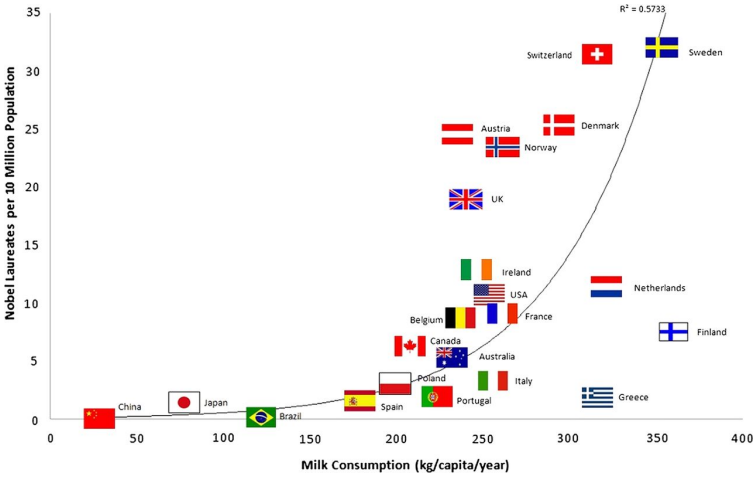
- Find 100 women age 40 of which 50 have been smoking a pack a day for 10 years while the other 50 have been smoke free for 10 years.
- Measure lung capacity for each of the 100 women.
- Analyze, interpret and draw conclusions from data.

Fisher's Hypothesis

- Suppose there is a gene that causes smoking to appear to be a very pleasurable experience.
- Suppose the same gene also causes emphysema, lung cancer, throat cancer, etc.
- People who have the gene will be more likely to smoke than people who do not have the gene.
- People who have the gene will be more likely to get emphysema, lung cancer, throat cancer, etc
- So is it really smoking that causes health problems? Maybe it is just the gene?
- A **confounding variable** is related both to group membership and to the outcome of interest. Its presence makes it hard to establish the outcome as being a direct consequence of group membership.

Correlation does not imply causation!

Correlation between countries' annual milk consumption (kg/capital/year) and the number of Nobel laureates per 10 million population.



8.2 Basic principles of experimental design

8.2.1 Replication

In all experiments, some variation is introduced because of the fact that the experimental units such as individuals or plots of land in agricultural experiments cannot be physically identical. This type of variation

can be removed by using a number of experimental units. We therefore perform the experiment more than once, i.e., we repeat the basic experiment.

Replication allows us to estimate the experimental error and to perform statistical analysis.

8.2.2 Randomization

Randomization is a random process of assigning treatments to the experimental units.

The purpose of randomization is to remove bias and other sources of uncontrollable variation.

Another advantage of randomization (accompanied by replication) is that it forms the basis of any valid statistical test. Hence the treatments must be assigned at random to the experimental units.

Example *Corn yield*

You want to compare the yield for two types of corn (type A and type B).
We have several small fields which are available, but the fertility at one side of the land is different from the fertility at the other side.

First suggestion of assigning the different types to the 10 subfields:

A	A	B	B	B
A	A	A	B	B

Problem here: If we detect a difference in the yield, we cannot detect whether it comes from the different type of corn or whether it comes from the difference in fertility of the ground.

Remark:systematic arranging the type of corn over the several plots $\neq$ randomization

A	B	A	B	A
B	A	B	A	B

8.2.3 Blocking

It has been observed that all sources of uncontrollable variation are not removed by randomization and replication. A block is a subset of experimental conditions that are expected to be more homogeneous than the rest.

Blocking refers to the method of creating homogeneous blocks of data in which the nuisance factor is kept constant and the factor of interest is allowed to vary.

Blocking is used to eliminate the variability due to the difference between block.

Example *Corn yield*
Example of blocking

Field 1	Field 2	Field 3	Field 4	Field5
A	B	A	A	B
B	A	B	B	A

Within each block, the types are randomly assigned.

Remark:
Both blocking and randomization deal with nuisance [1] factors.

- Blocking can only be used when the nuisance factor is under our control (e.g. choice of materials or substances).
- If the nuisance factor is not under our control, then randomization remains the only tool available.

  'Block what you can, randomize what you cannot!'

9 The general linear model

In previous chapters we have seen

- Linear regression (simple and multiple)
- ANOVA (one-way and two-way)

The above models are special cases of the **General Linear Model** (GLM).

Models with continuous response variable

Explanatory variables	Response variable	Method
Continuous	Continuous	Regression
Categorical	Continuous	ANOVA
Continuous and categorical	Continuous	GLM

[1] A nuisance factor is a factor that has some effect on the response, but is of no interest to the experimenter. However, the variability it transmits to the response needs to be minimized or explained. Hence, nuisance factors needs to be taken into account in an analysis.

Chapter 10: Logistic regression

Contents

1 Introduction

Example *Political party*
Consider the *political_party.xlsx* file. Import this excel file in R as `political_party`. In this data set, one of the variables is the variable *Republican* which indicates whether a person votes for the Republican Party or not. We have 283 respondents ($n = 283$).

Variable	Type	Description
`political_party`	nominal	1: Republican 2: Democrat 3: Independent
`Republican`	response	based on the variable `political_party` 0: Not Republican 1: Republican
`gender`	indicator	0: Female 1: Male
`pro_capital_punishment`	continuous	10 point scale, higher values indicating greater support for the position.
`pro_welfare_reform`	continuous	10 point scale, higher values indicating greater support for the position.
`pro_fed_support_ed` (Federal support of education)	continuous	10 point scale, higher values indicating greater support for the position.

We always want to estimate the probability of *response* $= 1$ **(here:** $P[Republican = 1]$**).** The category with value 1 is called the "target category". We will start with univariate logistic regression with as explanatory variable `pro_capital_punishment`.

2 Regression model with binary response variable

- Consider the regression model $Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$ with $Y_i \in \{0, 1\}$. Then, $E(Y) = \beta_0 + \beta_1 x$.
- Assuming that Y_i is a Bernoulli distributed random variable, the following table holds:

Y_i	Probability
1	$P(Y_i = 1) = p$
0	$P(Y_i = 0) = 1 - p$

We can show that $E(Y) = 1 \cdot p + 0 \cdot (1 - p) = p$

→ Combining these results gives: $E(Y) = \beta_0 + \beta_1 x = p$

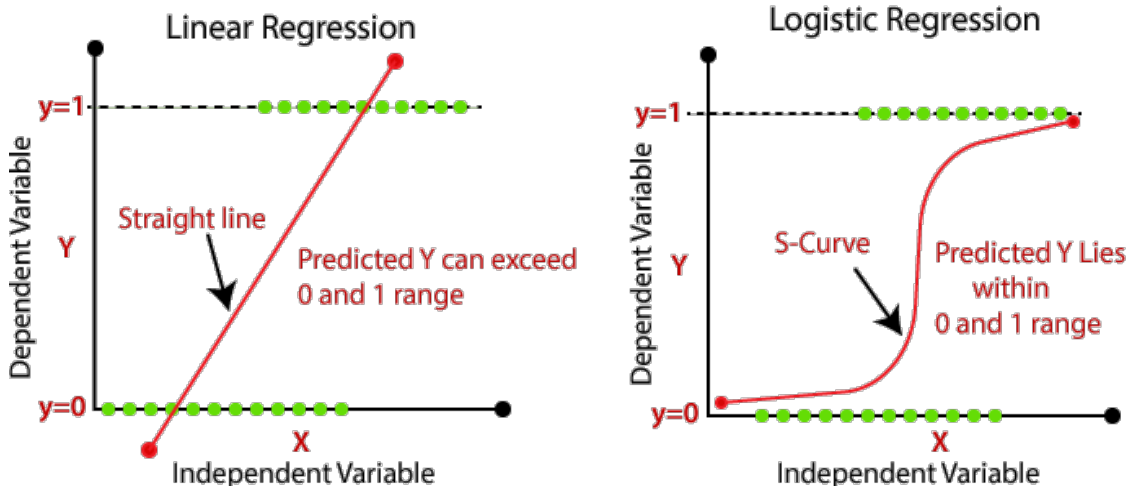
Interpretation:
The average response is the probability that $Y = 1$.

Problem:
Restriction on the response function:
$0 \leq E(Y) = p \leq 1$

⇒ a linear response function is not possible!! Linear regression is used to predict a continuous dependent

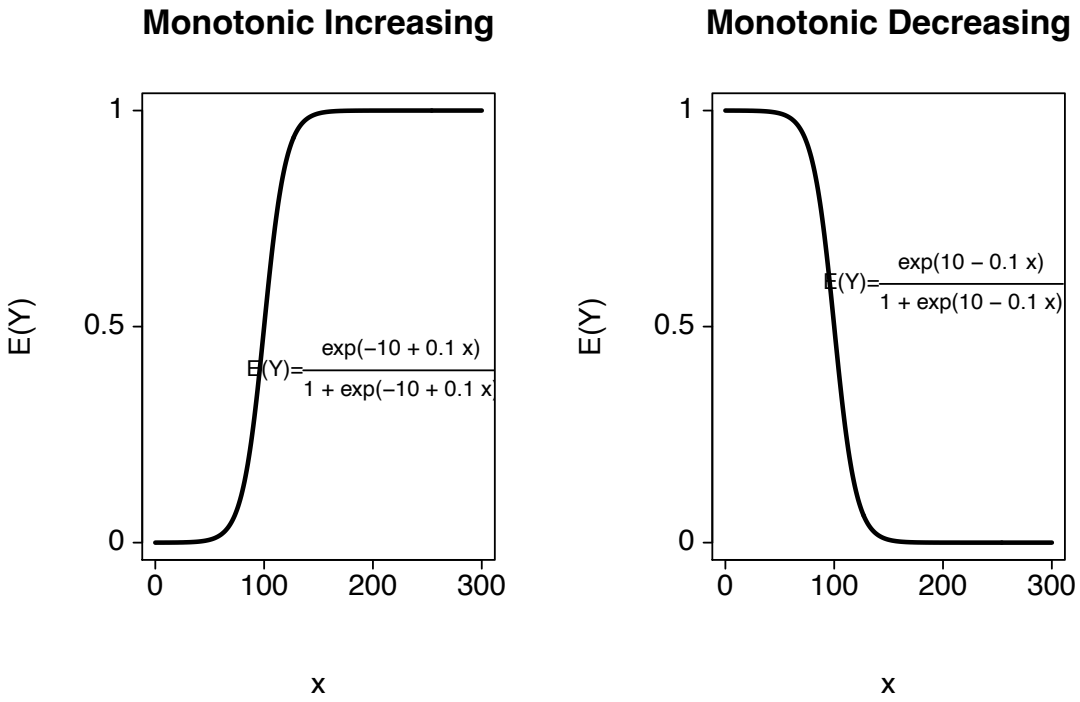
variable using a given set of independent variables.

Logistic Regression is used to predict a binary (0 or 1) dependent variable using a given set of independent variables.



3 Simple logistic regression

3.1 Logistic response function



The logistic response function has the form:
$p = E(Y) = \frac{\exp(\beta_0+\beta_1 x)}{1+\exp(\beta_0+\beta_1 x)}$
or
$p = E(Y) = \frac{1}{1+\exp(-\beta_0-\beta_1 x)}$

It can be seen that the relationship between the probability $p(= P[Y = 1])$ and the independent variable x is represented by a logistic curve. Note that this relationship is nonlinear.

