**Fallen Danish soldiers during the great war – R-studios**

Jens Nielsen

**Abstract.**

*This paper seeks to answers broad questions about the fallen Danish soldiers during the 1. world war. These will be questions related to the nature of the fighting, where the soldiers saw action, and who the typical soldier was. This will be done with the help of the digital tool R-studios which allows easy processing of large amounts of data. The dataset used was made by* the national archive *of southern Jutland (Landsarkivet for Sønderjylland and the museum of southern Jutland (Museum Sønderjylland).*

**Keywords:**

*Danish soldiers, 1. world war, R-studios*

**Free text.**

*Description of your script or software in maximum 5 pages for first Original Software Publication – see suggested format;*

**Introduction**

Digital tools allow us to answer questions that previously were outside the realm of possibility. For even processing large amounts of data manually might be possible in theory. The use of computers and digital software allows us to do in minutes what would have taken hundreds of hours to accomplice manually.

Back in May 2012 the national archive of southern Jutland (Landsarkivet for Sønderjylland) and the museum of southern Jutland (Museum Sønderjylland) gathered together a team consisting of 15 genealogists and employees. Their task was to collect data concerning fallen Danish soldiers from southern Jutland during the 1. world war. Using information from churchbooks, memorial stones both in Southern Jutland and France. The list has since been updates multiple times. The most recent update as of December 2019 being the work done I April 2018. The list now includes information on 6581 fallen Danish soldiers who fought on the German side during the 1. world war. For more information on this dataset please contact project leader Hanne C. Christensen from the museum of southern Jutland (Museum Sønderjylland). [Hanne2@museum-sonderjylland.dk](mailto:Hanne2@museum-sonderjylland.dk)

The dataset includes a multitude of information included but not limited to: Civil status, age at the time of death, place of death, cause of death, regiment, rank etc.

**Problems and Background**

Interesting as it may be to browse around this dataset, its true value becomes apparent when you apply statistical analysis. This dataset includes 6581 fallen Danish soldier. That is very close to the entirety of the fallen Danish soldiers. It is therefore possible answer questions related the typical soldier, and the way and where he fought and died.

Where did the Danish soldiers primarily see action? The western front? Or the Eastern front?

What was the nature of the fighting? Did most die on the battlefield? Or did they die later from injuries sustained? What about sickness?

Among the casualties, where there more young unmarried men? Or older married men?

These are some of the questions I will seek to answer.

* *Introduce the background and related work in literature (cite or list algorithms used, other scripts and software etc.) – hvad er det?*

**Software Framework**

For this project R-studio in the software needed.

*3.1 Software Architecture/Prerequisites*

* *Give a short overview of the overall software architecture, dependencies and prerequisites – spørg om det er nok*

***Data Acquisition and Processing***

As I have a keen interest in the 1. world war I was looking for a dataset related to this topic. The 100-year anniversary for the reunification of southern Jutland is just around the corner as pr. 2019. This made the topic of Danish soldier’s service in the German army even more relevant.

As I know that Max Odsbjerg Pedersen, (whom is connected to my course “Digital methods for Historians”), previously have been working with the 1. world war, I decided to email him just in case he had any dataset concerning this topic lying around. It turned out that Max had been in contact with Hanne C. Christensen from the museum of southern Jutland, and she send the dataset to Max as a CSV-file. Max had already been cleaning the dataset up a bit for easy insertion into R-studio. Max was so kind as to share this dataset with me after Hanne gave permission. This permission came with three conditions:

1. I was to make clear that I got the list of fallen Danish soldiers from The Museum of Southern Jutland. (Museum Sønderjylland)
2. I was to make clear that I am working with the list from April 2018
3. I was to send the result of my work to Hanne

Conditions which I have now fulfilled.

**Implementation and Empirical Results**

I started by loading tidyverse in RStudio. I needed this in order to visualize the data later on.

