IFN 509: Data Exploration and Mining

Assessment 2

Team Name: No Breakfast

Group No. 18

|  |  |
| --- | --- |
| Student Name | Student Id |
| Lam Hoi Yen | n11675403 |
| Chonlita Saikandok | n11734264 |
| Yang Quan Marc Loo | n11703661 |

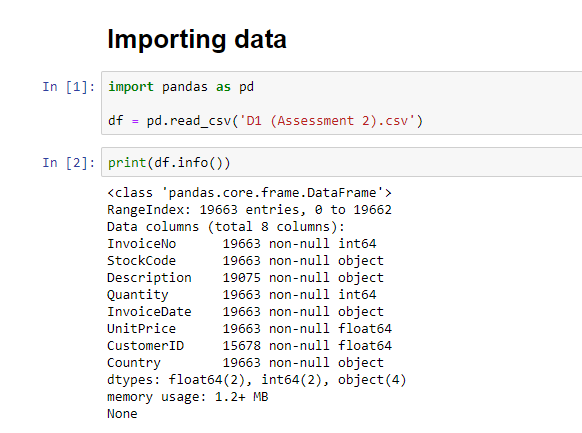
|  |  |  |  |
| --- | --- | --- | --- |
|  | Lam Hoi Yen | Chonlita Saikandok | Yang Quan Marc Loo |
| Lam Hoi Yen | <33.33%> | <33.33%> | <33.33%> |
| Chonlita Saikandok | <33.33%> | <33.33%> | <33.33%> |
| Yang Quan Marc Loo | <33.33%> | <33.33%> | <33.33%> |

**Case Study 1: Discovering behavioural patterns in online retail data through Association Mining**

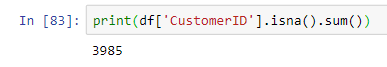
**1. What pre-processing was required on the dataset before building the association mining**

**model? What variables did you include in the analysis? Justify your choice.**

We first import the dataset and understand the data types.



After importing, the variables are evaluated through the use of describe, value\_count and unique. There are missing values found in ‘Description’ and ‘CustomerID’. However, for apriori algorithm, ‘Customer ID’ is not used thus na should be ignored.



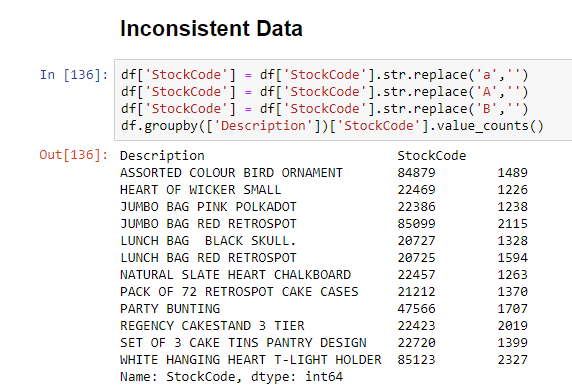
Moreover, ‘StockCode’ might be used for association mining. This variable is supposed to have 12 unique items, similar to ‘Description’.

|  |  |
| --- | --- |
| **Stock Code** | **Description** |
|  |  |

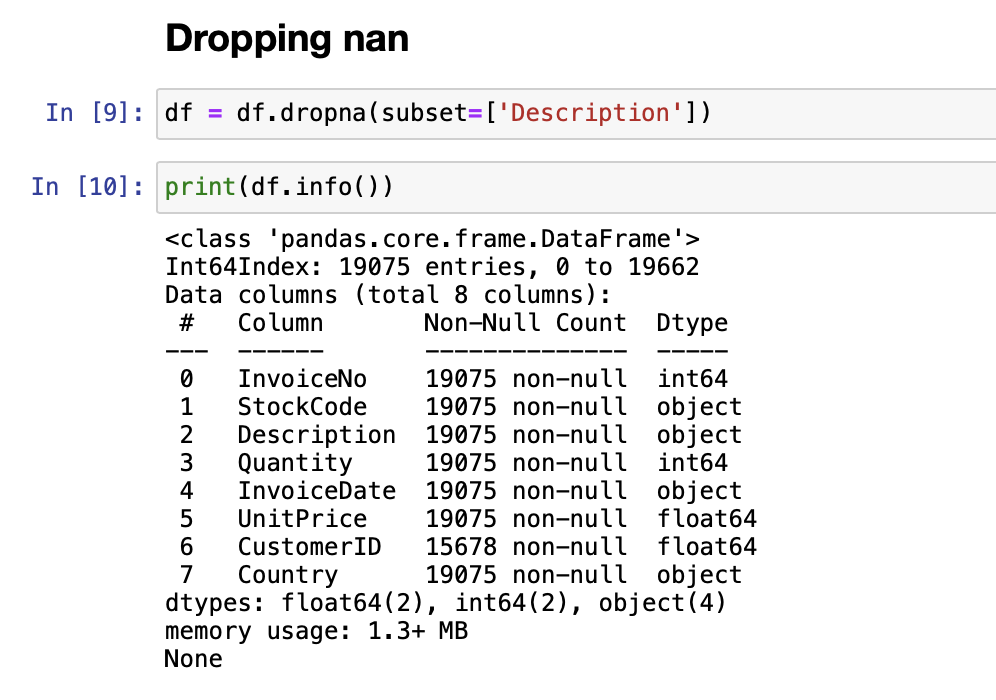
As ‘StockCode’ and ‘Description’ should have the same unique numbers (12 items), groupby function could be used to see why both variables are not aligned. From the groupby below, it is noted that there is inconsistency in ‘StockCode’ under the Description of “WHITE HANGING HEART T-LIGHT HOLDER”. The remaining StockCode that is not found under any Description signifies that the Description is missing or nan.

|  |  |
| --- | --- |
| **Stock Code (Inconsistent Data)** | **Stock Code (Description has missing values/nan)** |
|  |  |

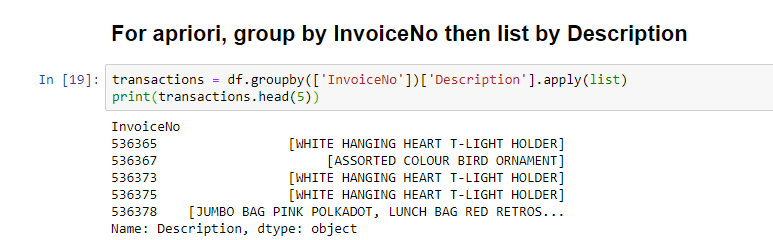
Prior to cleaning missing values, the inconsistency of ‘StockCode’ under existing ‘Description’ should be fixed. Assuming that ‘StockCode’ should consist solely of numerical characters and not contain any alphabet, we will only present the ‘StockCode’ in numerical format (represented by an object type) by replacing ‘a’, ‘A’, and ‘B’ characters in the ‘StockCode’ column with blanks (i.e., we removed such characters).

****

Following on, the missing values of ‘Description’ should be dropped as apriori algorithm’s code will not be able to run float type (nan).

****

After dropping the nan, the variables that are included in the analysis will be ‘InvoiceNo’ and ‘Description’. As association mining serves to understand a combination of items in the same transaction, ‘InvoiceNo’ is a unique transaction that records a combination of items - Description by a customer. Hence, the variables shown below will be used.

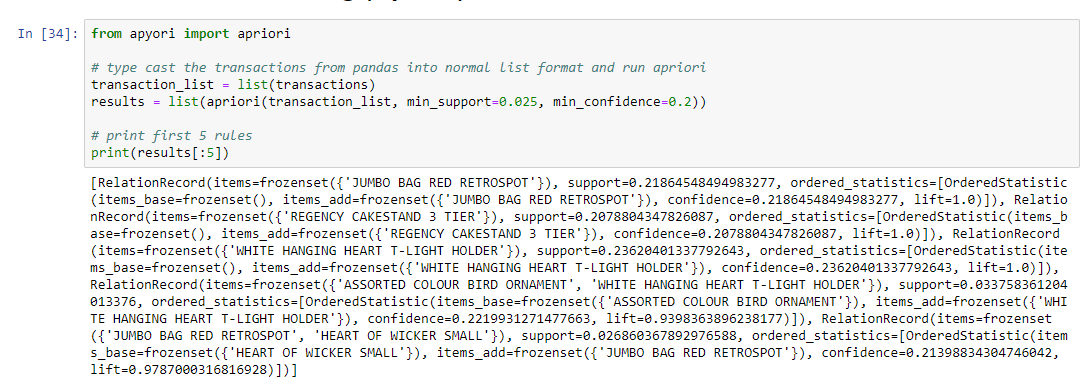


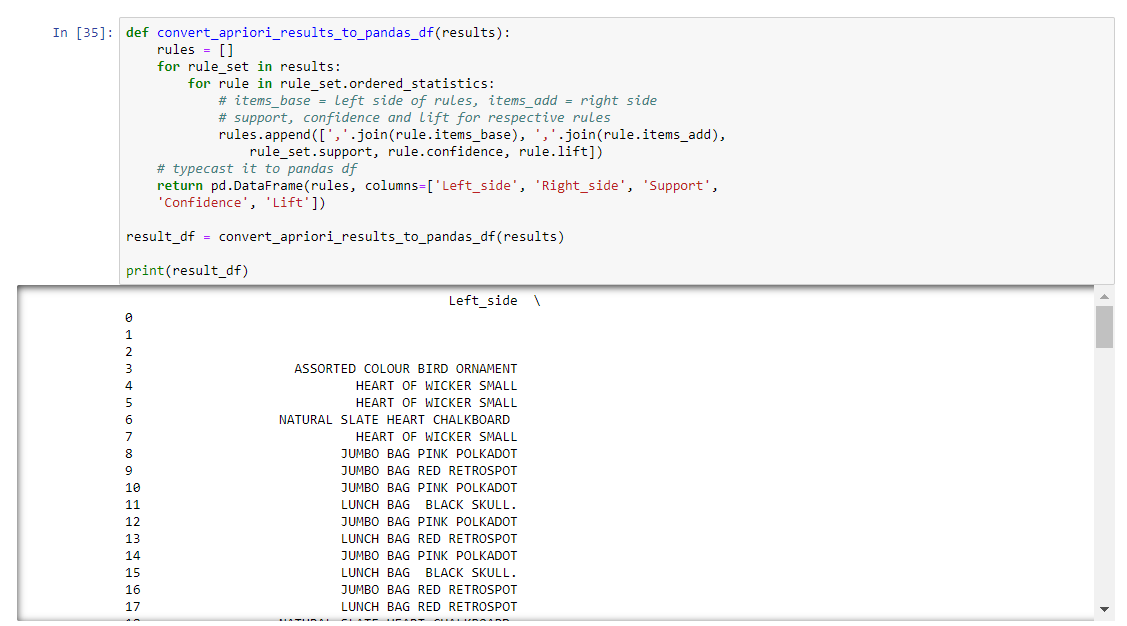
**2. Conduct association mining and answer the following:**

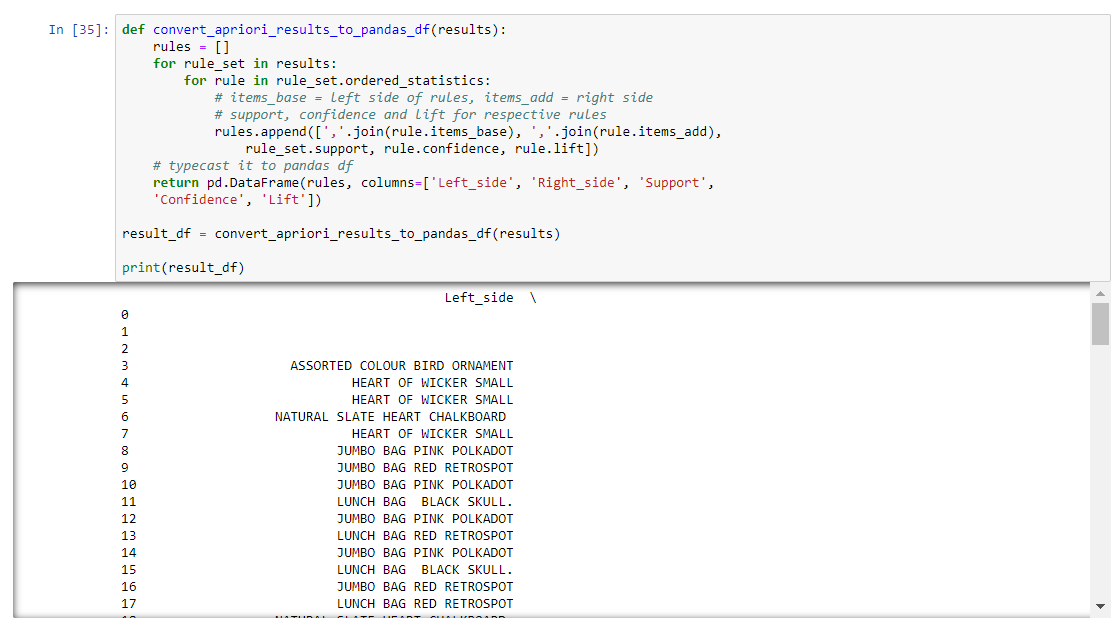
**a. What ‘min\_support’ and `min\_confidence’ thresholds were set for this mining exercise? Rationalize why these values were chosen.**

The ‘min\_support’ of 0.025 and ‘min\_confidence’ of 0.2 are chosen. We experimented with different values for ‘min\_support’ and ‘min\_confidence’ to find the most suitable thresholds for our dataset. By choosing these thresholds, we sought to achieve a balance between identifying meaningful associations and avoiding an excessive number of rules. The selection of the ‘min\_support’, 0.025 signifies that only itemsets that appeared in at least 2.5% of the transactions were examined. Similar to the ‘min\_confidence’ threshold of 0.2, we only considered association rules with a confidence level of at least 20%.

