Bussod Ilona and Fribourg Melvin

Monday 11th of May 2020:

That first day was organized into 4 parts:

1) Meeting and establishment of our goals: we had a meeting with Bastian Greshake Tzovaras in order to fill our roadmap, and talk about what we are going to do for this 3 weeks of CRI Fellows project (SCORE). To do so, we navigated on Bastian's website named QuantifiedFlu.
   Then we downloaded all the CSV files (ongoing symptoms data and the retrospective data) from the website that we are going to use in order to make an analysis. Those analyses and our work will be reported in several formats: graphs, blog pages, and this daily report.
   But what are those CSV files? They contain all illness-related episodes (such as symptoms, temperature, heart rate, cough…) reported by people filling either form on a daily basis or sharing their data trough Fitbit or Oura rings wearables (data from Google Fit and Apple Watch will be also soon collected).

2) Implementation of a word cloud: our first mission was to create a word cloud made of comments from the dataset of people who participate in the project. From that representation, the main goal is to observe if one word is more common than another. Indeed, associate comments and symptoms could inform us about similarities in illness episodes.
   The word cloud below was generated on Rstudio by using the data of 44 files from 'Ongoing symptom tracking data' available on the website. We also checked some code information on the website stdha.

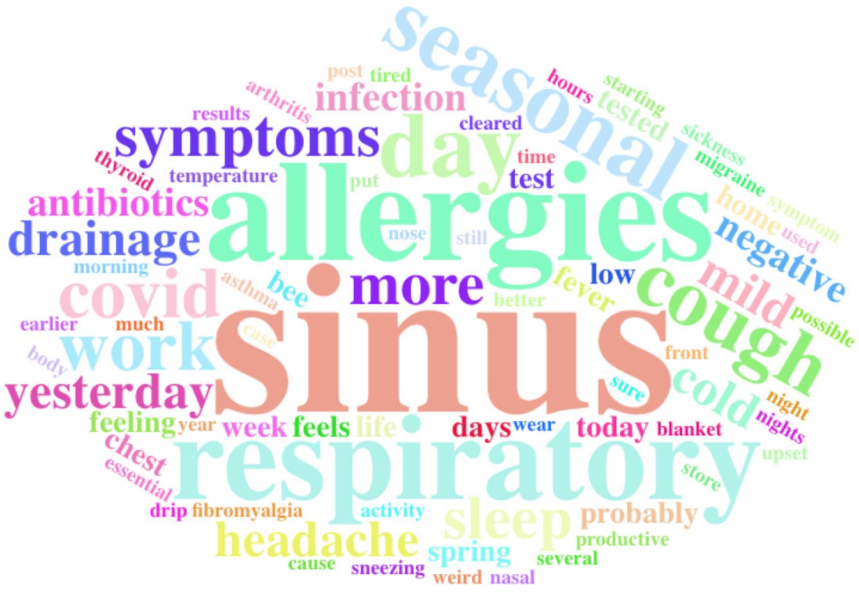


Figure 1: Word Cloud list of Ongoing symptom tracking data (CSV files)

The R Code we used is available below:

```
library("tm")
library("SnowballC")
library("RColorBrewer")
library("wordcloud")
```

```
library("wordcloud2")
library('dplyr')

text <-
readLines(file.choose("~/Documents/FdV/L2/S2/Bastian/Wordcloud.txt"))
docs <- Corpus(VectorSource(text))

dtm <- TermDocumentMatrix(docs)
matrix <- as.matrix(dtm)
words <- sort(rowSums(matrix),decreasing=TRUE)
data <- words[1:90 ]
df <- data.frame(word = names(data),freq=data)

wordcloud2(data=df, color='random-light', minRotation = -pi/6,
  maxRotation = -pi/6, minSize = 10,  size = 1)
```

3) Creation of new visualizations: we then focused on the 'retrospective dataset' in order to create new visualizations for the datasets. We first thought that stacked area charts could be interesting in order to compare Fitbit and Oura Ring wearables. Indeed, we would like to analyze if the data collected by those two different brands of wearables capture in a different way heart rate or temperature. However, we faced some issues. The 1st one is the consequent amount of data (more than 5000 for one person) so R codes are taking quite a long time to run. The second one is the issue of the time variable written as '2020-03-25T22:35:46.860000+00:00' which is not considered as a continuous variable on R but as a discrete one. Nevertheless, we did not solve that problem yet, notably because of the double information in the time variable: date and hour.

4) What's next? Finally, we defined our next objectives. One goal will be to create a word cloud in order to make links between comments and symptoms in the future days and observe if some comments are more related to symptoms than others. We would also like to separate day and night data in other visualizations formats in order to compare the influence of sleep on heart rate and in order to have something more visual. Another wish would be to model a correlation matrix so as to observe the links between some symptoms and if they are naturally related (for instance, with seasonal flu, most people have cough and fever at the same time). Moreover, because we joined the Open Humans Community on Slack under the recommendations of Bastian, we are waiting to exchange and share what we are doing in order to have feedback and think about other ideas.

Tuesday 12th of May 2020:

1) Second meeting and establishment of our goals of the day: we had our second meeting with Bastian in order to talk about the word cloud we created yesterday and what we thought it would be interesting to do for the rest of the day. Moreover, Bastian indicated that we did not have data during the daily life of people but only during the night (because day data could be altered by other parameters such as sport, food, feelings...). Taking this into consideration, we forgot the idea to separate day and night data in order to compare the influence of sleep on heart rate as said yesterday.

2) Word Cloud: As said before, we discussed the word cloud (figure 1) that we made Monday 11th, and we discussed how it would be interesting to create others word clouds to see if some comments are more related to symptoms than others (we talked about it in part 4 on Monday 11th). To do so, we took all the comments of data givers according to their symptoms, that we divided into 3 different graphs: people who have allergies, people who suspected to have Covid-19, and people who suspected to have a cold.

Figure 2:

Word Clouds on comments of data givers, according to their symptoms ("Allergies", "COVID", "Cold")

We can see several words according to data givers symptoms. We can deduce that the vocabulary people are employing is different according to their symptoms. Someone getting allergy episodes is more going to speak about "cough", "headache" or "sinus" than someone who is positive to Covid-19 or someone who is having cold episodes. The R code we used is the same as yesterday by putting different text files (.txt).

3) Our first idea of visualization: We then focused on the 'Retrospective dataset' and more particularly on a file named '5.CSV'.
This file contains heart rate values for different data type:
   - "Fibtit_intraday": every 5 minutes during the night of the person, the Fitbit wearable captured the heart rate
   - "Fitbit_summary": each day, the Fitbit wearable gave a mean value for the heart rate

We had to adopt a methodology in order to obtain our first visualizations. Indeed, for the 'Fitbit_intraday' data, the variable 'timestamps' was written as '2020-02-17 23:30:00+00:00'. Thanks to Bastian, we saw on that website that we had to use the lubridate library in R, in order to have our 'timestamps' variable into a date format (year-month-day hour-minute-seconds).
However, for the 'Fitbit_summary' data, the same variable 'timestamps' was written as '2020-02-17' so only like year-month-day.

We had to separate those 2 datasets because we had to use 2 different functions from the lubridate library and so our two different formats of timestamps variable could be considered as dates. We generated the first visualization as a line chart to analyze more easily the dynamics of the heart rate given by the fitbit_summary.

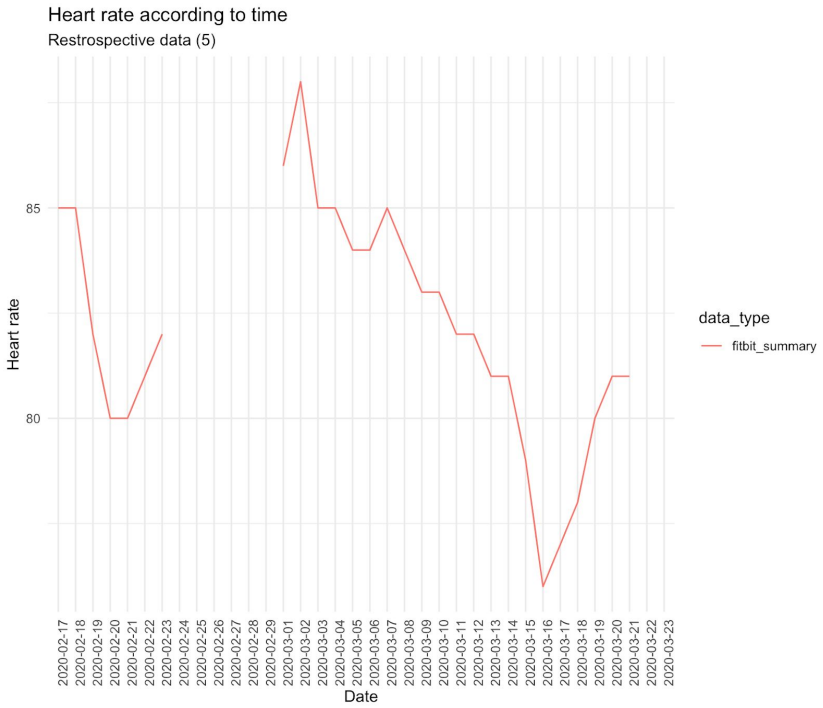


Figure 3:

Line chart of the fitbit_summary representing the heart rate according to the time (day by day) of the retrospective data (file "5.csv")

We can observe a gap between 2020-02-23 and 2020-03-01 due to the lack of data. We have two hypotheses from that observation: either the Fitbit wearable did not capture data for those dates or the concerned person forgot to wear it.

For doing that graph, we used a R code, available below:

```
library(ggplot2)
library(dplyr)
library(lubridate)

data <- read.csv("~/Documents/FdV/L2/S2/Bastian/Retrospective data/5.csv",
sep=";")
#Fitbit_Summary
data2 <- data %>%
  filter(data_type == 'fitbit_summary')%>%
  select(timestamp, value, data_type)
ymd(data2$timestamp)

ggplot(data2, aes(x=timestamp, y=value, group=data_type, color=data_type))
+
  geom_line() + theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  labs(title = "Heart rate according to time",
```

```
subtitle = "Retrospective data (5)",
x = "Date", y = "Heart rate")
```

We also made a graph for the Fitbit_intraday. However, we encountered some issues related to the date format. As we can see, the date format as '2020-02-17 23:30:00+00:00' is taking a lot of character place in the x-axis legend and is plotting as a black line.

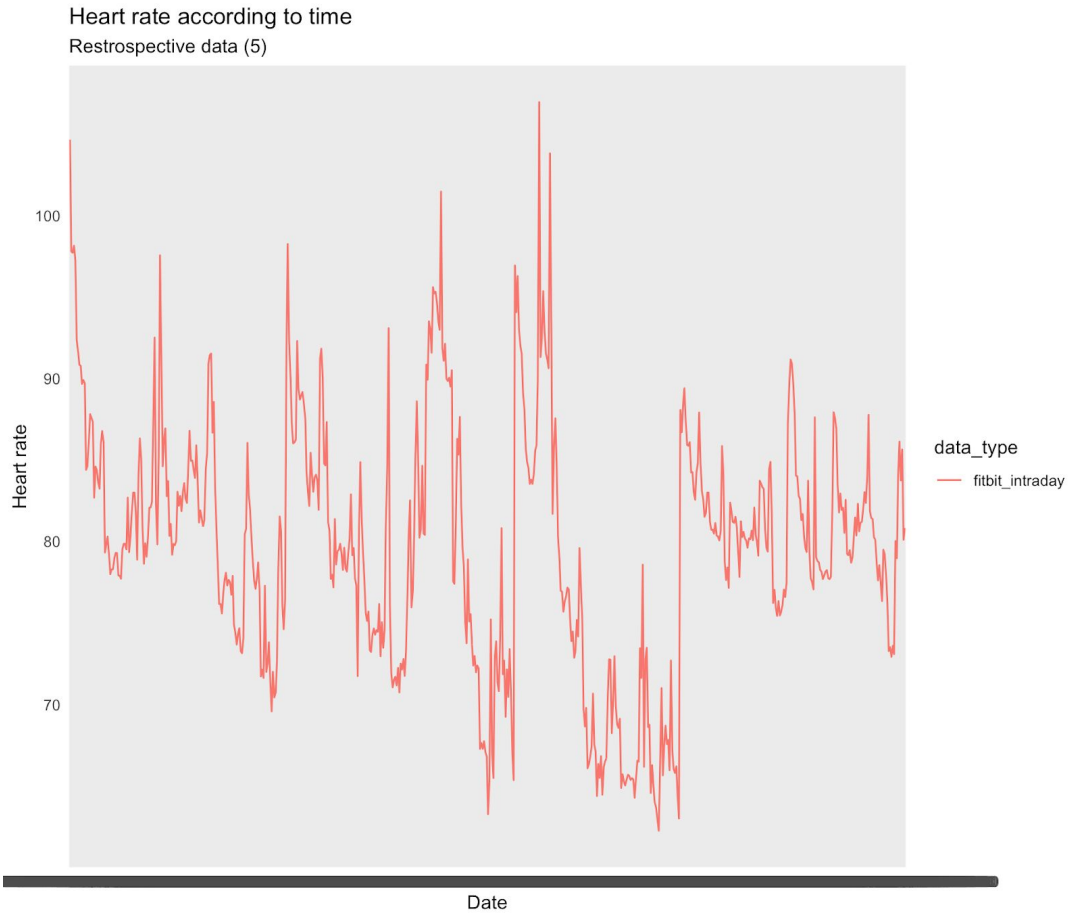


Figure 4:

Line chart of the fitbit_intraday representing the heart rate according to the time (day by day) of the retrospective data (file "5.csv")

The R code we used is the same as Figure 3 by selecting fitbit_intraday instead of fitbit_summary.

