MCOMD2AIC Artificial Intelligence Computing – Assignment 1

Nathan Shuttleworth

Due 22/4/21

# Introduction

In this investigation, I am looking to use clustering to find new insight from the England Covid-19 data provided by the government on the 22nd February 2021. I am going to primarily look at the link between a person’s Standard Occupation Code (SOC) and the likelihood that they will test positive for Coronavirus, as well as the actual percentage who do test positive, again linked by SOC.

An SOC is defined as, by the Office for National Statistics, as: “The object to be classified using the Standard Occupational Classification (SOC) is the concept of a ‘job’. Defined as a set of tasks or duties to be carried out by one person, the notion of a job represents a basic element in the employment relationship… Jobs are classified into groups according to the concepts of ‘skill level’ and ‘skill specialisation’.” (Office for National Statistics, SOC Vol. 1, 2020). What the SOC is, essentially a code given to each formal job in the country. There are two ways of classifying SOC, by 2-digits or 4-digits. The 2-digit codes run from 10- 99, and the 4-digit codes go from 1000 to 9999. I will primarily be using the 4-digit format for this investigation.

The SOC ranks occupations based on apparent skill level and qualification, with jobs such as Corporate managers occupying the lower numbers, and what is seen as unskilled, primary sector roles such as shelf fillers and waiters occupying the numbers closer to the end.

I hypothesise that there will be a correlation between the percentage of a sample testing positive for Covid-19 and the SOC, people with a higher SOC will be more likely to test positive. I think this because Jobs with a higher SOC tend to be more manual, they are more likely to take place outside and so cannot be done from the safety of the home. This puts these workers in more contact with others and thus increasing their chances of catching the virus and testing positive.

# Method

## Preparing the data

I began my investigation by gathering all the necessary data. I downloaded a copy of the Covid Infection Survey from the Government website. I then looked at table 3b, which gives the percentage of people testing positive for Coronavirus by 4-digit SOC. I chose to use this table instead of 3a which gives the same but for 2-digit SOC because there is more data available and so should provide deeper insight.

The software I primarily used was Microsoft Excel, because of its ease of use, wide availability and built-in formula functionality.

I created a new CSV (Comma Separated Value) document and copied over the data from columns A, D and E. This is the data representing the 4-digit SOC, size of the sample being tested and the percentage of people in that sample who tested positive. There is a wide variety in the number of people being tested, with some samples having over a thousand participants others only having as few as 11. Due to this, I decided to calculate which of these values could be considered outliers, to allow the investigation to be a better representation of what I am trying to investigate.

Firstly, I calculated upper and lower quartiles for the sample size, which are 106 and 449, making the interquartile range 343. I also calculated the median value which is 224. Using these values, I calculated the upper and lower bounds for the range of acceptable data, which are 408.5 and 963.5. Therefore, I will ignore any data with a sample size lower than -408.5 and greater than 963.5.

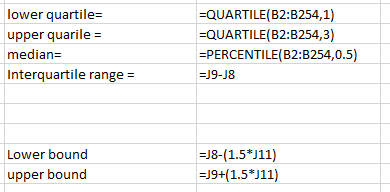


Figure - Formulae used to calculate outliers

Once this data had been removed, I then also removed the sample size column and saved the data as a CSV file, as well as removing the data headers and using the format controls to put a space in front of all the values in the 2nd column. The data was now fully formatted for the next part.

## Processing Approach

Firstly, I needed to find a suitable implementation of the K-means clustering algorithm. The one I adapted was made in Python (Turner, S., 2021). This implementation works using the sripy.cluster, numpy, csv and matplotlib modules. It first opens the CSV file and goes through each line printing them out and adding them to an array as the data type float. The values are then plotted on a scatter graph, and the centroids of the clusters are shown.

