ANOVA vs. linear regression: relationship and assumptions

Vera Demberg

Saarland University

WS 2021/2022

What we covered so far

We have seen simple linear regression and simple forms of ANOVAs.

They can actually do the same thing!

What we covered so far

We have seen simple linear regression and simple forms of ANOVAs.

They can actually do the same thing!

We will therefore look at two things today:

1. The relation between linear regression and ANOVA
2. checking assumptions / model checking

Table of Contents

1. ANOVA as a special type of regression
2. Checking assumptions
   - Homogeneity of variance
   - Normality of residuals
   - Uncorrelated predictors
3. Confidence Intervals for regression coefficients
4. Basics of experimental design

What we've seen so far:

- ANOVA as a more general type of t-test
- Linear regression as a more general type of correlation

What is the relationship between ANOVA and linear regression?

Reminder: linear regression

A linear regression model has the form:

$$Y_p = b_0 + b_1 X_{1p} + b_2 X_{2p} + \epsilon_p$$

can also write as:

$$\hat{Y}_p = b_0 + b_1 X_{1p} + b_2 X_{2p}$$

or the same in R:

```
lm(response ∼ predictor1 + predictor2 , data=someframe)
```

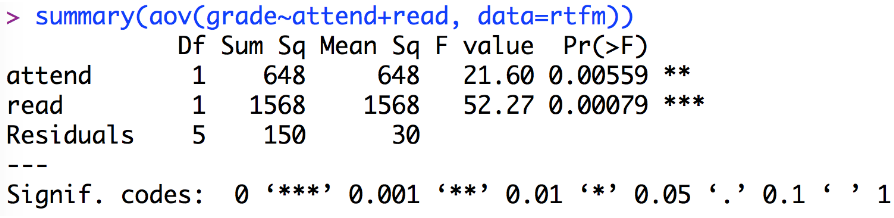
an Example

Let's take a look at a dataset:

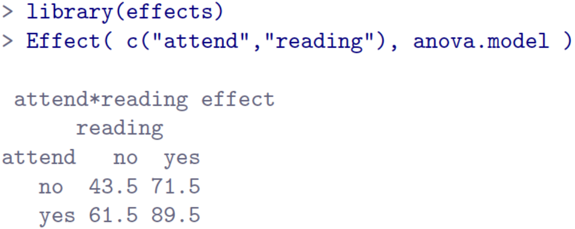
person, p	grade, Y_p	attendance, X_{1p}	reading, X_{2p}
1	90	1	1
2	87	1	1
3	75	0	1
4	60	1	0
5	35	0	0
6	50	0	0
7	65	1	0
8	70	0	1

What type of test to choose?

We have a ratio scale variable and two categorical predictors → ANOVA



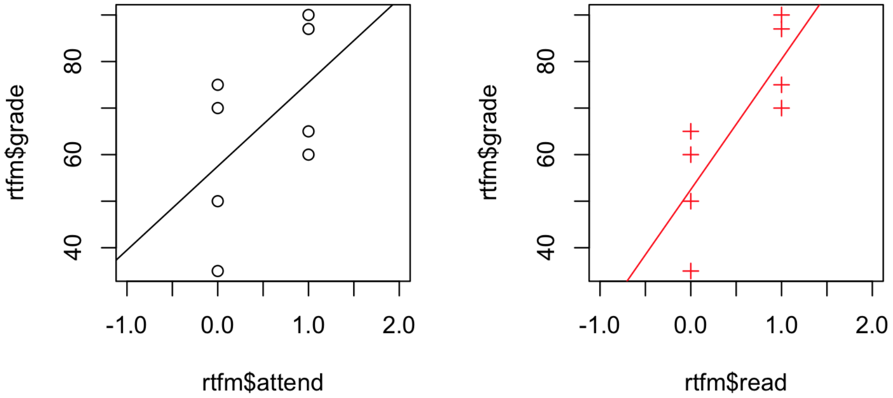
```
> summary(aov(grade~attend+read, data=rtfm))
            Df Sum Sq Mean Sq F value  Pr(>F)    
attend       1    648     648   21.60 0.00559 ** 
read         1   1568    1568   52.27 0.00079 ***
Residuals    5    150      30                    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

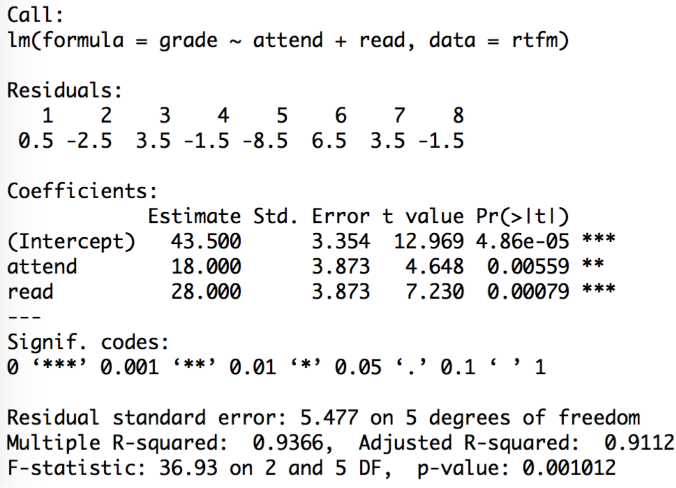


```
> library(effects)
> Effect( c("attend","reading"), anova.model )

 attend*reading effect
      reading
attend   no  yes
   no  43.5 71.5
   yes 61.5 89.5
```

Modelling it as linear regression

We don't violate assumptions when we treat this as regression.
"1" for attend means more attendance than "0";
"1" in reading means more reading than "0".



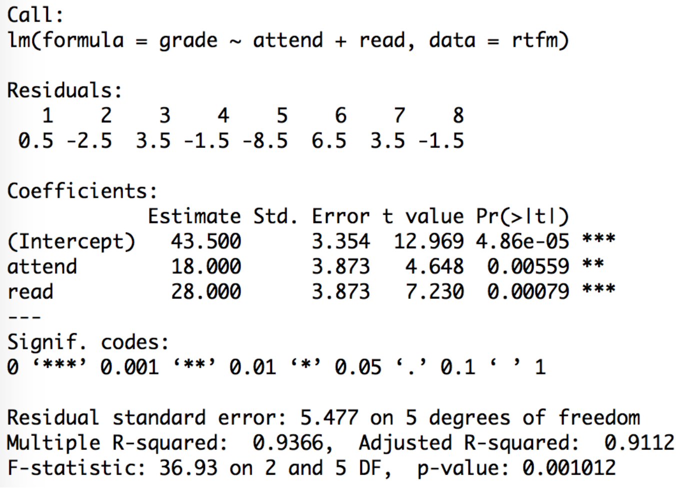


