QHCI 2021 - Project 2

Tobias Boner (17-707-878)

10.05.2021

This project is based on the paper *How We Type: Movement Strategies and Performance in Everyday Typing (https://dl.acm.org/doi/10.1145/2858036.2858233)*. We used the data set the authors have provided within the course of their study. The aim of the data analysis will be to answer the following to research questions (*RQ*):

- *RQ1: Do overall typing speed differ between touch-typist and non-touchtypist?*
- *RQ2: How much does the result from RQ1 influenced by whether the participants type known-words or random string?*

The analysis will base on the **within-subject design** of the study including **one independent variable *(IV)*** that has **three levels**. The **IV *stimulus tpye*** refers to the type of string users had to type in for the transcription and has three levels:

- *sentences*: easy regular sentences
- *random*: random sequences of letters
- *mix*: a mix of the two other conditions

For the experiments they used a **randomized order** for the three *stimulus types*.

Data Analysis

Within this part we will analyze the data set and try to answer the two research questions.

Setup

Before we can jump in the analysis we start with setting up the required dependencies for this R notebook.

```
# load the packages
library(tidyverse)
library(lme4)
library(ggplot2)
library(broom.mixed)
library(lmerTest)
library(plyr)
library(lsm)
library(car)
library(stringr)
```

Some of the required packages have to be installed extra.

```
# define addtitional packages
additional_packages <- c("multcomp", "PearsonDS", "ARTool", "qqplotr", "psych", "moments")

# only install additional packages when they are not already included
install.packages(setdiff(additional_packages, rownames(installed.packages())))
```

To finish the setup we need to load the remaining packages fo finish the setup.

```
# load the remaining additional packages
import::from(moments, skewness)
import::from(MASS,mvrnorm)
import::from(broom.mixed,tidy)
import::from(multcomp, glht, mcp, adjusted)
import::from(PearsonDS, rpearson)
import::from(ARTool, art, artlm, art.con)
import::from(qqplotr, stat_qq_line, stat_qq_point, stat_qq_band)
import::from(psych, d.ci)
import::from(cowplot, plot_grid)
import::from(modelr, add_residuals, add_predictions)
import::from(car, leveneTest, sigmaHat)
```

Data Processing

Next we can start with the data processing. First of, we define a helper function `read_log_files` that will help read in all the different log files.

```
# Regular expression to help parse the log files
regex_parse <- "User(\\d+)_T(\\d+)_(.*).csv"

# helper function to read in log files
read_log_files <- function(fileName){
  csv_data <- read_csv(str_c('./log/',fileName)) %>%
      mutate(length = str_length(stimulus)) %>%
      group_by(stimulus_index) %>%
      summarise(wpm = (as.double(max(length)-1)/5)/ as.double(max(input_time_ms)) * 60000, length = max(length)) %>%
      mutate(user_id = as.numeric(str_replace(fileName, regex_parse, "\\1")), condition = str_replace(fileName, regex_parse,
"\\3"))
    return(csv_data)
}
```

Thereafter, we define all the existing log files (`log_files`) and and the csv containing the participant information (`participant_info`).

```
# list of all log files
log_files <- list.files('./log')

# file that includes the participant information
participant_info <- read_csv("participant_info.csv")

# backup file that is already processed
combined_log_background <- read_csv(("combined_log_background.csv"))
```

Now we are ready to process the log files and create the data we need.

```
# data wrangling
data <- as_tibble_col(lapply(log_files, read_log_files), col='nested') %>%
        unnest(nested) %>% merge(y=participant_info, by='user_id', all.x = TRUE ) %>%
        mutate(typing_style = if_else(is_touchtypist, 'Non-touchtypist', 'Touchtypist')) %>%
        select(-is_touchtypist, -length)

# ? TODO add comment
data_with_cond <- data %>% mutate(condition = factor(condition, levels = c("Sentences", "Mix", "Random")), typing_style = fa
ctor(typing_style, levels = c("Non-touchtypist", "Touchtypist")))
```

> Note that the *IV stimulus type* is now referred to as *condition* in our data set with the same three levels (*Sentences, Random, Mix*). The *typing style* will still be referred to as *typing style*.

