Gaussova distribuce a testování hypotéz

Mojmír Dočekal

2024-04-30

Contents

Chapter 1

Dvě témata

- Gaussova distribuce
- testování hypotéz: t-distribuce, t.test

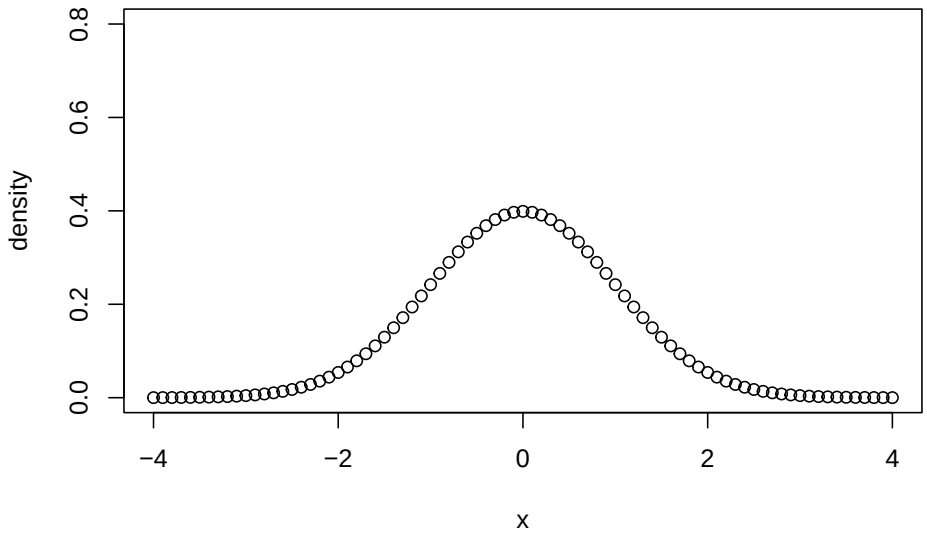
Chapter 2

Gaussova distribuce

2.1 Nejznámější typ distribuce

- klasický příklad

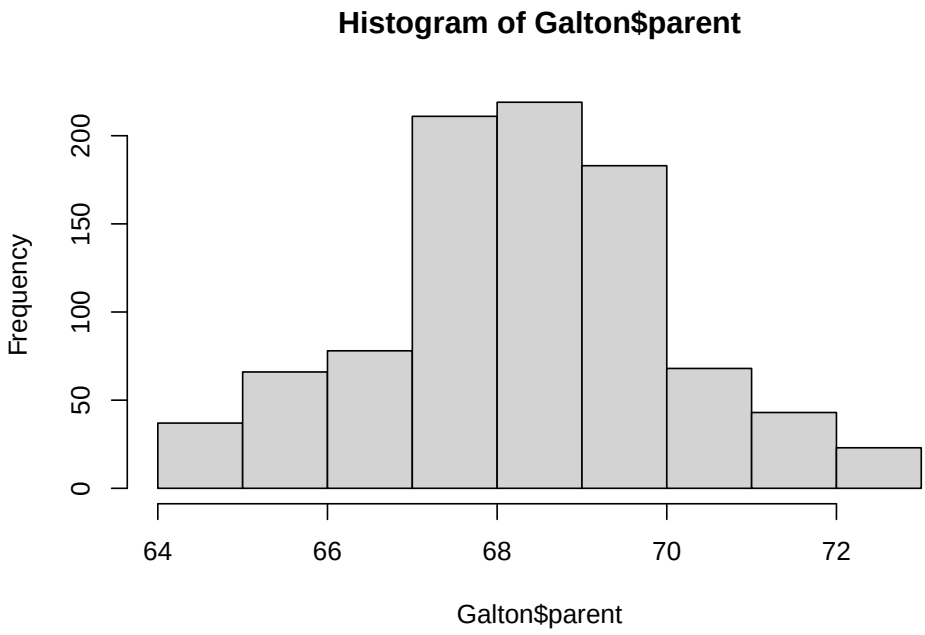
```
x = seq(-4, 4, 0.1)
y = dnorm(x)

plot(x, y, xlab = "x", ylab = "density", ylim = c(0, 0.8))
```



- některé příklady:

1. průměrná výška, váha, ...

```
library(HistData)

data("Galton")

hist(Galton$parent)
```




```
summary(Galton$parent)
```

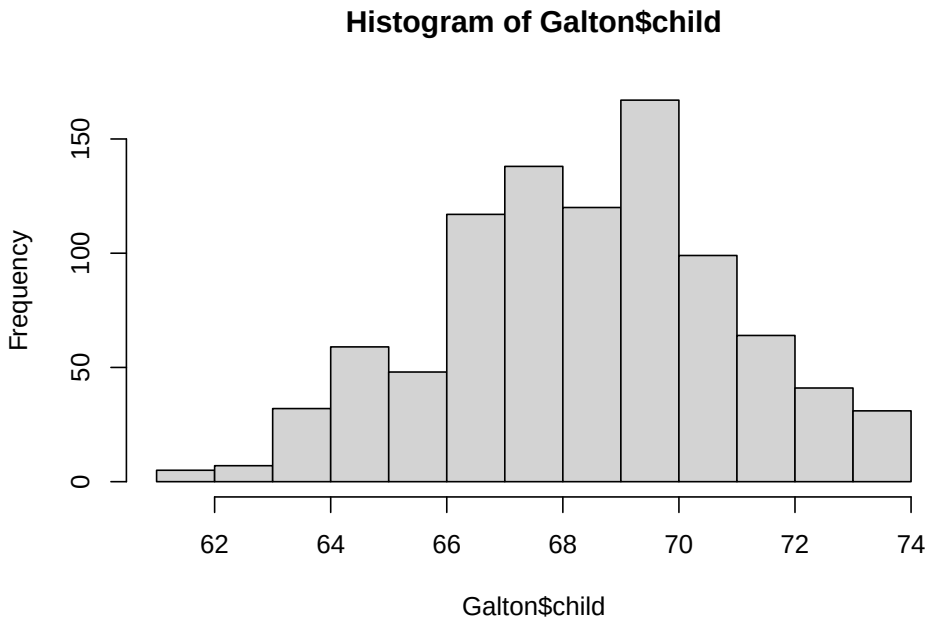
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   64.00   67.50   68.50   68.31   69.50   73.00
```

```
sd(Galton$parent)
```

```
## [1] 1.787333
```

- a děti:

```
library(HistData)

data("Galton")

hist(Galton$child)
```




```
summary(Galton$child)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   61.70   66.20   68.20   68.09   70.20   73.70
```

```
sd(Galton$child)
```

```
## [1] 2.517941
```

2. chyby v měření (historie)

- historie
- Gaussovo nármální rozložení
- objeveno Gaussem při řešení problému dráhu trpasličí planety Ceres
- vycházel z chyb měření (každé měření nutně vnáší chybu)
- jejich aproximací přesně vypočítal, kde se Ceres objeví po průletu za sluncem

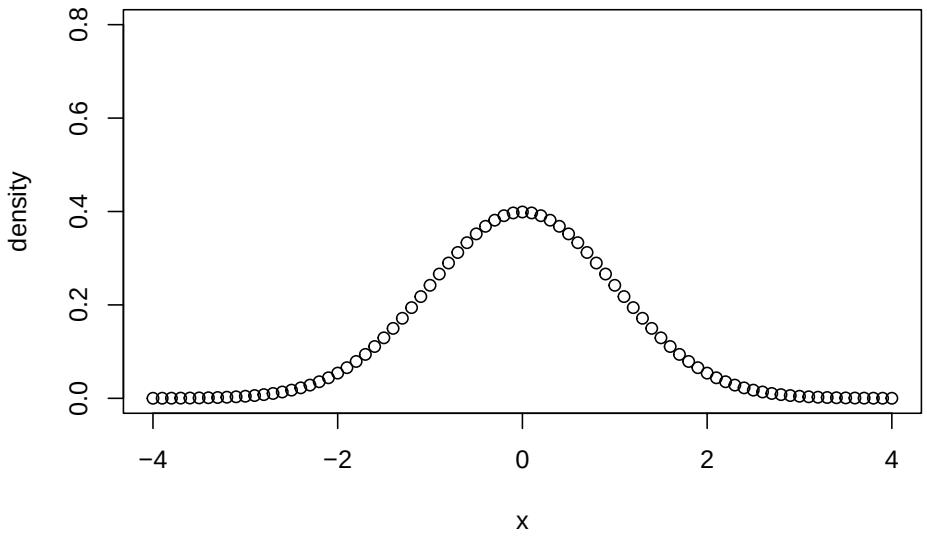
První využití ve vědách o člověku

- Adolph Qutelet
- autor Body Mass Indexu
- 'homme moyen': mean
- aplikace Gaussovy distribuce na "sociální fyziku"
- random effects z experimentu

2.2 Rozdíly mezi Gaussovými křivkami

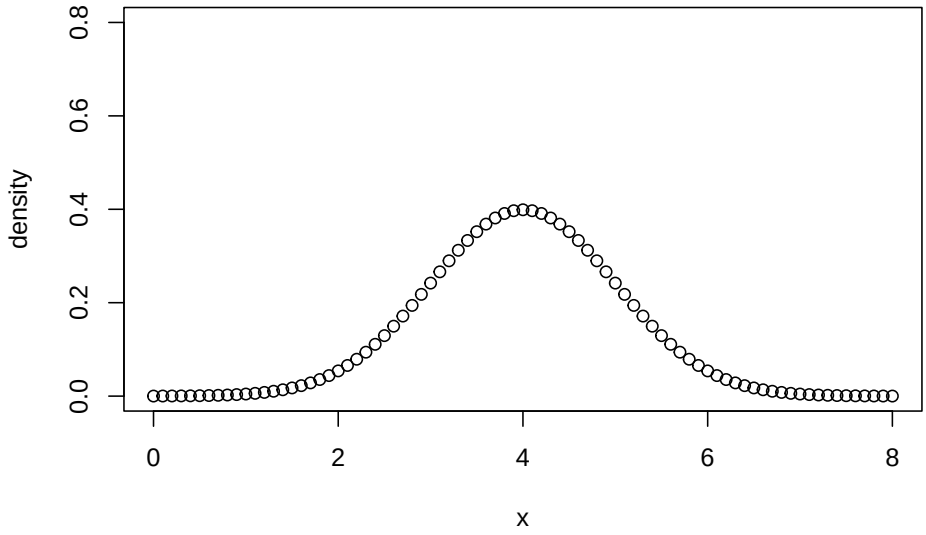
- hustota pravděpodobnosti (density function)

```
x = seq(-4, 4, 0.1)
y = dnorm(x)

plot(x, y, xlab = "x", ylab = "density", ylim = c(0, 0.8))
```



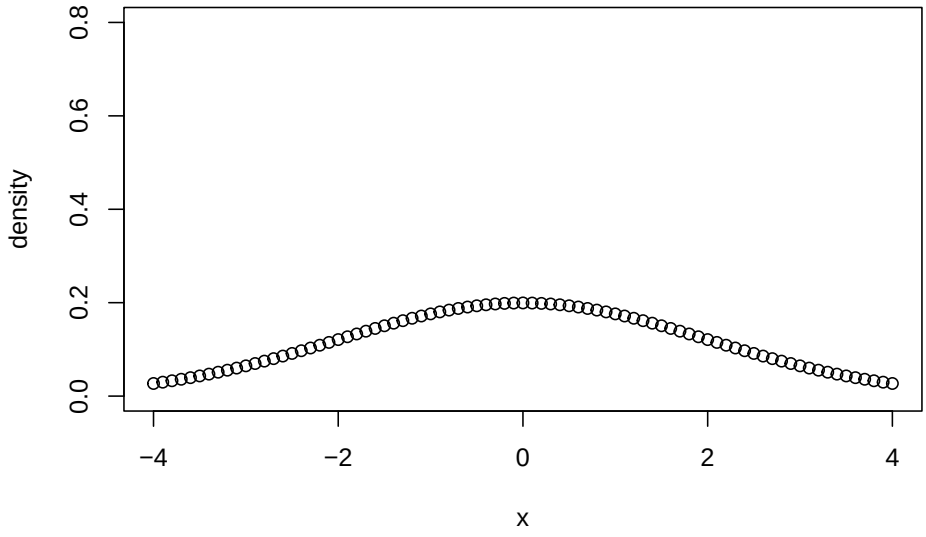
- other mean