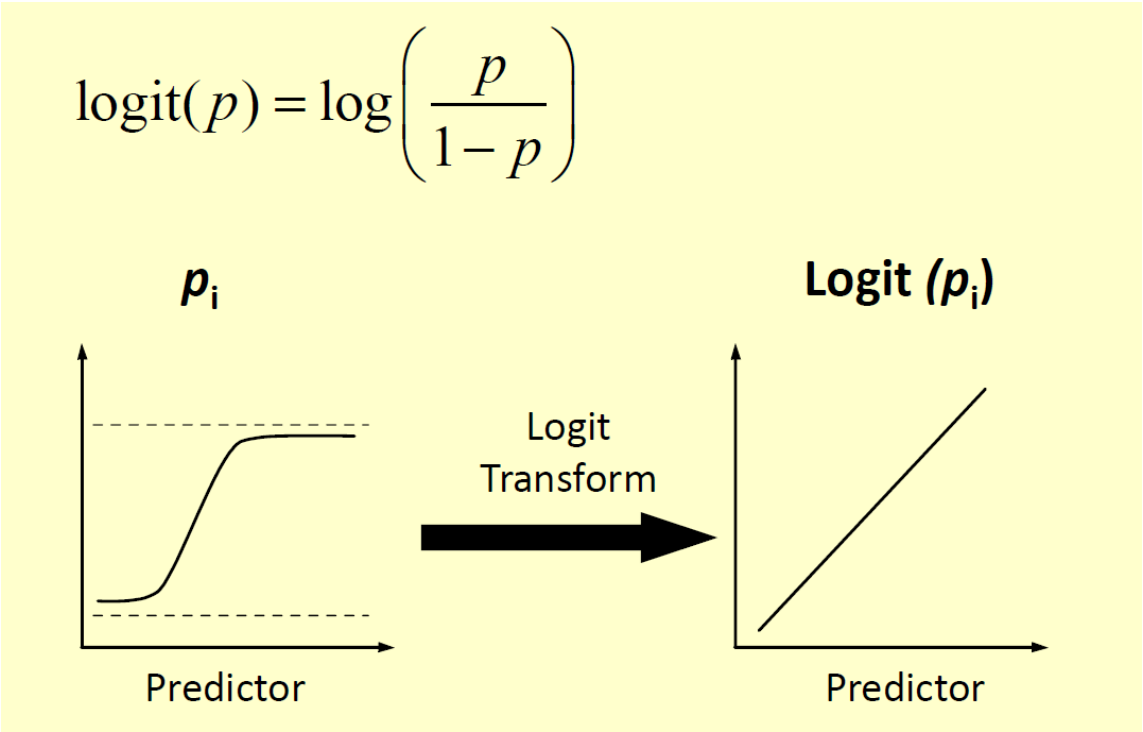
3.2 Properties of logistic response function

Some properties of the logistic response function:

- Either monotone increasing or monotone decreasing (depending on sign of β_1).
- Is almost linear in the range where $E(Y)$ ranges from 0.2 to 0.8.
- It approaches 0 and 1 at the two ends of the x range.
- **It can be linearized:** The logistic response function can be transformed to a linear one: Using the LOGIT transformation (i.e., $p' = \ln(\frac{p}{1-p})$), we obtain:
  $p' = \beta_0 + \beta_1 x$
  with $p = P(Y = 1)$, p' the logit mean response and $\frac{p}{1-p}$ the odds.

$\ln(\frac{p}{1-p}) = \beta_0 + \beta_1 x$
is called the Logit or Log(odds).

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right)$$



3.3 Interpretation of the odds

If the probability of an event is p, then the odds O of the event is
$O = \frac{p}{1-p} = \frac{probability\ of\ event}{probability\ of\ no\ event}$

- The odds for winning the lottery is the probability of winning the lottery divided by the probability of not winning the lottery.

- The odds of having a Facebook account is the probability of having a Facebook account divided by the probability of not having a Facebook account.

An odds of 4 means that the expected number of events is four times the number of no events.

Probability p	Odds O
0.1	0.11
0.2	0.25
0.3	0.43
0.4	0.67
0.5	1
0.6	1.5
0.7	2.33
0.8	4
0.9	9

$Odds < 1$ corresponds with $p < 0.5$.
Odds do have a lower bound of 0, but there is no upper bound.

Once you have *odds*, you can derive the probability of the event by
$p = \frac{odds}{1+odds}$

3.4 Assessing the model: the log-likelihood statistic

We state the simple logistic regression model as
$p = E(Y) = \frac{\exp(\beta_0+\beta_1 x)}{1+\exp(\beta_0+\beta_1 x)}$
with $p = P(Y = 1)$ and x the explanatory variable.

Remark:

- In regression analysis, *method of least squares* was used to obtain parameter estimates.
- In logistic regression, *maximum likelihood estimation* is used to obtain parameter estimates.

3.4.1 Log-likelihood function

Y_i are independent Bernoulli random variables with $P(Y_i = 1) = p_i$.

The probability function is: $f_i(Y_i) = p_i^{Y_i}(1-p_i)^{1-Y_i}$ where $Y_i \in \{0, 1\}$ (i.e., Y_i can only take the values 0 and 1).

Joint probability function: $g(Y_1, Y_2, ...Y_n) = \prod_{i=1}^{n} f_i(Y_i) = \prod_{i=1}^{n} p_i^{Y_i}(1-p_i)^{(1-Y_i)}$

Taking the natural logarithm:
$\log_e(g(Y_1, Y_2, ..., Y_n)) = \log_e\left(\prod_{i=1}^{n} p_i^{Y_i}(1-p_i)^{(1-Y_i)}\right)$

Or the **log-likelihood** can be written as:
$\log_e(g(Y_1, Y_2, ..., Y_n)) = \sum_{i=1}^{n}\left[Y_i \log_e\left(\frac{p_i}{1-p_i}\right)\right] + \sum_{i=1}^{n} \log_e(1-p_i)$

Remark:
$p_i = \frac{\exp(\beta_0+\beta_1 x_i)}{1+\exp(\beta_0+\beta_1 x_i)}$
and
$1 - p_i = \frac{1}{1+\exp(\beta_0+\beta_1 x_i)}$

3.4.2 Maximum likelihood estimates

- Chose those estimates for β_0 and β_1 which maximizes the log-likelihood.

- Find maximum likelihood estimates for β_0 and β_1: b_0 and b_1.
- Substitute these into the response function to obtain *fitted response function* $\hat{p}$: $\hat{p}_i = \frac{\exp(b_0+b_1 x_i)}{1+\exp(b_0+b_1 x_i)}$
- Use the logit transformation to obtain *fitted logit response function* $\hat{p}' = b_0 + b_1 x$ with $\hat{p}' = \log_e\left(\frac{\hat{p}}{1-\hat{p}}\right) = b_0 + b_1 x = \log(\hat{odds})$

3.5 How to obtain parameter estimates in R

A maximum likelihood estimation procedure can be used to obtain the parameter estimates. Since no analogical procedure exists, an iterative procedure is employed to obtain these estimates.

Example *Political party*

```
PP <- political_party
names(PP)
```

```
## [1] "subid"                  "political_party"        "gender"
## [4] "pro_capital_punishment" "pro_welfare_reform"     "pro_fed_support_ed"
## [7] "Republican"
```

```
head(PP)
```

```
## # A tibble: 6 x 7
##   subid political_party gender pro_capital_pun~ pro_welfare_ref~
##   <dbl>           <dbl>  <dbl>            <dbl>            <dbl>
## 1     1               2      0                3                6
## 2     2               2      0                2                5
## 3     3               2      0                1                6
## 4     4               2      0                4                7
## 5     5               2      0                4                6
## 6     6               2      0                4                6
## # ... with 2 more variables: pro_fed_support_ed <dbl>, Republican <dbl>
```

```
# Model 1
glm.log1 <- glm(Republican ~ pro_capital_punishment, family = binomial(link = logit), data = PP)
summary(glm.log1)
```

```
##
## Call:
## glm(formula = Republican ~ pro_capital_punishment, family = binomial(link = logit),
##     data = PP)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3090  -0.9461  -0.8183   1.3498   1.6646
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -1.44778    0.38716   -3.74 0.000184 ***
## pro_capital_punishment  0.17520    0.08263    2.12 0.033988 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 361.18  on 282  degrees of freedom
## Residual deviance: 356.62  on 281  degrees of freedom
```

```
## AIC: 360.62
##
## Number of Fisher Scoring iterations: 4
```

The estimated logistic regression function is:
$\hat{p} = P(Republican = 1) = \frac{exp(-1.448+0.175 \cdot ProCapPun)}{1+exp(-1.448+0.175 \cdot ProCapPun)}$

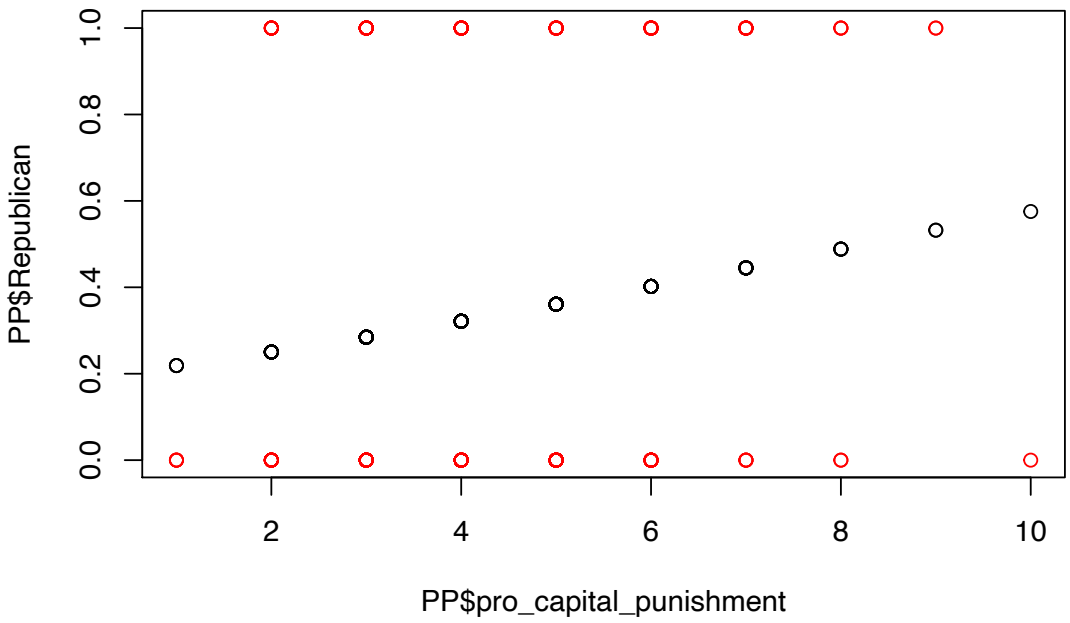
1. Look at predicted values

```
combine <- data.frame(cbind(PP$pro_capital_punishment, PP$Republican, fitted(glm.log1)))
colnames(combine) <- c("pro_capital_punishment", "Republican", "Fitted value")
head(combine,5)
```

```
##   pro_capital_punishment Republican Fitted value
## 1                      3          0    0.2845133
## 2                      2          0    0.2502308
## 3                      1          0    0.2188158
## 4                      4          0    0.3214787
## 5                      4          0    0.3214787
```

2. This predicted and observed values can be visualized in the following graph.

```
plot(PP$pro_capital_punishment, PP$Republican, type="p", col="red")
points(PP$pro_capital_punishment, fitted(glm.log1), col="black")
```



3.6 Interpretation of b_1

3.6.1 General

- The interpretation of b_1 is not the same interpretation as the slope in a linear regression model. (= the value of b is there the change in the outcome resulting from a one unit change in the predictor variable)
- The interpretation of b_1 can be: The value of b_1 is the change in the logit of the outcome resulting from a one unit change in the predictor variable.
- We will explain the interpretation by using the concept of *odds*.

Consider the value of the fitted logit response function at $x = x_j$:
$\hat{p}'(x_j) = b_0 + b_1 x_j$

Consider the value of the fitted logit response function at $x = x_j + 1$ (a one unit increase):
$\hat{p}'(x_j + 1) = b_0 + b_1(x_j + 1)$

The difference between the two fitted values:
$\hat{p}'(x_j + 1) - \hat{p}'(x_j) = b_1$

Now, $\hat{p}' = \log_e\left(\frac{\hat{p}}{1-\hat{p}}\right) =$ log of the estimated odds.

$b_1 =$ the difference between the two fitted values:
$b_1 = \hat{p}'(x_j + 1) - \hat{p}'(x_j)$
$b_1 = \log_e(\hat{odds}_{x+1}) - \log_e(\hat{odds}_x)$
$b_1 = \log_e\left(\frac{\hat{odds}_{x+1}}{\hat{odds}_x}\right)$

$\Rightarrow$ Odds ratio $= OR = \frac{\hat{odds}_{x+1}}{\hat{odds}_x} = \exp(b_1)$

$\Rightarrow \hat{odds}_{x+1} = \exp(b_1) \cdot \hat{odds}_x$

$\Rightarrow$ **The estimated odds are multiplied by** $\exp(b_1)$ **for any unit increase in** x**.**

3.6.2 Example interpreting odds ratio for continuous explanatory variable

Example *Political party*
Here the explanatory variable is a continuous variable (`pro_capital_punishment`)

To obtain the odds ratio, we need to know $\exp(b_1)$

```
glm.log1$coefficients
```

```
##            (Intercept) pro_capital_punishment
##             -1.4477794              0.1751987
```

```
exp(glm.log1$coefficients)
```

```
##            (Intercept) pro_capital_punishment
##              0.2350917              1.1914830
```

Thus $b_1 = 0.175$ and $\exp(b_1) = 1.191$.

The estimated odds are multiplied by 1.20 **for any unit increase in** *pro_capital_punishment*.

Interpretation: The odds of voting Republican is 1.20 times larger for each additional point on the *pro_capital_punishment* score.