Data Exploration and Description

After having processed the data, we now want to inspect and explore it to get a better overview. To do so we can describe its basic properties.

General Information

Thus, it is a good idea to have a quick look at the data by using `head(data)`.

	user_id <dbl>	wpm <dbl>	condition <chr>	typing_style <chr>
1	5307	72.46826	Sentences	Non-touchtypist
2	5307	64.70295	Mix	Non-touchtypist
3	5307	16.96913	Random	Non-touchtypist
4	68349	11.95815	Random	Touchtypist
5	68349	34.31902	Mix	Touchtypist
6	68349	46.01317	Sentences	Touchtypist

6 rows

Next we check the structure of the data object with `str(data)` and can see the data types of each column.

```
## 'data.frame':    90 obs. of  4 variables:
##  $ user_id     : num  5307 5307 5307 68349 68349 ...
##  $ wpm         : num  72.5 64.7 17 12 34.3 ...
##  $ condition   : chr  "Sentences" "Mix" "Random" "Random" ...
##  $ typing_style: chr  "Non-touchtypist" "Non-touchtypist" "Non-touchtypist" "Touchtypist" ...
```

Out data set contains 4 columns (*user_id, wpm, condition, typing_style*) and 90 rows for 30 different users.

Descriptive Statisctics

Now that we know how the data looks and what type of data it contains, we can have a look at the data itself. To get a broad overview of the data we use the `summary` function.

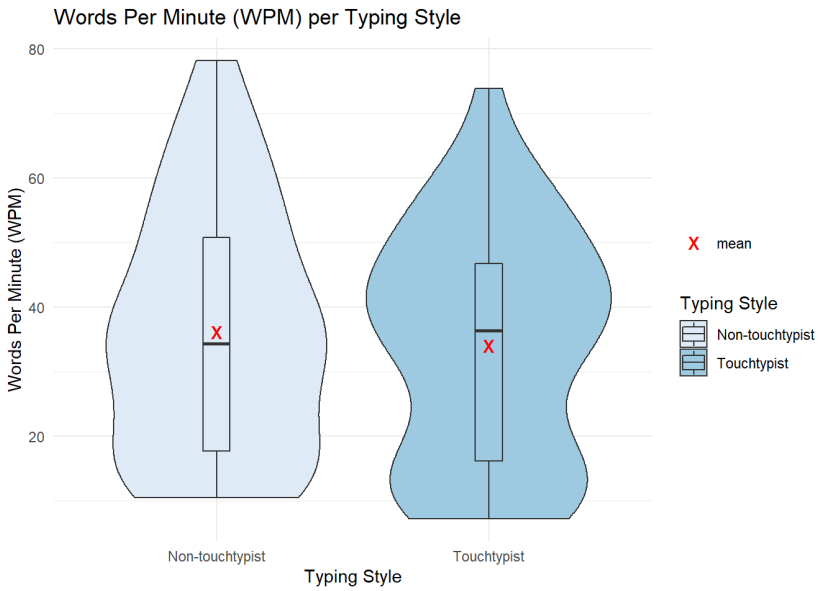
```
##     user_id            wpm          condition         typing_style
##  Min.   :  5307   Min.   : 7.257   Length:90          Length:90
##  1st Qu.:221357   1st Qu.:16.877   Class :character   Class :character
##  Median :374661   Median :34.693   Mode  :character   Mode  :character
##  Mean   :399680   Mean   :35.065
##  3rd Qu.:535385   3rd Qu.:49.085
##  Max.   :934313   Max.   :78.256
```

Out of this summary we get several statistical measures for the first three columns *user_id, wpm, condition* that contain numerical data. For the remaining two rows *condition, typing_style* including strings we get again the length of the columns.

In order to answer the research questions we will now have a look at the *words per minute (wpm)* first with regards to the *typing style* (RQ1) and then in connection with the different *conditions* (RQ2). To support this process we will also create suitable visualizations for the data sets.

RQ1 - words per minute (wpm) by typing styles

First we create a violin plot that includes boxplots for *words per minute (wpm)* by the *typing styles* of users.



Out of this first visualization, we can already imply that there is no big difference between the two *typing styles* with regards to *words per minutes (wpm)*. This is also shown in the numbers.

