Regression Modelling (Baayen)

```
lm1 = lm( var1 ~ var2 )
```

var1 ↑ dependent variable, to be modelled

var2 ↑ predictor

lm - linear model

Slope and intercept are estimated using Least Square Regression line

↓

by minimizing the squared vertical distance between the data points and the line

coef(lm1) - returns the model's coefficients

abline(lm1) - will draw the line

Correlation - degree to which the data points cluster around the regression line

The degree of correlation - correlation coefficient

ρ - correlation coeff for a population
r - correlation coeff for a sample

$-1 \leq p \leq 1$

$-1 \leq r \leq 1$

↑ perfect positive correlation

↓ perfect negative correlation

this measure ~~can be useful~~ can be used to think is how useful is to fit a straight line to the data

r^2 (or R^2) - measure for evaluating how much of the scatter is accounted for

or R^2 quantifies the proportion of the variance in the data that is captured and explained by the regression model

When we try to fit a model to a data set, the goal is to be able to predict values of the dependent variables

⇓

The better the prediction the higher it. And ~~the~~ with lousy ~~predictors~~ predictors, little variance is explained by the model

↳ in this situation R^2 is close to 0.

summary(lm1) returns a summary of the lm

- residuals
- coefficients
  - intercept, slope
  - each coeff. comes with 3 other numbers:
    - standart error
    - t-value
    - p-value

p value tells us if the coefficient is significantly different from zero (~~and hence, potentially useful~~)

if coefficient is zero, there are no relations at all between the predictor and the dependant variable

t-test is used to ascertain if p-coeff is significantly different from zero, and, hence, potentially useful.

t-value = p-coeff / st. error

st. error - measure on how we're sure about the estimate of the coefficient.

the smaller the standart error, the smaller is the confidence interval around the estimate ⇒

```
names(summary(lm1)) ← names
summary(lm1)$coef
summary(lm1)$coef[,3]
```

Residual standart error - measure of how unsuccessful the model is

the better model, the smaller its residual standart error

multiple R-squared - r^2, squared correlation coefficient

We get r by square root of R^2

another way of calculating r:

```
cor(a, b)
cor.test(a, b)
```

↓

Also tests if it's significantly differs from 0.
and lists 95% confidence intervals

F-value - a test if the linear model as a whole succeeds in explaining a significant portion of the variance.

we also can describe quadratic terms with lm

lm 1 ~ lm(a ~ b + I(c ^ 2))

↑

linear means that the relationship can be expressed as sum (linear combination)

Basic rules:

- visualize
- beware of outliers
- fit straight lines aren't always the best
- keep it simple

For factors.

lm(x ~ some. factor)

↓

factor, not numerical value

↓

gets converted into numerical vectors (one or more)

let's assume s.f has 2 levels

⇓

so it's converted into one numerical vector with 1 and 0.

However, intercept and slope need special interpretation

Dummy Coding - way to deal with factor variables

one level is signed out as the Default or Reference Level, which is contrasted to others

in this case intercept would ~~mean~~ represent the group mean for the default level

Class = c("animal", "plant")
↗
default level

so intercept would be mean for animals

the second value (Classplant) represents the contrast (i.e. difference) between the group mean of the plants and of the animals

In other words, mean of plants level is [(intercept) + classplant]

t-value tells that the adjustment is statistically significant
(i.e. the means differ significantly)

lm can be applied to factors with more than 2 levels

anova - analysis of variance

to explain some of the variation within groups

↓

reports F value
which reports that there are (not) significant differences in the means

but it doesn't tell us what differences are involved
(need to run "summary")

↓

it will show all variables except the default one, so the difference is with this variable

(this default row is (Intercept))

Tukey's Honestly Significant Difference

TukeyHSD in R

for multiple comparisons
for detecting significant differences

aov(breaks ~ tension, data = warpbreaks)

special for analysis of variance

(output is exactly the same as if we applied anova to em)

tc = TukeyHSD(warpbreaks.aov)

	diff	lwr	upr	p adj
M-L	-10	-19,55	-0,44	0.03
H-L	-14,72	-24.29	-5.16	0.00
H-M	-4.72	-14,29	-4.83	0.46

this table lists the differences in the means

the lower and the upper end points of the confidence intervals

adjusted p-value

plot(tc) will draw these differences

(those that intersect - - - dotted line - are not significantly different)

linear regression — numerical predictor

analysis of variance — factor as predictor

analysis of covariance — with both numeric and factors

In R, lm used for all these analyses: regression, variance, covariance

all are built on the same fundamental principles

lm1 = lm(a ~ b * c)

(c ↑ factor)

lets consider the following model

lm1 = lm(meanSizeRating ~ meanFamiliarity * Class + I(meanFamiliarity^2))

(Class ↑ factor; meanFamiliarity ↑ numeric)

Coefficients	Est
(Intercept)	4,43
meanFamiliarity	-0,63
I(meanFamiliarity^2)	0,11
Classplant	-1,01
meanFamiliarity : Classplant	-0.21

summarizes how the preceding coefficients should be modified in order to make them more precise for the nouns that fall into „plant" category

(meanSizeRating) (Classfactor)

The coefficient „Classplant" tells us that we should substract ~~-0,21~~ -1,01 from the intercept in order to obtain the (modified) mean for the plants

The final coefficient „meanFamiliarity:Classplant" tells us that the coefficient „meanFamiliarity" should be decreased by -0,21 in order to make it precise for plants

The last coefficient - interaction between meanFamiliarity and Class

meanFamiliarity * Class

q

interaction

⇓ the same is

meanFamiliarity + Class +
meanFamiliarity:Class

meanFamiliarity : Class

↑

interaction of the predictor to its left and right

What the interaction tells us is that the linear coefficient of meanFamiliarity has to be adjusted downwards when dealing with plants rather than with animals

↑

(categories in factor)

- For animals the coefficient is −0.63
- For plants, we add the coef. for the interaction of meanFamiliarity by Class

to this coef: −0,63−0,212−0,84

↑

i.e the linear term of meanFamil. differs for plants and animals