4) Our second idea of visualization: one of our wishes was to represent a correlation matrix in order to observe the links between some symptoms and some diseases like cold, flu, or Covid-19. To do so, we first created a test matrix to correlate symptoms and disease with a probability of correlation in a range from 0 (no correlation) to 1 (high correlation) made in an empiric way. Then, we implemented this matrix in RStudio by representing it with an ellipse correlogram as presented in Figure 5.

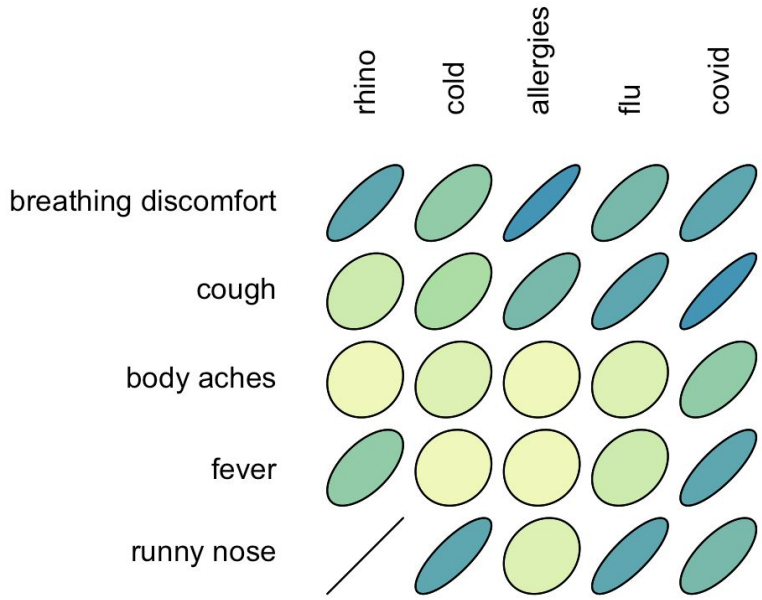


Figure 5: Ellipse Correlogram linking symptoms and diseases

In order to interpret this graph, the more the ellipse is thin and the color is dark, the more the correlation between the symptoms and the disease is high. We can then analyze, according to our empiric test matrix, that rhinopharyngitis is highly correlated with a runny nose and that allergies are really correlated with breathing discomfort for instance.

The R code used for the ellipse correlogram is right below:

```
library(ellipse)
library(RColorBrewer)
library(corrgram)

ilo <- matrix(c(0.5, 0.1 ,0.8, 0.20, 0.6, 0.8, 0.3, 0.8,0.2, 0.7, 0.9,
                0.8, 0.7, 0.6, 0.8, 0.7, 0.1, 0.2, 0.1, 0.9, 0.3, 0.6,
                1, 0.1, 0.8),
           nrow = 5, ncol = 5,
           dimnames = list(c("cough","fever",'runny nose','body
aches','breathing discomfort'),
                           c("cold","flu",'covid','allergies','rhino')))

my_colors <- brewer.pal(5, "Spectral")
my_colors <- colorRampPalette(my_colors)(100)

ord <- order(ilo[1, ])
data_ord <- ilo[ord, ord]
plotcorr(data_ord , col=my_colors[data_ord*50+50] , mar=c(1,1,1,1))

corrgram(ilo, order=TRUE, lower.panel=panel.shade, upper.panel=panel.pie,
```

```
      text.panel=panel.txt, main="Correlation between symptoms and
diseases")
```

5) What's next? On one hand, we would like to improve our word clouds in several ways: delete punctuations that are still present, try different color palettes but also reorganize the word clouds in order to represent a letter to have a prettier representation. For instance, we would like to transform figure 1 in the shape of the letter 'QF' (like Quantified Self).

   On the other hand, we have to focus way more on the dataset from different wearables and think about other kinds of graphs in order to make comparisons but also to analyze the data. We thought about creating an interactive graph that can be downloaded as a GIF. However we encountered some issues with the library 'hrbrthemes' so we will have to do way more research before having the result we want. For now, the main issue which has to be solved in the date format in the x legend axis as explained in part 3) in Figure 4. We also thought about the temperature value in some dataset which is in Fahrenheit and we would have the wish to put it in Celsius or maybe do both.

   Moreover, about the ellipse correlogram we can clearly say that the representation is not logical at first sight. Naturally, we would like to interpret the graphs as the more the ellipse is large and the color is dark, the more the correlation between the symptoms and the disease is high. Bastian shared a link where we could try new correlation graphs and we will try to implement these ones with the dataset from the Quantified Flu website.

Wednesday 13th of May 2020:

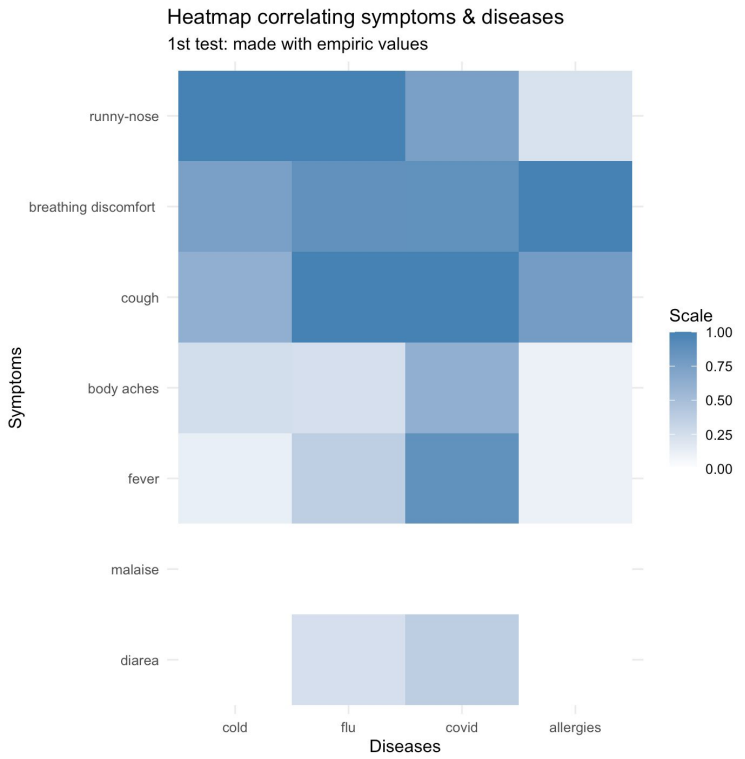
1) **Letter clouds:** According to the idea of the previous days, we improved our word clouds by generating letters Clouds of data givers' comments, according to their symptoms, in order to have a prettier representation for the word clouds. We had some issues with the RStudio viewer, indeed, most of the time, the visualization window didn't show the graph even if the code was correct. We found on the internet that a lot of people encountered the same issues: it comes from the display window which cannot show those graphs considered as 'too big'. We then had to reload and extend the window in order to obtain our plots.

Figure 6: *Letter Clouds on comments of data givers, according to their symptoms (QF for general comments of the website Quantified Flu, A for Allergies, 19 for COVID-19, C for Cold).*

For doing those representation, we used the R code, available below:

```
library(devtools)
library(np)
library(tm)
library(SnowballC)
library(RColorBrewer)
library(wordcloud2)
library(dplyr)

text <- readLines(file.choose("~/Bureau/WordcloudCold.txt"))
docs <- Corpus(VectorSource(text))
dtm <- TermDocumentMatrix(docs)
matrix <- as.matrix(dtm)
words <- sort(rowSums(matrix),decreasing=TRUE)
data <- words[1:57]
demoFreqC <- data.frame(word = names(data),freq=data)

letterCloud(demoFreqC,word = "C", color = "random-light",
                    backgroundColor = "white",size =1)
```

2) **<u>Heatmap made on empiric values:</u>** Instead of the ellipse correlogram made yesterday, we, under the advice of Bastian, made a heatmap that we can see below.

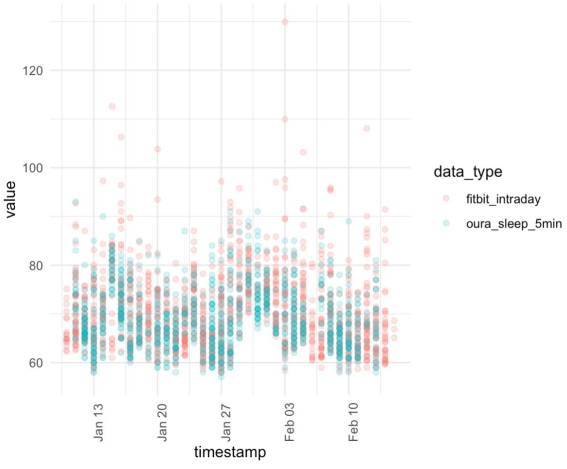


<u>Figure 7:</u> Heatmap by our empiric values

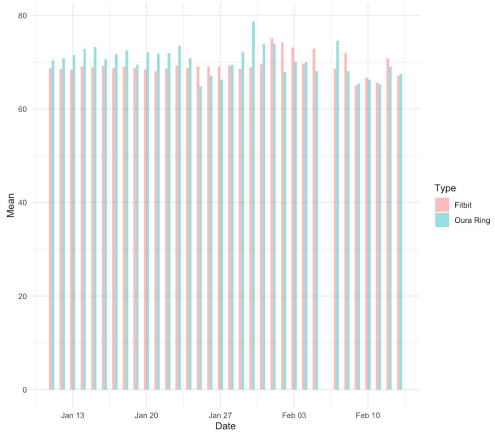
Now, we can easily observe that the more a case is dark, the more symptoms and diseases are related.
The R code we used in order to make the heatmap on Figure 7 is:

```
library(ggplot2)
library(dplyr)
library(reshape2)
library(plyr)
library(plotrix)
library(scales)

data <- read.csv("~/Downloads/correlo.csv")
data$X <- with(data,reorder(X, cold))
data.m <- melt(data)
data.m <- ddply(data.m, .(variable), transform, Scale=rescale(value,
0:100))

ggplot(data.m, aes(variable, X)) +
  geom_tile(aes(fill=Scale)) +
  scale_fill_gradient(low='white', high='steelblue') + theme_minimal() +
  labs(title = "Heatmap correlating symptoms and diseases",
      subtitle = "1st test: made with empiric values",
      x = "Diseases", y = "Symptoms")
```

3) **<u>Visualizations to compare data:</u>** In order to compare the Fitbit and the Oura ring, we decided to plot some visualizations in Rstudio. We first thought about the geom_point function, however, the data we took were those from fitbit_intraday and oura_sleep_5min so with a 5min scale. That made a graph with a lot of dots, something unreadable as we can see in Figure 8.



<u>Figure 8:</u> *Geom_point function*



<u>Figure 9:</u> *Geom_bar function*

Then, we calculated the average, by day (more accurate to say by night) so for each date, for fitbit_intraday and oura_sleep_5min for the 1.csv file in the retrospective dataset in order to compare

more easily the heart rate values of those two wearables to see if there are any differences. We thought about a geom_bar function in order to see the gap between highest values and lowest ones, however, we did not find it pertinent as we can see in Figure 9. Indeed, the main information we want to analyze, i.e. is there a difference between heart rate values captured by different wearables, is difficult to answer.
Finally, we decided to plot the graph by the geom_line function by using the same averages made previously and we obtained Figure 10.

