Kurs Bio144: Datenanalyse in der Biologie

Lecture 7: ANCOVA, short introduction to Linear Algebra

Stefanie Muff (Lecture) & Owen L.Petchey (Practical)

University of Zurich

31 December, 2020

Overview

- ANCOVA
- Introduction to linear Algebra

Note: ANCOVA = ANalysi of COVAriance (Kovarianzanalyse)

Course material covered today

- "Getting Started with R" chapter 6.3
- "Lineare regression" chapters 3.A (p. 43-45) and 3.4, 3.5 (p. 39-42)

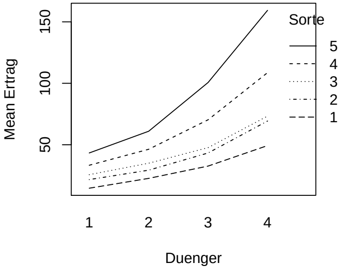
Recap of ANOVA

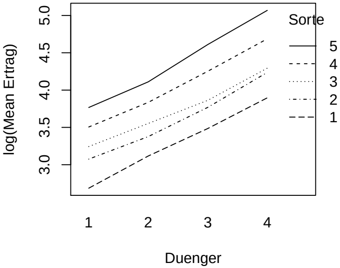
- ANOVA is a method to test if the means of two or more groups are different.
- Post-hoc tests and contrasts, including correction for p-values, to understand the differences between the groups.
- Two-way ANOVA for factorial designs, interactions.
- ANOVA is a special case of linear regression with categorical covariates.

Recap of two-way ANOVA example

Remember: Influence of four levels of fertilizer (DUENGER) on the yield (ERTRAG) on 5 species (SORTE) of crops was investigated. For each DUENGER $\times$ ERTRAG combination, 3 repeats were taken.

Interaction plot with ERTRAG and log(ERTRAG) as response:





Remember: We used log(ERTRAG), because the residual plots were otherwise not ok.

```r
r.duenger2 <- lm(log(ERTRAG) ~ DUENGER*SORTE,d.duenger)
anova(r.duenger2)
```

```
## Analysis of Variance Table
##
## Response: log(ERTRAG)
##               Df  Sum Sq Mean Sq  F value Pr(>F)
## DUENGER        3 11.6917  3.8972 854.0505 <2e-16 ***
## SORTE          4  8.5202  2.1300 466.7851 <2e-16 ***
## DUENGER:SORTE 12  0.0929  0.0077   1.6958 0.1045
## Residuals     40  0.1825  0.0046
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Questions:

- Number of parameters?
- Degrees of freedom (60 data points)?
- Interpretation?

