BACHELOR

Data Analytics in the Emergency Department

van Rijckevorsel, Lars P.A.

Award date:
2023

Data Analytics in the Emergency Department

Bachelor End Project

L.P.A. van Rijckevorsel
1573284

Dr. Renata Medeiros de Carvalho

Final thesis

Eindhoven, June 2023

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Department of Applied Physics
Optionally, specify a research group

Abstract

This paper addresses the problem of patients staying longer than four hours in the emergency room of a hospital. This so called 'overstaying' leads to long waiting times and more staff needed in hospitals. Data was provided, which contains variables about the ER admissions, waiting times and patients. By exploring these data, seemingly interesting facts and correlations were found. To make the claims for the correlations foolproof, plots should be created and tests should be conducted, in order to do this, the data was preprocessed first. When a clean data frame with all necessary data was created, a lot of plots were created to make clear correlations stand out. Where some plots did disappoint others showed clear correlations between a couple of variables. For those correlations a hypothesis test was drafted and was tested by the pooled two sample z-test for proportions. The test returned a p-value which showed that the null hypothesis should either be accepted or rejected. With these results a follow-up plan for the ER could be sketched and conclusions could be drawn.

Contents

1 Introduction

In the news, there is a lot to do about the high performance pressure and lack of personnel in a lot of domains. Also in the health sector there are lots of problems regarding employees. This leads to longer queues and waiting times for patients that need to use the health system. According to a survey conducted by 'Zorgvisie', hospitals put in hard work in order to ensure a smooth and fast flow of patient traffic, but, unfortunately, the waiting times in the emergency room (ER) keep growing [2].

For this thesis, we will analyse a lot of data on the patient that enters the ER and his or her injury. This data was provided by the ER of a hospital in The Netherlands. On the basis of this data, this thesis will attempt to find ways to reduce the number of patients that will overstay, i.e. patients that are staying at the ER for more than 4 hours. How this goal will be achieved will be specified in the research question.

1.1 Research question

While performing the data exploration (4), there was a clear correlation between the age and time of stay, as well as the complaint the patient had and their time of stay. This leads to the following research question:

"How can the age and complaints of patients be used to identify overstaying patients? And how can this be applied to reduce the number of overstaying patients in the ER?"

There is going to be a thorough investigation on how age and complaint relate to the time of stay. From the results, an advise report for the ER will be formed that they can use in order to help more patients with less employees.

2 Literature Review

This is not the first project related to the waiting times in the emergency department, this topic is very hot for years now. Especially during and after Covid 19, there are now significant staff shortages and healthcare is very important.

2.1 Purnell

An example of a paper that deals with the increasing waiting times in hospitals is the one written by Larry Purnell in 1995 [5]. Even though it is written in 1995 and thus might be outdated, it is really useful and sketches the problem and its causes. He looked at a survey from 1990 which identified the top three most important causes. The survey, existing of 44 questions that were later reduced to 21 items, was sent to 500 general hospitals through the USA. These were the most important causes:

1. Nursing shortage
2. Quality of care
3. Emergency department overcrowding

They also found that when a hospitals closes, neighboring hospital get more crowded. This seems of course very logical. Solutions for the problem are successful systems initiated by simple protocols. For example, a protocol where the triage nurse sees every patient that comes in within the first 10 minutes of arrival and then provides treatment. There was also a hospital that instituted a fast-track system for non-urgent cases.

2.2 Laskowski

There are also some models emergency departments use in order to reduce the waiting times. Marek Laskowski writes about models that simulate the process and factors that influence the waiting times [4]. This paper is way more extensive that the previous paper that was touched upon. With tons of illustrations Laskowski sketches the process that a patient undergoes from walking in until exiting the ER.

In the paper he presents 2 models that investigates the patient's waiting times. One agent based modeling framework. This one simulates the emergency departments and technologies that are well suited to enhance simulation with statistical data that is collected in real time. The second model is a traditional queuing model with which the queue in the emergency department is simulated. These models can be used in future work and also show that by augmenting policies regarding machine learning the ERs can work more efficient. In the conclusion is stated that however the paper was meant to investigate the ER, these models can be used to study multiple hospital situations.

3 Methodology

The methods or approaches that are taken in order to answer the research question will be discussed now. First the data will be explored. The data file 'Basis en Triage' will be initialised as data frame. The treatment times and time at the ER get added to the columns by doing calculations with the time.

Since the total time at the ER for every patient is known now, the length of the stay under several conditions can be evaluated, such as for age, complaint, arrival hour of the day. During the data exploration phase the correlation between the time spent at the ER and other variables was tested so it became clear which factors have a significant impact on the time spent at the ER. These factors were chosen to investigate and to test whether their influence on the time of stay is really significant enough to speak about a causation. This project will be focused on the age of the patient and their complaint, since those variables say something about the severity of the injury of the patient and the severity is unconsciously in most cases connected to the length of the stay.

In order to investigate more regarding complaints, complaints of the same origin are merged together. For example, there is 'POB', 'pob', 'borst' and 'pijn op borst', they all mean the same, namely: chest pain. These complaints should all be merged into one category so it will be easier to see correlations between complaints and the time of stay. Categories will be chosen by using common sense and the classification by the WHO [1].

When the complaints are categorized, the analysing of the data and testing for correlations can start. With this the correlation between complaint, age and time spent at the ER in hours is meant. 95% confidence bounds will be used in order to improve the certainty of the correlation. If there seems to be a correlation the mean time spent at the ER is tested with a two-sampled Z-score test to check whether the difference is significant. This is all done in order to find the group that is the bottleneck for the ERs and are the cause of other patients overstaying while they could have been helped in just a few minutes.

After the problematic group of patients is found, the correlations between the age, complaints and 'dismissal to' will also be evaluated since this can form a solid basis for sending the patients to the hospital more quickly. For example, if patients who are around 70 and have heart problems were sent to the hospital 28 of the 30 times, next time a patient that checks these conditions can be sent to the hospital immediately and this will save much valuable time.

There are a few charts that are mandatory to plot in order to find whether there is a correlation between the variables. The following charts will be plotted:

- Line chart of average time at ER per age group for every complaint.
- Pie charts of frequency 'dismissal to' under different complaints

After plotting charts that illustrate the correlations, they will also be tested and proven with a confidence interval of 95%. There is a hypothesis test created, here is tested whether some patient groups are more likely to be dismissed to the hospital that other patient groups. A significant difference is needed to make the claim foolproof. A two sample z-test for proportions is used to proof this [3]. This proofs whether the difference between the proportions in which patients are dismissed to the hospital between different patient groups is significant.

The end goal of this project is, like mentioned before, reducing the waiting time of patients at the ER. However to take it even further, from the results of this project, a plan for the ER will be written that will explain how they can work more efficiently and keep the waiting times for patients lower with the same amount of staff.

4 Data exploration

Before the data exploration phase the total time spent at the ER is calculated by subtracting the arrival time from the departure time. This creates a value that represents the length of the stay of the patient. The main goal is the reduce the patients that overstay, so stay at the ER for longer than 4 hours. In figure 1 the histogram of waiting time in hours for all patients is plotted. With this variable the data exploration could start.

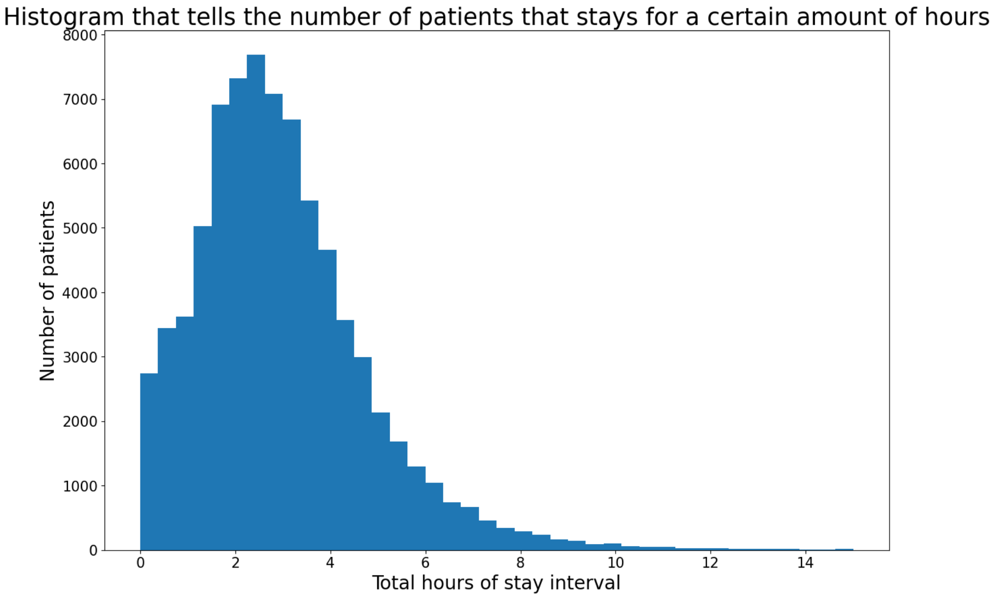

Figure 1: The histogram of the total hours of the stay of the patients

In this graph the histogram of the number of patients that stays a certain amount of time is given. Here can be seen that most patients stay until around 3 hours. This is quite long especially when you realise this is for the emergency department, where people go when they need quick help.