```
x = seq(0, 8, 0.1)
y = dnorm(x, mean = 4)

plot(x, y, xlab = "x", ylab = "density", ylim = c(0, 0.8))
```



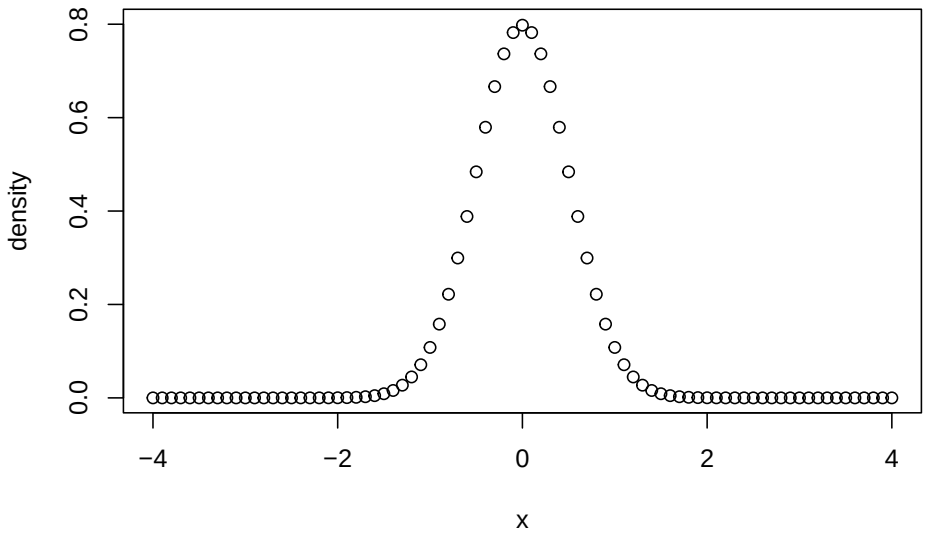
- jiná standardní odchylka

```
x = seq(-4, 4, 0.1)
y = dnorm(x, mean = 0, sd = 2)

plot(x, y, xlab = "x", ylab = "density", ylim = c(0, 0.8))
```



- ještě jiná standardní odchylka

```
x = seq(-4, 4, 0.1)
y = dnorm(x, mean = 0, sd = 0.5)

plot(x, y, xlab = "x", ylab = "density", ylim = c(0, 0.8))
```



2.3 Random effects v experimentech

- odpovědi nemají nutně normální distribuci
- ani na fillery ani na podmínky
- načtení itemů

Nicméně random effects mají Gaussovo rozdělení

- napřed deskriptivní statistika

```
items <- items[-which(items$participant %in% c(32)),]

items <- items %>%
     mutate(condition=replace(condition, condition=="item-méně_než", "fewer")) %>%
     mutate(condition=replace(condition, condition=="item-nanejvýš", "at-most")) %>%
     mutate(condition=replace(condition, condition=="item-ne_víc_než", "no-more")) %>%
     mutate(condition=replace(condition, condition=="item-trochu_méně", "slightly-less
     as.data.frame()

ddply(items, .(condition), summarise, Means = mean(rating1, na.rm=TRUE))
```

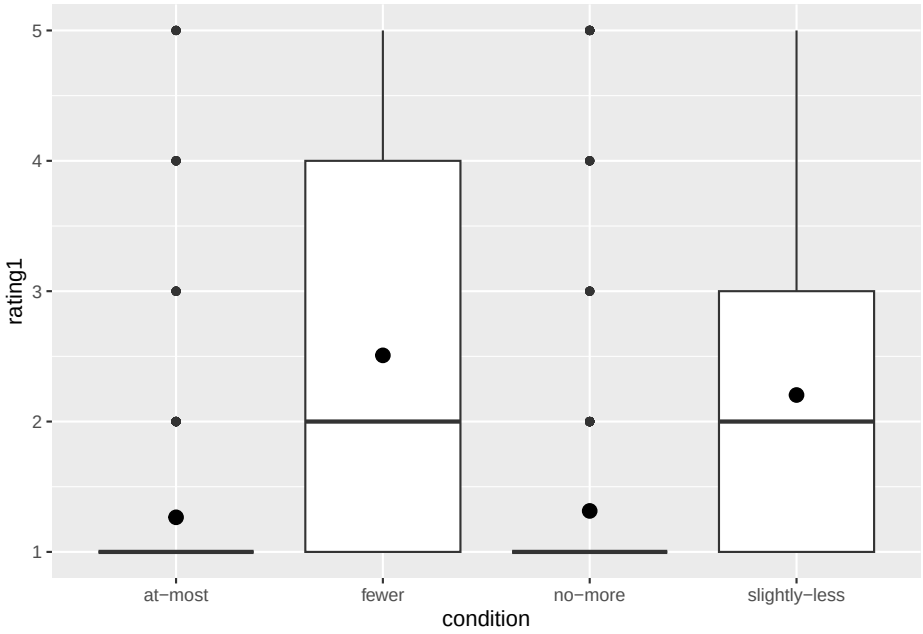
```
##       condition    Means
## 1       at-most 1.265464
## 2         fewer 2.507732
## 3       no-more 1.314433
## 4 slightly-less 2.203608
```

```
ddply(items, .(condition), summarise, Medians = median(rating1,na.rm=TRUE))
```

```
##       condition Medians
## 1       at-most       1
## 2         fewer       2
## 3       no-more       1
## 4 slightly-less       2
```

```
library(ggplot2)
data.to.plot <- items

graph_to_plot <- ggplot(data.to.plot, aes(condition,rating1))
graph_to_plot + geom_boxplot() + stat_summary(fun.y=mean, geom="point", size=3)
```

```
## Warning: The `fun.y` argument of `stat_summary()` is deprecated as of ggplot2 3.3.0.
## i Please use the `fun` argument instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

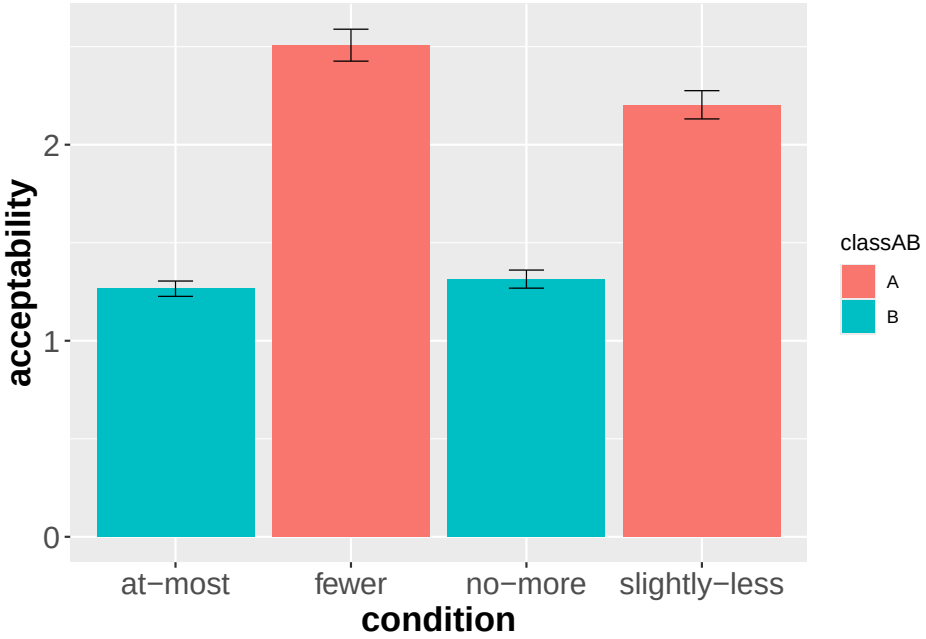



```
items$classAB <- "A"
items$classAB[items$condition == "at-most"] <- "B"
items$classAB[items$condition == "no-more"] <- "B"
```

```
p <- ggplot(items, aes(condition, rating1, fill = classAB)) +
  stat_summary(geom = "bar", fun.y = mean, position = "dodge") +
  stat_summary(geom = "errorbar", fun.data = mean_se, size=.3,
               width=.2,
               position=position_dodge(.9))
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
p  + labs(y = "acceptability") +
   theme(axis.text=element_text(size=15),
           axis.title=element_text(size=17,face="bold"))
```




```
# histograms
```

- vybereme jednu, druhou podmínku
- pak sloučíme

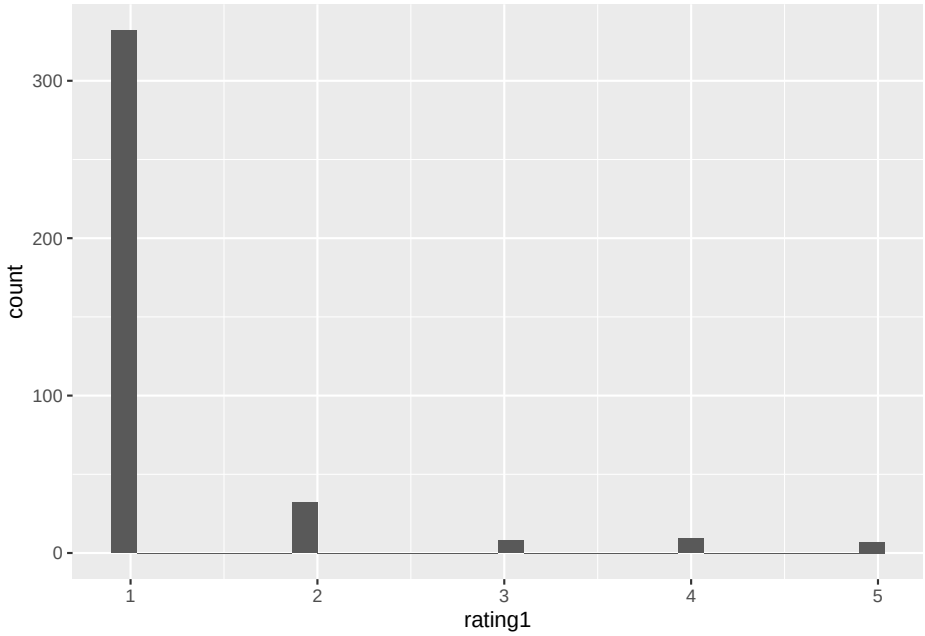
```
# first by condition

at_most <- filter(items, condition=="at-most")
head(at_most)
```

```
##   materials participant item condition position question.order rating1
```

```
## 1     Items           1    2   at-most       21              1       1
## 2     Items           1    6   at-most       32              1       1
## 3     Items           1   10   at-most       28              1       1
## 4     Items           1   14   at-most        9              1       1
## 5     Items           2    1   at-most       18              1       1
## 6     Items           2    5   at-most       26              1       1
##
## 1                                                          Kontext: Aleš čte na obalu piv
## 2                                    Kontext: Aleš si čte pracovní řád firmy, kde je napsáno:
## 3          Kontext: Aleš se dívá na válečný dokument, kde uslyší následující větu: „Ten den mo
## 4                                                 Kontext: Aleš čte článek o jedné exotické
## 5                                          Kontext: Aleš čte na obalu čokolády následující vět
## 6 Kontext: Aleš si čte studijní řád místní univerzity, kde vidí následující větu: "Diplomová p
##   classAB
## 1       B
## 2       B
## 3       B
## 4       B
## 5       B
## 6       B
```

```
write.csv(at_most, "at_most.csv")

ggplot(at_most, aes(x=rating1)) +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



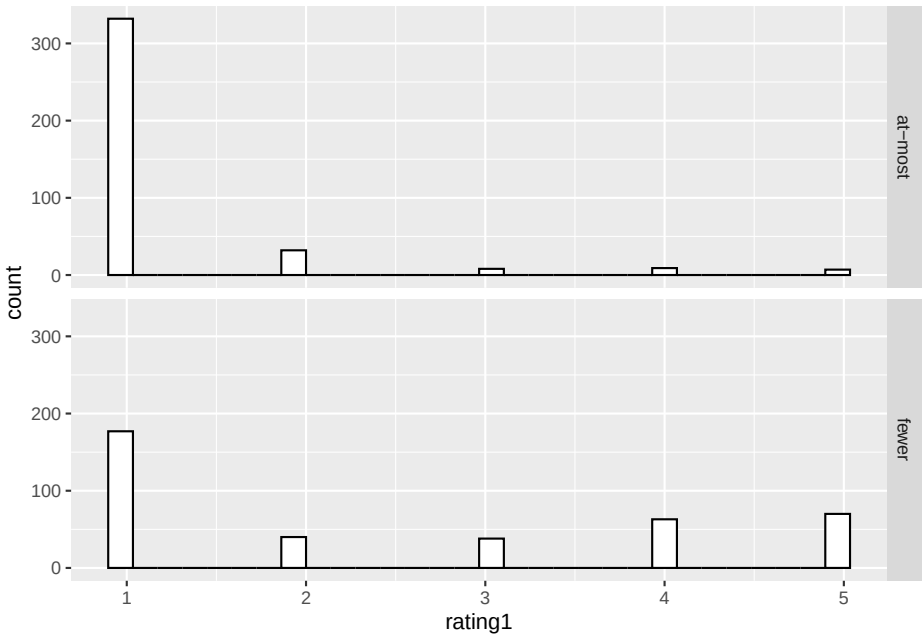