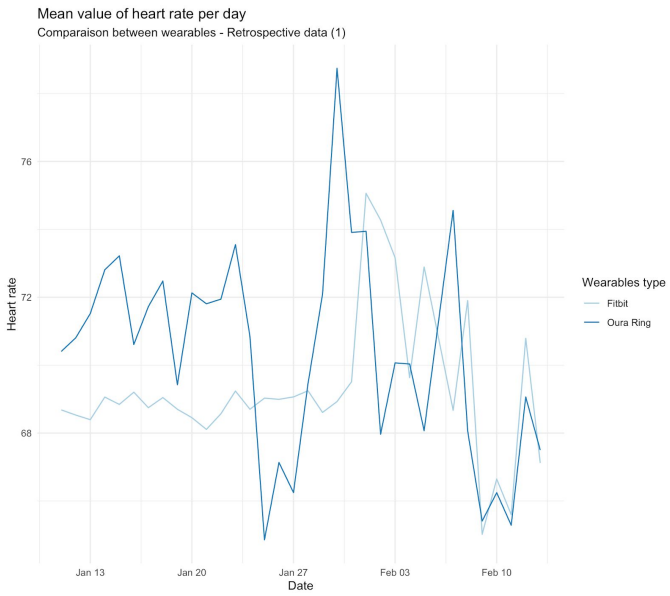


Figure 10: *Geom_line function*

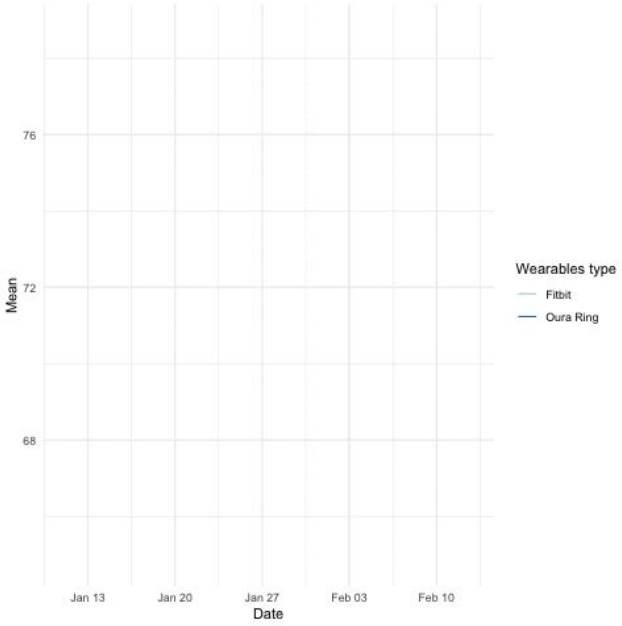


Figure 11: *GIF of the Geom_line function*

We also made a GIF in order to have something more visual in Figure 11. We can see that there is quite a difference between the average values of those two wearables. Indeed, around Feb10 both are quite similar but however, the dynamics are not really looking the same. We also asked ourselves if making mean values for each date was something relevant.

The R code we used in order to make the geom_line function on Figure 10 is right below:

```
library(ggplot2)
library(tidyverse)
library(dplyr)
library(tidyr)
library(lubridate)
library(readr)
library(viridis)

file <- read.csv("~/Documents/FdV/L2/S2/Bastian/Plot1/meanvalues.csv",
sep=";")

#Fitbit_intraday + Oura_sleep_5min
file$Date <- as.Date(file$Date, '%Y-%m-%d')
```

```
ggplot(file, aes(x=Date, y=Mean, group=Type, color=Type)) +
  geom_line() + theme_minimal() +
  scale_color_brewer("Wearables type",palette="Paired")+
  labs(title = "Mean value of heart rate per day",
      subtitle = "Comparison between wearables - Retrospective data (1)",
      x = "Date", y = "Heart rate")
```

4) **What's next?** We then discussed with Bastian about the geom_line analysis that we made and the relevance of it. He told us to use the resting heart rate (which is the lowest heart rate per night) for fitbit_intraday and oura_sleep_5min instead of an average because it will be more interesting and closely related. This is one of our next steps. Moreover, about the heatmap we made, we have planned to do it with our dataset in order to model a more realistic visualization. As usual, we also want to find other graphic representations with other data to have better comprehension. For the next week, we also need to find problems to answer and organize our future research about one precise subject.

Thursday 14th of May 2020:

1) **Visualizations to compare data:** As Bastian advised us, we first did a geom_line with the resting heart rate (which is the lowest heart rate) and also with the maximum heart rate thanks to the data from the retrospective folder and particularly the file 1.csv. We considered, uniquely, the 'fitbit_intraday' variable and the 'oura_ring_5min' one. In order to compare those two wearables, we used the function geom_line in Rstudio.

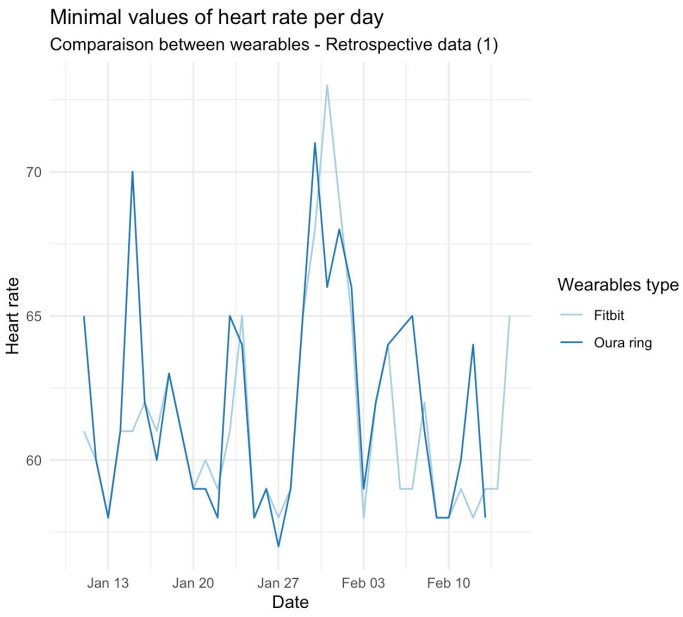


Figure 12: *Geom_line function of Fitbit and Oura Ring minimum values*

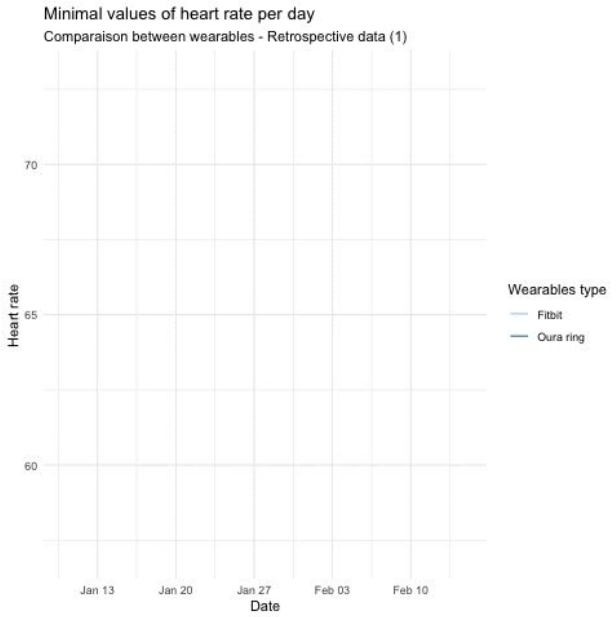


Figure 13: *Gif of geom_line function of Fitbit and Oura Ring minimum values*

As we can see in this graphic, the variation of the resting heart rate for the two wearables are pretty much similar, even if sometimes there are huge variations (for instance, the 15th January, Oura ring got a peak for the heart rate but not the Fitbit wearable).

Then, we did the same representation for the maximum values of heart rate in order to compare those two wearables again for the same dataset.

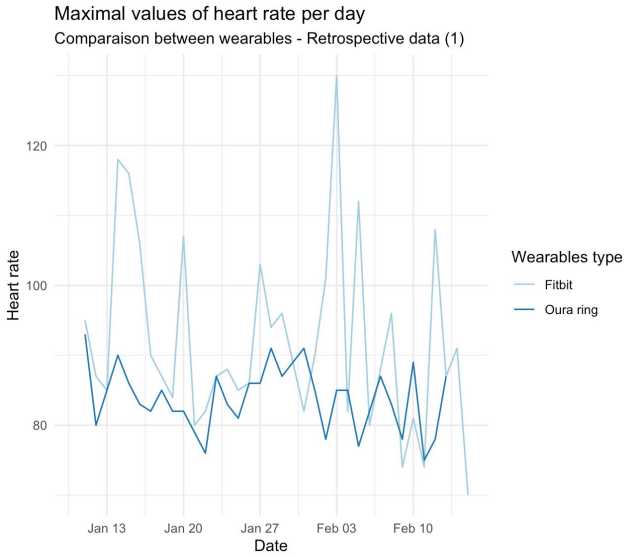


Figure 14: *Geom_line function of Fitbit and Oura Ring base on the maximum values of heart rate*

We can see a bigger variation for all the peaks for maximal values which are different from minimum values. Fitbit registers way higher values of heart rate than Oura Ring. We can submit the hypothesis that those two wearables do not have the same way of measuring heart rate; which is something crucial in the precision and reliability of the data. In order to plot the geom_line function we used the same R code as yesterday.
However, we had to precise the comparison we wanted to make and so we decided to make boxplots with all the heart rates available in the file '1.csv' taken in the retrospective data folder.

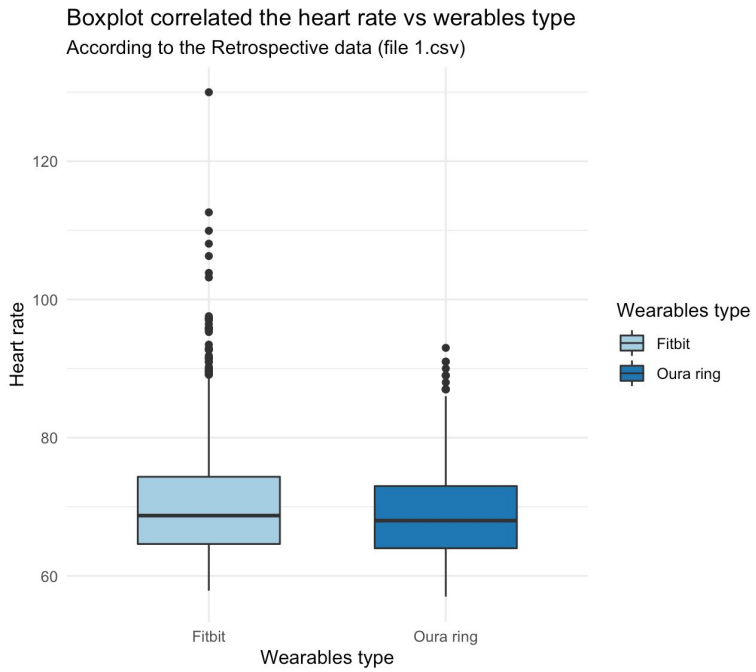


Figure 15: Boxplots comparing our two wearables for the heart rates data

As we can see, the median of the two boxplots are quite the same, there are just some differences between the maximum and minimum values (those from Oura ring wearable seems to be lower).

The R code we used to make the boxplots is:

```
library(ggplot2)
library(tidyverse)
library(dplyr)
library(tidyr)
library(lubridate)
library(readr)
library(viridis)

box <- read.csv("~/Documents/FdV/L2/S2/Bastian/Retrospective data/1.csv")

box <- box %>%
  filter(data_type == c('fitbit_intraday','oura_sleep_5min'))%>%
  select(data_type,value)

box$data_type[box$data_type == "fitbit_intraday"] <- "Fitbit"
box$data_type[box$data_type == "oura_sleep_5min"] <- "Oura ring"

ggplot(box, aes(x=data_type, y=value, fill=data_type)) +
  geom_boxplot(width=0.6)+ theme_minimal() +
  scale_fill_brewer("Wearables type",palette="Paired")+
  scale_x_discrete("Wearables type")+
  scale_y_continuous("Heart rate")+
```

```
labs(title="Boxplot correlated the heart rate vs wearables type",
   subtitle="According to the Retrospective data (file 1.csv)")
```

2) **Improvements of our heatmap:** as said yesterday, we planned to redo our heatmap with our dataset in order to model a more realistic visualization. We collected inside the 'ongoing symptoms data' folder all the medical conditions (diseases, viruses or health troubles) and their associated symptoms. To do so, we created a sheet where we indicated all the frequencies (in %) of symptoms (anosmia, body ache, ...) according to the medical condition they are related to (allergies, sinus, ...).