In the next step, different variables were plotted against the length of the stay. Some examples are: age, specialism, arrival hour, arrival date, temperature of the patient, complaint and where they were dismissed to. From these tests only a few variables gave an explainable correlation and we decided to focus on these variables:

- **Age**, this can be seen in figure 2. A slowly increasing line can be seen, this is no evidence that there is a causation but it seems obvious that someone from an older age is more vulnerable for injuries and should be treated more carefully which will take more time. This will be tested in this report.
- **Complaint**, some complaints are of itself connected to more severe injury and a longer time of stay. In figure 3 it becomes clear that for different injuries the median time at the ER differs quite a lot.

In this report, the correlation between those two variables and the length of stay will be analysed. This will lead to results that tell how long a patient of a certain age with a certain injury will stay at the ER. Additionally, there can be looked at the 'dismissal to' variable, if a person of a certain age

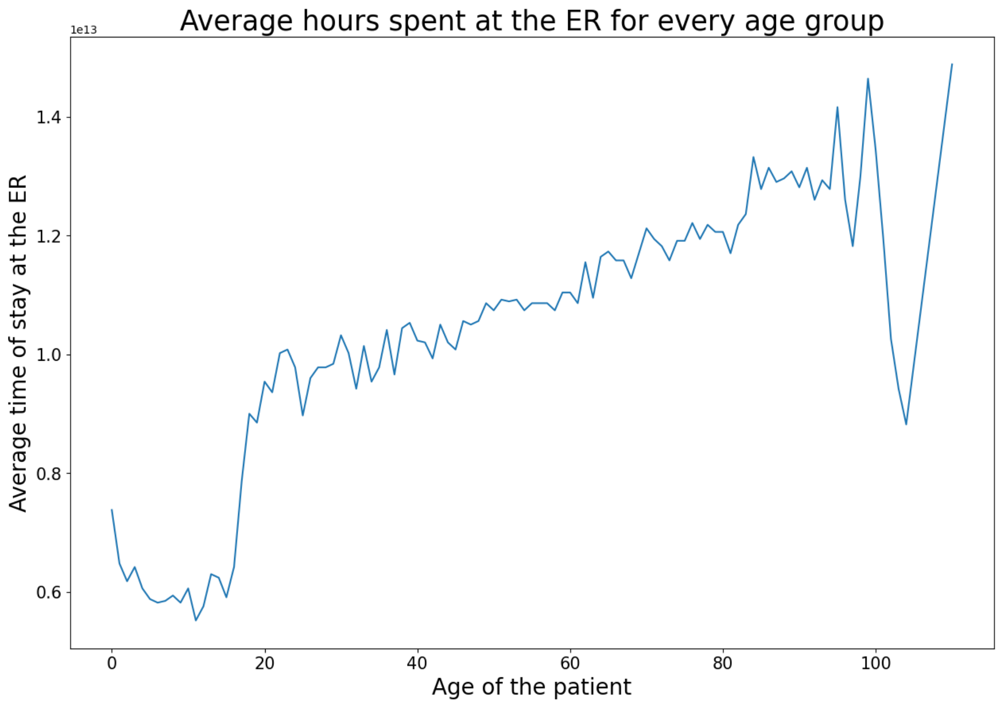

Figure 2: The line chart of the median time of stay at the ER for patients of a certain age

and with a certain injury is always sent to the hospital, this can be done immediately and this will shorten the time the patient will stay at the ER.

4.1 Useful findings

After exploring the data, some interesting findings emerged. Age and the complaint had clear differences in the stay at the ER. They have a large correlation with the total time spent at the ER. After this finding, the research question was formulated, since age and complaint clearly significantly influence the time the patient spends at the ER. Now there can be investigated whether the age and complaint influence this time so much that it can be considered as a causation and thus a solution to reduce the time at the ER can be drafted.

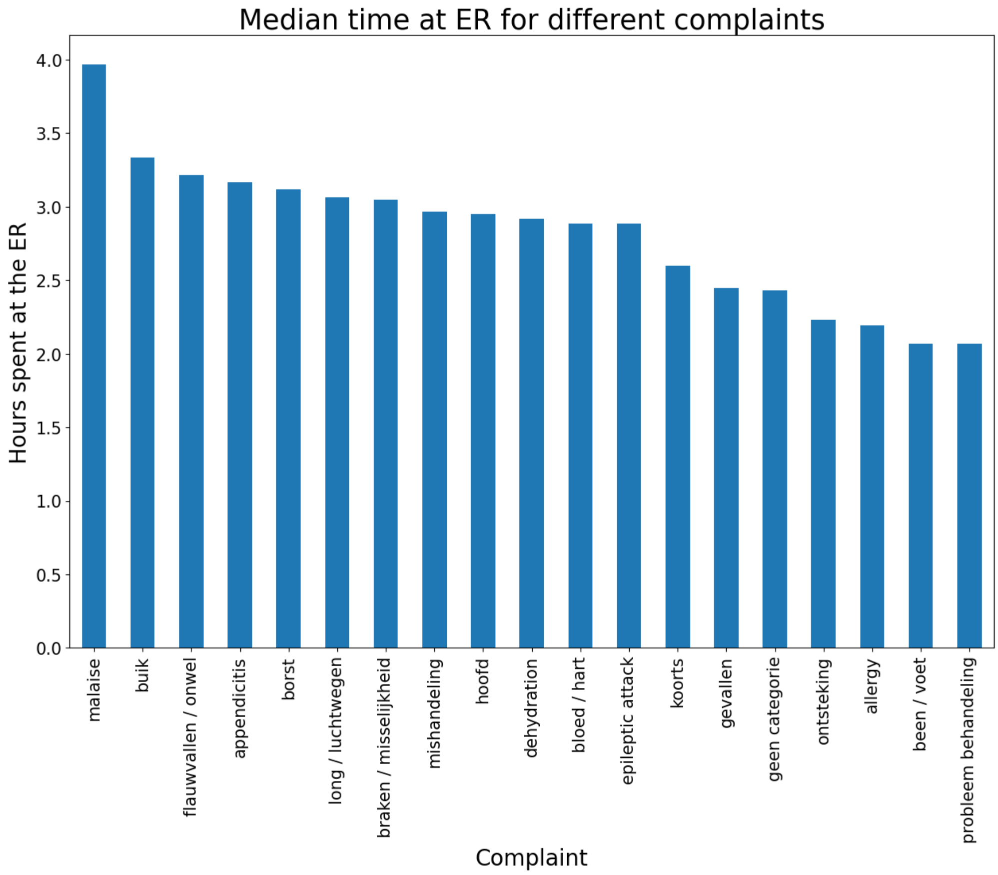

Figure 3: The barchart with the different median time at the ER for patients with different complaints

5 Data preprocessing

Some data needed some preprocessing in order to analyse it and work with it. An example is the merging of the complaints.

5.1 Time at the ER

The dataset that was provided contained some dates and times. These were the following:

- ArrivalDate, the date a patient arrives at the ER.
- ArrivalTime, the time a patient arrives at the ER.
- Start Treatment Date, the date the treatment of the patient has started.
- Start TreatmentTime, the time the treatment of the patient has started.
- TriageDate, the date the urgence of the patients injury is established.
- TriageTime, the time the urgence of the patients injury is established.
- DepartDate, the date the patient departs from the ER. Either home or to the hospital.
- DepartTime, the time the patient departs from the ER. Either home or to the hospital.

This is useful but not structured data. For this project it was decided to clean this data and add the differences as columns, so they are easy to work with when they are needed in further investigation.

First the dates and times are combined into one date time object for every start time. Subsequently, their difference is calculated. This leads to the following columns being added:

- TimeToTreatment, this column contains the time from the arrival at the ER to the beginning of the treatment of the patient.
- TimeToTriage, this column contains the time from arrival at the ER to the triage
- TreatmentToTriage, this column contains the time between the start of the treatment and the triage
- TreatmentTime, this column contains the time between the start of the treatment and the departure of the patient.
- TimeAtTheER, this column contains the total time the patient spent at the ER. From arrival to depart

These time differences are important for the following experiments and analysis. The TimeAtTheER column was used for all the experimentation. In order to calculate with this, the variable was transformed into entire hours and added to the column called 'TotalHours'. This variable was used for all experiments below.