```
Call:
lm(formula = grade ~ attend + read, data = rtfm)

Residuals:
   1    2    3    4    5    6    7    8 
 0.5 -2.5  3.5 -1.5 -8.5  6.5  3.5 -1.5 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   43.500      3.354  12.969 4.86e-05 ***
attend        18.000      3.873   4.648  0.00559 ** 
read          28.000      3.873   7.230  0.00079 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.477 on 5 degrees of freedom
Multiple R-squared:  0.9366,	Adjusted R-squared:  0.9112 
F-statistic: 36.93 on 2 and 5 DF,  p-value: 0.001012
```

Things to note...

the t value is calculated as

$$t = \frac{\hat{b}}{SE(\hat{b})}$$



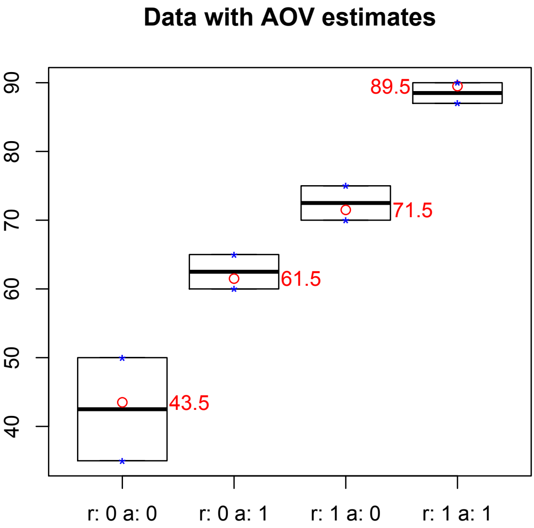
```
Call:
lm(formula = grade ~ attend + read, data = rtfm)

Residuals:
   1    2    3    4    5    6    7    8
 0.5 -2.5  3.5 -1.5 -8.5  6.5  3.5 -1.5

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   43.500      3.354  12.969 4.86e-05 ***
attend        18.000      3.873   4.648  0.00559 **
read          28.000      3.873   7.230  0.00079 ***
---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.477 on 5 degrees of freedom
Multiple R-squared:  0.9366,	Adjusted R-squared:  0.9112
F-statistic: 36.93 on 2 and 5 DF,  p-value: 0.001012
```

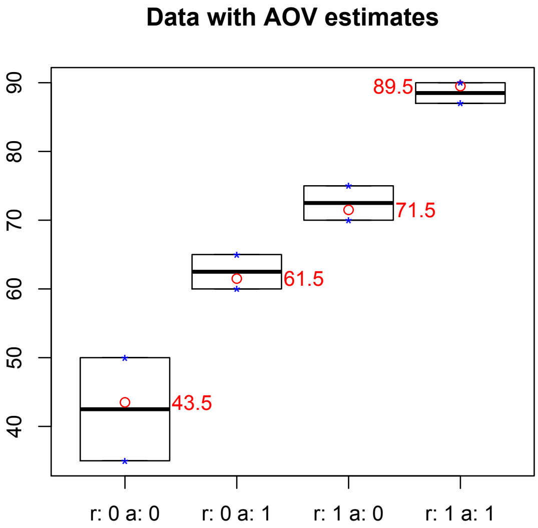
no reading, no attendance: 43.5

reading but no attendance: $43.5 + 18 = 61.5$

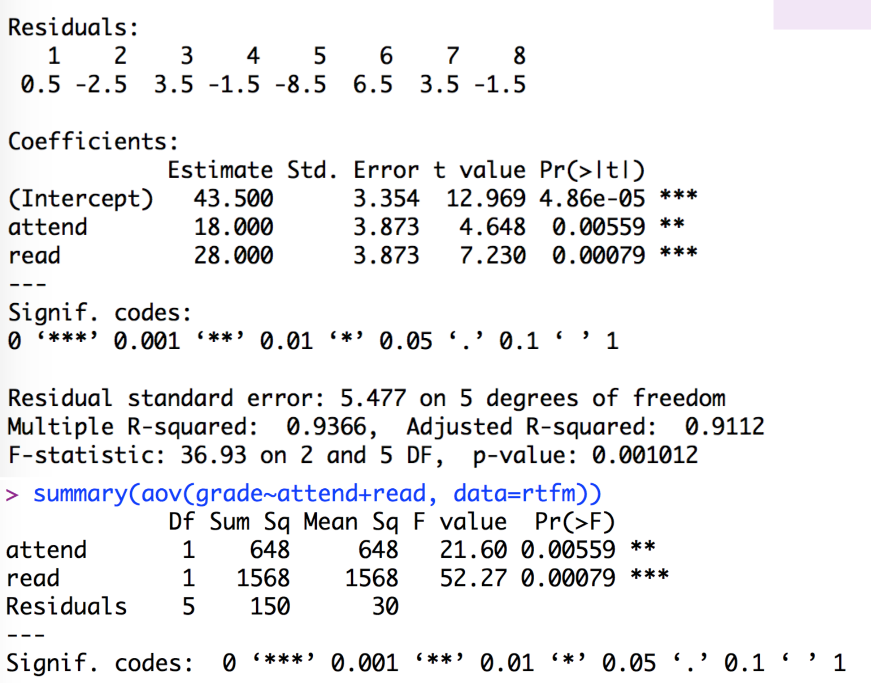
attendance but no reading: $43.5 + 28 = 71.5$

attendance and reading: $43.5 + 18 + 28 = 89.5$


Data with AOV estimates
90
80
70
60
50
40
89.5
71.5
61.5
43.5
r: 0 a: 0
r: 0 a: 1
r: 1 a: 0
r: 1 a: 1



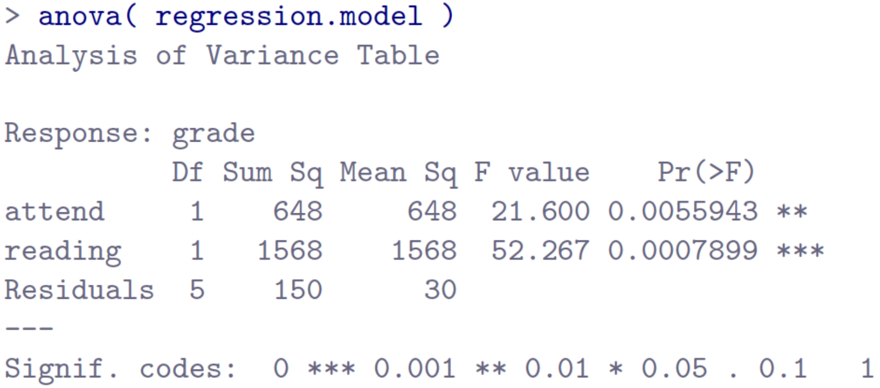
AOV effect estimates are exactly identical to the effects we got out of the linear regression model.