With the chosen ‘min\_support’ and ‘min\_confidence’ thresholds, 43 association rules have been identified. This also enables us to identify the top 5 most commonly purchased items with ‘JUMBO BAG PINK POLKADOT’ in question 3 below.

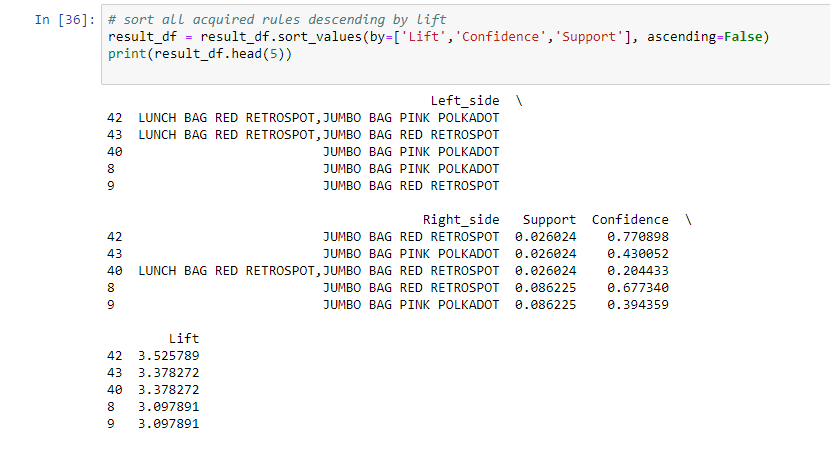


****

****

**2. Report the top-5 (interesting) rules and interpret them.**

The following are the top 5 most interesting association rules identified based on the highest confidence and lift values:



Explanation for rules identified:

1. **Rule 42: {JUMBO BAG PINK POLKADOL, LUNCH BAG RED RETROSPOT} => {JUMBO BAG RED RETROSPOT}**

* Support: 2.60%
* Confidence: 77.09%
* Lift: 3.526

With a support of 2.60%, it means these 3 items appear together in about 2.60% of all transactions. The high confidence of 77.09% indicates that 77.09% of customers who purchase both ‘JUMBO BAG PINK POLKADOL’ and ‘LUNCH BAG RED RETROSPOT’, are significantly more likely to also purchase ‘JUMBO BAG RED RETROSPOT’. The lift value of 3.526, which is greater than 1, suggests a substantial positive effect. Based on the high confidence and lift values, these items appear to have a strong association.

1. **Rule 43: {LUNCH BAG RED RETROSPOT, JUMBO BAG RED RETROSPOT} => {JUMBO BAG PINK POLKADOT}**

* Support: 2.60%
* Confidence: 43.01%
* Lift: 3.378

This association rule indicates that when customers purchase both ‘LUNCH BAG RED RETROSPOT’ and ‘JUMBO BAG RED RETROSPOT, there is a moderate possibility, 43.01%, that they will also buy ‘JUMBO BAG PINK POLKADOT’. The proportion of transactions that contain three items is 2.60% of all transactions. Although the confidence level is moderate, the lift value of 3.378 suggests a significant association between these items.

1. **Rule 40: {JUMBO BAG PINK POLKADOT} => {LUNCH BAG RED RETROSPOT, JUMBO BAG RED RETROSPOT}**

* Support: 2.60%
* Confidence: 20.44%
* Lift: 3.378

The 2.60% support highlights that 2.60% of all transactions contain ‘JUMBO BAG PINK POLKADOT’, LUNCH BAG RED RETROSPOT, and JUMBO BAG RED RETROSPOT. Within these transactions, there is a moderate probability of 20.44% that customers will purchase both ‘LUNCH BAG RED RETROSPOT’ and ‘JUMBO BAG RED RETROSPOT’ together in the same transaction. The lift value reinforces that the purchase of ‘JUMBO BAG PINK POLKADOT’ is 3.378 times more likely to be associated with the joint purchase of ‘LUNCH BAG RED RETROSPOT’ and ‘JUMBO BAG RED RETROSPOT’ compared to if these items were bought independently.

1. **Rule 8: {JUMBO BAG PINK POLKADOT} => {JUMBO BAG RED RETROSPOT}**

* Support: 8.62%
* Confidence: 67.73%
* Lift 3.098

The support value of 8.62% implies that ‘JUMBO BAG PINK POLKADOT’ and ‘JUMBO BAG RED RETROSPOT’ co-occur in roughly 8.62% of all transactions. At a confidence level of 67.73%, it shows that in nearly 68% of instances, the purchase of ‘JUMBO BAG PINK POLKADOT’ is associated with the purchase of ‘JUMBO BAG RED RETROSPOT’. The lift value exceeding 1 (3.098) indicates that buying ‘JUMBO BAG PINK POLKADOT’ has a favorable impact on the purchase of ‘JUMBO BAG RED RETROSPOT’.

1. **Rule 9: {JUMBO BAG RED RETROSPOT} => {JUMBO BAG PINK POLKADOT}**

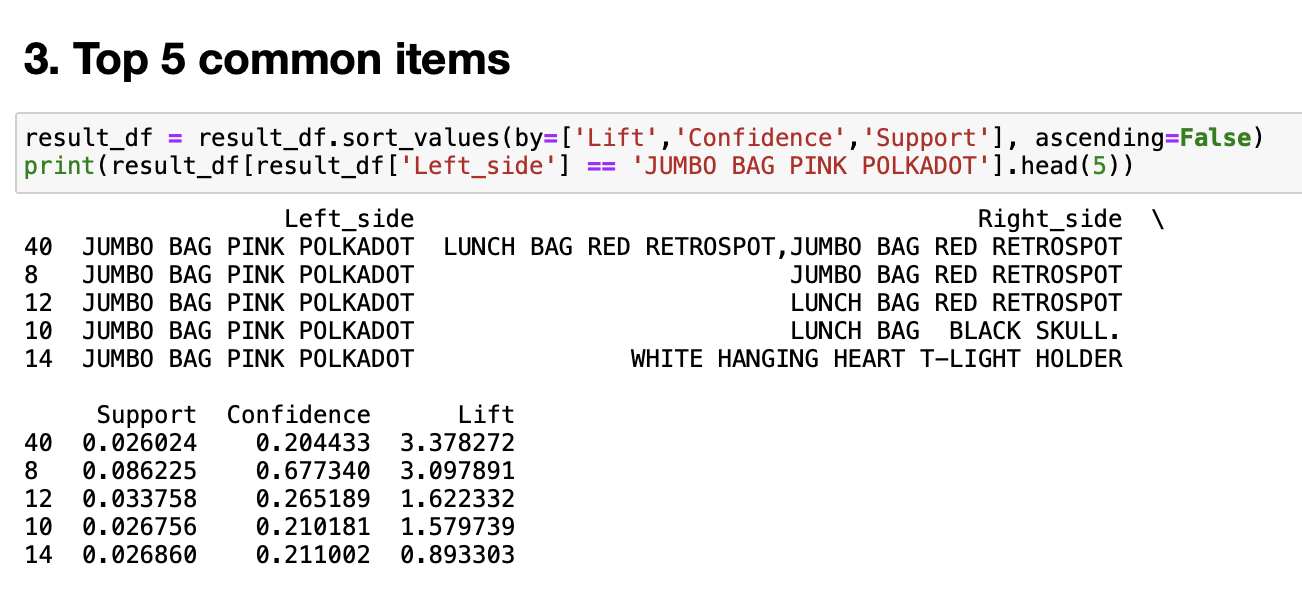
* Support: 8.62%
* Confidence: 39.44%
* Lift: 3.098

The 8.62% support value represents the prevalence of ‘JUMBO BAG RED RETROSPOT’ and JUMBO BAG PINK POLKADOTbeing purchased. There is a moderate likelihood (approximately 39.44%) that customers who purchase ‘JUMBO BAG RED RETROSPOT’ will also add ‘JUMBO BAG PINK POLKADOT’ to their purchase. The lift value of 3.098, which is greater than 1, signifies that the joint purchase of ‘JUMBO BAG RED RETROSPOT’ has a substantial influence on the purchase of ‘JUMBO BAG PINK POLKADOT’ .

**3. List top-5 common items that customers have purchased along with ‘JUMBO BAG PINK POLKADOT’.**

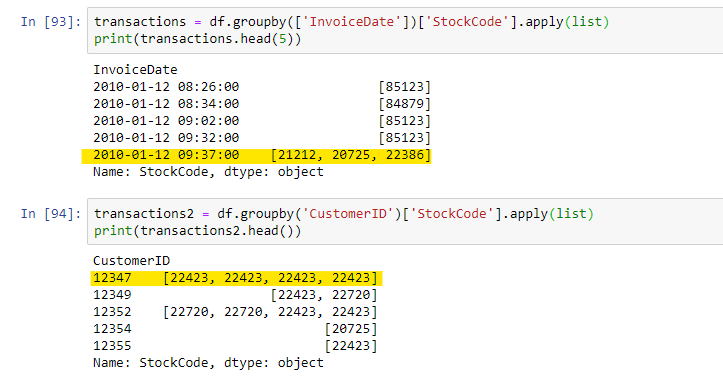
The top 5 items frequency purchased along with ‘JUMBO BAG PINK POLKADOT’ are as follows:

1. LUNCH BAG RED RETROSPOT, JUMBO BAG RED RETROSPOT
2. JUMBO BAG RED RETROSPOT
3. LUNCH BAG RED RETROSPOT
4. LUNCH BAG BLACK SKULL
5. WHITE HANGING HEART T-LIGHT HOLDER



**4. Can you perform sequence analysis on this dataset? If yes, present your results. If not, rationalize why.**

We are unable to perform sequence analysis on the dataset. As for now, a customer is able to purchase two or more items at the same time. Hence, we cannot determine which item is purchased first or which item is purchased after for a particular customer.



If we do sequential mining based on “InvoiceDate” and not the customer, the results will be irrelevant as it will be based after “InvoiceDate” instead of the purchasing behaviour of the customer. For instance, if sequential mining is performed using “InvoiceDate”, “StockCode”, it might indicate that Customer A purchases a particular or combination of items at a particular time, which is followed by Customer B purchasing a particular or combination of items at another time. Customer A and B purchasing behaviour should not be related, thus sequential mining cannot be carried out.

**5. In what ways can the results of this task be utilised by the relevant decision-makers?**

The result of analysing association rules provide decision-makers with actionable insights applicable across multiple aspects. By understanding which products tend to be bought together, decision-makers are able to optimise inventory level and offer discounts or bundle deals for frequently purchased items.

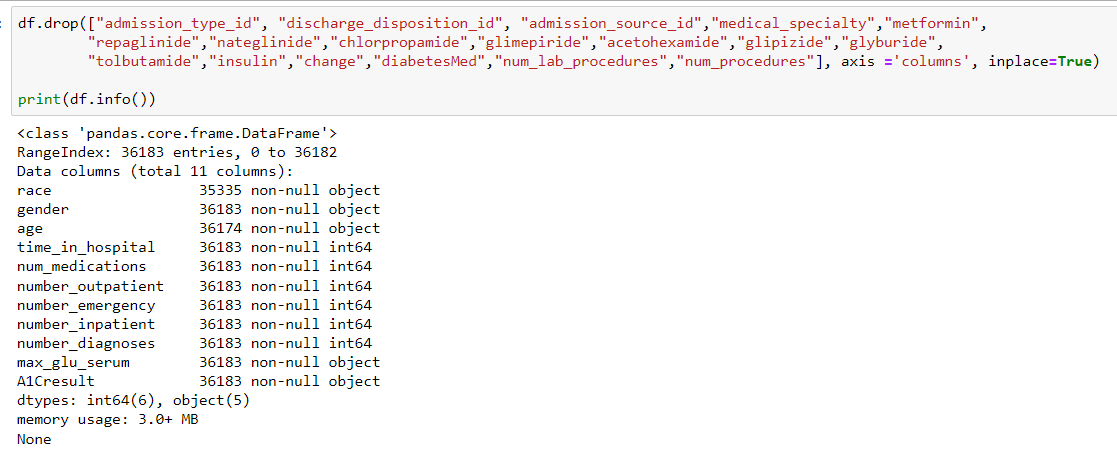
Items that are commonly purchased together can also be placed in close proximity in physical stores or online platforms so as to increase the chance of upselling. This can lead to improved business performance and profitability.

