

# Assignment Portfolio

## Homework 1

### Exercise 1

1. What regular expressions do you use to extract all the dates in this blurb: <http://bit.ly/regexexercise2> and to put them into the following format YYYY-MM-DD ?
2. `\d{1,2}.\d{1,2}..\d{1,4}`

### Exercise 2

1. Write a regular expression to convert the stopwordslist (list of most frequent Danish words) from Voyant in <http://bit.ly/regexexercise3> into a neat stopword list for R (which comprises "words" separated by commas, such as <http://bit.ly/regexexercise4>). Then take the stopwordslist from R <http://bit.ly/regexexercise4> and convert it into a Voyant list (words on separate line without interpunction)

Regular expression

`(\S+\b)(\n)`

Substitution

`"$1",`

### Exercise 3

Words in an essay can never accurately capture the ideal spreadsheet. However the ideal spreadsheet should be easy to work with, draw its information from good sources and have good human readability, meaning that it should be obvious to the reader the different columns and rows mean. Furthermore it is important to avoid special characters, so they don't interfere with the programming language used to read the dataset. Which special characters are okay to use will differ depending on the program. For example in R an underscore is unlikely to interfere. One should also only have one value per cell.

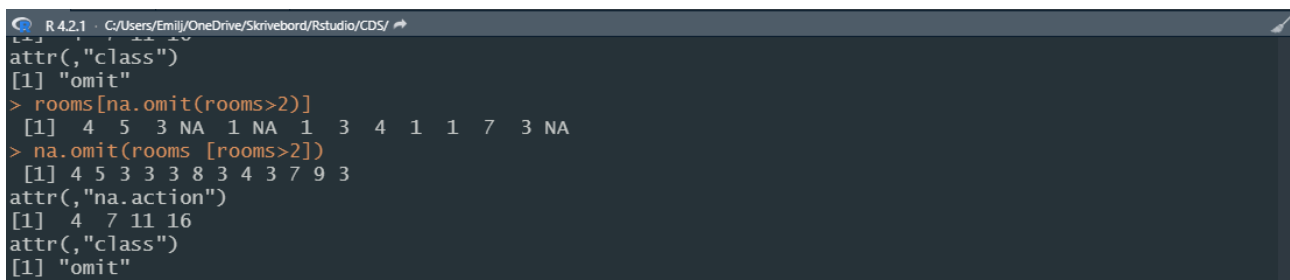
## Exercise 4

`[Dd]*s\s[Mm]anibus|[Dd][Mm]|[Dd][Mm][Ss]|[Dd]*s\s[Mm]anibus\sSacrum`

24062 matches

## Homework 3

### Exercise 1



```
R 4.2.1 - C:/Users/Emilj/OneDrive/Skrivebord/Rstudio/CDS/
attr("class")
[1] "omit"
> rooms[na.omit(rooms>2)]
[1] 4 5 3 NA 1 NA 1 3 4 1 1 7 3 NA
> na.omit(rooms [rooms>2])
[1] 4 5 3 3 3 8 3 4 3 7 9 3
attr("na.action")
[1] 4 7 11 16
attr("class")
[1] "omit"
```

Code:

```
rooms <- c(1, 2, 4, 5, 1, 3, 1, NA, 3, 1, 3, 2, 1, NA, 1, 8, 3, 1, 4, NA, 1, 3, 1, 2, 1, 7, 1, 9, 3, NA)
```

```
na.omit(rooms [rooms>2])
```

