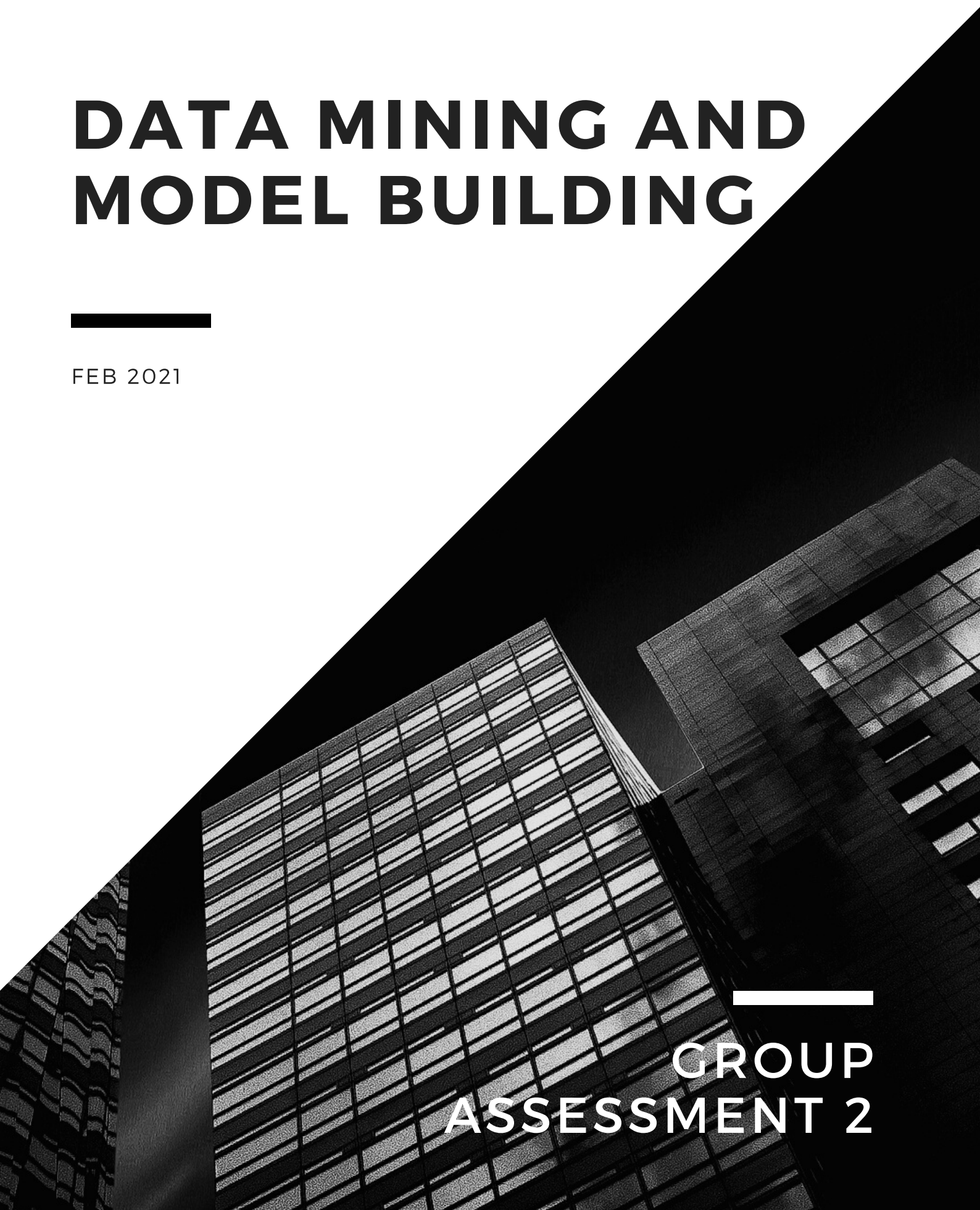
DATA MINING AND MODEL BUILDING

FEB 2021



GROUP ASSESSMENT 2

Data Mining for Improved Healthcare

Project (a): Association mining to find hotspots based on a Patient Route Data

Question 1
What pre-processing was required on the dataset before building the association mining model? What variables did you include in the analysis? Justify your choice.

Convert date as datetime format. Drop global_num as it duplicates with patient_id. Have also drop latitude and longtitude as location is a better representation for the analysis for the routes by name.

Question 2
Conduct association mining and answer the following:

a. What 'min_support' and `min_confidence' thresholds were set for this mining exercise? Rationale why these values were chosen?

Locations meeting min_support are considered as frequent. To filter out frequent locations, have set min_support = 0.01 (minimum of 1% probability of both right side and left side occurring) and letting min_confidence be free. Reasonable number of rules (8 rules) with Lift > 1 (positive correlation) are found with confidence level between 8% - 50%. Unable to find at least 5 rules with lift > 1 if min_support is set above 0.01.

Lift > 1 are the most interesting rules since it occurs the most. The higher the lift, the more interesting the rule. The highest Lift is 5.458061.

b. Report the top-5 rules and interpret them.

No.	Left side	Right side	Support	Confidence	Lift
1	Jorapokhar_Jharkhand	Channapatna_Karnataka	0.012075	0.305556	5.458061
2	Channapatna_Karnataka	Jorapokhar_Jharkhand	0.012075	0.215686	5.458061
3	Gokak_Karnataka	Sardarshahar_Rajasthan	0.027442	0.500000	3.399254
4	Sardarshahar_Rajasthan	Gokak_Karnataka	0.027442	0.186567	3.399254
5	Lucknow_Uttar Pradesh	Sardarshahar_Rajasthan	0.018661	0.215190	1.462970

Rule 1: Highest lift of 5.46, patient from Jorapokhar is 5.46 times more likely to travel to Channapatna than a patient chosen at random. 30.56% confident that patient in Jorapokhar have also travelled to Channapatna and 1.21% probability patient travelled to both towns out of the dataset.

Rule 2: Highest lift of 5.46, patient from Channapatna is 5.46 times more likely to travel to Jorapokhar than a patient chosen at random. 21.57% confident that patient in Channapatna have also travelled to Jorapokhar and 1.21% probability patient travelled to both towns out of the dataset.

Rule 3: Lift of 3.4, patient from Gokak is 3.4 times more likely to travel to Sardarshahar than a patient chosen at random. 50% confident that patient in Gokak have also travelled to Sardarshahar and 2.74% probability patient travelled to both towns out of the dataset.

Rule 4: Lift of 3.4, patient from Sardarshahar is 3.4 times more likely to travel to Gokak than a patient chosen at random. 18.66% confident that patient in Sardarshahar have also travelled to Gokak and 2.74% probability patient travelled to both towns out of the dataset.

Rule 5: Lift of 1.46, patient from Lucknow is 1.46 times more likely to travel to Sardarshahar than a patient chosen at random. 21.52% confident that patient in Lucknow have also travelled to Sardarshahar and 1.87% probability patient travelled to both towns out of the dataset.

Question 3
Identify top-5 common routes that COVID-19 positive patients from the town Ranebennur in Karnataka state have travelled.

Min_support = 0.004 (minimum of 0.4% probability of both Ranebennur _Karnataka (left side) and the right side occurring). There is at least 0.4% probability patients from Ranebennur have also travelled to Barh, Alipurduar, Sirohi, Ratnagiri or Gokak.