```
Residuals:
   1    2    3    4    5    6    7    8 
 0.5 -2.5  3.5 -1.5 -8.5  6.5  3.5 -1.5 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   43.500      3.354  12.969 4.86e-05 ***
attend        18.000      3.873   4.648  0.00559 ** 
read          28.000      3.873   7.230  0.00079 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.477 on 5 degrees of freedom
Multiple R-squared:  0.9366,	Adjusted R-squared:  0.9112 
F-statistic: 36.93 on 2 and 5 DF,  p-value: 0.001012

> summary(aov(grade~attend+read, data=rtfm))
            Df Sum Sq Mean Sq F value  Pr(>F)    
attend       1    648     648   21.60 0.00559 ** 
read         1   1568    1568   52.27 0.00079 ***
Residuals    5    150      30                    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

linear regression as anova

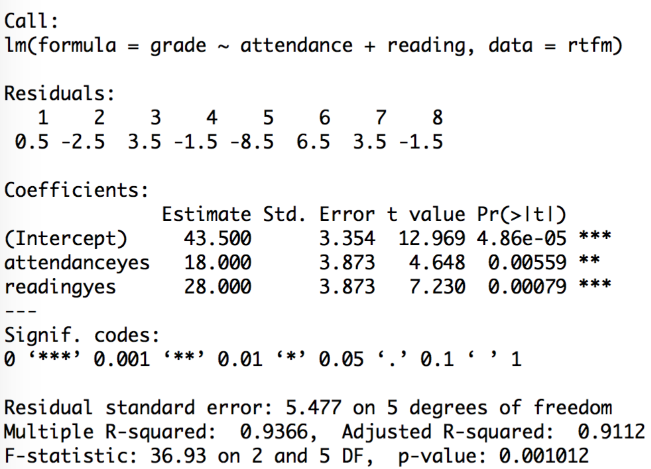
We can even ask R to give us the AOV output for a linear regression model:



```
> anova( regression.model )
Analysis of Variance Table

Response: grade
          Df Sum Sq Mean Sq F value    Pr(>F)
attend     1    648     648  21.600 0.0055943 **
reading    1   1568    1568  52.267 0.0007899 ***
Residuals  5    150      30
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

Dealing with categorical values

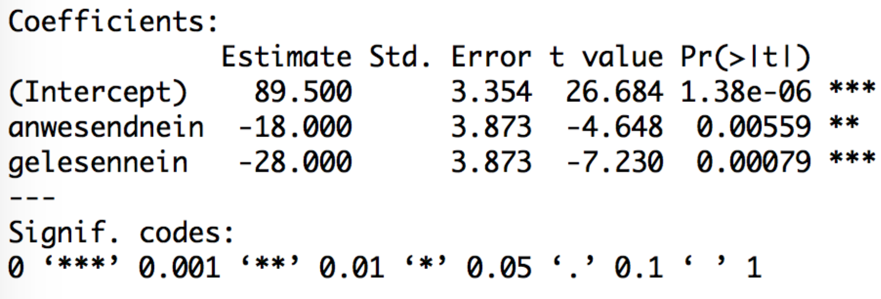
In our above example we encoded the categories for reading and attendance as 0 and 1 ourselves. What happens if we put in the original categorical variable (encoded as "yes" and "no")?



```
Call:
lm(formula = grade ~ attendance + reading, data = rtfm)

Residuals:
   1    2    3    4    5    6    7    8 
 0.5 -2.5  3.5 -1.5 -8.5  6.5  3.5 -1.5 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)     43.500      3.354  12.969 4.86e-05 ***
attendanceyes   18.000      3.873   4.648  0.00559 ** 
readingyes      28.000      3.873   7.230  0.00079 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.477 on 5 degrees of freedom
Multiple R-squared:  0.9366,	Adjusted R-squared:  0.9112 
F-statistic: 36.93 on 2 and 5 DF,  p-value: 0.001012
```

Default coding of factors in R

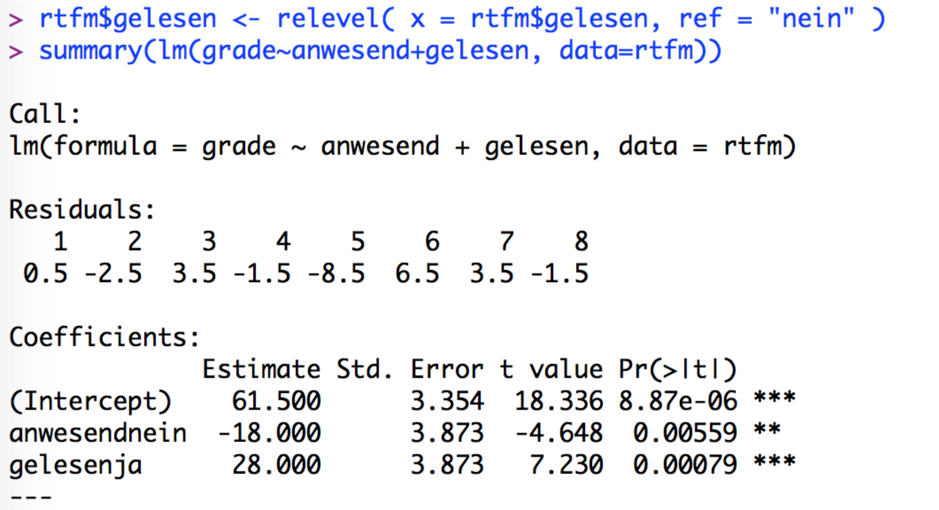
But how does R know to encode "yes" as 1 and "no" as 0?
actually, it doesn't. It just uses the one that comes first in the alphabet as a base category.
If we had encoded things as "ja" and "nein", we would have gotten the following instead:



```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   89.500      3.354  26.684 1.38e-06 ***
anwesendnein -18.000      3.873  -4.648  0.00559 ** 
gelesennein  -28.000      3.873  -7.230  0.00079 ***
---
Signif. codes:  
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

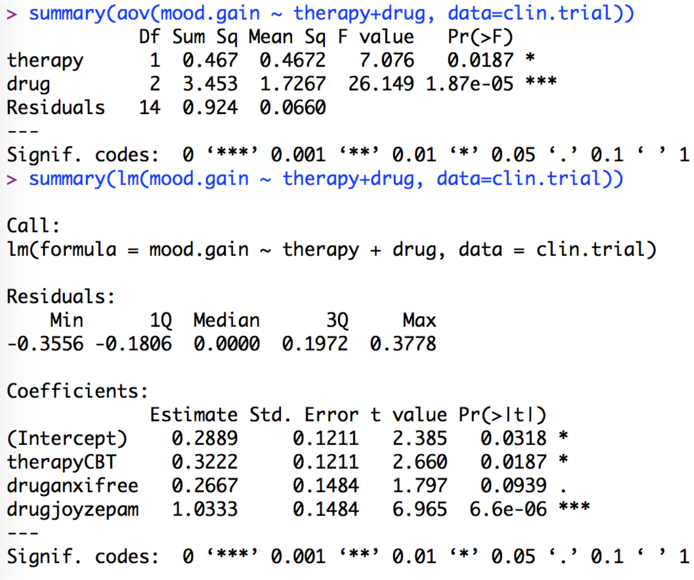
relevel

But we can ask R to switch this around for presentation / interpretation purposes.



```
> rtfm$gelesen <- relevel( x = rtfm$gelesen, ref = "nein" )
> summary(lm(grade~anwesend+gelesen, data=rtfm))

Call:
lm(formula = grade ~ anwesend + gelesen, data = rtfm)

Residuals:
   1    2    3    4    5    6    7    8 
 0.5 -2.5  3.5 -1.5 -8.5  6.5  3.5 -1.5 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   61.500      3.354  18.336 8.87e-06 ***
anwesendnein -18.000      3.873  -4.648  0.00559 ** 
gelesenja     28.000      3.873   7.230  0.00079 ***
---
```

What if we have more than 2 levels?



```
> summary(aov(mood.gain ~ therapy+drug, data=clin.trial))
            Df Sum Sq Mean Sq F value   Pr(>F)    
therapy      1  0.467  0.4672   7.076   0.0187 *  
drug         2  3.453  1.7267  26.149 1.87e-05 ***
Residuals   14  0.924  0.0660                     
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> summary(lm(mood.gain ~ therapy+drug, data=clin.trial))

Call:
lm(formula = mood.gain ~ therapy + drug, data = clin.trial)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.3556 -0.1806  0.0000  0.1972  0.3778 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept)    0.2889     0.1211   2.385   0.0318 *  
therapyCBT     0.3222     0.1211   2.660   0.0187 *  
druganxifree   0.2667     0.1484   1.797   0.0939 .  
drugjoyzepam   1.0333     0.1484   6.965  6.6e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

What if we have more than 2 levels?

drug	druganxifree	drugjoyzepam
"placebo"	0	0
"anxifree"	1	0
"joyzepam"	0	1

[1] (the `anova()` function allows to test for the difference between two models, we will talk about this in more detail later.)

What if we have more than 2 levels?

drug	druganxifree	drugjoyzepam
"placebo"	0	0
"anxifree"	1	0
"joyzepam"	0	1

The anova gives us an estimate of the effect of both drugs bundled together:[1]

```
> nodrug.regression <- lm( mood.gain ~ therapyCBT, clin.trial.2 )
> anova( nodrug.regression, drug.regression )
Analysis of Variance Table