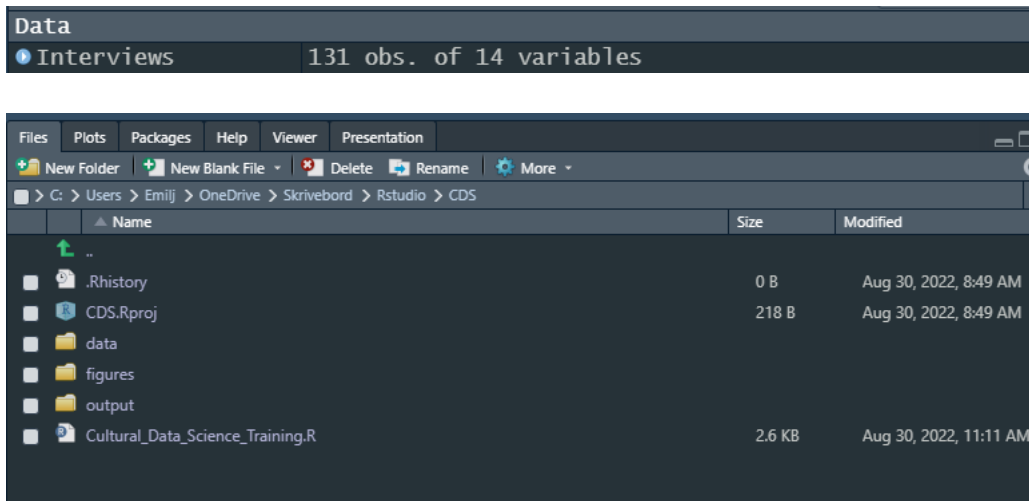
### Exercise 2

code: `class(rooms)`

numeric

### Exercise 3

```
Interviews <- read.csv("data/SAFI_clean.csv")
```



<https://github.com/Digital-Methods-HASS/AU633260> Emil B Jacobsen

### Homework 4

#### DESCRIPTION

For this Visualisation assignment, you engage in **one of the two** tasks and submit an rmarkdown and html document (ie, a knitted result) that collects the results of both tasks:

Choose **one** of the two options below and follow the instructions in rmarkdowns. These have slightly different content depending on what you wish to practice, whether facets in ggplot or animation. For the latter, pay attention to the prerequisites R specifies for your system:

1) Historical homicide trends across Western Europe (ggplot practice) <https://github.com/Digital-Methods-HASS/HomicideHistory>

OR

2) Global development since 1957 (learn how to create animations with *gganimate* package!)

:) <https://github.com/Digital-Methods-HASS/GlobalDevelopment>

```
```{r load-data}
western_europe <- read_csv("data/homicide-rates-across-western-europe.csv")
```

## Inspect the data

How clean and analysis-ready is the dataset? Do you understand what the column names represent? what is the difference between rate and homicide number?

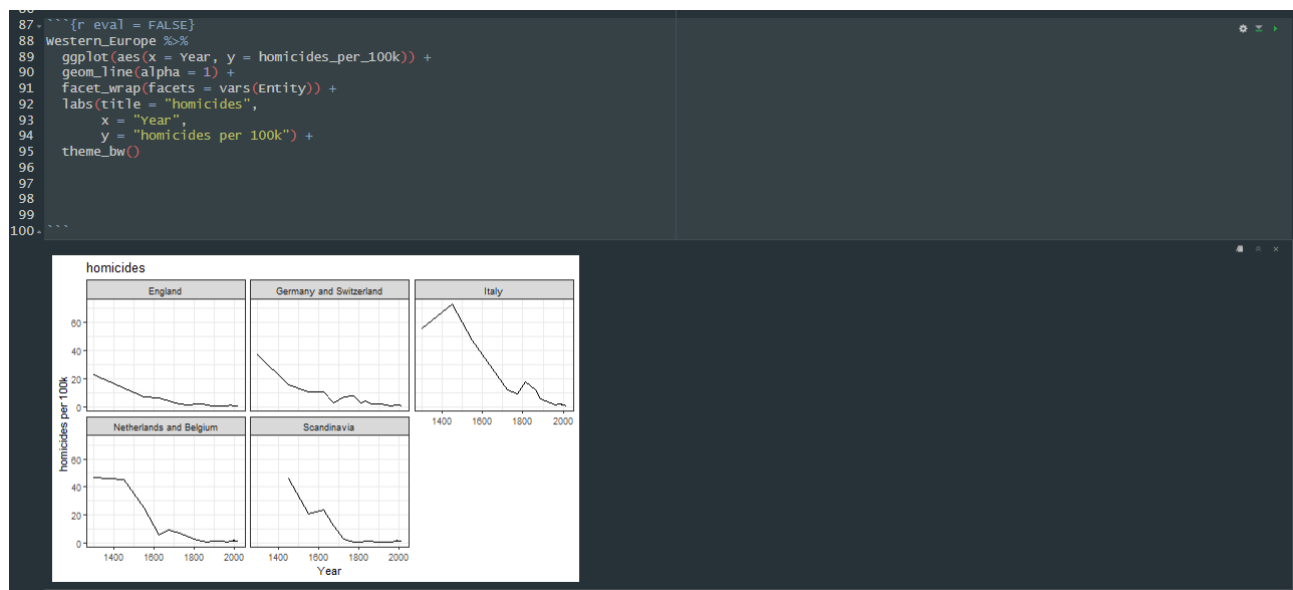
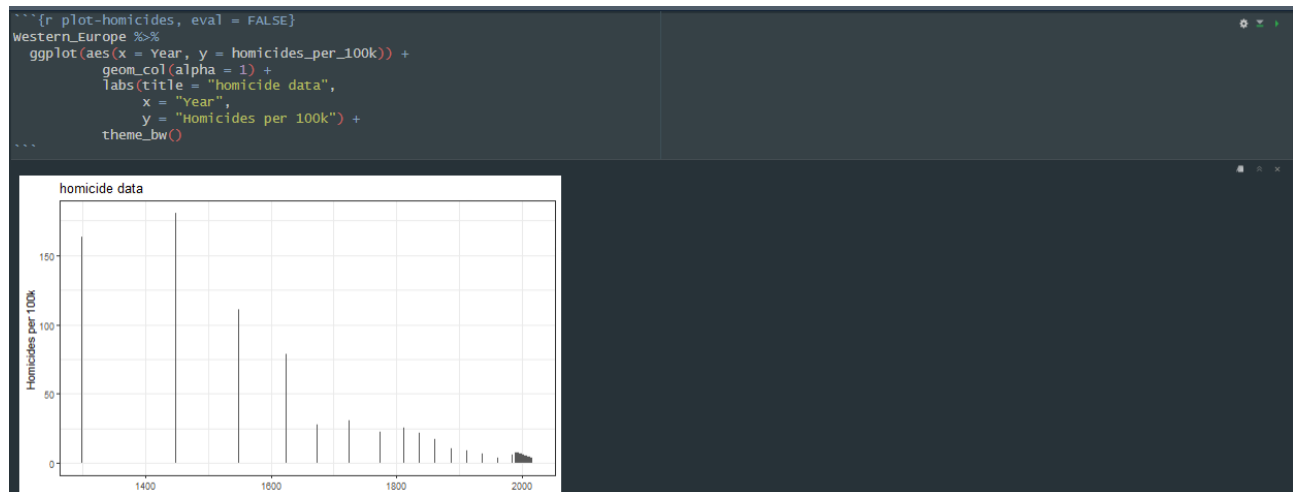
```{r inspect}
head(western_europe)
```

Ok, the data look good except for the column `Homicide rate in Europe over long-term (per 100,000)` which is not very easy to work with.

- Use the `names()` function and assignment key to relabel this column to `homicides_per_100k`