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.2.1 v purrr 0.3.3  
## v tibble 2.1.3 v dplyr 0.8.3  
## v tidyr 1.0.0 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.4.0

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(readxl)

I then loaded my dataset into RStudio

faldne <- read\_excel("data/Kopi af faldne\_MAX.xlsx")  
faldne

## # A tibble: 6,580 x 28  
## Kilde `Kildens stedna~ `År og nr. for ~ `Regiments-nr` `Type: Infanter~  
## <chr> <chr> <chr> <chr> <chr>   
## 1 <NA> <NA> <NA> <NA> <NA>   
## 2 AK <NA> Mindeblad, Flen~ 31 Reserve-Infante~  
## 3 DBR Møgeltønder 1916-001 60 Infanterie-   
## 4 DBR Møgeltønder 1918-016 7 Garde-Infanteri~  
## 5 DBR Aabenraa 1916-034 118 Infanterie-   
## 6 DBR Lysabild 1914-019 86 Füsilier-   
## 7 DBR Broager 1918-017 75 Reserve-Infante~  
## 8 A-IE <NA> E-03203 86 Infanterie-   
## 9 DBR Felsted 1915-033 <NA> Lehr-Infanterie   
## 10 MS Sønderborg <NA> <NA> <NA>   
## # ... with 6,570 more rows, and 23 more variables: Bataillon <chr>,  
## # Kompagni <chr>, Andet <chr>, `Militær rang` <chr>, Efternavn <chr>,  
## # `Fornavn(e)` <chr>, `Alder ved død` <dbl>, Bopæl <chr>,  
## # `Bopæls-sogn` <chr>, `Fødsels-dato` <chr>, Fødested <chr>, Fødesogn <chr>,  
## # Dødssted <chr>, Dødsland <chr>, Dødsdato <chr>, `Døds-årsag` <chr>,  
## # `Indtasters bemærkninger f.eks. læseproblemer, regimentsnavn eller  
## # supplerende oplysninger og/eller kilder.` <chr>, `Lokale mindesten eller  
## # mindetavler for faldne` <chr>, Marselisborg <chr>, `Opført på mindetavler i  
## # Braine, Frankrig. 2013` <chr>, `Minde-blad på LAA, MSS eller ADCB:` <chr>,  
## # `Evt. foto på MSS:` <chr>, `Evt. ansøgning i Invalidenævnets  
## # Efterladtesager, LAA:` <chr>

This looks way more structured I RStudio by the way! As you can see the dataset contains 6570 rows. These are individual soldiers. These soldiers are sorted under 23 variable such as “first name” “last name” “regiment” etc.

I now set out to write some code which could make some sense out of all this raw data. A trick to make your life easier in programming is to repurpose old code for your own needs.

After talking to Max he told me that he previously had been works on project to find the most common words used under the hashtag #dkpol.

**Code from Max**:[[1]](#footnote-1)



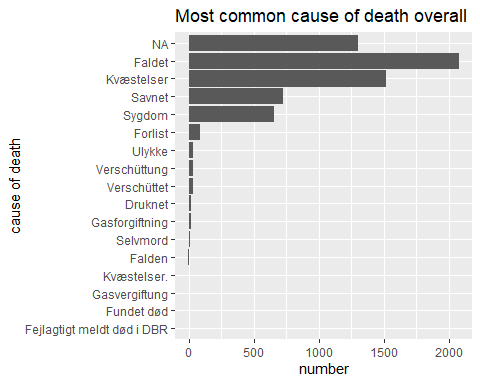
As shown above this code is made to select the top 15 most common word, count them and display them along a X and Y axis.

**Most common cause of death**

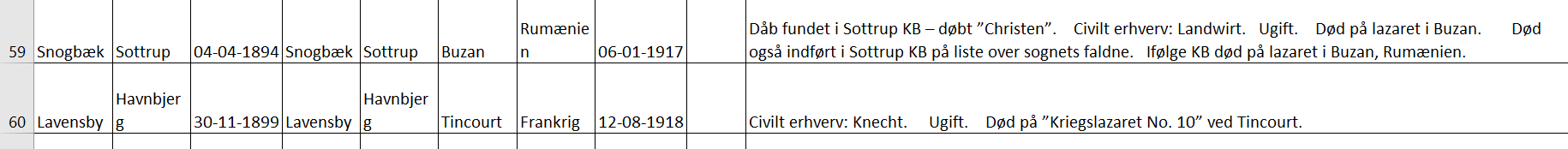
I first wanted to find out what the most common cause of death was among the Danish soldiers. I order to modify the code for this use I simple had to change “word” with “cause of death”.(Døds-årsag in danish). Now the code is counting the 15 most common words as they appear in the column “Døds-årsag” within the CSV-file.