K-means clustering is a very simple and commonly used clustering algorithm, and it is well suited to finding anomalies and finding the centre of a group of data. The algorithm works by having the user determine the number of centroids, and a given number of inputs. Initially the centroids are place randomly. Each centroid is then assigned an input that is closest to it in Euclidean distance. A new centroid is then found by taking the average of all the points assigned to that cluster (Mubaris, NK., 2017). These steps then repeat until the centroids stop moving.

I copied the code from the first file into the Python IDLE, and made some changes such as editing the number of centroids from 2 to 9 and also changed the file to be read from. I chose 9 centroids because the SOC data goes from 1000 to 9999, there would likely be a cluster in the middle of each of the intervals.

Once I had a graph created showing the clusters, I then got the code to output the co-ordinates of the centroids:

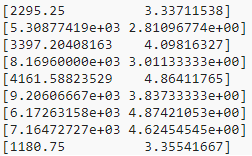


Figure 2- Co-ordinates of centroids

I then formatted this data so that it was more readable and easier to manipulate (figure 3), I changed it out of standard from and rounded it to the nearest hundredth.

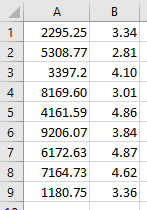


Figure 3- Centroid co-ordinates formatted ready for plotting

# Results

The result of the data that I ran through the clustering algorithm, which was to find a link between SOC and likelihood of testing positive for covid-19, gave this result:

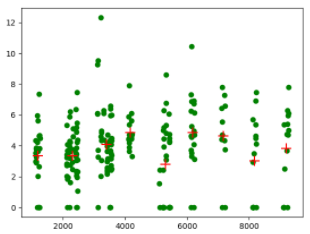


Figure - Graph showing SOC against % testing positive in a sample

Looking at the graph, there is a very slight correlation between the two sets of values, and there are some clear clusters.

Figure - Graph of cluster centroids with line of best fit

To help find better insight and correlation between the data, I decided to plot just the centroids on a graph with a line of best fit. I chose to use a polynomial best fit line with order 2 as it would better suit the curve that the data obviously shows. Whilst there is a correlation, it is neither a positive or negative one. However, when examining the graph in figure 4, I noticed that the centroid for the ~5000 SOC cluster is a lot lower than many of the points appear to be. With this in mind, I redrew the graph in figure 5, but removed this erroneous seeming data.

Figure - Redrawn graph removing erroneous data

This new graph shown in figure 6 shows a much more promising trend between the values, there is a clearer highpoint around the 5000 SOC mark.

From the initial graph of data that was put the through the K-means clustering algorithm, I did not get the result I was expecting. I was anticipating for there to be a more noticeable positive correlation between SOC and the percentage of people testing positive, but for the most part this was not true. Instead, the SOC range which was more likely to test positive was in the 4000- 6000 range. This is not wholly accurate however since the centroid for the ~5000 SOC is the lowest, but looking at the graph it appears that there is a larger cluster higher than where the centroid is. This implies that the k-means algorithm is not wholly accurate.

A conclusion that I can draw from this is that contrary to my initial belief, occupations with a mid-level SOC have a slightly higher amount of people testing positive. One theory as to why this could be the case is because jobs at the low end are ones that can take place from the safety of the home, considerably lowering the chances of contracting the virus. At the other end, jobs that are regularly done outside are more likely to have adequate precautions in place. Meanwhile, mid-range jobs include a lot of what could be described as ‘office jobs’, as well as a lot of occupations that have alternately closed and opened, so the precautions that these people have are not likely to be as well developed, and so this laxness is what makes it more likely for them to contract the virus.

# Conclusions

In conclusion, I think that this investigation went fairly well in terms of being able to find new insights from the data given. Although the results gave a different conclusion to what I was expecting, I was still able to find one by using clustering and come up with a reason as to why that was the case.

One aspect of this investigation that could have been improved is regarding the implementation of the K-means clustering algorithm. Whilst it worked effectively in creating clusters for 8/9 groups of data, it made an error with one which affected the trendlines. This data should have been the highest if you were going by the trend of the rest of the data, but instead it was among the lowest by several %.