```
# two conditions

at_most_no_more <- filter(items, condition=="at-most" | condition == "fewer")
head(at_most_no_more)
```

```
##   materials participant item condition position question.order rating1
## 1     Items           1    1     fewer       17              1       4
## 2     Items           1    2   at-most       21              1       1
## 3     Items           1    5     fewer        5              1       5
## 4     Items           1    6   at-most       32              1       1
## 5     Items           1    9     fewer       34              1       2
## 6     Items           1   10   at-most       28              1       1
##
## 1                                        Kontext: Aleš čte na obalu čokolády násle
## 2                                                  Kontext: Aleš čte na
## 3 Kontext: Aleš si čte studijní řád místní univerzity, kde vidí následující větu: "
## 4                                  Kontext: Aleš si čte pracovní řád firmy, kde
## 5             Kontext: Aleš si čte zprávu valné hromady firmy EB, kde stojí následu
## 6          Kontext: Aleš se dívá na válečný dokument, kde uslyší následující větu:
##   classAB
## 1       A
## 2       B
## 3       A
## 4       B
## 5       A
```

```
## 6       B
```

```
ggplot(at_most_no_more, aes(x = rating1)) +
  geom_histogram(fill = "white", colour = "black") +
  facet_grid(condition ~ .)
```

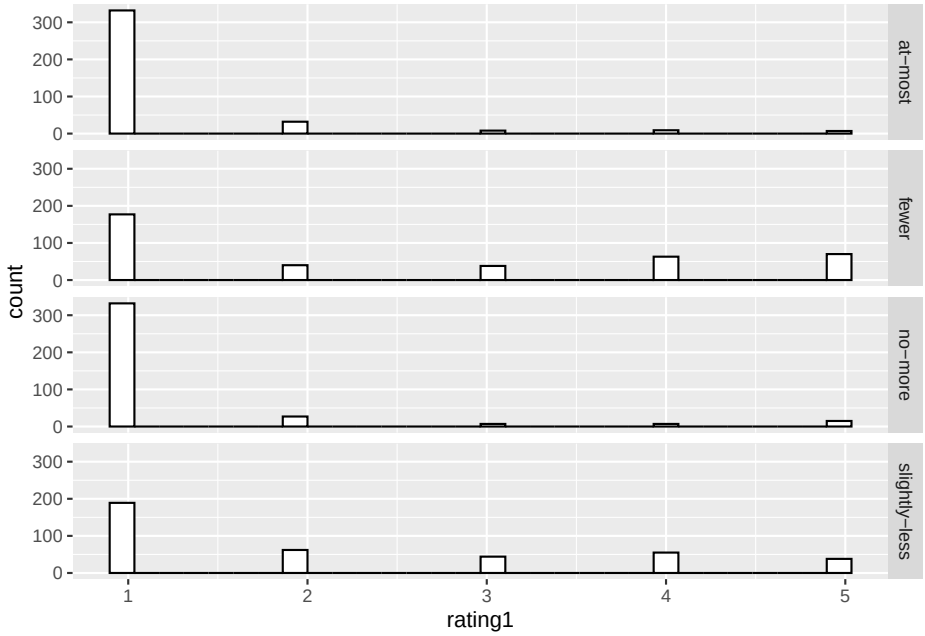
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



- jasně ne-Gaussovské (normální rozložení)

```
# all conditions
# https://r-graphics.org/recipe-distribution-multi-hist

ggplot(items, aes(x = rating1)) +
  geom_histogram(fill = "white", colour = "black") +
  facet_grid(condition ~ .)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



- testy

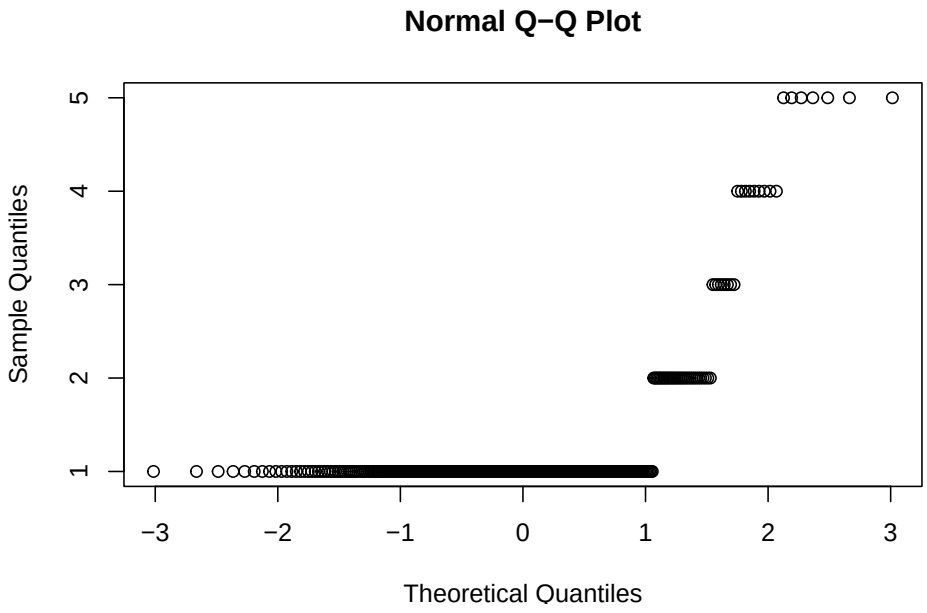
```
# tests for normality

shapiro.test(filter(at_most_no_more, condition == "at-most")$rating1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  filter(at_most_no_more, condition == "at-most")$rating1
## W = 0.39231, p-value < 2.2e-16
```

```
# filter(at_most_no_more, condition == "at-most")$rating1
# filter(at_most_no_more, condition == "at-most")

# not normal distribution
# Wilcox test

wilcox.test(filter(at_most_no_more, condition == "at-most")$rating1, +
              filter(at_most_no_more, condition == "fewer")$rating1)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  filter(at_most_no_more, condition == "at-most")$rating1 and +filter(at_most_
## W = 42624, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

```
# quantile graphs

qqnorm(filter(at_most_no_more, condition == "at-most")$rating1)
```



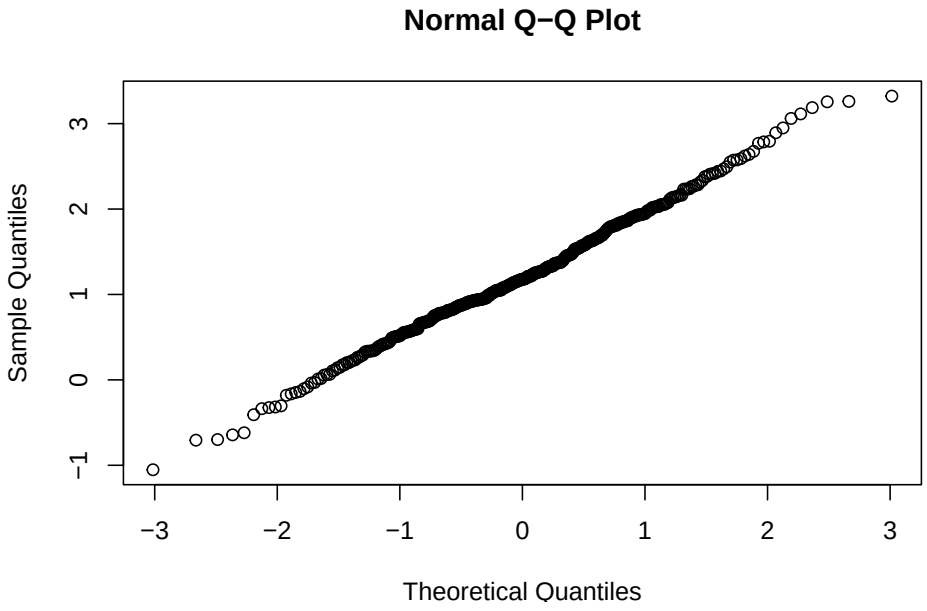

```
mean(filter(at_most_no_more, condition == "at-most")$rating1)
```

```
## [1] 1.265464
```

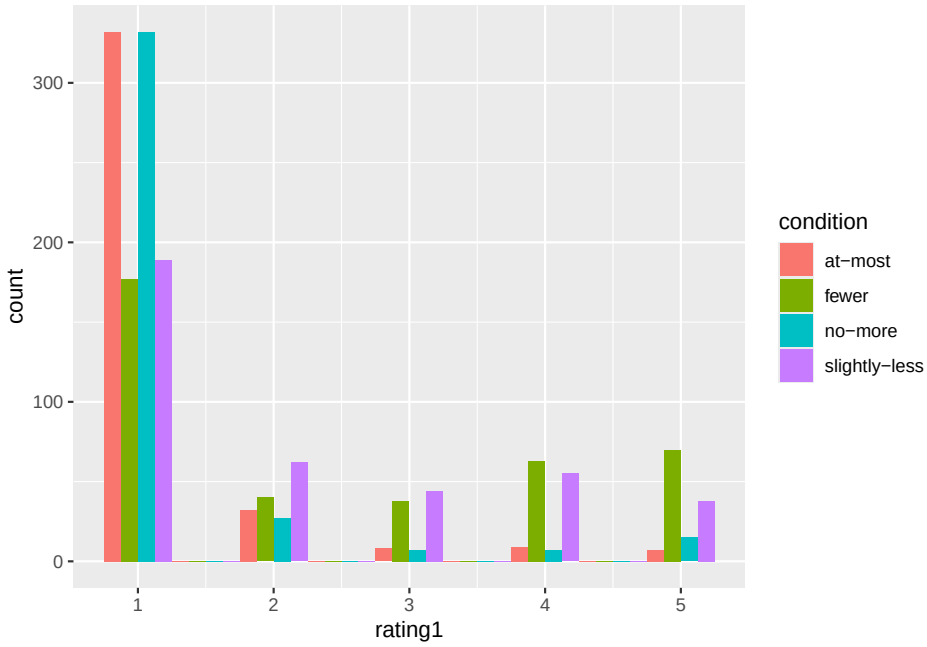
```
sd(filter(at_most_no_more, condition == "at-most")$rating1)
```

```
## [1] 0.7703438
```

```
# vs. Gauss distribution

qqnorm(rnorm(length(filter(at_most_no_more, condition == "at-most")$rating1), 1.27, 0.77))
```