```
summary(r.duenger2)
```

```
##
## Call:
## lm(formula = log(ERTRAG) ~ DUENGER * SORTE, data = d.duenger)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.120968 -0.045595  0.008984  0.049072  0.102175
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      2.68505    0.03900  68.846  < 2e-16 ***
## DUENGER2         0.43165    0.05516   7.826 1.36e-09 ***
## DUENGER3         0.79997    0.05516  14.504  < 2e-16 ***
## DUENGER4         1.21152    0.05516  21.966  < 2e-16 ***
## SORTE2           0.38979    0.05516   7.067 1.51e-08 ***
## SORTE3           0.55799    0.05516  10.117 1.38e-12 ***
## SORTE4           0.82018    0.05516  14.870  < 2e-16 ***
## SORTE5           1.08169    0.05516  19.612  < 2e-16 ***
## DUENGER2:SORTE2 -0.12949    0.07800  -1.660    0.105
## DUENGER3:SORTE2 -0.10613    0.07800  -1.361    0.181
## DUENGER4:SORTE2 -0.04924    0.07800  -0.631    0.531
## DUENGER2:SORTE3 -0.12180    0.07800  -1.562    0.126
## DUENGER3:SORTE3 -0.18034    0.07800  -2.312    0.026 *
## DUENGER4:SORTE3 -0.16061    0.07800  -2.059    0.046 *
## DUENGER2:SORTE4 -0.10138    0.07800  -1.300    0.201
## DUENGER3:SORTE4 -0.05311    0.07800  -0.681    0.500
## DUENGER4:SORTE4 -0.02954    0.07800  -0.379    0.707
## DUENGER2:SORTE5 -0.08779    0.07800  -1.125    0.267
## DUENGER3:SORTE5  0.04370    0.07800   0.560    0.578
## DUENGER4:SORTE5  0.09014    0.07800   1.156    0.255
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06755 on 40 degrees of freedom
```

Analysis of Covariance (ANCOVA)

An ANCOVA is an analysis of variance (ANOVA), including also at least one continuous covariate.

ANCOVA unifies several concepts that we approached in this course so far:

- Linear regression
- Categorical covariates
- Interactions (of continuous and categorical covariates)
- Analysis of Variance (ANOVA)

As such, it is a **special case of the linear regression model**.

Given a categorical covariate x_i and a continuous covariate z_i. Then the ANCOVA equation (without interactions) is given as

$$y_i = \beta_0 + \beta_1 x_i^{(1)} + ... + \beta_k x_i^{(k)} + \beta_z z_i + \epsilon_i \ ,$$

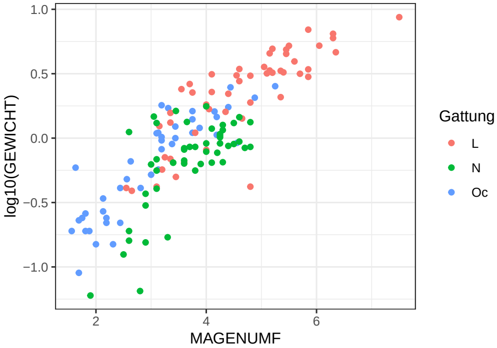
where $x_i^{(k)}$ is the kth dummy variable ($x_i^{(k)}$=1 if ith observation belongs to category k, 0 otherwise).

Note 1: It is straightforward to add an interaction of x_i with z_i.

Note 2: Again, for identifiability reason, we typically set $\beta_1 = 0$.

Once more: the earthworms

"Magenumfang" was used to predict "Gewicht" of the worm, including as covariate also the worm species.



Categorical and continuous covariates were used to predict a continuous outcome → ANCOVA.

```r
r.lm <- lm(log(GEWICHT) ~  MAGENUMF + Gattung,d.wurm)
summary(r.lm)$coef
```

```
##                Estimate Std. Error      t value     Pr(>|t|)
## (Intercept) -2.5355459 0.22147279 -11.4485663 8.617670e-22
## MAGENUMF     0.7118725 0.04528843  15.7186392 1.232126e-32
## GattungN    -0.5151344 0.11009219  -4.6791186 6.760621e-06
## GattungOc   -0.0907298 0.12791000  -0.7093254 4.793107e-01
```

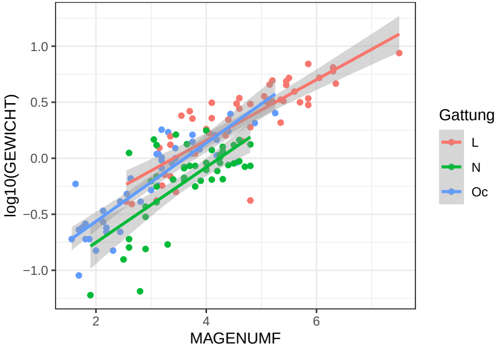
Important: The *p*-values for the entries `GattungN` and `GattungOc` are not very meaningful (why?).

To understand if “Gattung” has an effect, **we need to carry out an F-test** $\rightarrow$ ANOVA table:

```r
anova(r.lm)
```

```
## Analysis of Variance Table
##
## Response: log(GEWICHT)
##            Df  Sum Sq Mean Sq F value    Pr(>F)
## MAGENUMF    1 104.866 104.866  409.69 < 2.2e-16 ***
## Gattung     2   7.177   3.589   14.02 2.842e-06 ***
## Residuals 139  35.579   0.256
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We also included an **interaction term** between `MAGENUMF` and `Gattung` to allow for different slopes:



→ We again need the F-**test** to check whether the respective interaction term is needed:

```
r.lm2<- lm(log(GEWICHT) ~  MAGENUMF * Gattung,d.wurm)
anova(r.lm2)
```

```
## Analysis of Variance Table
##
## Response: log(GEWICHT)
##                   Df  Sum Sq Mean Sq  F value    Pr(>F)
## MAGENUMF           1 104.866 104.866 414.4743 < 2.2e-16 ***
## Gattung            2   7.177   3.589  14.1835 2.521e-06 ***
## MAGENUMF:Gattung   2   0.917   0.458   1.8112    0.1673
## Residuals        137  34.662   0.253
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

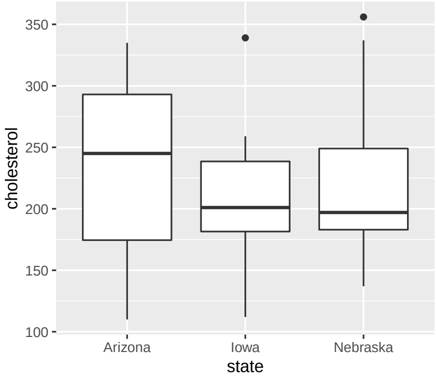
→ $p = 0.167$, thus interaction is probably not relevant.

A new example: cholesterol levels

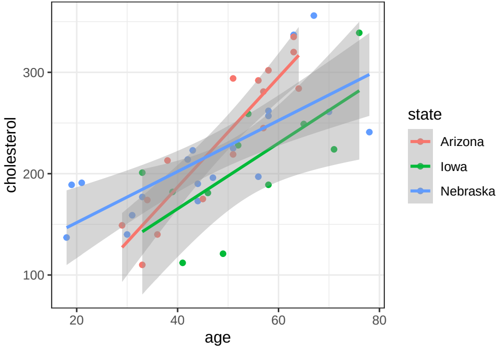
Example: Cholesterol levels [mg/ml] for 45 women from three US states (Iowa, Nebraska, Arizona), were measured.

Question: Do these levels differ between the states?

Age (years) may be a relevant covariable.



The scatter plot gives an idea about the model that might be useful here:



→ We include state, age and the interaction of the two.

Doing the analysis:

```r
r.lm <- lm(cholesterol ~ age*state,data=d.chol)
anova(r.lm)
```

```
## Analysis of Variance Table
##
## Response: cholesterol
##            Df Sum Sq Mean Sq F value    Pr(>F)
## age         1  96524   96524 61.8961 1.424e-09 ***
## state       2  11474    5737  3.6789   0.03438 *
## age:state   2  12665    6332  4.0606   0.02501 *
## Residuals 39  60819    1559
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Interpretation?

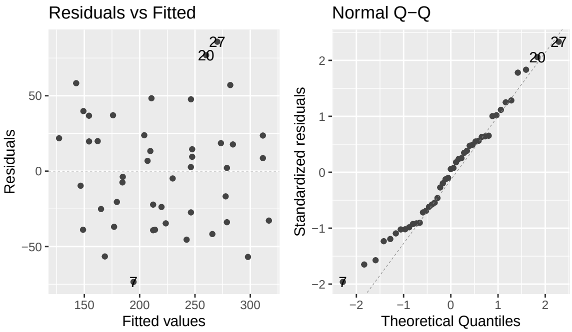
Compare the results from the previous slide to the estimated coefficients:

```
r.lm <- lm(cholesterol ~ age*state,data=d.chol)
summary(r.lm)$coef
```

```
##                     Estimate Std. Error    t value     Pr(>|t|)
## (Intercept)       -29.895169 43.7353712 -0.6835467 4.983027e-01
## age                 5.416908  0.8679635  6.2409400 2.396876e-07
## stateIowa          65.706383 66.7677031  0.9841043 3.311303e-01
## stateNebraska     131.192935 50.8573164  2.5796276 1.377434e-02
## age:stateIowa      -2.178763  1.2672928 -1.7192264 9.350204e-02
## age:stateNebraska  -2.896470  1.0166558 -2.8490174 6.967607e-03
```

Note: The *p*-values for the age coefficient is not the same as in the ANOVA table.
Reason: `anova()` tests the models against one another in the **order** specified.