Skewness is a commonly used measure of the symmetry of a statistical distribution. A negative skewness indicates that the distribution is left skewed and the mean of the data (average) is less than the median value

Non-Parametric Statisctics

- Median:
  - On the one hand Non-touchtypist has a `median` of 34.3066834 *wpms* and on the other hand Touchtypist has a `median` of 36.384337 *wpms*.
- Min and Max Values:
  - For the `min` values Non-touchtypist has a value of 10.4895105 *wpms* and Touchtypist has a value of 7.2573329 *wpms*.
  - Regarding the `max` values Non-touchtypist has a value of 78.2560089 *wpms* and Touchtypist has a value of 73.8996559 *wpms*.
- Interquartile Range (IQR) :
  - The `IQR` of Non-touchtypist amounts to 33.150654 and on the `IQR` for Touchtypist to 30.5350213.

These non-parametric statistical properties confirm the initial impressions from the visualization above that there are only slight differences between the two *typing styles* with regards to *words per minute (wpm)*. The `median` , `min` , `max` , as well as the `IQR` reveal only small differences.

Parametric Statistics

- Mean:
  - On the one hand Non-touchtypist has a `mean` of 36.1797244 *wpms* and on the other hand Touchtypist has a `mean` of 34.0888643 *wpms*.
- Spread:
  - For the variance (`var`), Non-touchtypist has a value of 356.6256903 and Touchtypist has a value of 321.8579526.
  - Regarding the standard deviation, (`sd`) Non-touchtypist accounts to 18.8845357 and Touchtypist to 17.9404.

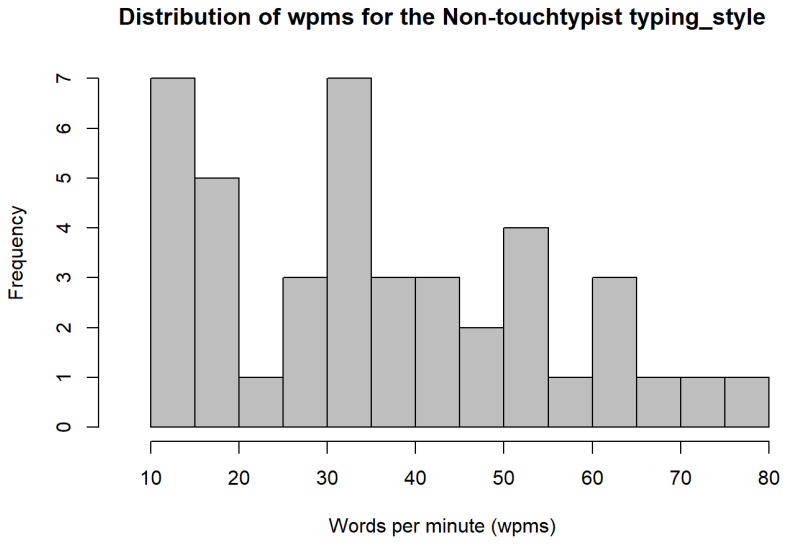
These numbers shows that also the parametric statistical properties in the form of `mean` , variance(`var`) and standard deviation(`sd`) are quite similar.

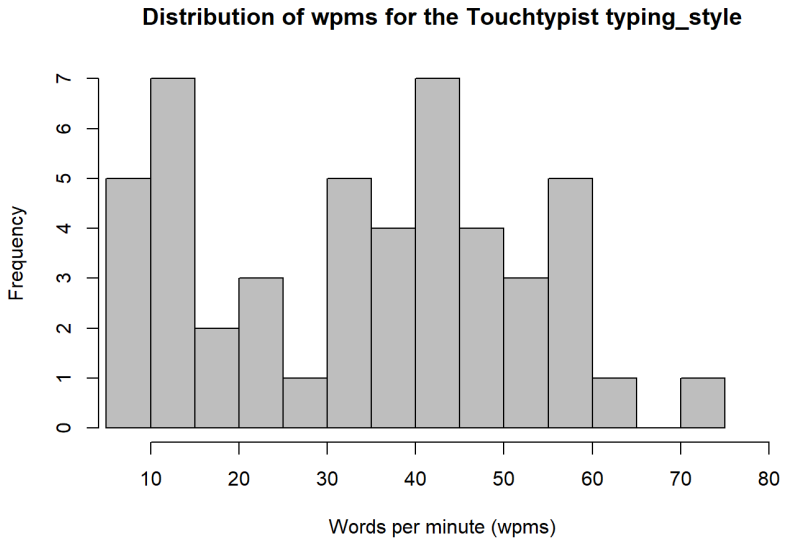
Now that we have concrete numbers for both the `mean` and the `median` for *typing styles* by *wpms* we can make a statement about the distributions of the data and their shapes. For Non-touchtypist that have a `mean` of 36.1797244 *wpms* and a `median` of 34.3066834 *wpms*, the shape is skewed right. The same applies For Touchtypist that have a `mean` of 34.0888643 *wpms* and a `median` of 36.384337 *wpms*, where the shape is skewed-left.