```
ggplot(items, aes(x = rating1, fill = condition)) +
  geom_histogram(position = "dodge", alpha = 1, binwidth = 0.5)
```




```
summary(items)
```

```
##    materials          participant          item       condition
```

```
##  Length:1552        Min.   : 1.00   1      : 97   Length:1552
##  Class :character   1st Qu.:25.00   10     : 97   Class :character
##  Mode  :character   Median :50.00   11     : 97   Mode  :character
##                     Mean   :49.68   12     : 97
##                     3rd Qu.:74.00   13     : 97
##                     Max.   :98.00   14     : 97
##                                     (Other):970
##     position     question.order    rating1        content
##  Min.   : 3.00   Min.   :1      Min.   :1.000   Length:1552
##  1st Qu.:10.75   1st Qu.:1      1st Qu.:1.000   Class :character
##  Median :18.50   Median :1      Median :1.000   Mode  :character
##  Mean   :18.41   Mean   :1      Mean   :1.823
##  3rd Qu.:26.25   3rd Qu.:1      3rd Qu.:2.000
##  Max.   :34.00   Max.   :1      Max.   :5.000
##
##    classAB
##  Length:1552
##  Class :character
##  Mode  :character
##
##
##
##
```

```
# linear model

library(lmerTest)
```

```
## Loading required package: lme4
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'lmerTest'
```

```
## The following object is masked from 'package:lme4':
##
##     lmer
```

```
## The following object is masked from 'package:stats':
##
##     step
```

```
items$condition <- as.factor(items$condition)

items$condition <- relevel(items$condition, ref="at-most")

m1 <- lmer(as.numeric(rating1) ~ condition  + (1|participant) + (1|item), data=items)
```

```
summary(m1)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: as.numeric(rating1) ~ condition + (1 | participant) + (1 | item)
##    Data: items
##
## REML criterion at convergence: 4879.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.1573 -0.6556 -0.1995  0.4451  3.6016
##
## Random effects:
##  Groups      Name        Variance Std.Dev.
##  participant (Intercept) 0.1438   0.3793
##  item        (Intercept) 0.1341   0.3662
##  Residual                1.2334   1.1106
## Number of obs: 1552, groups:  participant, 97; item, 16
##
## Fixed effects:
##                           Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)             1.276e+00  1.142e-01 3.003e+01  11.176  3.2e-12 ***
## conditionfewer          1.231e+00  7.980e-02 1.437e+03  15.428  < 2e-16 ***
## conditionno-more        3.811e-02  7.980e-02 1.437e+03   0.478    0.633
## conditionslightly-less 9.162e-01  7.983e-02 1.438e+03  11.477  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) cndtnf cndtnn-
## conditinfwr -0.349
## conditnn-mr -0.349  0.500
## cndtnslght- -0.349  0.500  0.500
```

```
library(emmeans)
```

A konečně random effects

```
# random effects

m1.ranef <- ranef(m1)

m1.ranef
```

```
## $participant
##     (Intercept)
```

```
## 1  -0.047403661
## 2  -0.047403661
## 3   0.115362893
## 4  -0.372936769
## 5   0.033979616
## 6   0.033979616
## 7   0.725737470
## 8  -0.169478577
## 9   0.440896000
## 10 -0.535703323
## 11  0.115362893
## 12 -0.535703323
## 13 -0.413628407
## 14  0.237437808
## 15 -0.535703323
## 16  0.033979616
## 17  0.440896000
## 18 -0.088095300
## 19  0.033979616
## 20  0.115362893
## 21 -0.413628407
## 22 -0.088095300
## 23  0.156054531
## 24  0.196746169
## 25 -0.495011684
## 26  0.847812385
## 27 -0.210170215
## 28 -0.047403661
## 29 -0.047403661
## 30  0.156054531
## 31 -0.210170215
## 33 -0.535703323
## 34 -0.495011684
## 35  0.278129446
## 36 -0.047403661
## 37 -0.047403661
## 38  0.522279277
## 39 -0.006712023
## 40  0.033979616
## 41  0.074671254
## 42  0.237437808
## 43 -0.372936769
## 44  0.318821085
## 45 -0.291553492
## 46  0.156054531
## 47 -0.454320046
```

```
## 48 -0.210170215
## 49  0.074671254
## 50  0.115362893
## 51  0.278129446
## 52 -0.250861854
## 53  0.440896000
## 54 -0.006712023
## 55 -0.413628407
## 56  0.400204362
## 57 -0.210170215
## 58  0.074671254
## 59  0.318821085
## 60 -0.047403661
## 61 -0.047403661
## 62  0.278129446
## 63 -0.128786938
## 64  0.359512723
## 65  0.074671254
## 66  0.074671254
## 67 -0.088095300
## 68 -0.047403661
## 69 -0.250861854
## 70 -0.128786938
## 71  0.522279277
## 72 -0.169478577
## 73  0.603662554
## 74  0.400204362
## 75  0.156054531
## 76 -0.047403661
## 77  0.400204362
## 78 -0.006712023
## 79 -0.210170215
## 80 -0.006712023
## 81 -0.535703323
## 82 -0.372936769
## 83 -0.047403661
## 84  0.400204362
## 85 -0.495011684
## 86 -0.088095300
## 87 -0.210170215
## 88  0.278129446
## 89  0.115362893
## 90 -0.250861854
## 91 -0.413628407
## 92 -0.210170215
## 93  0.278129446
```

```
## 94 -0.088095300
## 95  0.033979616
## 96 -0.088095300
## 97  0.562970916
## 98  0.033979616
##
## $item
##    (Intercept)
## 1   0.15475432
## 10 -0.07879328
## 11  0.19274689
## 12 -0.56296464
## 13 -0.51379691
## 14  0.50501203
## 15 -0.05207469
## 16 -0.28989442
## 2   0.10953101
## 3  -0.31572869
## 4  -0.04507284
## 5   0.69147855
## 6   0.31668773
## 7   0.18333068
## 8   0.04908930
## 9  -0.34430505
##
## with conditional variances for "participant" "item"
```

```
m1.ranef$participant$`(Intercept)`
```

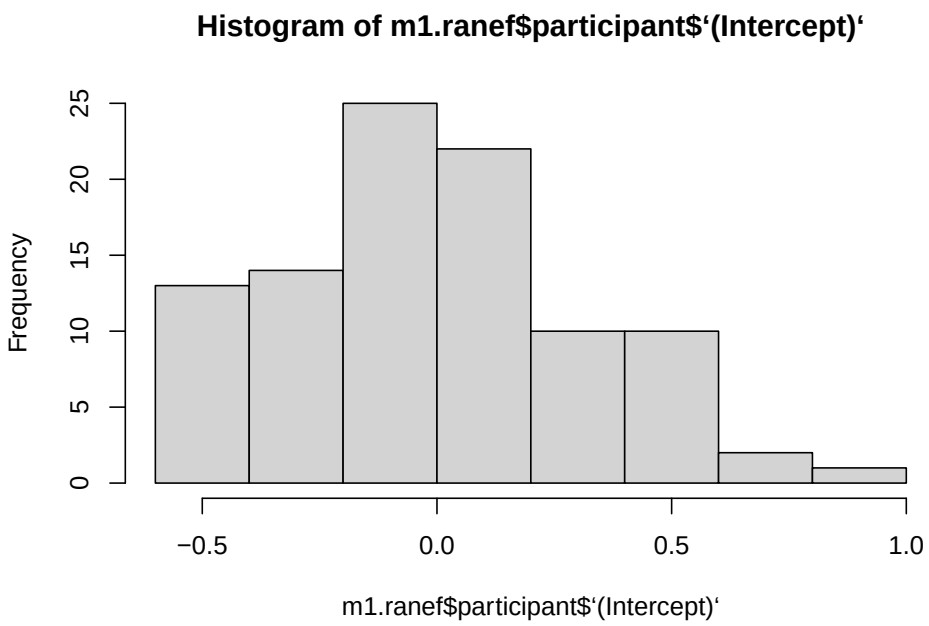
```
##  [1] -0.047403661 -0.047403661  0.115362893 -0.372936769  0.033979616
##  [6]  0.033979616  0.725737470 -0.169478577  0.440896000 -0.535703323
## [11]  0.115362893 -0.535703323 -0.413628407  0.237437808 -0.535703323
## [16]  0.033979616  0.440896000 -0.088095300  0.033979616  0.115362893
## [21] -0.413628407 -0.088095300  0.156054531  0.196746169 -0.495011684
## [26]  0.847812385 -0.210170215 -0.047403661 -0.047403661  0.156054531
## [31] -0.210170215 -0.535703323 -0.495011684  0.278129446 -0.047403661
## [36] -0.047403661  0.522279277 -0.006712023  0.033979616  0.074671254
## [41]  0.237437808 -0.372936769  0.318821085 -0.291553492  0.156054531
## [46] -0.454320046 -0.210170215  0.074671254  0.115362893  0.278129446
## [51] -0.250861854  0.440896000 -0.006712023 -0.413628407  0.400204362
## [56] -0.210170215  0.074671254  0.318821085 -0.047403661 -0.047403661
## [61]  0.278129446 -0.128786938  0.359512723  0.074671254  0.074671254
## [66] -0.088095300 -0.047403661 -0.250861854 -0.128786938  0.522279277
## [71] -0.169478577  0.603662554  0.400204362  0.156054531 -0.047403661
## [76]  0.400204362 -0.006712023 -0.210170215 -0.006712023 -0.535703323
## [81] -0.372936769 -0.047403661  0.400204362 -0.495011684 -0.088095300
```

```
## [86] -0.210170215  0.278129446  0.115362893 -0.250861854 -0.413628407
## [91] -0.210170215  0.278129446 -0.088095300  0.033979616 -0.088095300
## [96]  0.562970916  0.033979616
```

```
summary(m1.ranef$participant$`(Intercept)`)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.535703 -0.210170 -0.006712  0.000000  0.156055  0.847812
```

```
hist(m1.ranef$participant$`(Intercept)`)
```




```
shapiro.test(m1.ranef$participant$`(Intercept)`)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  m1.ranef$participant$`(Intercept)`
## W = 0.97906, p-value = 0.1239
```

```
# normal enough

mean(m1.ranef$participant$`(Intercept)`)
```

```
## [1] 9.104604e-16
```

```
sd(m1.ranef$participant$`(Intercept)`)
```

```
## [1] 0.3060158
```

```
shapiro.test(rnorm(100, mean = 5, sd = 3))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rnorm(100, mean = 5, sd = 3)
## W = 0.98962, p-value = 0.6337
```

A nakonec pro items ranef

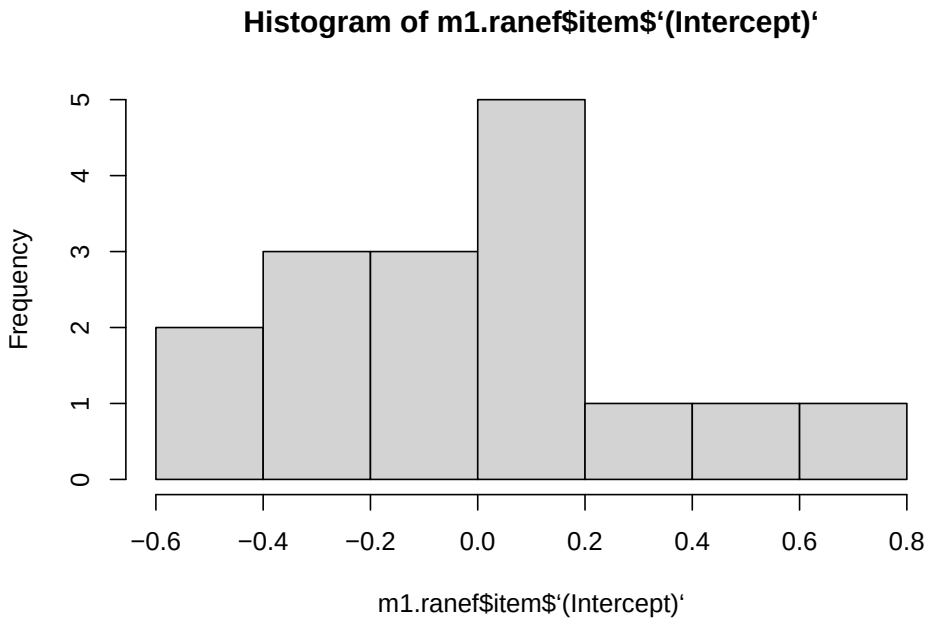
```
m1.ranef$item$`(Intercept)`
```