5.2 Merging complaints

In the data frame 'SEH basis en triage' there is a column called 'complaints'. This is the column we will be looking at for this part of the report. In this column there are for the same complaint different names. This can be considered as a problem since it is desired to keep the same complaints under the same term. So this complaints should be merged. An example of how this is done is with the complaints 'POB', 'pob', 'pijn op borst' and 'borst'. All those terms mean chest pain. With a quite extensive code which can be found in the appendix 9.1, the complaints with the same basis are merged into one complaint. The complaints that were used are:

- chest
- belly
- fever
- stuffy
- cardiovascular system
- head trauma
- fainting
- malaise
- appendicitis
- epileptic attack
- pneumonia
- allergic reaction
- assault
- dehydration
- leg
- vomiting
- problem after treatment
- ignition
- fall

Sometimes a complaint contains multiple keywords that link them to categories. In this case the complaint is connected to both categories. Additionally, sometimes the word 'geen' comes in front of a complaint. This means 'no', so in the code it is implemented that when this word comes in front of a complaint, it is not categorized as this complaint even though the keyword is there.

Sometimes there is a question mark (?) in the complaint this means that the practitioner is not sure whether the patient has the injury or not. This is not filtered out, for such a case it is assumed the patient has this particular injury.

5.3 Filtering outliers

After plotting the results, which can be read in the next chapter, it became clear that the data was dealing with some serious outliers. Even when using the rolling mean for the lines, the maximums of a few plots turned out to be unrealistically high. Some patients seem to have spent a really long time at the ER. For example there is a record which states that the time of stay was over 3000 hours. As many may conclude, this is absurd and seemingly impossible. For this reason there was decided to take a threshold of 20 hours since this seems already too long.

This results in filtering out 35 records which is not a lot when looking at the total of 77019 records. This leaves the dataset with a total of 76984 records. So the filtering did not lead to gigantic losses.

6 Results

In the chapter of results the focus on total hours spent at the ER for every age and complaint is separated from the dismissal to for every age and complaint. This is done because they are two different findings even though one might give us a reason to investigate a particular complaint in the other.

6.1 Time spent at the ER

Time spent at the emergency room in total is a really important factor since longer than 4 hours implies an overstay of the patient and this is something that should be avoided. The relation between a complaint and the time of the stay was already shown in figure 3 and the one for age in figure 2. In the following a more thorough study will be executed on these variables and their influence on the time of stay in the ER.

First, the plan was to improve on figure 3. This bar chart shows what the median time at the ER is for every different complaint. However, it does not tell the entire story, by using this kind of plots, a lot of questions still remain unanswered. For example, are there a lot of outliers? Are the differences narrowly or broadly distributed? How far reaches the 95% confidence intervals? A combination of boxplots can solve these questions. That is why previous named bar chart was translated into boxplots, they can be seen in figure 4.

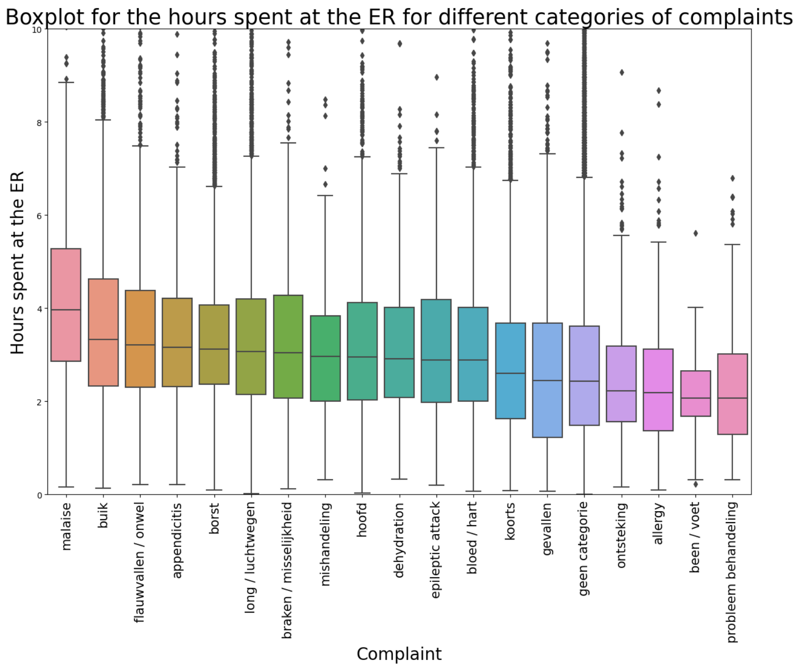

Figure 4: Boxplot of the hours spent at the ER for different complaint groups

From this boxplot, one is able too see a lot more. All in all are the confidence bounds respectively wide. It is clearly visible that almost every lower bound reaches towards zero, this means that every complaint can be treated in a relatively short time. An exception is for 'pneumonie' which is pneumonia. This means that in all cases pneumonia takes at least half an hour. Apart from this, it seems that there is a slow decrease when looking at different complaints. Only judging on this boxplot, the decision would be taken to investigate the top three longest taking results first. However, there is still a lot to investigate before the creation of the plan for the emergency department.

Secondly, there was an idea to plot the mean of the waiting time against the age for different complaints, including the 95% confidence bands. This could demonstrate that the time at the ER for different age groups under a certain category of complaint differ with such extent that there is a correlation between age and waiting time under that complaint. This plot can be found in figure 5. Here, the independent variable is the age and the dependent variable is the hours the patient stays at the ER, every graph gives the median for every different complaint.

As can be seen in the plots, there is no clear ascending line. In figure 6, a sketch given on how the plot should look if there was a clear difference in the younger and older ages under a certain complaint. If, in a certain point the lower quantile line gets above the upper quantile line, it is clear that the age at that point has an influence on the length of the stay.

However, as stated in the previous paragraph and as can be in figure 5, this is not the case in any of the graphs. This concludes that age alone has no significant effect on the length of the stay when looking at complaints alone. So the next step is to try and combine these plots within one plot. This is done in figure 7.

As can be seen in the figure this becomes very messy because of the number of lines. When the confidence lines get removed, as we get in figure 8.

Apart from the spike in the complaint 'gevallen', which translates to fell, around the age of 50, there is no clear difference between the complaints. This means that there is no correlation between age and time spent at the ER within the same complaints. Additionally, the time of stay for different ages between the different categories of complaints is also not significantly different. This is also a result but not a breakthrough as was wished for in this project.

6.2 Different focus

The consequences for the lack of breakthrough evidence is that the results of this project are still unsatisfactory. It was decided to keep looking for interesting findings, the goal is now to experiment more specific things.

The first step to this is to change the specific age variable to age intervals. This can be intervals of 10 or 20 years. Another possibility is to split the data in half and compare the groups of under and over 40 with each other. It would also be possible to compare the differences between children and adults. In figure 8, for most complaints there is a clear increase from 0 to around the age of 20. This might be interesting to investigate so in this section of the report we will look at the comparison of time spent at the ER between children and adults. Here, patients under the age of 20 are meant as children.

Before experimenting with this, some preprocessing is needed. There is a column added to the data with the name 'Type', which implies the type of patient and with this a child or adult is meant. After adding this variable, the plots could be created and can be found in figures 9 and 10.

Here one can see a clear difference between the median waiting time of the adults and the children. Those of the children is significantly lower, but there is not enough evidence to sustain this proof under a confidence interval of 95%. To go even further this was also checked for different complaints, but without confidence interval. This can be seen in figure 11.

Figure 5: The median stay at the ER (y-axis) plotted against the age for all complaints (x-axis)

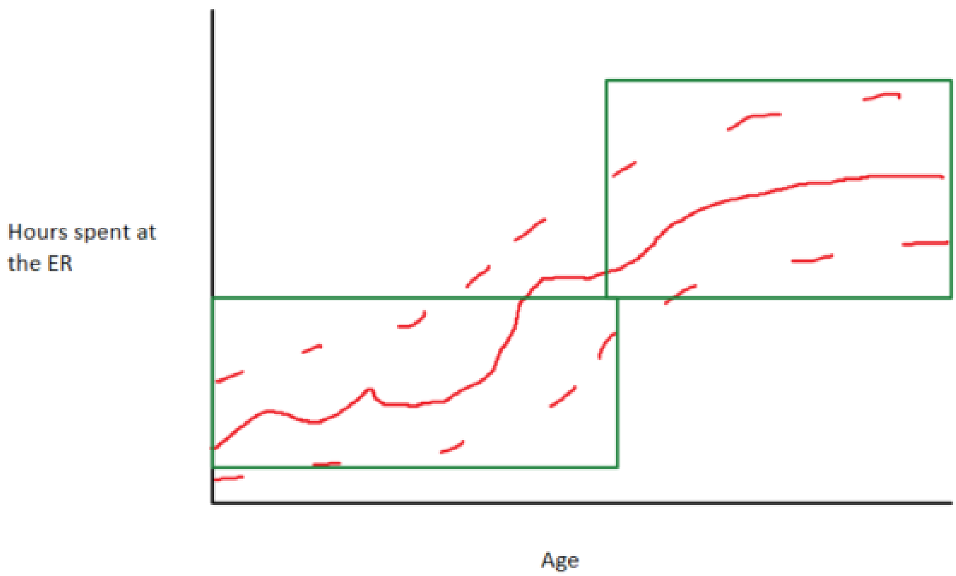

Figure 6: A sketch of how the relation between age and time of stay at the ER under one complaint should be if there was a significant correlation

From figure 11, one can conclude that among all complaints there is about the same distribution as with all complaints combined in one figure as was done in figure 9. Minor differences can be spotted. At 'gevallen', adults clearly stay longer than children, while at 'ontsteking' which means abscess the difference is minimal.