Remark:

1. Since $\exp(b_1) = 1.191 > 1$, it indicates that as the predictor increases, the odds of the outcome occurring increase.
2. Consider `subject 1` who has `pro_capital_punishment = 3` and consider `subject 4` who has `pro_capital_punishment = 4` (and hence a one unit increase of the explanatory variable).

```r
head(combine, 5)
```

```
##   pro_capital_punishment Republican Fitted value
## 1                      3          0    0.2845133
## 2                      2          0    0.2502308
## 3                      1          0    0.2188158
## 4                      4          0    0.3214787
## 5                      4          0    0.3214787
```

$Odds\ for\ Republican = \frac{P(Republican=1)}{P(Republican=0)}$

$Odds\ subject\ 4 = 1.20 \cdot Odds\ subject\ 1$
The odds to vote Republican is 1.20 times higher for `subject 4` compared to `subject 1`.

	$P(Republican = 1)$	$P(Republican = 0)$	$Odds$	$Odds\ ratio$
Subject 1	0.28451	0.71549	0.39764	1.1915
Subject 4	0.32148	0.67852	0.47379	

3.7 Simple logistic regression model with categorical explanatory variable

Predictor variables can be categorical. When you want to use these in logistic regression models, you have to be aware of the way R is coding the categories in order to correctly interpret the results.

3.7.1 Use of binary predictor variables

Binary predictor variables should be coded as 0 or 1.

Example *Political party*
The binary predictor *gender* is coded as 0 (female) and 1 (male).

Gender	Coding (data set)
Female	0
Male	1

We use a logistic regression model for $P(Republican = 1)$ with *gender* as only explanatory variable.

```r
# Logistic regression with binary explanatory variable
glm.log2 <- glm(Republican ~ gender, family = binomial(link = "logit"), data = PP)
summary(glm.log2)
```

```
##
## Call:
## glm(formula = Republican ~ gender, family = binomial(link = "logit"),
##     data = PP)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2557  -0.5749  -0.5749   1.1010   1.9400
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.1823     0.1748   1.043    0.297
## gender       -1.8989     0.2861  -6.638 3.19e-11 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 361.18  on 282  degrees of freedom
## Residual deviance: 310.76  on 281  degrees of freedom
## AIC: 314.76
##
## Number of Fisher Scoring iterations: 3
```

With odds ratio

```
# To obtain the odds ratio
exp(glm.log2$coefficients)
```

```
## (Intercept)      gender
##   1.2000000   0.1497396
```

Interpreting the odds ratio

- The odds ratio to vote Republican for males to females is 0.15 (which is $\exp(-1.9)$).
- The odds to vote Republican for males is 0.15 times the odds to vote Republican for females.
- The odds to vote Republican for females is $\frac{1}{0.15} = 6$ times the odds to vote Republican for males.

Remark:
The logistic regression model is estimated by
$\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = 0.182 - 1.9 \cdot gender$ with $p = P(Republican = 1)$.

- $\log(\hat{odds})$ for voting Republican for females ($gender = 0$):
  $\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = 0.182$
- $\log(\hat{odds})$ for voting Republican for males ($gender = 1$):
  $\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = 0.182 - 1.9 = -1.718$

$\log(\hat{odds}_{female})$	0.182
$\hat{odds}_{female}$	1.20
$\log(\hat{odds}_{male})$	-1.718
$\hat{odds}_{male}$	0.18
Odds Ratio (male to female)	0.15
Odds Ratio (female to male)	6.67

We can ask for the estimated probabilities

```
combine2 <- data.frame(cbind(PP$gender, PP$Republican, fitted(glm.log2)))
colnames(combine2) <- c("gender", "Republican", "Fitted value")
combine2[c(1,18:22),]
```

```
##    gender Republican Fitted value
## 1       0          0    0.5454545
## 18      0          0    0.5454545
## 19      0          0    0.5454545
## 20      0          0    0.5454545
## 21      1          0    0.1523179
## 22      1          0    0.1523179
```

3.7.2 Use of categorical predictor variable (not binary)

Example *Titanic*

For this example, import the data set *titanic.xlsx* as `titanic`.

```
names(titanic)
```

```
## [1] "Class"     "Age"       "Sex"       "survived"  "Class_New"
```

We want to investigate whether the variable *Class_New* can be used as predictor variable for surviving the titanic. The *Class_New* variable is a categorical predictor with 4 levels as indicated below:

Name	Description
Class_New	1: 1st class 2: 2nd class 3: 3rd class 4: crew
survived	0: no 1: yes
Sex	0: female 1: male
Age (group)	0: child 1: adult

Since *Class_New* is numeric, R assumes by default that it is continuous. Therefore, we use the function `as.factor()`

```
titanic$class.f <- as.factor(titanic$Class_New)
glm.log1 <- glm(survived ~ class.f, family = binomial(link = logit), data = titanic)
summary(glm.log1)
```

```
##
## Call:
## glm(formula = survived ~ class.f, family = binomial(link = logit),
##     data = titanic)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3999  -0.7623  -0.7401   0.9702   1.6906
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.5092     0.1146   4.445 8.79e-06 ***
## class.f2     -0.8565     0.1661  -5.157 2.51e-07 ***
## class.f3     -1.5965     0.1436 -11.114  < 2e-16 ***
## class.f4     -1.6643     0.1390 -11.972  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2769.5  on 2200  degrees of freedom
## Residual deviance: 2588.6  on 2197  degrees of freedom
## AIC: 2596.6
##
## Number of Fisher Scoring iterations: 4
```

Then 3 new Dummy variables are created. By default, the first category is the reference category. Here, we want to compare the crew ($Class\_New = 4$). Hence, we take this category as reference category.

```r
# We want to change the reference group.
# We want Class_New = 4 to be the reference category.
titanic$Class_Ref <- relevel(titanic$class.f, ref = "4")
glm.log2 <- glm(survived ~ Class_Ref, family = binomial(link = logit), data = titanic)
summary(glm.log2)
```

```
##
## Call:
## glm(formula = survived ~ Class_Ref, family = binomial(link = logit),
##     data = titanic)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3999  -0.7623  -0.7401   0.9702   1.6906
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.15516    0.07876 -14.667  < 2e-16 ***
## Class_Ref1   1.66434    0.13902  11.972  < 2e-16 ***
## Class_Ref2   0.80785    0.14375   5.620 1.91e-08 ***
## Class_Ref3   0.06785    0.11711   0.579    0.562
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2769.5  on 2200  degrees of freedom
## Residual deviance: 2588.6  on 2197  degrees of freedom
## AIC: 2596.6
##
## Number of Fisher Scoring iterations: 4
```

```r
# To obtain the odds ratio
exp(glm.log2$coefficients)
```

```
## (Intercept)  Class_Ref1  Class_Ref2  Class_Ref3
##   0.3150074   5.2822069   2.2430799   1.0702008
```

3.7.2.1 How to interpret the odds ratio?

- Odds ratio of *1st class* to crew = 5
  The odds to survive the titanic is 5 times larger for passengers from first class than for the crew.
- Odds ratio of *2nd class* to crew = 2
  The odds to survive the titanic is 2 times larger for passengers from second class than for the crew.
- Odds ratio of *3rd class* to crew = 1 and is not significant.

3.7.2.2 The model is estimated as

Let $p = P(survived = 1)$, then
$\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = -1.155 + 1.664 \cdot Ind_{Class1} + 0.808 \cdot Ind_{Class2} + 0.068 \cdot Ind_{Class3}$

- For passengers from *Class 1*:
  $\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = -1.155 + 1.664 = 0.509$
- For passengers from *Class 2*:
  $\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = -1.155 + 0.808 = -0.347$

- For passengers from *Class 3*: Not significant different than for the crew
  $\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = -1.155 + 0.068 = -1.087$

Class	$\log(odds)$	$odds$	$prob$	OR
Class 1	0.509	1.664	0.625	5.28
Class 2	-0.347	0.707	0.414	2.24
Class 3	-1.087	0.337	0.252	1.07
Crew	-1.155	0.315	0.240	

3.8 Goodness of fit

There exists several measures to investigate the goodness-of-fit of your model.

- Chi-square goodness of fit test (to test whether the logistic response function is appropriate - see Hosmer and Lemeshow)
- Wald test of significant coefficients
- Deviance: $-2\cdot$ Log likelihood
- Pseudo R^2
- ROC curve (predictive power of the logistic model)

Example *Political party*
The logistic regression model (with $p = P(Republican = 1)$)
$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 \cdot pro\_capital\_punishment$
is estimated by
$\log\left(\frac{\hat{p}}{1-\hat{p}}\right) = -1.448 + 0.175 \cdot pro\_capital\_punishment$

What is the fit of this logistic model?

```
glm.log1 <- glm(Republican ~ pro_capital_punishment, family = binomial(link = logit), data = PP)
summary(glm.log1)
```

```
##
## Call:
## glm(formula = Republican ~ pro_capital_punishment, family = binomial(link = logit),
##     data = PP)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3090  -0.9461  -0.8183   1.3498   1.6646
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)           -1.44778    0.38716   -3.74 0.000184 ***
## pro_capital_punishment  0.17520    0.08263    2.12 0.033988 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 361.18  on 282  degrees of freedom
## Residual deviance: 356.62  on 281  degrees of freedom
## AIC: 360.62
##
## Number of Fisher Scoring iterations: 4
```

3.8.1 Hosmer-Lemeshow goodness of fit test

The Hosmer-Lemeshow goodness of fit test assess whether the predicted probabilities match the observed probabilities.

H_0 : **The logistic regression model fits the data.**
versus
H_1 : **The logistic regression model does not provide a good fit.** .

(Hence, here we hope to have a $p - value$ larger than the chosen significance level.)

Step 1: Based on the estimated logistic regression model, calculate the predicted probabilities of success for all observations.

Step 2: Order the data by these predicted probabilities (from small to large).

Step 3: Split the data into (approximately) 10 groups as follows. The first group consists of those observations with the lowest 10% predicted probabilities. The second group consists of the observations with the next 10% lowest predicted probabilities etc.

Reasoning of the Hosmer-Lemeshow test:
Suppose now (artificially) that our total sample size is 100 (and hence we have 10 groups of 10 observations each).

- Suppose now that all observations in the 1^{st} group have predicted probability 0.1.
- Then, if the H_0 is true, we would expect 1 observation that has $Y = 1$.
- If indeed H_0 is true, the observed proportion of observations with $Y = 1$ in that group will be around 0.1.
- In case we would have observed 8 observations in that 1^{st} group with $Y = 1$ then this would suggest that the model was not fitting the data well.

Step 4:

- Compute in each group the expected number of observations with $Y = 1$ and the observed number of observations with $Y = 1$.
- Compute in each group the expected number of observations with $Y = 0$ and the observed number of observations with $Y = 0$.

Remark:
How to compute the expected number of observations with $Y = 1$?
In practice, each observation in a group will have a different predicted probability.

- In every group we compute the average of the predicted probabilities for that group $(Y = 1) = \hat{\pi}_i$
- In every group, we can compute the expected number of observations with $(Y = 1) = n_i\hat{\pi}_i$, with n_i the number of observations in group i.

Step 5: We compute the Pearson goodness of fit statistic and the corresponding p-value.
Test statistic:
$\sum_{i=1}^{10} \frac{(O_{1i}-E_{1i})^2}{E_{1i}} + \frac{(O_{0i}-E_{0i})^2}{E_{0i}} \sim \chi^2_8$
with:

- O_{1i} the observed number of $(Y = 1)$ in the i^{th} group.
- E_{1i} the observed number of $(Y = 1)$ in the i^{th} group.
- O_{0i} the observed number of $(Y = 0)$ in the i^{th} group.
- E_{0i} the observed number of $(Y = 0)$ in the i^{th} group.

In R

The function `hoslem.test` (from package `ResourceSelection`) executes the Hosmer-Lemeshow goodness of fit test.

```
install.packages("ResourceSelection")
library(ResourceSelection)
```

```
Republican <- PP$Republican
hoslem <- hoslem.test(Republican, fitted(glm.log1))
hoslem
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  Republican, fitted(glm.log1)
## X-squared = 14.594, df = 8, p-value = 0.06754
```

```
combine <- cbind (hoslem$observed, hoslem$expected)
combine
```

```
##               y0 y1    yhat0     yhat1
## [0.219,0.25]  23  9 24.11828  7.881725
## (0.25,0.285]  36 22 41.49823 16.501769
## (0.285,0.321] 48 20 46.13945 21.860553
## (0.321,0.361] 57 15 46.02061 25.979391
## (0.361,0.402] 17 14 18.53389 12.466112
## (0.402,0.575]  7 15 11.68955 10.310450
```

Remark:
This test is not powerful when you have a small number of observations. You can only trust the $p-value$ if the underlying assumption for a Pearson chi-square statistic is satisfied. This assumes that the expected number of observations in each cell is at least 5 (at least in 20% of the cells).