**Case Study 2: Discovering clusters describing common diabetic patient encounters**

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

For the pre-processing, we first dropped Unrelated Data and Unique Identifiers as shown below:

* Unique Identifiers: "admission\_type\_id", "discharge\_disposition\_id", "admission\_source\_id", "medical\_specialty"
* Unrelated Data: "metformin", "repaglinide", "nateglinide", "chlorpropamide", "glimepiride", "acetohexamide", "glipizide", "glyburide", "tolbutamide", "insulin", "change", "diabetesMed", "num\_lab\_procedures", "num\_procedures"



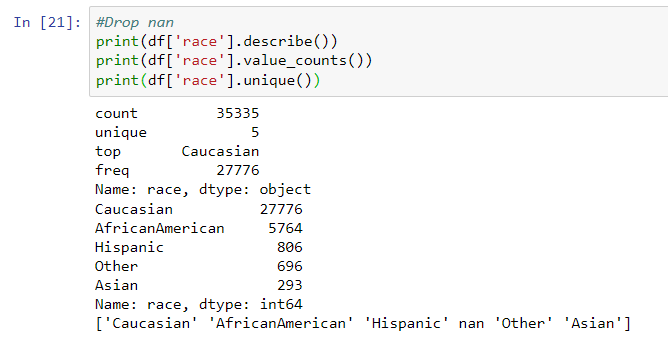
These remaining variables are relevant to the clustering analysis. These datasets could be generalised to the following characteristics:

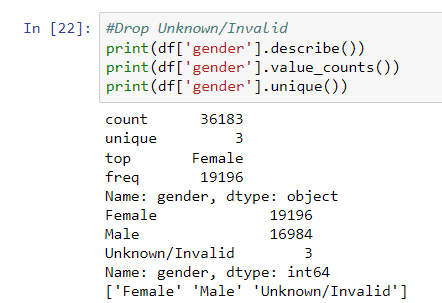
* Demographic characteristics: race, gender, age
* Health Status: time\_in\_hospital, num\_medications, number\_diagnoses, max\_glu\_serum, A1Cresult
* Utilisation of Medical Services: number\_inpatient, number\_emergency, number\_outpatient

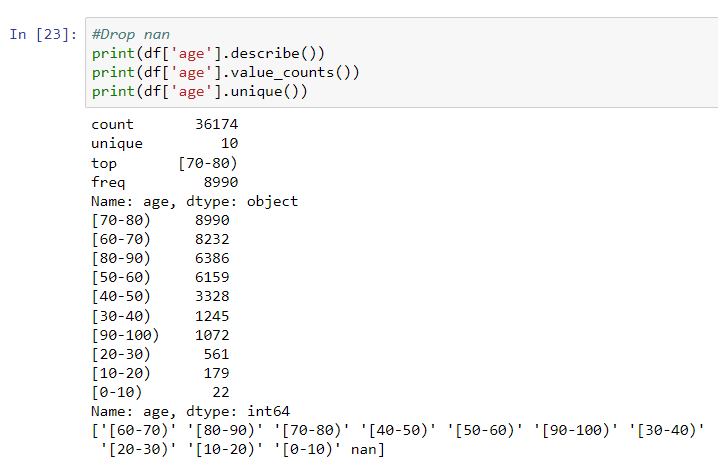
Clustering analysis will allow us to understand the combined clinical characteristics of the patient, including its demographic, health status and utilisation of medical services. However, before performing cluster analysis, summary statistics and visualisation plots will be carried out prior to data cleaning.

**Categorical Value**

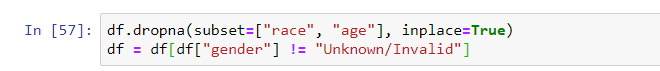
In the previous df.info(), it is observed that race, gender, age do not have 36183, the missing values could be seen through the codes below.



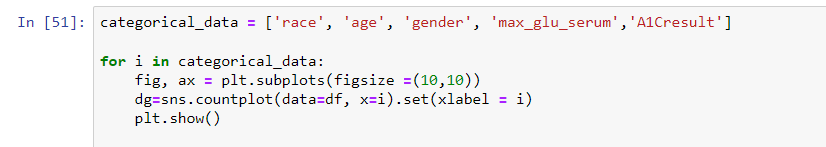




The missing values will be dropped as it will affect the clustering analysis.



To understand the distribution of the categorical data, a count plot could be made:



|  |  |
| --- | --- |
| **Variable** | **Count Plot** |
| race |  |
| age |  |
| gender |  |
| max\_glu\_serum |  |
| A1Cresult |  |

From the data distribution, it could be seen that age and gender is more distributed while the race is more centred around caucasian, max\_glu\_serum and A1Cresult are centred towards none. Even though race has a higher proportion towards caucasian, the data is meaningful as it could suggest that the area has a higher population of caucasians. However, in the case of A1C result and max\_glu\_serum, the value of “none” provides no significance for the clustering analysis. It is rather essential to opt out these variables for the clustering process.

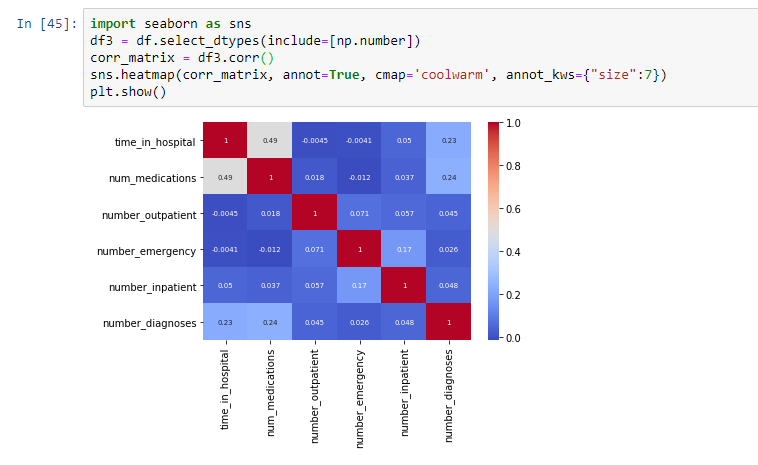
**Numerical Variables**

After analysing the categorical variables, distribution plots could be made for the numerical variables.



|  |  |
| --- | --- |
| **Variable** | **Distribution Plot** |
| time\_in\_hospital |  |
| num\_medications |  |
| number\_outpatient |  |
| number\_emergency |  |
| number\_inpatient |  |
| number\_diagnoses |  |

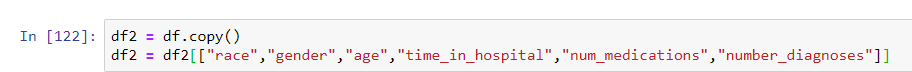
From the distribution plot, all of the variables are rather skewed but the variables with extreme cases are number\_emergency, number\_inpatient and number\_outpatient which have most values around 0. This could also be shown through their low correlation with other variables under the correlation matrix:



If most of the values are situated around 0 and provide no correlation with other variables, they should be opted out for the clustering process.

For the clustering analysis, A1C result, max\_glu\_serum, number\_emergency, number\_inpatient and number\_outpatient are excluded with the explanation provided previously. The remaining variables are selected:

* Demographic characteristics: race, gender, age
* Health Status: time\_in\_hospital, num\_medications, number\_diagnoses



All categorical variables are mapped prior to conducting clustering analysis. Following on, all variables are standardised under data transformation as Clustering is sensitive to inputs on different scales.



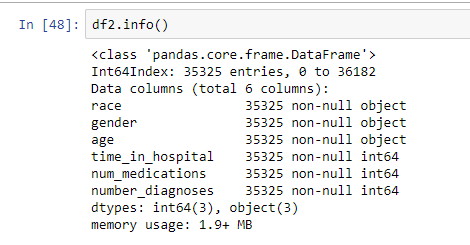
**2. Build a clustering model to profile the characteristics of diabetic patients. Answer the following:**

**a) What clustering algorithm have you used?**

Kprotoype is used as the variables selected for the clustering contain both numerical and categorical data. It is a suitable option that can effectively manage both categories of variables.

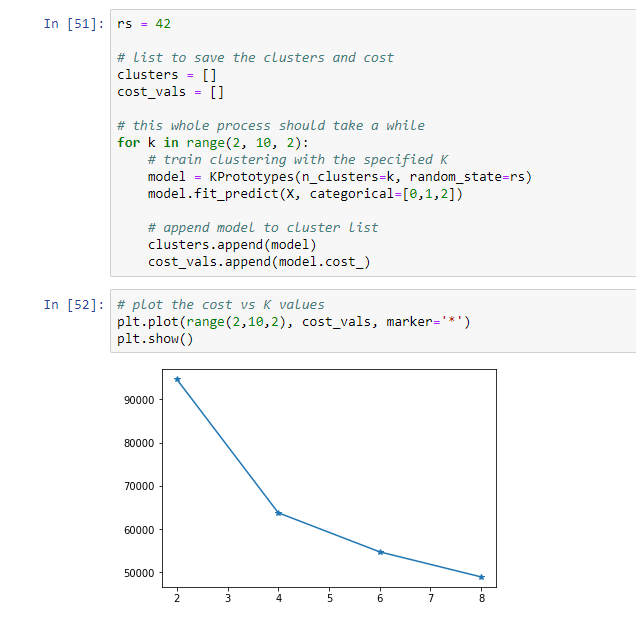
**b) List the attributes used in this analysis.**

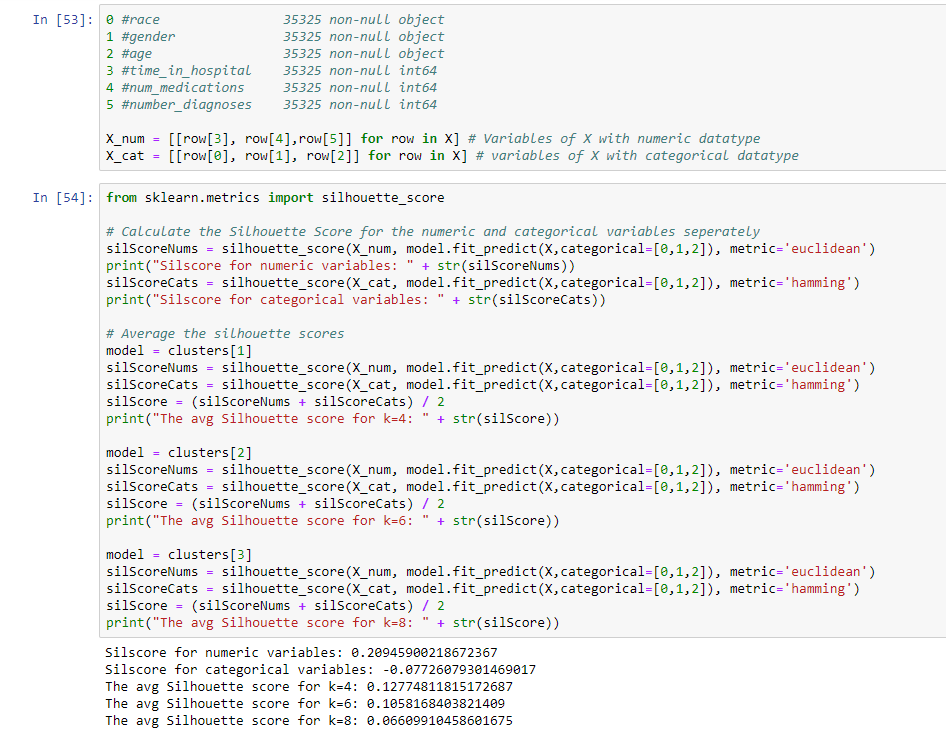
The reason why the attributes are chosen is provided in the Task2 Q1.



