Rotman School of Management
UNIVERSITY OF TORONTO

Rotman

Group Number:	7
Assignment Title:	Group Assignment 1
Course Code:	RSM8413
Instructor Name:	Gerhard Trippen

In submitting this **group** work for grading, we confirm:

- That the work is original, and due credit is given to others where appropriate.
- That all members have contributed substantially and proportionally to each group assignment.
- That all members have sufficient familiarity with the entire contents of the group assignment so as to be able to sign off on them as original work.
- Acceptance and acknowledgement that assignments found to be plagiarized in any way will be subject to sanctions under the University's Code of Behaviour on Academic Matters.

Please **check the box and record your student number** below to indicate that you have read and abide by the statements above:

☐	1002183031	☐	1006604701
☐	1002897378	☐	1007045118
☐	1007554745	☐	1005627403

Executive Summary

Background

As new employees of the Public Health Agency of Canada, we were asked to examine a dataset containing information on COVID-19-related behaviours collected by the Institute of Global Health Innovation Imperial College London. We were tasked with preparing the data, carrying out exploratory data analysis and creating an optimal k-NN model to carry out our analysis of the dataset.

Most salient findings

Through our Exploratory Data Analysis (EDA), we found that while observations were different for Canada and the USA, the trends in all variables were quite similar. Also, there were a lot of variables with missing values which we dealt with using a variety of methods. We also noticed that the data is not representative in terms of age.

After our EDA, we created a KNN classifier. A key part of this was identifying the relevant predictors to include in the model and to do this, we narrowed down the pool of predictors using our findings in the EDA section and by examining the correlations between our variables. Finally, after fitting the model, we compared the importance of the variables included in our model with the use of DecisionTreeClassifier. The most important predictor is i12_health_16, “Avoided going to shops”. This makes sense as avoiding going to shops would be a very good indicator of whether an individual avoids going out in general.

For the model creation portion of our task, we created a version of the target variable, “Avoided going out in general” (i12_health_6), which binned its five existing categories into two categories that represented going out versus not going out. This essentially turned our target variable into a binary variable. After tuning our model parameters and carrying out our predictions, we found that the KNN classifier had a higher accuracy when the binary version of the target variable was used. A binary variable also has positive feasibility implications for future studies as it is easier to gauge if someone goes out or not versus gauging how much a person goes out.

The model was tuned by determining the optimal k value and by using the Decision Tree Classifier feature importance score. We used K-fold cross-validation processes to avoid overfitting. In the end, we found that the optimal K value was k= 19. When this value of k was used, the KNN model had the highest accuracy.

Data preparation

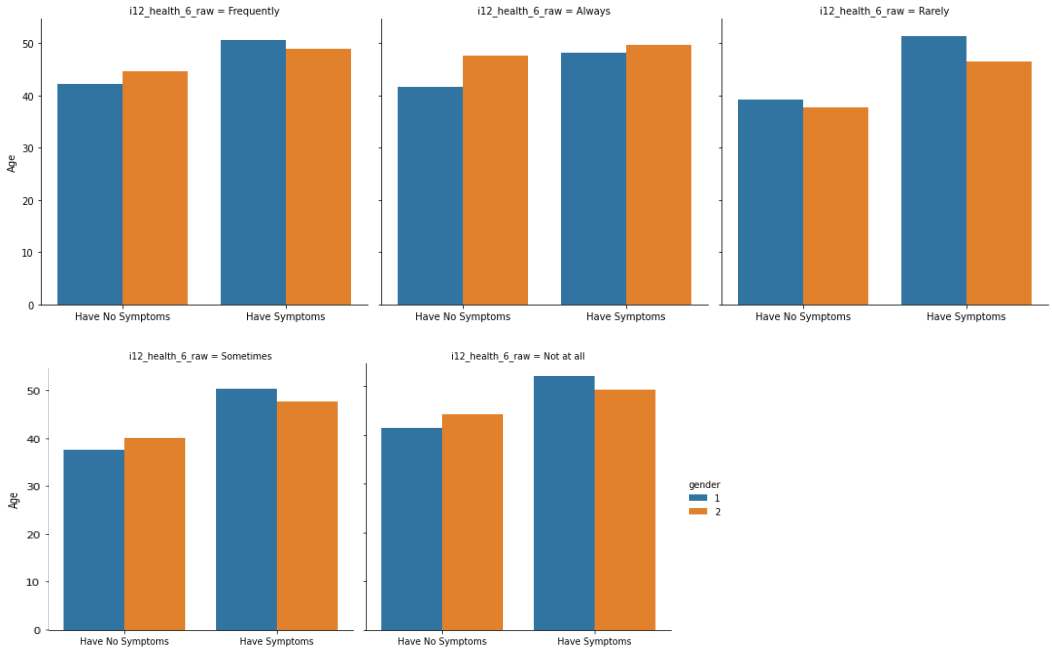
- Import the packages and modules from packages needed.
- Read the “COVID-19BehaviorData_CAN_USA.csv” file provided.