3.8.2 Wald test to test significance of regression coefficients

Wald test is used to test the statistical significance of each covariate in the model.

Statement of hypotheses:
$H_0 : \beta_j = 0$
versus
$H_1 : \beta_j \neq 0$.

The test statistic of the Wald test is
$W = \frac{Estimate}{Standard\ Error}$

Under the null hypothesis, $W \sim N(0,1)$.

Example *Political party*

Statement of hypotheses:
$H_0 : \beta_{pro\_capital\_punishment} = 0$
versus
$H_1 : \beta_{pro\_capital\_punishment} \neq 0$.

```
summary(glm(formula = Republican ~ pro_capital_punishment, family = binomial(link = logit),
            data = PP))$coefficients
```

```
##                            Estimate Std. Error   z value     Pr(>|z|)
## (Intercept)             -1.4477794 0.38715651 -3.739520 0.0001843721
## pro_capital_punishment  0.1751987 0.08263244  2.120218 0.0339877032
```

$p-value = 0.034 < 0.05$, hence $pro\_capital\_punishment$ is a significant variable in this logistic model.

3.8.3 Deviance

$Deviance = -2 \cdot$ (Log-likelihood of fitted model)

Deviance is a statistic that compares the *log-likelihood of the fitted model* to the *log-likelihood of a saturated model*.

A **saturated model** is a model with n parameters that fits the n observations.
$\rightarrow$ n parameters for n observations
$\rightarrow$ perfect fit! (residuals will all be zero)
$\rightarrow$ **Log-likelihood for a saturated model** $= 0$**.**

Compare this log-likelihood value for the saturated model ($= 0$) with the log-likelihood value for the fitted model.

A **fitted model** is a logit model with less parameters than in the saturated model.
$\rightarrow$ # parameters in fitted model $<$ # parameters in saturated model
$\rightarrow$ log-likelihood fitted model $<$ log-likelihood saturated model ($= 0$)

We now look at the difference between both (=deviance)
Deviance
$= 2 \cdot$ (Log-likelihood of saturated model) $- 2 \cdot$ (Log-likelihood of fitted model)
$= 0 - 2 \cdot$ (Log-likelihood of fitted model)
$\rightarrow$ This difference is always positive

The smaller the *deviance*($= -2 \cdot$ (Log-likelihood of fitted model)), the closer the fitted model is to the saturated model.
$\rightarrow$ This statistic **can be used as a goodness of fit criterion!**

The larger the *deviance*($= -2 \cdot$ (Log-likelihood of fitted model)), the poorer the fit is between the fitted model and the saturated model.

Example *Political party*

```
summary(glm(formula = Republican ~ pro_capital_punishment, family = binomial(link = logit),
            data = PP))
```

```
##
## Call:
## glm(formula = Republican ~ pro_capital_punishment, family = binomial(link = logit),
##     data = PP)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3090  -0.9461  -0.8183   1.3498   1.6646
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -1.44778    0.38716   -3.74 0.000184 ***
## pro_capital_punishment  0.17520    0.08263    2.12 0.033988 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 361.18  on 282  degrees of freedom
## Residual deviance: 356.62  on 281  degrees of freedom
## AIC: 360.62
##
```

```
## Number of Fisher Scoring iterations: 4
```

$-2 \cdot$ (Log-likelihood of fitted model) $= 356.62$

In R: the deviance ($-2 \cdot$ (Log-likelihood of fitted model)) is also given and can be seen as a generalization of the residual (or error) sum of squares (regression analysis). It is often used as a measure to compare several models, each a subset of the other, and to test whether the model with more terms is significantly better than the model with fewer terms.

3.8.4 Pseudo R^2

- **In regression analysis**, the R^2 represents that proportion of variance which is explained by the regression model
  $R^2 = \frac{modelSS}{TotalSS} = \frac{TotalSS - ErrorSS}{TotalSS}$
- **In logistic regression**, it is not possible to compute an R^2 but we can define something similar. It expresses the proportional reduction in the log-likelihood measure. It **measures how the badness of fit improves** as a result of including explanatory variables.

$Pseudo\ R^2 = 1 - \frac{-2 \cdot (\text{Log-likelihood of fitted model})}{-2 \cdot (\text{Log-likelihood of null model})}$

The *null model* is the model with only the intercept.

In R

Computing pseudo R^2 in R with function `pR2()` from the package `pscl`

```
library(pscl)
pR2(glm.log1)
```

```
## fitting null model for pseudo-r2
```

```
##           llh       llhNull            G2      McFadden          r2ML
## -178.30939464 -180.59207832    4.56536736    0.01264000    0.01600262
##          r2CU
##    0.02219739
```

We only have a small value of 0.012 which means that we can only explain a small part of the deviance by the variable *pro_capital_punishment*.

3.9 Classification of observations

Example *Political party*

Dependent variable: *Republican*

Predictor variable: *pro_capital_punishment*

```
glm.log1 <- glm(Republican ~ pro_capital_punishment, family = binomial(link = logit), data = PP)
combine <- data.frame(cbind(PP$pro_capital_punishment, PP$Republican, fitted(glm.log1)))
colnames(combine) <- c("pro_capital_punishment", "Republican", "Fitted value")
head(combine,5)
```

```
##   pro_capital_punishment Republican Fitted value
## 1                      3          0    0.2845133
## 2                      2          0    0.2502308
## 3                      1          0    0.2188158
## 4                      4          0    0.3214787
## 5                      4          0    0.3214787
```

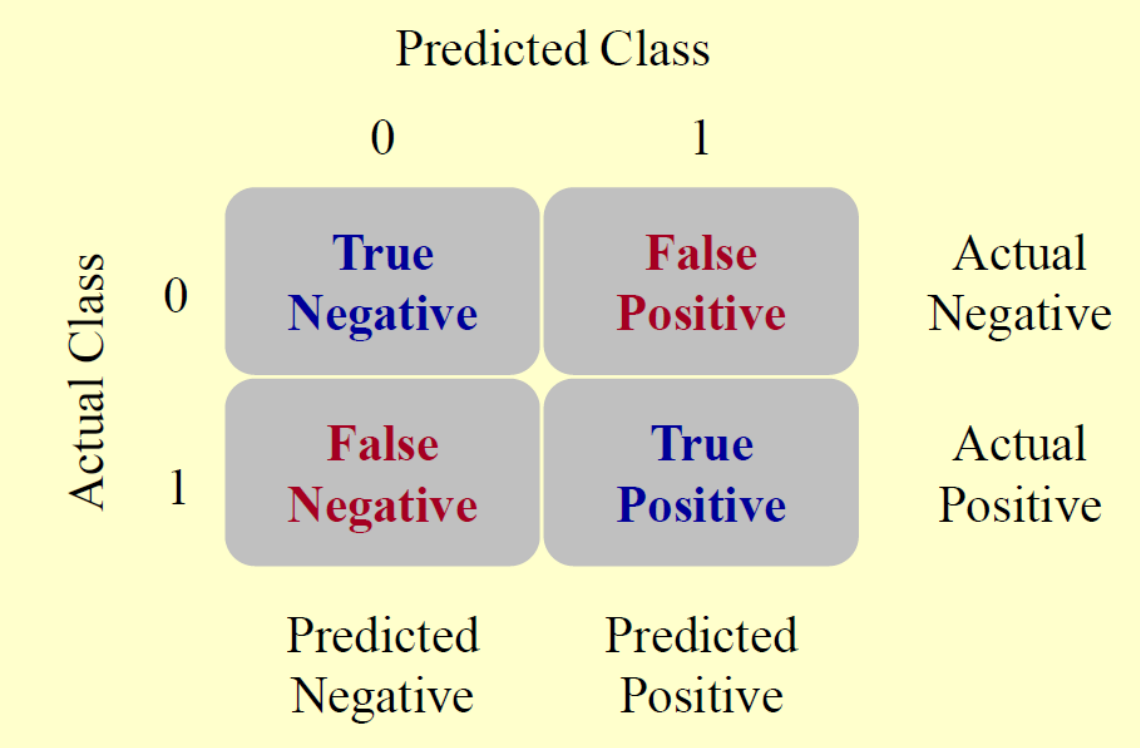
Before observations can be classified, the probabilities needs to be estimated. By using the fitted model, the estimated probability (predicted value) can be computed for each observation. Next, these probabilities can be used to classify observations into two groups.

If predicted probability > 0.5 then observation is classified as voting Republican ($pred\_group = 1$).
If predicted probability < 0.5 then observation is classified as not voting Republican ($pred\_group = 0$).

Classification table:

```
table(Republican, fitted(glm.log1) > 0.5)
```

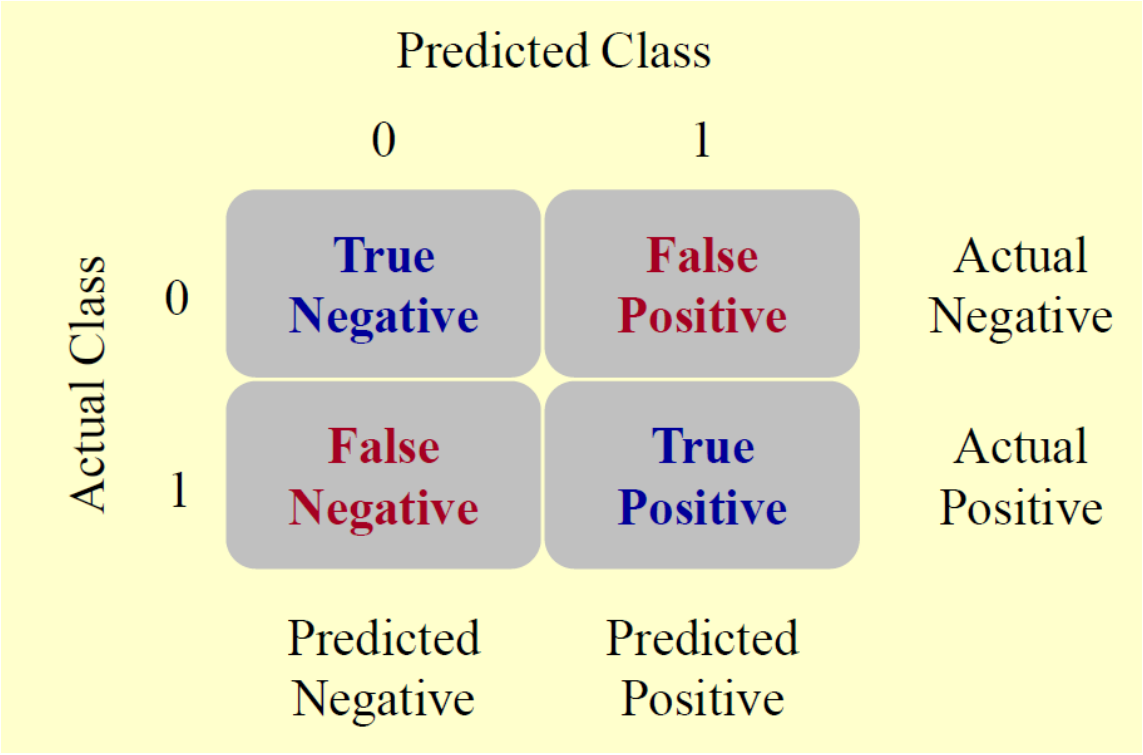
```
##
## Republican FALSE TRUE
##          0   187    1
##          1    93    2
```



3.10 ROC curve

3.10.1 What is a ROC curve?

The Receiving Operating Characteristic (ROC) curves are graphs that are used to evaluate and compare the performance of classification models. The **ROC curve** is a visual measure for the predictive ability of the (logistic) regression model. The area under the ROC curve (which is abbreviated as AUC) indicates the performances of a binary classifier in a single value.



The following terms are important for understanding the ROC curve:

- **False positive**: Non-event (actual class = 0) which is predicted as event (predicted class = 1).
- **False negative**: Event (actual class = 1) which is predicted as non-event (predicted class = 0).
- **Sensitivity**: Proportion of events (actual class = 1) which are predicted as events (predicted class = 1). The sensitivity is also referred to as *true positive rate*.
  $Sensitivity = \frac{True\ positives}{True\ positives + False\ negatives}$
- **Specificity**: Proportion of non-events (actual class = 0) which are predicted as non-events (predicted class = 0).
  $Specificity = \frac{True\ negatives}{True\ negatives + False\ positives}$
- The **false positive rate** is proportion of non-events (actual class = 0) got incorrectly classified by the classifier.
  $false\ positive\ rate = 1 - specificity = \frac{False\ postives}{True\ negatives + False\ positives}$

These values vary according to the chosen cut-off value.