	Left_side	Right_side	Support	Confidence	Lift
89	Ranebennur_Karnataka	Barh_Bihar	0.004391	0.050633	4.193326
86	Ranebennur_Karnataka	Alipurduar_West Bengal	0.004391	0.050633	1.921941
193	Ranebennur_Karnataka	Sirohi_Rajasthan	0.004391	0.050633	1.647378
187	Ranebennur_Karnataka	Ratnagiri_Maharashtra	0.007684	0.088608	1.416167
125	Ranebennur_Karnataka	Gokak_Karnataka	0.004391	0.050633	0.922532

Question 4
Can you perform sequence analysis on this dataset? If yes, present your results. If not, rationalize why.
Yes, it is possible to perform sequence analysis as date is available.

```
1 get_association_rules(sequences, 0.001, 0.1)
```

	Left_rule	Right_rule	Support	Confidence
0	[Chittorgarh_Rajasthan]	[Ratnagiri_Maharashtra]	0.002195	0.200000
1	[Chittorgarh_Rajasthan]	[Ratnagiri_Maharashtra, Gola Gokrannath Kheri_...	0.001098	0.100000
2	[Chittorgarh_Rajasthan]	[Delhi_Delhi]	0.001098	0.100000
3	[Chittorgarh_Rajasthan, Channapatna_Karnataka]	[Delhi_Delhi]	0.001098	0.500000
4	[Chittorgarh_Rajasthan]	[Sagar_Karnataka]	0.001098	0.100000
5	[Chittorgarh_Rajasthan, Belgaum_Karnataka]	[Sagar_Karnataka]	0.001098	1.000000
6	[Chittorgarh_Rajasthan, Belgaum_Karnataka, Cha...	[Sagar_Karnataka]	0.001098	1.000000
7	[Chittorgarh_Rajasthan, Chatrapur_Odisha]	[Sagar_Karnataka]	0.001098	1.000000
8	[Chittorgarh_Rajasthan]	[Sagar_Karnataka, Suri_West Bengal]	0.001098	0.100000
9	[Chittorgarh_Rajasthan, Belgaum_Karnataka]	[Sagar_Karnataka, Suri_West Bengal]	0.001098	1.000000
10	[Chittorgarh_Rajasthan, Belgaum_Karnataka, Cha...	[Sagar_Karnataka, Suri_West Bengal]	0.001098	1.000000

Question 5
How can the outcome of this study be used by the relevant decision-makers?
Can understand the pattern of travel and to could put in place travel limits to contain the virus within the detected locations and predict possible future locations that might be infected based on the travel routes.

Project (b): Clustering Diabetes data

Question 1
What pre-processing was required on the dataset (D2.csv) before building the clustering model?

Gender: Map: 'Female':0, 'Male':1, 'Unknown/Invalid':np.NaN

Boolean type: change and diabetesMed – although they are numeric in nature in Python where False is assigned as 0 and True is assigned as 1 and mapping is not strictly necessary. But for clarity, have also done the mapping.
change: Replace: False: 0, True: 1
diabetesMed: Replace: False: 0, True: 1

Drop row
race: 848 missing values (2.34%)
age: 9 missing values
chlorpropamide: 9 missing values
gender: 3 NaN

Drop column

medical_specialty - 32203 records are value 'invalid' - (about 62% of total data) – not useful for analysis
admission_source_id - most are admitted under 1: Emergency duplicated with admission_type_id (also majority is Emergency(7)
max_glu_serum (most value is "none" i.e. test not taken)
A1Cresult (most value is "none" i.e. not measured)

Drop diabetes medicines

As majority of the below diabetes medicines are not prescribed plus it already has a variable on diabetesMed and change have captured information about diabetes medication. Hence, the following are drop due to redundancies and to reduce dimensionality.

- **metformin**
- **repaglinide**
- **nateglinide**
- **chlorpropamide**
- **glimepiride**
- **acetohexamide**
- **glipizide**
- **glyburide**
- **tolbutamide**
- **insulin**

Categorical variables: discretised the categorical variables.

age: mapped = {'[90-100)':1, '[80-90)': 2, '[70-80)': 3, '[60-70)': 4, '[50-60)': 5,
'[40-50)': 6, '[30-40)': 7, '[20-30)': 8, '[10-20)': 9, '[0-10)': 10}

race: mapped = {'Caucasian':1, 'AfricanAmerican': 2, 'Hispanic': 3, 'Asian': 4, 'Other': 5}

Question 2

Build a clustering model to profile the characteristics of diabetic patients. Answer the followings:

a. What clustering algorithm have you used?
Partitioning approach via kprototype method as the data has both categorical and numerical values.

b. List the attributes used in this analysis.

race
gender
age
time_in_hospital
number_emergency
diabetesMed

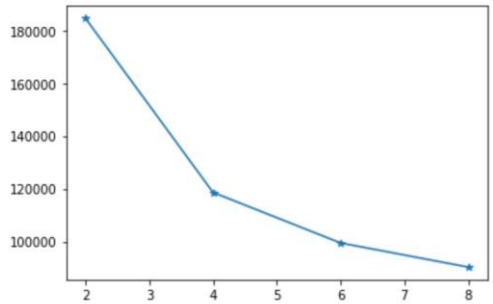
Trial and errors with different combinations of features and k to compare performance levels. Below table captures few selected better performing example for illustrations:

S: Scaled data, #: Number of features

S	Features	#	K	Silhouette	Remarks
	Overall 35317 records				
N	Race, gender, age, time_in_hospital, number_emergency, diabetesMed	6	4	0.21015647482744995	
Y	Race, gender, age, time_in_hospital, number_emergency, diabetesMed	6	4	0.30124843080511493	Scaled better
Y	Race, age, time_in_hospital, number_emergency, diabetesMed	5	4	0.29341957574098493	Dropped gender
Y	Race, age, time_in_hospital, number_emergency	4	4	0.26238896189768646	Dropped diabetesMed
Y	Race, age, time_in_hospital, number_emergency, number_inpatient, diabetesMed	6	4	0.24549219438558126	Add inpatient and diabetesMed
	Race Specific – Caucasian (27766 records)				Lower with race specific
Y	Gender, age, time_in_hospital, number_emergency, diabetesMed	5	4	0.2897633435136228	Lower with race
Y	age, time_in_hospital, number_emergency, diabetesMed	4	4	0.2867942058665231	Dropped gender, slightly worse
	Race Specific – Asian (293 records)				
Y	Gender, age, time_in_hospital, number_emergency, diabetesMed	5	4	0.2622282951137327	Lower Caucasian
Y	age, time_in_hospital, number_emergency, diabetesMed	4	4	0.2622983843622477	Dropped gender, slightly better

c. What is the optimal number of clusters identified? How did you reach this optimal number?

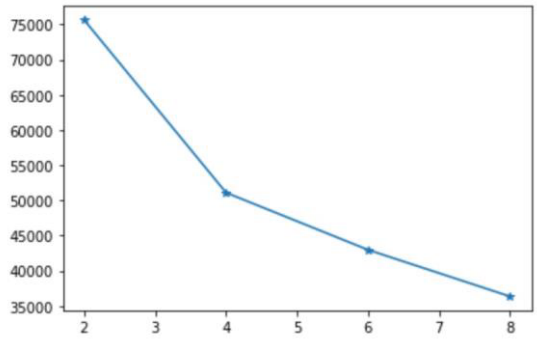
Identified the optimal of 4 clusters. Using both elbow method and highest Silhouette score of 0.21015647482744995 without normalisation.




```
The avg Silhouette score for k=4: 0.21015647482744995
The avg Silhouette score for k=6: 0.13377962427045706
The avg Silhouette score for k=8: 0.029875739127108274
```

d. Did you normalise the variables? What was its effect on the model – Does the variable normalization process enable a better clustering solution?

