Introduction to regression analysis

Natalia Levshina © 2023

MPI for Psycholinguistics
Visualization and Data Analysis with R 2023

Outline

1. Linear regression: main concepts and functions
2. Introduction to logistic regression
3. Introduction to mixed-effects models

Post-Christmas blues

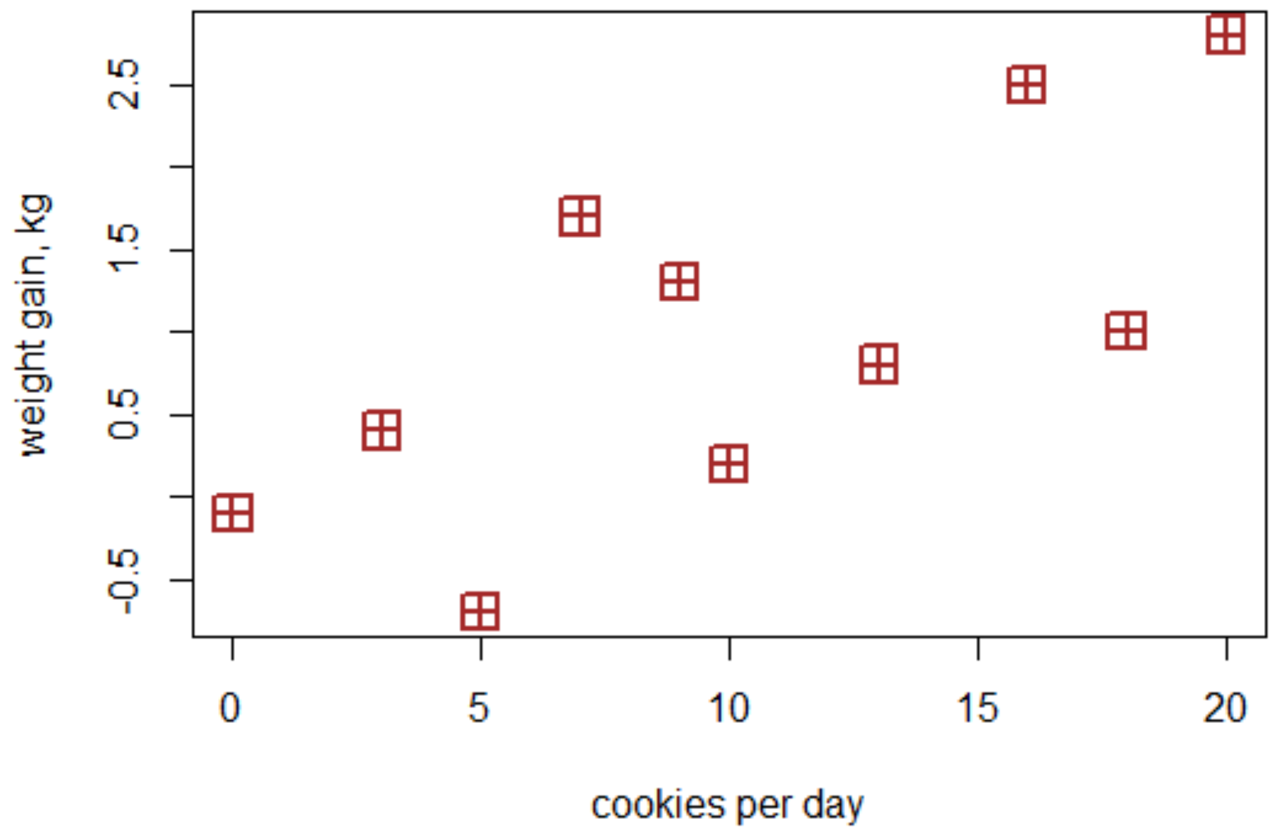
Name	Cookies eaten per day	Kilos gained
John	0	-0.1
Mary	3	0.4
Bill	5	-0.7
Jane	7	1.7
Laura	9	1.3
Ann	10	0.2
Chris	13	0.8
Eve	16	2.5
Peter	18	1.0
Steve	20	2.8

Data

```
cookies <- c(0, 3, 5, 7, 9, 10, 13, 16, 18, 20)
gain <- c(-0.1, 0.4, -0.7, 1.7, 1.3, 0.2, 0.8,
2.5, 1.0, 2.8)
```

Make a traditional scatterplot:

```
plot(x = cookies y = gain, xlab = "cookies per
day", ylab = "weight gain, kg", pch = 12, col =
"brown")
```



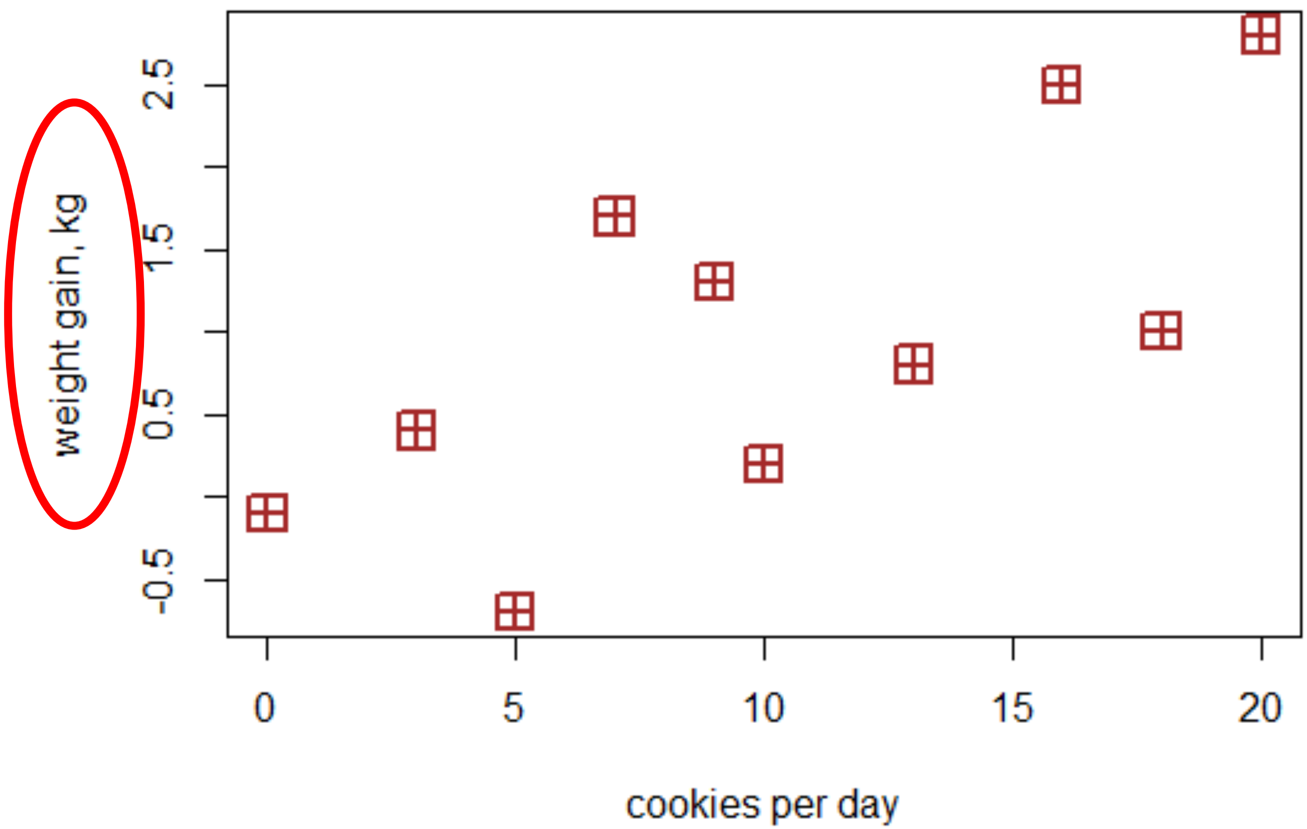
weight gain, kg
-0.5
0.5
1.5
2.5
0
5
10
15
20
cookies per day

Fundamental concepts of regression

- Dependent variable (response): weight gain
- Independent variable (predictor): cookies
- Regression line: line that can be drawn through the cloud of points such as the distances between the line and all points are as small as possible
- Intercept: value of y where the line crosses the y-axis
- Slope: increase of y per unit of x
- Fitted values: the y-coordinates of the projections of the points on the line
- Residuals: the differences between the observed and fitted response values (the distances between the points and their projections on the line)