- Data Cleaning & Manipulation
    - Transform variables that have text input to corresponding values according to the "DataKeyCOVID-19BehaviorData_CAN-USA.csv" file, output csv file with cleaned data.
    - Read cleaned data file and original data file again, creating a new data frame by joining the two data frames. We consider using the original data file for EDA, while continuing to clean the new data frame for KNN.
    - Create the variable "Country" (with values "USA and CAN"), according to the variable "RecordNo". That is mainly for EDA.
    - Scale numeric variables, including variables age, i1_health, i2_health, i7a_health, i13_health using min_max_scaler.
    - Redefine the target variable based on if they avoided going out frequently or not, so i12_health_6 will be given a value of 0 or 1 based on the original value. We will use both original and redefined target variables for our further analysis and compare them.
    - Handle missing data. (A diagram of number of missing values for each variable is in Appendix)
        - Drop variables that have over 50% missing data. These include I5a_health, i6_health, I7b_health, I8_health, I12_health_9, i12_health_10, I14_health_1, I14_health_2, I14_health_3, I14_health_4, I14_health_5, I14_health_6, I14_health_7, I14_health_8, I14_health_9, I14_health_10, I14_health_96, I14_health_98, i14_health_99 and i4_health_other.
        - Drop i5_health_1, i5_health_2, i5_health_3, i5_health_4, i5_health_5, as there are highly correlated with each other and with I5_health_99 and we chose to keep I5_health_99 because it provides information on I5_heath_1 to 5.
        - Impute I5_health_99, i11_health, d1_health_98, d1_health_99 using mode (), as the number of missing values is not large.
        - Drop d1_health_1, d1_health_2 to d1_health_13 as they are correlated with d1_health_98 and d1_health_99. Again, we chose d1_health_98 and d1_health_99 because they provide information on d1_health_1 to 13.
        - Re-assign a value of zero to the missing data for I9_health and I10_health
    - Create a data frame for KNN from the cleaned data frame and drop other irrelevant variables for KNN. These include Index, RecordNo, endtime, qweek and region_state. For qweek, the week number is only for recording, and it is fundamentally different between first half data and second half (Canada and USA), so the value itself is meaningless to our analysis.
- Data Types Table