As data are in different scales, variables are normalised as Euclidean distance (for numeric variables) and Hamming distance (for categorical variables) requires normalisations for comparisons. Using both elbow method (with lower y-values after normalization) and highest Silhouette score of 0.30124843080511493 (an improvement of 43%)




```
The avg Silhouette score for k=4: 0.30124843080511493
The avg Silhouette score for k=6: 0.10184572283778222
The avg Silhouette score for k=8: 0.11416481344802408
```

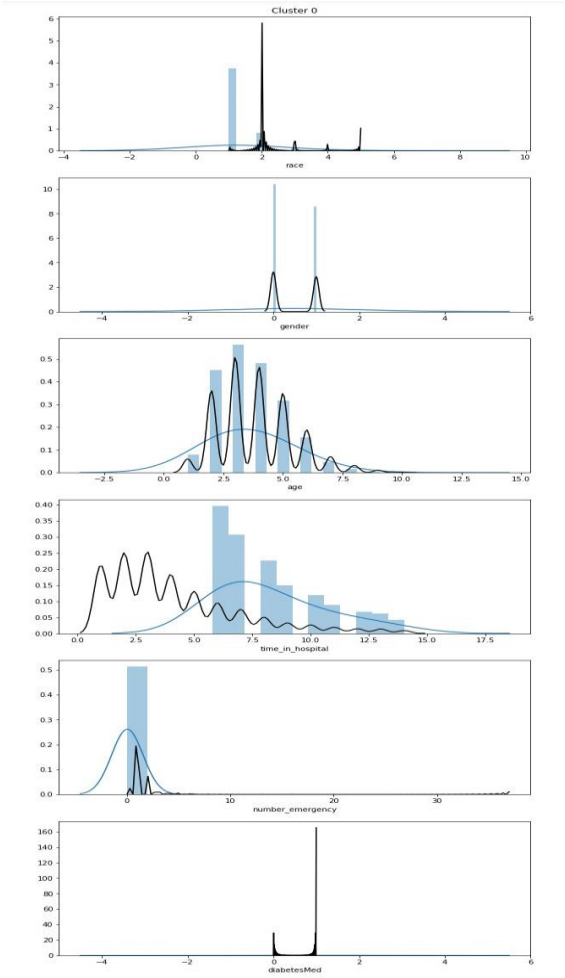
Question 3

For the model with the optimal number of clusters:

a. Visualize the clusters using 'pairplot' and interpret the visualization.

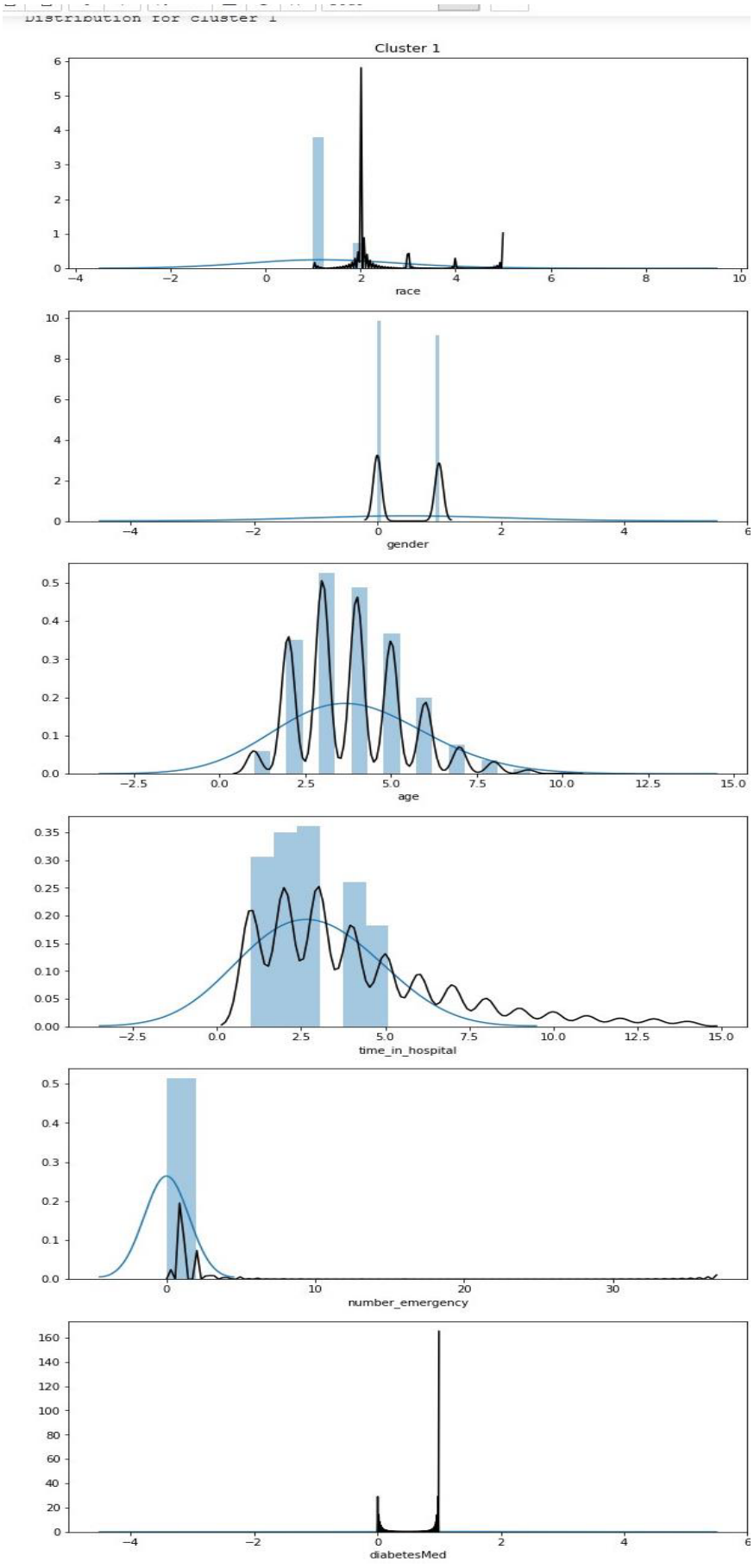
Cluster 0

Race is focused on Caucasian. About the same distribution in both gender the cluster with slighter more in Female. Majority of age group is in 60-70. Right leaning in time in hospital with long tail. Low number of emergency. Flat in diabetesMed.



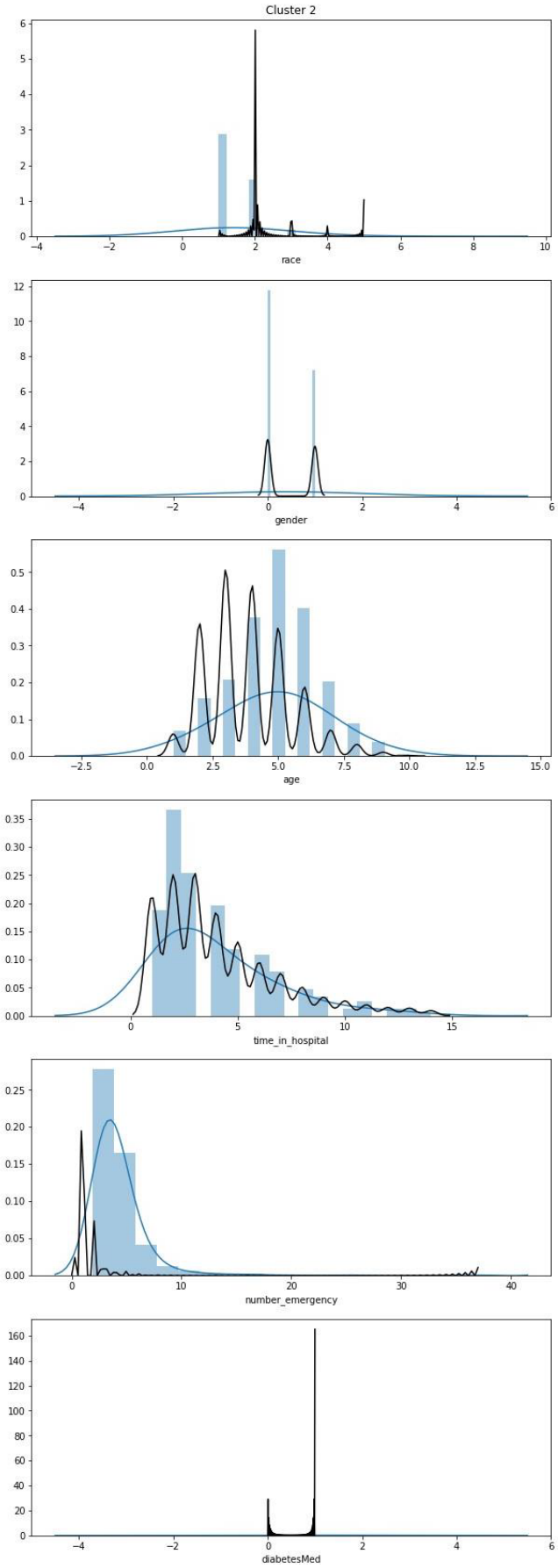
Cluster 1

Race is mainly sitting in Caucasian in this cluster. About the same distribution in both gender the cluster with slighter more in Female. Majority of age group is in 60-70. Similar distribution as in the general distribution in time in hospital. Low number of emergency. Flat in diabetesMed.