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

We identified the optimal number of clusters by using the elbow method. Upon examining the elbow plot, there were three potential values for K: K=4, K = 6, and K = 8. Subsequently, we calculated the silhouette scores for each of these values. By assessing the silhouette scores, K=4 stands out as the most favourable option, as it has the highest average silhouette score.





**d) Did you normalise/standardize the variables? What was its effect on the model – does the variable normalization/standardize process enable a better clustering solution?**

For categorical variables, we first created a dictionary to map category labels to numerical values. For example, we used a ‘race\_map’ to map race categories i.e., Caucasian, AfricanAmerican, Hispanic, Other, and Asian to numerical values 1, 2, 3, 4, and 5 respectively. We similarly created an ‘age\_map’ and ‘gender\_map’ for age categories.

For both numerical variables and encoded categorical variables, initially, we converted the DataFrame “df” into a numerical matrix denoted “X”. Then, we performed a scaling operation on the data contained in matrix “X” by using StandardScaler to standardise or adjust the scale of each feature to have a mean of 0 and a standard deviation of 1, thereby ensuring that all variables have the same scales.

**3) For the model with the optimal number of clusters,**

**a) Visualize the clusters using ‘pairplot’ and interpret the visualization.**

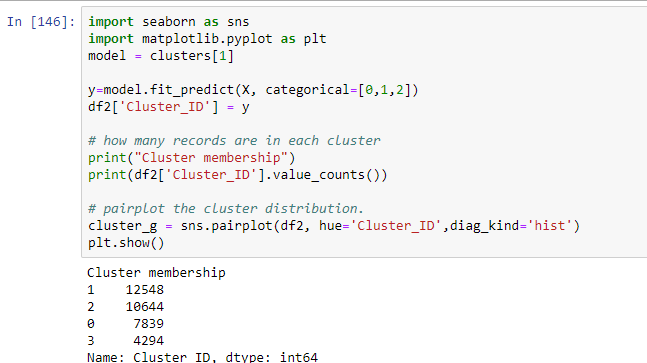
The pairplot is not as useful in this scenario as overlapping occurs. The pairplot below shows 4 clusters, but cluster 0 labelled with colour “blue” is mostly overlapped with other clusters. The following are the interpretations:

* Race
  + Distributed quite evenly. Cluster 0 can’t be seen which might be due to overlapping.
* Gender
  + Distributed quite evenly. Cluster 0 can’t be seen which might be due to overlapping.
* Age
  + Cluster 1 spans across (20-100) with the highest age distribution around (70-80), Cluster 2 has similar age spans of (20-100) with the highest age distribution around (60-70), Cluster 3 has covers mostly older age span (30-100) with highest age distribution of around (70-80) as well. Cluster 0 can’t be seen which might be due to overlapping, but it seemingly covers from age 10 onwards.
  + ***\*\*Note that the numbers are an approximation from the pairplot***
* Time in hospital
  + Cluster 0 is observed to center mostly on 1 day, possibly covering lower time in hospital. Cluster 1 highest distribution is on 2 days, covering mostly lower time in hospital but longer than cluster 0. Cluster 2 is observed to distribute mostly on 4 days and is more spread out to longer days compared to Cluster 1. Cluster 3 spans the longest time in the hospital.
* num medications
  + Cluster 0 is observed to gather around a lower number of medications. Cluster 1 seems to be higher than cluster 0 but still on the lower end. Cluster 2 is more spread out with a higher number of medications than cluster 1. Cluster 3 covers the most medications
* num diagnoses
  + Cluster 0 is observed to have the lowest number of diagnoses. Cluster 1 seems to have a higher number of diagnoses than cluster 0 . Cluster 2 is not much different from cluster 1 but has a slightly higher number of diagnoses. Cluster 3 seems to have the same span with Cluster 1 but has lower frequency, as represented by the Cluster membership.

From the aforementioned explanation, and the other forms of combination observed in the pairplot (i.e age and time in hospital, time in hospital and medications etc). The concluding interpretation will be:

* Cluster 0 covers youngest patients from children, with lowest time in hospital, number of medications and number of diagnoses.
* Cluster 1 covers young adult patients onwards with lower time in hospital, number of medications, moderate number of diagnoses.
* Cluster 2 covers young adult patients onwards with mid time in hospital, number of medications and moderate number of diagnoses.
* Cluster 3 covers working adult patients onwards with higher time in hospital, number of medications and moderate number of diagnoses.

However, the overlapping might cause misinterpretation, thus cluster distribution will be performed in the next task.

****

****

**b) Characterize the nature of each cluster by giving it a descriptive label and a brief description.**

The Cluster distribution below provides a more insightful analysis. Gender’s distribution is similar in every cluster. The clusters are labelled as shown below and the explanations will be provide subsequently:

* Cluster 0 is labelled as "Minimal Intensity" (Low utilisation of medical resources)
* Cluster 1 is labelled as "Moderate Intensity" (Moderate utilisation of medical resources)
* Cluster 2 is labelled as "Enhanced Intensity" (Higher utilisation of medical resources and regular checkups)
* Cluster 3 is labelled as "Critical Intensity" (High utilisation of medical resources and immediate action monitoring)

Observations of distribution plot:

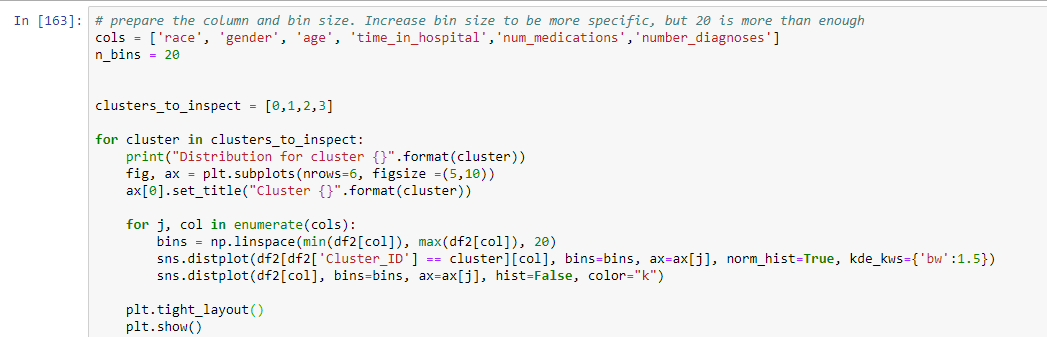
* Age
  + cluster 0: Age highest about (60-70), and there is more distribution on younger age
  + cluster 1: Age highest about (70-80), more distribution around (70-80)
  + cluster 2: Age highest about (60-70), more distribution around (60-70), higher (70-80) and (80-90) to cluster 1
  + cluster 3: Age highest about (70-80), more distribution around (70-80), more similar to cluster 1
* Time in hospital
  + cluster 0: Lower time in hospital, about 1-3 mostly, peaked 1
  + cluster 1: higher time in hospital, about 1-3 mostly, peaked 2
  + cluster 2: higher time in hospital, about 3-5 mostly, peaked 4 and 5 is near peak
  + cluster 3: highest time in hospital, about 8 onwards, peaked 8 and 9 is near peak
* num\_medications
  + cluster 0: Low num medications, covers on 0-30 and alot, peak 10
  + cluster 1: Low num medications, covers on 0-25, peak 10 but more distribution on 11 than cluster 0
  + cluster 2: higher num medications, covers on 10-40, peak 20
  + cluster 3: highest num medications, covers on 0-60, peak 28
* num\_diagnoses
  + cluster 0: Low num diagnoses, covers on 0-7, peak 6
  + cluster 1: higher num diagnoses, covers on 7 and 9, peak 9
  + cluster 2: higher num diagnoses, covers on 7 and 9, peak 9
  + cluster 3: higher num diagnoses, covers on 5 and 9, peak 9

Taking a comparison of both pairplot and distribution plot, the concluding interpretation is made:

|  |  |  |
| --- | --- | --- |
| **Clusters** | **Initial Pair plot interpretation** | **Concluding interpretation after reviewing distribution plot** |
| Cluster 0 | Cluster 0 covers youngest patients from children, with lowest time in hospital, number of medications and number of diagnoses. | Cluster 0 covers more young patients, with lowest time in hospital, low number of medications and lowest number of diagnoses. |
| Cluster 1 | Cluster 1 covers young adult patients onwards with lower time in hospital, number of medications, moderate number of diagnoses | Cluster 1 covers more older patients with lower time in hospital, low number of medications and higher number of diagnoses |
| Cluster 2 | Cluster 2 covers young adult onwards patients with mid time in hospital, number of medications and moderate number of diagnoses. | Cluster 2 covers more older patients with mid time in hospital, higher number of medications and higher number of diagnoses. |
| Cluster 3 | Cluster 3 covers working adult patients onwards with higher time in hospital, number of medications and moderate number of diagnoses. | Cluster 3 covers most older patients with higher time in hospital, highest number of medications and higher number of diagnoses. |

It could be seen that there is worsening health from Cluster 0 to 3, particularly with rising time in hospital and number of medications. Though age and number of diagnoses are increasing across the cluster, relatively smaller differences could be seen from Cluster 1 to 3, except for Cluster 0 that clearly reflects young patients and low number of diagnoses.

As it can be based on worsening health, Cluster 0 will be a less intense patient that requires lower utilisation of hospital resources while Cluster 3 will be a high intensity patient that might need immediate actions. Hence, we thereby labelled the Clusters based on severity and intensity.

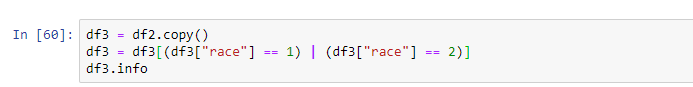


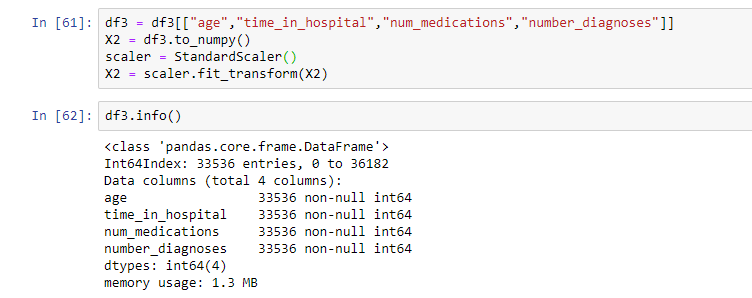
|  |  |
| --- | --- |
| **Cluster Labels (Distribution plot)** | |
|  |  |
|  |  |

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

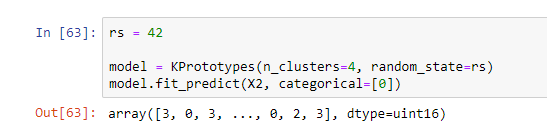
**a) List the attributes used in this analysis.**

To profile patients of specific races, Caucasian and Asian are filtered. Gender is not included as it did not add any significance in the previous clustering task.



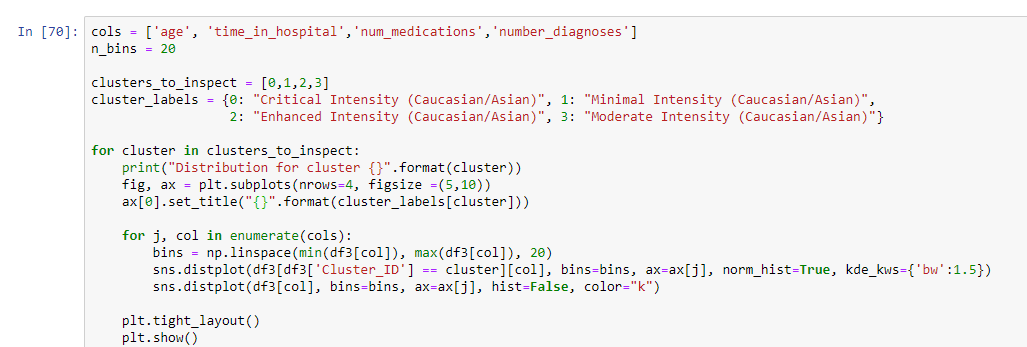


Following on, clustering is performed on the caucasian and asian patients.



**b) What difference do you see in this clustering interpretation when compared to the previous one (task 3)?**

To see the differences, a distribution plot is built instead of a pairplot as it shows a more detailed visualisation.