Fundamental concepts of regression

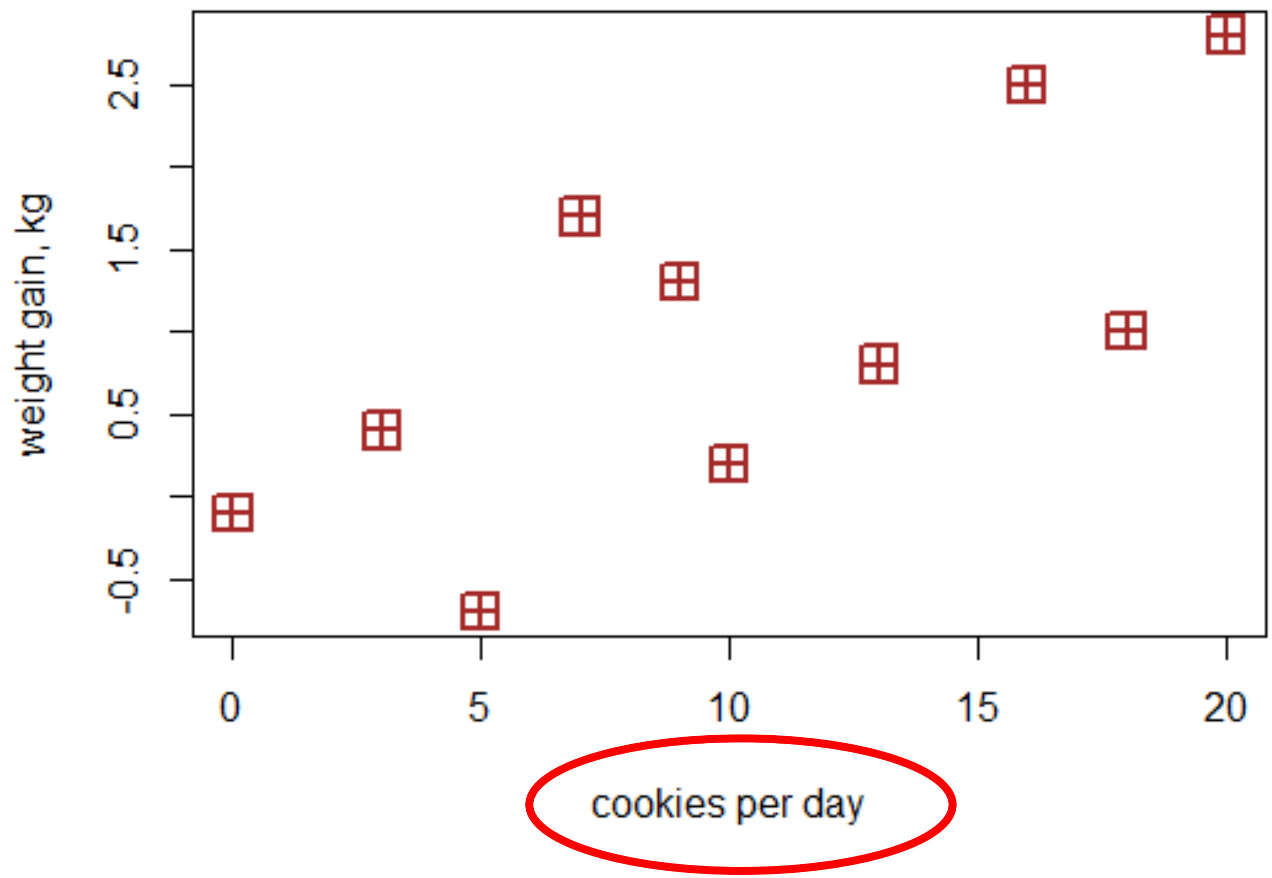
- Dependent variable (response): weight gain
- Independent variable (predictor): cookies
- Regression line: line that can be drawn through the cloud of points such as the distances between the line and all points are as small as possible
- Intercept: value of y where the line crosses the y-axis
- Slope: increase of y per unit of x
- Fitted values: the y-coordinates of the projections of the points on the line
- Residuals: the differences between the observed and fitted response values (the distances between the points and their projections on the line)



weight gain, kg
-0.5
0.5
1.5
2.5
0
5
10
15
20
cookies per day

Fundamental concepts of regression

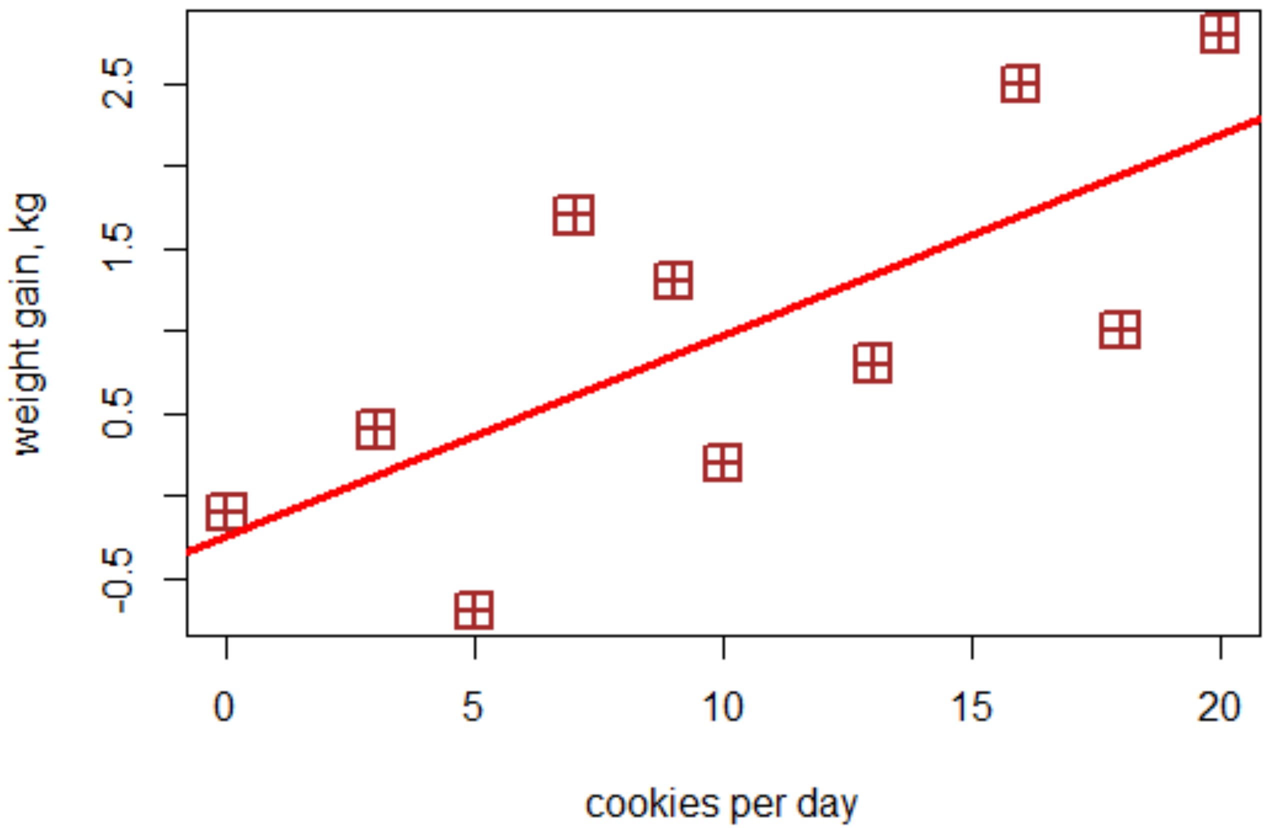
- Dependent variable (response): weight gain
- Independent variable (predictor): cookies
- Regression line: line that can be drawn through the cloud of points such as the distances between the line and all points are as small as possible
- Intercept: value of y where the line crosses the y-axis
- Slope: increase of y per unit of x
- Fitted values: the y-coordinates of the projections of the points on the line
- Residuals: the differences between the observed and fitted response values (the distances between the points and their projections on the line)