Cluster 2

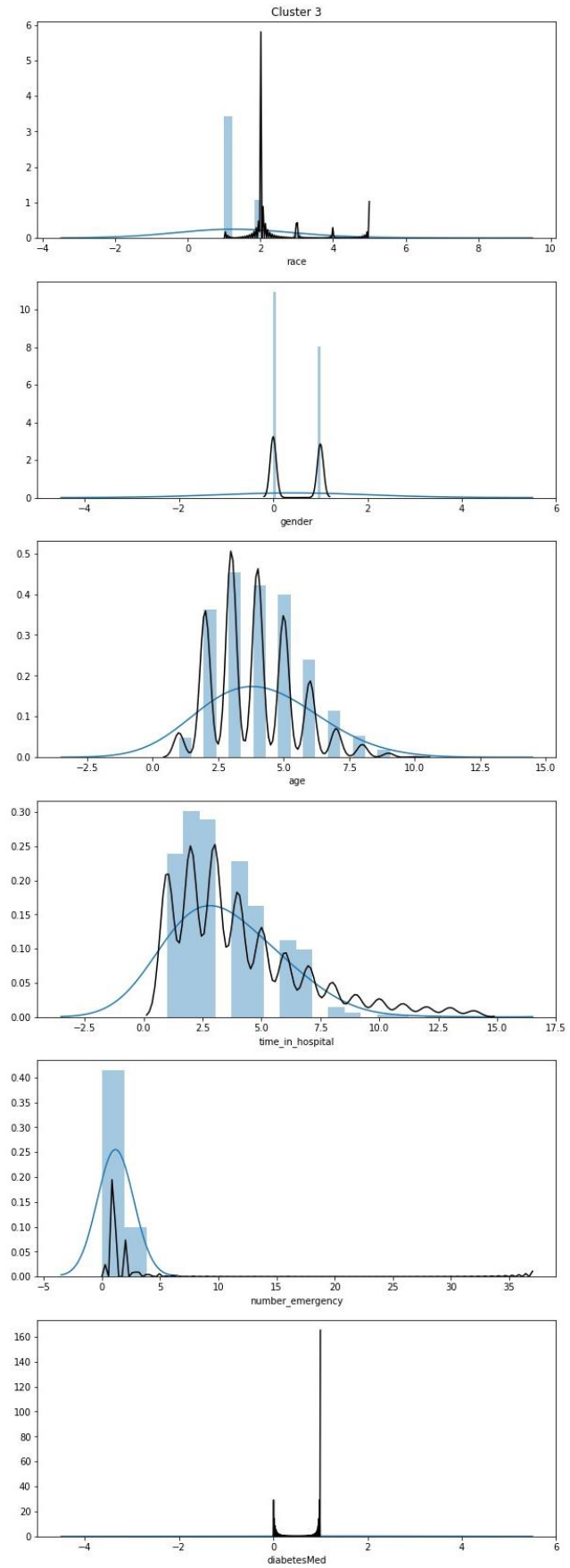
Race is mainly sitting in Caucasian in this cluster. Almost 1/3 more female than male. Mainly spread across 70-40 years old. Slightly left leaning in time in hospital with long tail. Slightly leaning right in the number of emergency. Flat in diabetesMed.



Cluster 3

Race is mainly sitting in Caucasian in this cluster. About the same distribution in both gender the cluster with more in Female. Wider spread in age across 90-40 years old. Higher in time in hospital. Similar distribution in the number of emergency. Flat in diabetesMed.

```
Distribution for cluster 3
```



b. Characterize the nature of each cluster by giving it a descriptive label and a brief description. Hint: use cluster distribution.

Cluster 0: Higher time_in_hospital and lower number_emergency

Cluster 1: Older and younger age groups with lower time_in_hospital

Cluster 2: Higher number_emergency across age groups and races.

Cluster 3: Lower number_emergency and time_in_hospital across age groups and races.

```
Cluster membership
1    23905
0     8159
3     2918
2      335
Name: Cluster_ID, dtype: int64
```

Question 4

Build another clustering model using an algorithm that helps to profile the patients of specific races including Asian and Caucasian. Use the best setting (e.g., variable normalisations, optimal K, etc) obtained in the previous model. Answer the followings:

a. List the attributes used in this analysis.

As data is specific to race, hence the race variable can be drop. Model for Caucasian (27766 records) perform better than Asian (293 records) with Silhouette score of 0.2897633435136228 and 0.2622282951137327 respectively. Silhouette score for the previous full model has higher Silhouette score of 0.30124843080511493 and seem to have reached the global minimum.

gender (have tried dropping gender, but performance slightly worsen for Caucasian group but slightly improved for Asian group)
age
time_in_hospital
number_emergency
diabetesMed

b. What difference do you see in this clustering interpretation when compared to the previous one?

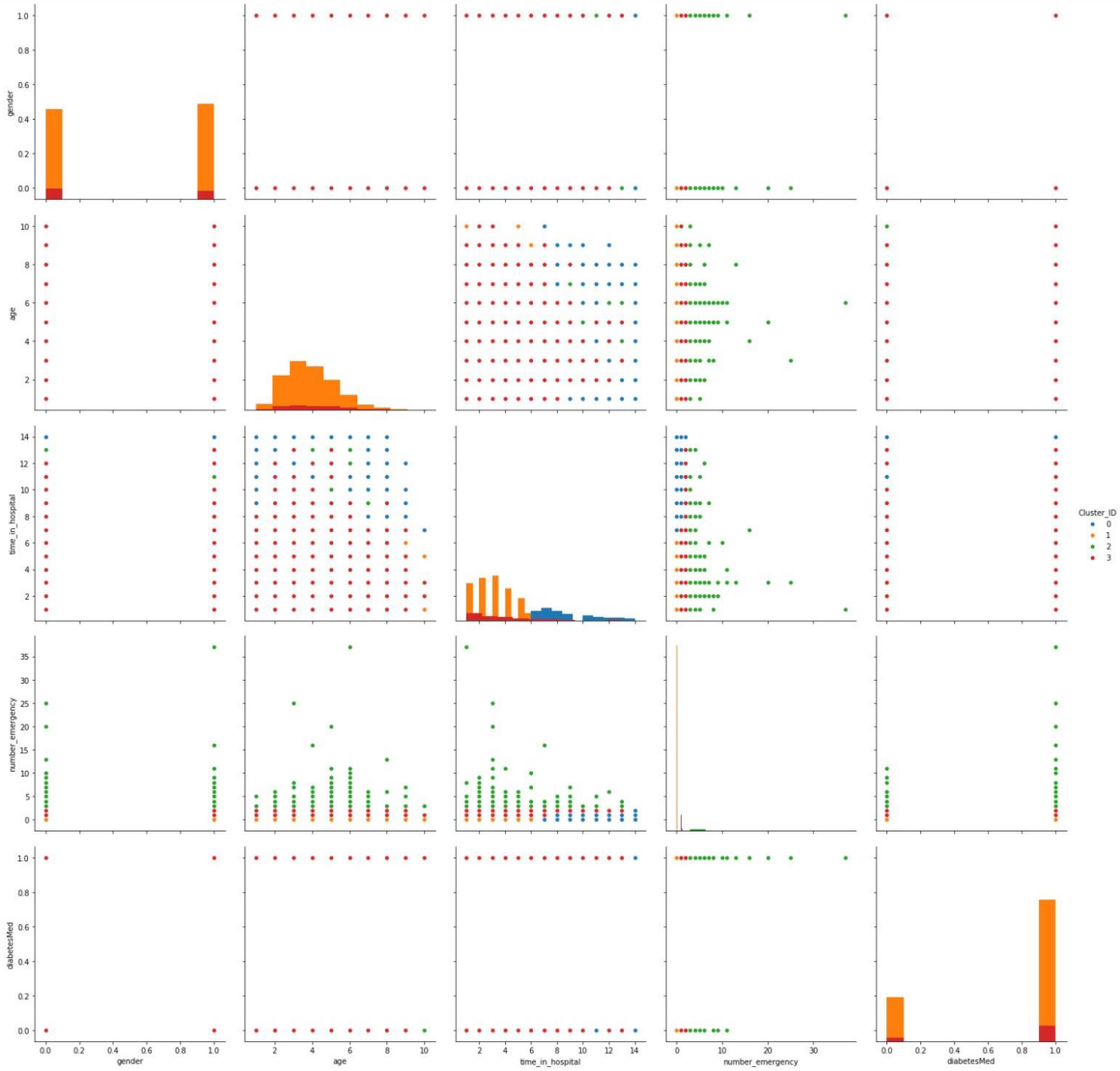
Caucasians:

Cluster 0: Higher time_in_hospital and lowest number_emergency

Cluster 1: Younger age group with shorter time_in_hospital

Cluster 2: Higher number_emergency across age groups and races.

Cluster 3: Lower/medium number_emergency across age groups and races.



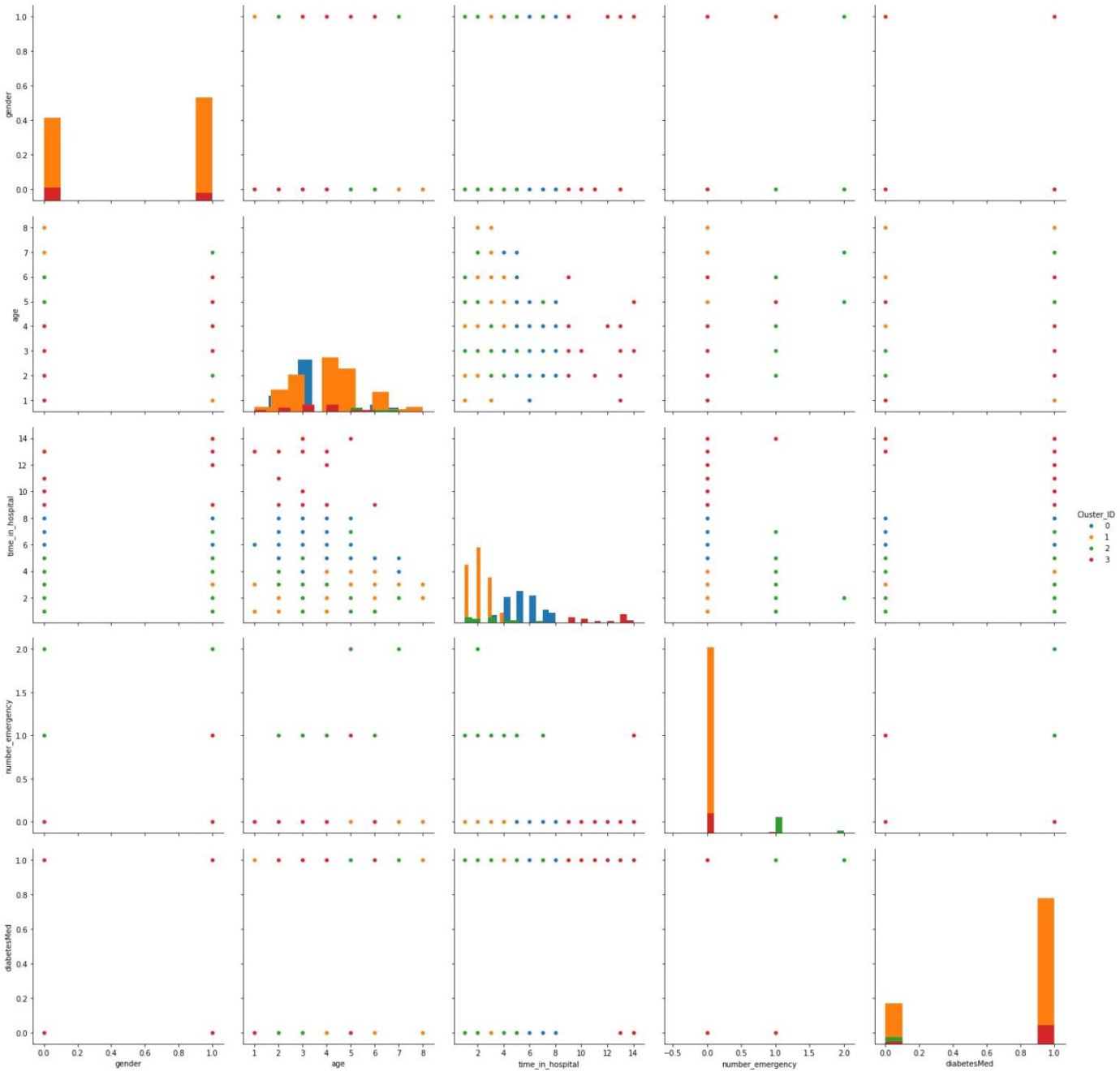
Asians:

Cluster 0: Medium time_in_hospital

Cluster 1: Shorter time_in_hospital

Cluster 2: Lower time in time_in_hospital and medium/higher number_emergency

Cluster 3: Highest time_in_hospital and lowest number_emergency



Caucasians and Asians have different clusters classifications and is more targeted to the race.

Question 5

How can the outcome of this study be used by the relevant decision-makers?

The patients can be grouped according to the clusters characteristics in order to pay extra attention to these groups with higher risks of readmission.

Project (c): Building and Evaluating Predictive models

Before further analysis and pre-processing, have checked if the target class are balanced. Confirmed they are balance where roughly of equal size. Not readmitted with 27277 records and Admitted with 23453 records.

Predictive modelling using Decision Tree

Question 1
What pre-processing was required on the dataset (D3.csv) before decision tree modelling? What distribution split between training and test datasets have you used?

Gender: Map: 'Female':0, 'Male':1, 'Unknown/Invalid':np.NaN

Boolean type: Change and diabetesMed – although they are numeric in nature in Python where False is assigned as 0 and True is assigned as 1 and mapping is not strictly necessary. But for clarity, have also done the mapping.

change: Replace: False: 0, True: 1

diabetesMed: Replace: False: 0, True: 1

Drop row

- **race**: 1016 missing values (1.96%)
- **age**: 10 missing values
- **chlorpropamide**: 9 missing values
- **gender**: 4 NaN

Drop variables

- **medical_specialty** - 31200 (62%) records are "Invalid"
- **acetohexamide** - unary (only 'no')
- **tolbutamide** - - urary (only 'no')
- **max_glu_serum** - majority are none
- **A1Cresult** - majority are none

Drop diabetes medicines
Most are not prescribed. In connection, **diabetesMed** and **change** are already included as variables to capture if medicine is taken.

- **metformin**
- **repaglinide**
- **nateglinide**
- **chlorpropamide**
- **glimepiride**
- **glipizide**
- **glyburide**
- **insulin**

Total 50730 records after data are cleaned. One hot encoding is applied to the pre-processed data frame.

Target: **readmitted** (readmitted = 1, not readmitted = 0)

Have split the training data to 70% and test data to 30%.

Question 2
Build a decision tree using the default setting. Answer the followings:

a. What is the classification accuracy of training and test datasets?