```
##  [1]  0.15475432 -0.07879328  0.19274689 -0.56296464 -0.51379691  0.50501203
##  [7] -0.05207469 -0.28989442  0.10953101 -0.31572869 -0.04507284  0.69147855
## [13]  0.31668773  0.18333068  0.04908930 -0.34430505
```

```
summary(m1.ranef$item$`(Intercept)`)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.562965 -0.296353  0.002008  0.000000  0.185685  0.691479
```

```
hist(m1.ranef$item$`(Intercept)`)
```




```
shapiro.test(m1.ranef$item$`(Intercept)`)
```

```
##
##  Shapiro-Wilk normality test
##
```

```
## data:  m1.ranef$item$`(Intercept)`
## W = 0.97494, p-value = 0.9109
```

```
# normal enough

mean(m1.ranef$item$`(Intercept)`)
```

```
## [1] 5.646915e-14
```

```
sd(m1.ranef$item$`(Intercept)`)
```

```
## [1] 0.3499262
```

```
shapiro.test(rnorm(100, mean = 5, sd = 3))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rnorm(100, mean = 5, sd = 3)
## W = 0.991, p-value = 0.7448
```

```
#t.test(m1.ranef$item$`(Intercept)`,m1.ranef$participant$`(Intercept)`)
```

Chapter 3

Testování hypotéz

3.1 Historie

- t.test
- t-distribuce: typ normální distribuce s relativně malým počtem trials ve vzorku, známým mean a neznámou (v populaci) SD
- autor: William Sealy Gosset, pseudonym Student
- historie (Guinness brewery)
- testování hypotéz:

1. Nulová hypotéza (data jsou čistě náhodná)
2. Experiment a testování toho, jak nepravděpodobný by byl výsledek měření, pokud by byla pravdivá nulová hypotéza
3. Zamítnutí nulové hypotézy, pokud je p pod úrovní "threshold" – standardně 0.05

3.2 Příklad s mincí

```
log_coin <- c(TRUE, FALSE)

flips <- sample(log_coin, size = 100, replace = TRUE, prob = c(0.2, 0.8))

str(flips)
```

```
##  logi [1:100] TRUE TRUE FALSE FALSE FALSE FALSE ...
```

```
summary(flips)
```

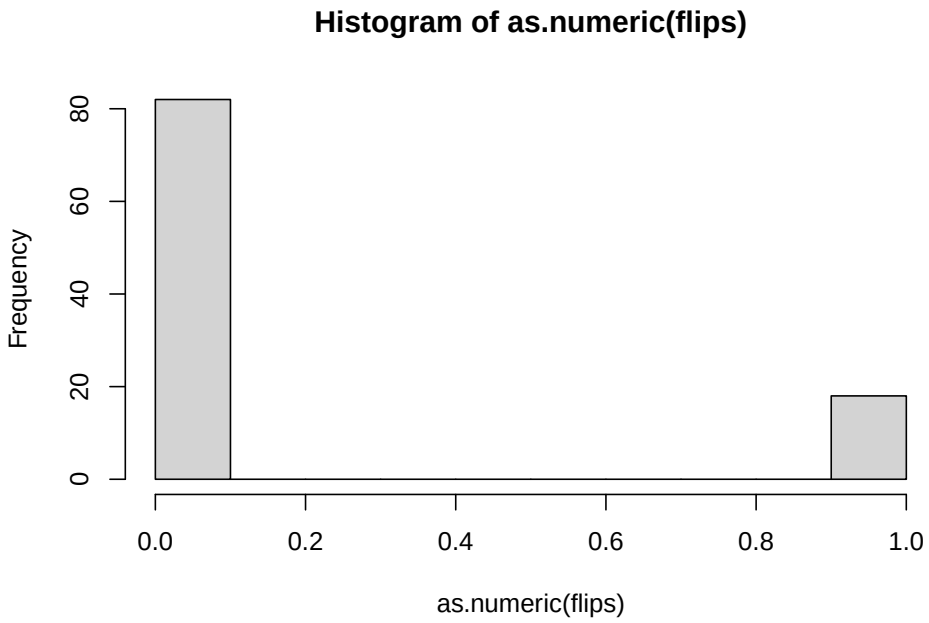
```
##    Mode   FALSE    TRUE
```

```
## logical      82      18
```

```
t.test(flips, mu=0.5)
```

```
##
##  One Sample t-test
##
## data:  flips
## t = -8.2875, df = 99, p-value = 5.789e-13
## alternative hypothesis: true mean is not equal to 0.5
## 95 percent confidence interval:
##  0.1033848 0.2566152
## sample estimates:
## mean of x
##      0.18
```

```
hist(as.numeric(flips))
```



- jasně neférová mince
- menší rozdíl: 60/40

```
log_coin <- c(TRUE, FALSE)

flips <- sample(log_coin, size = 100, replace = TRUE, prob = c(0.4, 0.6))

str(flips)
```

```
##  logi [1:100] FALSE FALSE FALSE TRUE TRUE TRUE ...
```

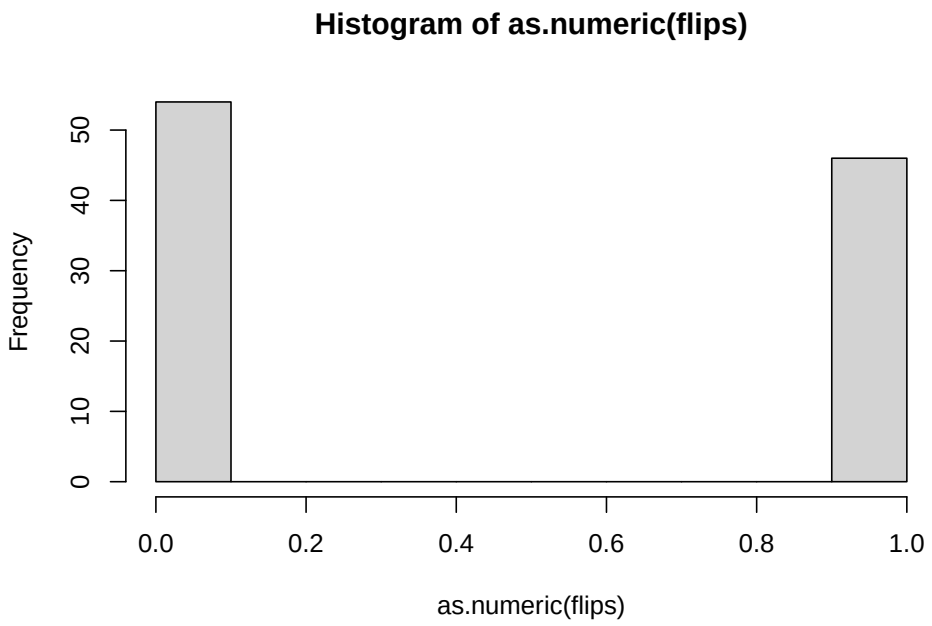
```
summary(flips)
```

```
##    Mode   FALSE    TRUE
## logical      54      46
```

```
t.test(flips, mu=0.5)
```

```
##
##  One Sample t-test
##
## data:  flips
## t = -0.79855, df = 99, p-value = 0.4265
## alternative hypothesis: true mean is not equal to 0.5
## 95 percent confidence interval:
##  0.3606089 0.5593911
## sample estimates:
## mean of x
##      0.46
```

```
hist(as.numeric(flips))
```



- někdy ano, někdy, v závislosti na náhodném generování - 53/47:

```
log_coin <- c(TRUE, FALSE)

flips <- sample(log_coin, size = 100, replace = TRUE, prob = c(0.47, 0.53))

str(flips)
```

```
##  logi [1:100] FALSE TRUE TRUE FALSE FALSE TRUE ...
```

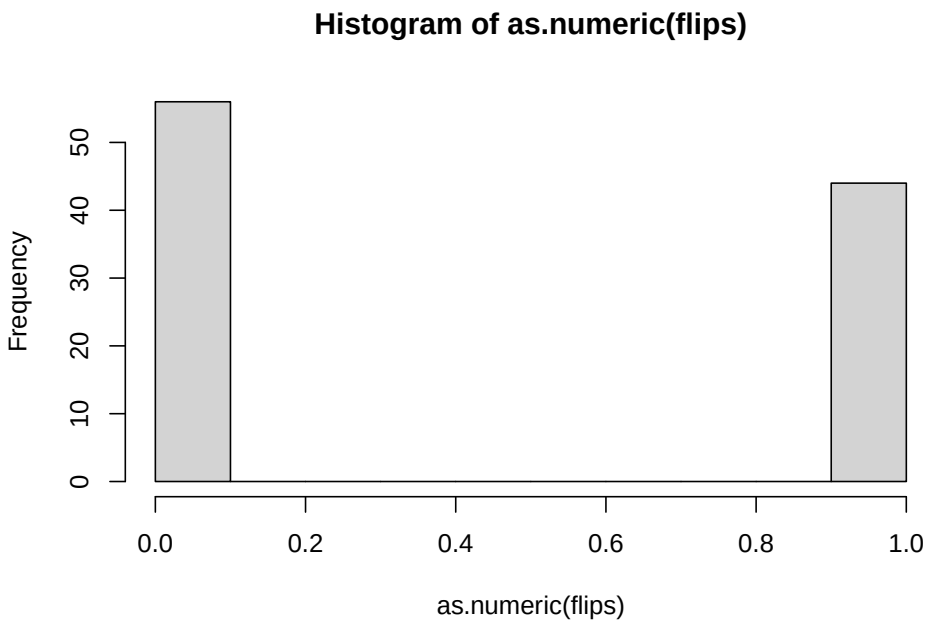
```
summary(flips)
```

```
##    Mode   FALSE    TRUE
## logical      56      44
```

```
t.test(flips, mu=0.5)
```

```
##
##  One Sample t-test
##
## data:  flips
## t = -1.2027, df = 99, p-value = 0.232
## alternative hypothesis: true mean is not equal to 0.5
## 95 percent confidence interval:
##  0.3410099 0.5389901
## sample estimates:
## mean of x
##      0.44
```

```
hist(as.numeric(flips))
```



- ne
- nulová hypotéza: mince je férová
- až v posledním případě byla nulová hypotéza potvrzena
- v přechozích byla s p … odmítnuta – tj. p říká s jakou pravděpodobností by platila nulová hypotéza

3.3 Lingvistická aplikace

- Baayen (2008) : chap. 4.2
- začíná už v 4.1: testy na normální rozložení
- quantile-quantile grafy
- Shapiro-Wilk test (s. 73)
- 4.1.2: rozdíl mezi populací a mean
- s. 75

```
library(languageR)

t.test(durationsOnt$DurationPrefixNasal, mu = 0.053)
```

```
##
##  One Sample t-test
##
## data:  durationsOnt$DurationPrefixNasal
## t = -1.5038, df = 101, p-value = 0.1358
## alternative hypothesis: true mean is not equal to 0.053
## 95 percent confidence interval:
##  0.04561370 0.05401646
## sample estimates:
##  mean of x
## 0.04981508
```