   The 'other' column corresponds to people who did not indicate their medical condition but indicated their symptoms. As we did not know where to list their values, we decided to create the other column. That column also contains all people's medical conditions that were linked to hard activities that require efforts such as: sport, gardening, intense work but also to medical conditions due to pathologies such as fibromyalgia. However, we wanted to precise that the data we used in order to do the heatmap is very low and many of the files were not containing any values so that we could have frequencies closer to reality. For instance, we can clearly see that cough symptoms are missing for COVID-19 or for rhinovirus which seems weird and not really accurate.

	allergies	sinus	migraine	other	cold	rhinovirus	covid-19	asthma
anosmia	10	0	0	0	0	0	0	0
body ache	40	0	50	27	33	0	100	0
chills	0	0	0	15	0	0	0	0
cough	28	0	32	67	100	0	0	100
diarrhea	25	0	0	36	33	0	100	0
ear ache	0	0	0	6	0	0	0	0
fatigue	15	0	18	58	33	100	0	0
headache	100	100	100	39	50	0	33	0
nausea	23	0	4	36	33	0	0	0
runny nose	83	100	54	33	17	0	0	0
short breath	3	0	0	15	0	0	0	0
sore throat	3	100	0	12	17	100	0	0
wet cough	45	100	36	21	17	0	67	0
stomach ache	5	0	0	0	0	0	0	0

Figure 16: Summary sheet of the frequencies of symptoms correlated to medical conditions

We implemented that table into a csv file and plotted the heatmap from it in Rstudio by using the code right below:

```
library(ggplot2)
library(dplyr)
```

```
library(reshape2)
library(plyr)
library(plotrix)
library(scales)

data <- read.csv("~/Documents/FdV/L2/S2/Bastian/Correlogram/heatmap.csv")

data$X <- with(data,reorder(X, cold))
data.m <- melt(data)
data.m <- ddply(data.m, .(variable), transform, Scale=rescale(value, 0:1))

ggplot(data.m, aes(variable, X)) +
  geom_tile(aes(fill=Scale)) +
  scale_fill_gradient(low='white', high='steelblue') + theme_minimal() +
  labs(title = "Heatmap correlating symptoms and diseases",
      subtitle = "Made with ongoing symptoms data",
      x = "Diseases", y = "Symptoms")
```

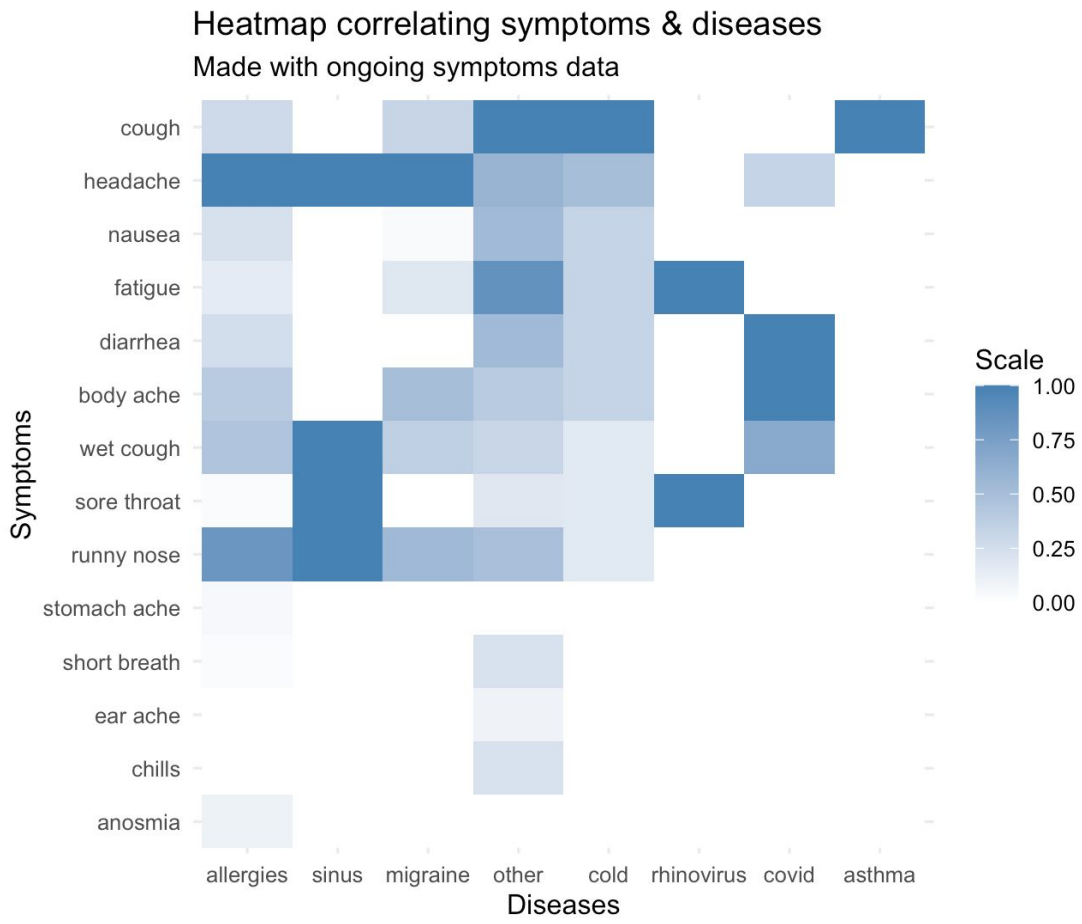


Figure 17: Heatmap made in order to compare symptoms and medical conditions based on 'ongoing symptoms data' folder

3) **What's next!** Bastian suggested us to see what will happen if we plot the difference min(fitbit_intraday) - min(oura_sleep_5min) per day and then, make a distribution of these

differences, as the boxplots are hard to compare as it is right now, because the data is not paired. Indeed, the Oura ring and Fitbit data could be in a completely different order and so that the difference between them every day will be very high (even if the boxplots would still be the same as long as the values in both datasets are similar). Then, our goal will be, for next monday, to define a precise problem we want to answer and to exploit all the data we need in a precise, constructive and detailed way with web research.

Friday 15th of May 2020:

1) **Geom_line of the difference between fitbit_intrada and oura_sleep_5min per day:** As Bastian suggested to us yesterday, we calculated the difference between the minimal values per day for fitbit_intraday and the oura_sleep_5min. We did the same for the maximum values. We then implemented those calculus in Rstudio and plotted the graphs below.

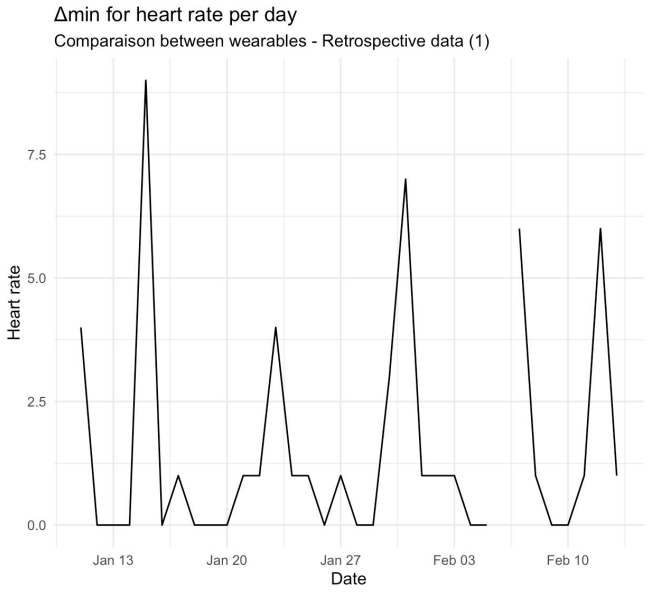


Figure 19: Geom_line of min(fitbit_intraday) - min(oura_sleep_5min) per day

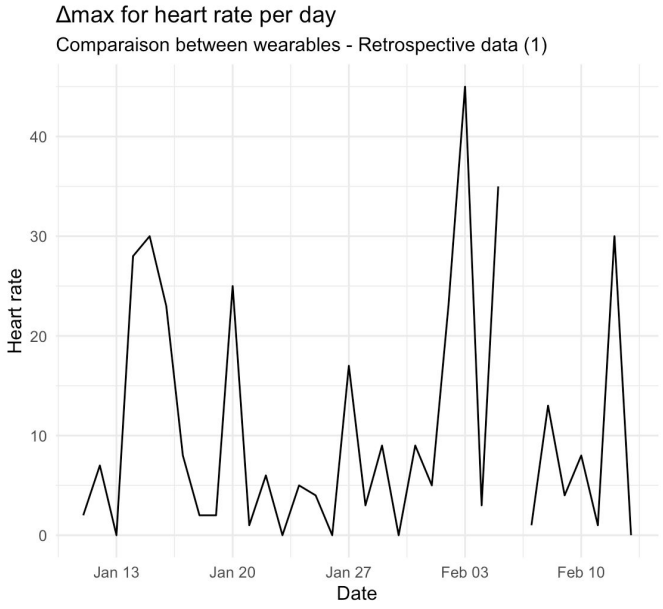


Figure 20: Geom_line of max(fitbit_intraday) - max(oura_sleep_5min) per day

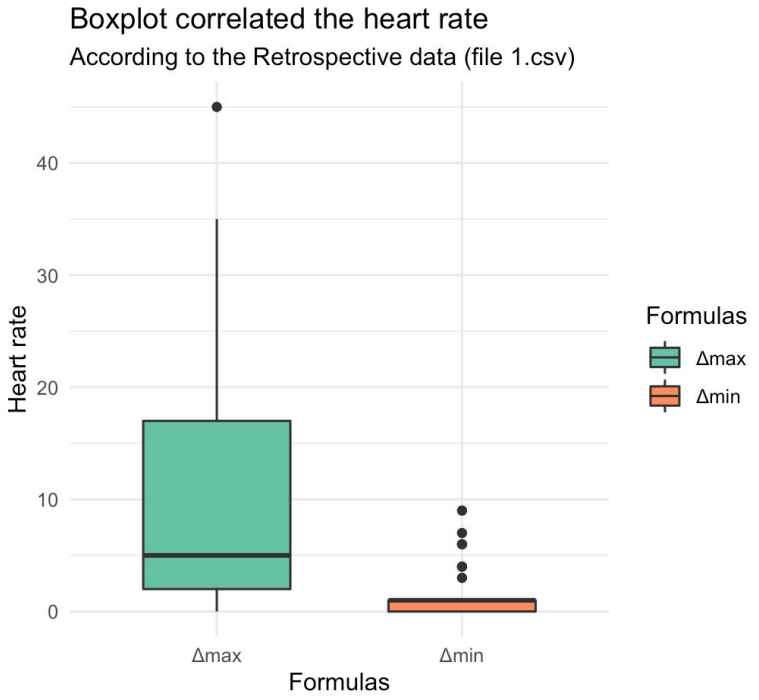


Figure 21: Boxplots of the Δmax and Δmax

In the figure 19, we can see that Δmin has low variations. Indeed the difference between those two wearables is between 0 and 8 hours whereas for Δmax in figure 20, the difference is between 0 and 45 hours approximately. We can deduce a big variability for the Δmax. This can be explained by the fact that the resting heart rate may be more stable and so captured in a similar way for the two wearables. The boxplots are summarizing these two observations. The variations of Δmax are way more bigger than those for Δmin.

2) **New heatmap with the correlation between symptoms:** Like yesterday, we redid a heatmap with our dataset to see the correlation between symptoms. We collected inside the 'ongoing symptoms data' folder all the symptoms, and compare the frequencies when other symptoms happened. We created a sheet where we indicated all the frequencies (in %) of symptoms (anosmia, body ache, …) according to them. The goal is to understand if some are always related to others.

	anosmia	body ache	chills	cough	diarrhea	ear ache	fatigue	headache	nausea	runny nose	short breath	sore throat	wet cough
anosmia	100	0	0	0	8	0	0	7	5	5	0	8	0
body ache	0	100	6	16	6	0	19	50	15	37	0	2	25
chills	0	6	100	0	8	0	11	6	0	3	0	0	0
cough	0	16	0	100	0	4	6	43	2	27	0	3	9
diarrhea	8	6	8	0	100	0	18	19	8	18	0	8	3
ear ache	0	0	0	4	0	100	0	0	4	0	0	0	0
fatigue	0	19	11	6	18	0	100	20	27	16	10	8	9
headache	7	50	6	43	19	0	20	100	31	70	0	2	26

nausea	5	15	0	2	8	4	27	31	100	18	6	7	2
runny nose	5	37	3	27	18	0	16	70	18	100	0	2	22
short breath	0	0	0	0	0	0	10	0	6	0	100	2	0
sore throat	8	2	0	3	8	0	8	2	7	2	2	100	2
wet cough	0	25	0	9	3	0	9	26	2	22	0	2	100

Figure 22: Summary sheet of the intensities and correlations between symptoms

We implemented that table into a csv file and plotted the heatmap from it in Rstudio by using the code right below:

```
sympt <- read.csv("~/Documents/FdV/L2/S2/Bastian/Correlogram/sympt.csv")

sympt.m <- melt(sympt)
sympt.m <- ddply(sympt.m, .(variable), transform, Scale=rescale(value,
0:1))

ggplot(sympt.m, aes(variable, X)) +
 geom_tile(aes(fill=Scale)) +
 scale_fill_gradient(low='white', high='steelblue') + theme_minimal() +
 labs(title = "Heatmap correlating symptoms",
      subtitle = "Made with ongoing symptoms data",
      x = "Symptoms", y = "Symptoms")
```