6.3 Discharge from the emergency room

Another option to investigate the variable 'dismissal to'. This column tells where the patient moves to when they leave the hospital. This variable can take a few standard values. These are:

- Sent home, control outpatient: The patient is sent home but should get a checkup at a outpatient clinic for any side effects of the treatment
- Sent home, no checkup
- Sent home, general practitioner checkup
- Sent to nursing home
- Sent to hospital
- Sent to outpatient clinic
- Transfer to different hospital
- Sent to general practitioners department
- Social medical indication: This means that the patient is too healthy to stay in the hospital but they are unable to return home yet

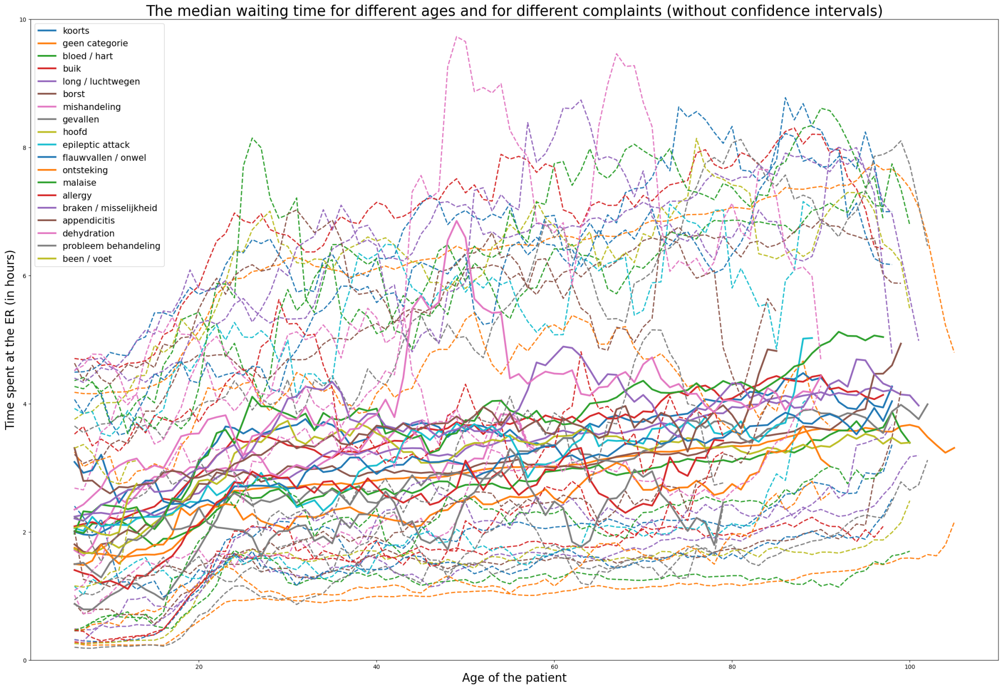

Figure 7: All plots from figure 5 plotted in one figure

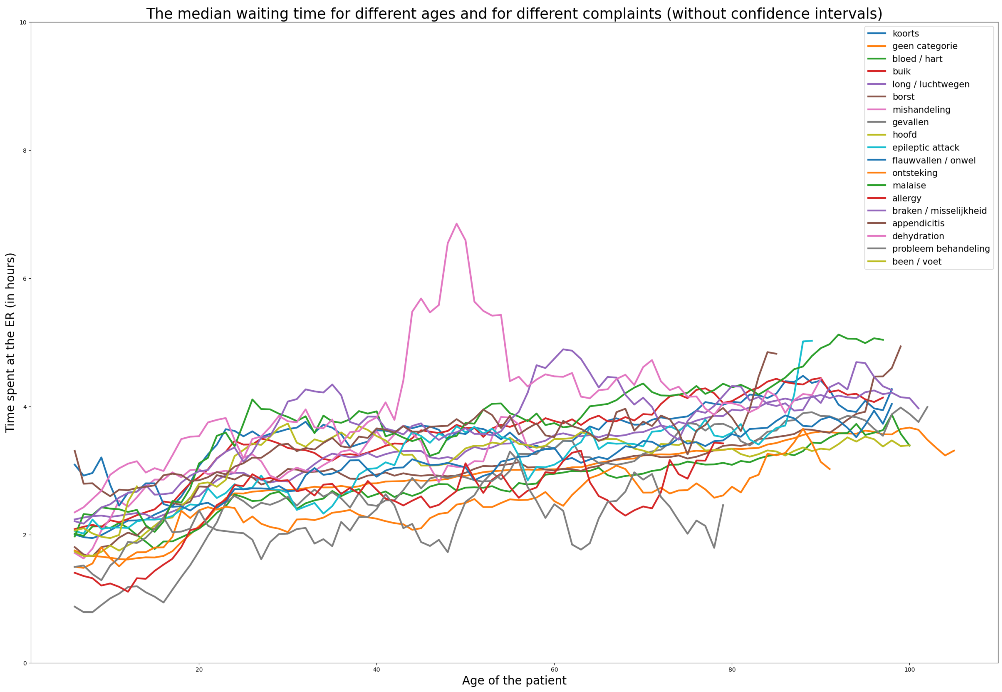

Figure 8: The median time of stay of the patients for every age (without confidence intervals)

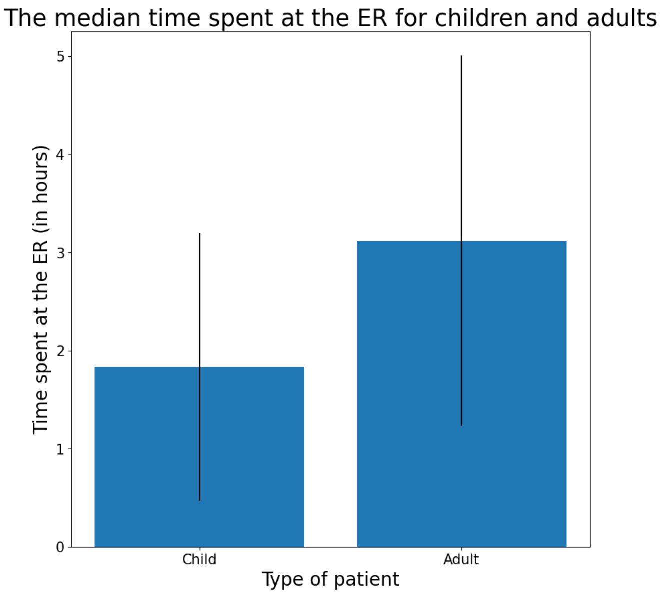

Figure 9: The median waiting time in hours for adults and children with their 95% confidence interval

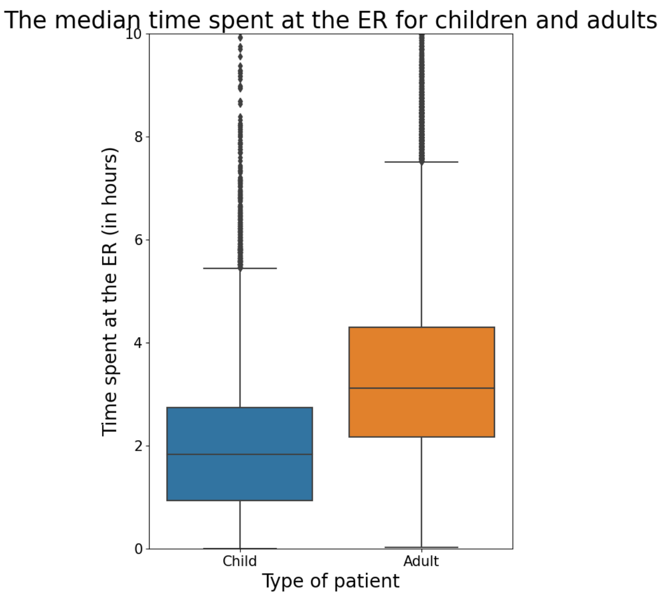

Figure 10: The median time spent at the ER in hours for adults and children in boxplots.