weight gain, kg
2.5
1.5
0.5
-0.5
0
5
10
15
20
cookies per day

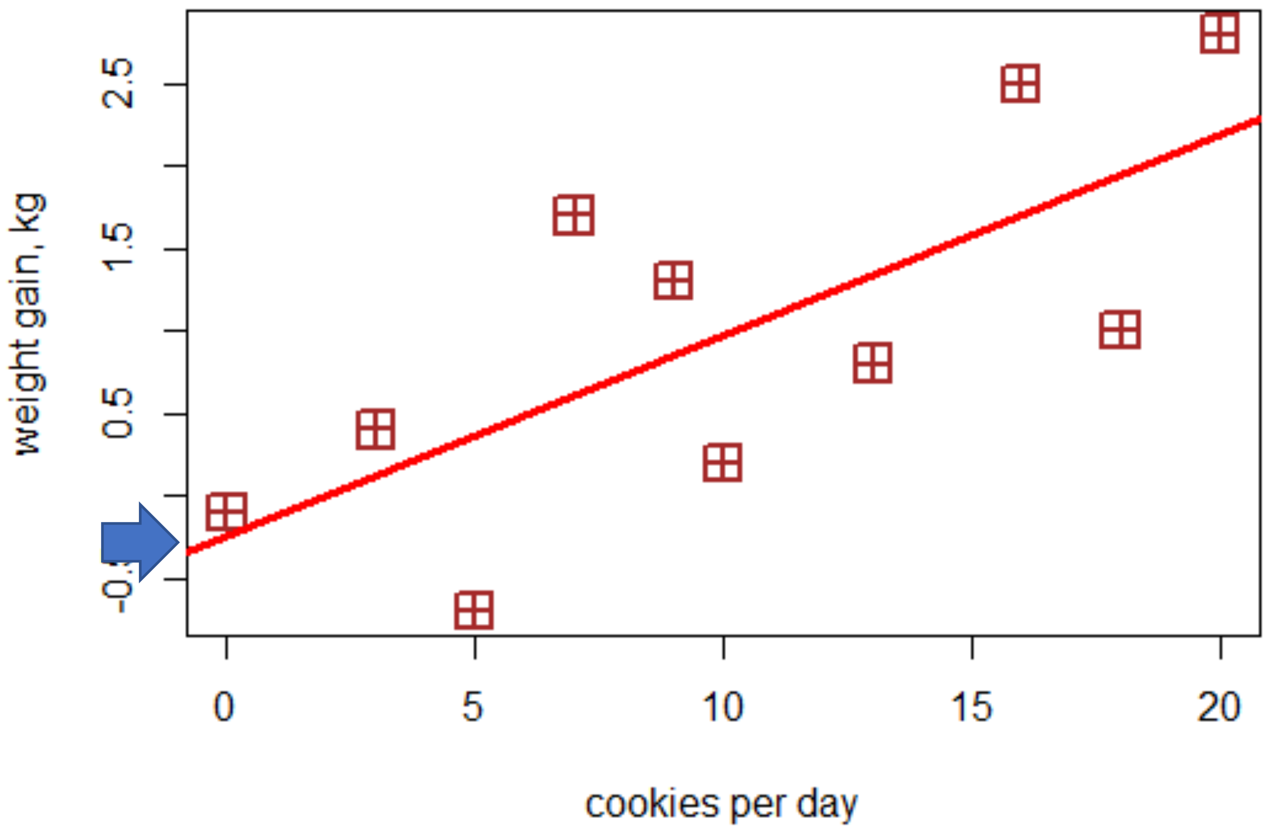
Fundamental concepts of regression

- Dependent variable (response): weight gain
- Independent variable (predictor): cookies
- Regression line: line that can be drawn through the cloud of points such as the distances between the line and all points are as small as possible
- Intercept: value of y where the line crosses the y-axis
- Slope: increase of y per unit of x
- Fitted values: the y-coordinates of the projections of the points on the line
- Residuals: the differences between the observed and fitted response values (the distances between the points and their projections on the line)


weight gain, kg
-0.5
0.5
1.5
2.5
0
5
10
15
20
cookies per day

Fundamental concepts of regression

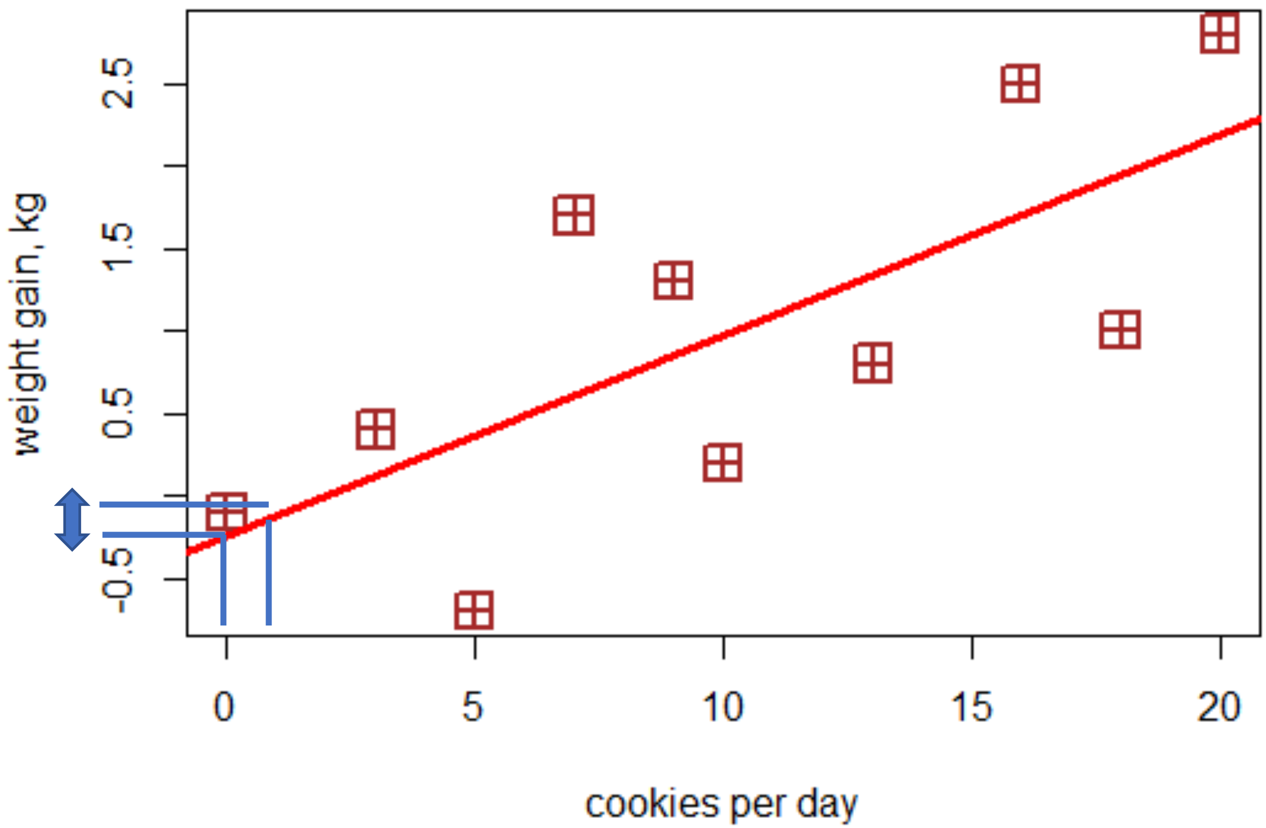
- Dependent variable (response): weight gain
- Independent variable (predictor): cookies
- Regression line: line that can be drawn through the cloud of points such as the distances between the line and all points are as small as possible
- Intercept: value of y where the line crosses the y-axis
- Slope: increase of y per unit of x
- Fitted values: the y-coordinates of the projections of the points on the line
- Residuals: the differences between the observed and fitted response values (the distances between the points and their projections on the line)



weight gain, kg
2.5
1.5
0.5
-0.5
0
5
10
15
20
cookies per day

Fundamental concepts of regression

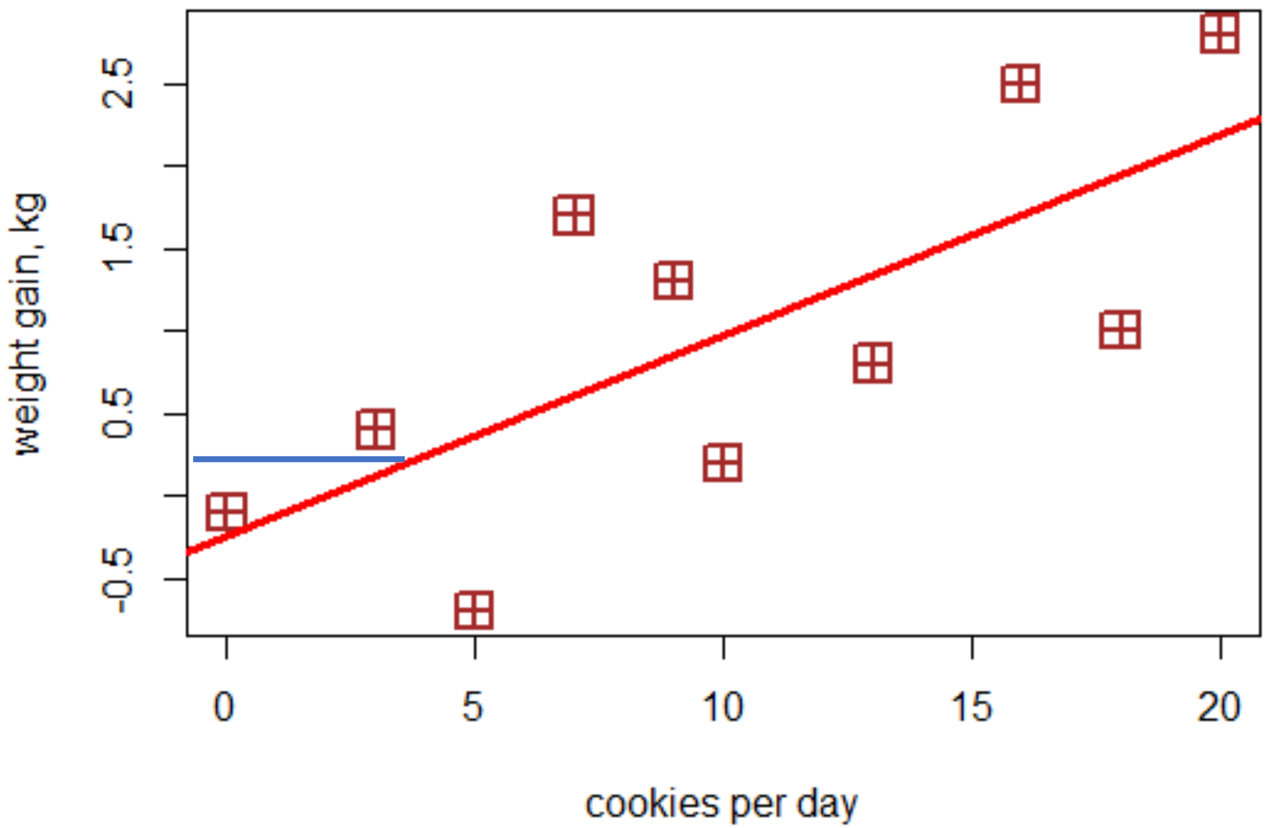
- Dependent variable (response): weight gain
- Independent variable (predictor): cookies
- Regression line: line that can be drawn through the cloud of points such as the distances between the line and all points are as small as possible
- Intercept: value of y where the line crosses the y-axis
- Slope: increase of y per unit of x on the line
- Fitted values: the y-coordinates of the projections of the points on the line
- Residuals: the differences between the observed and fitted response values (the distances between the points and their projections on the line)