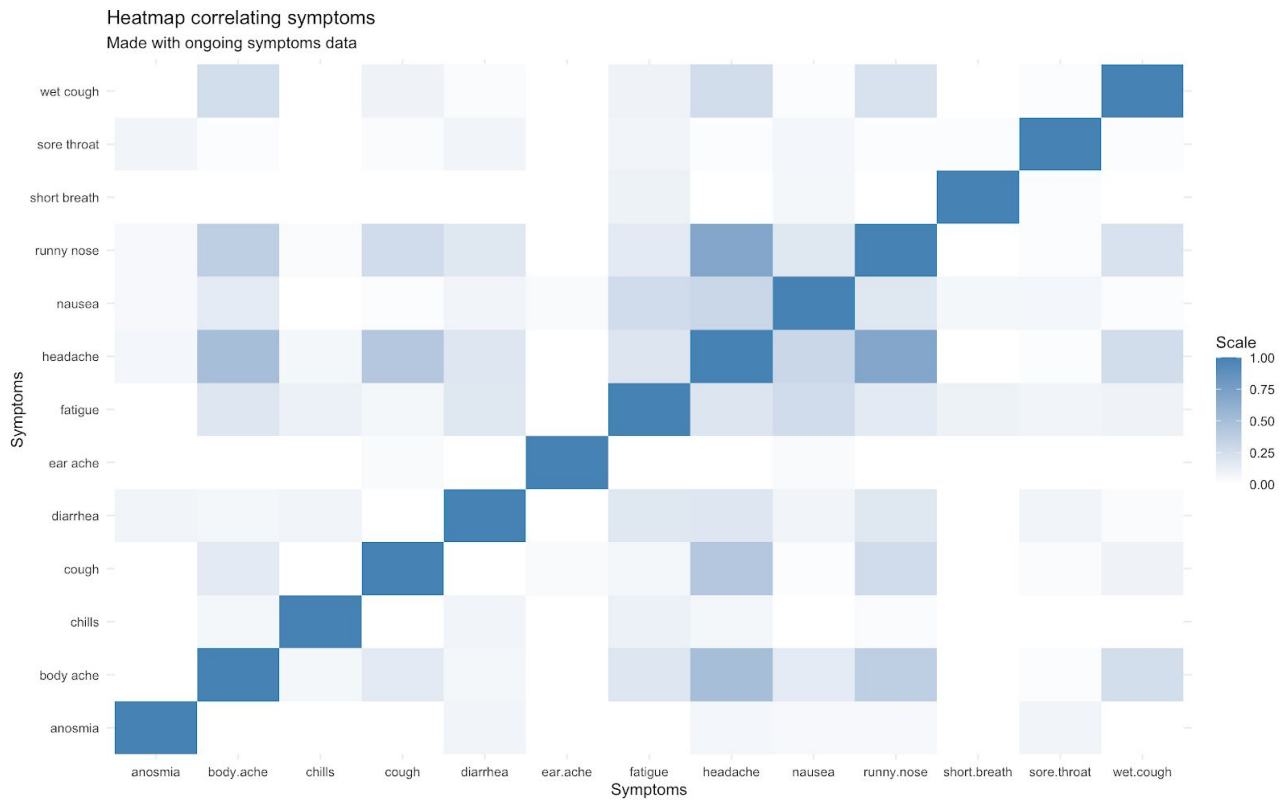


Figure 23: Heatmap made in order to see the correlation between symptoms based on 'ongoing symptoms data' folder

3) **Hclust function:** Bastian suggested us to cluster the rows and the columns in order to see interesting patterns from the heatmap we made in figure 23. In order to do that, we have to use the function hclust in Rstudio where we can use a tutorial available on that website. We then tried to use the function hclust before melting the data. However, we are always having an error message for now.

4) **What's next:** We first have to solve the problem of the hclust function. We have to make more research and to understand the syntax of that function. What's more? We want to find a problem in order to prepare an outline and to answer it. We will prepare some questions that we will expose to Bastian in order to exchange and have his feedback.

Monday 17th of May 2020:

1) **Hclust function from the heatmap in figure 23:** After looking on the Internet to understand the hclust function, we succeeded in implementing the data. We obtained the graph below. In order to fill the sheet, we looked at all the symptomas reports in the ongoing symptome data folder. For instance, if someone for the same date indicated anosmia, body ache and diarrhea, with respective intensities (2, 1 and 2), we wrote down +3 (2+1) in the intersection between anosmia and body ache, +4 (2+2) in anosmia and diarrhea and +3 (1+2) in body ache and diarrhea.

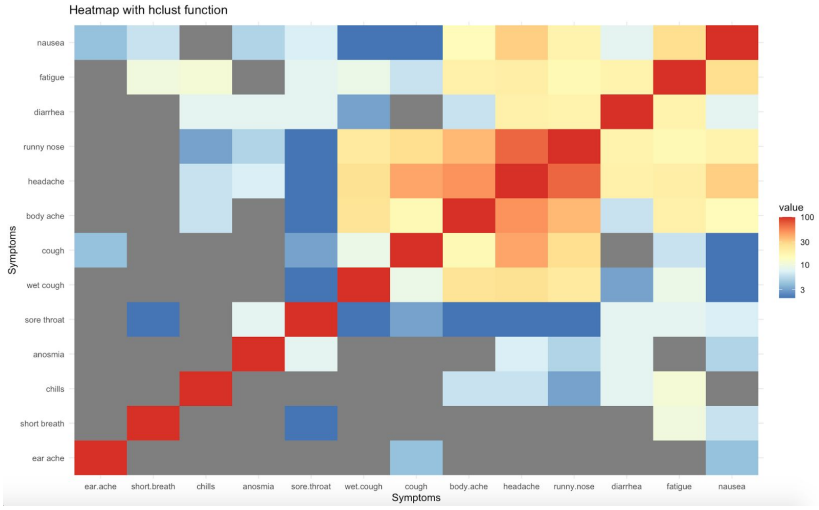


Figure 24: Clustering Heatmap made in order to see the correlation between symptoms based on 'ongoing symptoms data' folder

Bastian suggested that we count how often things co-occur regardless of the values. We created a sheet where we indicated all these observations. In order to fill the sheet, we looked at all the symptomas reports in the ongoing symptome data folder. For instance, if someone for the same date indicated anosmia, body ache and diarrhea, with different intensities, we wrote down +1 in the intersection between anosmia and body ache, +1 in anosmia and diarrhea and +1 in body ache and diarrhea. We then used the hclust function on a new table we made right below.

	anosmia	body ache	chills	cough	diarrhea	ear ache	fatigue	headache	nausea	runny nose	short breath	sore throat	wet cough	stomach ache
anosmia	0	4	2	3	1	0	5	4	1	3	0	5	0	0
body ache	4	0	5	11	4	0	13	25	4	18	0	4	16	0
chills	2	5	0	2	2	0	5	5	0	2	0	1	1	0
cough	3	11	2	0	2	1	6	16	3	11	0	1	7	0
diarrhea	1	4	2	2	0	0	5	6	3	6	0	2	3	0
ear ache	0	0	0	1	0	0	0	1	0	0	0	0	0	0
fatigue	5	13	5	6	5	0	0	13	7	10	1	8	5	0
headache	4	25	5	16	6	1	13	0	10	28	0	6	14	1
nausea	1	4	0	3	3	0	7	10	0	6	0	3	1	0
runny nose	3	18	2	11	6	0	10	28	6	0	0	5	15	0
short breath	0	0	0	0	0	0	1	0	0	0	0	1	0	0
sore throat	5	4	1	1	2	0	8	6	3	5	0	0	3	0
wet cough	0	16	1	8	3	0	5	14	1	15	1	3	0	0
stomach ache	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Figure 25: Summary sheet of the intensities and correlations between symptoms

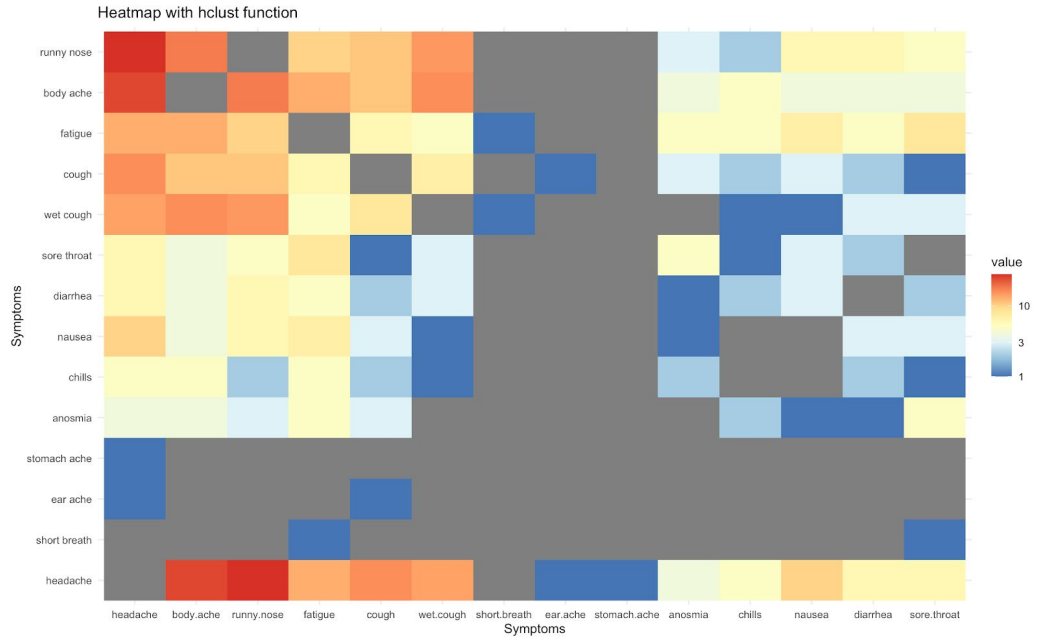


Figure 26: Clustering Heatmap made in order to see the correlation between symptoms based on 'ongoing symptoms data' folder

For the two figures above, we represented the frequencies of symptoms that happened with other symptoms. The more symptoms happen at the same time, the more the intersection square of those

two symptoms will be with hot colors. If the intersection square is red, it means that the two symptoms happened often at the same time. If the intersection square is deep blue, it means that the two symptoms did not happen hardly at the same time. If the intersection square is grey, it means that the two symptoms never happened at the same time.

The code we used in order to make the clusterings plots is:

```
library(ggplot2)
library(reshape2)

df <-
read.csv2("~/Documents/FdV/L2/S2/Bastian/Correlogram/Hclust/symp2.csv",
sep=',')

dat <- df[,2:15] #numerical columns
rownames(dat) <- df[,1]
# clustering
row.order <- hclust(dist(dat))$order
col.order <- hclust(dist(t(dat)))$order
dat_new <- dat[row.order, col.order]
# reshape into dataframe
df_m <- melt(as.matrix(dat_new))
names(df_m)[c(1:2)] <- c("Symptoms", "vsSymptoms")
ggplot(data = df_m,
      aes(x = vsSymptoms, y = Symptoms, fill = value)) +
  geom_raster() +
  xlab("Symptoms") +
  scale_fill_distiller(palette = "RdYlBu", trans = "log10") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1),
      axis.text.y = element_blank()) +
  ggtitle("Heatmap with hclust function") + theme_minimal()
```

2) **Establishment of our problematic:** From the general problematic of the Quantified Flu website which is “Whether the flu or the common cold: Can wearables warn us when we're getting sick?”, we decided to focus on ***“From Oura ring and Fitbit wearables data, can we find a pattern by looking at temperature or heart rate graphs which could be synchronized to a sickness event?”***.

In order to answer our problem, we decided to make an outline:

I- Wearables data, can we rely on?
   a) Information about Oura ring data capture
   b) Information about Fitbit data capture
   c) Comparison between them

II- Usage of the retrospective data folder
   a) Plot a temperature graph for each person per type of wearable

b) Plot a heart rate graph for each person per type of wearable
c) Plot one graph per person summarizing temperature and heart rate through time

III- Sickness event and answer to our problematic
a) Associate the date of the sickness event in each graph per person
b) Compare all the graphs to deduce if a pattern is visible
c) Response to the problematic

IV- Discussion
a) Number of data
b) New dataset from Apple watch and Google Fit wearables, what are the differences?
c) Additional variables to take into consideration?

3) **What's next ?** Start our outline.

Monday 19th of May 2020:

1) **Meeting with Bastian:** During this meeting, we were also with Basile Morane, a master student who is doing his internship. We spoke about what we did, and exchanged in order to improve our work. As nobody has entirely treated the Ongoing symptoms data folder, Bastian told us to work on it too. We then revised our outline.