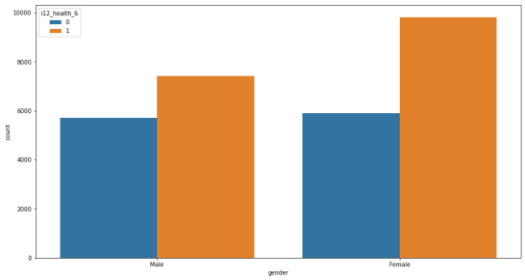
#	Column	Dtype
0	Index	object
1	RecordNo	object
2	endtime	object
3	qweek	int32
4	i1_health	float64
5	i2_health	float64
6	i7a_health	float64
7	i3_health	category
8	i4_health	category
9	i5_health_1	category
10	i5_health_2	category
11	i5_health_3	category
12	i5_health_4	category
13	i5_health_5	category
14	i5_health_99	category
15	i5a_health	category
16	i6_health	category
17	i7b_health	category
18	i8_health	category
19	i9_health	category
20	i10_health	category
21	i11_health	category
22	i12_health_1	category
23	i12_health_2	category
24	i12_health_3	category
25	i12_health_4	category
26	i12_health_5	category
27	i12_health_7	category
28	i12_health_8	category
29	i12_health_9	category
30	i12_health_10	category
31	i12_health_11	category
32	i12_health_12	category
33	i12_health_13	category
34	i12_health_14	category
35	i12_health_15	category
36	i12_health_16	category
37	i12_health_17	category
38	i12_health_18	category
39	i12_health_19	category
40	i12_health_20	category
41	i13_health	float64
42	i14_health_1	category
43	i14_health_2	category
44	i14_health_3	category
45	i14_health_4	category
46	i14_health_5	category
47	i14_health_6	category
48	i14_health_7	category
49	i14_health_8	category
50	i14_health_9	category
51	i14_health_10	category
52	i14_health_96	category
53	i14_health_98	category
54	i14_health_99	object
55	i14_health_other	category
56	d1_health_1	category
57	d1_health_2	category
58	d1_health_3	category
59	d1_health_4	category
60	d1_health_5	category
61	d1_health_6	category
62	d1_health_7	category
63	d1_health_8	category
64	d1_health_9	category
65	d1_health_10	category
66	d1_health_11	category
67	d1_health_12	category
68	d1_health_13	category
69	d1_health_98	category
70	d1_health_99	category
71	weight	float64
72	gender	category
73	age	float64
74	region_state	object
75	household_size	category
76	household_children	category
77	employment_status	category
78	i12_health_6	category

Exploratory Data Analysis

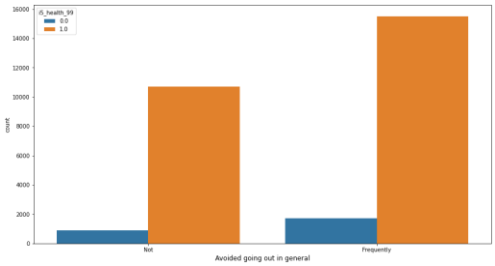
The following plots demonstrate the gender and age information based on whether having some possible COVID symptoms for different levels on avoiding going out in general. It can be seen that older people are more likely to have COVID-related symptoms (such as dry cough, fever, loss of sense of smell and taste, shortness of breath, or difficulty breathing) even though they have frequently avoided going out in general.



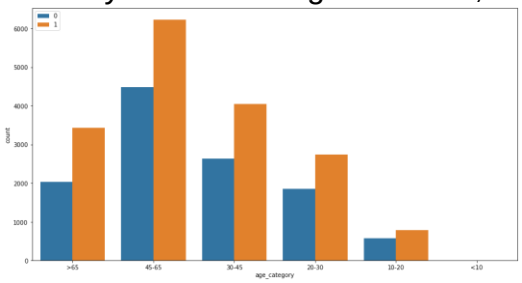
It appears that females are staying at home more frequently than males.



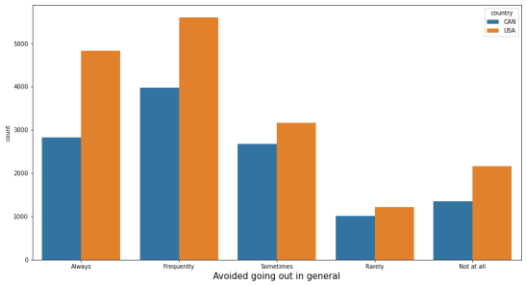
From the plot below, more people, who actually avoided going out in general frequently, have COVID-related symptoms. This is interesting as it suggests that people may have contact with the virus at home as well. It is also possible that people with more COVID-related symptoms decided to stay home more to protect others.



The plot below reveals a problem with the dataset we have. There are no observations in the age category below 10, and nearly half of the observations are in the age from 45 to 65. So, it is not a representative dataset. Thus, we should handle age information carefully when building the model, and discuss the age when interpreting results.



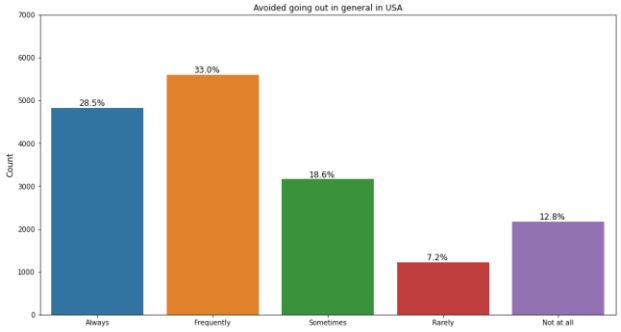
From the plot and crosstab analysis, the dataset has more observations in the USA than in Canada.