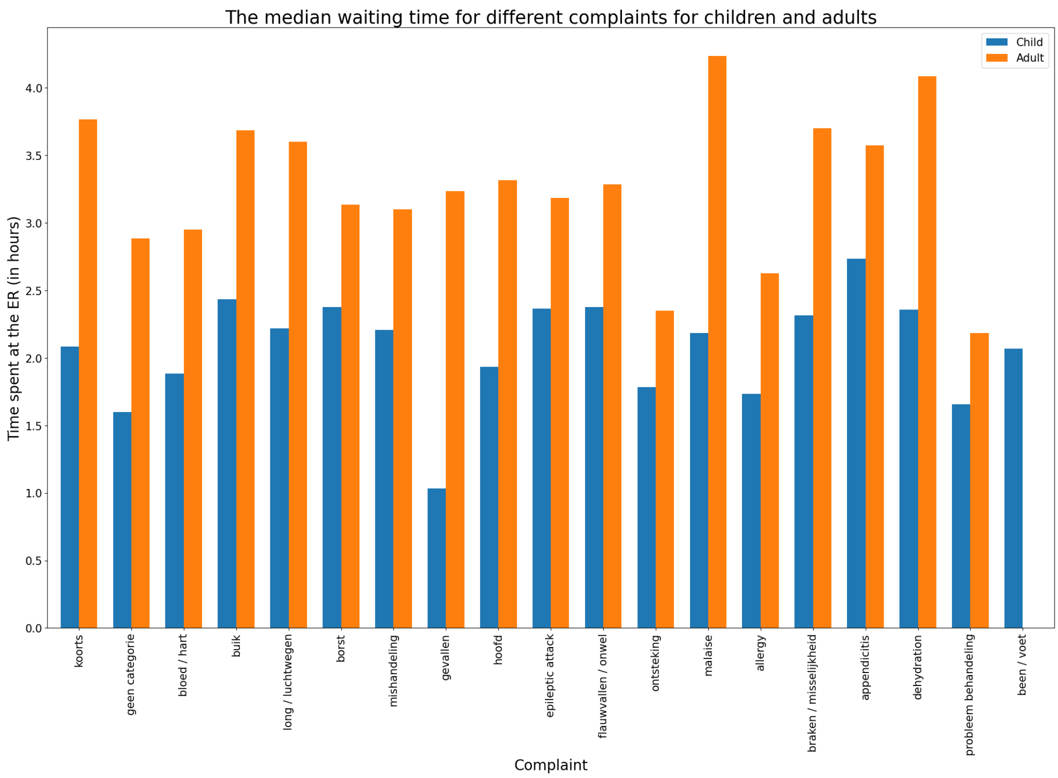

Figure 11: The median time spent at the ER in hours under different categories of complaints for both adults and children

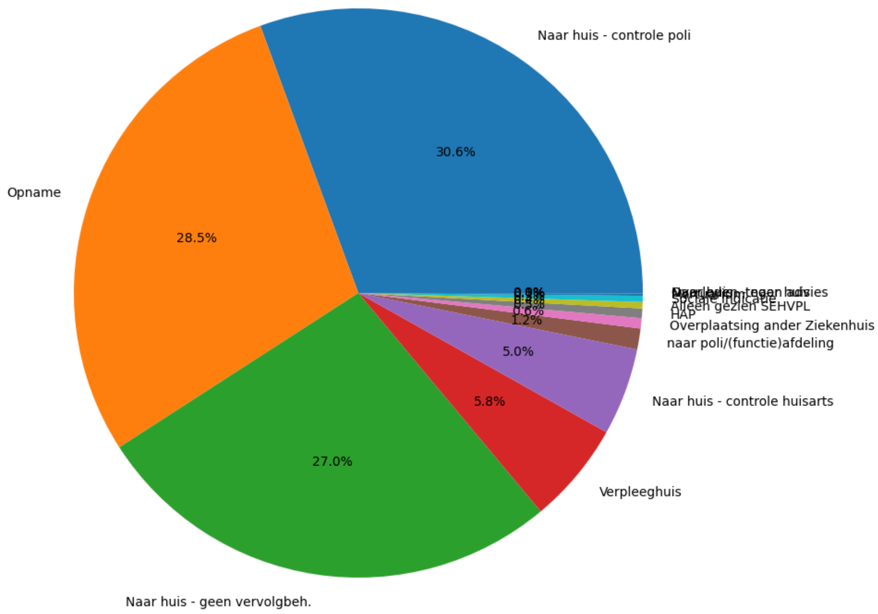

Figure 12: Pie chart of the frequency of occurence of 'Dismissal To' values in percentages of the entire dataset

- Mortuary: Bodies of deceased patient are stored here, the patient passed away
- Went home against advice of the staff
- Deceased
- Null, unknown.

These are quite a lot of items and during the data exploration it became clear that the dismissals are not equally distributed so when looking into these dismissals and their distributions, some complaints will be pooled and some will be removed. Moreover, the most important goal that should be achieved is to get to know which combination of complaint and age should be sent to the hospital immediately in order to keep the waiting time for the other patients shorter. An example of something this solves is that the people who decease at the ER can be removed from the data, since the ER does not send them somewhere when they arrive.

The pie chart displaying the entire data where the patients are dismissed to can be seen in figure 12. The largest three slices of the piechart are:

- 'Opname', this stands for send to the hospital
- 'Naar huis - controle poli', this stands for send home, control outpatient
- 'Naar huis - geen vervolgbeh.', this stands for send home, no further checkup needed

The other slices are so small they can be disregarded. However, during the experimentation, it is decided to take the 'Sent home, general practitioner checkup' together with the 'sent home, control outpatient'. This is because it is almost the same handling and leads to using 5% more data. In the end, 91.1% of the data is used and the other 8.9% is dropped for the next experimentations. This will not have a significant impact on the outcome of the project.

When some categories are pooled and the small groups are set to 'other' it gives the pie chart that can be seen in figure 13. The main dismissal destinations are evenly distributed varying from 27% to 33.5%. The goal is to investigate whether this is different under other complaints or ages so that they can be sent to the hospital earlier which leads to shorter waiting times and less patients overstaying.

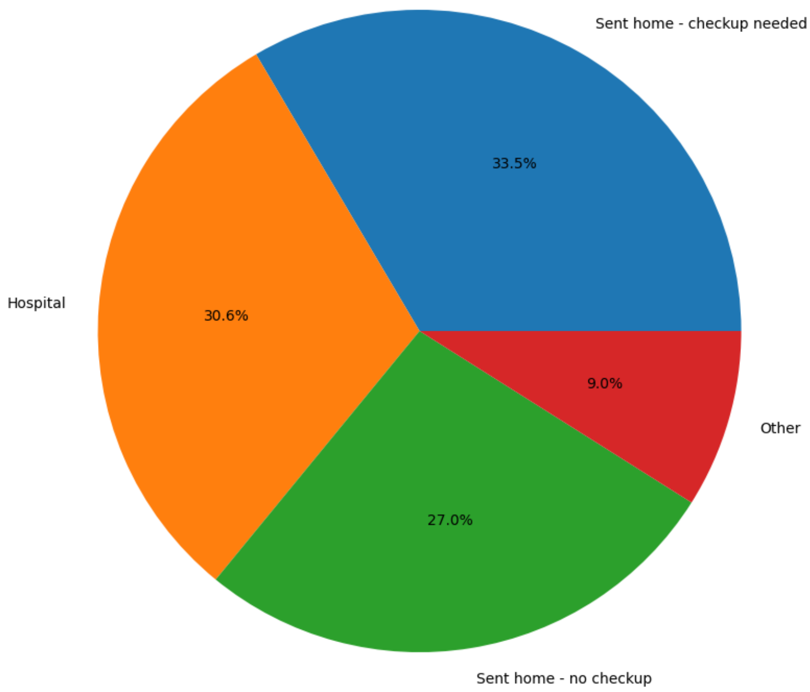

Figure 13: Pie chart of the frequency of occurence of 'Dismissal To' values, now with the data cleaned and the low occuring complaints put under 'other'

First, the dismissal for complaints only was checked. For this, a bar chart is created with four bars for every complaint, this bar chart can be seen in figure 14. Within this figure, the goal is to find where the blue line is significantly higher than the others, because the blue line stands for the patient being send to the hospital after treatment in the ER, this would imply that the patient could have been send to the hospital immediately.