2) **Outline:**
I- Wearables data, can we rely on?
a) Information about Oura ring data capture
b) Information about Fitbit data capture
c) Comparison between them (min and max)

II- Usage of the retrospective data folder
a) Plot a temperature graph for each person per type of wearable
b) Plot a heart rate graph for each person per type of wearable
c) Plot one graph per person summarizing temperature and heart rate through time

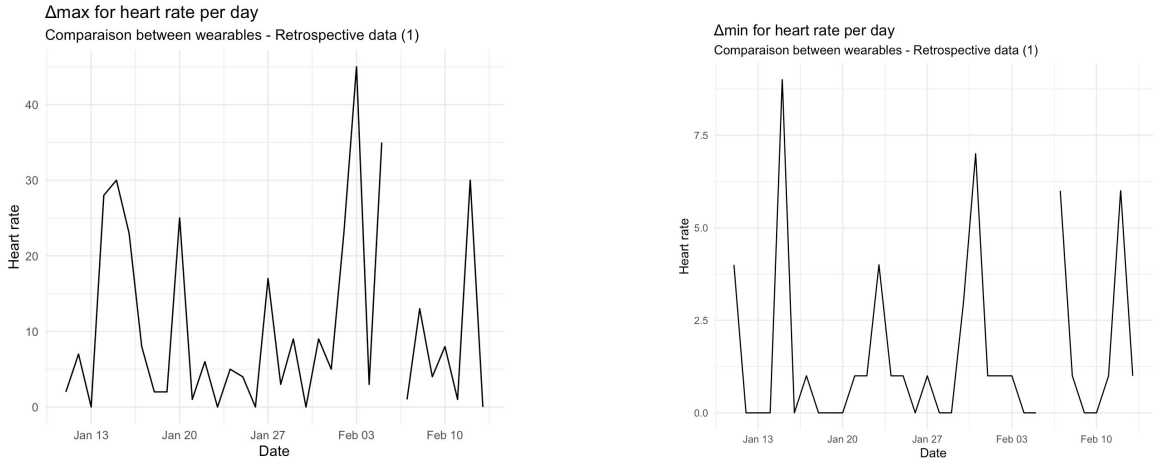
III- Usage of the ongoing symptoms data folder
a) Heatmap per day for each person
b) Heart rate and temperature graphs for each person
c) Combined the two with cowplot

IV- Sickness event and answer to our problematic
a) Associate the date of the sickness event in each graph per person
b) Compare all the graphs to deduce if a pattern is visible
c) Response to the problematic

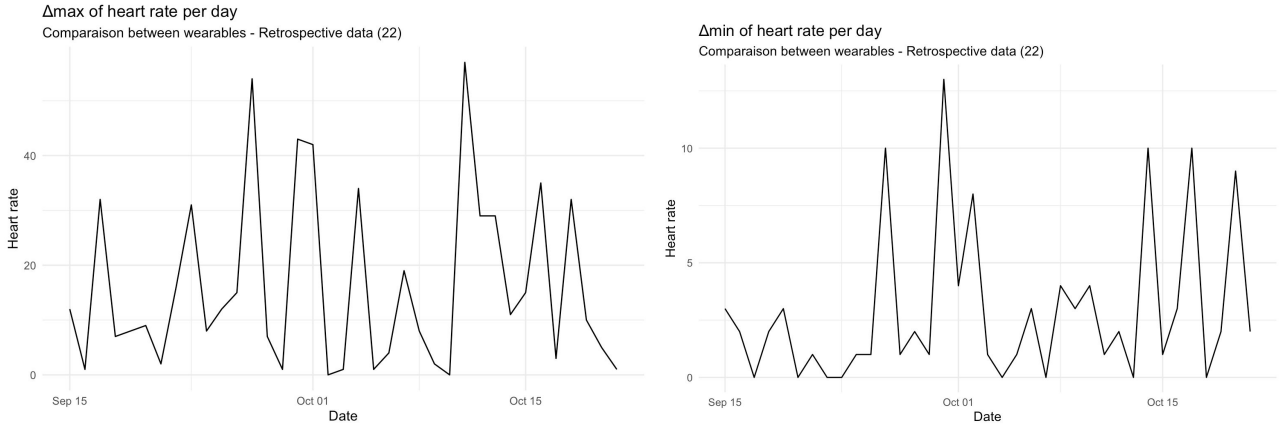
V- Discussion
a) Number of data

b) New dataset from Apple watch and Google Fit wearables, what are the differences?
c) Additional variables to take into consideration?

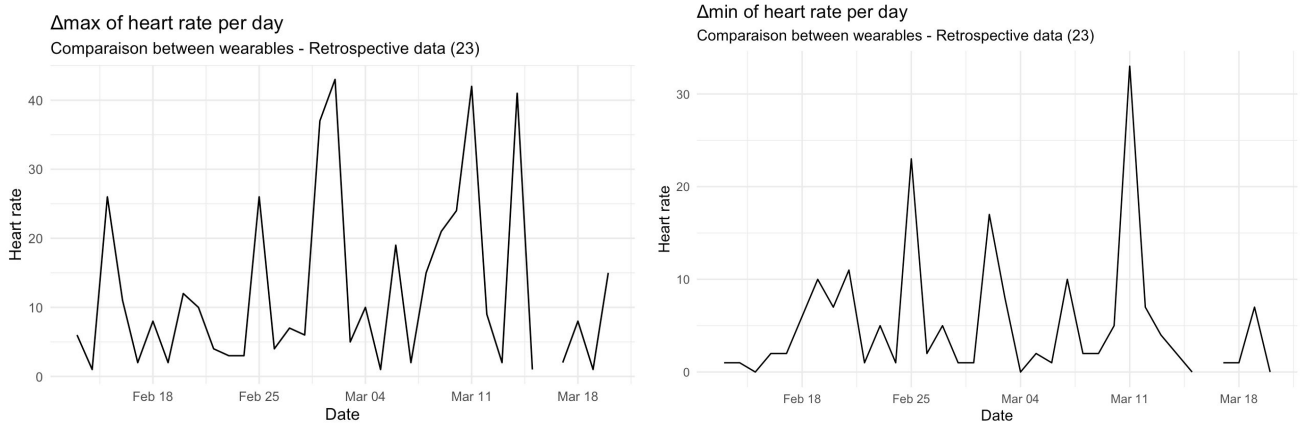
3) **<u>Wearables data, can we rely on?</u>** We plotted all the graphs we could in the Retrospective data folder in order to compare Δmin and Δmax...



<u>Figure 27:</u> Geom_line of Δmax(max(fitbit_intraday) - max(oura_sleep_5min)) per day and Δmin(min(fitbit_intraday) - min(oura_sleep_5min)) per day for the 1.csv retrospective file



<u>Figure 28:</u> Geom_line of Δmax(max(fitbit_intraday) - max(oura_sleep_5min)) per day and Δmin(min(fitbit_intraday) - min(oura_sleep_5min)) per day for the 22.csv retrospective file



<u>Figure 29:</u> Geom_line of Δmax(max(fitbit_intraday) - max(oura_sleep_5min)) per day and Δmin(min(fitbit_intraday) - min(oura_sleep_5min)) per day for the 23.csv retrospective file

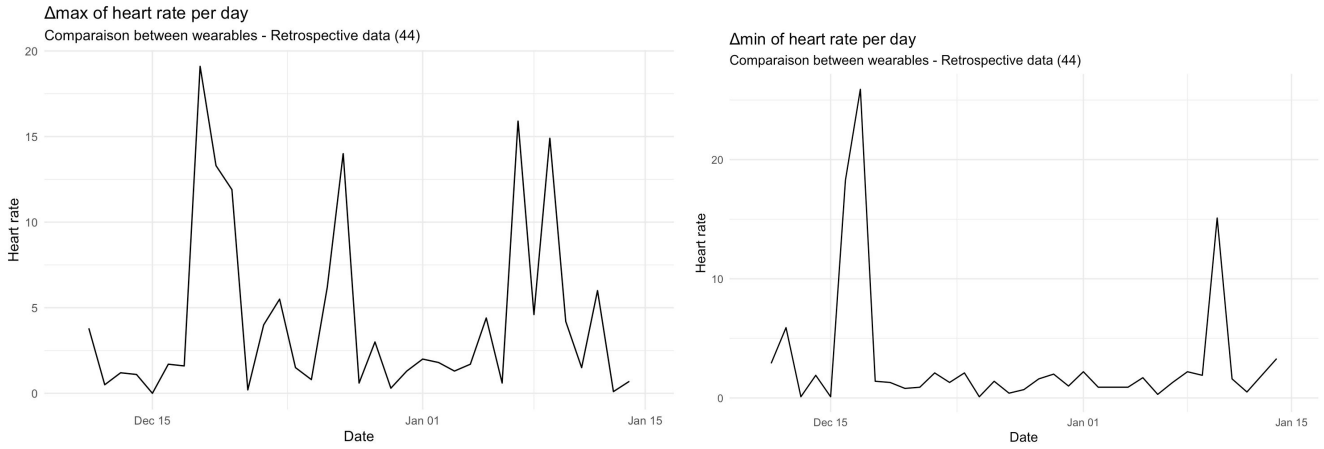


Figure 30: Geom_line of Δmax(max(fitbit_intraday) - max(oura_sleep_5min)) per day and Δmin(min(fitbit_intraday) - min(oura_sleep_5min)) per day for the 44.csv retrospective file

However, for the file 5 and 43, we could not plot those graphs because respectively, only Ouraring data has been captured and only two dates were comparable.

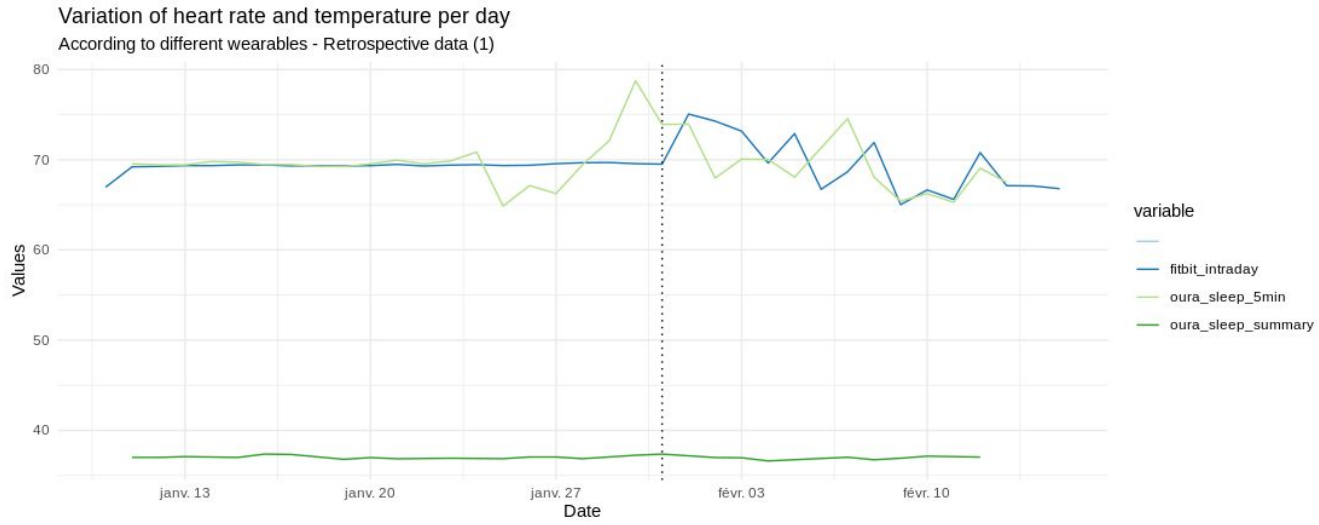
In order to compare these two wearables, we also decided to calculate the average of the percentage of similarity per file (so the average of all the dates) and then to calculate the total average for all the files.

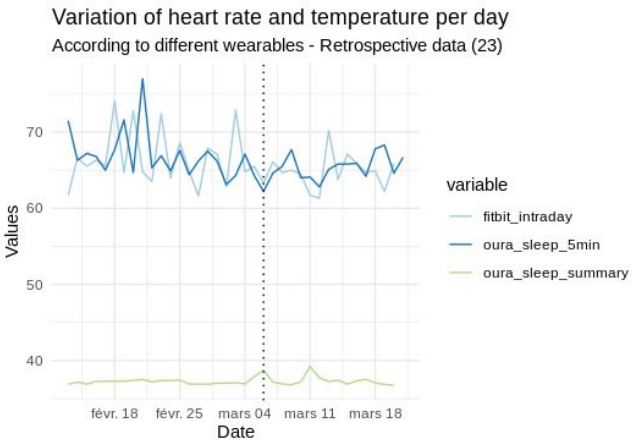
File (.csv)	average of % of sim. min.	average of % of sim. max.
1	98	90
5	NA	NA
22	95	83
23	92	87
43	91	95
44	95	95
TOTAL	**94**	**90**

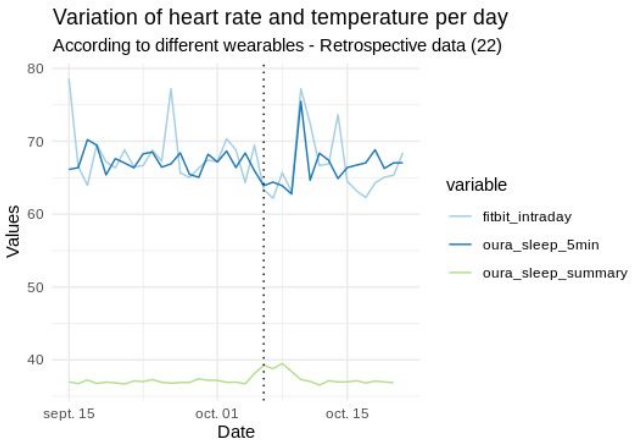
Figure 31: Table of the averaging percentages of similitude for maximum and minimum values, for 1, 5, 22, 23, 43 and 44 csv retrospective files

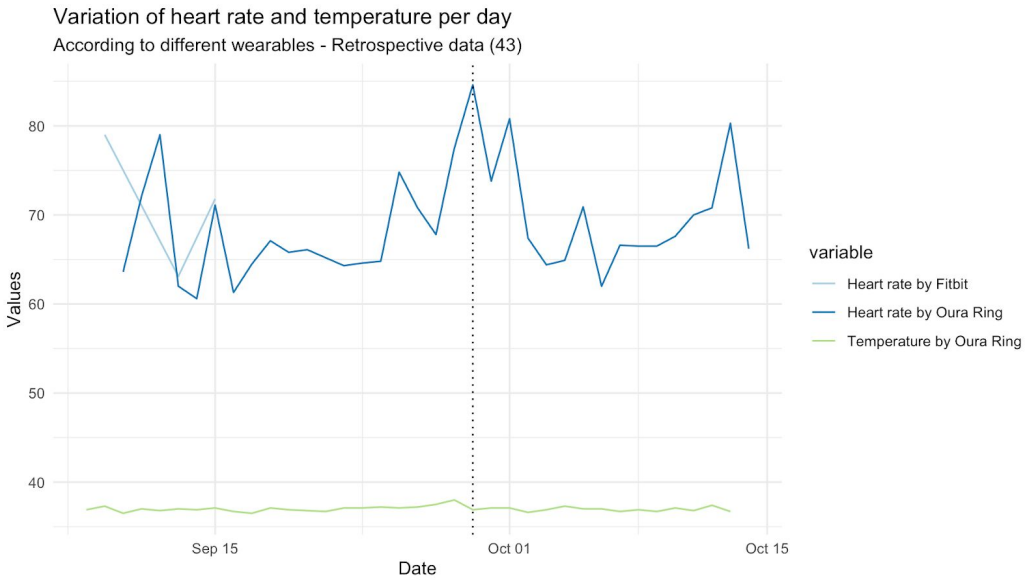
Thanks to that spreadsheet, we can first conclude that Oura Ring and Fitbit wearables have 94% of similarity concerning the lowest heart rate so the resting one and 90% of similarity concerning the highest one. However, we have to take into consideration the fact that we are treating only a few files (5) and that we should have way more data in order to conclude in a proper way.