```
data("durationsOnt")

head(durationsOnt$DurationPrefixNasal)
```

```
## [1] 0.043905 0.065099 0.092454 0.064830 0.097262 0.085662
```

```
mean(durationsOnt$DurationPrefixNasal)
```

```
## [1] 0.04981508
```

```
library(languageR)

simplex = ratings[ratings$Complex == "simplex", ]
freqAnimals = simplex[simplex$Class == "animal", ]$Frequency
freqPlants = simplex[simplex$Class == "plant", ]$Frequency

t.test(freqAnimals, freqPlants)
```

```
##
##  Welch Two Sample t-test
##
```

```
## data:  freqAnimals and freqPlants
## t = 2.674, df = 57.545, p-value = 0.009739
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.193183 1.344315
## sample estimates:
## mean of x mean of y
##  5.208494  4.439745
```

- data from chap. 2.2

3.4 Zpět k random efektům

- napřed znovu načteme data

A postupně se propracujeme k random-effectům (normální rozdělení)

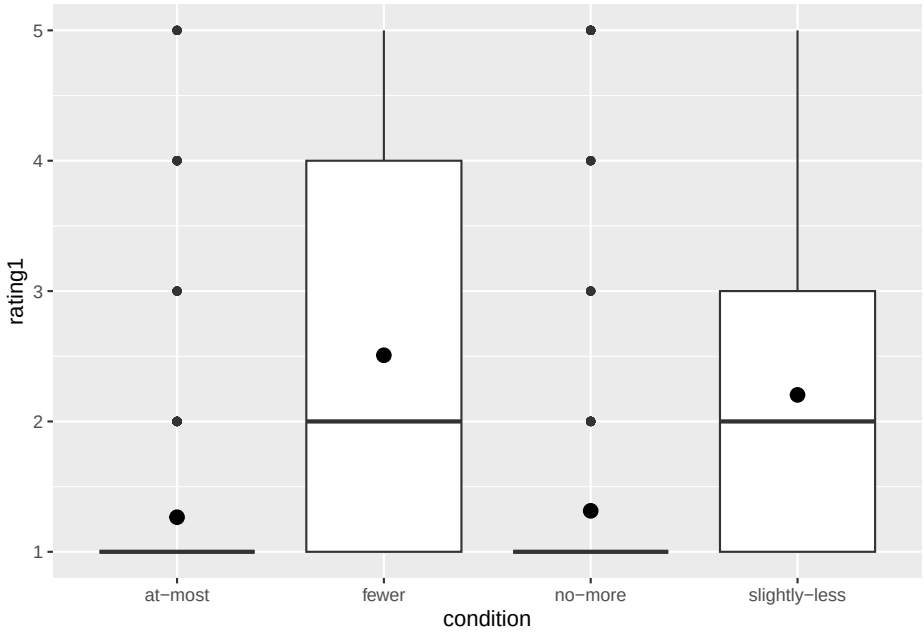
```
items <- items[-which(items$participant %in% c(32)),]

items <- items %>%
     mutate(condition=replace(condition, condition=="item-méně_než", "fewer")) %>%
     mutate(condition=replace(condition, condition=="item-nanejvýš", "at-most")) %>%
     mutate(condition=replace(condition, condition=="item-ne_víc_než", "no-more")) %>%
     mutate(condition=replace(condition, condition=="item-trochu_méně", "slightly-less
     as.data.frame()

ddply(items, .(condition), summarise, Means = mean(rating1, na.rm=TRUE))
```

```
##       condition    Means
## 1       at-most 1.265464
## 2         fewer 2.507732
## 3       no-more 1.314433
## 4 slightly-less 2.203608
```

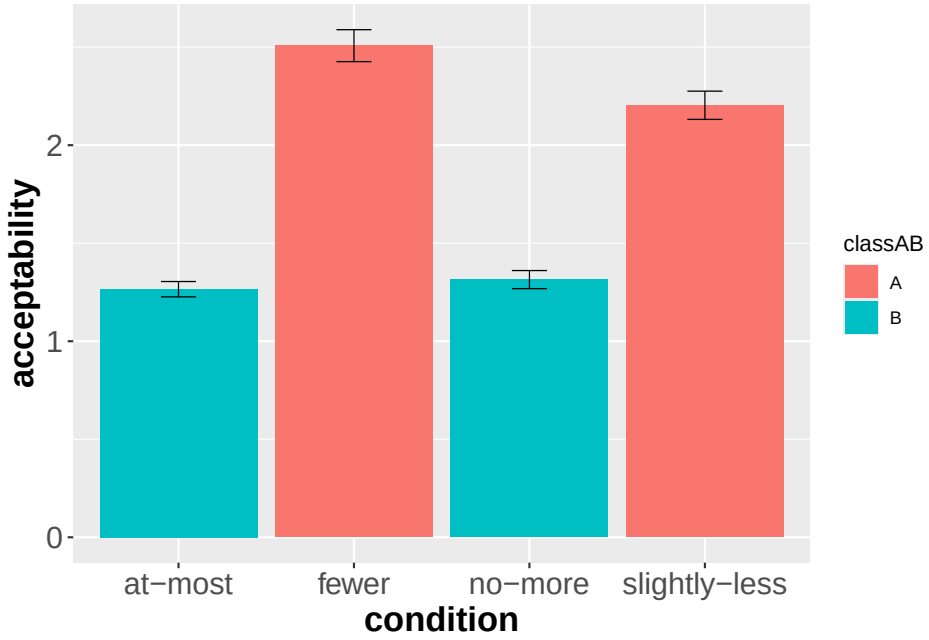
```
ddply(items, .(condition), summarise, Medians = median(rating1,na.rm=TRUE))
```

```
##       condition Medians
## 1       at-most       1
## 2         fewer       2
## 3       no-more       1
## 4 slightly-less       2
```

```
library(ggplot2)
data.to.plot <- items

graph_to_plot <- ggplot(data.to.plot, aes(condition,rating1))
graph_to_plot + geom_boxplot() + stat_summary(fun.y=mean, geom="point", size=3)
```




```
items$classAB <- "A"
items$classAB[items$condition == "at-most"] <- "B"
items$classAB[items$condition == "no-more"] <- "B"



p <- ggplot(items, aes(condition, rating1, fill = classAB)) +
  stat_summary(geom = "bar", fun.y = mean, position = "dodge") +
  stat_summary(geom = "errorbar", fun.data = mean_se, size=.3,
               width=.2,
               position=position_dodge(.9))
p  + labs(y = "acceptability") +
   theme(axis.text=element_text(size=15),
           axis.title=element_text(size=17,face="bold"))
```



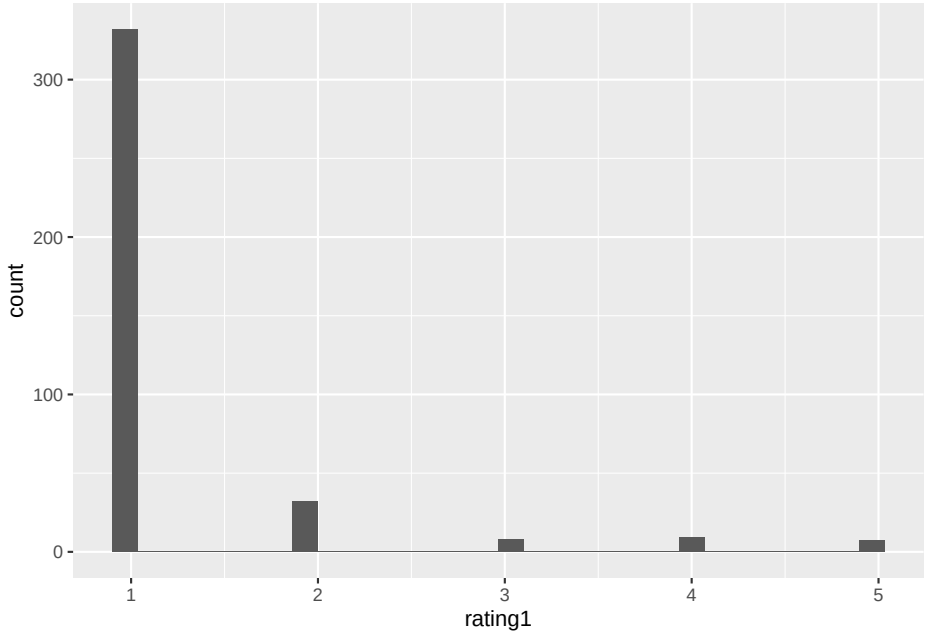

```
# histograms

# first by condition

at_most <- filter(items, condition=="at-most")
head(at_most)
```

```
##   materials participant item condition position question.order rating1
## 1     Items           1    2   at-most       21              1       1
## 2     Items           1    6   at-most       32              1       1
## 3     Items           1   10   at-most       28              1       1
## 4     Items           1   14   at-most        9              1       1
## 5     Items           2    1   at-most       18              1       1
## 6     Items           2    5   at-most       26              1       1
##
## 1                                                                  Kontext: Aleš čte na
## 2                                        Kontext: Aleš si čte pracovní řád firmy, kde
## 3             Kontext: Aleš se dívá na válečný dokument, kde uslyší následující větu:
## 4                                                    Kontext: Aleš čte článek o jedn
## 5                                          Kontext: Aleš čte na obalu čokolády násl
## 6 Kontext: Aleš si čte studijní řád místní univerzity, kde vidí následující větu: "
##   classAB
## 1       B
## 2       B
## 3       B
```

```
## 4       B
## 5       B
## 6       B
```

```
write.csv(at_most, "at_most.csv")

ggplot(at_most, aes(x=rating1)) +
  geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



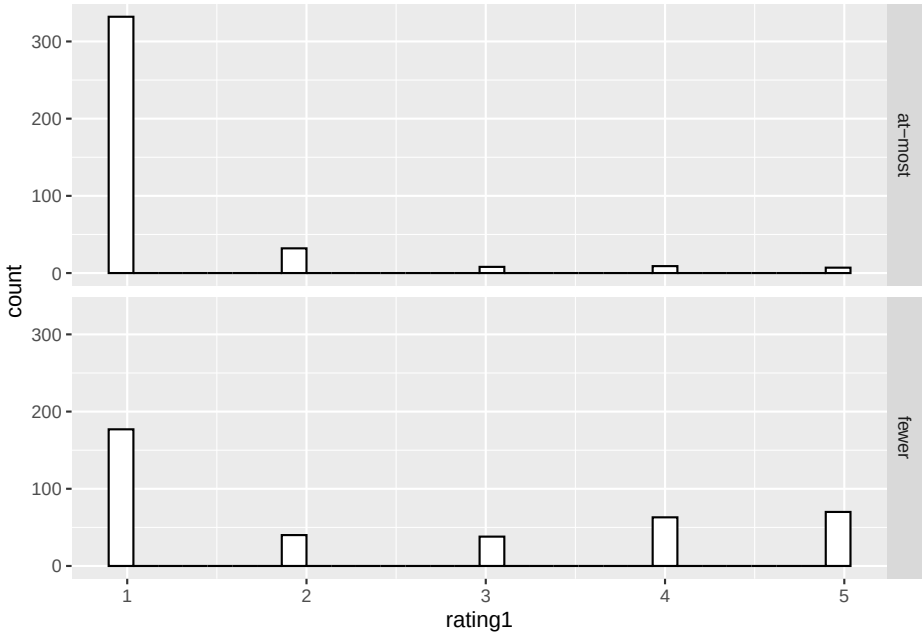

```
# two conditions

at_most_no_more <- filter(items, condition=="at-most" | condition == "fewer")
head(at_most_no_more)
```

```
##   materials participant item condition position question.order rating1
## 1     Items           1    1     fewer       17              1       4
## 2     Items           1    2   at-most       21              1       1
## 3     Items           1    5     fewer        5              1       5
## 4     Items           1    6   at-most       32              1       1
## 5     Items           1    9     fewer       34              1       2
## 6     Items           1   10   at-most       28              1       1
##
## 1                                        Kontext: Aleš čte na obalu čokolády následující vět
## 2                                                    Kontext: Aleš čte na obalu piv
## 3 Kontext: Aleš si čte studijní řád místní univerzity, kde vidí následující větu: "Diplomová p
```

```
## 4                                    Kontext: Aleš si čte pracovní řád firmy, kde
## 5              Kontext: Aleš si čte zprávu valné hromady firmy EB, kde stojí následu
## 6          Kontext: Aleš se dívá na válečný dokument, kde uslyší následující větu:
##   classAB
## 1       A
## 2       B
## 3       A
## 4       B
## 5       A
## 6       B
```