From the visualisation below, all clusters remain approximately similar thus we had added labels to the new clustering similar to the previous task. The only slight difference is in enhanced intensity where the time in hospital from both tasks have different peaks. The reason for the small difference is due to the high proportion of caucasians in the data set, which proves that the previous clustering was heavily influenced by the caucasian race.

|  |  |
| --- | --- |
| **Clustering for all races** | **Clustering for Caucasian and Asian** |
|  |  |
|  |  |
|  |  |
|  |  |

**5) In what ways can the results of this task be utilized by the relevant decision-makers?**

By assigning labels that reflect the intensity of the patient: “Minimal Intensity”; "Moderate Intensity"; "EnhancedIntensity"; "Critical Intensity", we are able to assess the risk of the patients. Subsequently, medical professionals can decide on their resource allocation and patient engagement.

Under resource allocation, medical professionals will be able to allocate medical resources based on the level of intensity. For instance, if there are two patients: one is “Critical Intensity” and the other is “Enhanced Intensity”, medical professionals can allocate limited supplies to the patient under “Critical Intensity” as she/he has higher health risk.

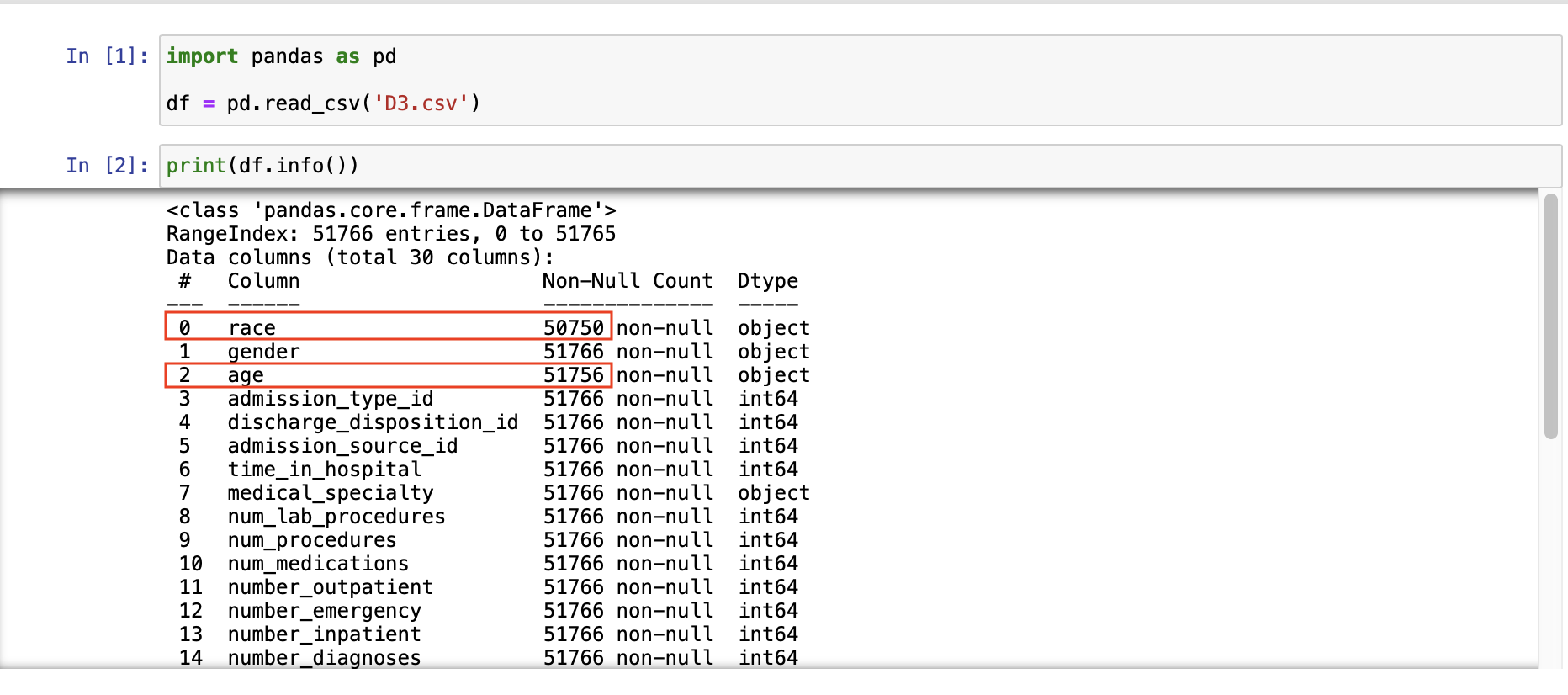
Under patient engagement, patients under “Minimal Intensity” may receive simple advice on managing their health, while for “Critical Intensity”, medical professionals might have to provide frequent checkups and interactions with the patients.

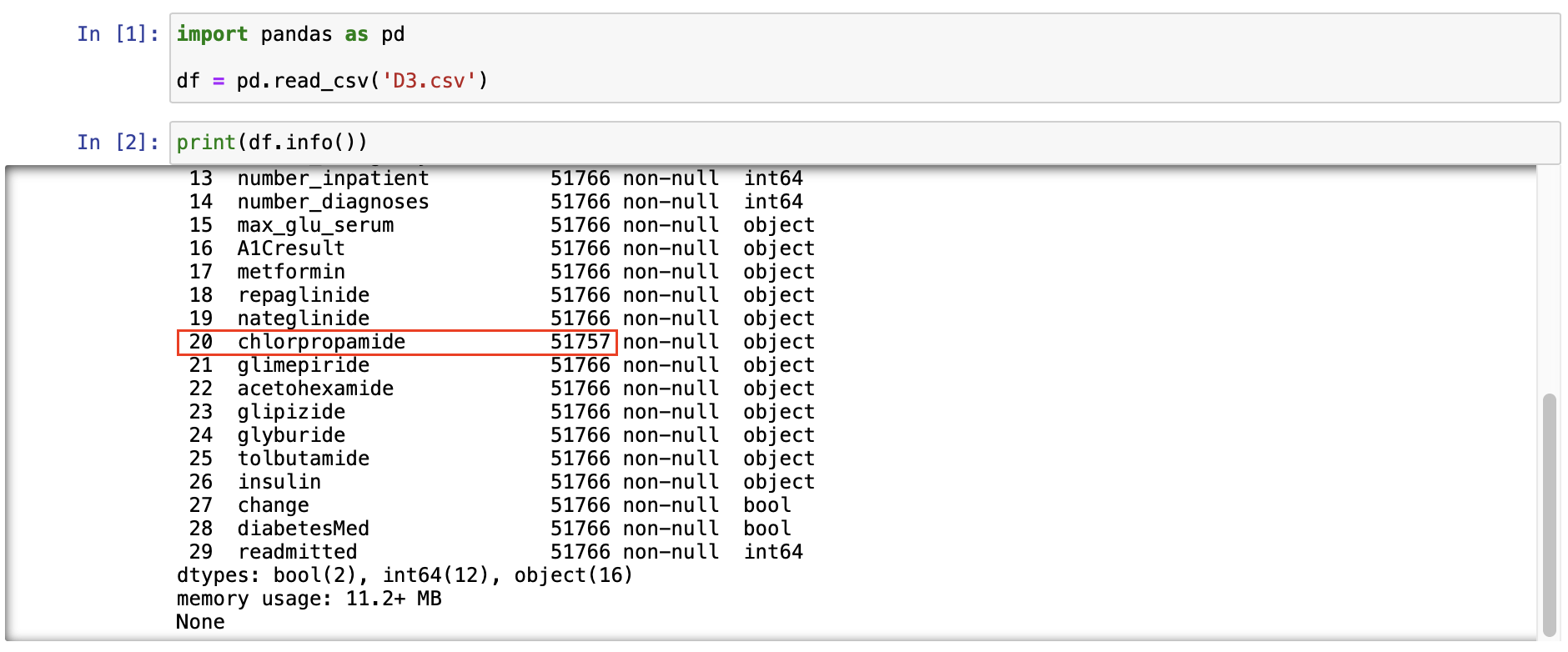
**Case Study 3: Decision Tree Modelling**

**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?**

We first imported the pandas library and read a CSV file named ‘D3.csv’ into a pandas DataFrame. After printing a summary of the DataFrame’s information. We observed that there are missing values in the following columns:

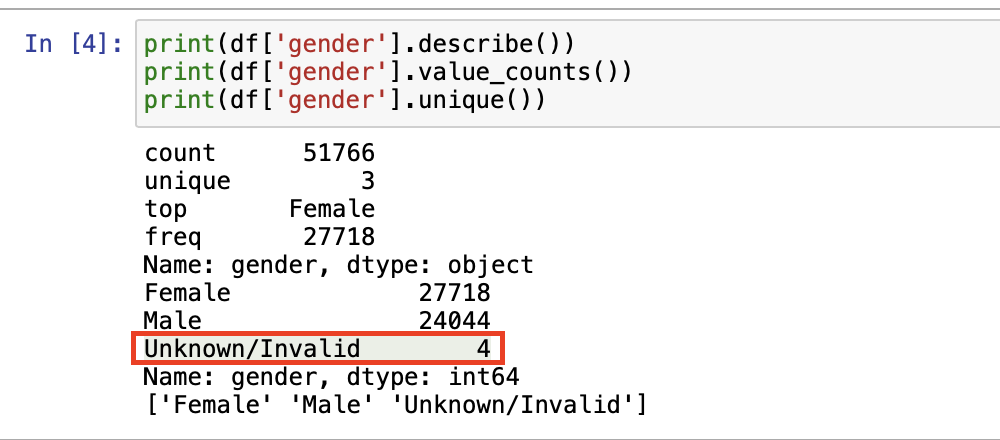
* ‘race’: 1016 missing values found
* ‘age’: 10 missing values found
* ‘chlorpropamide’: 9 missing values found



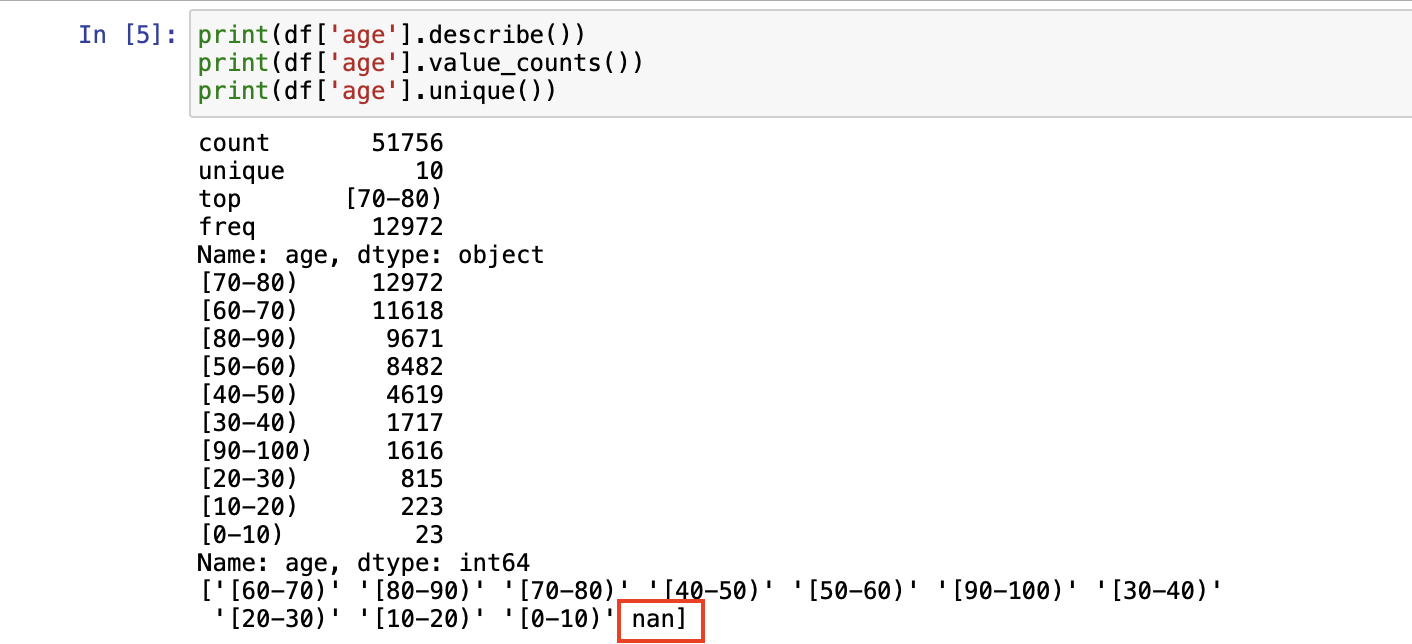


Subsequently, we examined each column using ‘described’, ‘value\_counts’ and ‘unique’ commands to identify any inconsistencies, missing values, or other issues within the data.

