

# Digital Methods: Learning Journal Template

Magnus Lindhardt Iversen

Autumn 2020

# 1 30/10/2020

## 1.1 Thoughts / Intentions

**11:04am:** This is my first time doing a digital method course and i'm looking forward to learning new stuff about coding and computer skills. This journal will be some kind of diary where i can put my thoughts down on paper and try to learn from my working progress. For this exercise i will be using overleaf which will be my first time using it. Trial and error will probably be the keywords for this course.

**1. Homework:** 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. Homework:** Write a regular expression to convert the stopwordlist from Voyant in <http://bit.ly/regexexercise3> into a neat stopword list for R (where words are surrounded by "word" and separated by commas, such as <http://bit.ly/regexexercise4>). Then take the stopwordlist from R <http://bit.ly/regexexercise4> and convert it into a Voyant list (words on separate line without interpunction)

## 1.2 Action

### 1. Homework

- I click on the link and pressed ctrl a and then ctrl c to copy all of the text and put it in regex101.com.
- The regular expression I used to extract all of the dates were: (backslash d+).(backslash d+)..?(backslash.tuborg sign (4))
- The()’s job i to classify the date, month and year so i have three groups.
- The d matches any single digit and the + will capture more.
- The dot will capture any kind of sign or space.
- The question mark matches when preceeding characters occurs zero or one time
- The tuborg sign with number 4 will catch any four digit number = the year in this case
- So in this way i now have all the dates and then i click substitution and use the dollarsign plus the number 1,2 and 3 to divide the numbers into day, month and year. After that I can move one group lets say the year to stand first and then it will automatically move the year to stand first.

### 2.Homework

- First i take the list from voyant and make it into a stopword list to R.

- So i go to the link and cypypaste all of it into regex101.
- Then i type in (" , ")
- The () is to capture the group with "" ,comma and space.
- Then i type backslash n and i get all the words seperated. the n will match a newline character and divide the words. it its under substitution i type in this.
- I tried to take the stopwordlist from R and Convert it into a voyant list.
- I capture all of the words by typing backslah w+, but then I still miss some danish letters as æ,ø,å. and i dont know how to fix this.. and sadly i dont know how to arrange it into a vojant format aswell.
- Okay, after conslting with my team, i found out that its suppose to be like this: (.\*dollarsign)backslash n
- because . captures all signs and \* makes it doing this from zero to unlimited times. so now we have all letters and numbers included.
- the backslash n finds the ending of a word.
- after this you type the following under substitution: "dollarsign1" ,.And you have now organized it into a voyant format. the dollarsign 1 transform the words into one single group of words.

### 1.3 Final Thoughts

The first lecture and homework has indtroduced me to a lot of new stuff. especially a lot of new programs that i need to learn as we go. For todays homework i still need to learn the basic coding skills, which is a bit difficult and its hard to keep the bigger picture clear. I think more and smaller task with some good tutorials good help a lot. But we will see how it goes next week. over and out for today. Note to self: try to work with my group next week.

## 2 8/11/2020

### 2.1 Thoughts / Intentions

This is the second time doing homework in this course. And this weeks lesson was about spreadsheets and how to use openrefine. It was nice only to use one program this time, but still i could have used more time learning the program. Still I learned a lot from the lesson and also from the tutorial, and i think i would be able to use this program for my exam if i used some more time learning it.

**1. Homework:** Answer the question of: "What are basic principles for using spreadsheets for good data organisation?"

**2. Homework:** Does OpenRefine alter the raw data during sorting and filtering?

**3. Homework:** Fix the interviews dataset in OpenRefine enough to answer this question: "Which two months are reported as the most water-deprived/driest by the interviewed farmer households?"

**4. Homework:** Finish the tutorial and report what village \*you\* think hides behind the number '49'.

### 2.2 Action

#### 1. Homework

- To handle your data it is a good idea to make them as easy to understand and read for everyone. So make the data manageable by sorting your data in a good way.
- Variables should get their own column and observations their own row.
- Never change the raw data, always have a backup.
- Only make one or few tables in one spreadsheet and after that make another window if you want to make more.
- Be consistent and transparent in your work. Have the same standard procedure.

#### 2. Homework

- No, Openrefine always keep the original data.

#### 3. Homework

- To answer this question i went to the column with month-no-water and then i clicked faced, and text facet. Then a box pops up and I can see that September and October are the most driest month according to the farmers.

#### 4. Homework

I finished the tutorial and followed the steps through. To figure out what village was under the number '49' I put the Village categories next to the longitude and altitude, and by sorting them like the tutorial said I could see that the cities longitude and altitude was the same as Chirodzo, so Village number '49' is Chirodzo.

### 2.3 Final Thoughts

I found this weeks topic easier than last week and with a higher yield. I think i understood this program faster and it seems useful in todays homework.

## 3 13/11/2020

### 3.1 Thoughts / Intentions

This week was about learning shell. I like this weeks topic and found it rather easier than earlier lectures. But I am still not sure when it will be that useful. So for now it was just an introduction, that was good, but i don't think i will be using it moving forward. In my opinion and with my knowledge i think the program is not that relevant for history student, but maybe for other things. But nonetheless it's always fun to learn something new, and maybe it will be useful after today's homework, so we have a real example to learn from.

**Homework:** Your supervisor has shared a folder of photos on Science-data.dk with you (password is 2020CDS, folder is 500Mb and contains 189 images) and needs your help with a couple diagnostics:

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)

2) Some of the image files are empty, a sign of error in the data processing or 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?

### 3.2 Action

- First I download the files from the link above and unfold it to my desktop. After that I right click on the files with the photo and click "get to git bash".
- Now i am in the files case and can begin finding the biggest files.
- Okay first I press `ls` to see what in this file case = all JPG files
- Then i make a `wc` (word count) on all the JPG files to see what photo is the largest. After that I create a new text for this list called `size`.
- It all looks like this: `wc -l *.JPG > size.txt`
- Now i want the biggest files first, so i sort the list i just created.
- `sort -nr size.txt` = numerically ordered list
- I now have the three biggest files first and they are all JPG
- 1. 9254.
- 2. 9231
- 3. 9232

I am now going to do the second task of today

**2. Homework:**

- okaaaay, so after trying going through the tutorial and using "grep" and "find" to find out how to select all the empty files i didn't succeed.. BUT ofc I just googled it, and 2 sec after i got the answer for my q!! so NOTE TO SELF: start with google because its much more effecient!
- So what i did was: I just wrote "find size -0" and now i got my list with files with 0 kb size.
- after that i made a list with their name by doing this: find size -0 & nulldata.txt
- and how to count them was done by doing this: wc -l nulldata.txt = 76 nulldata files.

### 3.3 Final thoughts

I am really glad with todays homework because i manage to do them all very well and even though i struggled sometimes i found the answers myself by asking google and trying and trying. So good experience with this program, its easy to use and easy find help and i like that. I still dont know if i'm gonna use this program again, but i think i would be more likely to use it now compared to just 2 ours before when i started doing these homework. One thing i didn't learn today is how to insert images.. i tried and found a manual but it still didn't work. So next time i'm in the class i will ask someone :)