When analysing figure 14, there are a few striking factors that stand out. With this, the relatively high blue lines are meant. A high blue line relative to the other lines for the same complaint means that the complaint results in the patient being sent to the hospital. With 'high' in this context is meant about double the height of the other lines individually. This is the case with the following

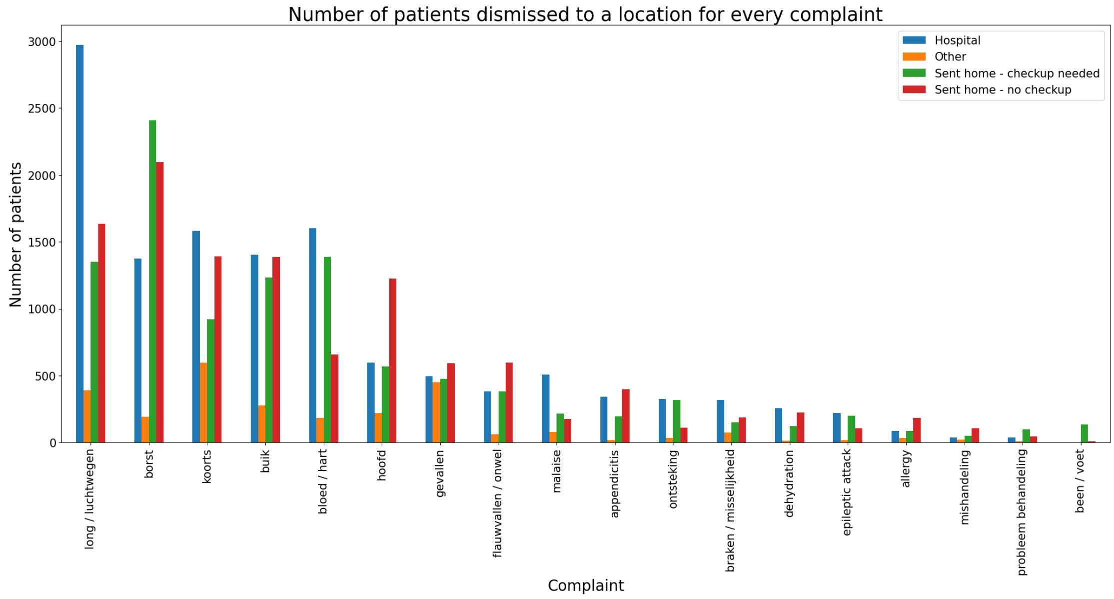

Figure 14: This bar chart tells how much patients are dismissed to a certain location for different complaint categories they fall under

complaints:

1. 'long/luchtwegen', which means lungs or respiratory system
2. 'malaise', which means feeling unwell without particular reason

The next step is to investigate the 'dismissal to' variable not only for the complaints but also for the different ages. To make this more easy, the two categories of complaints will be taken as sample because they have more patients who are sent to the hospital. Additionally, the age will be set as an interval of 20 years.

The data is preprocessed again. Now there are age intervals in the data. When looking at number of patients in each group, as can be seen in figure 15, it becomes clear that from some ages there are more patients in the data than others. This will be solved by undersampling and it will help with re-balancing the data. After rebalancing, every age interval occurs as frequent as the other ones and that means that the frequency bar chart can be seen in figure 16

With the undersampeled data, the bar charts are plotted for the complaint categories that gave respectively the highest value for patients dissmissed to the hospital. So 'long / luchtwegen', which is respiratory system and 'malaise'. This is executed for age group of 20 years. The results are visible in figure 17 and 18.

The combination of an older age leads more often to being dismissed to the hospital and the categories of complaints that lead to the most dismissals to the hospital show that older people, especially from the age of 60 and older, with these complaints have a higher probability of being send to the hospital after a visit at the ER.

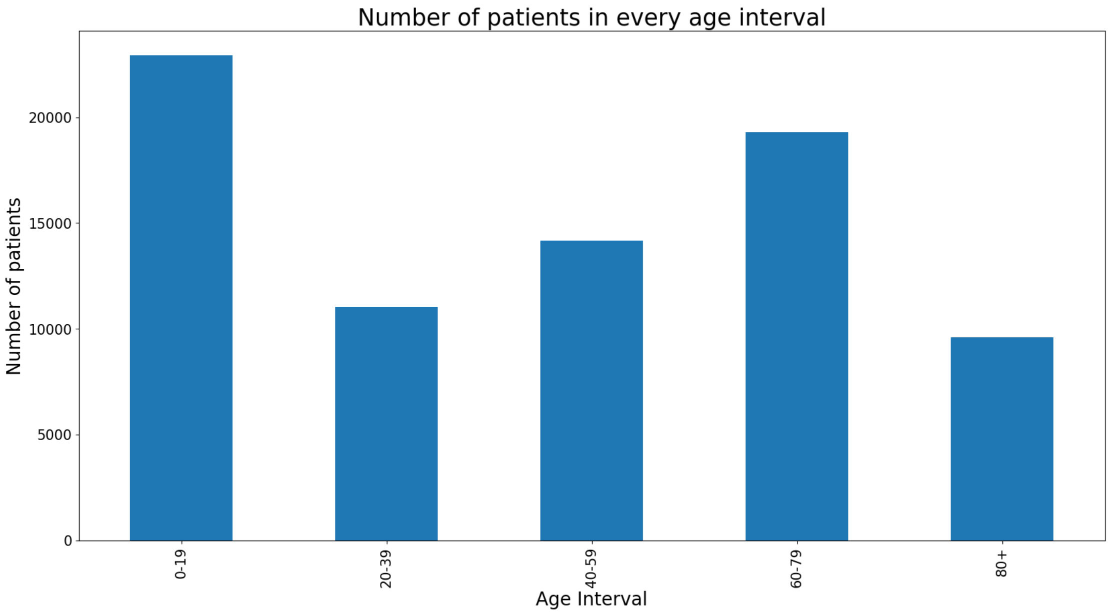

Figure 15: The number of patients in each age group

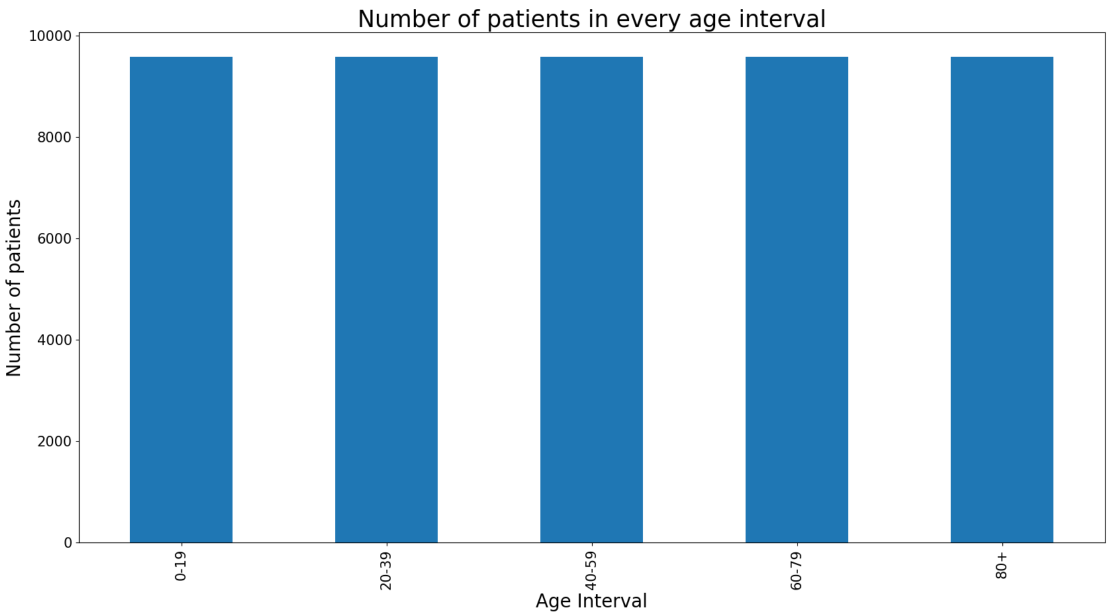

Figure 16: The number of patients in each age group after undersampling

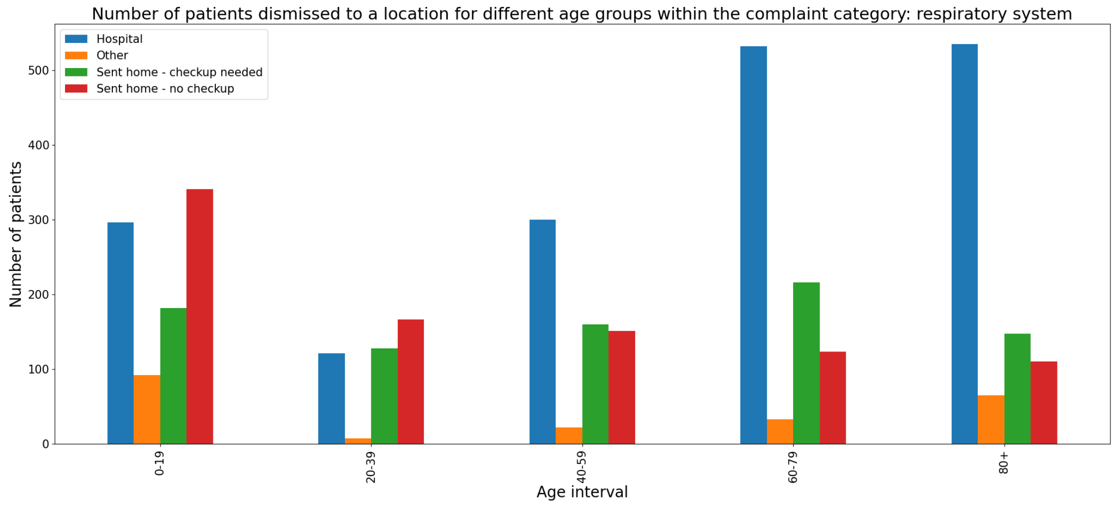

Figure 17: The bar chart that shows where people are transfered after their visit at the ER, for different age groups under the complaint category: respiratory system