```{r relabel column}
# YOUR CODE

names(western_europe)[4] <- 'homicides_per_100k'
western_europe
```
```



```

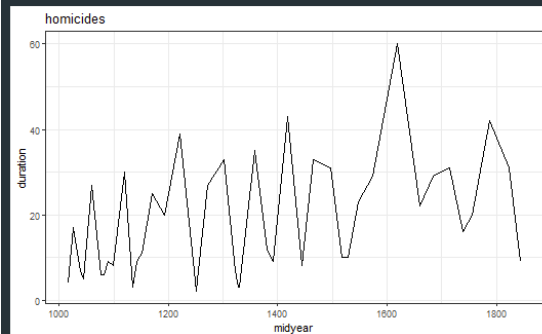
- Start a ggplot plotting midyear on x axis and duration on y axis
- Try 'geom_smooth()' for geometry
- Provide meaningful labels and a title
- How would you characterize the trend compared to the homicides above?

```

```

'''{r eval = FALSE}
kings %>%
  ggplot(aes(x = Midyear, y = Duration)) +
  geom_line(alpha = 1) +
  labs(title = "homicides",
        x = "midyear",
        y = "duration") +
  theme_bw()
'''

```



3) Question: In <250 words articulate your answer on the basis of the data visualisations to the following question: are we more civilized today?

,

To answer the the question one must first define what is meant by civilized. If it is not civilized to murder each other, then yes we are. If killing is neither civilized nor un civilized then this data is not really relevant. But assuming that this is not the case based on this data then we truly have become more civilised throughout time.

Submit here a *\*link\** to **your au#####** repository in <https://github.com/Digital-Methods-HASS> ... which leads directly to the place where you have posted your solution as **both .Rmd** and **.html** files.