Model 1: mood.gain ~ therapyCBT
Model 2: mood.gain ~ druganxifree + drugjoyzepam + therapyCBT
  Res.Df  RSS Df Sum of Sq    F  Pr(>F)
1     16 4.38
2     14 0.92  2      3.45 26.1 1.9e-05 ***
```

[1] (the anova() function allows to test for the difference between two models, we will talk about this in more detail later.)

Many different ways of coding a variable:
https://stats.idre.ucla.edu/r/library/r-library-contrast-coding-systems-for-categorical-variables/

Summary

Summary so far:

- Linear regression and ANOVA are doing the same thing under the hood. They only show different output.
- we talked about binary encoding of categorical effects in regression models
- we talked about interpreting the regression model output with categorical variables
- we saw how to relevel a categorical variable

Summary

Summary so far:

- Linear regression and ANOVA are doing the same thing under the hood. They only show different output.
- we talked about binary encoding of categorical effects in regression models
- we talked about interpreting the regression model output with categorical variables
- we saw how to relevel a categorical variable

Now, we'll proceed to our open topics in *assumption checking* for ANOVA and linear regression models.

Table of Contents

1. ANOVA as a special type of regression
2. Checking assumptions
   - Homogeneity of variance
   - Normality of residuals
   - Uncorrelated predictors
3. Confidence Intervals for regression coefficients
4. Basics of experimental design

Checking assumptions

We will now talk more about the *assumptions* underlying regression / ANOVA.

- Normality of residuals
  → QQ-plot or Shapiro-Wilk test
- Homogeneity of variance
- no bad outliers
  → plotting and calculation of leverage
- Linearity (linear regression with ratio / interval scale predictor)
  → transformation (e.g. logarithm, squaring etc.)
- Uncorrelated predictors
- Independence of residuals (all relevant predictors included)
- Independence of observations (no repeated measures)

Checking assumptions

We will now talk more about the *assumptions* underlying regression / ANOVA.

- Normality of residuals
  → QQ-plot or Shapiro-Wilk test
- Homogeneity of variance
- no bad outliers
  → plotting and calculation of leverage
- Linearity (linear regression with ratio / interval scale predictor)
  → transformation (e.g. logarithm, squaring etc.)
- Uncorrelated predictors
- Independence of residuals (all relevant predictors included)
- Independence of observations (no repeated measures)

We'll now go through those assumptions and how to check them.

Model checking for linear regression models

Linear regression models come with some standard plots that R generates for you, so you can check your model.

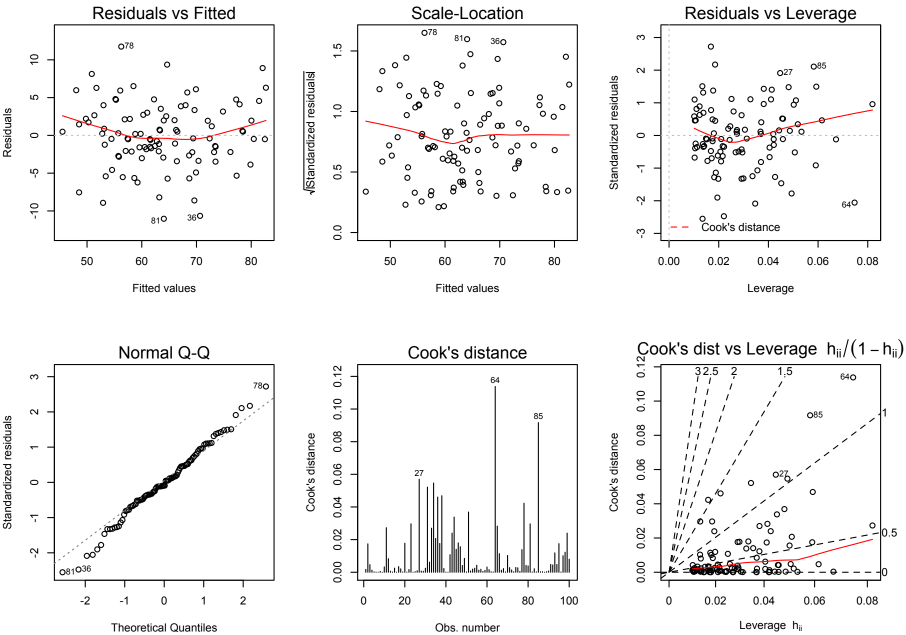
Model checking for linear regression models

Linear regression models come with some standard plots that R generates for you, so you can check your model.

The following slides will show the lm standard plots for the drugs model and the grumpiness model.

`par(mfcol=c(2,3))`
`plot(lm(mood.gain` $\sim \mathit{therapy} + \mathit{drug}, \mathit{data} = \mathit{clin.trial}), \mathit{which} = \mathit{seq}(1, 6))$

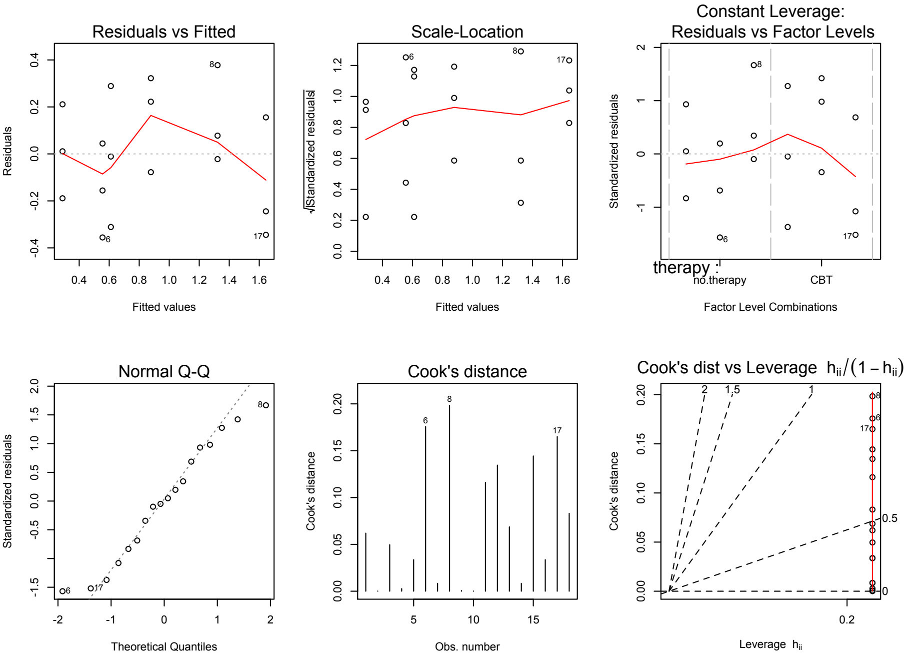
Checking assumptions



Measures: Leverage and Cook's distance

Leverage is a measure of how far away the independent variable values of an observation are from those of the other observations. High-leverage points are those observations, if any, made at extreme or outlying values of the independent variables such that the lack of neighboring observations means that the fitted regression model will pass close to that particular observation.

Cook's distance measures the effect of deleting a given observation. Points with a large Cook's distance are considered to merit closer examination in the analysis.


Residuals vs Fitted
Residuals
Fitted values
0.4
0.2
0.0
-0.2
-0.4
0.4
0.6
0.8
1.0
1.2
1.4
1.6
8
6
17
Scale-Location
√|Standardized residuals|
Fitted values
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Constant Leverage:
Residuals vs Factor Levels
Standardized residuals
2
1
0
-1
therapy :
no.therapy
CBT
Factor Level Combinations
Normal Q-Q
Standardized residuals
Theoretical Quantiles
2.0
1.5
1.0
0.5
0.0
-0.5
-1.5
-2
-1
0
1
2
Cook's distance
Cook's distance
Obs. number
0.20
0.15
0.10
0.05
0.00
5
10
15
Cook's dist vs Leverage $h_{ii}/(1-h_{ii})$
Cook's distance
Leverage h_{ii}
0.20
0.15
0.10
0.05
0.00
0.2
2
1.5
1
0.5
0

Homogeneity of variance

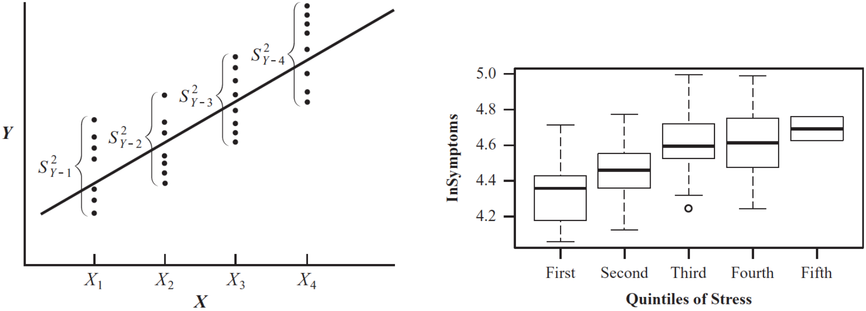


Figure 9.4 a) Scatter diagram illustrating regression assumptions; b) Similar plot for the data on Stress and Symptoms

Levene Test

So what does Levene's test actually do?

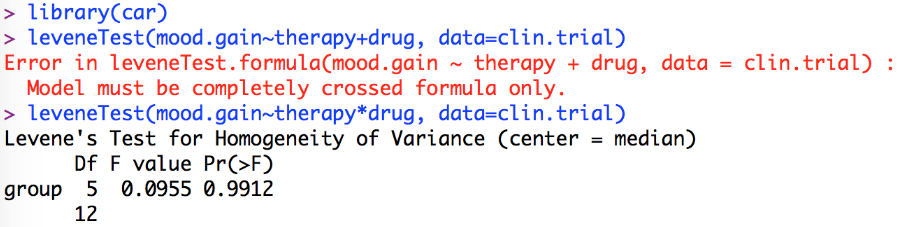
$$Z_{ik} = |Y_{ik} - \bar{Y}_k|$$

for each data point *i* in *k* groups.

Z_{ik} describes the absolute value of deviation for each point from the group mean. Now we're interested in whether these mean deviations for each group are the same or not.
→ we can do that! This means calculating an ANOVA over the Z values!

Homogeneity of variance: Levene test

The Levene test requires the model to include also all interactions.