4) **Temperature, heart rate and sickness events according to wearables per person:** For all the retrospective files, we plotted graphs per person, representing the heart and temperature (when reported) and we associated a black dotted vertical line which is reporting the sickness event. Here are the 6 graphs we made.


Variation of heart rate and temperature per day
According to different wearables - Retrospective data (1)
Values
80
70
60
50
40
janv. 13
janv. 20
janv. 27
févr. 03
févr. 10
Date
variable
fitbit_intraday
oura_sleep_5min
oura_sleep_summary


Variation of heart rate and temperature per day
According to different wearables - Retrospective data (23)
Values
70
60
50
40
févr. 18
févr. 25
mars 04
mars 11
mars 18
Date
variable
fitbit_intraday
oura_sleep_5min
oura_sleep_summary


Variation of heart rate and temperature per day
According to different wearables - Retrospective data (22)
Values
80
70
60
50
40
sept. 15
oct. 01
oct. 15
Date
variable
fitbit_intraday
oura_sleep_5min
oura_sleep_summary


Variation of heart rate and temperature per day
According to different wearables - Retrospective data (43)
Values
80
70
60
50
40
Sep 15
Oct 01
Oct 15
Date
variable
Heart rate by Fitbit
Heart rate by Oura Ring
Temperature by Oura Ring

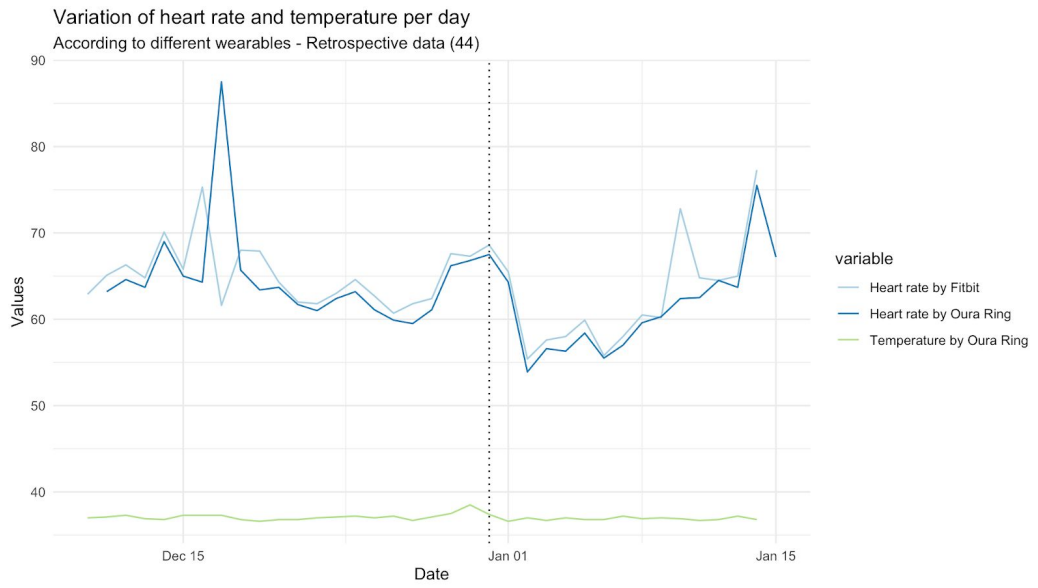


Figure 32: Geom_line of average values per day of Fitbit and Oura Ring wearables, for 5.csv, 22.csv 23.csv, 43.csv and 44.csv retrospective files

On those graphics, the dotted lines represent the date of the sickness events. As we can see, after or at the same time as the sickness events, there is a peak in heart rate and temperature. A peak can be characteristic of a sickness event. Maybe we can predict sickness events just by seeing a peak in temperature and in heart rate.

5) **What's next ? :**

In order to have a better comparison between graphics, and to compare the behaviour of values before the sickness events, we will plot on the same graph, all the values of Fitbit and Oura Rings. The fitbit_intradays, ourasleep_5_min and oura_sleep summary will be separated by categories, in order to have less errors because of the different wearables.

Friday 22nd of May 2020:

1) **Combination of graphs:** In order to visualize a pattern before or after the date of the sickness event, Bastian suggested combining all the graphs in the same one and scale them on the same 'fake' sickness date.
   To do so, we made one graph per type of variable: heart rate captured by Fitbit wearable, heart rate captured by Oura ring wearable and temperature captured by Oura ring wearable.

   We removed the data of the file 5.csv because they were considered as interfacts.

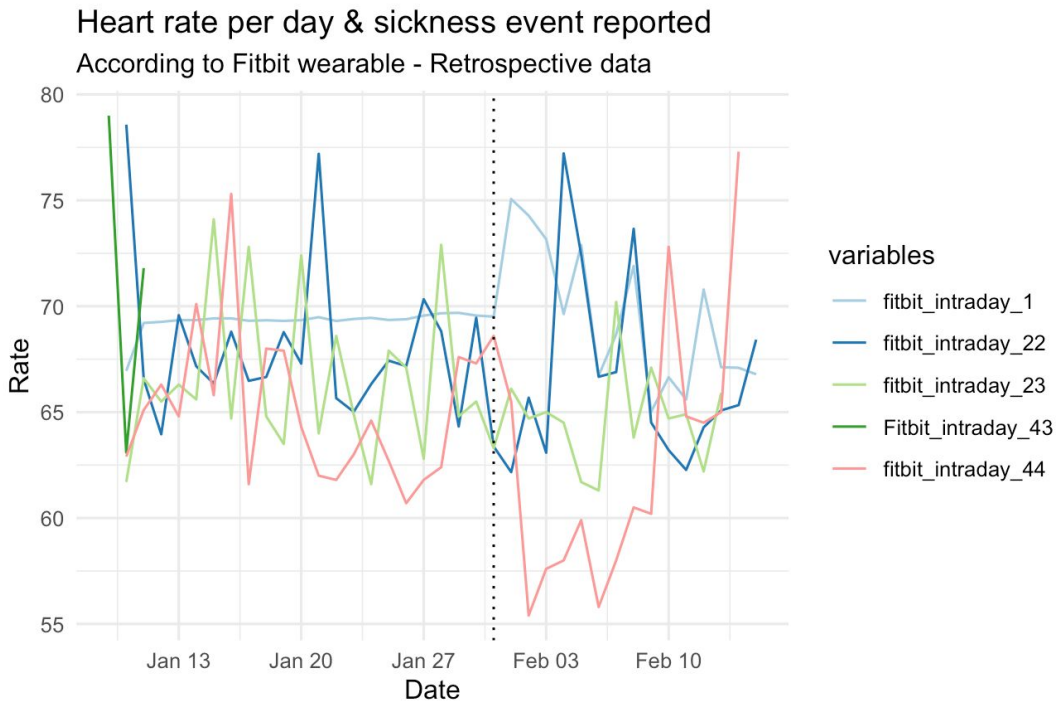


Figure 33: Heart rate captured by Fitbit wearable for the csv files (1, 22, 23, 43, 44) in the retrospective data folder

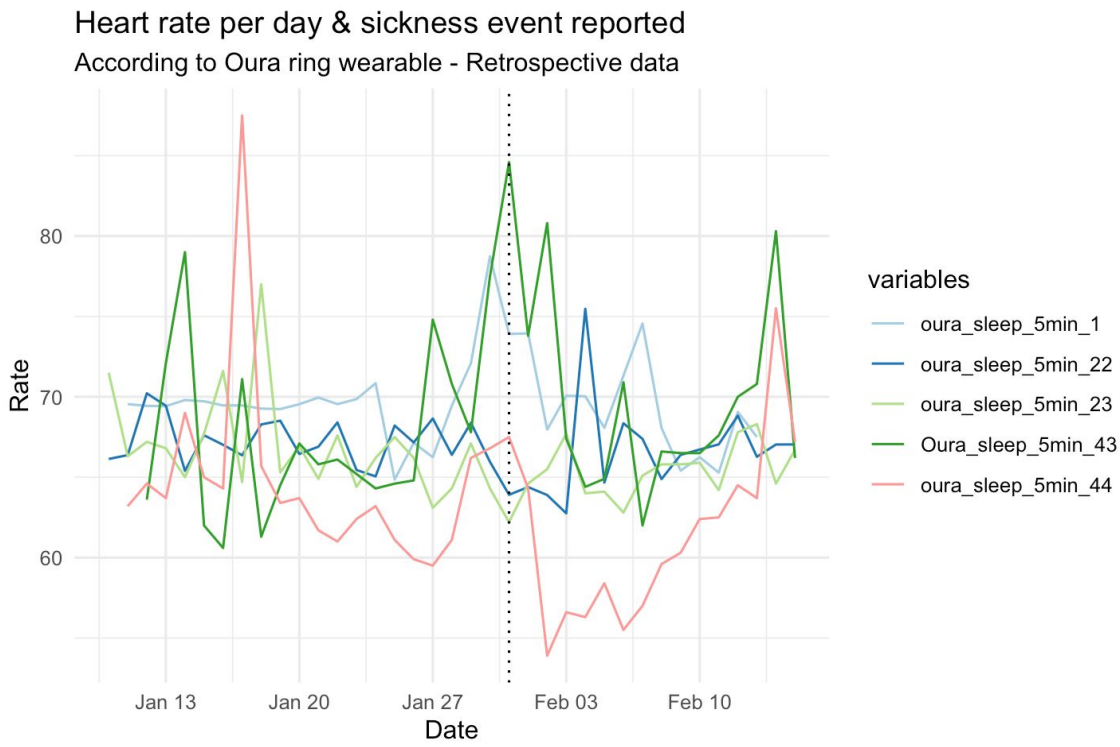


Figure 34: Heart rate captured by Oura ring wearable for the csv files (1, 22, 23, 43, 44) in the retrospective data folder

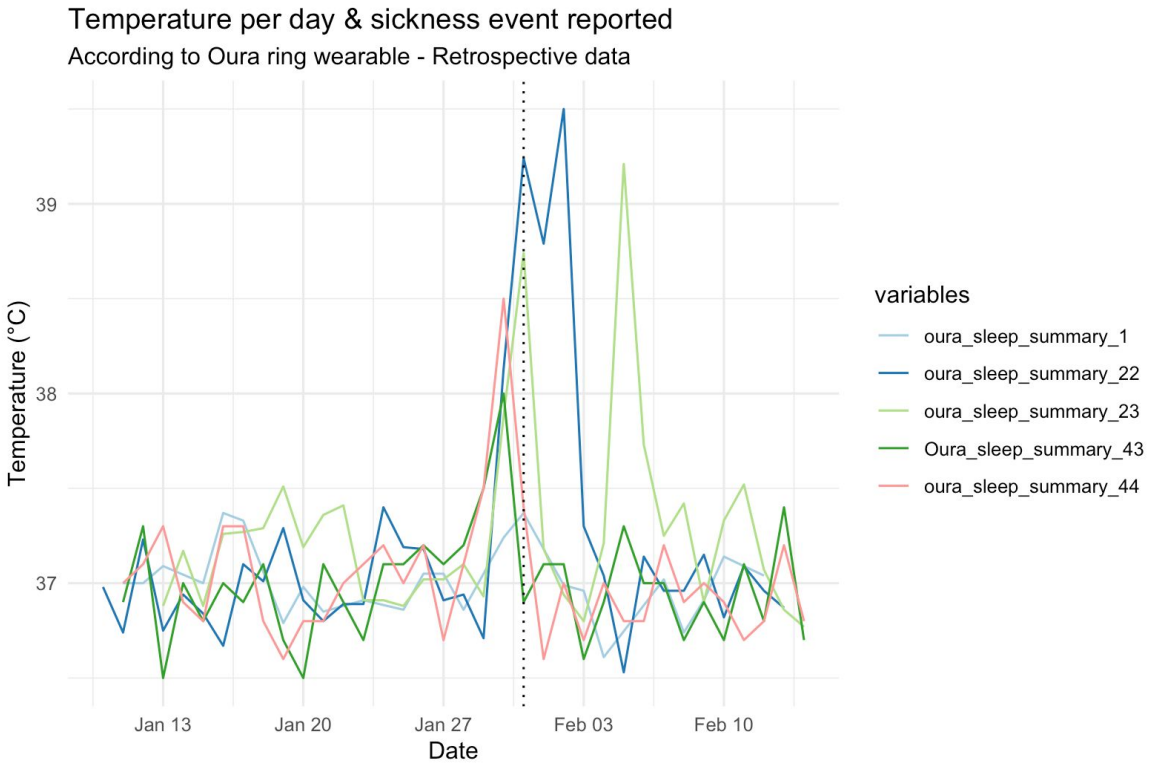


Figure 35: Temperature captured by Oura ring wearable for the csv files (1, 22, 23, 43, 44) in the retrospective data folder

On the figures 33, 34 and 35, we can barely see a pattern around the sickness event date. Indeed, observing something from all the curves associated with each file of the retrospective data folder is complicated.