As always, some model checking is necessary:



→ This seems ok.

An introduction to linear Algebra

Who has some knowledge of linear Algebra?

Overview

- The basics about
  - vectors
  - matrices
  - matrix algebra
  - matrix multiplication
- Why is linear Algebra useful?
- What does it have to do with data analysis and statistics?
- Regression equations in matrix notation.

Motivation

Why are vectors, matrices and their algebraic rules useful?

- **Example 1:** The observations for a covariate x or the response y for all individuals $1 \leq i \leq n$ can be stored in a vector (vectors and matrices are always given in **bold** letters):

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} , \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix} .$$

- **Example 2:** Covariance matrices for multiple variables. Say we have $\mathbf{x}^{(1)}$ and $\mathbf{x}^{(2)}$. The covariance matrix is then given as

$$\begin{pmatrix} Var(\mathbf{x}^{(1)}) & Cov(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) \\ Cov(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) & Var(\mathbf{x}^{(2)}) \end{pmatrix} .$$

- **Example 3:** The data (e.g. of some regression model) can be stored in a matrix:

$$\tilde{X} = \begin{pmatrix} 1 & x_1^{(1)} & x_1^{(2)} \\ 1 & x_2^{(1)} & x_2^{(2)} \\ \ldots & \ldots & \ldots \\ 1 & x_n^{(1)} & x_n^{(2)} \end{pmatrix} .$$

  This is the so-called design matrix with a vector of 1's in the first column.

- **Example 4:** A linear regression model can be written compactly using matrix multiplication:

$$y = \tilde{X} \cdot \tilde{\beta} + e \ ,$$

  with $\tilde{\beta}$ the vector of regression coefficients and e the vector of errors

Why do we discuss this topic in our course?

- Useful for compact notation.
- Enabels you to understand many statistical texts (books, research articles) that remain inaccessible otherwise.
- Useful for efficient coding, e.g. in R, which helps to increase speed and to reduce error rates.
- More advanced concepts often rely on linear algebra, e.g. principal component analysis (PCA) or random effects models.
- Is part of a general education (Allgemeinbildung) ;-)

Marices

An $n \times m$ Matrix is given as

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix} ,$$

with rows $1 = 1, \dots, n$ and columns $j = 1, \dots, m$.

Quadratic matrix: $n = m$. Example:

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 3 & 2 \\ 6 & 1 & 9 \end{pmatrix}$$

Symmetric matrix: $a_{ij} = a_{ji}$. Example:

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{pmatrix} .$$

The diagonal of a quadratic matrix is given by $(a_{11}, a_{22}, \ldots, a_{nn})$. Example: the diagonal of the above matrix is given as

$$(a_{11}, a_{22}, a_{33}) = (1, 3, 5) .$$

Diagonal matrix: A matrix that has entries $\neq 0$ only on the diagonal. Example:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{pmatrix} .$$

Transposing a matrix: Given a matrix A. Exchange the rows by the columns and vice versa. This leads to the transposed matrix $A^\top$:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix} \quad \Rightarrow \quad A^\top = \begin{pmatrix} a_{11} & a_{21} & \dots & a_{n1} \\ a_{12} & a_{22} & \dots & a_{n2} \\ \vdots & \vdots & & \vdots \\ a_{1m} & a_{2m} & \dots & a_{nm} \end{pmatrix} .$$

Examples (note also the change of dimensions):

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{pmatrix} \quad \Rightarrow \quad A^\top = \begin{pmatrix} 1 & 5 & 9 & 13 \\ 2 & 6 & 10 & 14 \\ 3 & 7 & 11 & 15 \\ 4 & 8 & 12 & 16 \end{pmatrix}$$

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \end{pmatrix} \quad \Rightarrow \quad A^\top = \begin{pmatrix} 1 & 5 \\ 2 & 6 \\ 3 & 7 \\ 4 & 8 \end{pmatrix}$$

- Transposing a matrix twice leads to the original matrix:

$$(A^\top)^\top = A \ .$$

- When a matrix is symmetric, then

$$A^\top = A \ .$$

This is true in particular for diagonal matrices.