2.5
1.5
0.5
-0.5
weight gain, kg
0
5
10
15
20
cookies per day

Fundamental concepts of regression

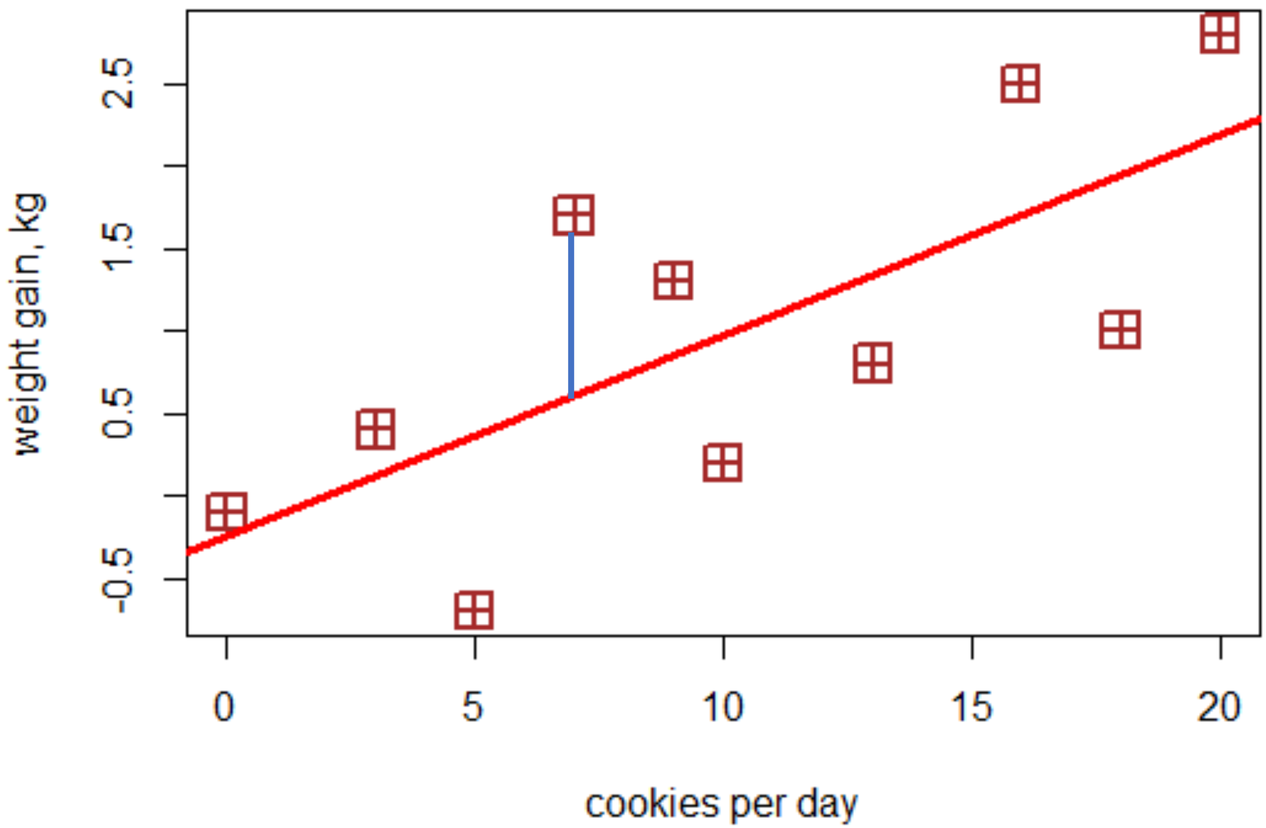
- Dependent variable (response): weight gain
- Independent variable (predictor): cookies
- Regression line: line that can be drawn through the cloud of points such as the distances between the line and all points are as small as possible
- Intercept: value of y where the line crosses the y-axis
- Slope: increase of y per unit of x
- Fitted values: the y-coordinates of the projections of the points on the line
- Residuals: the differences between the observed and fitted response values (the distances between the points and their projections on the line)



weight gain, kg
-0.5
0.5
1.5
2.5
0
5
10
15
20
cookies per day

Fundamental concepts of regression

- Dependent variable (response): weight gain
- Independent variable (predictor): cookies
- Regression line: line that can be drawn through the cloud of points such as the distances between the line and all points are as small as possible
- Intercept: value of y where the line crosses the y-axis
- Slope: increase of y per unit of x
- Fitted values: the y-coordinates of the projections of the points on the line
- Residuals: the differences between the observed and fitted response values (the distances between the points and their projections on the line)



weight gain, kg
-0.5
0.5
1.5
2.5
0
5
10
15
20
cookies per day

Basic linear regression with R

```
xmas_lm <- lm(gain ~ cookies)
summary(xmas_lm)

Call:
lm(formula = gain ~ cookies)
```

Intercept and slope in lm summary

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.23645    0.49341  -0.479   0.6446
cookies      0.12143    0.04151   2.925   0.0191 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
'.' 0.1 ' ' 1
```

Fitted values of lm

```
fitted(xmas_lm)
         1          2          3          4
-0.2364469  0.1278442  0.3707050  0.6135658
         5          6          7          8
 0.8564266  0.9778570  1.3421481  1.7064393
         9         10
1.9493001  2.1921609
```

Residuals

```
residuals(xmas_lm)
         1          2          3          4
 0.1364469  0.2721558 -1.0707050  1.0864342
         5          6          7          8
 0.4435734 -0.7778570 -0.5421481  0.7935607
         9         10
-0.9493001  0.6078391
```

Fitted, residuals and observed values

	fitted	residuals	gain
1	-0.2364469	0.1364469	-0.1
2	0.1278442	0.2721558	0.4
3	0.3707050	-1.0707050	-0.7
4	0.6135658	1.0864342	1.7
5	0.8564266	0.4435734	1.3
6	0.9778570	-0.7778570	0.2
7	1.3421481	-0.5421481	0.8
8	1.7064393	0.7935607	2.5
9	1.9493001	-0.9493001	1.0
10	2.1921609	0.6078391	2.8

The magic of linear regression

Fitted value of y = intercept + slope*value of x

$\bar{y} = \alpha + \beta x$

0bserved value of y = intercept + slope * value of x + residual

$y = \bar{y} + \varepsilon = \alpha + \beta x + \varepsilon$

How good is the fit?

```
Multiple R-squared:  0.5169
Adjusted R-squared:  0.4565
```

R-squared equals Pearson's correlation coefficients (one predictor):

```
cor(gain, cookies)^2
[1] 0.5168564
```

Adjusted R-squared is usually a more realistic estimate. It may be substantially smaller if

a) there are many useless predictors

b) the model overfits the data, e.g. due to small sample size, like here.

Using variables from a dataset

```
xmas_data <- data.frame(n_cookies = cookies,
weight_gain = gain)

xmas_lm <- lm(weight_gain ~ n_cookies, data =
xmas_data)

summary(xmas_lm) # identical
```

…

Categorical predictors

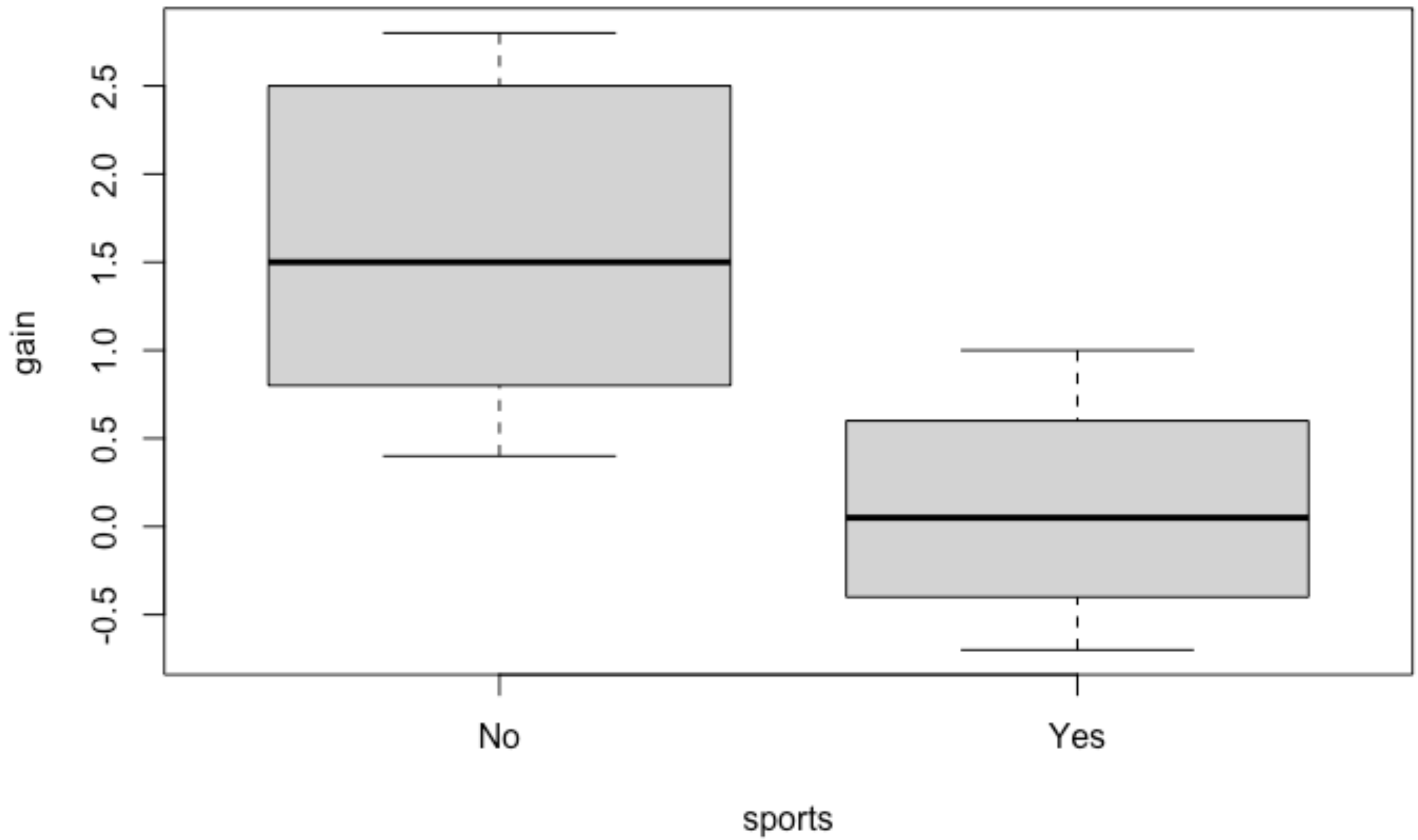
- But, some of our friends also did sports regularly, and some didn’t.

Name	Cookies eaten per day	Kilos gained	Sports
John	0	-0.1	Yes
Mary	3	0.4	No
Bill	5	-0.7	Yes
Jane	7	1.7	No
Laura	9	1.3	No
Ann	10	0.2	Yes
Chris	13	0.8	No
Eve	16	2.5	No
Peter	18	1.0	Yes
Steve	20	2.8	No

Data