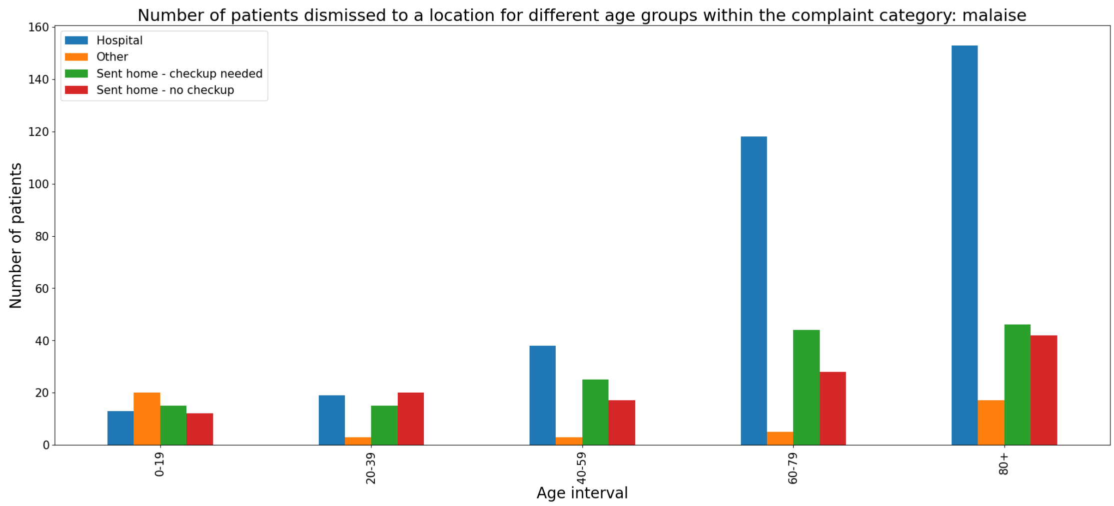

Figure 18: The bar chart that shows where people are transfered after their visit at the ER, for different age groups under the complaint category: malaise

6.4 Hypothesis test

The next step is to test this claim under a 95% confidence interval. Even though the graph shows a very clear difference between being sent to the hospital or home, a hypothesis test must be conducted in order to make a foolproof claim.

We set up a hypothesis, and test this hypothesis. If this test is conducted with a 95% confidence interval and the p-value is smaller than 0.05, the null hypothesis can be rejected. That is the goal since it proves our alternative hypothesis is true. The following hypothesis test was set up for this experiment.

$H_0 : P_{age<60} = P_{age\geq 60}$
$H_1 : P_{age<60} < P_{age\geq 60}$
$\alpha = 0.05$
$P =$ The probability a patient is sent to the hospital after visiting the ER.

The age of 60 is chosen because in the graphs in figures 17 and 18 it is clearly visible that the age groups from 60 and above have a significant higher bar than the intervals at younger ages.

The hypothesis will be tested using the two sample Z-test for proportions. This test is explained in the article of Ajitesh Kumar [3]. The theoretical formula can be found at Equation 1. Luckily, in python, there is a very simple formula that can be used for this z-test. The code and libraries used for this can be found in appendix 9.2. The results are visible in table 1. Here, a p-value of under 0.05 can be observed and this means that the null hypothesis can be rejected and the alternative hypothesis is approved in both cases. The probability of going to the hospital when suffering from respiratory complaints or malaise when you are over 60 years old are significantly higher than suffering from the same problem when you are under 60. To be precise, for respiratory system, 60,6% of all patients over 60 with the complaint are dismissed to the hospital. For malaise, this percentage is 59.8%. It would save the ER a lot of time if people who enter the ER and are 60 or over and suffering from the previously named complaints, are send to the hospital immediately when entering the ER. This is part of the plan for emergency rooms to reduce the number of overstaying patients.

$$Z = \frac{\hat{p_1} - \hat{p_2}}{\sqrt{\hat{p}(1-\hat{p})(\frac{1}{n_1} + \frac{1}{n_2})}} \tag{1}$$

	Respiratory system	Malaise
Test statistic	14.717	5.854
p-value	$5.059e^{-49}$	$4.809e^{-9}$

Table 1: Table of statistics of the proportional z-test

7 Follow up plan ER

After previous results, the main goal is to sketch a plan that might help the ER to reduce number of patients that overstay, so stay at the ER for longer than 4 hours. The approach that was taken during this project is to investigate which groups of patients lead to the longest time spent at the ER and try to focus on this group. Unfortunately, there was no group of patients that stayed significantly longer than the rest of the patients.

Then it was decided to experiment with the 'Dismissal To' variable. If a particular patient group is almost always sent to the hospital, it would be convenient to send them to the hospital immediately when they arrive at the ER. This way the pressure on the ERs can be eased and this will automatically lead to less overstaying patients. The condition that should be taken into account is that hospitals should have enough hospital beds and personnel available. If this is not the case, the ERs are not able to send patients to the hospital without trying to help or treat them because this would lead to much pressure on the capacity of the hospital.

From the results of this investigation, there can be concluded that if the ER considers to send people straight to the hospital, they can send the people over sixty that suffer from respiratory problems or malaise. From all patients that meet these conditions about 60% goes to the hospital. This number is not overwhelming so it is suggested that the ER checks how severe the situation is for this patient and when it is over a certain level, they are send to the hospital. This will lead to the reduction of patients staying over four hours.

8 Conclusion

To conclude, there was no specific group of patients that stayed at the ER longer than other groups because of their conditions. However, older patients tend to stay longer than younger patients. For example, the median stay at the ER for adults is higher than the stay of children, but this difference is not significant, there is no difference when adding 95% confidence bounds. When looking at the 'Dismissal To' variable, there were two complaint categories that had higher values for being dismissed to the hospital. Those were investigated further. The patients were also split into two age groups, 'under 60' and '60 and over', this was done so the effect of age could be measured. The results were interesting, there was a clear difference in being sent to the hospital for patients under and over 60 for these complaints. The suggestion for the ER is to look at the severity of the injury of a patient that is over 60 and suffering from problems with their respiratory system or malaise. This way patients can be dismissed from the ER to the hospital immediately at arrival and this will automatically lead to a shorter waiting times at the ER and a smaller number of patients overstaying.