Vectors

A vector is nothing else than n numbers written in a column:

$$b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

Transposing a vector leads to a *row vector*:

$$\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}^\top = \begin{pmatrix} b_1 & b_2 & \dots & b_n \end{pmatrix}$$

Note: By definition (by default), a vector is always a column vector.

Addition and subtraction

- Adding and subtracting matrices and vectors is only possible when the objects have the same dimension.
- Examples: Elementwise addition (or subtraction)

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} + \begin{pmatrix} 3 & 2 & 1 \\ 6 & 5 & 4 \end{pmatrix} = \begin{pmatrix} 4 & 4 & 4 \\ 10 & 10 & 10 \end{pmatrix}$$

$$\begin{pmatrix} 1 \\ 4 \end{pmatrix} - \begin{pmatrix} 3 \\ 9 \end{pmatrix} = \begin{pmatrix} -2 \\ -5 \end{pmatrix}$$

- But this addition is not defined:

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} + \begin{pmatrix} 3 & 6 \\ 2 & 5 \\ 1 & 4 \end{pmatrix} =$$

Multiplication by a scalar

Multiplication with a "number" (scalar) is simple: Multiply each element in a vector or a matrix.

Examples:

$$3 \cdot \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} = \begin{pmatrix} 3 & 6 & 9 \\ 12 & 15 & 18 \end{pmatrix}$$

$$-2 \cdot \begin{pmatrix} 1 \\ 4 \\ -2 \end{pmatrix} = \begin{pmatrix} -2 \\ -8 \\ 4 \end{pmatrix}$$

Matrix multiplication

The multiplication of two matrices A and B is defined if

number of columns in A = number of rows in B.

It is easiest to explain matrix multiplication with an example:

$$
\begin{pmatrix} 2 & 1 \\ -1 & 0 \\ 3 & 1 \end{pmatrix} \cdot \begin{pmatrix} 3 & 1 \\ 4 & -2 \end{pmatrix} = \begin{pmatrix} 2\cdot 3 + 1 \cdot 4 & 2 \cdot 1 + 1 \cdot (-2) \\ (-1)\cdot 3 + 0 \cdot 4 & (-1) \cdot 1 + 0 \cdot (-2) \\ 3 \cdot 3 + 1 \cdot 4 & 3 \cdot 1 + 1 \cdot (-2) \end{pmatrix}
$$

$$
= \begin{pmatrix} 10 & 0 \\ -3 & -1 \\ 13 & 1 \end{pmatrix}
$$

▸ Matrix multiplication app

Matrix multiplication rules I

Attention: Matrix multiplication does not follow the same rules as scalar multiplication!!!

- It can happen that $A \cdot B$ can be calculated, but $B \cdot A$ is not defined (see example on previous slide).
- In general: $A \cdot B \neq B \cdot A$, even if both are defined.
- It can happen that $A \cdot B = 0$ (0 matrix), although both $A \neq 0$ and $B \neq 0$.
- The Assoziativgesetz holds: $A \cdot (B \cdot C) = (A \cdot B) \cdot C$.
- The Distributivgesetz holds:

$$\begin{aligned} A \cdot (B + C) &= A \cdot B + A \cdot C \\ (A + B) \cdot C &= A \cdot C + B \cdot C \end{aligned}$$