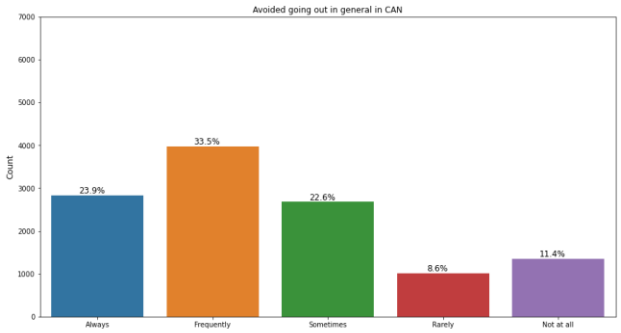


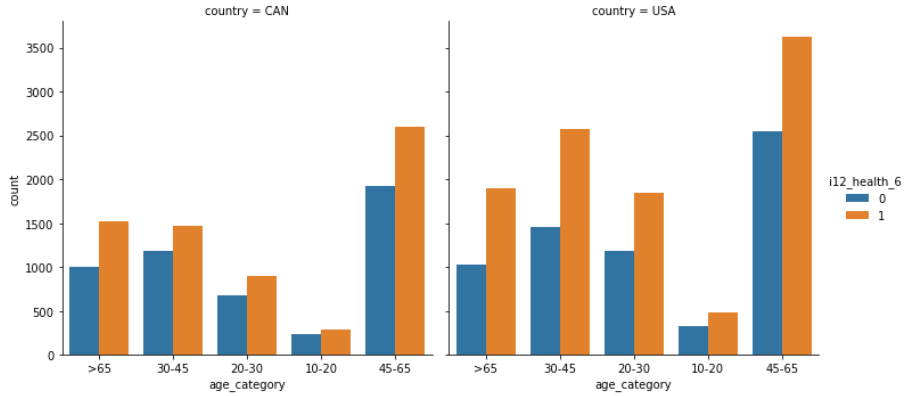
country	CAN	USA	All
i12_health_6_raw			
Always	2831	4832	7663
Frequently	3970	5604	9574
Not at all	1346	2167	3513
Rarely	1018	1219	2237
Sometimes	2678	3160	5838
All	11843	16982	28825

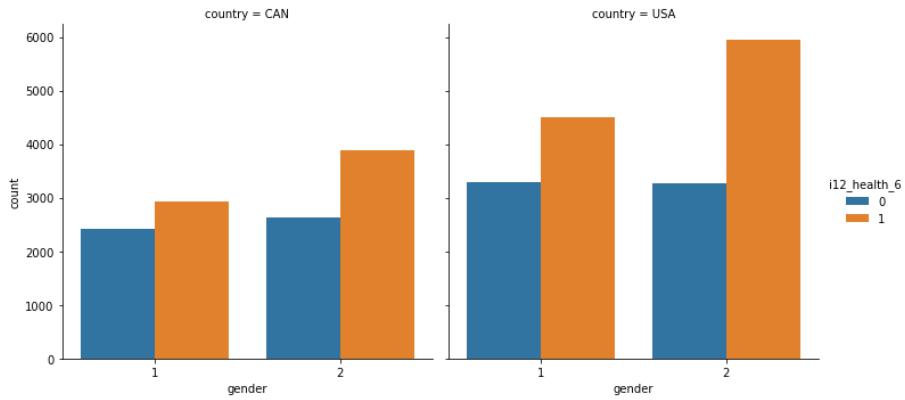
Since Canada and the USA have different medical procedures and government regulations regarding COVID, we did some analysis based on the two countries