```
sports <- c("Yes", "No", "Yes", "No", "No",
"Yes", "No", "No", "Yes", "No")
```

```
boxplot(gain ~ sports)
```


gain
-0.5
0.0
0.5
1.0
1.5
2.0
2.5
No
Yes
sports

How to represent them in regression?

- We can use dummy variables, representing categories as numbers.
- There are several ways of representing categorical variables in R.
- The default is so-called treatment contrasts (dummy coding).
- Binary variables: reference level = 0, the other = 1. One coefficient which shows the difference between the reference level and the other one.

Treatment contrasts

```
sports <- as.factor(sports)
levels(sports)
[1] "No"  "Yes"

contrasts(sports)
       Yes
No       0
Yes      1
```

Treatment contrasts with lm()

```
xmas_lm1 <- lm(gain ~ sports)
summary(xmas_lm1)
```

…

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   1.5833     0.3514   4.505  0.00199 **
sportsYes    -1.4833     0.5557  -2.669  0.02839 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'
0.1 ' ' 1
```

Interpreting the coefficient

```
aggregate(gain ~ sports, FUN = mean)
  sports     gain
1     No 1.583333
2    Yes 0.100000

1.583333 - 0.1
[1] 1.483333
```

Treatment coding of more than two levels

- Reference level = 0, each of the rest = 1.
- One coefficient for each level with the exception of the reference level, each shows the difference between the given level and the reference level.

```
gender <- c("M", "F", "D", "M", "F", "D")
gender <- as.factor(gender)
contrasts(gender)
```

```
     F M
D    0 0
F    1 0
M    0 1
```

Sum contrasts

- Often used in ANOVA
- Binary variables: The first level is coded as 1, the second as -1.

```
sports_sum <- sports
contrasts(sports_sum) <- contr.sum
contrasts(sports_sum)
```

```
    [,1]
No     1
Yes   -1
```

Sum contrasts with lm()

```
xmas_lm2 <- lm(gain ~ sports_sum)
summary(xmas_lm2)
…
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.8417     0.2778   3.029   0.0163 *
sports_sum1   0.7417     0.2778   2.669   0.0284 *
…
```

What does the beta coefficient stand for? How do you interpret the intercept?

Sum contrasts for more than two levels

```
contrasts(gender) <- contr.sum
contrasts(gender)
  [,1] [,2]
D    1    0
F    0    1
M   -1   -1
```

Multiple regression

```
xmas_lm3 <- lm(gain ~ cookies + sports)

summary(xmas_lm3)

…
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.45842    0.40927   1.120   0.2996
cookies       0.09926    0.02968   3.344   0.0124 *
sportsYes    -1.17729    0.37977  -3.100   0.0173 *
```

Is the model better?

1. Compare the R^2 values with the previous models. Is there an improvement?

2. Another useful statistic is Akaike Information Criterion (AIC). It is based on a trade-off between the goodness of fit of the model (e.g., captured by R^2), and the simplicity. It helps against both **overfitting** and **underfitting**.

```
AIC(xmas_lm)
[1] 28.24568
```

- The smaller AIC, the better.
- AIC is used as a relative, not absolute indicator. You can only compare models based on the same data!

Interactions

- An interaction means that the effects of two or more predictors are not additive.
- For example, chocolate is tasty, pizza is tasty, but chocolate + pizza???

```
xmas_lm4 <- lm(gain ~ cookies*sports)
xmas_lm4 <- lm(gain ~ cookies + sports +
cookies:sports) #alternatively
```

Interaction term in lm

```
summary(xmas_lm4)
…
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        0.20379    0.53648   0.380   0.7171
cookies            0.12172    0.04232   2.876   0.0282 *
sportsYes         -0.71992    0.71257  -1.010   0.3513
cookies:sportsYes -0.04704    0.06124  -0.768   0.4716
```

Testing interactions + model selection

```
anova(xmas_lm3, xmas_lm4)
```

```
Analysis of Variance Table

Model 1: gain ~ cookies + sports
Model 2: gain ~ cookies + sports + cookies:sports
  Res.Df    RSS Df Sum of Sq    F Pr(>F)
1      7 2.2823
2      6 2.0780  1   0.20434 0.59 0.4716
```

Exercise

1. Add a binary variable "Gender" and test if it has a significant effect alone and in the presence of the other predictors.
2. Use anova() to test if adding this variable is useful for the model.
3. Does AIC become better or worse if you add this variable? What about different versions of R^2?

Outline

1. Linear regression: main concepts and functions

2. Introduction to logistic regression

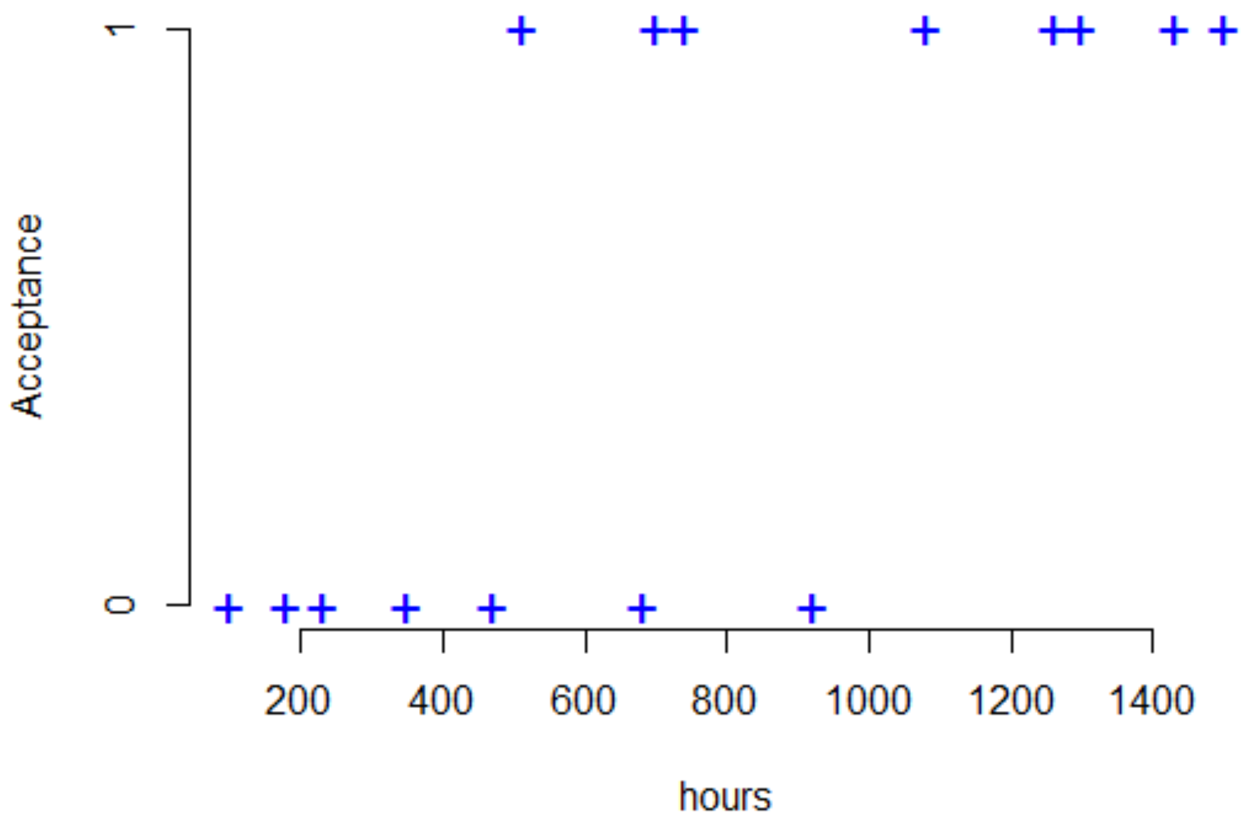
3. Introduction to mixed models

Binary outcome

- Binomial (dichotomous) logistic regression is used when the response variable is binary.