Matrix multiplication rules II

- Transposing inverts the order: $(A \cdot B)^\top = B^\top \cdot A^\top$.
- The product $A \cdot A^\top$ is always symmetric.
- All these rules also hold for vectors, which can be interpreted as $n \times 1$ matrices:

$$a \cdot b^\top = \begin{pmatrix} a_1 b_1 & a_1 b_2 & \dots & a_1 b_m \\ a_2 b_1 & a_2 b_2 & \dots & a_2 b_m \\ \vdots & \vdots & & \vdots \\ a_n b_1 & a_n b_2 & \dots & a_n b_m \end{pmatrix}$$

If a and b have the same length:

$$a^\top \cdot b = \sum_i a_i b_i$$

Short exercise

Given vectors a and b and matrix C:

$$a = \begin{pmatrix} 1 \\ -2 \\ 3 \\ 0 \end{pmatrix}, \quad b = \begin{pmatrix} -2 \\ 4 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \\ -1 & 2 \end{pmatrix}$$

Calculate, if defined

- $a^\top \cdot b$
- $a \cdot b^\top$
- $C \cdot a$
- $C \cdot b$

The length of a vector

The length of a vector $a^\top = (a_1, a_2, \ldots, a_n)$ is defined as $||a||$ with

$$||a||^2 = a^\top \cdot a = \sum_i a_i^2 \; .$$

This is basically the Pythagoras idea in 2, 3, ... n dimensions.

In 2 dimensions: $||a|| = \sqrt{a_1^2 + a_2^2}$:

Identity matrix (Einheitsmatrix)

The identity matrix (of dimension m) is probably the simplest matrix that exists. It has 1's on the diagonal and 0's everywhere else:

$$I = \begin{pmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \ldots & 1 \end{pmatrix}$$

Multiplication with the identity matrix leaves a $m \times n$ matrix A unchanged:

$$A \cdot I = A \ .$$

Inverse matrix

Given a quadratic matrix A that fulfills

$$B \cdot A = I \ ,$$

then B is called the inverse of A (and vice versa). One then writes

$$B = A^{-1} \ .$$

Note:

- In that case it also holds that $A \cdot B = I$.
- Therefore: $A = B^{-1} \quad \Leftrightarrow \quad B = A^{-1}$

- The inverse of A may not exist. If it exists, A is regular, otherwise singular.
- $(A^{-1})^{-1} = A$.
- The inverse of a matrix product is given as

$$(A \cdot B)^{-1} = B^{-1} \cdot A^{-1} \ .$$

- It is

$$(A^\top)^{-1} = (A^{-1})^\top \ .$$

  Therefore one may also write $A^{-\top}$.

Linear regression in matrix notation

Linear regression with n data points can be understood as an equation system with n equations.

Remember example 4 from slide 21: We said that a linear regression model can be written compactly using matrix multiplication:

$$y = \tilde{X} \cdot \tilde{\beta} + e \ .$$

Task: Verify this now, using a model with two variables $x^{(1)}$ and $x^{(2)}$ and

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \ \tilde{X} = \begin{pmatrix} 1 & x_1^{(1)} & x_1^{(2)} \\ 1 & x_2^{(1)} & x_2^{(2)} \\ \dots & \dots & \dots \\ 1 & x_n^{(1)} & x_n^{(2)} \end{pmatrix}, \ \tilde{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix}, \ e = \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{pmatrix}.$$

It can be shown (see Stahel 3.4f,g) that the least-squares estimates $\hat{\beta}$ can be calculated as

$$\hat{\beta} = (\tilde{X}^\top \tilde{X})^{-1} \cdot \tilde{X}^\top \cdot y$$

Does this look complicated?

Let's test this in R ....