```
> library(car)
> leveneTest(mood.gain~therapy+drug, data=clin.trial)
Error in leveneTest.formula(mood.gain ~ therapy + drug, data = clin.trial) :
  Model must be completely crossed formula only.
> leveneTest(mood.gain~therapy*drug, data=clin.trial)
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  5  0.0955 0.9912
      12
```

When homogeneity of variances is violated...

What if homogeneity of variances is violated?
We already saw this for the t-test: original Student's t test vs. Welch t test.
An equivalent solution exists for one way ANOVAs, i.e. when including a predictor with more than one level.

Tests for normality

We already talked about QQ plots and the Shapiro-Wilk normality test.

But what if normality assumptions are violated?

Tests for normality

We already talked about QQ plots and the Shapiro-Wilk normality test.

But what if normality assumptions are violated?

We already talked about this at various occasions as well:
We need to use a **nonparametric** test.

non-parametric ANOVA equivalent

For the t-test, the non-parametric alternative was the Wilcoxon test.

non-parametric ANOVA equivalent

For the t-test, the non-parametric alternative was the Wilcoxon test.

For the ANOVA (with a predictor including more than two levels in a categorical variable), we can use the **Kruskal-Wallis test**.

non-parametric ANOVA equivalent

The Kruskal-Wallis test is essentially an ANOVA calculated based on transforming the original data into ranks (and comparing the rank of each data point to the average rank for its group and the average group ranks to the overall average rank).

non-parametric ANOVA equivalent

The Kruskal-Wallis test is essentially an ANOVA calculated based on transforming the original data into ranks (and comparing the rank of each data point to the average rank for its group and the average group ranks to the overall average rank).

(and then there's some adjustment for ties).

non-parametric ANOVA equivalent

The Kruskal-Wallis test is essentially an ANOVA calculated based on transforming the original data into ranks (and comparing the rank of each data point to the average rank for its group and the average group ranks to the overall average rank).

(and then there's some adjustment for ties).

You won't be calculating this one by hand, so we'll just take a look at what you need to do in R.

Kruskal-Wallis test in R

```
> kruskal.test(mood.gain ~ drug, data = clin.trial)

        Kruskal-Wallis rank sum test

data:  mood.gain by drug
Kruskal-Wallis chi-squared = 12.076, df = 2, p-value = 0.002386
```

Checking assumptions

- ✓ Normality of residuals
  → QQ-plot or Shapiro-Wilk test
- ✓ Homogeneity of variance
- ✓ no bad outliers
  → plotting and calculation of leverage
- ✓ Linearity (linear regression with ratio / interval scale predictor)
  → transformation (e.g. logarithm, squaring etc.)

Checking assumptions

- ✓ Normality of residuals
  → QQ-plot or Shapiro-Wilk test
- ✓ Homogeneity of variance
- ✓ no bad outliers
  → plotting and calculation of leverage
- ✓ Linearity (linear regression with ratio / interval scale predictor)
  → transformation (e.g. logarithm, squaring etc.)
- Independence of residuals (all relevant predictors included)
- Uncorrelated predictors
- Independence of observations (no repeated measures)

Checking assumptions

- ✓ Normality of residuals
  → QQ-plot or Shapiro-Wilk test
- ✓ Homogeneity of variance
- ✓ no bad outliers
  → plotting and calculation of leverage
- ✓ Linearity (linear regression with ratio / interval scale predictor)
  → transformation (e.g. logarithm, squaring etc.)
- Independence of residuals (all relevant predictors included)
- Uncorrelated predictors
- Independence of observations (no repeated measures)

We'll now go through those assumptions and how to check them.

Assumption of uncorrelated predictors

In order to understand this more properly, it's useful to first talk about how to estimate the standard error for a coefficient in a linear regression model.

$$Y = b_0 + b_1 X_1 + b_2 X_2 + \epsilon$$

The standard error of a regression coefficient $SE(\hat{b})$ is estimated as follows:

$$SE(\hat{b}_j) = \sqrt{\widehat{var}(\hat{b}_j)}$$

$$\widehat{var}(\hat{b}_j) = \frac{s^2}{(n-1)} \times \frac{1}{\widehat{var}(X_j)} \times \frac{1}{1 - R_j^2}$$

Standard error of the coefficient estimate

$$Y = b_0 + b_1 X_1 + b_2 X_2 + \epsilon \qquad SE(\hat{b}_j) = \sqrt{\widehat{var}(\hat{b}_j)}$$

$$\widehat{var}(\hat{b}_j) = \frac{s^2}{(n-1)} \times \frac{1}{\widehat{var}(X_j)} \times \frac{1}{1 - R_j^2}$$

The standard error of a coefficient $\hat{b}_j$ depends on:

- s^2: greater scatter in the data around the regression surface leads to proportionately more variance in the coefficient estimates
- n: greater sample size results in proportionately less variance in the coefficient estimates
- $\widehat{var}(X_j)$: greater variability in a particular covariate leads to proportionately less variance in the corresponding coefficient estimate
- $\frac{1}{1-R_j^2}$: variance inflation factor, reflecting all other factors that influence the uncertainty in the coefficient estimates.

Variance Inflation Factor

The *variance inflation factor* measures how much of the variance of one predictor can be explained by the other predictors in the model.

$$VIF_k = \frac{1}{1 - R^2_{(-k)}}$$

$R^2_{(-k)}$ refers to R-squared value you would get if you ran a regression using X_k as the outcome variable, and all the other X variables as the predictors.

Variance Inflation Factor

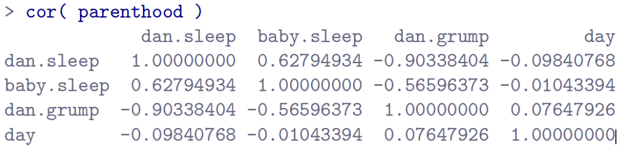
The *variance inflation factor* measures how much of the variance of one predictor can be explained by the other predictors in the model.

$$VIF_k = \frac{1}{1 - R^2_{(-k)}}$$

$R^2_{(-k)}$ refers to R-squared value you would get if you ran a regression using X_k as the outcome variable, and all the other X variables as the predictors.

The square root of the VIF tells you how much wider the confidence interval for the corresponding coefficient b_k is, relative to what you would have expected if the predictors are all nice and uncorrelated with one another.

VIF in R

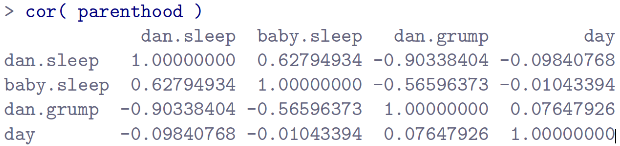