Example *Political party*

1. We have obtained this classification table for a *cut-off value of 0.5*.

Classification table:

```
table(Republican, fitted(glm.log1) > 0.5)
```

```
##
## Republican FALSE TRUE
##          0   187    1
##          1    93    2
```

$Sensitivity = \frac{2}{2+93} = 0.021$
$Specificity = \frac{187}{187+1} = 0.995$

$False\ positive\ rate = \frac{1}{187+2} = 0.005$

2. For a *cut-off value of 0.9*.
   A high-cut off value implies that almost everything is predicted as a non-event. Hence sensitivity will be small and false positive will be small.

Classification table:

```
table(Republican, fitted(glm.log1) > 0.9)
```

```
##
## Republican FALSE
##          0   188
##          1    95
```

$Sensitivity = \frac{0}{95} = 0$
$Specificity = \frac{188}{188} = 1$
$False\ positive\ rate = \frac{0}{188} = 0$

3. For a *cut-off value of 0.1*.
   A low-cut off value implies that almost everything is predicted as a success. Hence sensitivity will be high and false positive will be high.

Classification table:

```
table(Republican, fitted(glm.log1) > 0.1)
```

```
##
## Republican TRUE
##          0  188
##          1   95
```

$Sensitivity = \frac{95}{95} = 1$
$Specificity = \frac{0}{188} = 0$
$False\ positive\ rate = \frac{188}{188} = 1$

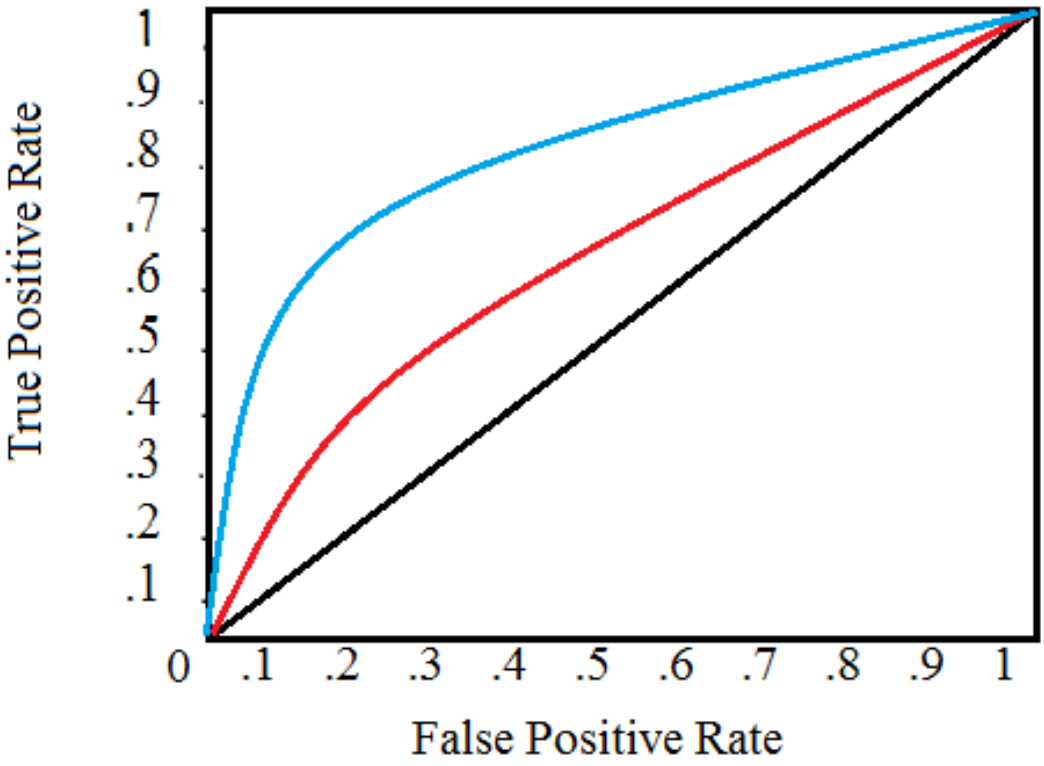
4. The optimal solution would be to have:
   - A small proportion of false positive
   - A large number of sensitivity

Once we have computed sensitivity and specificity pairs for each possible cutoff point, the ROC curve is a plot of sensitivity on the y axis by false positive rate (=1-specificity) on the x axis.

This curve is called the receiver operating characteristic (ROC) curve. The area under the ROC curve ranges from 0.5 and 1.0 where larger values indicate a better fit.

The image below shows ROC curves of a few logistic regression models. [1]

[1] Figure is from https://www.statisticshowto.com/receiver-operating-characteristic-roc-curve/.

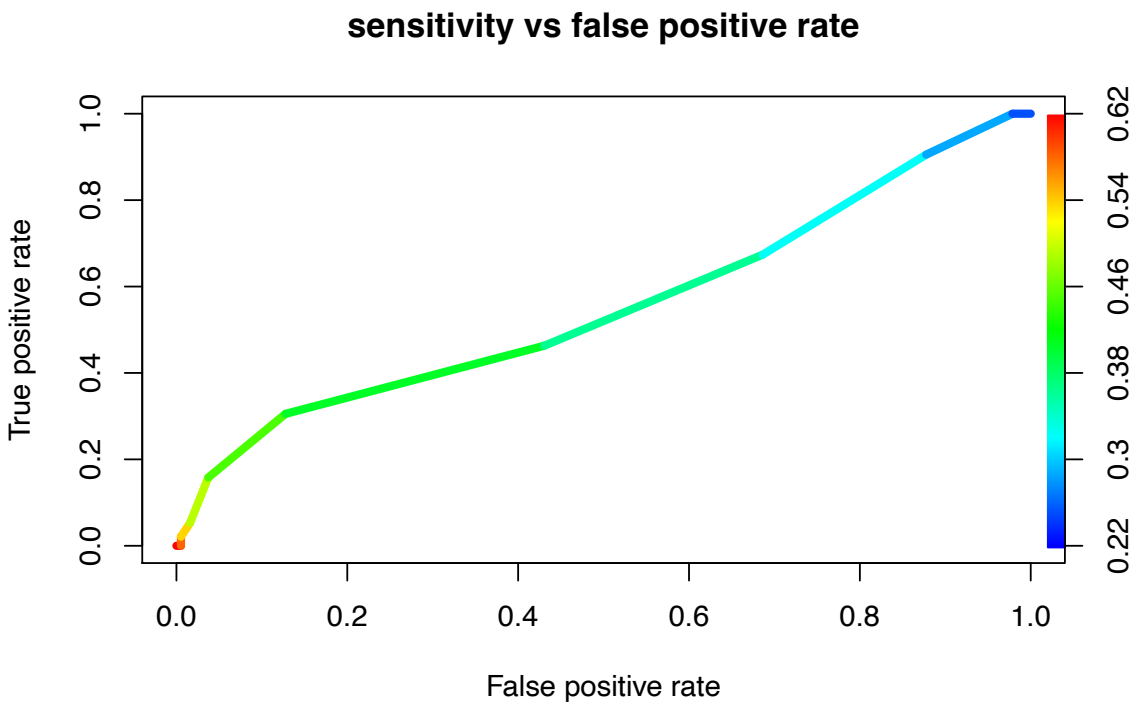


The classifier corresponding to the red curve is less accurate than the classifier corresponding to the blue curve.

3.10.2 How to obtain the ROC curve in R

```
install.packages("ROCR")
library(ROCR)
```

```
glm.log1 <- glm(Republican ~ pro_capital_punishment, family = binomial(link = logit), data = PP)
predict <- fitted(glm.log1)
pred <- prediction(predict, Republican)
perf <- performance(pred, measure = "tpr", x.measure = "fpr")
# Colorize argument in following plot function:
# This logical determines whether the curve(s)
# should be colorized according to cutoff
plot(perf, main = "sensitivity vs false positive rate", colorize = TRUE,
     colorkey.relwidth = 0.5, lwd = 4.5)
```



X-axis: False positive rate = 1 - specificity
Y-axis: True positive rate = sensitivity

Area under the ROC curve:

```
perf_auc <- performance(pred, measure = "auc")
perf_auc@y.values
```

```
## [[1]]
## [1] 0.5539194
```

We here have an AUC (area under the ROC curve) of 0.554 which is not good. The model does not have a good discriminating ability.

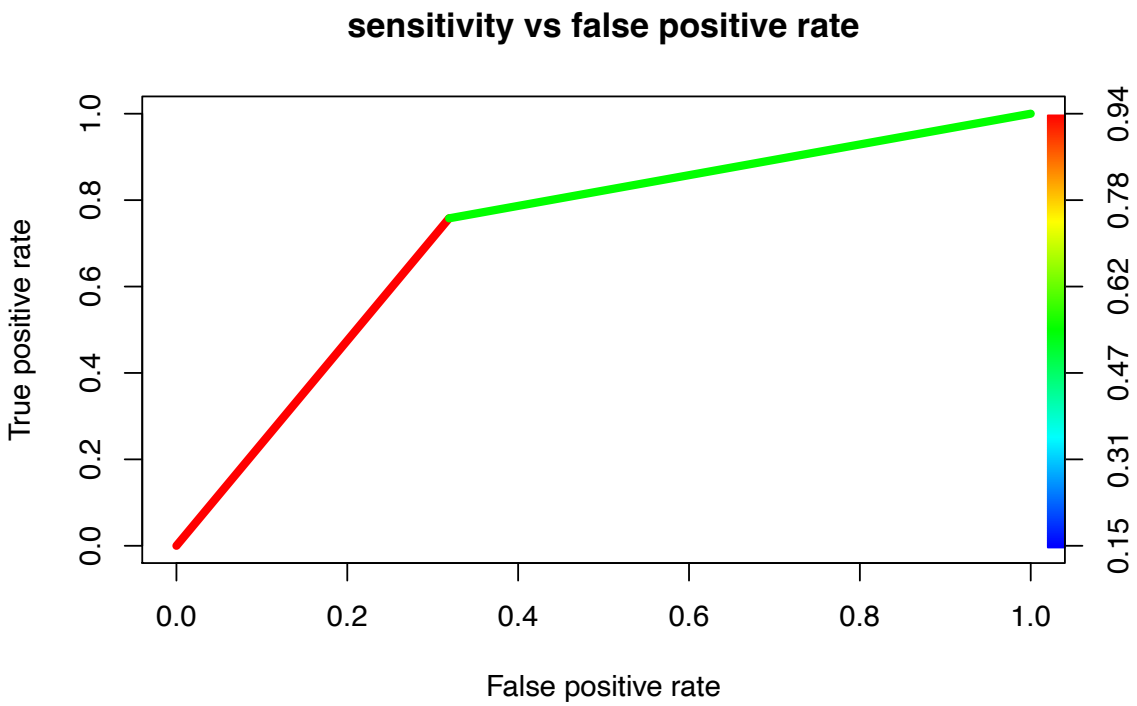
Remark:
Models with a higher predictive power has a higher AUC.