faldne %>%  
 count(`Døds-årsag`, sort = TRUE) %>%  
 top\_n(15) %>%  
 mutate(`Døds-årsag` = reorder(`Døds-årsag`, n)) %>%  
 ggplot(aes(x = `Døds-årsag`, y = n)) +  
 geom\_col() +  
 xlab(NULL) +  
 coord\_flip() +  
 labs(x = "cause of death",  
 y = "number",  
 title = "Most common cause of death overall")

## Selecting by n



The first thing I notice is the relatively large amount of NA hit. Working with incomplete data is however to be expected is however to be expected as one works with historical sources.



These are two examples from the category NA. Cause of death is not specifically noted. It is however noted that they died in a military hospital.

Most Danish soldier cause of death is noted simple as “fallen” (“Faldet” in Danish). This is a broad but implies as quick death in action. “Injury” (“Kvæstelser” in Danish) is however almost as numerous. This is also a broad term, but it indicates that a great amount of the Danish soldiers died as a result of being severely wounded. Properly after the fighting had stopped.

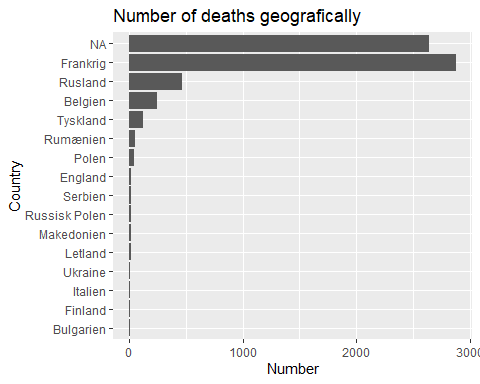
I also notice that a substantial amount of the Danish soldiers died from “sickness” (“Sygdom” in Danish). Or is simply noted as “missing” (“Savnet” in Danish). I will return to this later.

**Where Danish soldiers saw action**

In order to find out where the Danish soldiers saw action, I modified my code so that it lists the top 15 most common words in the column “country at time of death” (“Dødsland” in Danish).

faldne %>%  
 count(Dødsland, sort = TRUE) %>%  
 top\_n(15) %>%  
 mutate(Dødsland = reorder(Dødsland, n)) %>%  
 ggplot(aes(x = Dødsland, y = n)) +  
 geom\_col() +  
 xlab(NULL) +  
 coord\_flip() +  
 labs(x = "Country",  
 y = "Number",  
 title = "Number of deaths geografically")

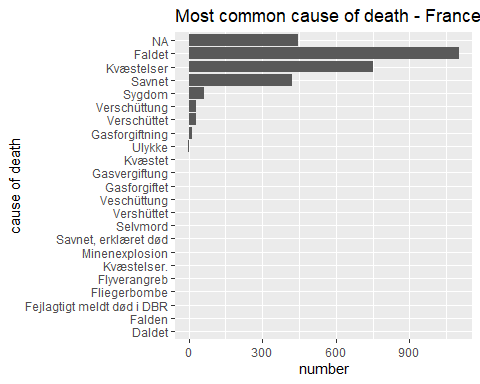
## Selecting by n



# Cause of death in various countries

faldne %>%  
 filter(Dødsland == "Frankrig") %>%   
 count(`Døds-årsag`, sort = TRUE) %>%  
 top\_n(15) %>%  
 mutate(`Døds-årsag` = reorder(`Døds-årsag`, n)) %>%  
 ggplot(aes(x = `Døds-årsag`, y = n)) +  
 geom\_col() +  
 xlab(NULL) +  
 coord\_flip() +  
 labs(x = "cause of death",  
 y = "number",  
 title = "Most common cause of death - France")