Both shapes are only slightly skewed as can also be seen in the visualization where the `mean` and `median` are very close. Moreover, the distribution of Touchtypist is more bimodal and skewed-left whereas the shape of Non-touchtypist is just skewed right or unimodal, as can be seen in the visualization as well.

Eventually, the numbers confirmed the initial impressions from the visualizations that there is not a big difference between the two *typing styles* with regards to the *wpms*. But there is a slight difference for example in the distribution and shape of the data.

Histograms



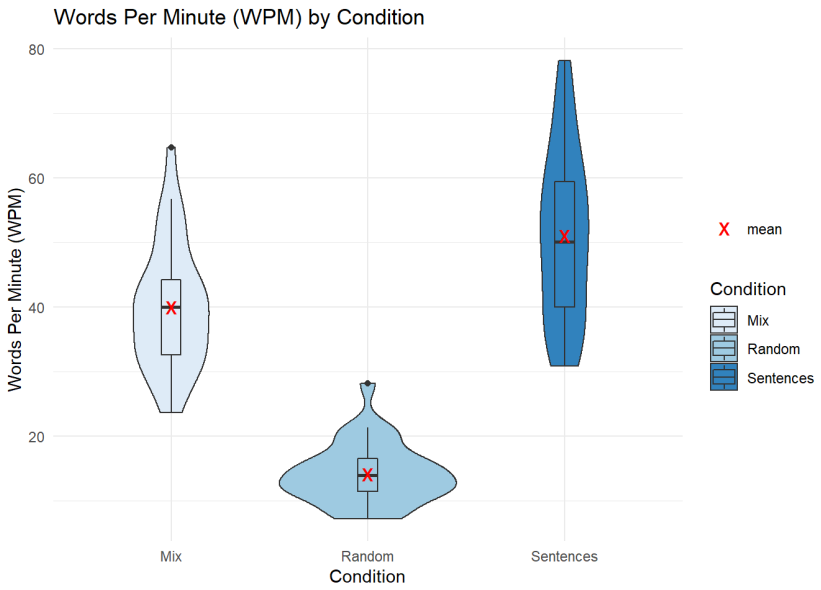



```
# r function for skewness
by(data = data$wpm, INDICES = data$typing_style, FUN = skewness)
```

```
## data$typing_style: Non-touchtypist
## [1] 0.4017579
## ------------------------------------------------------------
## data$typing_style: Touchtypist
## [1] 0.04985883
```

RQ2 - words per minute (wpm) by condition (stimulus type)

With RQ2 in mind we create the same plot again for *words per minute (wpm)* by the different *conditions*.



In contrast to the visualization before, in this case we immediately see that the violin and boxplots appear quite different with regards to the *conditions* and *word per minutes (wpm)*. Hence we should again have a closer look at the numbers again.

Non-Parametric Statistics

- Median:
  - Mix has a `median` of 40.0147473 *wpms*, Random has a `median` of 13.9851792 *wpms* and Sentences has a `median` of 50.1618746 *wpms*.
- Min and Max Values:
  - For the `min` values Mix has a value of 23.6483974 *wpms*, Random has a value of 7.2573329 *wpms* and Sentences has a value of 30.893995 *wpms*.
  - Regarding the `max` values Mix has a value of 64.7029456 *wpms*, Random has a value of 28.1579977 *wpms* and Sentences has a value of 78.2560089 *wpms*.
- Interquartile Range (IQR) :
  - The `IQR` of Mix amounts to 11.600442, the `IQR` for Random to 5.1599867 and on the `IQR` for Sentences to 19.4893757.