```
> cor( parenthood )
             dan.sleep  baby.sleep   dan.grump         day
dan.sleep   1.00000000  0.62794934 -0.90338404 -0.09840768
baby.sleep  0.62794934  1.00000000 -0.56596373 -0.01043394
dan.grump  -0.90338404 -0.56596373  1.00000000  0.07647926
day        -0.09840768 -0.01043394  0.07647926  1.00000000
```

We can use the function `vif()` from the `car` package.

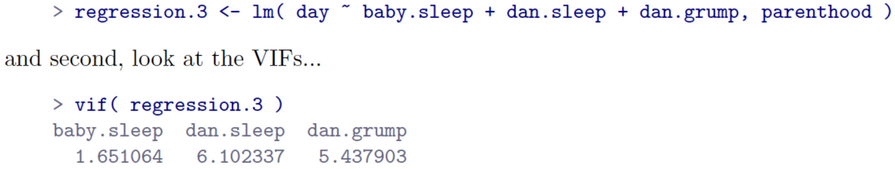
The VIF for a predictor that's uncorrelated with other predictors is 1.

VIF in R



```
> cor( parenthood )
             dan.sleep  baby.sleep   dan.grump         day
dan.sleep   1.00000000  0.62794934 -0.90338404 -0.09840768
baby.sleep  0.62794934  1.00000000 -0.56596373 -0.01043394
dan.grump  -0.90338404 -0.56596373  1.00000000  0.07647926
day        -0.09840768 -0.01043394  0.07647926  1.00000000
```

We can use the function `vif()` from the `car` package.
The VIF for a predictor that's uncorrelated with other predictors is 1.



```
> regression.3 <- lm( day ~ baby.sleep + dan.sleep + dan.grump, parenthood )
```

and second, look at the VIFs...

```
> vif( regression.3 )
baby.sleep  dan.sleep  dan.grump 
  1.651064   6.102337   5.437903 
```

The correlation between the predictors is causing a lot of uncertainty wrt. the coefficients in our model.

Table of Contents

1 ANOVA as a special type of regression

2 Checking assumptions
- Homogeneity of variance
- Normality of residuals
- Uncorrelated predictors

3 Confidence Intervals for regression coefficients

4 Basics of experimental design

Now that we're at it... CIs for regression coefficients

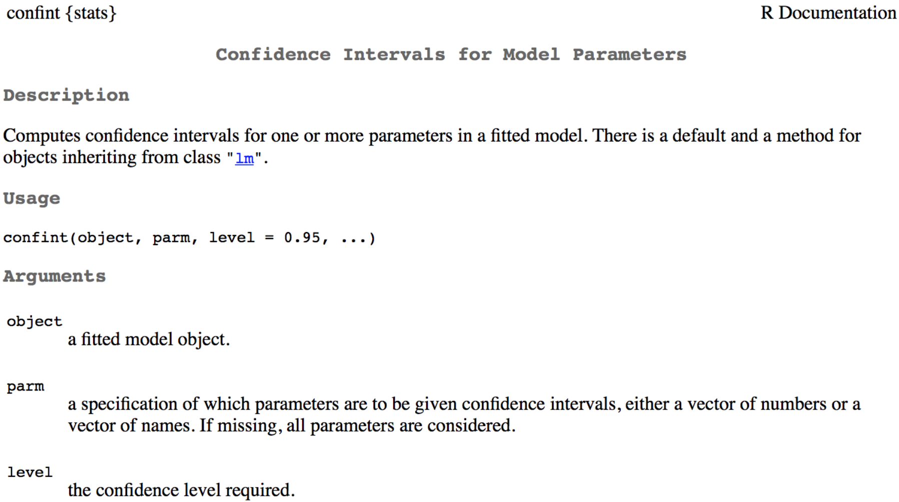
In order to understand this more properly, it's useful to first talk about how to estimate the standard error for a coefficient in a linear regression model.

$$Y = b_0 + b_1X_1 + b_2X_2 + \epsilon$$

$$CI(b) = \hat{b} \pm (t_{crit} \times SE(\hat{b}))$$

t_{crit}: critical value of the appropriate t distribution
$SE(\hat{b})$: standard error of regression coefficient

confint() in R



confint {stats} R Documentation

Confidence Intervals for Model Parameters

Description

Computes confidence intervals for one or more parameters in a fitted model. There is a default and a method for objects inheriting from class "lm".

Usage