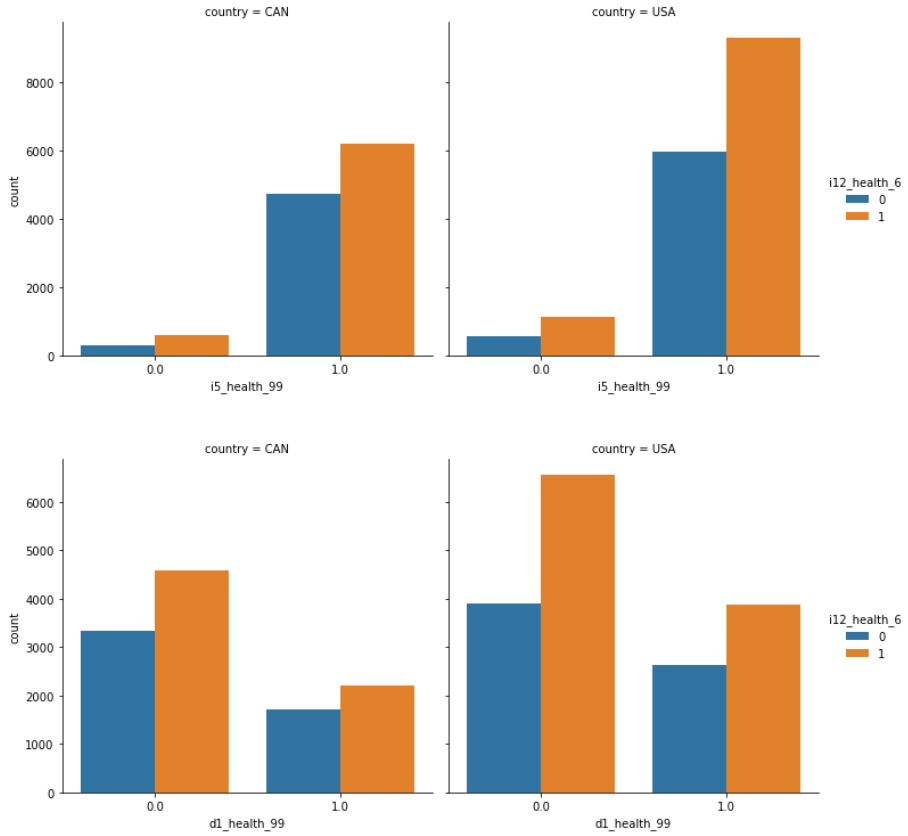
separately. But surprisingly, there are not a lot of differences between the two countries from the plots.











K-Nearest Neighbor Model (k-NN)

We are trying to develop a model using a K-NN **Classifier**.

Distance Metric

The distance metric is the important hyper-parameter through which we measure the distance between feature values and new test data inputs. Here, we use the Euclidean approach to calculate the distance between test samples and trained values. Euclidean distance is one of the most used distance metrics. It is calculated using the Minkowski Distance formula by setting ***p's*** value to ***2***. This will update the distance ***'d'*** formula.

$$d(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

We measure the distance along a straight line from point (x1, y1) to point (x2, y2).

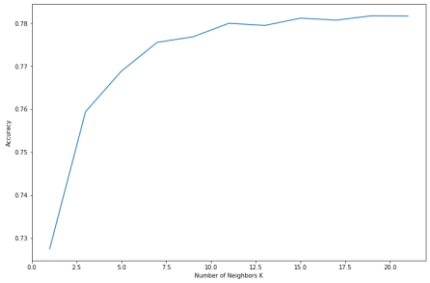
Choice of K

Initially, a random value is initialized to K and computed. Since choosing a small value of K leads to unstable decision boundaries. So, substantial K value is better for classification as it smoothens the decision boundaries. For the purpose of efficient parameter tuning, we use GridSearchCV. It allows us to define a grid of parameters that will be searched using K-fold cross-validation. This is similar to an automated version of

the "for loop". Here, our Grid object has done 10-fold cross validation on a KNN model using classification accuracy as the evaluation metric. Grid_result.best_score_ and grid_result.best_params_ provide us the best score and 'k' values achieved by the specified range in the parameter grid. We found 19 as the optimal value for k.

Model Accuracy Evaluation

In order to evaluate the developed classifier model, accuracy was used as the metric. In the figure below a plot displaying the predictive accuracy level of the model with respect to the number of nearest neighbors (i.e. k). As seen, the highest accuracy is when k = 19. Even though the expectation is that when k is increased the accuracy should keep increasing, however after k = 19, the model starts to suffer from overfitting and ignoring the predictor information which results in a reduction in test_set accuracy.