Train accuracy: 0.9997183971163864
Test accuracy: 0.5595637032656547

Very good train result but poorly in test ➔ indication of overfitting ➔ need to consider how to optimise the decision tree, perhaps need to do tree pruning.

b. What is the size of the tree (number of nodes and rules)?
21359 nodes and 10680 rules

c. Which variable is used for the first split?
number_inpatient

d. What are the 5 important variables (in the order) in building the tree?

```
num_lab_procedures : 0.20933721729538135
num_medications : 0.159751455402881
time_in_hospital : 0.09892290740379038
number_inpatient : 0.0689223885219774
discharge_disposition_id : 0.06739319016146295
```

e. What parameters have been used in building the tree? Detail them.
Have used the default hyperparameters:

criterion – method to evaluate the quality of the split. Have use gini for this model

max_depth – the maximum depth of the tree. The higher the depth the more complex the model and more tree nodes. This model has not set a max limit. Risk of overfitting as seen in the poor test result above.

min_samples_leaf – the minimum number of samples in a leaf node. This allows to limit the minimum size of a leaf node. This model has set the parameter to 1 (almost no limitation)

Question 3
Build another decision tree tuned with GridSearchCV. Answer the followings:

a. What is the classification accuracy of training and test datasets?
Train accuracy: 0.643941313959055
Test accuracy: 0.6354556803995006

Train accuracy is a touch better than test accuracy, perhaps model is slightly overfitted. Further investigation is required.

b. What is the size of the tree (number of nodes and rules)?
119 nodes and 60 rules

c. Which variable is used for the first split?
number_inpatient

d. What are the 5 important variables (in the order) in building the tree?

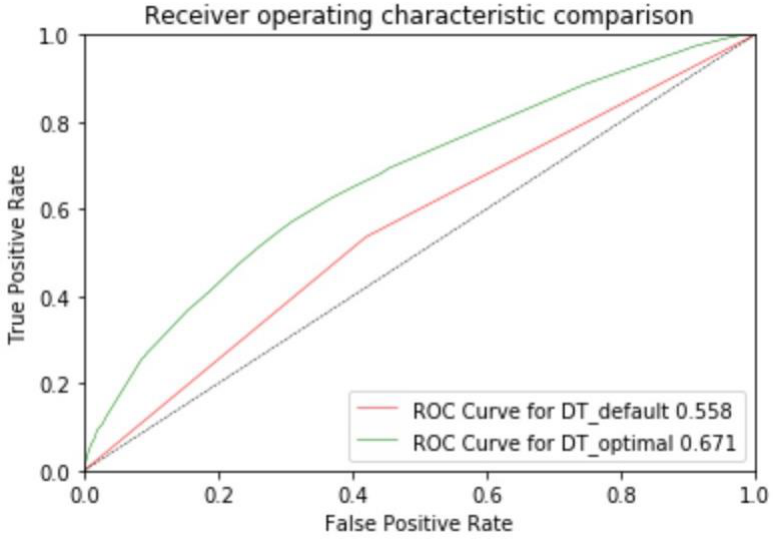
```
number_inpatient : 0.5508416972167725
discharge_disposition_id : 0.18730161959108713
number_emergency : 0.07644029985665715
number_diagnoses : 0.03945930795427858
diabetesMed : 0.036002344800024205
```

e. Report if you see any evidence of model overfitting.

There is performance improvement in this model even though the test accuracy is lower than train accuracy. As depth of tree decreases, bias has increase whilst the variance decrease. As the complexity of the model is reduced, the variance will also reduce. The variance between train and test accuracy is lowered in this model but not enough. There is still slight indication of overfitting.

Question 4

What differences do you observe between these two decision tree models (with and without fine-tuning)? How do they compare performance-wise? Produce the ROC curve for both DTs. Explain why those changes may have happened.

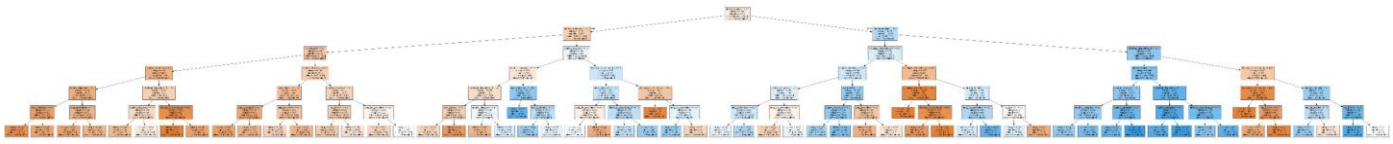


The GridSearchCV model perform better than the default model with ROC AUC score of 0.671 and 0.558 respectively. With optimal parameters of 'criterion': 'entropy', 'max_depth': 6, and 'min_samples_leaf': 15. F1 score improves as the tree depth is decrease, and the complexity of the model is reduced. Bias has increase that traded off with variance decrease.

Question 5

From the better model, can you identify which patients could potentially be "readmitted"? Can you provide general characteristics of those patients?

In general, patients have prior stayed in hospital twice or more and discharge_disposition_id > 10, that is, requiring long term medical care after discharge or transfer to hospice upon discharged are more likely to be readmitted.



Predictive modelling using Regression

Question 1

What pre-processing was required on the dataset before regression modelling? What distribution split between training and test datasets have you used?

Using the same pre-processed data from decision tree question. In addition, have scale the data for the regression model. Have split into 70% training and 30% testing.

Question 2

Build a regression model using the default regression method with all inputs. Build another regression model tuned with GridSearchCV. Now, choose a better model to answer the followings:

a. Explain why you chose that model.

GridSearchCV model with hyperparameter of C = 0.1 compare to the default model with C = 1. They both have the same train and test accuracy results and same classification report results. There is no obvious difference between the two. Will choose the default over GridSearchCV model since smaller C requires stronger regularisation.

```
# Training logistic regression model with default setting

from sklearn.linear_model import LogisticRegression

model = LogisticRegression(random_state=rs)

# fit it to training data
model.fit(X_train, y_train)
```

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=100,
                   multi_class='warn', n_jobs=None, penalty='l2',
                   random_state=77, solver='warn', tol=0.0001, verbose=0,
                   warm_start=False)
```

```
# training and test accuracy
print("Train accuracy:", model.score(X_train, y_train))
print("Test accuracy:", model.score(X_test, y_test))

# classification report on test data
y_pred = model.predict(X_test)
print(classification_report(y_test, y_pred))
```

```
Train accuracy: 0.6285939568021176
Test accuracy: 0.6282935803929299
              precision    recall  f1-score   support

           0       0.62      0.80      0.70      8183
           1       0.65      0.43      0.51      7036

    accuracy                           0.63     15219
   macro avg       0.63      0.61      0.61     15219
weighted avg       0.63      0.63      0.61     15219
```

```
# print out the optimum
print(cv.best_params_)
```

```
{'C': 0.1}
```

```
cv.fit(X_train, y_train)

print("Train accuracy:", cv.score(X_train, y_train))
print("Test accuracy:", cv.score(X_test, y_test))

# classification report on test data
y_pred = cv.predict(X_test)
print(classification_report(y_test, y_pred))
```

```
Train accuracy: 0.6285939568021176
Test accuracy: 0.6282935803929299
              precision    recall  f1-score   support

           0       0.62      0.80      0.70      8183
           1       0.65      0.43      0.51      7036

    accuracy                           0.63     15219
   macro avg       0.63      0.61      0.61     15219
weighted avg       0.63      0.63      0.61     15219
```

b. Name the regression function used.

Logistic regression is used as the target variable is categorical, readmitted or not readmitted.

c. Did you apply standardization of variables? Why would you normalise the variables for regression mining?

Yes, the variables are standardized. Input variables on different scales make comparison between data points difficult. It also adversely affects training model optimisation performance as gradient descent algorithm would put more weights on larger scale inputs to update much faster than smaller scale inputs.

d. Report the variables included in the regression model.

16 variables are used in the logistic model:

1. race
2. gender
3. age
4. admission_type_id
5. discharge_disposition_id
6. admission_source_id
7. time_in_hospital
8. num_lab_procedures
9. num_procedures
10. num_medications
11. number_outpatient
12. number_emergency
13. number_inpatient
14. number_diagnoses
15. change
16. diabetesMed

e. Report the top-5 important variables (in the order) in the model.