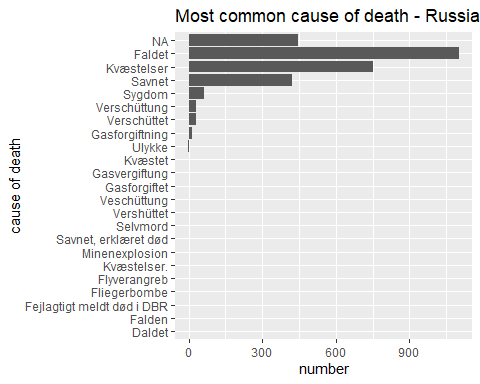
## Selecting by n



#Course of death in Russia

faldne %>%  
 filter(Dødsland == "Frankrig") %>%   
 count(`Døds-årsag`, sort = TRUE) %>%  
 top\_n(15) %>%  
 mutate(`Døds-årsag` = reorder(`Døds-årsag`, n)) %>%  
 ggplot(aes(x = `Døds-årsag`, y = n)) +  
 geom\_col() +  
 xlab(NULL) +  
 coord\_flip() +  
 labs(x = "cause of death",  
 y = "number",  
 title = "Most common cause of death - Russia")

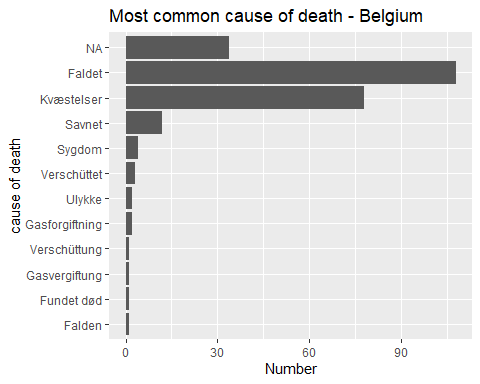
## Selecting by n



#Cource of death in Belgium

faldne %>%  
 filter(Dødsland == "Belgien") %>%   
 count(`Døds-årsag`, sort = TRUE) %>%  
 top\_n(15) %>%  
 mutate(`Døds-årsag` = reorder(`Døds-årsag`, n)) %>%  
 ggplot(aes(x = `Døds-årsag`, y = n)) +  
 geom\_col() +  
 xlab(NULL) +  
 coord\_flip() +  
 labs(x = "cause of death",  
 y = "Number",  
 title = "Most common cause of death - Belgium")

## Selecting by n

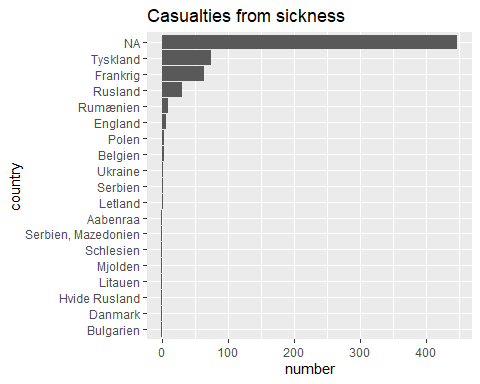


**Casualties from sickness**

In all three countries it is a very small percentage which dies from sickness. That did not seem right as we saw in the overall view of all casualties that sickness claimed almost as many lives as the category “missing”. (“Savnet” in Danish). I therefore wanted account for all the casualties caused by sickness.

faldne %>%  
 filter( `Døds-årsag` == "Sygdom") %>%   
 count(`Dødsland`, sort = TRUE) %>%  
 top\_n(15) %>%  
 mutate(`Dødsland` = reorder(`Dødsland`, n)) %>%  
 ggplot(aes(x = `Dødsland`, y = n)) +  
 geom\_col() +  
 xlab(NULL) +  
 coord\_flip() +  
 labs(x = "country",  
 y = "number",  
 title = "Casualties from sickness")

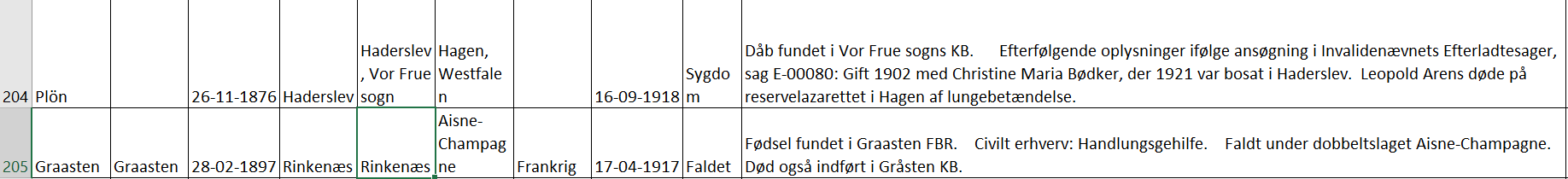
## Selecting by n



The vast majority is not accounted for. Which makes it hard to say anything definitive. It is however noticeable that Germany (“Tyskland” in Danish) has more deaths than France. This might indicate that sick and wounded soldiers often were transported away from the frontline in France and Belgium and back to Germany for treatment. This does however require more research.