Model Accuracy vs. Number of Neighbors

Classification Matrix

We created two target variables for 12_health6, one where the variable is multiclass and the other binary. The first variable answers how often people choose to not go out, while the other determines if people go out or not. These two target variables led to widely different accuracy rates, where the one with more classes has around a 52% accuracy rate vs 79% for the binary variable. This is because it is much easier for the model to predict when there are only two choices and more data to learn from. The distribution of yes or no is more even than the multiclass target variable as well. Additionally, it is more useful to understand if people go outside or not, than to have people determine themselves how often they go out. This is because people have different definitions for going out frequently or rarely.

After investigating both models, we split the data into training and test sets and printed the classification and confusion matrix report for both the training and the test set. This allows us to make sure overfitting has not occurred with cross-validation. K Fold validation was also used to make sure the training data was shuffled 10 times to reduce bias.

After conducting the GridSearchCV function in python to determine the optimal model in terms of what parameters to select, we reran the KNN Classification model again using those parameters for both target variables. The results are shown below.

Binary Target Variable Results

```
predict on test dataset
[[1777 1106]
 [ 437 3887]]
predict on trainig dataset
[[ 5500  3205]
 [ 1192 11721]]
```

report on test dataset

	precision	recall	f1-score	support
0	0.80	0.62	0.70	2883
1	0.78	0.90	0.83	4324
accuracy			0.79	7207
macro avg	0.79	0.76	0.77	7207
weighted avg	0.79	0.79	0.78	7207

report on training dataset

	precision	recall	f1-score	support
0	0.82	0.63	0.71	8705
1	0.79	0.91	0.84	12913
accuracy			0.80	21618
macro avg	0.80	0.77	0.78	21618
weighted avg	0.80	0.80	0.79	21618

Multi-Class Target Variable Results

```
predict on test dataset
[[1267  602   72    3   14]
 [ 602 1462  267    7   28]
 [ 159  709  475   28   64]
 [  36  186  226   26   81]
 [  21  130  206   37  499]]
predict on trainig dataset
[[3840 1558  253   10   44]
 [1630 4887  594   18   79]
 [ 441 2087 1669   55  151]
 [  97  584  628  134  239]
 [  58  374  521  107 1560]]
```

report on test dataset

	precision	recall	f1-score	support
1	0.61	0.65	0.63	1958
2	0.47	0.62	0.54	2366
3	0.38	0.33	0.35	1435
4	0.26	0.05	0.08	555
5	0.73	0.56	0.63	893
accuracy			0.52	7207
macro avg	0.49	0.44	0.45	7207
weighted avg	0.51	0.52	0.50	7207