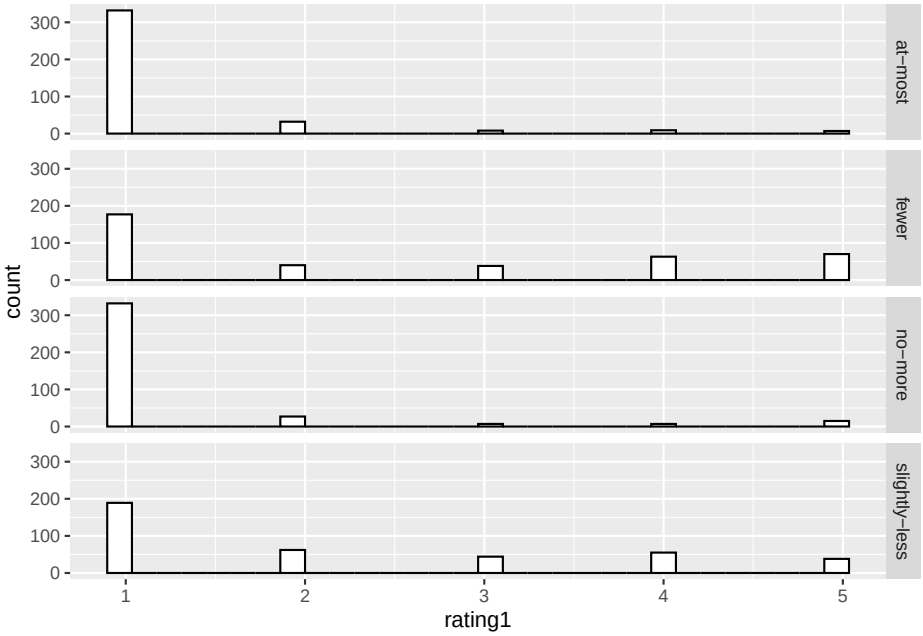
```
ggplot(at_most_no_more, aes(x = rating1)) +
  geom_histogram(fill = "white", colour = "black") +
  facet_grid(condition ~ .)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# all conditions
# https://r-graphics.org/recipe-distribution-multi-hist

ggplot(items, aes(x = rating1)) +
  geom_histogram(fill = "white", colour = "black") +
  facet_grid(condition ~ .)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



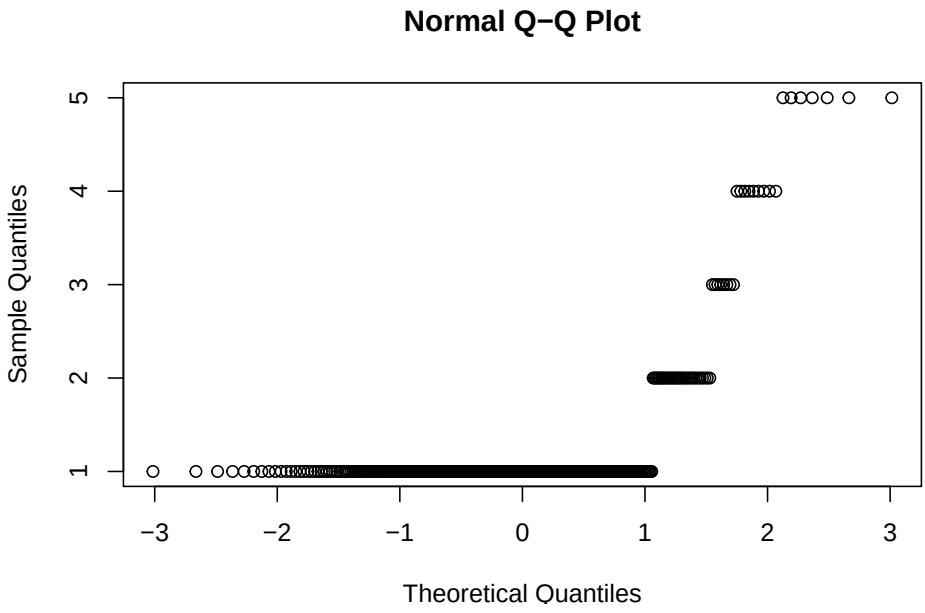

```
# tests for normality

shapiro.test(filter(at_most_no_more, condition == "at-most")$rating1)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  filter(at_most_no_more, condition == "at-most")$rating1
## W = 0.39231, p-value < 2.2e-16
```

```
# filter(at_most_no_more, condition == "at-most")$rating1
# filter(at_most_no_more, condition == "at-most")

# not normal distribution
# Wilcox test

wilcox.test(filter(at_most_no_more, condition == "at-most")$rating1, +
              filter(at_most_no_more, condition == "fewer")$rating1)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  filter(at_most_no_more, condition == "at-most")$rating1 and +filter(at_most_no_more, co
## W = 42624, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

```
# quantile graphs

qqnorm(filter(at_most_no_more, condition == "at-most")$rating1)
```



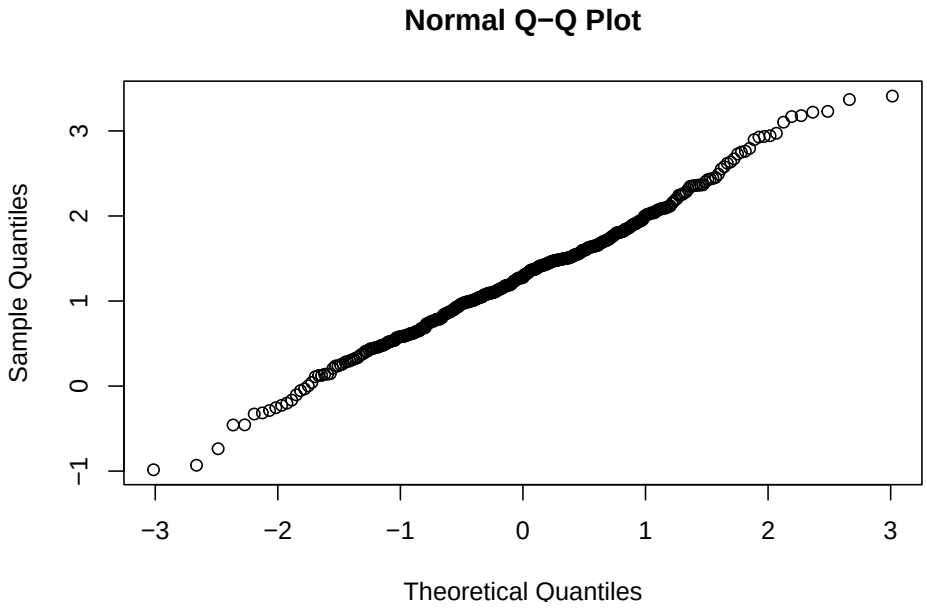

```
mean(filter(at_most_no_more, condition == "at-most")$rating1)
```

```
## [1] 1.265464
```

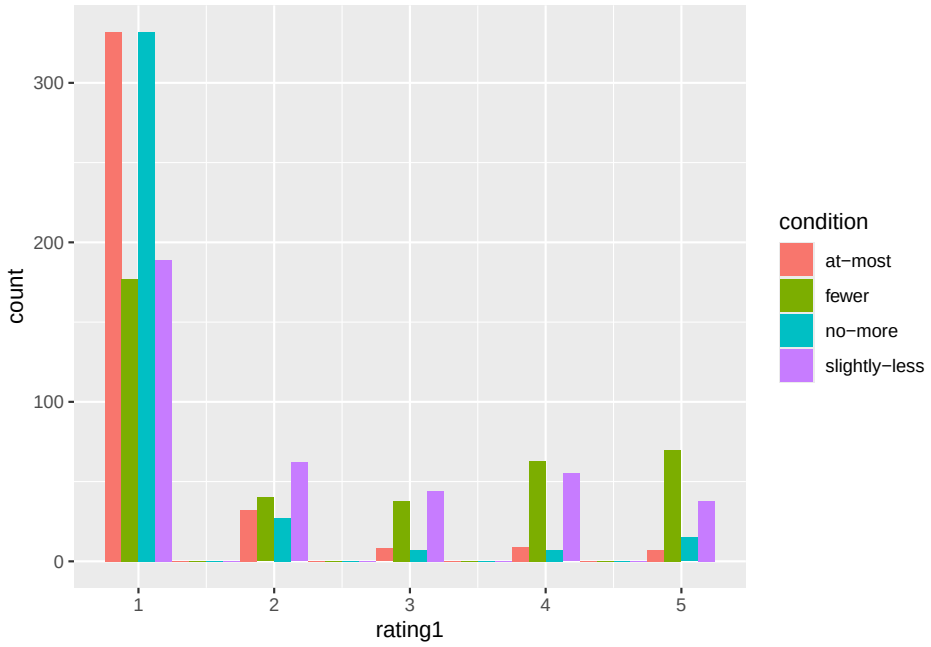
```
sd(filter(at_most_no_more, condition == "at-most")$rating1)
```

```
## [1] 0.7703438
```

```
# vs. Gauss distribution

qqnorm(rnorm(length(filter(at_most_no_more, condition == "at-most")$rating1), 1.27, 0.7
```




```
ggplot(items, aes(x = rating1, fill = condition)) +
  geom_histogram(position = "dodge", alpha = 1, binwidth = 0.5)
```




```
summary(items)
```

```
##    materials          participant          item       condition
```

```
##  Length:1552        Min.   : 1.00   1      : 97   Length:1552
##  Class :character   1st Qu.:25.00   10     : 97   Class :character
##  Mode  :character   Median :50.00   11     : 97   Mode  :character
##                     Mean   :49.68   12     : 97
##                     3rd Qu.:74.00   13     : 97
##                     Max.   :98.00   14     : 97
##                                     (Other):970
##     position     question.order    rating1        content
##  Min.   : 3.00   Min.   :1      Min.   :1.000   Length:1552
##  1st Qu.:10.75   1st Qu.:1      1st Qu.:1.000   Class :character
##  Median :18.50   Median :1      Median :1.000   Mode  :character
##  Mean   :18.41   Mean   :1      Mean   :1.823
##  3rd Qu.:26.25   3rd Qu.:1      3rd Qu.:2.000
##  Max.   :34.00   Max.   :1      Max.   :5.000
##
##    classAB
##  Length:1552
##  Class :character
##  Mode  :character
##
##
##
##
```

```
# linear model

library(lmerTest)

items$condition <- as.factor(items$condition)

items$condition <- relevel(items$condition, ref="at-most")

m1 <- lmer(as.numeric(rating1) ~ condition  + (1|participant) + (1|item), data=items)

summary(m1)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: as.numeric(rating1) ~ condition + (1 | participant) + (1 | item)
##    Data: items
##
## REML criterion at convergence: 4879.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.1573 -0.6556 -0.1995  0.4451  3.6016
```

```
##
## Random effects:
##  Groups      Name        Variance Std.Dev.
##  participant (Intercept) 0.1438   0.3793
##  item        (Intercept) 0.1341   0.3662
##  Residual                1.2334   1.1106
## Number of obs: 1552, groups:  participant, 97; item, 16
##
## Fixed effects:
##                          Estimate Std. Error        df t value Pr(>|t|)
## (Intercept)             1.276e+00  1.142e-01 3.003e+01  11.176  3.2e-12 ***
## conditionfewer          1.231e+00  7.980e-02 1.437e+03  15.428  < 2e-16 ***
## conditionno-more        3.811e-02  7.980e-02 1.437e+03   0.478    0.633
## conditionslightly-less 9.162e-01  7.983e-02 1.438e+03  11.477  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) cndtnf cndtnn-
## conditinfwr -0.349
## conditnn-mr -0.349  0.500
## cndtnslght- -0.349  0.500  0.500
```

```
library(emmeans)
```