3.10.3 Example

Example ***Political party***

Dependent variable: *Republican*
Predictor variable: *gender*
Plot the ROC curve and compute the AUC.

```
glm.log2 <- glm(Republican ~ gender, family = binomial(link = "logit"), data = PP)
pred2 <- prediction(fitted(glm.log2), Republican)
perf2 <- performance(pred2, measure = "tpr", x.measure = "fpr")
plot(perf2, main = "sensitivity vs false positive rate", colorize = TRUE,
     colorkey.relwidth = 0.5, lwd = 4.5)
```

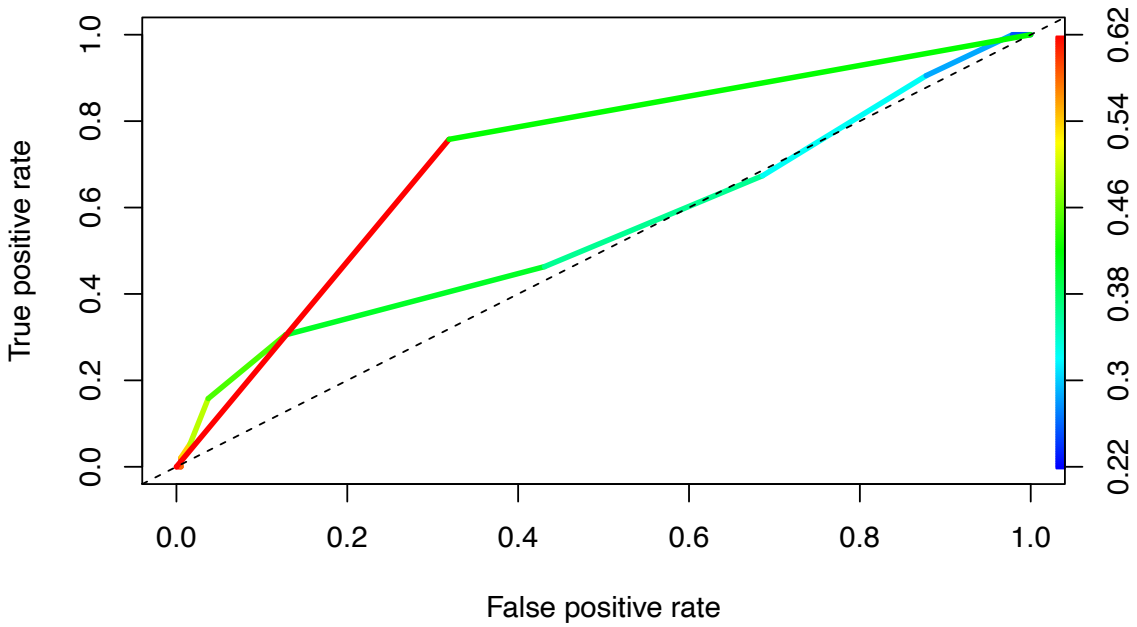



```
performance(pred2, measure = "auc")@y.values
```

```
## [[1]]
## [1] 0.7193729
```

Remark:
In case you want to show two ROC curves on the same plot:

```
plot(perf, colorize = TRUE, lwd = 3)
plot(perf2, add = TRUE, colorize = TRUE, lwd = 3)
abline(0, 1, lty = 2)
```



4 Multiple logistic regression

4.1 General

In **simple logistic regression**, we have only 1 predictor variable:
$P(Y = 1) = E(Y) = \frac{\exp(\beta_0+\beta_1 x)}{1+\exp(\beta_0+\beta_1 x)}$

In case we have **more predictor variables** (e.g., p), the model becomes
$P(Y = 1) = E(Y) = \frac{\exp(\beta_0+\beta_1 x_1+\beta_2 x_2+...+\beta_p x_p)}{1+\exp(\beta_0+\beta_1 x_1+\beta_2 x_2+...+\beta_p x_p)}$.

Or the model can be written as
$P(Y = 1) = \frac{1}{1+\exp(-\beta_0-\beta_1 x_1-\beta_2 x_2-...-\beta_p x_p)}$.

By using the *logit transformation*
$p' = \log_e\left(\frac{p}{1-p}\right)$
we obtain the *logit response function*:
$p' = \log_e\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p$.

Properties:

- monotonic and sigmoid in shape
- Almost linear when p is between 0.2 and 0.8
- Predictor variables may be interaction effects, curvature, quantitative qualitative.
- A logistic regression model with only qualitative variables is called a *log-linear* model
- Maximum likelihood estimation is used to find estimates for the parameters.

4.2 Example

Example *Political party*

We now perform a logistic regression analysis with 4 explanatory variables of which 3 scale variables ($pro\_capital\_punishment, pro\_welfare\_reform$ and $pro\_fed\_support\_ed$) and 1 indicator variable ($gender$).

4.2.1 Hierarchical step by step (manually)

Model A: model with 4 explanatory variables

```
glm.log.A <- glm(Republican ~ pro_capital_punishment + pro_welfare_reform +
                   pro_fed_support_ed + gender, family = binomial(link = logit),
                 data = PP)
summary(glm.log.A)
```

```
##
## Call:
## glm(formula = Republican ~ pro_capital_punishment + pro_welfare_reform +
##     pro_fed_support_ed + gender, family = binomial(link = logit),
##     data = PP)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0351  -0.7323  -0.3916   0.8786   2.2831
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -2.84174    0.96470  -2.946  0.00322 **
## pro_capital_punishment  0.67520    0.13240   5.100 3.40e-07 ***
## pro_welfare_reform      0.06017    0.10056   0.598  0.54963
## pro_fed_support_ed      0.04408    0.11274   0.391  0.69580
## gender                 -3.07436    0.41479  -7.412 1.25e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 361.18  on 282  degrees of freedom
## Residual deviance: 273.97  on 278  degrees of freedom
## AIC: 283.97
##
## Number of Fisher Scoring iterations: 5
```

Model B: model with 3 explanatory variables

Since Federal Support Education is not significant, we drop this variable from the model and refit the model.

```
glm.log.B <- glm(Republican ~ pro_capital_punishment + pro_welfare_reform + gender,
                 family = binomial(link = logit), data = PP)
summary(glm.log.B)
```

```
##
## Call:
## glm(formula = Republican ~ pro_capital_punishment + pro_welfare_reform +
##     gender, family = binomial(link = logit), data = PP)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -3.0417  -0.7337  -0.3903   0.8987   2.2858
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)             -2.60435    0.74361  -3.502 0.000461 ***
## pro_capital_punishment   0.68069    0.13209   5.153 2.56e-07 ***
## pro_welfare_reform       0.05909    0.10043   0.588 0.556316
## gender                  -3.06831    0.41452  -7.402 1.34e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 361.18  on 282  degrees of freedom
## Residual deviance: 274.12  on 279  degrees of freedom
## AIC: 282.12
##
## Number of Fisher Scoring iterations: 5
```

Model C: model with 2 explanatory variables
Since $pro\_welfare\_reform$ is not significant, we drop this variable from the model and refit the model.

```
glm.log.C <- glm(Republican ~ pro_capital_punishment + gender,
                 family = binomial(link = logit), data = PP)
summary(glm.log.C)
```

```
##
## Call:
## glm(formula = Republican ~ pro_capital_punishment + gender, family = binomial(link = logit),
##     data = PP)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0501  -0.7243  -0.3807   0.9068   2.3068
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)              -2.2777     0.4873  -4.675 2.95e-06 ***
## pro_capital_punishment    0.6920     0.1308   5.290 1.22e-07 ***
## gender                   -3.0782     0.4143  -7.429 1.09e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 361.18  on 282  degrees of freedom
## Residual deviance: 274.47  on 280  degrees of freedom
## AIC: 280.47
##
## Number of Fisher Scoring iterations: 5
```

In this model, all variables are significant.

4.2.2 Comparison of several models

Suppose you want to compare the following models

- Model 0: Intercept only
- Model 1: $gender$
- Model 2: $gender + pro\_capital\_punishment$
- Model 3: $gender + pro\_capital\_punishment + pro\_welfare\_reform$
- Model 4: $gender + pro\_capital\_punishment + pro\_welfare\_reform + pro\_fed\_support\_ed$

Source	Deviance ($= -2 \cdot$ log-likelihood)	pseudo R^2
Model 0: Intercept only	361.184	
Model 1: $gender$	310.764	0.16
Model 2: $gender +$ $pro\_capital\_punishment$	274.472	0.26
Model 3: $gender +$ $pro\_capital\_punishment +$ $pro\_welfare\_reform$	274.124	0.24
Model 4: $gender +$ $pro\_capital\_punishment$ $+pro\_welfare\_reform +$ $pro\_fed\_support\_ed$	273.971	0.24

Comparing models by comparing the deviances

- For each fitted model, the deviance is calculated, which is $-2 \cdot$ Log-Likelihood.
- Difference between the deviance for two fitted models can be used to compare two nested models. This concept is explained in the next topic.

4.3 Partial deviance

4.3.1 General

1. **Full logistic model**: model with response function (and $p-1$ predictor variables). Where
   $E(Y) = \frac{\exp(\mathbf{x}'\beta_\mathbf{F})}{1+\exp(\mathbf{x}'\beta_\mathbf{F})}$
   with $\mathbf{x}'\beta_\mathbf{F} = \beta_0 + \beta_1 x_1 + ... + \beta_{p-1}x_{p-1}$
   Deviance for the *full model*: $DEV(x_1, ..., x_{p-1})$
2. **reduced logistic model: model with only $q-1$ predictor variables where $q < p$. Where
   $E(Y) = \frac{\exp(\mathbf{x}'\beta_\mathbf{R})}{1+\exp(\mathbf{x}'\beta_\mathbf{R})}$
   with $\mathbf{x}'\beta_\mathbf{R} = \beta_0 + \beta_1 x_1 + ... + \beta_{q-1}x_{q-1}$
   Deviance for the *reduced model*: $DEV(x_1, ..., x_{q-1})$
3. We want to check whether we can drop a set of predictor variables by formulating the null hypothesis:
   $H_0 : \beta_q = \beta_{q+1} = ... = \beta_{p-1} = 0$ with $(q < p)$.
   versus
   H_1 : **not all β_k in H_0 are equal to zero.**

If $DEV_{reduced}$ is not much larger than DEV_{full}
→ reduced model provides almost as close a fit as the full model → Do not reject H_0

If $DEV_{reduced}$ is much larger than DEV_{full}
→ reduced model provides much worse fit compared to the full model → Reject H_0

Partial deviance $= DEV(x_1, ..., x_{q-1}) - DEV(x_1, ..., x_{p-1})$

Properties:

- If H_0 holds and n is large, then *partial deviance* $\sim \chi^2_{p-q}$
- Decision rule:
  We compute the partial deviance and the corresponding p-value
  - If $p - value < 0.05$, then reject H_0
  - If $p - value > 0.05$, then do not reject H_0

4.3.2 Example

Example *Political party*

Source	Deviance ($= -2 \cdot$ log-likelihood)	pseudo R^2
Model 0: Intercept only	361.184	
Model 1: *gender*	310.764	0.16
Model 2: *gender* + *pro_capital_punishment*	274.472	0.26
Model 3: *gender* + *pro_capital_punishment* + *pro_welfare_reform*	274.124	0.24
Model 4: *gender* + *pro_capital_punishment* +*pro_welfare_reform* + *pro_fed_support_ed*	273.971	0.24

```
glm.log.M1 <- glm(Republican ~ gender, family = binomial(link = logit), data = PP)
glm.log.M2 <- glm(Republican ~ gender + pro_capital_punishment,
                  family = binomial(link = logit), data = PP)
glm.log.M3 <- glm(Republican ~ gender + pro_capital_punishment + pro_welfare_reform,
                  family = binomial(link = logit), data = PP)
glm.log.M4 <- glm(Republican ~ gender + pro_capital_punishment + pro_welfare_reform +
                    pro_fed_support_ed, family = binomial(link = logit), data = PP)
```

a) Compare model 2 to model 1

In model 1, we have *gender* as explanatory variable. In model 2, we have *gender* and *pro_capital_punishment* as explanatory variables. We are interested in the improvements of model 2 over model 1.

$H_0 : \beta_{pro\_capital\_punishment} = 0$
versus
$H_1 : \beta_{pro\_capital\_punishment} \neq 0$

Difference in deviance:
Partial deviance $= 310.8 - 274.5 = 36.3$.
This is the change in the deviance resulting from adding the variable *pro_capital_punishment* to the model.

Compare several models:

```
anova(glm.log.M1, glm.log.M2, test = "Chisq") # Note the argument 'test = "Chisq"'!
```

```
## Analysis of Deviance Table
##
## Model 1: Republican ~ gender
## Model 2: Republican ~ gender + pro_capital_punishment
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1       281     310.76
## 2       280     274.47  1   36.292 1.698e-09 ***
## ---
```

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

We have a $p-value$ of 0.00001 which is smaller than 0.05. Hence, the variable *pro_capital_punishment* should be added to the model because it improves the model.

b) Compare model 3 to model 2

$H_0 : \beta_{pro\_welfare\_reform} = 0$
versus
$H_1 : \beta_{pro\_welfare\_reform} \neq 0$

Difference in deviance $\$ = 0.348\$$

```
anova(glm.log.M2, glm.log.M3, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Republican ~ gender + pro_capital_punishment
## Model 2: Republican ~ gender + pro_capital_punishment + pro_welfare_reform
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       280     274.47
## 2       279     274.12  1  0.34849    0.555
```

We have a $p-value$ of 0.555. Since $p-value > 0.05$ the variable $pro\_welfare\_reform$ should not be added to the model because it has virtually no effect on the fit (the deviance has hardly changed).

c) Compare model 4 to model 2

$H_0 : \beta_{pro\_fed\_support\_ed} = \beta_{pro\_welfare\_reform} = 0$
versus
$H_1 : \beta_{pro\_fed\_support\_ed} \neq 0$ or $\beta_{pro\_welfare\_reform} \neq 0$ (At least one of these coefficients is not 0).

Difference in deviance $= 274.472 - 273.971 = 0.501$.
Degrees of freedom $= 4 - 2 = 2$

```
anova(glm.log.M2, glm.log.M4, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Republican ~ gender + pro_capital_punishment
## Model 2: Republican ~ gender + pro_capital_punishment + pro_welfare_reform +
##     pro_fed_support_ed
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       280     274.47
## 2       278     273.97  2  0.50122   0.7783
```

Compute the corresponding $p-value$ as $P(\chi^2_2 \geq 0.501) = 0.778$. The $p-value$ is large indicating that we can drop $pro\_welfare\_reform$ and $pro\_fed\_support\_ed$ from the model.