The `median` , `min` , `max` , as well as the `IQR` this time show significant differences that are also obvious in the visualizations. Upon these we can already make a statement about RQ2.

- Condition Mix:
  - User who had to type strings in form of Mix were the slowest compared to the other conditions. This shows in the lowest `median` (`r` median_wpm_by_condition[[1]]), min `(23.6483974) and` max` (64.7029456) values.
  - But these users had a very small spread which is by the `IQR of` (11.600442) that is lower than the others.
- Condition Random:
  - User who had to type strings in Random form were faster than Mix strings but slower than regular Sentences.
  - They had larger spread indicated by the `IQR of` (19.4893757) but not the largest compared to the others .
- Condition Sentences:
  - User who had to type strings in form of Sentences were the fastest, which is shown in the highest `median` (50.1618746), `min` (30.893995) and `max` (78.2560089) values compared to the other conditions.
  - Also they had the largest spread indicated by the `IQR of` (19.4893757) that is higher than the others.

At this point we can already say that there are obvious differences between the *conditions*.

Parametric Statistics

- Mean:
  - For the Mix users have a `mean` of 39.985118 *wpms*, forRandom a `mean` of 14.1680566 *wpms* and forSentences has a `mean` of 51.0406225 *wpms*.
- Spread:
  - For the variance (`var`), Mix has a value of 93.8200508, Random has a value of 21.8960003 and Sentences has a value of 172.6837276.
  - Regarding the standard deviation, (`sd`) Mix accounts to 9.6860751, Random to 4.6793162 and Sentences to 13.1409181.

After getting the concrete values for `mean` , variance(`var`) and standard deviation(`sd`) of the data, we can again make a few statements about the distribution and the shape. From the visualization we already see that the mean (*marked as red dots*) and the median for all of the three *conditions* are very very close to each other.

Looking at the concrete numbers for both the `mean` and the `median` for *condtitions* by *wpms* this gets clearer:

- Condition Mix:
  - For Mix that have a `mean` of 39.985118 *wpms* and a `median` of 40.0147473 *wpms*.
- Condition Random:
  - The same applies for Random with a `mean` of 14.1680566 *wpms* and a `median` of 13.9851792 *wpms*.
- Condition Sentences:
  - For Sentences the `mean` is 51.0406225 *wpms* and the `median` is 50.1618746 *wpms*.

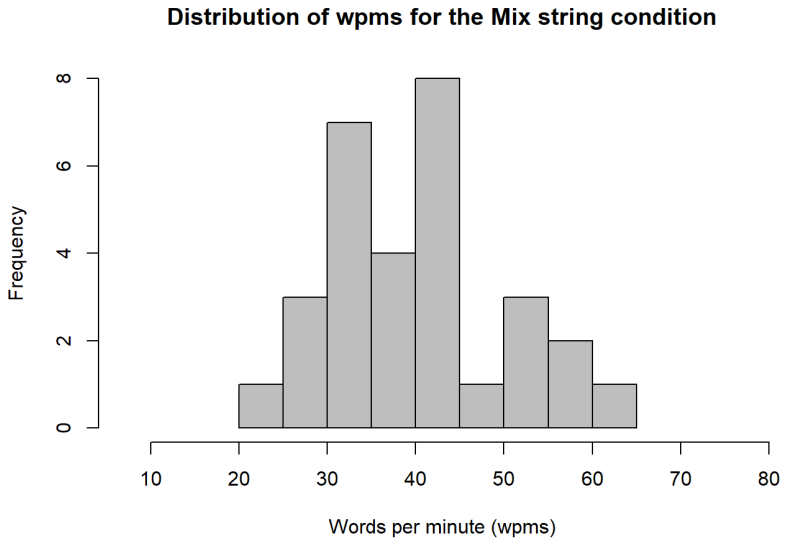
Out of this numbers we see that the differences for Mix and Random are very marginal and do not really imply a skewness as can be seen on the visualizations. Thus, we can say that both distributions are very very close to a normal distribution. For the Random *condition* the spread is also very small, which says that most people are constantly slow when typing in Random strings. For the Mix *condition* the spread is already bigger, where some people show to get along better with typing in Mix strings.