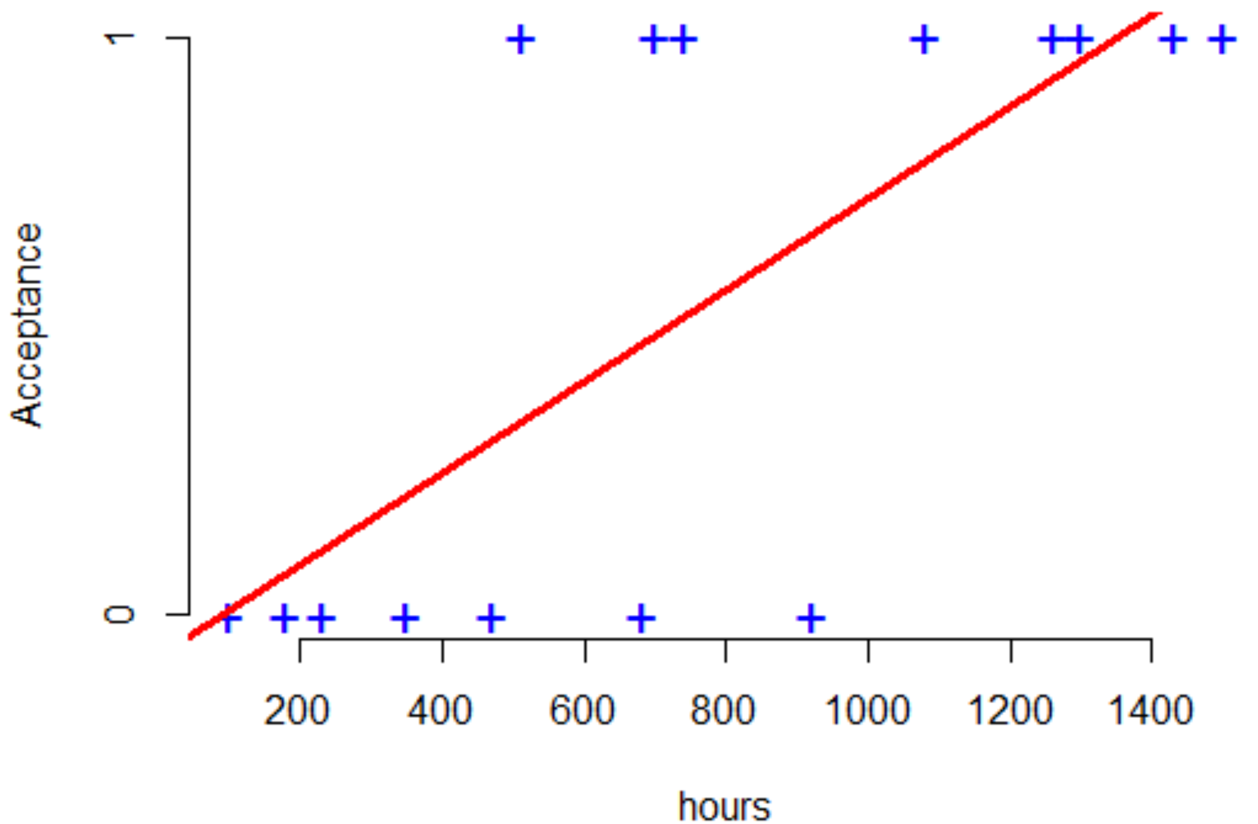
Success of submissions to journals

Person ID	hours	acceptance
1	100	No
2	180	No
3	230	No
4	350	Yes
5	470	No
6	510	No
7	680	No
8	700	Yes
9	740	Yes
10	920	Yes
11	1080	No
12	1260	Yes
13	1300	Yes
14	1430	Yes
15	1500	Yes



Acceptance
1
0
200
400
600
800
1000
1200
1400
hours

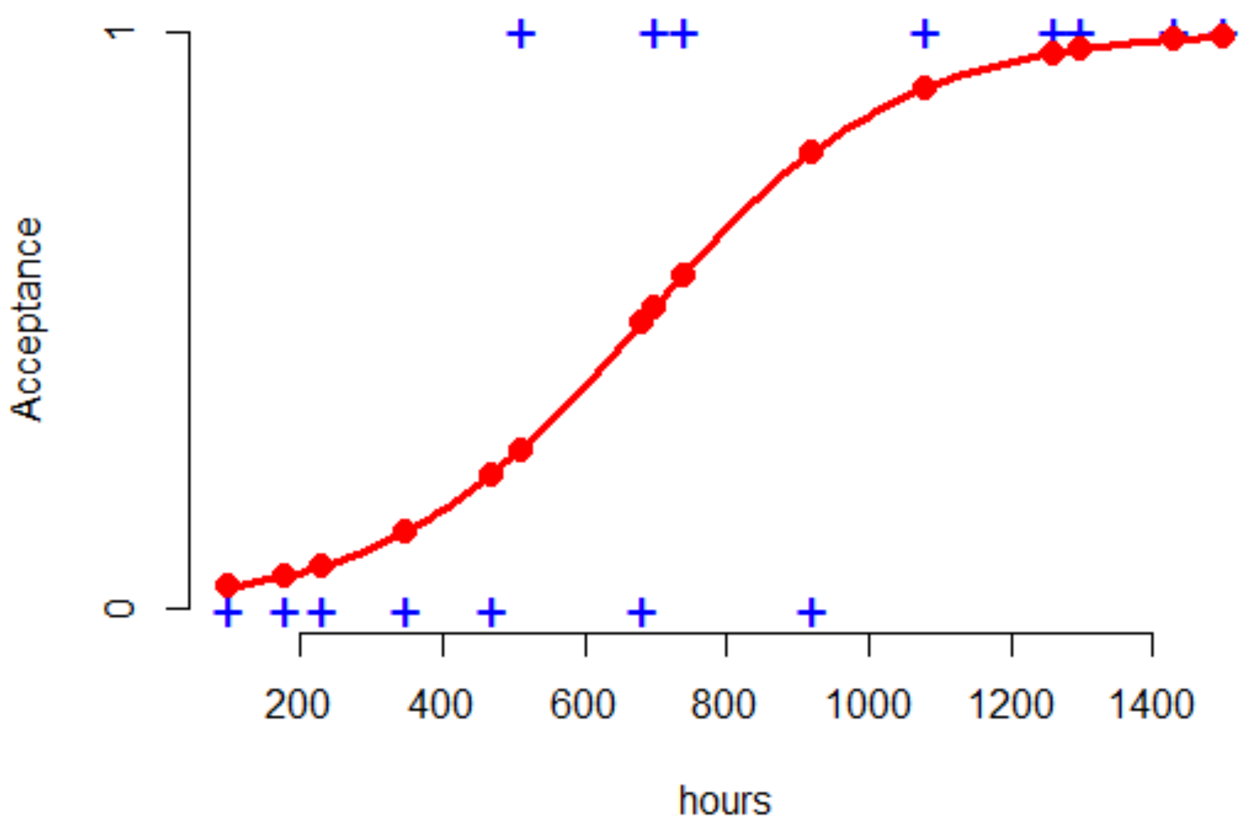
Linear regression



A small problem

- There’s a small problem with fitted values.
- For example, if you spend 2000 hours, your acceptance will be 1.51. If you spend 1 hour, your acceptance will be -0.07.

Logistic regression



The hocus pocus

- Linear model:

$$\bar{y} = \alpha + \beta x$$

- Logistic model:

$$\log \frac{P\ (y=1)}{P\ (y=0)} = \alpha + \beta x$$

Logit, or log odds

Fitted values in logistic regression

```
dat[c(5:7), ]
  acceptance hours predicted
5         No   470 0.2328821
6        Yes   510 0.2750818
7         No   680 0.4948156
```

- If you spend 2000 hours, your acceptance will be 0.999.
- If you spend 1 hour, your acceptance will be 0.02.
- Sounds more reasonable.