The optimal model is model 2 with *gender* and *pro_capital_punishment*.

4.4 Interpreting the output

4.4.1 Interpreting the parameter estimates

Example *Political party*

```
summary(glm.log.M2)$coefficients
```

```
##                          Estimate Std. Error   z value     Pr(>|z|)
## (Intercept)           -2.2777137  0.4872583 -4.674551 2.945979e-06
## gender                -3.0782127  0.4143275 -7.429419 1.090758e-13
```

```
## pro_capital_punishment  0.6919594  0.1307984  5.290274 1.221335e-07
```

```
exp(glm.log.M2$coefficients)
```

```
##             (Intercept)                  gender pro_capital_punishment
##              0.10251833             0.04604147             1.99762585
```

The estimated logistic regression function is
$\hat{p} = P(Republican = 1) = \frac{\exp(-2.278-3.078 \cdot gender+0.6920 \cdot pro\_captial\_punishment)}{1+\exp(-2.278-3.078 \cdot gender+0.6920 \cdot pro\_captial\_punishment)}$

$p' = \log_e \left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p$

$\hat{p}' = -2.278 - 3.078 \cdot gender + 0.6920 \cdot pro\_captial\_punishment$

- Coefficient for *gender*: $b_{gender} = -3.078$ odds ratio for *gender*: $OR_{gender} = \exp(b_{gender}) = 0.046$

The odds to vote Republican for male ($gender = 1$) is 0.05 times the odds to vote Republican for female ($gender = 0$), when taking *pro_captial_punishment* into account.
OR The odds to vote Republican for female is 20 times the odds to vote Republican for male, when taking *pro_captial_punishment* into account.

- Coefficient for *pro_capital_punishment*: $b_{pcp} = 0.692$ odds ratio for *pro_capital_punishment*: $OR_{pcp} = \exp(b_{pcp}) = 2.00$

Per one unit increase on the score of *pro_capital_punishment*, the odds for voting Republican is increasing 2 times, taking *gender* into account.

4.4.2 Classification table

```
table(Republican, fitted(glm.log.M2)>0.5)
```

```
##
## Republican FALSE TRUE
##          0   165   23
##          1    50   45
```

4.4.3 Generalized R^2 value

Mc Fadden R^2

```
library(pscl)
pR2(glm.log.M2)
```

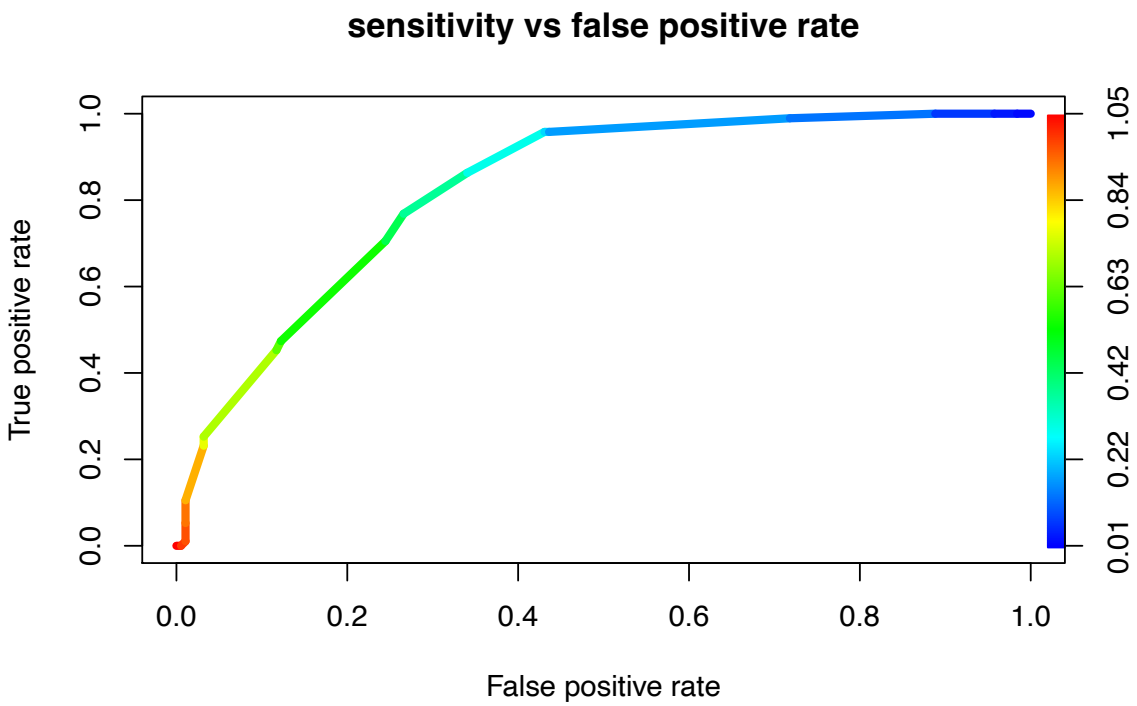
```
## fitting null model for pseudo-r2
```

```
##          llh      llhNull           G2     McFadden         r2ML         r2CU
## -137.2361584 -180.5920783   86.7118398    0.2400765    0.2639095    0.3660715
```

Pseudo R^2 value is 0.24.

4.4.4 Create the ROC curve and area under the curve

```
pred.M2 <- prediction(fitted(glm.log.M2), Republican)
perf.M2 <- performance(pred.M2, measure = "tpr", x.measure = "fpr")
plot(perf.M2, main = "sensitivity vs false positive rate",
     colorize = TRUE, colorkey.relwidth = 0.5, lwd = 4.5)
```




```
performance(pred.M2, measure = "auc")@y.values
```

```
## [[1]]
## [1] 0.8275756
```

The AUC is now 0.828

5 References

Meyers, L. S., Gamst, G. & Guarino, A.J. (2017) Applied Multivariate research, Design and Interpretation, 3rd ed., Sage Edge

Chapter 11: Introduction to Poisson regression

Contents

1 Introduction

Explanatory variables	Response variable	Method
Continuous	Continuous	Regression
Categorical	Continuous	ANOVA
Continuous	Dummy (0 or 1)	Logistic regression
Continuous	Count	Poisson regression

Poisson regression is used when the **Response variable is a count of something per unit or per time interval**.

Examples:

- number of people in an organization
- number of visits to a physician
- number of arrests in the past year

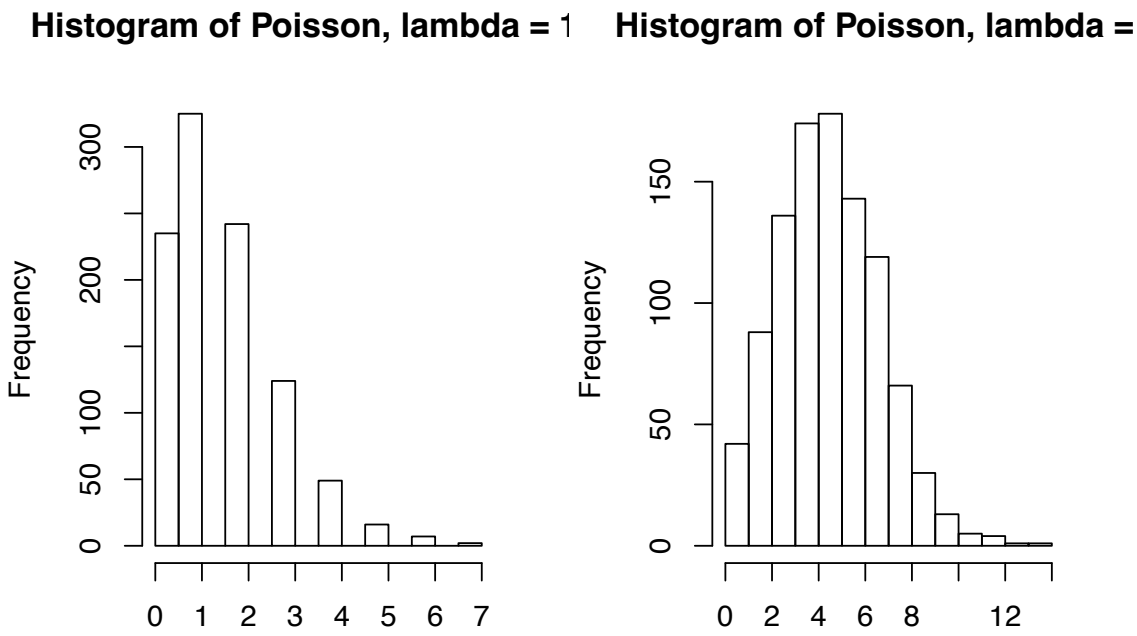
Poisson regression model assumes that the response variable Y has a Poisson distribution (with parameter λ): $Y \sim Poisson(\lambda)$.

We saw in chapter 3: "We often model counts per unit or counts per interval by a Poisson distribution. Let Y be the random variable associated with such a count, and let λ be the appropriate expected rate of occurrences. The probability function for Y is
$p(y) = \frac{\lambda^y}{y!}\exp(-\lambda)$ for $y = 0, 1, 2, ...$ and $\lambda > 0$.

For $Y \sim Poisson(\lambda)$, we can show that

$\mu = E(Y) = \lambda$
$\sigma^2(Y) = \lambda$
$\sigma(Y) = \sqrt{\lambda}$

```
lam_1.5 <-rpois(1000, 1.5)
lam_5 <- rpois(1000, 5)
par(mfrow = c(1,2))
hist(lam_1.5, breaks = 15, main = "Histogram of Poisson, lambda = 1.5", xlab="")
hist(lam_5, breaks = 15, main = "Histogram of Poisson, lambda = 5", xlab="")
```



Properties:

1. As λ increases, the mode moves away from 0.
2. $E(Y) = Var(Y) = \lambda$

In Poisson regression, one wants to express how the parameters λ depends on the explanatory variables.

Hence:

1. We write λ_i to allow the parameter to vary across individuals.
2. Because $\lambda > 0$, we usually express the $\log(\lambda)$ as a linear function of the explanatory variables:

   $$\log(\lambda_i) = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + ... + \beta_k x_{ki}$$

Example *horseshoe crabs*

We consider the data of the horseshoe crabs (see Agresti (1996)) with $n = 173$.
Import data set *crab.txt* as `crab` in R

```
crab <- read.table(file = file.choose(), header = TRUE)
names(crab)
head(crab, n = 6)
```

```
## [1] "obs" "C"   "S"   "Wt"  "W"   "Sa"
```

```
##   obs C S   Wt    W Sa
## 1   1 2 3 28.3 3.05  8
## 2   2 3 3 26.0 2.60  4
## 3   3 3 3 25.6 2.15  0
## 4   4 4 2 21.0 1.85  0
## 5   5 2 3 29.0 3.00  1
```

```
## 6   6 1 2 25.0 2.30  3
```

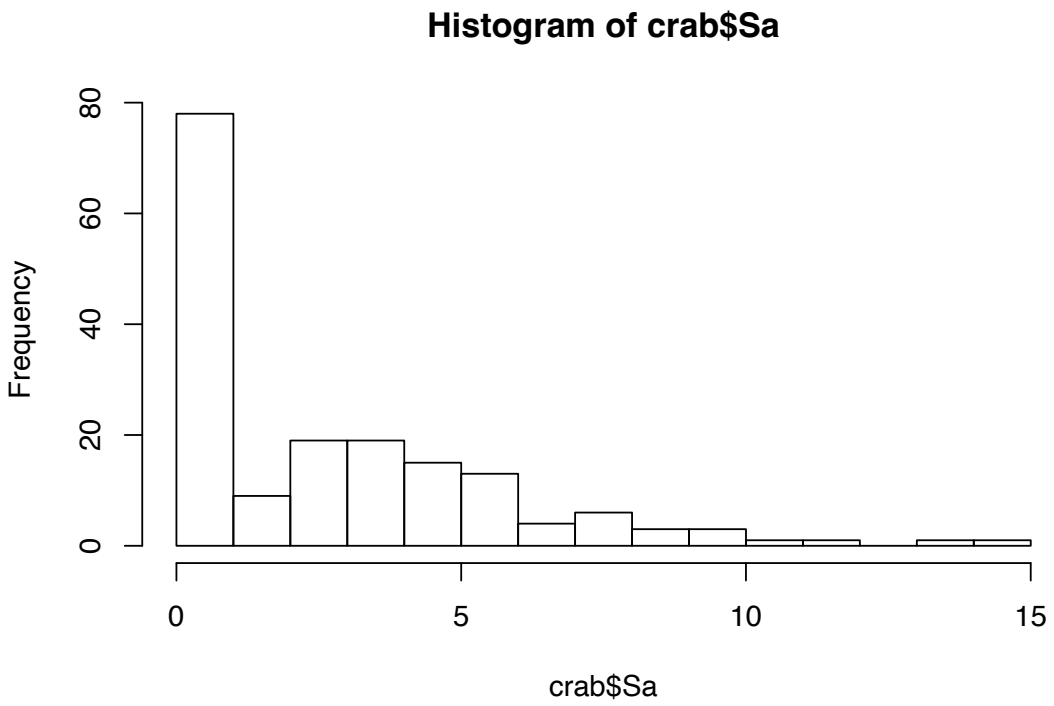
Each female horseshoe crab in the study had a male crab attached to her in her nest. The study investigates factors that affect whether the female crab had any other males, called satellites, residing near her.