```
number_inpatient : 0.5103462345596862
number_emergency : 0.2766585053967742
number_outpatient : 0.1304126501309137
diabetesMed : 0.10716840866984222
number_diagnoses : 0.09966966723373262
```

f. What is the classification accuracy on training and test datasets?

Train accuracy: 0.6285939568021176
Test accuracy: 0.6282935803929299

g. Report any sign of overfitting in this model.

Train and test accuracy are almost the same. There is almost no indication of overfitting.

Question 3

Build another regression model on the reduced variables set. Perform dimensionality reduction with Recursive feature elimination. Tune the model with GridSearchCV to find the best parameter setting. Answer the followings:

a. Was dimensionality reduction useful to identify a good feature set for building the accurate model?

Yes, it has reduced from 29 to 26 features.

b. What is the classification accuracy on training and test datasets?

Train accuracy: 0.6285939568021176
Test accuracy: 0.6279650436953808

c. Report any sign of overfitting.

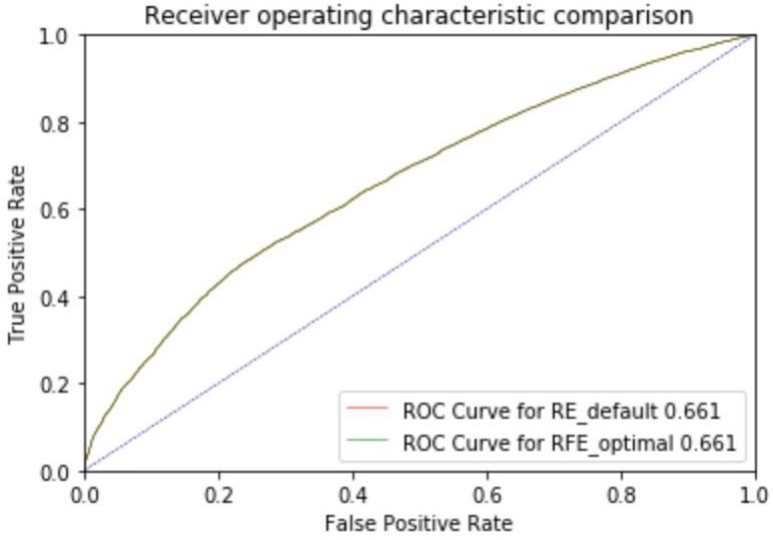
Train accuracy is slightly higher than test accuracy which is a sign of slight overfitting in the model. There is a very slight possibility model is overfitting, further analysis required to verify.

d. Report the top-3 important variables (in the order) in the model.

```
number_outpatient : 0.5106284774974944
num_medications : 0.2772363423172884
num_procedures : 0.12985952827545874
```

Question 4

Produce the ROC curve for all different regression models. Using the best regression model, can you identify which patients could potentially “readmitted”? Can you provide general characteristics of those patients?



```
ROC index on test for RE_default: 0.6610745165121024
ROC index on test for RFE_optimal: 0.6610612730520442
```

The default model perform a tad better than the optimal model. The default model has 3 more features than the RFE model. Risk of overfitting is lower with the default model as the test and train accuracy results are extremely close.

Patients with higher frequency in number_inpatient, number_emergency, number_outpatient, number_diagnoses, and use of diabetesMed are more likely to be readmitted.

Predictive modelling using Neural Networks

Question 1
What pre-processing was required on the dataset before neural network modelling? What distribution split between training and test datasets have you used?

Using the same pre-processed data from decision tree question. In addition, have scale the data for the neural network model. Have split into 70% training and 30% testing.

Question 2
Build a Neural Network model using the default setting. Answer the following:

a. Explain the parameters used in building this model, e.g., network architecture, iterations, activation function, etc.

solver='adam' : adam stands for Adaptive Moment Estimation. Solver is the algorithm to find the optimum weight in the neural network. Starts with set of randomly generated weight, then with each epoch of adam, predictions are made on X_train and the error value (cost) is calculated, The aim is to minimise the cost function.

activation='relu' : relu stands for rectified linear unit. Activation function is the function used in the hidden layers of the neural network.

hidden_layer_sizes=(100,) : number of layers and neurons. Default is one input layer, one hidden layer with 100 neurons and one output layer.

alpha=0.0001 : the learning rate for the adam algorithm. Larger alpha means the adam will take "larger" steps and train faster, but it might miss the optimal solution. Smaller alpha results in "smaller" steps, a slower training speed it might get stuck at the local minimum.

max_iter=200 : maximum number of iterations (epoch) to search for the minimal cost.

b. What is the classification accuracy on training and test datasets?

Train accuracy: 0.6902086677367576
Test accuracy: 0.6231684079111637

c. Did the training process converge and result in the best model?

The training process did not converge at default setting. Maximum iteration is changed to 300 with the following accuracy result:

Train accuracy: 0.6904621103320098
Test accuracy: 0.6134437216637099

Train accuracy is higher than test accuracy. That is an indication of overfitting to the training set.

Question 3
Refine this network by tuning it with GridSearchCV. Report the trained model.

a. Explain the parameters used in building this model, e.g., network architecture, iterations, activation function, etc.

activation='relu'

hidden_layer_sizes=(16,) : one input layer, one hidden layer with 16 neurons and one output layer.

alpha=0.0001

max_iter=200

b. What is the classification accuracy on training and test datasets?

Train accuracy: 0.650812424319225
Test accuracy: 0.6338787042512649

```
Train accuracy: 0.650812424319225
Test accuracy: 0.6338787042512649
              precision    recall  f1-score   support

           0       0.64      0.73      0.68      8183
           1       0.62      0.53      0.57      7036

    accuracy                           0.63     15219
   macro avg       0.63      0.63      0.63     15219
weighted avg       0.63      0.63      0.63     15219

{'alpha': 0.0001, 'hidden_layer_sizes': (16,)}
```

c. Did the training process converge and result in the best model?

Yes, the training process converge.

d. Do you see any sign of over-fitting?

Train accuracy is slightly higher than test accuracy. That is an indication of slight overfitting to the training set.

Question 4

Let us see if feature selection helps in improving the model? Build another Neural Network model with reduced features set. Perform dimensionality reduction by selecting variables with a decision tree (use the best decision tree model that you have build in the previous modelling task). Tune the model with GridSearchCV to find the best parameters setting. Answer the followings:

a. Did feature selection favour the outcome? Any change in network architecture? What inputs are being used as the network input?

REF model feature selection using 5 variables from decision tree. The train and test accuracy gap is smaller though the test accuracy is the same but the train accuracy is lower compared to previous model. It is inconclusive if REF is performing better.

Parameters used in the model:

activation='relu'

solver='adam'

alpha = 0.00001

hidden_layer_sizes' = 18

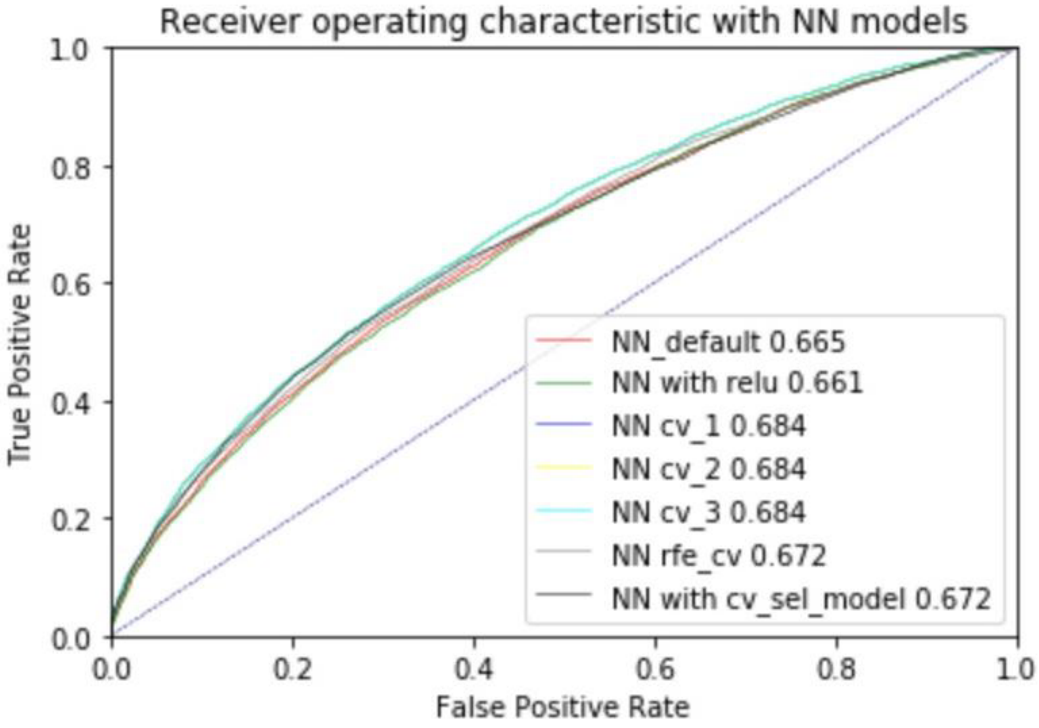
b. What is the classification accuracy on training and test datasets?