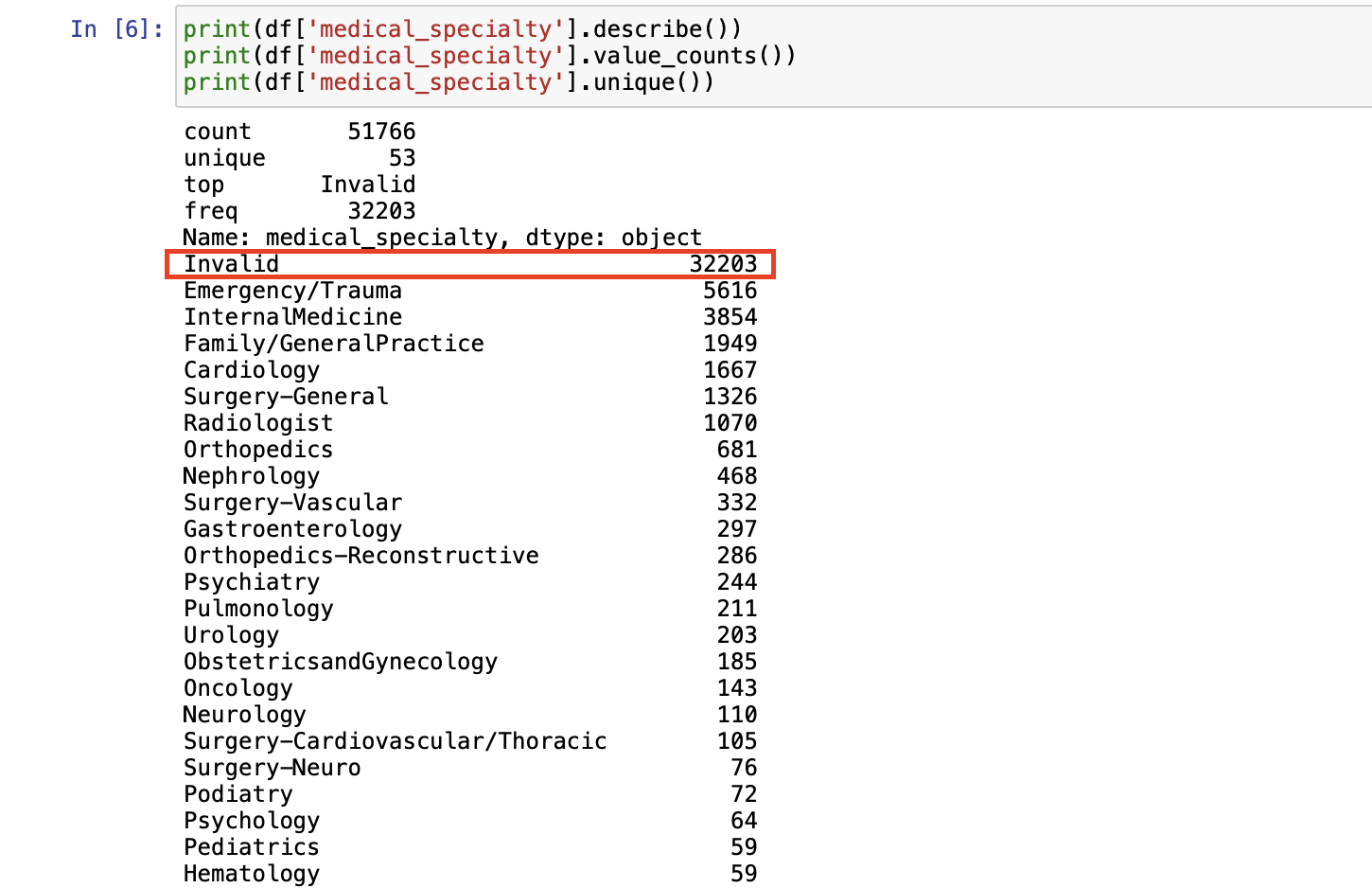
In the ‘gender’ column, there are 4 instances labelled as ‘Unknown/Invalid’. These ‘Unknown/Invalid’ attributes within the dataset might reflect missing or undefined values and are non-informative.



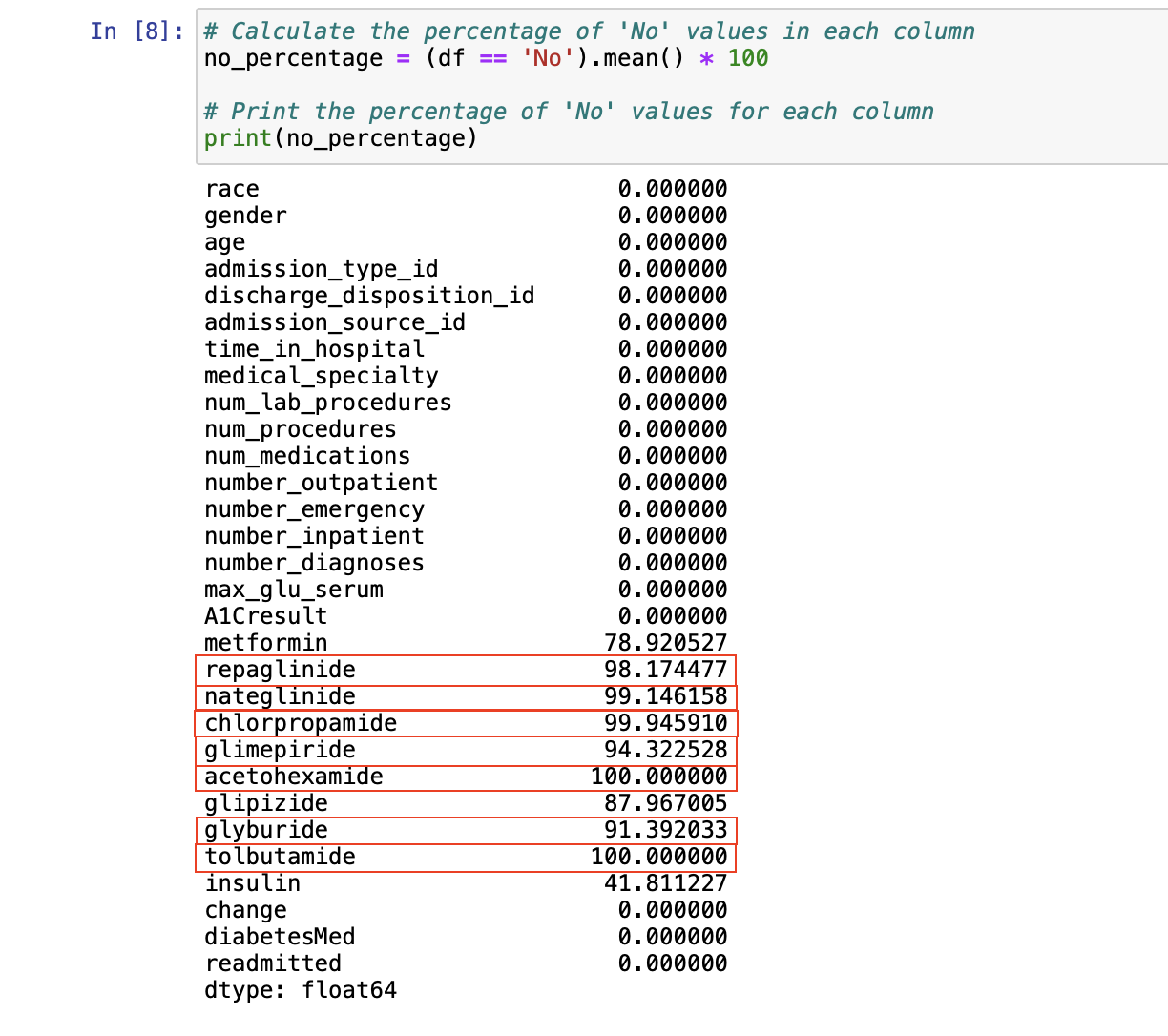
In the ‘age’ column, there is a unique attribute denoted as ‘nan’. This signifies the presence of missing or unclassified data in the dataset.



Also, a significant amount of the data in the ‘medical\_specialty’ column (32,203 entries) is labelled as “Invalid”, which could potentially impact our decision tree modelling.



Additionally, to assess the presence of ‘No’ values within each column of the dataset, we calculated the percentage of ‘No’ values in each column. Our aim is to detect columns in which the portion of the data containing ‘No’ values exceeds 90%. We intend to drop such variables from the dataset as they may not provide significant information.

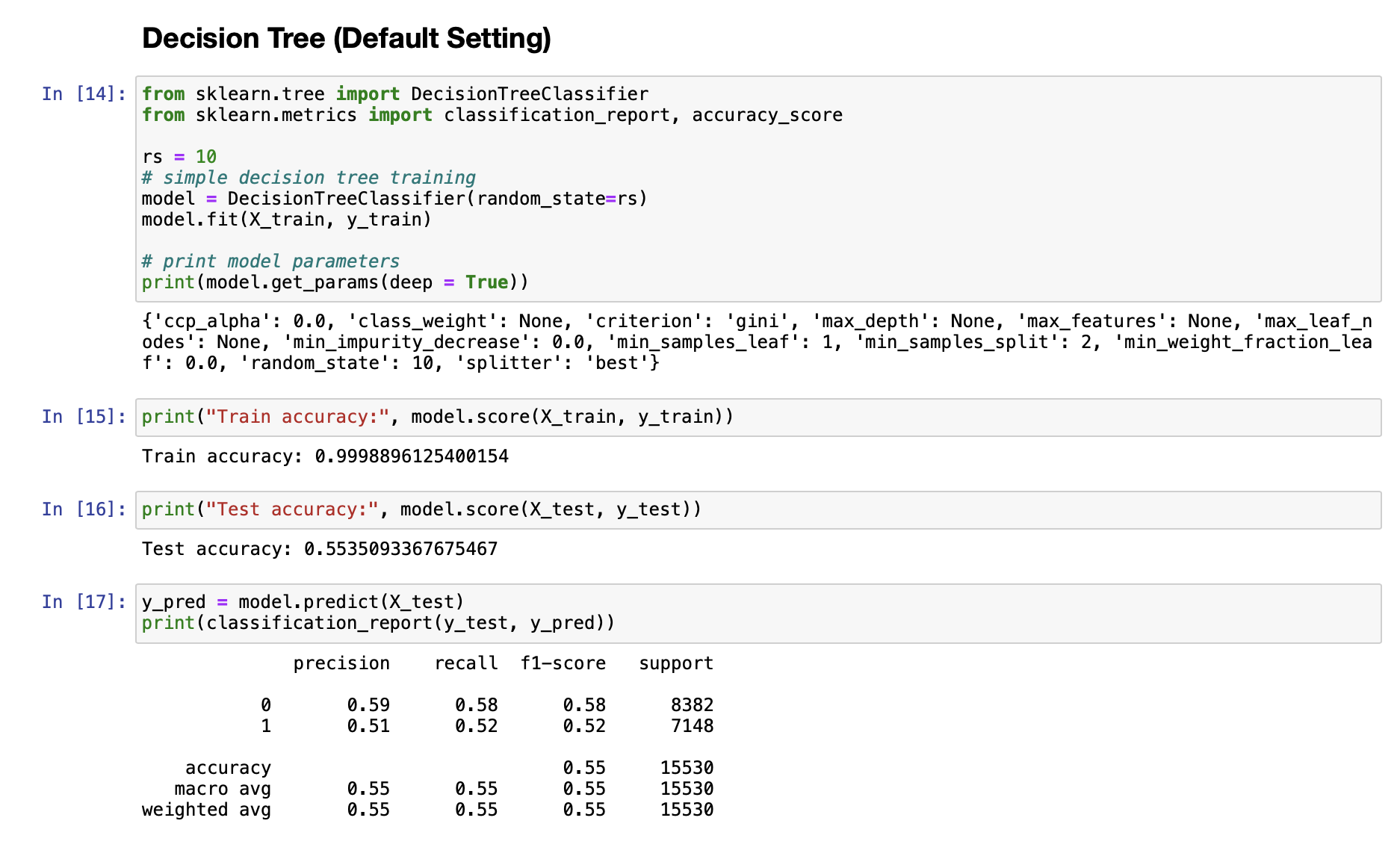


After observing the dataset, we performed the following data preprocessing tasks:

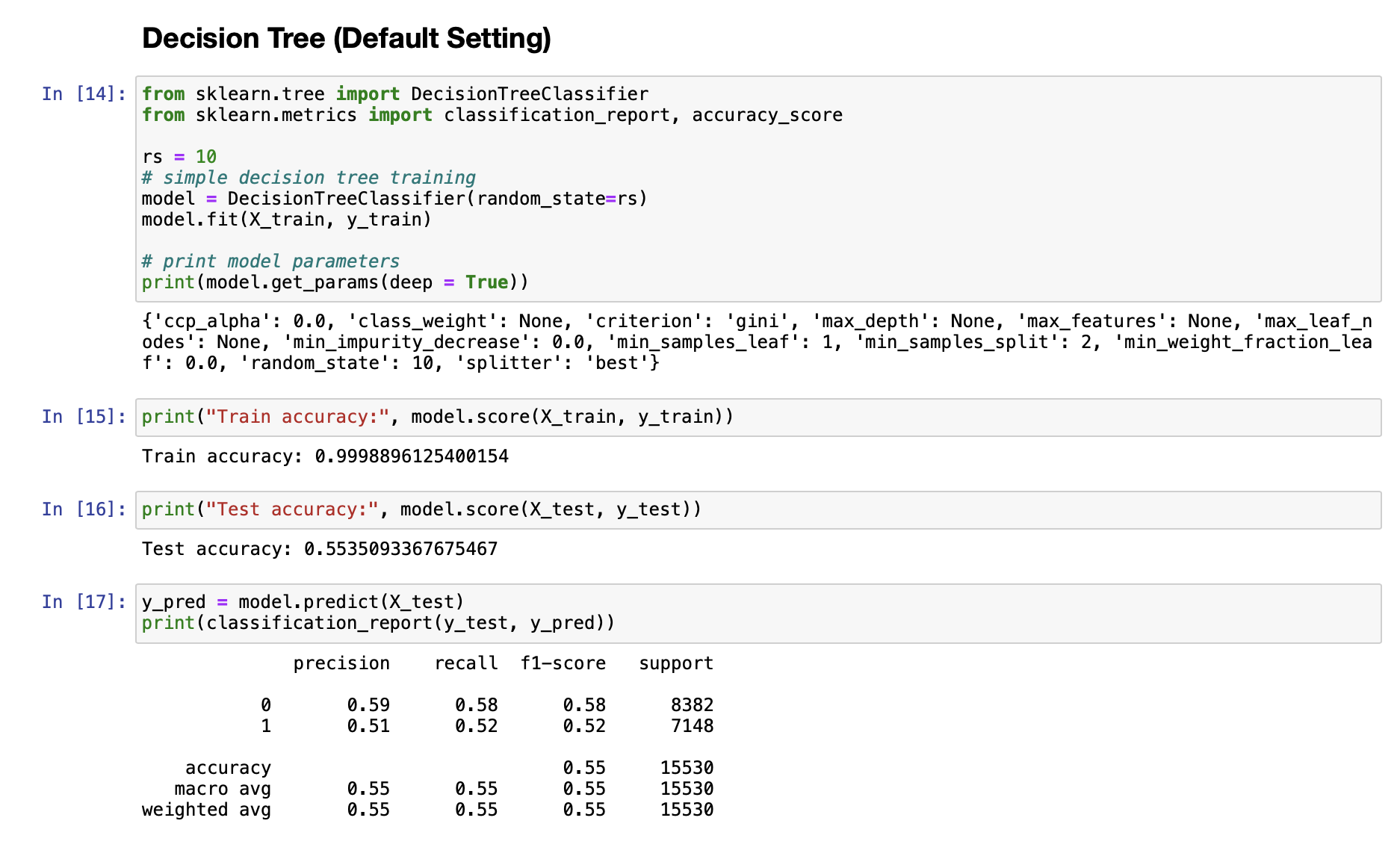
* **Replacing ‘Unknown/Invalid’ Gender Values:** We replaced ‘Unknown/Invalid’ values in the ‘gender’ column with NaN and then filled the NaN values with the most frequent gender found in the dataset (the mode).
* **Imputing Missing Values with Mode:** We imputed the missing values in the ‘race’ and ‘age’ colums by filling them with the respective modes of these columns.
* **Dropping Unused Columns:** We dropped columns that were considered unused or unnecessary for our decision tree modelling. The columns ‘admission\_type\_id’, ‘discharge\_disposition\_id’, and ‘admission\_source\_id’ serve as unique identifiers and do not contain information directly relevant to predictive modeling. Therefore, these columns were dropped. Similarly, the ‘medical\_specialty’ column was removed due to the significant amount of ‘Invalid’ values, which may not provide meaningful information for analysis. Additionally, several medication-related columns, including ‘repaglinide’, ‘nateglinide’, ‘chlorpropamide’, ‘glimepiride’, ‘acetohexamide’, ‘glyburide’, and ‘tolbutamide’ were dropped, as these columns predominantly consist of ‘No’ values (exceeding 90%) and thus may not have made significant contributions to predictive modelling.
* **One-Hot Encoding for Categorical Variables:** To work with the categorical variables effectively, we used one-hot encoding to convert categorical variables into a binary format.
* **Splitting Target and Input Data:** We separated the dataset into two parts: the target variable ‘readmitted’ and the input data (all other columns except ‘readmitted’) since we would like to predict the ‘readmission’ outcome based on the remaining data. In addition, we divided the data into a training set (X\_train, y\_train) and a testing set (X\_test, y\_test), with 70% of the data used for training and 30% for testing. The ‘random\_state’ parameter was set to ensure reproducibility of the split.



**2. Build a decision tree using the default setting. Answer the following:**

****

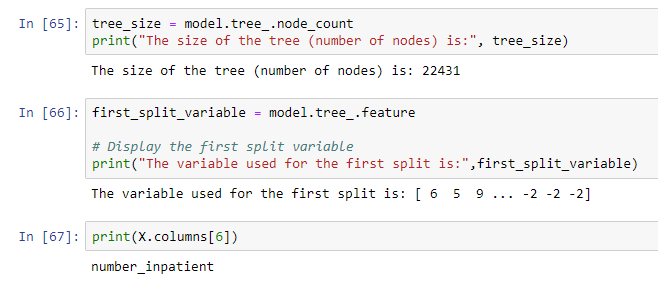
**a. What is the classification accuracy of training and test datasets?**

****

The train accuracy is 99.99% while the test accuracy is 55.35%. This indicates that the maximal tree has a high level of overfitting on the training dataset. Hence, hyperparameters have to be tuned later.

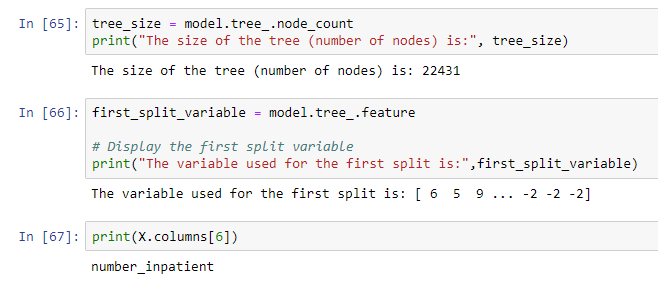
Moreover, classification reports could be made on the test dataset. For the average of both class 0 (no readmission) and class 1 (readmission), the weighted average of precision, recall and f1-score are 55%. Precision indicates that 55% of all positive predictions are correct; Recall indicates that 55% of real positive observations are correct; F1-score indicates the mean of precision and recall is 55%. It can be seen that Class 0 has higher accuracy in all positive prediction and real positive observation as compared to class 1.

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

****

The default decision tree has the number of nodes of 22431, which is too huge to be visualised.

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

****

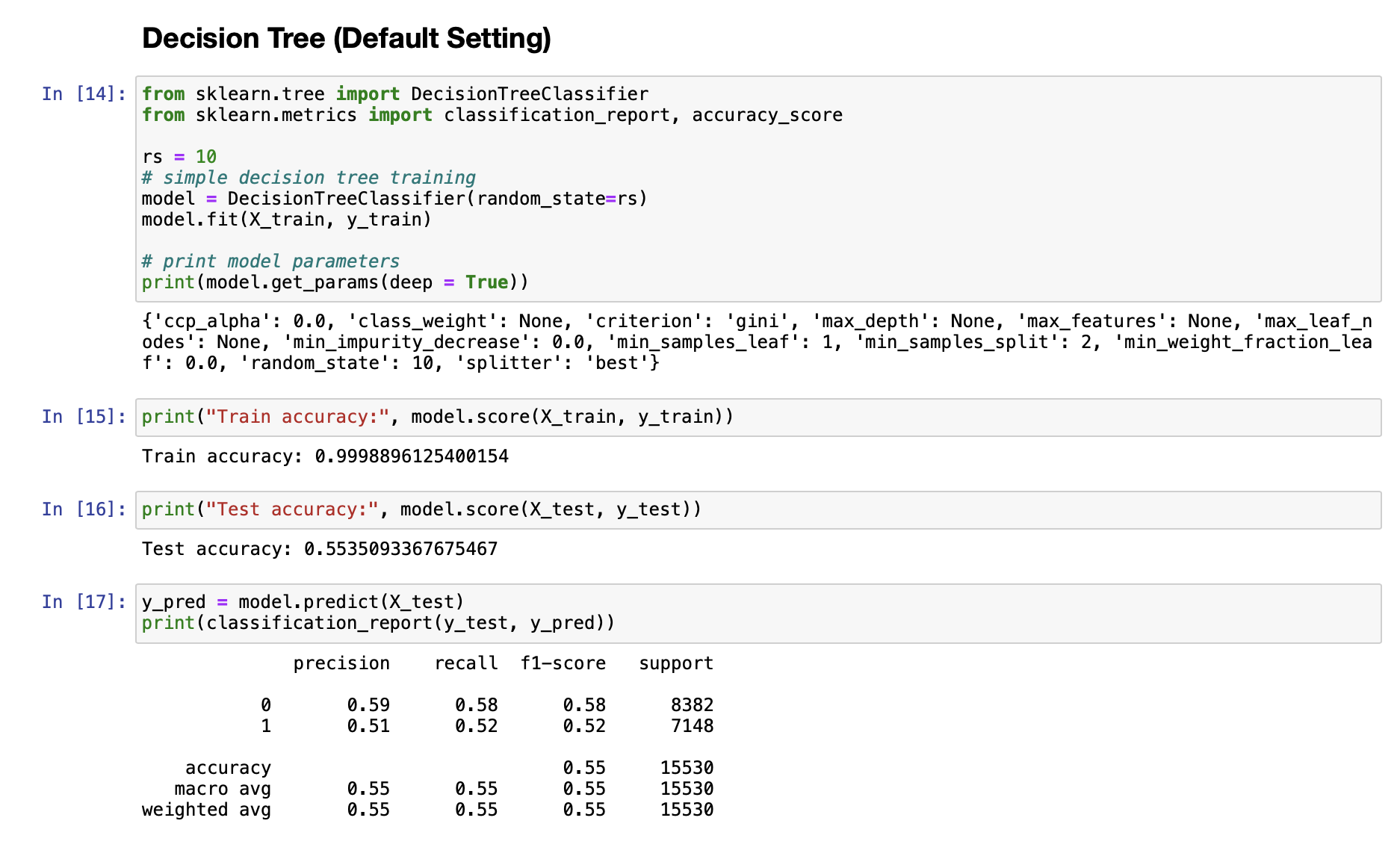
The variable used for the first split is the variable of column index 6, in other words it reflects the number\_inpatient. This indicates that number\_inpatient has the best first split as our hyperparameter splitter requires the best split on each node.

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



The top 5 important features are num\_lab\_procedure, num\_medications, time\_in\_hospital, number\_inpatent, num\_procedures.