The response outcome for each female crab is her number of satellites (Sa).

Explanatory variables that are thought to affect this are

- The female crab's color (C)
- Spine condition (S),
- Weight (Wt),
- Carapace width (W).

```
hist(crab$Sa, breaks = 15)
```



Descriptive statistics

```
list(mean = mean(crab$Sa), var = var(crab$Sa))
```

```
## $mean
## [1] 2.919075
##
## $var
## [1] 9.912018
```

2 The Poisson regression model

2.1 Poisson model for counts

In Poisson regression, we suppose that the expected count $E(Y) = \lambda$ can be determined by a set of explanatory variables:
$\log(\lambda) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_k x_k$

Here, we will only illustrate the simple Poisson regression model with one explanatory variable, we write:
$\log(\lambda) = \beta_0 + \beta_1 x_1$.
This is equivalent to:
$\lambda = \exp(\beta_0 + \beta_1 x) = \exp(\beta_0)\exp(\beta_1 x)$

2.2 Interpretation of parameter estimates

- $\exp(\beta_0)$ = effect on the mean of Y, (that is λ), when $x = 0$.

- $\exp(\beta_1)$ = with every unit increase in x, the predictor variable has multiplicative effect of $\exp(\beta_1)$ on the mean of Y, (that is λ).
  To see this, consider the following:
  - assume $\lambda_1 = \exp(\beta_0)\exp(\beta_1 x)$
  - assume $\lambda_2 = \exp(\beta_0)\exp(\beta_1(x+1))$
    → Hence, $\lambda_2 = \exp(\beta_1)\lambda_1$
    * If $\beta_1 = 0$, then $\exp(\beta_1) = 1$, the expected count $\lambda = E(Y) = \exp(\beta_0)$, and Y and x are not related.
    * If $\beta_1 > 0$, then $\exp(\beta_1) > 1$, and $\lambda_2 > \lambda_1$.
    * If $\beta_1 < 0$, then $\exp(\beta_1) < 1$, and $\lambda_2 < \lambda_1$.

2.3 Parameter estimation

Similar to the case of logistic regression, the **maximum likelihood estimators (MLEs) for** $\beta_0, \beta_1, \ldots$ **etc.** are obtained by finding the values that maximizes log-likelihood.

In general, there are no closed-form solutions, so the MLEs are obtained by using iterative algorithms such as *Newton-Raphson* (NR), *Iteratively reweighted least squares* (IRWLS), etc.

2.4 Illustration in R

Example *horseshoe crabs*

- Model 0: intercept only

Intercept-only model: $\log(\lambda) = \beta_0$
Fitting the intercept-only model. This model implies the expected number of satellites per each crab is the same.

```
model.0 <- glm(Sa ~ 1, family = poisson(link = log), data = crab)
summary(model.0)
```

```
##
## Call:
## glm(formula = Sa ~ 1, family = poisson(link = log), data = crab)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4162  -2.4162  -0.5707   1.1045   4.9942
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.0713     0.0445   24.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 632.79  on 172  degrees of freedom
## Residual deviance: 632.79  on 172  degrees of freedom
## AIC: 990.09
##
## Number of Fisher Scoring iterations: 5
```

In this case:
$E(Sa) = exp(1.0713) = 2.919$

Then $\log(\lambda) = 1.0713$
or $\lambda = 2.919$.

- Model 1: single explanatory variable

Poisson regression of number of satellites (Sa) on Width (W)

```
model.1 <- glm(Sa ~ W, family = poisson(link = log), data = crab)
summary(model.1)
```

```
##
## Call:
## glm(formula = Sa ~ W, family = poisson(link = log), data = crab)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9306  -1.9981  -0.5627   0.9299   4.9992
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.4282     0.1789  -2.394   0.0167 *
## W             0.5892     0.0650   9.065   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 632.79  on 172  degrees of freedom
## Residual deviance: 560.84  on 171  degrees of freedom
## AIC: 920.14
##
## Number of Fisher Scoring iterations: 5
```

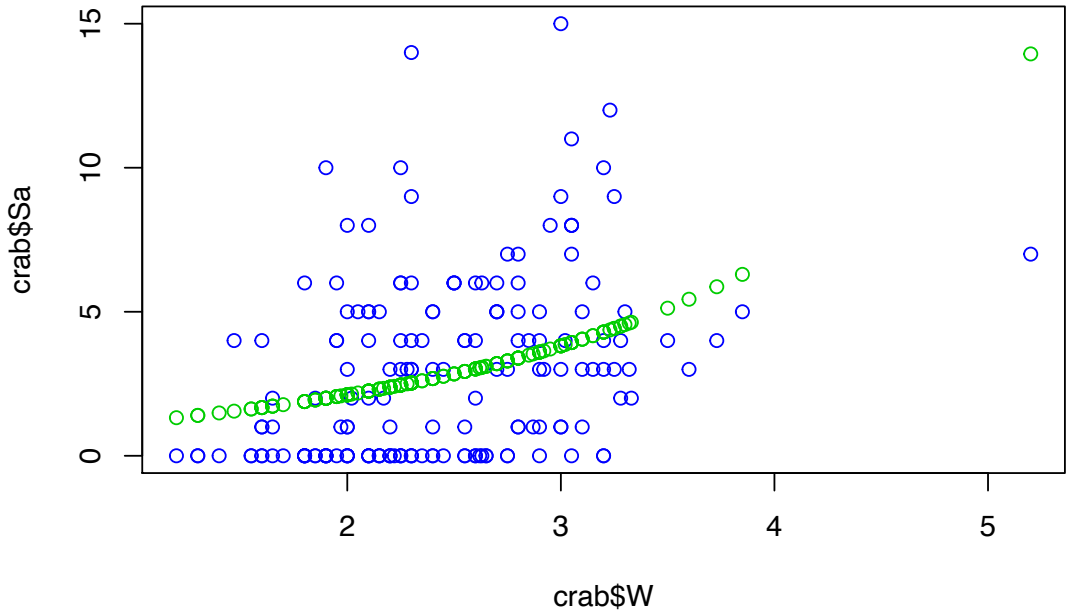
The estimated model is:
$\log(\lambda_i) = -0.43 + 0.59 \cdot W_i$
or
$\lambda_i = E(Sa_i) = \exp(-0.43) \cdot \exp(0.59 \cdot W_i)$

The slope is statistically significant.

Interpretation:
Since the estimate of $\beta_1 > 0$, the wider the female crab, the larger the expected number of male satellites on the multiplicative order as $exp(0.59) = 1.8$. More specifically, for one unit of increase in the width, the expected number of satellites (Sa) will increase and it will be multiplied by 1.8.

Visualize the Poisson regression model.

```
pred <- fitted(model.1)
plot(crab$W, crab$Sa, col = 4) # Scatter plot of Sa vs W
points(crab$W, pred, col = 3)
```



Is there a good fit?

The deviance is used as overall-goodness-of-fit statistic. This statistic has $n - p$ degrees of freedom (with n the number of observations and p the number of parameters in the model).

Example *horseshoe crabs*

```
anova(model.1)
```

```
## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Sa
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev
## NULL                   172     632.79
## W     1   71.949       171     560.84
```

In our example of the crab data, we have a residual deviance of 560 with 171 degrees of freedom (df). (with $n = 173$ and $p = 2$).

To compare model 1 with model 0 (the intercept-only model), we can also use the chi-square test for comparing the deviances.

```r
anova(model.0, model.1, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Sa ~ 1
## Model 2: Sa ~ W
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1       172     632.79
## 2       171     560.84  1   71.949 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model 1 is significant better than the intercept-only model (= model 0).

Remark:

1. The assumption says that, for a given set of explanatory values, the variance of the response is equal to its mean. Often, the variance is higher than that.
2. Note that the *Deviance* has an approximate χ^2-distribution with $n - p$ degrees of freedom, where n is the number of observations ($n = 173$ in the crab example) and p is the number of parameters ($p = 2$ in the crab example).
3. The expected value of a χ^2 random variable is equal to the degrees of freedom.

Hence, if our model fits the data well, the ratio of the *Deviance* to DF (the degrees of freedom), $\frac{Deviance}{DF}$, should be about one.

The ratio $\frac{deviance}{DF} = \frac{560}{171} = 3.2$ which is much larger than 1.

So if the residual deviance is larger than the residual degrees of freedom, this is an indication of overdispersion. Overdispersion means that observed variance is larger than the assumed variance. Sometimes this can be solved by including extra explanatory variables. Other solutions are running a negative binomial regression model or use of an adjustment for overdispersion.

If you have overdispersion (residual deviance is much larger than degrees of freedom), you may want to use `quasipoisson()` instead of `poisson()`.For a **quasi-Poisson** regression, the variance is assumed to be a linear function of the mean. Sometimes one also use the negative binomial model in those cases.

4. In case your data set has many zero counts, then one might consider a **zero-inflated Poisson regression model**. Then we assume there is one process which determines whether a female crab has satellites or not. There is another process which determines how many satellites a female crab has.

References

Univariate statistical analysis

- Moore D., McCabe G. and Graig B., Introduction to the practice of statistics, 6th edition, 2007, W.H. Freeman & Company

- Draghici S., Statistics and data analysis for microarrays using R and Bioconductor, 2nd edition, 2012, CRC Press

For regression and Anova

- Kutner M., Nachtsheim C., Neter J. and Li W. , Applied Linear Statistical Models, 5th edition, 2004, Irwin Professional Pub
- Hay-Jahans C, An R companion to linear statistical models, 2012, CRC Press

For logistic regression, generalized linear models:

- Agresti, Categorical Data Analysis, 2nd edition, 2002, John Wiley & Sons

Chapter 13 : Generalized linear model

Table of Contents

1. Introduction

The purpose of this chapter is to show that several seemingly unrelated models are actually all special cases of the generalized linear model.

The Generalized Linear Model can be written as:

$$g\big(E(Y_i)\big) = \alpha + \sum_{j=1}^{p} \beta_j X_{ji} + \epsilon_i$$

- Where g(E(Yi)) is some function of the expected value of Yi (this function g is called the link function).

- and $\varepsilon_i \sim F$ (i.e. the error term has some sort of distribution, e.g. normal in case of a linear regression model ,….) . This is called the distributional family.

2. Linear regression as a generalized linear model

In linear regression, we have following model formulation:

$$\big(E(Y_i)\big) = \alpha + \sum_{j=1}^{p} \beta_j X_{ji} + \epsilon_i$$

and $\varepsilon_i \sim N(0, \sigma^2)$.

- That is, the distributional "family" is normal.
- We predict E(Y). Hence , g(E(Y)) = E(Y). In this case G(.) is the *identity* link function.

3. Logistic regression as a generalized linear model

The *logistic regression model* can then be written as

$$ln\frac{P[Y_i = 1]}{1 - P[Y_i = 1]} = \alpha + \sum_{j=1}^{p} \beta_j X_{ji} + \epsilon_i$$

Note that

- When Y_i is a binary variable (can take the values 0 or 1), it does not have a normal distribution; rather it has a *binomial* distribution. The distributional family is binomial.
- The left hand side is not E(Y). The left hand side is expressed in log odds. We predict g(E(Y)), where g is the *logit* link function.

4. Poisson regression as a generalized linear model

The Poisson regression model is formulated as:

$$\log\big(E(Y_i)\big) = \alpha + \sum_{j=1}^{p} \beta_j X_{ji} + \varepsilon_i$$

Note that

- Here we assume a Poisson distribution for Y_i. The distributional family is Poisson.
- The left hand side is log(E(Y)), where g is the *log* link function.

5. How to use in R

We can use the **glm** function in R.

glm(formula, family = gaussian, data)

- The formula looks like

$$Y \sim X1 + X2$$

where X1 and X2 are the names of

- ✓ Continuous variables
- ✓ Categorical variables

- The family argument specifies
- ✓ the link function
- ✓ the variance function

E.g.

	Family argument
Linear Regression model	gaussian(link = "identity")
Logistic regression model	binomial(link = "logit")
Poisson regression model	poisson(link = "log")