Two most useful functions

- `glm()` from the basic distribution

For example:

```
your.glm <- glm(Outcome ~ PredictorX + PredictorY
+ …, family = binomial, data = yourData)
summary(your.glm)
```

- `lrm()` from package `rms` by Frank Harrell

For example:

```
your.lrm <- lrm(Outcome ~ PredictorX + PredictorY
+ …, data = yourData)
your.lrm
```

GLM

```
pubs_glm <- glm(acceptance ~ hours, data = dat,
family = binomial)
summary(pubs_glm)
```

…

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.813746   1.988412  -1.918   0.0551 .
hours        0.005578   0.002781   2.006   0.0449 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'
0.1 ' ' 1
```

Interpretation of intercept in logistic regression

- Intercept shows the log-odds of outcome = 1 (i.e. accepted) for x = 0 (i.e. 0 hours).
- To get the normal odds, use exp():

```
exp(-3.81)
[1] 0.02214818
```

- If you spend 0 hours, the odds of being accepted to being rejected are about 0.02.
- This doesn't make much sense, of course.
- Usually, the intercept does not represent very interesting information.

Interpretation of slope in logistic regression

- Slope = log-odds ratio.
- If the slope coefficient positive, the chances of outcome = 1 (accepted) increase with x (hours spent). If negative, they decrease.
- To get the normal odds ratio, use exp():

```
exp(0.005578)
[1] 1.005594
```

- This means that for every unit increase in x (i.e. for every hour), the odds of outcome = 1 (accepted) are multiplied by 1.006.

Outline

1. Linear regression: main concepts and functions

2. Introduction to logistic regression

3. Introduction to mixed models

Why are mixed-effects models important?

- Imagine some data about sales of ice-cream on one day by people in different countries.
- We also have data about the temperature on that day.
- What kind of relationship would you expect?

Creating a dataframe from scratch

```
Finland <- data.frame(Temperature = c(0, 3, 5, 10),
Sales = c(650, 730, 910, 1000))
Ireland <- data.frame(Temperature = c(5, 6, 12, 15),
Sales = c(600, 770, 810, 890))
Italy <- data.frame(Temperature = c(12, 15, 16, 20),
Sales = c(420, 500, 720, 800))
China <- data.frame(Temperature = c(17, 18, 22, 24),
Sales  = c(300, 480, 500, 790))
India <- data.frame(Temperature = c(22, 25, 26, 30),
Sales = c(160, 180, 300, 510))


icecream_data <- rbind(Finland, Ireland, Italy,
China, India)
icecream_data$Country <- c(rep("Finland", 4),
rep("Ireland", 4), rep("Italy", 4), rep("China", 4),
rep("India", 4))
```

Simple linear regression

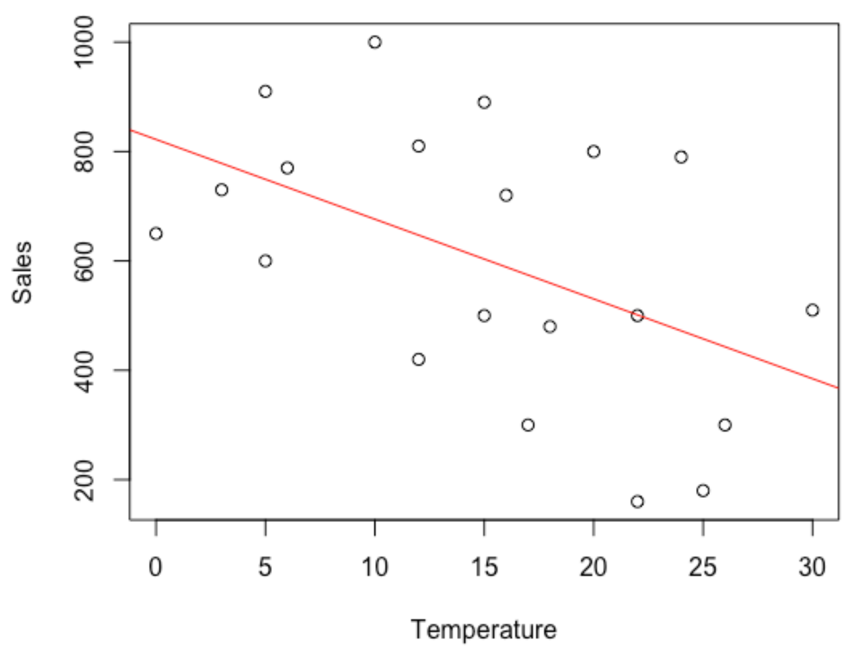
```
icecream_lm <- lm(Sales ~ Temperature, data =
icecream_data)
summary(icecream_lm)
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  822.049    102.288   8.037  2.3e-07 ***
Temperature  -14.591      5.932  -2.460   0.0243 *
```

The effect is negative: the warmer it is, the less ice-cream is sold. This is weird!

Scatterplot with regression line

```
plot(Sales ~ Temperature, data = icecream_data)
abline(icecream_lm, col = "red")
```

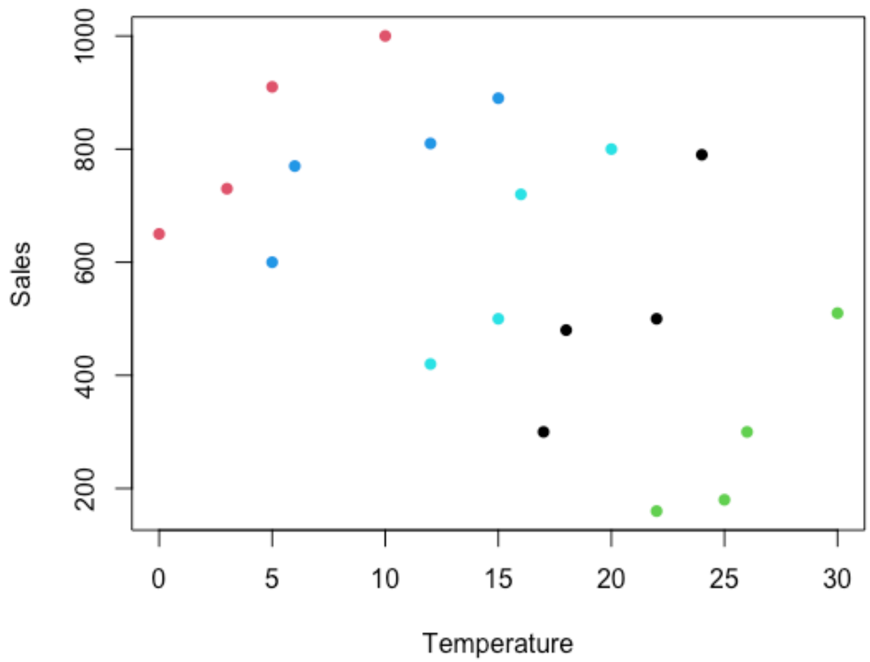


Dependent data points

- But the observations actually come only from five countries:
  - Finland
  - Ireland
  - Italy
  - China
  - India

Scatterplot with countries

```
plot(Sales ~ Temperature, col =
as.numeric(as.factor(icecream_data$Country)), data
= icecream_data, pch = 16)
```



Mixed models with random intercepts

```
library(lme4)
icecream_lmer <- lmer(Sales ~ (1|Country) +
Temperature, data = icecream_data)
summary(icecream_lmer)
```

…

```
Fixed effects:
            Estimate Std. Error t value
(Intercept)   56.900    245.898   0.231
Temperature   35.914      5.637   6.371
```

The effect is now positive: the warmer, the better the sales!

Mixed model: Random intercepts