Doing linear algebra in R

Let us look at model $y = \tilde{X} \cdot \tilde{\beta} + e$ with coefficients

$\beta_0 = 10, \beta_1 = 5, \beta_2 = -2$ and variables

i	$x_i^{(1)}$	$x_i^{(2)}$
1	0	4
2	1	1
3	2	0
4	3	1
5	4	4

Thus the model is given as

$$y_i = 10 + 5x_i^{(1)} - 2x_i^{(2)} + \epsilon_i \text{ , for } \quad 1 \leq i \leq n \ .$$

Let us start by generating the "true" response, calculated as $\tilde{X}\tilde{\beta}$

```
x1 <- c(0,1,2,3,4)
x2 <- c(4,1,0,1,4)
Xtilde <- matrix(c(rep(1,5),x1,x2),ncol=3)
Xtilde
```

```
##      [,1] [,2] [,3]
## [1,]    1    0    4
## [2,]    1    1    1
## [3,]    1    2    0
## [4,]    1    3    1
## [5,]    1    4    4
```

```
t.beta <- c(10,5,-2)
t.y <- Xtilde%*%t.beta
t.y
```

```
##      [,1]
## [1,]    2
## [2,]   13
## [3,]   20
## [4,]   23
## [5,]   22
```

Next, we generate the vector containing the $\epsilon_i \sim N(0, \sigma^2)$ with $\sigma^2 = 1$:

```
t.e <- rnorm(5,0,1)
t.e
```

```
## [1]  0.7606833 -0.3257157  0.6830309  0.9070262  0.9342162
```

which we add to the "true" $y = \tilde{X}\tilde{\beta}$ values, to obtain the "observed" values:

```
t.Y <- t.y  + t.e
t.Y
```

```
##            [,1]
## [1,]  2.760683
## [2,] 12.674284
## [3,] 20.683031
## [4,] 23.907026
## [5,] 22.934216
```

It is now possible to fit the model with `lm`:

```
r.lm <- lm(t.Y ~ x1 + x2)
summary(r.lm)$coef
```

Alternatively, we can use formula

$$\hat{\beta} = (\tilde{X}^\top \tilde{X})^{-1} \tilde{X}^\top y$$

to find the parameter estimates:

```
solve(t(Xtilde) %*% Xtilde)  %*%  t(Xtilde) %*% t.Y
```

```
##             [,1]
## [1,] 10.069826
## [2,]  5.157981
## [3,] -1.896970
```

- `solve()` calculates the inverse (here the inverse of $\tilde{X}^\top \tilde{X}$).
- `t()` gives the transposed (here of $\tilde{X}^\top$).

Task: Do this calculation by yourself and verify for each step that the dimensions of the matrices and the vector are indeed fitting, so that this expression is defined.

Appendix

Some R commands for matrix algebra

Reading vectors and a matrices into R:

```
a <- c(1,2,3)
a
```

```
## [1] 1 2 3
```

```
A <- matrix(c(1,2,3,4,5,6),byrow=T,nrow=2)
B <- matrix(c(6,5,4,3,2,1),byrow=T,nrow=2)
A
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    3
## [2,]    4    5    6
```

```
B
```

```
##      [,1] [,2] [,3]
## [1,]    6    5    4
## [2,]    3    2    1
```

Adding and subtracting:

```
A + B
```

```
##      [,1] [,2] [,3]
## [1,]    7    7    7
## [2,]    7    7    7
```

```
A - B
```

```
##      [,1] [,2] [,3]
## [1,]   -5   -3   -1
## [2,]    1    3    5
```

However, be careful, R sometims does unreasonable things:

```
A + a
```

```
##      [,1] [,2] [,3]
## [1,]    2    5    5
## [2,]    6    6    9
```

What happened here??

Matrix multiplication:

```
C <- A %*% t(B)
C
```

```
##      [,1] [,2]
## [1,]   28   10
## [2,]   73   28
```

```
A%*%a
```

```
##      [,1]
## [1,]   14
## [2,]   32
```

Matrix inversion (possible for quadratic matrices only):

```
solve(C)
```

```
##            [,1]       [,2]
## [1,]  0.5185185 -0.1851852
## [2,] -1.3518519  0.5185185
```

```
C %*% solve(C)
```

```
##      [,1] [,2]
## [1,]    1    0
## [2,]    0    1
```

Why does `solve(A)` or `solve(B)` not work?