9 Appendix

9.1 Merge complaints

```
merge_comp = pd.DataFrame({'old_complaint':basis_klacht, 'complaint 1': 0, 'complaint 2': 0, 'com


lst_borst = []
lst_buik = []
count = -1

# Merge the complaints that fit within one category, to that one category

for i in merge_comp['old_complaint']:
    count += 1

    # chest
    if 'borst' in i:
        if merge_comp['complaint 1'][count] == 0:
            merge_comp['complaint 1'][count] = 'borst'
        elif merge_comp['complaint 2'][count] == 0:
            merge_comp['complaint 2'][count] = 'borst'
        else:
            merge_comp['complaint 3'][count] = 'borst'
    elif 'pob' in i:
        if merge_comp['complaint 1'][count] == 0:
            merge_comp['complaint 1'][count] = 'borst'
        elif merge_comp['complaint 2'][count] == 0:
            merge_comp['complaint 2'][count] = 'borst'
        else:
            merge_comp['complaint 3'][count] = 'borst'

    # belly
    if 'buik' in i:
        if merge_comp['complaint 1'][count] == 0:
            merge_comp['complaint 1'][count] = 'buik'
        elif merge_comp['complaint 2'][count] == 0:
            merge_comp['complaint 2'][count] = 'buik'
        else:
            merge_comp['complaint 3'][count] = 'buik'

    # fever
    if 'geen koorts' in i:
        pass
    elif 'koorts' in i:
        if merge_comp['complaint 1'][count] == 0:
            merge_comp['complaint 1'][count] = 'koorts'
        elif merge_comp['complaint 2'][count] == 0:
            merge_comp['complaint 2'][count] = 'koorts'
```

```
    else:
        merge_comp['complaint 3'][count] = 'koorts'

# resperatory system
if 'dyspnoe' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'long / luchtwegen'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'long / luchtwegen'
    else:
        merge_comp['complaint 3'][count] = 'long / luchtwegen'
elif 'benauwd' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'long / luchtwegen'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'long / luchtwegen'
    else:
        merge_comp['complaint 3'][count] = 'long / luchtwegen'
elif 'exc copd' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'long / luchtwegen'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'long / luchtwegen'
    else:
        merge_comp['complaint 3'][count] = 'long / luchtwegen'
elif 'covid' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'long / luchtwegen'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'long / luchtwegen'
    else:
        merge_comp['complaint 3'][count] = 'long / luchtwegen'
elif 'hoesten' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'long / luchtwegen'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'long / luchtwegen'
    else:
        merge_comp['complaint 3'][count] = 'long / luchtwegen'
elif 'pneumonie' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'long / luchtwegen'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'long / luchtwegen'
    else:
        merge_comp['complaint 3'][count] = 'long / luchtwegen'
elif 'longontsteking' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'long / luchtwegen'
```

```
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'long / luchtwegen'
    else:
        merge_comp['complaint 3'][count] = 'long / luchtwegen'


# cardiovascular system
if 'cva' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'bloed / hart'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'bloed / hart'
    else:
        merge_comp['complaint 3'][count] = 'bloed / hart'
elif 'af' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'bloed / hart'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'bloed / hart'
    else:
        merge_comp['complaint 3'][count] = 'bloed / hart'
elif 'acs' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'bloed / hart'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'bloed / hart'
    else:
         merge_comp['complaint 3'][count] = 'bloed / hart'
elif 'ritmestoornis' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'bloed / hart'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'bloed / hart'
    else:
        merge_comp['complaint 3'][count] = 'bloed / hart'
elif 'palpitaties' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'bloed / hart'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'bloed / hart'
    else:
        merge_comp['complaint 3'][count] = 'bloed / hart'
elif 'pca' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'bloed / hart'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'bloed / hart'
    elif merge_comp['complaint 3'][count] == 0:
        merge_comp['complaint 3'][count] = 'bloed / hart'
```

```
elif 'dec cordis' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'bloed / hart'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'bloed / hart'
    elif merge_comp['complaint 3'][count] == 0:
        merge_comp['complaint 3'][count] = 'bloed / hart'
elif 'geen trombolise' in i:
    pass
elif 'trombolise' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'bloed / hart'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'bloed / hart'
    else:
        merge_comp['complaint 3'][count] = 'bloed / hart'

# head trauma
if 'trauma capitis' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'hoofd'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'hoofd'
    else:
        merge_comp['complaint 3'][count] = 'hoofd'
elif 'hoofdpijn' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'hoofd'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'hoofd'
    else:
        merge_comp['complaint 3'][count] = 'hoofd'

# fainting
if 'collaps' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'flauwvallen / onwel'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'flauwvallen / onwel'
    else:
        merge_comp['complaint 3'][count] = 'flauwvallen / onwel'
elif 'onwel' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'flauwvallen / onwel'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'flauwvallen / onwel'
    else:
        merge_comp['complaint 3'][count] = 'flauwvallen / onwel'
```

```
# malaise
if 'malaise' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'malaise'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'malaise'
    else:
        merge_comp['complaint 3'][count] = 'malaise'

# appendicitis
if 'appendicitis' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'appendicitis'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'appendicitis'
    else:
        merge_comp['complaint 3'][count] = 'appendicitis'
elif 'blinde darm' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'appendicitis'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'appendicitis'
    else:
        merge_comp['complaint 3'][count] = 'appendicitis'

# epileptic attack
if 'insult' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'epileptic attack'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'epileptic attack'
    else:
        merge_comp['complaint 3'][count] = 'epileptic attack'

# allergic reaction
if 'allergi' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'allergy'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'allergy'
    else:
        merge_comp['complaint 3'][count] = 'allergy'

# mishandeling / assault
if 'mishandeling' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'mishandeling'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'mishandeling'
```

```
    elif merge_comp['complaint 3'][count] == 0:
        merge_comp['complaint 3'][count] = 'mishandeling'

# dehydration
if 'dehydratie' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'dehydration'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'dehydration'
    elif merge_comp['complaint 3'][count] == 0:
        merge_comp['complaint 3'][count] = 'dehydration'
elif 'uitdroging' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'dehydration'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'dehydration'
    elif merge_comp['complaint 3'][count] == 0:
        merge_comp['complaint 3'][count] = 'dehydration'

# leg
if 'spaakverwonding' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'been / voet'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'been / voet'
    elif merge_comp['complaint 3'][count] == 0:
        merge_comp['complaint 3'][count] = 'been / voet'

# vomiting / nausea
if 'braken' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'braken / misselijkheid'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'braken / misselijkheid'
    elif merge_comp['complaint 3'][count] == 0:
        merge_comp['complaint 3'][count] = 'braken / misselijkheid'

# problem after treatment
if 'probleem behandeling' in i:
    if merge_comp['complaint 1'][count] == 0:
        merge_comp['complaint 1'][count] = 'probleem behandeling'
    elif merge_comp['complaint 2'][count] == 0:
        merge_comp['complaint 2'][count] = 'probleem behandeling'
    elif merge_comp['complaint 3'][count] == 0:
        merge_comp['complaint 3'][count] = 'probleem behandeling'

# ontsteking
if 'abces' in i:
    if merge_comp['complaint 1'][count] == 0:
```

```
            merge_comp['complaint 1'][count] = 'ontsteking'
        elif merge_comp['complaint 2'][count] == 0:
            merge_comp['complaint 2'][count] = 'ontsteking'
        elif merge_comp['complaint 3'][count] == 0:
            merge_comp['complaint 3'][count] = 'ontsteking'
    if 'ontsteking' in i:
        if merge_comp['complaint 1'][count] == 0:
            merge_comp['complaint 1'][count] = 'ontsteking'
        elif merge_comp['complaint 2'][count] == 0:
            merge_comp['complaint 2'][count] = 'ontsteking'
        elif merge_comp['complaint 3'][count] == 0:
            merge_comp['complaint 3'][count] = 'ontsteking'

    # fell
    if 'gevallen' in i:
        if merge_comp['complaint 1'][count] == 0:
            merge_comp['complaint 1'][count] = 'gevallen'
        elif merge_comp['complaint 2'][count] == 0:
            merge_comp['complaint 2'][count] = 'gevallen'
        elif merge_comp['complaint 3'][count] == 0:
            merge_comp['complaint 3'][count] = 'gevallen'
    elif 'val van trap' in i:
        if merge_comp['complaint 1'][count] == 0:
            merge_comp['complaint 1'][count] = 'gevallen'
        elif merge_comp['complaint 2'][count] == 0:
            merge_comp['complaint 2'][count] = 'gevallen'
        elif merge_comp['complaint 3'][count] == 0:
            merge_comp['complaint 3'][count] = 'gevallen'

merge_comp
```

9.2 Libraries and z-test

```
import numpy as np
import pandas as pd
from statsmodels.stats.proportion import proportions_ztest

filt = data_downsampled[data_downsampled['Complaint1'] == 'malaise']
size2 = filt.groupby(['sixtyOver','Dismissal New']).size()
unstck = size2.unstack()

def avgProb(ind, grp):
    return size2[ind]/sum(unstck.loc[grp])

probHospOver60 = avgProb(0, '60 and over')
probOtherOver60 = avgProb(1, '60 and over')
probHomeCheckOver60 = avgProb(2, '60 and over')
probHomeNoCheckOver60 = avgProb(3, '60 and over')
```

```
probHospUnd60 = avgProb(4, 'Under 60')
probOtherUnd60 = avgProb(5, 'Under 60')
probHomeCheckUnd60 = avgProb(6, 'Under 60')
probHomeNoCheckUnd60 = avgProb(7, 'Under 60')

# Define the counts and sample sizes for the two groups
count_over_60 = probHospOver60 * sum(unstck.loc['60 and over'])
n_over_60 = sum(unstck.loc['60 and over'])

count_under_60 = probHospUnd60 * sum(unstck.loc['Under 60'])
n_under_60 = sum(unstck.loc['Under 60'])

# Perform the hypothesis test
count = np.array([count_over_60, count_under_60])
nobs = np.array([n_over_60, n_under_60])

stat, pval = proportions_ztest(count, nobs)

# Print the test statistic and p-value
print('Test Statistic:', stat)
print('p-value:', pval)
```

References

[1] Houten. *Internationale Statistische Classificatie van Ziekten en met Gezondheid verband houdende Problemen*. 2021. URL: https://www.rivm.nl/media/volksgezondheid-en-zorg/centrum-gezondheid-en-maatschappij/ICD10TTversie_okt_2013/html/All%5C%20Chapters/Chapter%5C%209/html/content.html.

[2] Bart Kiers. "Langer wachten op spoedeisende hulp: Doorstroomsnelheid op de seh gestegen". In: *Zorgvisie* 47 (2017), pp. 16–19.

[3] Ajitesh Kumar. *Two sample z-test for proportions: Formula amp; Examples*. July 2022. URL: https://vitalflux.com/two-sample-z-test-for-proportions-formula-examples/.

[4] Marek Laskowski et al. "Models of emergency departments for reducing patient waiting times". In: *PloS one* 4.7 (2009), e6127.

[5] Larry Purnell. "Reducing waiting time in emergency department triage". In: *Nursing management* 26.9 (1995), 64Q.