```
Train accuracy: 0.6404775984906086
Test accuracy: 0.6338787042512649
              precision    recall  f1-score   support

           0       0.64      0.74      0.69      8183
           1       0.63      0.51      0.56      7036

    accuracy                           0.63     15219
   macro avg       0.63      0.63      0.62     15219
weighted avg       0.63      0.63      0.63     15219

{'alpha': 1e-05, 'hidden_layer_sizes': (18,)}
```

c. How many iterations are now needed to train this network?
max_iter=200

d. Do you see any sign of over-fitting? Did the training process converge and result in the best model?
There is still a slight sign of overfitting as test accuracy is slightly higher than train accuracy.

Question 5
Produce the ROC curve for all different NNs. Now, using the best neural network model, can you provide general characteristics of the patients identified by the model? If it is difficult (or even infeasible) to comprehend, discuss why?

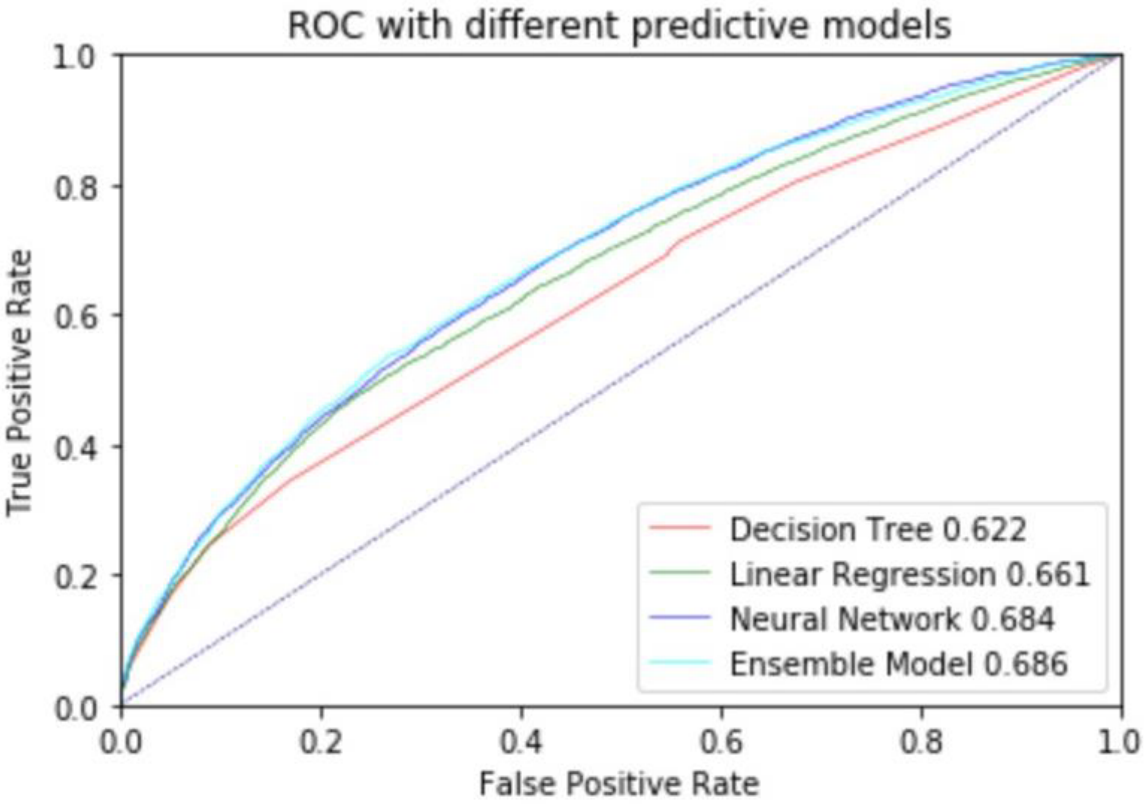


cv_1, cv_2, and cv_3 have produced similar results. It is difficult to have a generalize characteristics for patients who are more likely to be readmitted as NN model is operated in a black box and 29 features are used in the model (excluding readmitted as it is the target) from the dataset.

It is not possible to comprehend since NN is operated under a black box. We are unable to understand how it is done with the 16 hidden nodes and the weights used between features and layers.

Final remarks: Decision making

Question 1
Finally, based on all models and analysis, is there a model you will use in decision making? Justify your choice. Draw a ROC chart and Accuracy Table to support your findings.




```
ROC index on test for Decision Tree: 0.621844330968882
ROC index on test for Linear Regression: 0.6610745165121024
ROC index on test for Neural Network: 0.6842947396386121
ROC index on test for Ensemble Model: 0.6857473344432019
```

	Decision Tree	Linear Regression	Neural Network	Ensemble Model
Train accuracy	0.643941	0.628594	0.650812	0.649517
Test accuracy	0.635456	0.628294	0.633879	0.638600
ROC index on test	0.621844	0.661075	0.684295	0.685747

Preference would be to use linear regression. Although ensemble model has the best performance. Need to further investigate and statistical analysis to see if the model is performing statistically significantly better than linear regression in order to switch to a more complicated model. As ensemble model is extremely complex combining all three prediction models. It is extremely difficult to explain how the model works. Explainability is gaining more importance and momentum especially since the enactment of GDPR and particularly vital in healthcare. Linear regression is much easier to understand and can see how the features are contributing to the model. In connection, its train and test accuracy results are very close and is lesser subject to overfitting.

Question 2
Can you summarise the positives and negatives of each predictive modelling method based on *this analysis*?

Positive	Negative
Decision Tree	
Do not need to scale and normalise data Easy to interpret Can handle large number of features Can handle data with linear / non-linear characteristics Can handle continuous/categorical features Fast classification once trained	Difficult to read the tree diagram if large tree Tree changes a lot even with minor changes in the dataset Risk of overfitting Cannot handle lots of missing data Cannot handle complicated relationships between features
Logistic Regression	
Target variable is categorical Fast classification once trained Easy to interpret and identify important features from the absolute value of the coefficients Can handle continuous/categorical features	Cannot handle missing, noisy data Cannot handle large number of features High training time Not too flexible as based on “line of best fit”
Neural Network	
Can learn complicated relationships from large dataset Fast classification once trained Works well even with missing or noisy data Can handle data with linear / non-linear characteristics Can handle continuous/categorical features	Don’t understand what goes behind the model as it is a black box Takes long time to train the model Trial and errors for the hyperparameters e.g number of nodes, alpha Can easily overfit model if too many hidden nodes or underfit if not enough hidden nodes