Homework 3

Ilya Schurov, Olga Lyashevskaya, George Moroz, Alla Tambovtseva

Deadline: 10 February, 23:59

Submission

To submit your answers, you should fill in this sample .Rmd file.

1. Download suggested .Rmd file.
2. Open it in RStudio (*File-Open file*).
3. Do not change the titles starting with **###** in this file, just type your answers and type your code in special chuncks {r}.
4. Save changes to your file. Name it as `Surname_Name.Rmd`.
5. Upload this file to Dropbox via the link on the course page.

1. Exact binomial test

The null hypothesis is that $p = 0$ (i.e. no success is possible). In a dataset, there is only one success out of 1 000 000 observations. Will you reject the null hypothesis?

2. Position of verbs in verses

The dataset "The last words in verses" (we have already used this dataset in class) contains a sample of lines taken from the RNC Corpus of Russian Poetry. Actually, there are two samples comprising the texts written in the 1820s and 1920s. We took only one line per author to keep our observations as independent as possible.

Variables:

- `Decade` — decade of creation: 1820s, 1920s.
- `RhymedNwords` — the number of words in the rhyming position, see here.
- `RhymedNsyl` — the number of syllables in the rhyming position.
- `UPoS` — part of speech of the last word.
- `LineText` — a sampled verse.
- `Author` — the author of the text.

Can we decide that in verses written in 1920s, verbs are used in the rhyming posision more often or less often than expected for verbs in general? To calculate the probabilty to come across a verb in written Russian texts (general expectations), use the frequency dictionary of the Russian National Corpus, link (Lyashevskaya, Sharoff 2009).

2.1. General expectations

Read the RNC frequency dictionary data. Verbs are coded as 'v' in the `PoS` field, and their frequency is shown in the `Freq.ipm.` field (relative frequency, # items per million words in the corpus).

2.1.1. Calculate the probability to see verbs dividing the sum of their frequency by the sum of frequency of all words in the dictionary.

2.2. State hypothesis

What is your null hypothesis H_0 and what is the alternative hypothesis H_1?

2.3. Analyse data

Read the dataset "The last words in verses". Filter out the relevant observations from 1920s, calculate the number of verbs observed in the sample, and the sample size.

2.3.1. Use an exact binomial test to calculate p-value.

2.4. Interpret results

Give your interpretation of obtained p-value. Answer the initial question: Can we decide that in verses written in 1920s, verbs are used in the rhyming posision more often or less often than expected?

2.5. Verses in 1820s

Repeat 2.3 for verses written in the 1820s.

2.5.1. Write down your general conlusions about data provided for both 1920s and 1820s data.

3. Do people agree in their judgements regarding language?

In this fictitious experiment, you ask people to assess a certain utterance, whether some feature (interpretation, use of linguistic entity, etc.) is present or absent. Each person assesses only one utterance, and each utterance is independently assessed by two people. If both raters agree that the feature is present, you label the utterance with 1; you label it as 0 in all other cases. Just to give you a hint, here is our example in which we ask people to assess if the use of *for example* is correct with regards to word order:

```
@linguist: Is it possible to say in English:
Its influence can be seen in, _for example_,
the reforms of the court service in the   United Kingdom.
@rater1: yes
@rater2: yes
@linguist: Ok, then I will mark this utterance as 1.
```

3.1. Think of (another) interesting linguistic problem for such an experiment, describe it briefly below.

3.2. Suggest inter-rater agreement level p that according to your intuition takes place in reality (i. e. to what extent people agree, say, $p = 95\%$ or $p = 63\%$, depending you intuition about the complexity and nature of the task).

3.3. Let us consider a null hypothesis that the real p is actually equal to the value you gave in 3.2. Generate a (fictitious) dataset for your experiment assuming that the null hypothesis is true. Pick some number n of datapoints and use the function `rbinom(n, 1, p)` to generate a vector of values that are equal to 1 with the probability p and to 0 with the probability $1 - p$.

3.4. Use an exact binomial test to see whether you can or cannot reject the null hypothesis.

3.5. Provide interpretation. What type of error (type I or type II) would you make if you rejected the null hypothesis in this case?

3.6. Now let us run the same experiment 100 times. How many times the binomial test would suggest you to reject the null hypothesis?

```r
for (i in 1:100) {    # we run the following code 100 times
    # generate a new dataset

    # calculate the p-value for exact binomial test
        # note that binom.test(...)$p.value extracts p-value from the result of binom.test
    binom.p.test <- 1  # change whis line
    if (binom.p.test < 0.05) {
        print("Ooops!")
    }
}
```

Supplementary reading

Use of binomial test in linguistic research

- Gries, Stefan Th. "Phonological similarity in multi-word units." Cognitive Linguistics 22.3 (2011): 491-510. link. Stefan Gries proves that alliteration is observed in multi-word expressions more often than in general.

- Harald Bayen (2008: 51-52) evaluates the probability of observing exactly one occurrence of the word *hare* in the corpus sample of 1 mln words given its estimated frequency of 8.23 words per million according to the SELEX frequency database.

Supplementary R code

This code generates a dataset that consists of Utterances (strings of letters) and Responces corresponding to each utterance (either 0 or 1)

```r
# require(stringi)
n <- 1000 # the number of datapoins
df <- cbind.data.frame(Utterance = stringi::stri_rand_strings(10, 5),
                       # generate a random string
                       Responce = rbinom(n, 1, 0.2))
                       # generate an answer (either 0 ot 1)
                       # randomly with p(1) = 0.2
```

This code run binom.test *n* times with forward-pipes:

```r
require(dplyr)
require(broom)
m <- 5 # sample size in each run
n <- 10 # the number of experiments
dat <- replicate(n=n, expr = sample(0:1, size=m, replace=TRUE)) %>%
        t() %>%                         # transpose row and columns
        as.data.frame() %>%
        mutate(ID=row_number(), sum=rowSums(.), m=m) # add ID, row sums, sample size
dat2 <- dat %>%
        group_by(ID) %>%
        do(tidy(binom.test(.$sum, .$m, alternative = "two.sided"))) %>%
        select(ID, p.value)
```