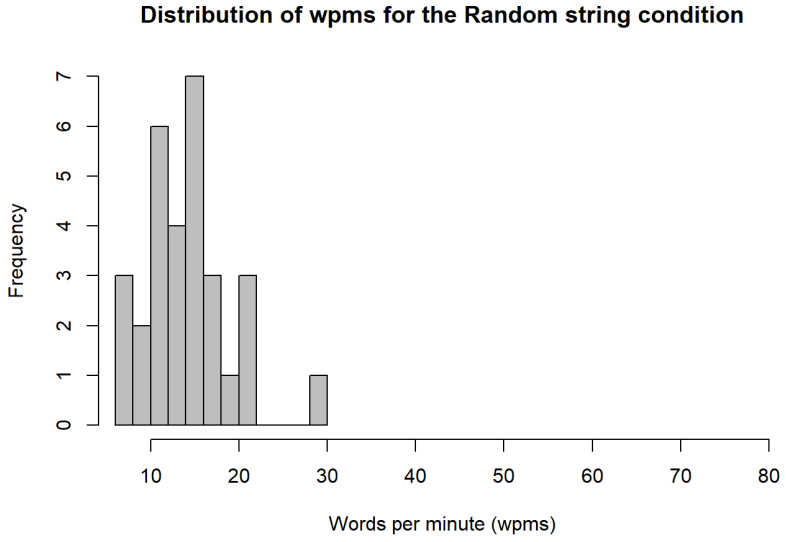
Then for the Sentences *condition* the difference `mean` and `median` is still very close but a bit larger, since it can also be visually be distinguished in the visualization. Hence, it can be argued that the shape of Sentences is minimally Sentences is just skewed right. The visualization shows that the spread here is bigger and there are the largest differences regarding *wpms* for users who had to type in whole regular Sentences.

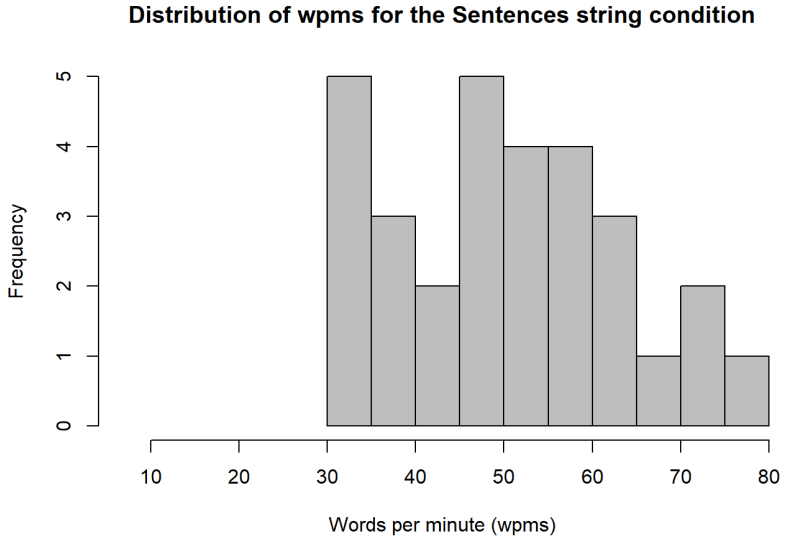
Both shapes are only slightly skewed as can also be seen in the visualization where the `mean` and `median` are very close. Moreover, the distribution of Random is more bimodal and skewed-right whereas the shape of Mix is just skewed left or unimodal, as can be seen in the visualization as well.

Eventually, for this case the the numbers related to the spread of the data (variance(`var`), standard deviation(`sd`), `IQR`) do confirm the impression of the visualization that there are significant differences. The `mean` and `median` do not show big differences at all for all three conditions.

Histograms







Statistical Assumptions

TODO: include inferential statistics here already?

Data Variation

Linear Model

test columns using inferential statistics

Construct Linear Model

Model-fit Assessment

Estimations

Test Statistical Assumptions for the model

Results