In doing this investigation, I also noticed an improvement that could be made to the original report. With the range of SOC values, and this applies for both 2- and 4-digit formats, there is not as wide a range as it initially seems. Whilst there is a wide range of data, going from the top to bottom of the SOC range, in between each 1000 the values only go from X100 to X300 at the most, rather than covering the full range from X100 to X999. This is the reason why the graph in figure 4 is rendered with discrete columns of data, rather than having the data more scattered around the graph and allowing clusters to be seen more naturally.

Another way in which this project could have been improved is by using a more substantial dataset. Alongside needing more variety in SOC values, there are also flaws in the data with regards to the amounts of people being tested in each sample. Some of the samples were very low, and so for the % of people who tested positive the result was zero. Whilst there are certainly occupations where you are less likely to test positive, with the current global situation I believe it is not impossible to return a positive test. Whilst I did calculate what would have been considered an outlier for the sample size, this only excluded extremely high values. To improve this if the investigation were to be repeated, I would ensure that the data supplied has an acceptable sample size.

An explanation as to why some groups tested 100% negative could be due to false negatives. As detailed by C. Mayers and K. Baker (2020), false negatives could be caused by “Poor sampling technique… sample degradation… sampling too early and sampling too late”. As home testing has become more and more common, especially in the last few months, it is not unlikely that many of the Covid-19 tests used to make up this survey were done by people at home. Average people are more likely to make mistakes when performing. I have not been able to find any information about how the tests used to make up the dataset were performed, so I can only assume that it would have been done as a mix of both at dedicated testing centres and home kits distributed by the NHS.

However, there is a downside to using a larger data set. Whilst K-means is a relatively simple and efficient clustering algorithm, as the dataset size increases, so does the amount of time and resources required to process the data. This could become quite expensive, and the cost may outweigh the value that the data and insights derived from it provides. There is also the increased cost from gathering data from more people. Whilst a vast amount of Covid-19 tests take place every day, there is not as much of a full infrastructure to ask people about their occupations at the same time as doing the tests.

If I were to repeat this investigation, I would also make some changes to how I would do it. Firstly, I would spend more time looking over the data, to see if there were any patterns that were immediately noticeable. My reason behind this is that the data I analysed did not have as strong a correlation as would have hoped, I was not able to glean as conclusive insights from it as I had hoped. Another change I would make is to program an implementation of K-means myself, as opposed to modifying one. Whilst I would still use libraries to help, putting it together myself would mean I have a better understanding of how it worked and more control over it.

# References

Mayers,C., Baker,K. (2020) *Impact of false-positives and false-negatives in the UK’s COVID-19 RT-PCR testing programme.* Published 3/6/2020. Accessed 31/3/21. <https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/895843/S0519_Impact_of_false_positives_and_negatives.pdf>

Mubaris,NK (2017) *K-Means Clustering in Python.* Published 01/10/2017*.* Accessed 30/3/21 <https://mubaris.com/posts/kmeans-clustering>

National Statistics, Office for (No definite author stated) (2020) *SOC 2020 Volume 1: Structure and descriptions of Unit Groups.* Updated 30th September 2020. Accessed 24/3/2021. <https://www.ons.gov.uk/methodology/classificationsandstandards/standardoccupationalclassificationsoc/soc2020/soc2020volume1structureanddescriptionsofunitgroups#toc>

Sutherland, E., Willis, Z. and Fordham E (2021) *COVID-19 Infection Survey*. Office for National Statistics. Published 22nd February 2021. Accessed 23/3/2021. [Coronavirus (COVID-19) infections in the community in England - Office for National Statistics](https://www.ons.gov.uk/peoplepopulationandcommunity/healthandsocialcare/conditionsanddiseases/datasets/coronaviruscovid19infectionsinthecommunityinengland)

Turner,S. (2021) *Covid1* Accessed 27/3/21. <https://trinket.io/python3/164fdcdbc6>