report on training dataset

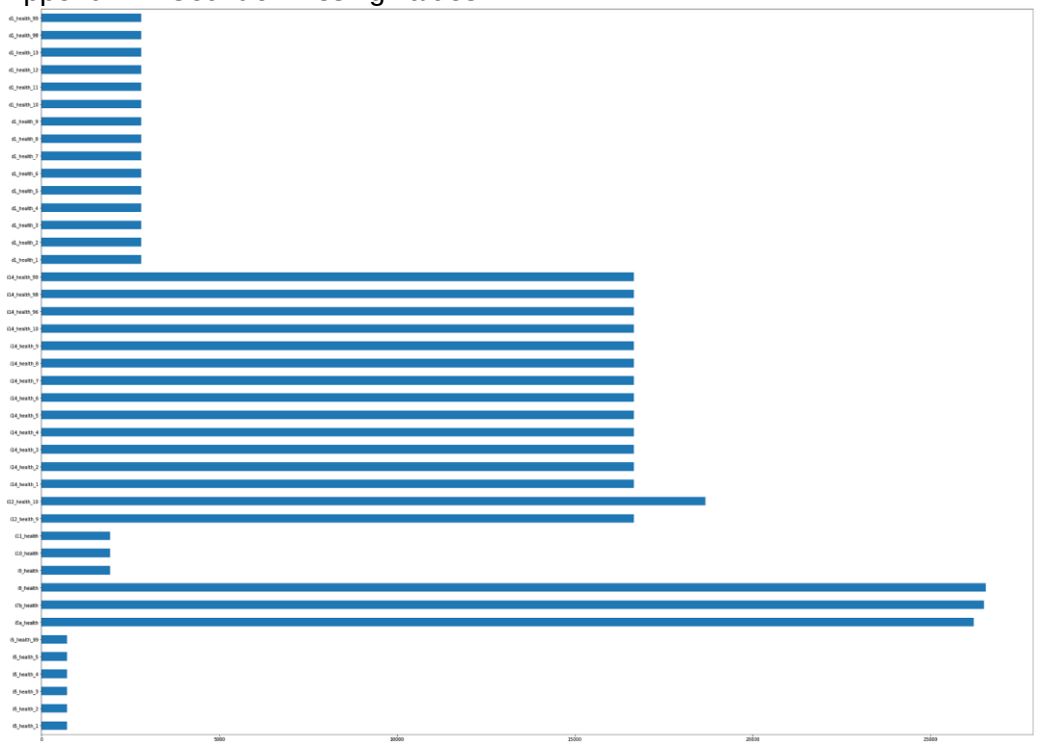
	precision	recall	f1-score	support
1	0.63	0.67	0.65	5705
2	0.51	0.68	0.59	7208
3	0.46	0.38	0.41	4403
4	0.41	0.08	0.13	1682
5	0.75	0.60	0.66	2620
accuracy			0.56	21618
macro avg	0.55	0.48	0.49	21618
weighted avg	0.55	0.56	0.54	21618

The classification report and confusion matrix were created using the classification report and confusion matrix functions from scikit learn.

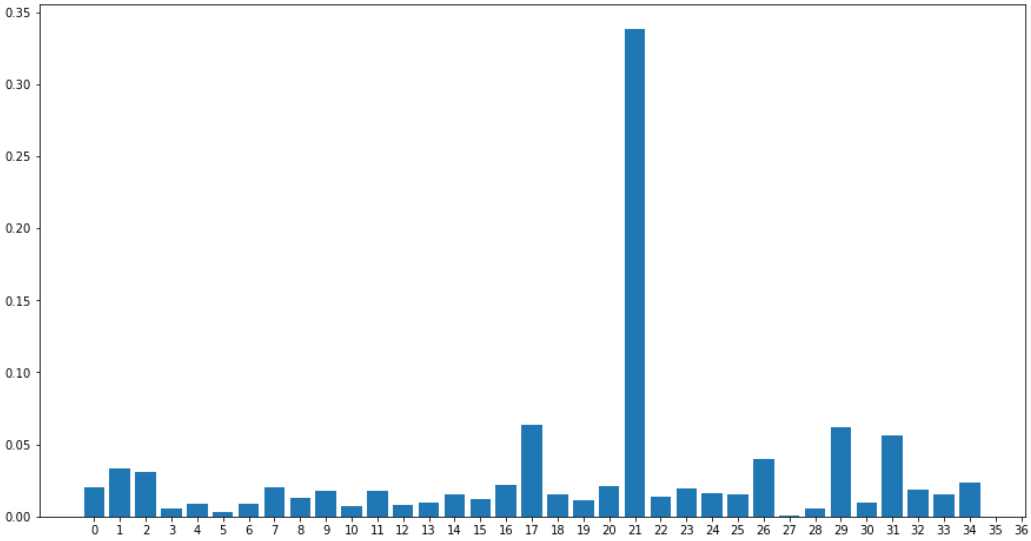
As you can see for the target variable with two classes, the training set had an accuracy of 80% and the test set 79%. As the accuracy does not seem to be significantly different, we can assume that the model is not overfit.

Appendix:

Appendix 1: Count of missing values



Appendix 2: Importance of features selected



References

Galarnyk, M. (2019). Understanding Decision Trees for Classification (Python). Retrieved 13 December 2020, from https://towardsdatascience.com/understanding-decision-trees-for-classification-python-9663d683c952

Larose, D., & Larose, C. (2015). *Data mining and predictive analytics* (2nd ed.). John Wiley & Sons, Inc.

Final Page

Grade: _____