In order to facilitate the lecture and to determine something in an easier way, Bastian suggested using the geom_smooth function in Rstudio from the same dataset.

2) **Geom_smooth function:** In order to represent the 'average' response, we used the geom_smooth function over all 5 data sets as Bastian suggested.

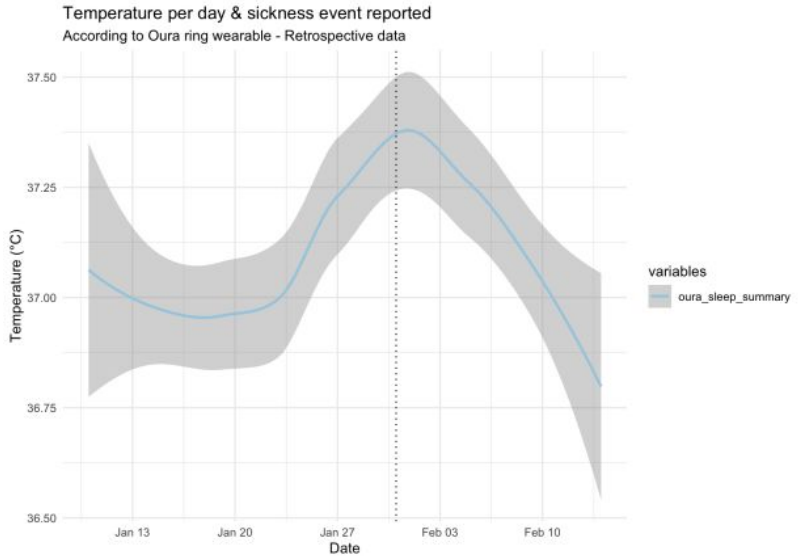


Figure 36: Geom_smooth function of the temperature for every oura_sleep_summary

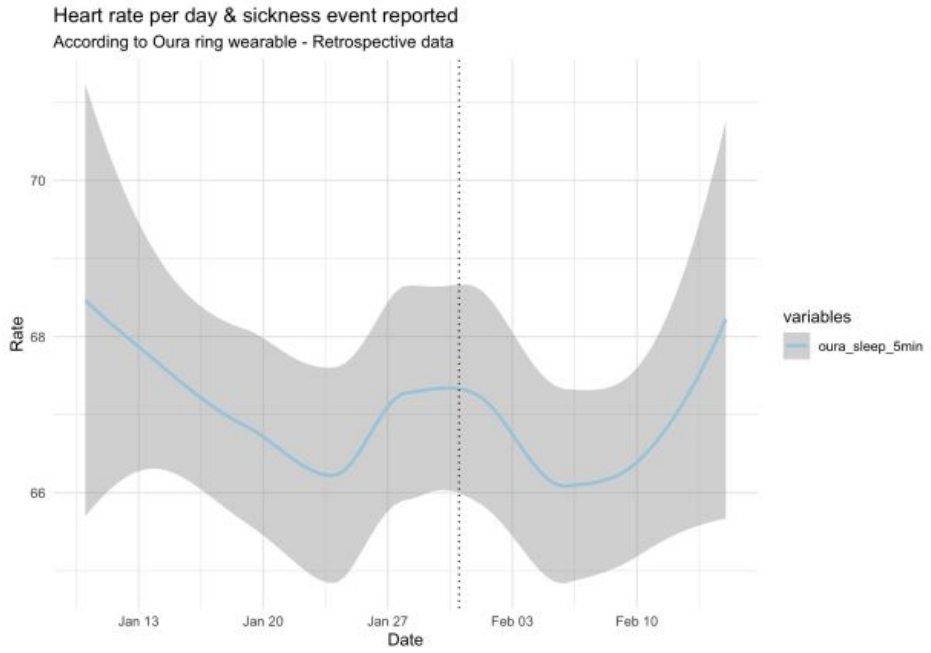


Figure 37: Geom_smooth function of the heart rate for every oura_sleep_5min

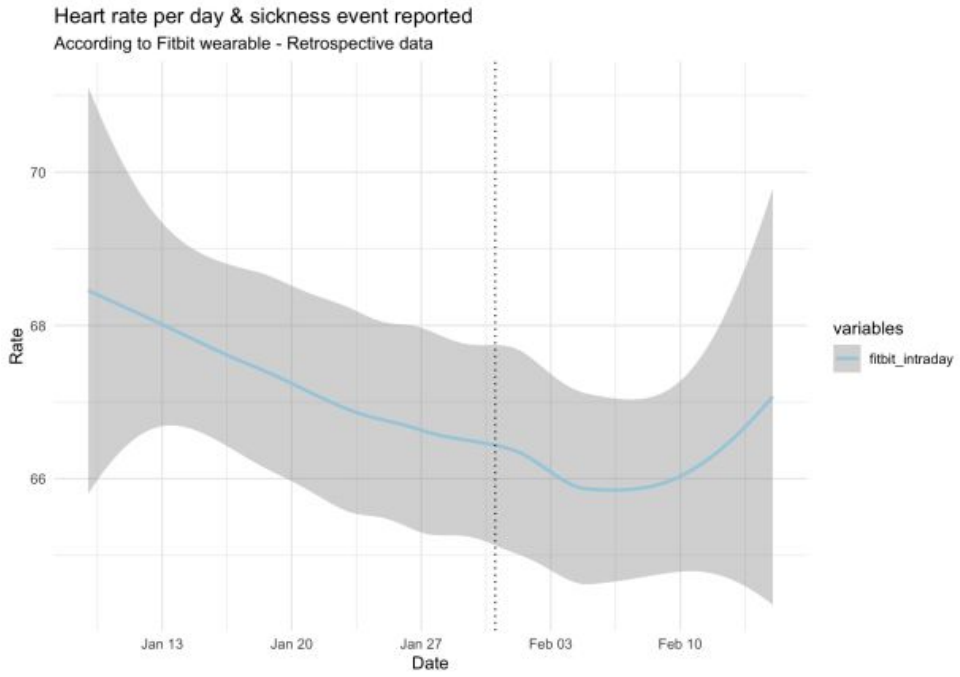


Figure 38: Geom_smooth function of the heart rate for every fitbit_intraday

On figures 36, 37 and 38, we used the geom_smooth function. On the first one, we were kind of impressed about the visible pattern. Indeed, our first conclusion will be that when we are getting sick, our temperature is increasing and the peak of this increase is qualified with the sickness event. However, for the figures 37 and 38, it was difficult to conclude something in a proper way. Indeed, the observations are kind of opposed. We can see that with the Oura Ring wearable, we could

conclude that the heart rate is also increasing around the sickness event; however for the same variable with another wearable (FitBit), we cannot see a logical and visible pattern.

To explain these differences, we thought about the number of data which is not really considerable. Also, we conclude before, that those two wearables have 94% of similarity according to the data captured and treated. This can also explain the differences we can observe.

3) **What's next ?**
As we're going to have a presentation for Bastian and his team next tuesday, we will have to prepare presentation slides, a speech text and also organize what we are going to tackle and present.

Monday 25th of May 2020:

1) **Preparation of the presentation slides for the meeting of Tuesday 26th of May 2020:** Bastian invited us to participate in an 'Open Human Community' meeting on Tuesday 26th of May 2020 at 7pm. Around 10 researchers should be participating in order to speak about the Quantified Flu website and what we did. The goal will be also to have some feedback from them. To do so, we prepared a presentation that you can find here. The most complex point was to choose only the crucial information and analysis we made. Bastian helped us improve our slides and particularly the second slides 'Data repartition' where we were not precise enough.

2) **Preparation of the script:** Once the presentation was quite finished, we prepared the oral script that you can find here. We had to be concise enough without forgetting important things. Bastian gave us feedback by putting comments on the doc. Most of the time, we were using some Rstudio terms such as 'hclust function' or 'geom_smooth function' which were not really cleared. We then defined all these terms and we splitted the text so that we will speak the same duration. -

Tuesday 26th of May 2020:

1) **Preparation of the presentation slides for the final meeting on Friday 29th of May 2020:** Because our presentation slides were finished for tonight, we decided to begin those for next Friday as we planned a rendez-vous at 1:30pm with Vincent Dahirel. We could easily retake what we did for the tonight presentation, however, some information was missing that we had to add such as a presentation of the Quantified Flu website, how did we download the datasets, how do the CSV files look like but also the issue we encountered.

2) **Rehearsals and adaptation of the script for the meeting of tonight:** After some rehearsals, we changed some sentences which were not clear.

3) **Meeting:** The meeting was on Zoom platform. The attendees, from the Open Humans Community, were: Mad Ball, Bastian Greshake Tzovaras, Christopher Wansing, Kate Wac, Gary Wolf, Richard Sprague, Danny Lämmerhirt, Justin Adams, Dave Blackwell, Abhik Chowdhury and us.

We are very thankful to have participated in that meeting. It was really interessant to have some feedback from professionals in that field. During this meeting, we spoke about the Quantified Flu website updates. First about the data visualizations that we made and we also shared our daily report so that they could see our R codes. Kate wrote some comments on our presentation so that we could improve some points.
Then, Bastian presented the work of Basile Moran, a master student currently in internship who made symptom-report heatmaps.

After these presentations, it was discussion time about how the data is captured and the influence we can have on what is captured.
First, for the heart rate: how much you sleep, the activities you do, what you eat, what you drink (alcohol, calorie drinks, ...).
Second, for how the data is captured: based on the graphs we made, comparing Fitbit and Oura ring, we observed a difference and maybe it's because of the software behind or algorithms which are analyzing Oura Ring wearable. Melvin answered that there is a huge similarity between the tow wearables so it may not be the software causing the variation.
Third thing was the criticism of the data analyzed all together because a lot of data is from one person and so it's hard to know how idiosyncratic or universal it is. Bastian answered that the fitbit might slide around, not as tight, while the Oura may be more consistently worn.
Fourth was about the efficacy of the wearables depending on the locations of the body. Kate answered that the gold standard is the chest and that she doesn't know if there's been study on the Oura ring itself. Bastian added that he recalled someone wearing two Oura rings & seeing discrepancies between those, so fingers may not be that consistent too. Then, Mad spoke about the consistency of how people can use their own device; Bastian answered that it was quite easy to try with a Fitbit by putting different spacing on the band on different days. It was also said that people can have a picture of the reliability of their own device by watching their own data through time.
Fifth and final point was about collecting info about supplements (for instance Zinc, vitamines, ...). That's an ongoing idea that should be easier to collect thanks to Apple Health watches which are keeping track of nutrition.

After that long discussion, the current changes & future ones have been discussed: the switching of symptom reports & retrospective events, the modification of the symptom reports, the addition of past symptom reports that were missed.
Finally, Christopher made a short experiment by showing us a Video Motion Magnification which is Making Pulse visible in order to find other ways to analyse pulse.

We really enjoyed contributing to a current subject and we felt inside this research. It was a pleasure to exchange and have feedback from the community.

Wednesday 27th of May 2020:

1) **Responses and improvements of our yesterday presentation:**
   We answered some propositions made by Katarzina about the precise number of people who indicated some symptoms such as anosmia, cough, ... We figured out that some information we wrote were not really exhaustives.

2) **Improvements on our final presentation:** We then improved our final presentation according to the comments Katarzina wrote down.

Bussod Ilona and Fribourg Melvin

Thursday 28th of May 2020:

1) **Preparation of our presentation:** We prepared our speech for our last presentation and established the final roadmap. We also finished the daily report.