```
ranef(icecream_lmer)
```

```
$Country
        (Intercept)
China    -264.84606
Finland   599.87220
India    -689.46203
Ireland   366.89895
Italy     -12.46305
```

Preference for ice-cream in each country (the mean is zero).

Interpretation of random intercepts

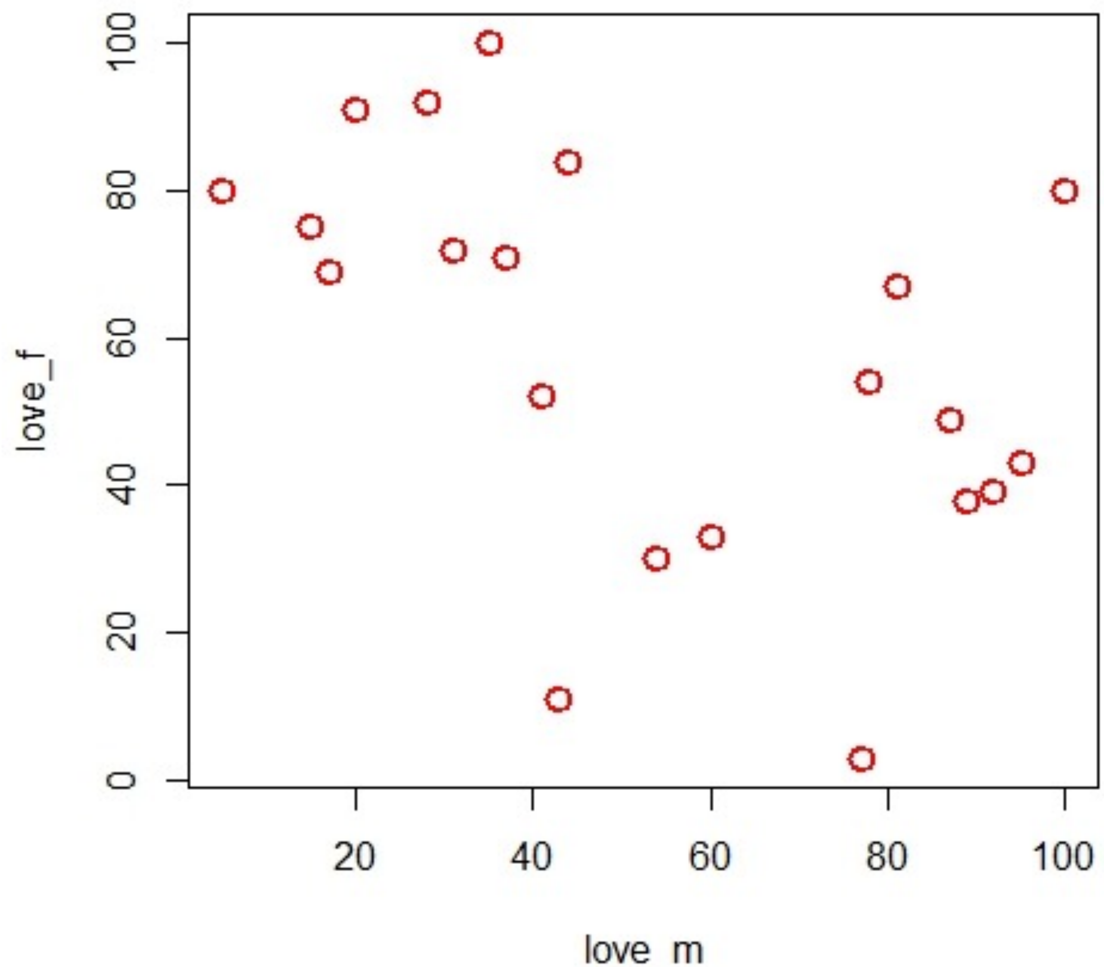
- Random intercepts are used when one and the same adjustment can be added to all members of one group.
- But the effect of the predictor is the same in each group.

Random slopes

- A hypothesis:

"The less we love her when we woo her, the more we draw a woman in."

Alexander Pushkin,
Eugene Onegin

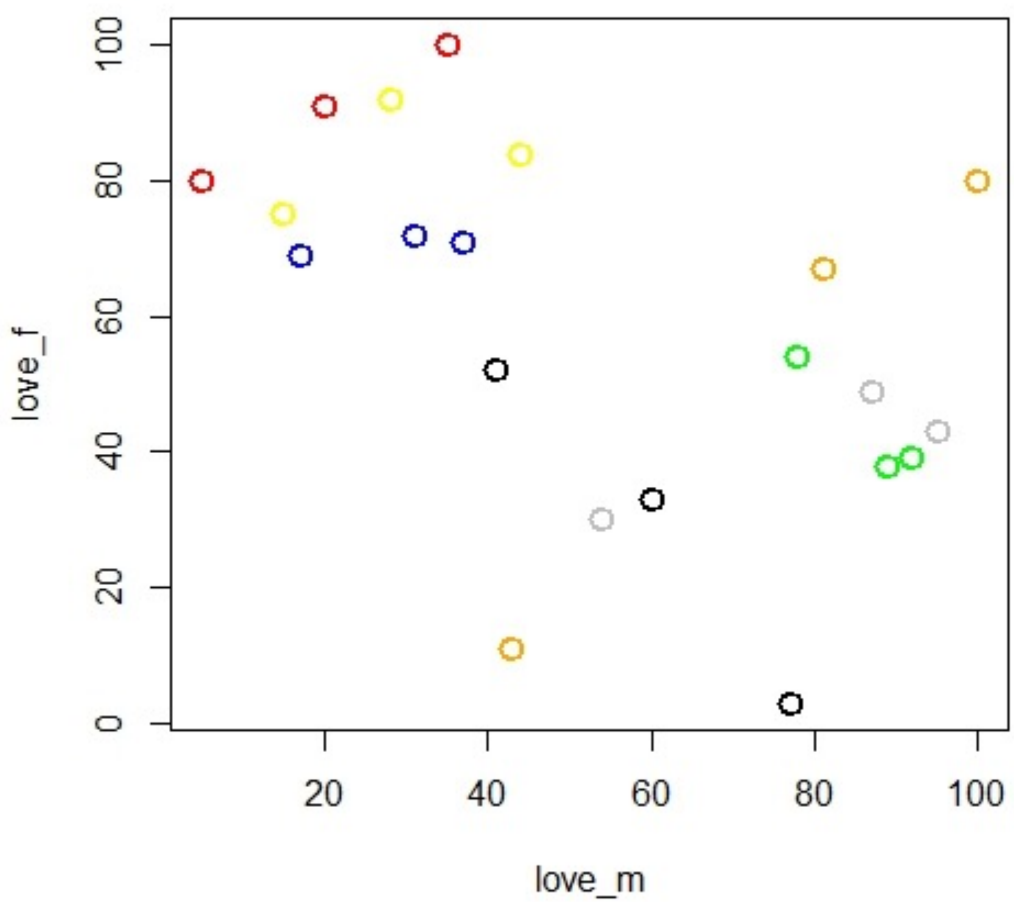


love_f
0
20
40
60
80
100
20
40
60
80
100
love_m

Simple linear model

	Estimate	Std.Error	t value	Pr(>|t|)	
(Intercept)	81.7032	10.9257	7.478	4.5e-07	***
love_m	-0.4276	0.1783	-2.398	0.0269	*

Dependent data points



Mixed model with random intercepts and slopes

- The effect is very weak and positive:

	Estimate	Std. Error	t value
(Intercept)	59.20473	22.01777	2.689
love_m	0.04455	0.34546	0.129

Interpretation of random slopes

- Random slopes show that the effect of a predictor on the response varies from group to group.

```
$Subject
  (Intercept)       love_m
a   20.786878  0.44339018
b   18.276733  0.15665153
c    9.441192  0.02183356
d   45.072418 -1.30241964
e   46.838881 -0.76804347
f  -45.174025  0.29898868
g  -95.242077  1.14959916
```

How to add random effects: popular configurations

- Random intercepts: (1|Group)
- Random intercepts and slopes: (1 + Predictor|Group) or simply (Predictor|Group)
- Random slopes only: (0 + Predictor|Group)
- Nested random effects, e.g. Pupil is nested under School:

(1|School/Pupil), which is equivalent to (1|School) + (1|School:Pupil). This means that intercepts vary within schools and intercepts of pupils vary within schools.

- Crossed random effects, e.g., Participants and Stimulus:
  (1|Participants) + (1|Stimulus)

Fixed or random?

- Theoretical considerations:
  - Groups are assumed to be randomly sampled from a population of groups!
    - participants from cities or different schools
    - subjects measured repeatedly ("repeated measures")
    - lexemes or semantic categories
    - languages or linguistic areas
  - The groups are 'noise'. They are not directly relevant for your hypothesis.

- Practical considerations:
  - If you have 2 to 5 groups in a grouping factor, you can just as well include them as proper fixed effects.
  - If you have more than 5 groups, it's better to enter them as random effects because there may not be enough data points to estimate the p-values in a reliable way.

RE specification and singular fit

```
icecream_lmer1 <- lmer(Sales ~ (1 +
Temperature|Country) + Temperature, data =
icecream_data)
boundary (singular) fit: see help('isSingular')
```

```
anova(icecream_lmer, icecream_lmer1)
refitting model(s) with ML (instead of REML)
Data: icecream_data
Models:
icecream_lmer: Sales ~ (1 | Country) + Temperature
icecream_lmer1: Sales ~ (1 + Temperature | Country) + Temperature
               npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
icecream_lmer     4 265.30 269.28 -128.65   257.30
icecream_lmer1    6 267.66 273.64 -127.83   255.66 1.6381  2     0.4409
```

A word on maximal models

- Some people (Barr et al. 2013) have argued that one needs to include all possible random effects, intercepts and slopes (so called ‘maximal’ models).
- However, Bates et al. (2015) have shown that this leads to overspecification and loss of statistical power. One also has convergence problems, which are not due to bad algorithms, but because the overly complex random effect structure is not supported by the data.
- This is why I recommend to fit parsimonious models and include only those random effects which are supported by your data and motivated by your theory. You can use the likelihood ratio test (with the help of *anova*) for that purpose.

Goodness of fit: Marginal and conditional R^2

- **Marginal R^2** represents the variance explained by fixed factors.
- **Conditional R^2** is variance explained by both fixed and random factors (i.e. the entire model).

```
library(MuMIn)
r.squaredGLMM(icecream_lmer)
The result is correct only if all data used by the
model has not changed since model was fitted.
           R2m       R2c
[1,] 0.2534215 0.9800652
```

Exercise

- Load the data frame data_all_clean (unless you have it already in your work space).

```
load("data_all_clean.R")
```

Which variable can be the response variables? Which are fixed effects? Which are random effects?

glmer() for logistic mixed-effects models

```
dm_glmer0 <- glmer(Marker ~ (1|Version_Group), data =
data_all_clean, family = binomial)

dm_glmer1 <- glmer(Marker ~ (1|Version_Group) +
Stimulus_Type, data = data_all_clean, family = binomial)

anova(dm_glmer0, dm_glmer1)
Data: data_all_clean
Models:
dm_glmer0: Marker ~ (1 | Version_Group)
dm_glmer1: Marker ~ (1 | Version_Group) + Stimulus_Type
          npar    AIC    BIC  logLik deviance Chisq Df Pr(>Chisq)
dm_glmer0    2 1362.1 1373.5 -679.05   1358.1
dm_glmer1    4 1347.8 1370.7 -669.89   1339.8 18.33  2  0.0001046 ***
```

Is the predictor useful?

glmer summary

```
summary(dm_glmer1)
Generalized linear mixed model fit by maximum likelihood
(Laplace Approximation) ['glmerMod']
 Family: binomial  ( logit )
Formula: Marker ~ (1 | Version_Group) + Stimulus_Type
   Data: data_all_clean

     AIC      BIC   logLik deviance df.resid
  1347.8   1370.6   -669.9   1339.8     2246
```

```
…
Random effects:
 Groups        Name        Variance Std.Dev.
 Version_Group (Intercept) 12.1     3.478
Number of obs: 2250, groups:  Version_Group, 55

Fixed effects:
                                      Estimate Std. Error z value Pr(>|z|)
(Intercept)                            -3.0067     0.5599  -5.370 7.88e-08 ***
Stimulus_TypeDifferent_Actions&Actors   0.1869     0.1871   0.999    0.318
Stimulus_TypeDifferent_Actors           0.7355     0.1820   4.042 5.29e-05 ***
---
```

Exercise

1. Add one more random intercept. Is it useful?

2. Add another fixed effect. Does it improve the model?

3. Try adding a random slope. Does it help?