```
confint(object, parm, level = 0.95, ...)
```

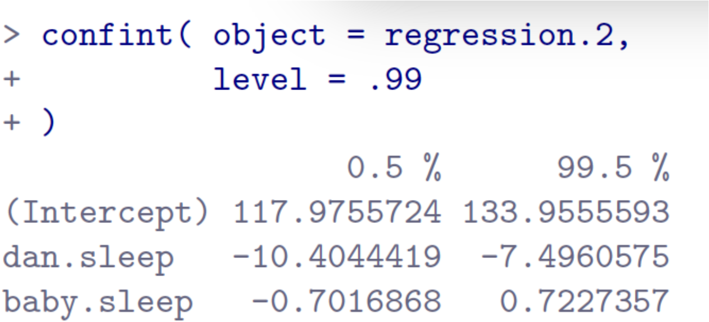
Arguments

`object`
a fitted model object.

`parm`
a specification of which parameters are to be given confidence intervals, either a vector of numbers or a vector of names. If missing, all parameters are considered.

`level`
the confidence level required.

confint() in R



```
> confint( object = regression.2,
+          level = .99
+ )
                  0.5 %      99.5 %
(Intercept) 117.9755724 133.9555593
dan.sleep   -10.4044419  -7.4960575
baby.sleep   -0.7016868   0.7227357
```

Checking assumptions

- ✓ Normality of residuals
  → QQ-plot or Shapiro-Wilk test
- ✓ Homogeneity of variance
- ✓ no bad outliers
  → plotting and calculation of leverage
- ✓ Linearity (linear regression with ratio / interval scale predictor)
  → transformation (e.g. logarithm, squaring etc.)

Checking assumptions

- ✓ Normality of residuals
  → QQ-plot or Shapiro-Wilk test
- ✓ Homogeneity of variance
- ✓ no bad outliers
  → plotting and calculation of leverage
- ✓ Linearity (linear regression with ratio / interval scale predictor)
  → transformation (e.g. logarithm, squaring etc.)
- ✓ Independence of residuals (all relevant predictors included)
- ✓ Uncorrelated predictors
- Independence of observations (no repeated measures)

Table of Contents

1. ANOVA as a special type of regression

2. Checking assumptions
   - Homogeneity of variance
   - Normality of residuals
   - Uncorrelated predictors

3. Confidence Intervals for regression coefficients

4. Basics of experimental design

Variables

Dependent variable (D.V.) or Response Variable

Any variable which is not directly manipulated by the experimenter; or, by extension, the outcome variable, that is measured during the experiment to study how it varies given (more or less controlled) variations in the I.V.

E.g.

- percentage of patients that recovers, maximum speed in km/h, the length of a word, reading times for object relative clauses.

NOTE: in this course, we'll assume there is always only one D.V. → Univariate Statistics

Variables (2)

In most experiments, we distinguish between two types of variables:

Independent variable (I.V.) or Predictor Variable

Any variable which is manipulated by the experimenter; or, by extension, any predictor variable, that is a variable of which we are evaluating the effects on another (the dependent) variable (the latter covers observational studies, like corpus studies, in addition to experimental work).

E.g.

- dose of a new drug, diameter in cm of a new type of bike wheels, the frequency of a word, the working memory span of participants, etc.

Factors

- Variables are often called factors when:
  1. they are I.V.'s **AND**
  2. they are categorical or nominal
- the latter means that the variable can only assume a limited range of values
- each of these values is called a **level** of the factor

Factors – example

- say our I.V. is age of our participants
- we can treat it as a continuous variable, ranging, say from 18 to 90
- but we can also turn it into a factor, if we are interested in the differences between age groups (e.g., 18-49, 50-64, 65-90)
    - Level 1 = { 18 - 49 }
    - Level 2 = { 50 - 64 }
    - Level 3 = { 65 - 90 }

An example dataset

Let's assume we have developed a new machine learning algorithm, and we want to find out about its performance. Specifically, we're interested in accuracy and in speed.

We have our own algorithm, a competitor algorithm and a baseline.

- what data would you collect?
- name a null-hypothesis and a H1.
- what would be the independent variable(s), and what would be the dependent variable?
- when considering the results for each of the benchmarks, is it important to keep in mind on which benchmark exactly

Designs

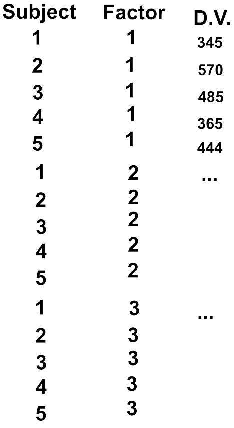
Related design

Also known as *within* (participants or items). All levels of your factor have been tested on all your particpants (or items)

Unrelated design

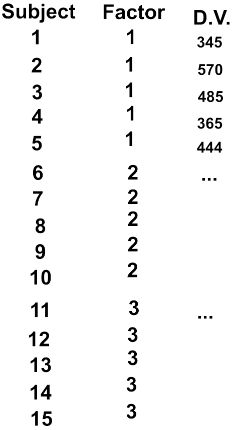
Also known as *between* (participants or items). Each participant (or item) is tested only on one level of your factor

Designs - in pics



Subject	Factor	D.V.
1	1	345
2	1	570
3	1	485
4	1	365
5	1	444
1	2	...
2	2	
3	2	
4	2	
5	2	
1	3	...
2	3	
3	3	
4	3	
5	3	

Figure: Related or within-subject design



Subject	Factor	D.V.
1	1	345
2	1	570
3	1	485
4	1	365
5	1	444
6	2	...
7	2	
8	2	
9	2	
10	2	
11	3	...
12	3	
13	3	
14	3	
15	3	

Figure: Unrelated or between-subject design

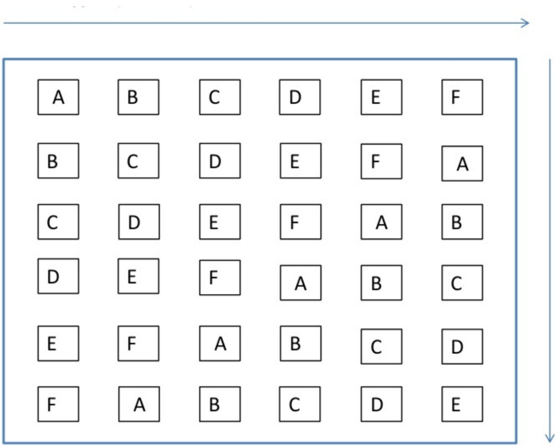
Designs: pros and cons

- related designs usually have more power (what this means will be explained shortly)
- related designs require extra care in the counterbalacing to avoid order effects

ok, so what about our ML alg example?

Is it a within or a between design? Why?

Latin square design



A	B	C	D	E	F
B	C	D	E	F	A
C	D	E	F	A	B
D	E	F	A	B	C
E	F	A	B	C	D
F	A	B	C	D	E