A konečně random effects

```
# random effects

m1.ranef <- ranef(m1)

m1.ranef
```

```
## $participant
##     (Intercept)
## 1  -0.047403661
## 2  -0.047403661
## 3   0.115362893
## 4  -0.372936769
## 5   0.033979616
## 6   0.033979616
## 7   0.725737470
## 8  -0.169478577
## 9   0.440896000
## 10 -0.535703323
## 11  0.115362893
## 12 -0.535703323
```

```
## 13 -0.413628407
## 14  0.237437808
## 15 -0.535703323
## 16  0.033979616
## 17  0.440896000
## 18 -0.088095300
## 19  0.033979616
## 20  0.115362893
## 21 -0.413628407
## 22 -0.088095300
## 23  0.156054531
## 24  0.196746169
## 25 -0.495011684
## 26  0.847812385
## 27 -0.210170215
## 28 -0.047403661
## 29 -0.047403661
## 30  0.156054531
## 31 -0.210170215
## 33 -0.535703323
## 34 -0.495011684
## 35  0.278129446
## 36 -0.047403661
## 37 -0.047403661
## 38  0.522279277
## 39 -0.006712023
## 40  0.033979616
## 41  0.074671254
## 42  0.237437808
## 43 -0.372936769
## 44  0.318821085
## 45 -0.291553492
## 46  0.156054531
## 47 -0.454320046
## 48 -0.210170215
## 49  0.074671254
## 50  0.115362893
## 51  0.278129446
## 52 -0.250861854
## 53  0.440896000
## 54 -0.006712023
## 55 -0.413628407
## 56  0.400204362
## 57 -0.210170215
## 58  0.074671254
## 59  0.318821085
```

```
## 60 -0.047403661
## 61 -0.047403661
## 62  0.278129446
## 63 -0.128786938
## 64  0.359512723
## 65  0.074671254
## 66  0.074671254
## 67 -0.088095300
## 68 -0.047403661
## 69 -0.250861854
## 70 -0.128786938
## 71  0.522279277
## 72 -0.169478577
## 73  0.603662554
## 74  0.400204362
## 75  0.156054531
## 76 -0.047403661
## 77  0.400204362
## 78 -0.006712023
## 79 -0.210170215
## 80 -0.006712023
## 81 -0.535703323
## 82 -0.372936769
## 83 -0.047403661
## 84  0.400204362
## 85 -0.495011684
## 86 -0.088095300
## 87 -0.210170215
## 88  0.278129446
## 89  0.115362893
## 90 -0.250861854
## 91 -0.413628407
## 92 -0.210170215
## 93  0.278129446
## 94 -0.088095300
## 95  0.033979616
## 96 -0.088095300
## 97  0.562970916
## 98  0.033979616
##
## $item
##    (Intercept)
## 1   0.15475432
## 10 -0.07879328
## 11  0.19274689
## 12 -0.56296464
```

```
## 13 -0.51379691
## 14  0.50501203
## 15 -0.05207469
## 16 -0.28989442
## 2   0.10953101
## 3  -0.31572869
## 4  -0.04507284
## 5   0.69147855
## 6   0.31668773
## 7   0.18333068
## 8   0.04908930
## 9  -0.34430505
##
## with conditional variances for "participant" "item"
```

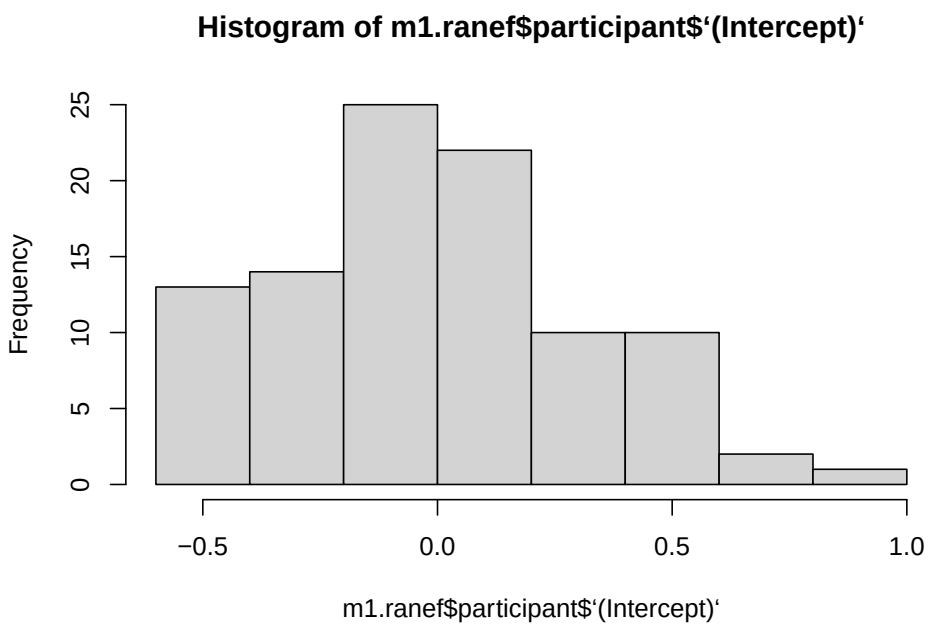
```
m1.ranef$participant$`(Intercept)`
```

```
##  [1] -0.047403661 -0.047403661  0.115362893 -0.372936769  0.033979616
##  [6]  0.033979616  0.725737470 -0.169478577  0.440896000 -0.535703323
## [11]  0.115362893 -0.535703323 -0.413628407  0.237437808 -0.535703323
## [16]  0.033979616  0.440896000 -0.088095300  0.033979616  0.115362893
## [21] -0.413628407 -0.088095300  0.156054531  0.196746169 -0.495011684
## [26]  0.847812385 -0.210170215 -0.047403661 -0.047403661  0.156054531
## [31] -0.210170215 -0.535703323 -0.495011684  0.278129446 -0.047403661
## [36] -0.047403661  0.522279277 -0.006712023  0.033979616  0.074671254
## [41]  0.237437808 -0.372936769  0.318821085 -0.291553492  0.156054531
## [46] -0.454320046 -0.210170215  0.074671254  0.115362893  0.278129446
## [51] -0.250861854  0.440896000 -0.006712023 -0.413628407  0.400204362
## [56] -0.210170215  0.074671254  0.318821085 -0.047403661 -0.047403661
## [61]  0.278129446 -0.128786938  0.359512723  0.074671254  0.074671254
## [66] -0.088095300 -0.047403661 -0.250861854 -0.128786938  0.522279277
## [71] -0.169478577  0.603662554  0.400204362  0.156054531 -0.047403661
## [76]  0.400204362 -0.006712023 -0.210170215 -0.006712023 -0.535703323
## [81] -0.372936769 -0.047403661  0.400204362 -0.495011684 -0.088095300
## [86] -0.210170215  0.278129446  0.115362893 -0.250861854 -0.413628407
## [91] -0.210170215  0.278129446 -0.088095300  0.033979616 -0.088095300
## [96]  0.562970916  0.033979616
```

```
summary(m1.ranef$participant$`(Intercept)`)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.535703 -0.210170 -0.006712  0.000000  0.156055  0.847812
```

```
hist(m1.ranef$participant$`(Intercept)`)
```




```
shapiro.test(m1.ranef$participant$`(Intercept)`)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  m1.ranef$participant$`(Intercept)`
## W = 0.97906, p-value = 0.1239
```

```
# normal enough

mean(m1.ranef$participant$`(Intercept)`)
```

```
## [1] 9.104604e-16
```

```
sd(m1.ranef$participant$`(Intercept)`)
```

```
## [1] 0.3060158
```

```
shapiro.test(rnorm(100, mean = 5, sd = 3))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rnorm(100, mean = 5, sd = 3)
## W = 0.98222, p-value = 0.1971
```

A nakonec pro items ranef

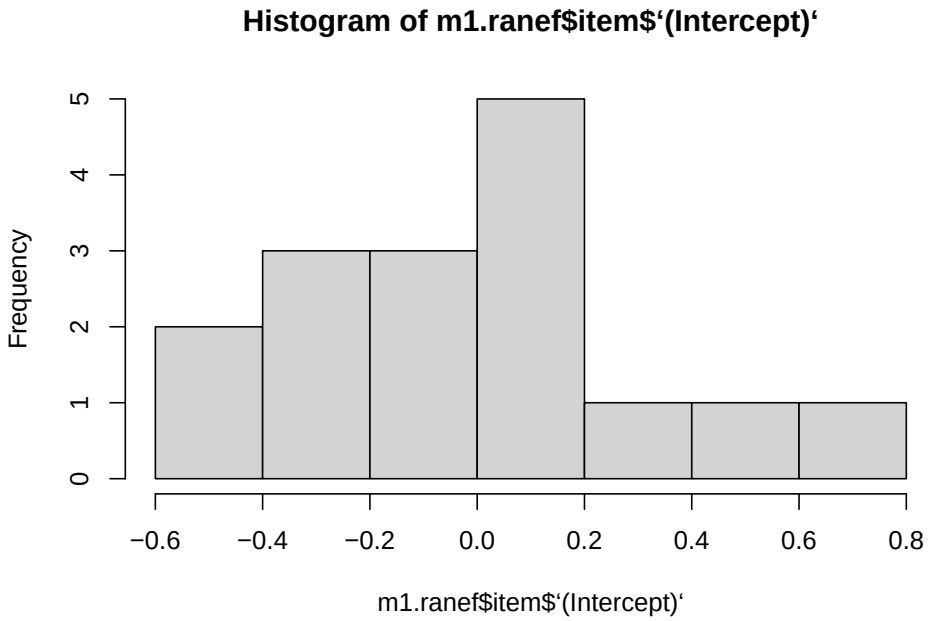
```
m1.ranef$item$`(Intercept)`
```

```
##  [1]  0.15475432 -0.07879328  0.19274689 -0.56296464 -0.51379691  0.50501203
##  [7] -0.05207469 -0.28989442  0.10953101 -0.31572869 -0.04507284  0.69147855
## [13]  0.31668773  0.18333068  0.04908930 -0.34430505
```

```
summary(m1.ranef$item$`(Intercept)`)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.562965 -0.296353  0.002008  0.000000  0.185685  0.691479
```

```
hist(m1.ranef$item$`(Intercept)`)
```




```
shapiro.test(m1.ranef$item$`(Intercept)`)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  m1.ranef$item$`(Intercept)`
## W = 0.97494, p-value = 0.9109
```

```
# normal enough

mean(m1.ranef$item$`(Intercept)`)
```

```
## [1] 5.646915e-14
```

```
sd(m1.ranef$item$`(Intercept)`)
```

```
## [1] 0.3499262
```

```
shapiro.test(rnorm(100, mean = 5, sd = 3))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rnorm(100, mean = 5, sd = 3)
## W = 0.98584, p-value = 0.3643
```

```
t.test(m1.ranef$item$`(Intercept)`,m1.ranef$participant$`(Intercept)`)
```

```
##
##  Welch Two Sample t-test
##
## data:  m1.ranef$item$`(Intercept)` and m1.ranef$participant$`(Intercept)`
## t = 5.9846e-13, df = 18.976, p-value = 1
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.1943237  0.1943237
## sample estimates:
##    mean of x    mean of y
## 5.646915e-14 9.104604e-16
```

3.5 Domácí úkol? (spíš ne)

- t.test mezi dvěma podmínkami

```
at_most <- filter(items, condition=="at-most")
no_more <- filter(items, condition == "fewer")

t.test(at_most$rating1,no_more$rating1)
```

```
##
##  Welch Two Sample t-test
##
## data:  at_most$rating1 and no_more$rating1
## t = -13.745, df = 556.31, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.419801 -1.064735
## sample estimates:
## mean of x mean of y
##  1.265464  2.507732
```

- i když vyjde, tak to není přísně vzato statistický důkaz, protože odpovědi subjektů nejsou (s největší pravděpodobností) v normální distribuci
- ukázat, zda mají podmínky normální distribuci:
  - graf
  - shapiro.test
  - případně použít wilcox.test a porovnat s t.testem – funguje i u dat, která nemají normální distribuci
- cesta k lineárním modelům

Bibliography

Baayen, R. H. (2008). *Analyzing linguistic data: A practical introduction to statistics using R*. Cambridge University Press, Cambridge.