<https://github.com/Digital-Methods-HASS/AU633260> Emil B Jacobsen/tree/main/homework emil

## Homework 5

### Exercise 1

Identify the names and format of the 3 biggest files. Can you come up with a command to generate a numerically ordered list of 3 biggest files? (hint: consider using **wc** to gauge image size)

**ls -S | head -3**

### Exercise 2

Some of the image files are empty, a sign of corruption. Can you **find** the empty photo files (0 kb size) , count them, and generate a list of their filenames to make their later replacement easier?

**find . -size 0M > corrupted\_files.txt**

### Exercise 3

**Optional/Advanced:** Imagine you have a directory [goodphotos/](#) (same password as above) with original non-zero-length files sitting at the same level as the current directory. How would you write a loop to replace the zero length files?

## Homework 7

Clone the repository at <https://github.com/Digital-Methods-HASS/WebscrapingPoliceKillings> and depending on your familiarity with R, either

1) adapt the web-scraping example to scrape homicide data from FBI site and produce a meaningful report on how homicide trends evolve around US in relation to this urban unrest

or

2) use the *rvest* library to scrape data of your interest (football statistics in Wikipedia?, gender representatives in different governments? global population by country in <https://www.worldometers.info/world-population/population-by-country/> )

or

3) produce data visualisations that shed light on another interesting aspect of the police killing data

Submit both the .rmd and the rendered .html files to your `au#####` github repository and paste link here.

[https://github.com/Digital-Methods-HASS/AU633260\\_Emil\\_B\\_Jacobsen](https://github.com/Digital-Methods-HASS/AU633260_Emil_B_Jacobsen)

## Homework 8

### DESCRIPTION

Choose whether you wish to practice Sentiment Analysis or Text mining. Using the pre-prepared repositories [https://github.com/Digital-Methods-HASS/CDS\\_W12](https://github.com/Digital-Methods-HASS/CDS_W12) and [www.github.com/maxodsbjerg/TextMiningStCroixAvis](https://github.com/maxodsbjerg/TextMiningStCroixAvis)

either

- 1) Reproduce the code in the repository and extend it following the suggestion (e.g., assess and consider the sentiment in the Game of Thrones) or your own body of text
- 2) find a suitable dataset or document and analyse it using the text-mining and sentiment-analysis approaches

Getting the document from the data folder

```
#getting the data
scw_path <- here("data", "Spanish_Civil_War.pdf")
scw_text <- pdf_text(scw_path)
scw_df <- data.frame(scw_text) %>%
  mutate(text_full = str_split(scw_text, pattern = '\\n')) %>% #splitting columns
  unnest(text_full) %>% #putting into regular columns
  mutate(text_full = str_trim(text_full)) #removing white spaces
```

Now each line, on each page, is its own row, with extra starting & trailing spaces removed.

# Get the tokens (individual words) in tidy format

Use `tidytext::unnest_tokens()` (which pulls from the `tokenizer` package, to split columns into tokens. We are interested in *words*, so that's the token we'll use:

```
scw_tokens <- scw_df %>%
  unnest_tokens(word, text_full)

# See how this differs from `scw_df`
# Each word has its own row!
```

## Counting words

```
scw_wc <- scw_tokens %>%
  count(word) %>%
  arrange(-n)
```

## Making and removing stopwords

```
scw_stop <- scw_tokens %>%
  anti_join(stop_words) %>%
  select(-scw_text)
```

## Counting words with stopword list

```
scw_swc <- scw_stop %>%
  count(word) %>%
  arrange(-n)
```

## Removing numbers

```
scw_no_numeric <- scw_stop %>%
  filter(is.na(as.numeric(word)))
```

## filtering 100 most used words

```
length(unique(scw_no_numeric$word))
## [1] 5713

scw_top100 <- scw_no_numeric %>%
  count(word) %>%
  arrange(-n) %>%
  head(100)
```



## Making a wordcloud to see how which authors are most used in the notes

```
ggplot(data = scw_top100, aes(label = word, size = n)) +
  geom_text_wordcloud_area(aes(color = n), shape = "square") +
  scale_size_area(max_size = 16) +
  scale_color_gradientn(colors = c("darkgreen", "blue", "red")) +
  theme_minimal()
```

3) create an informative or fun visualisation



3) submit here a **rendered Rmarkdown file** with a link to Github repository with your data, analysis and R.proj file

submit here a **rendered Rmarkdown file** with a link to Github repository with your data, analysis and R.proj file

<https://github.com/Digital-Methods-HASS/AU633260> Emil B Jacobsen