It is worth mentioning that the place of death actually often appears in the column “indtasters bemærkninger” (“reseacher´s remarks” in English). In the case below it is noted that Leopold Arens died from pneumonia at the reserve hospital I Hagen. Despite this it is not noted that he died in Germany.

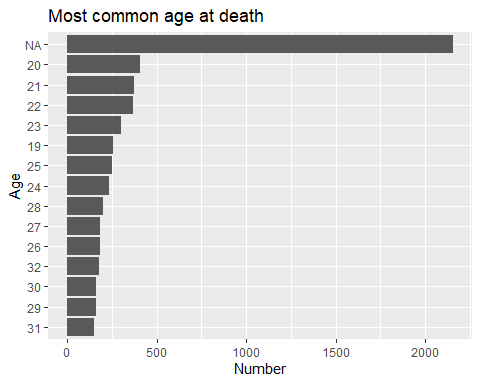
This goes to show that the digital tools I use is not perfect in any way. These methods can however be used to guide further research.



# Most common age at the time of death

faldne %>%  
 count(`Alder ved død`, sort = TRUE) %>%  
 top\_n(15) %>%  
 mutate(`Alder ved død` = reorder(`Alder ved død`, n)) %>%  
 ggplot(aes(x = `Alder ved død`, y = n)) +  
 geom\_col() +  
 xlab(NULL) +  
 coord\_flip() +  
 labs(x = "Age",  
 y = "Number",  
 title = "Most common age at death")

## Selecting by n



**6 Critical evaluation**

* Comparison with the state-of-the-art software if any **(**kindly cite relevant work, scripts, etc.**)**
* Evaluation of the learning process, time on task, vis-à-vis the product.

**7 Conclusions**

* Set out the conclusion of this original software publication

***\*Acknowledgements***

* *Optionally thank people and institutes you need to acknowledge*

***References***

* *At least 5 are required*

**B- Required Metadata**

**B1 Current executable software version**

*Ancillary data table required for sub version of the executable software: (x.1, x.2 etc.) kindly replace examples in right column with the correct information about your executables, and leave the left columns as they are*

*Table 1 – Software metadata*

|  |  |  |
| --- | --- | --- |
| **Nr** | **(executable) Software metadata description** | ***Please fill in this column*** |
| S1 | Current software version | *for example 1.1, 2.4 etc.* |
| S2 | Permanent link to executables of this version | *example : https://github.com/combogenomics/DuctApe/releases/tag/DuctApe-0.16.4* |
| S3 | Legal Software License | *List one of the approved licenses* |
| S4 | Computing platform / Operating System | *for example Android, BSD, iOS, Linux, OS X, Microsoft Windows, Unix-like , IBM z/OS, distributed / web based etc.* |
| S5 | Installation requirements & dependencies |  |
| S6 | If available Link to user manual - if formally published include a reference to the publication in the reference list | *Example http://mozart.github.io/documentation/ or* |
| S6 | Support email for questions |  |

**B2 Current code version**

*Ancillary data table required for subversion of the codebase. Kindly replace examples in right column with the correct information about your current code, and leave the left columns as they are*

*Table 2 – Code metadata*

|  |  |  |
| --- | --- | --- |
| **Nr** | **Code metadata description** | ***Please fill in this column*** |
| C1 | Current Code version | *For example v42* |
| C2 | Permanent link to code / repository used of this code version | *For example : https://github.com/mozart/mozart2* |
| C3 | Legal Code License | *List one of the approved licenses* |
| C4 | Code Versioning system used | *For example svn, git, mercurial, etc. put none if none* |
| C5 | Software Code Language used | *For example c++, python , r, etc.* |
| C6 | Compilation requirements, Operating environments & dependencies |  |
| C7 | If available Link to developer documentation / manual | *For example : http://mozart.github.io/documentation/* |
| C8 | Support email for questions |  |

1. <http://hax.odsbjerg.dk/twitter_scrape.html> visited 30-11-2019 [↑](#footnote-ref-1)