**e. What parameters have been used in building the tree? Detail them.**

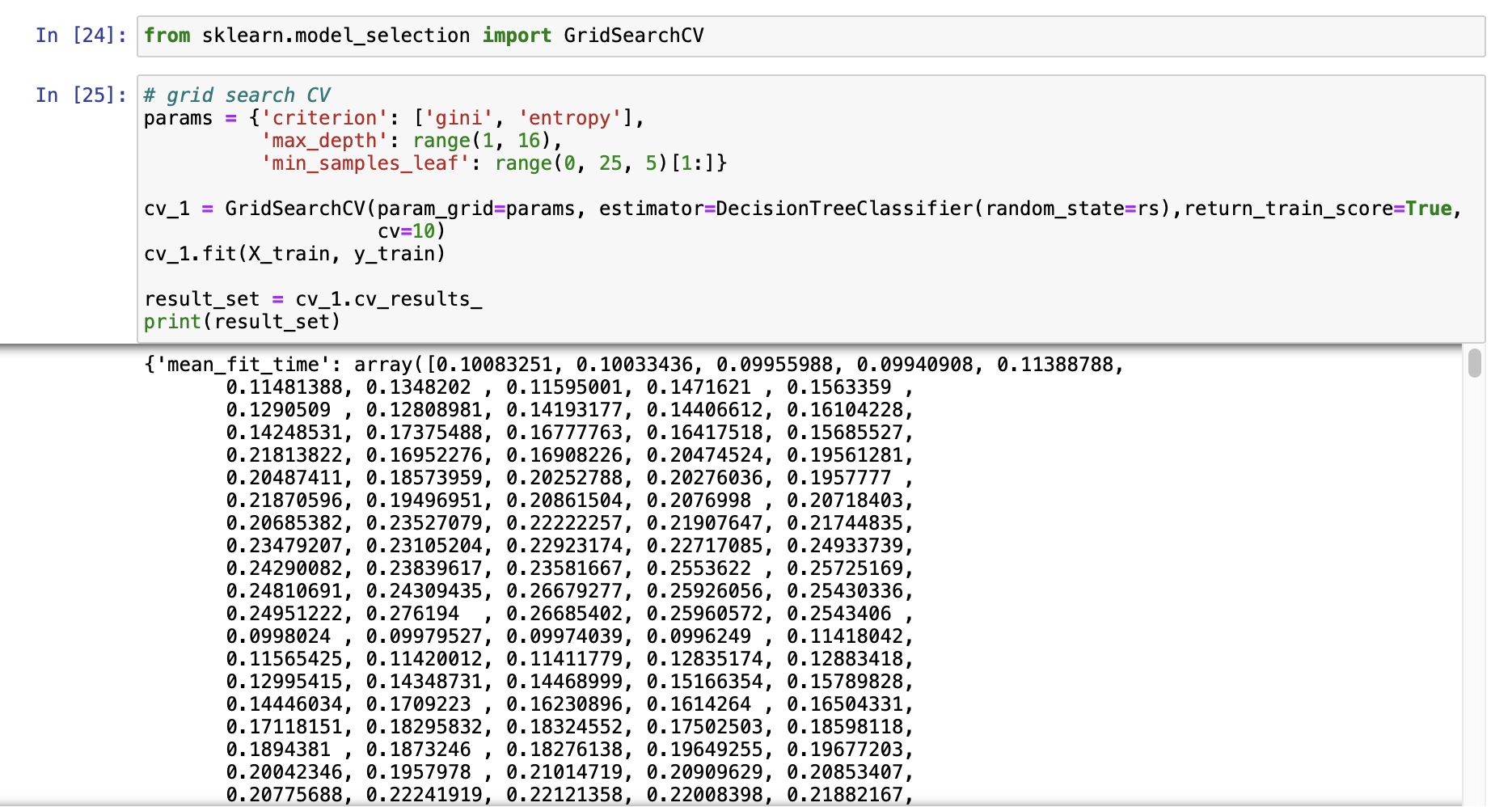
****

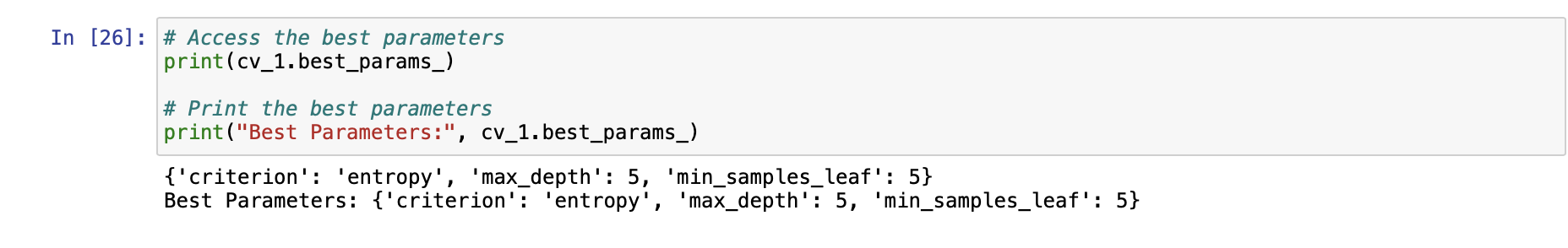
The hyperparameters that are specified before training:

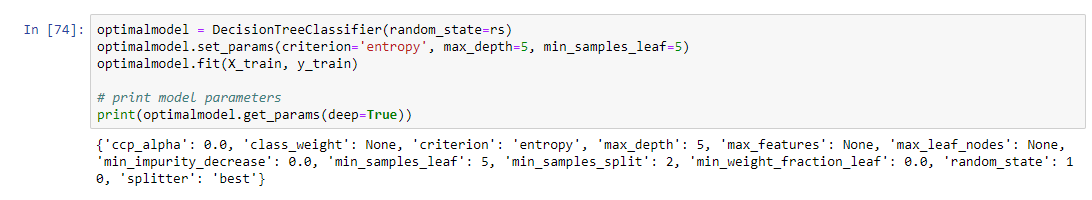
* Criterion: Gini
* Max\_depth: None
* min\_samples\_leaf: 1
* min\_samples\_split: 2
* splitter: best (choosing the best split on each node)
* random\_state: 10

The following parameters are learned during training:

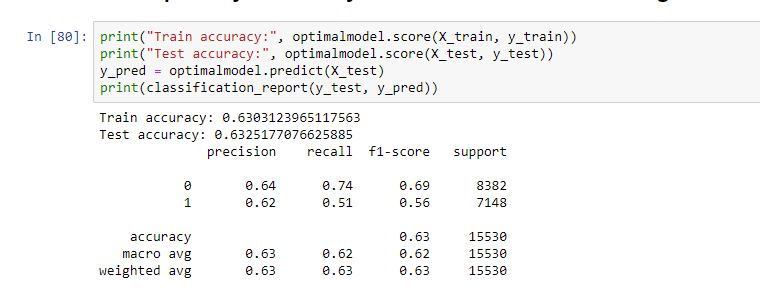
* ccp\_alpha: 0
* class\_weight: None
* max\_features: None
* max\_leaf\_nodes: None.
* min\_impurity\_decrease: 0
* min\_weight\_fraction\_leaf: 0

**3. Build another decision tree tuned with GridSearchCV. Answer the following: **





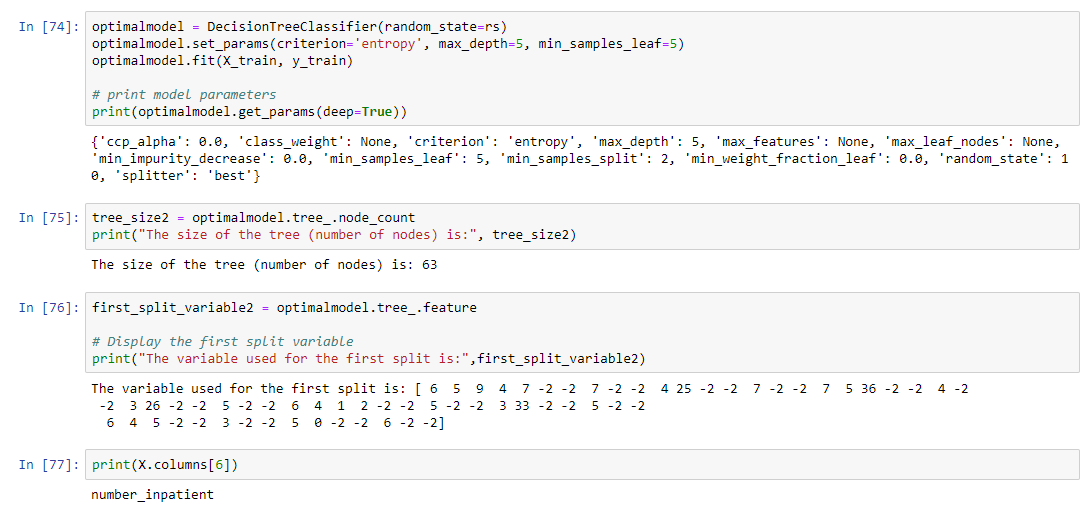
**a. What is the classification accuracy of training and test datasets?**

****

The train accuracy is 63.03% while the test accuracy is 63.25%. This indicates that no overfitting has occurred for the optimal model.

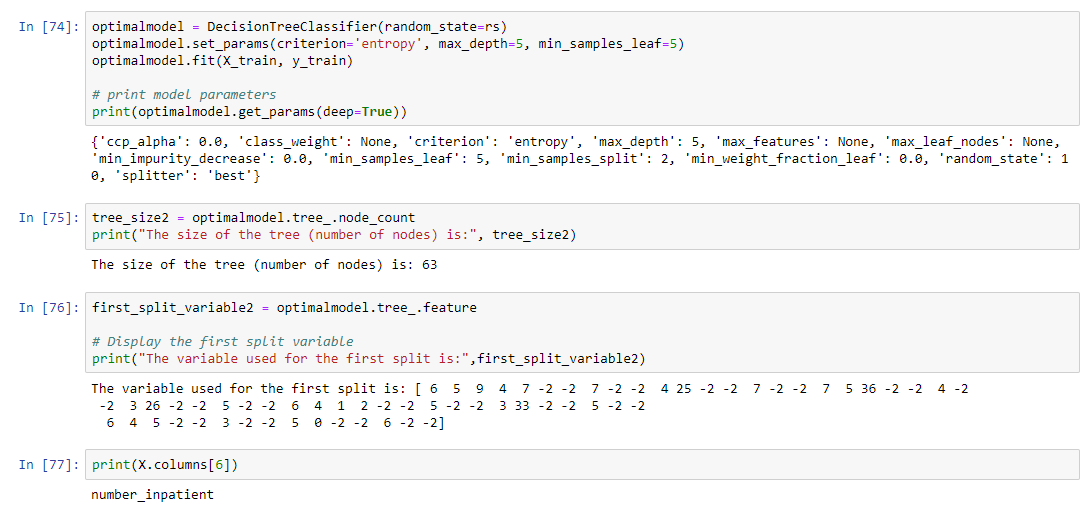
For the classification report, weighted average of precision, recall and f1-score are 63%. Precision indicates that 63% of all positive predictions are correct; Recall indicates that 63% of real positive observations are correct; F1-score indicates the mean of precision and recall is 63%. Moreover, similar to the default model, it can be seen that Class 0 has higher accuracy in all positive predictions and real positive observation as compared to class 1.

**b. What is the size of the tree (i.e. number of nodes and rules)?**



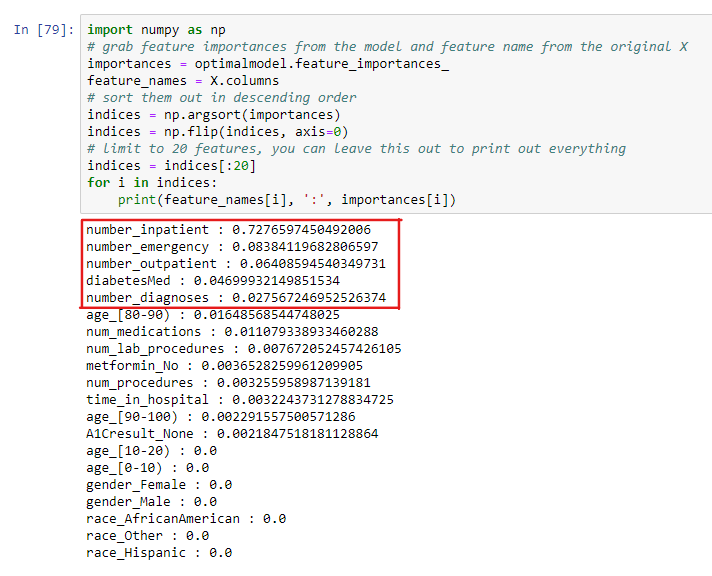
The optimal model has the number of nodes of 63, which is able to be visualised and interpreted in the 5th Question.

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



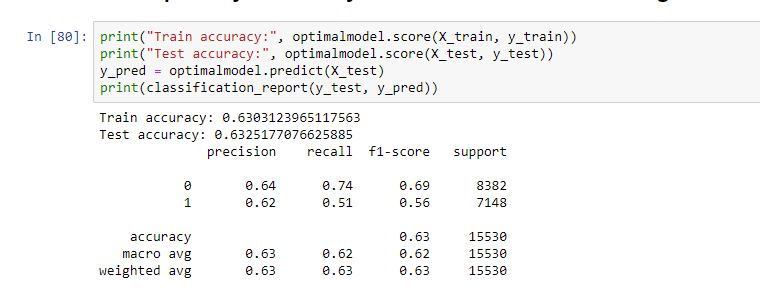
Similar to the default model, the variable used for the first split is the variable of column index 6 which is number\_inpatient. This indicates that number\_inpatient has the best first split as our hyperparameter splitter requires the best split on each node.

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

****

The top 5 important features are number\_inpatient, number\_emergency, number\_outpatient, diabetesMed, number\_diagnoses.

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

****

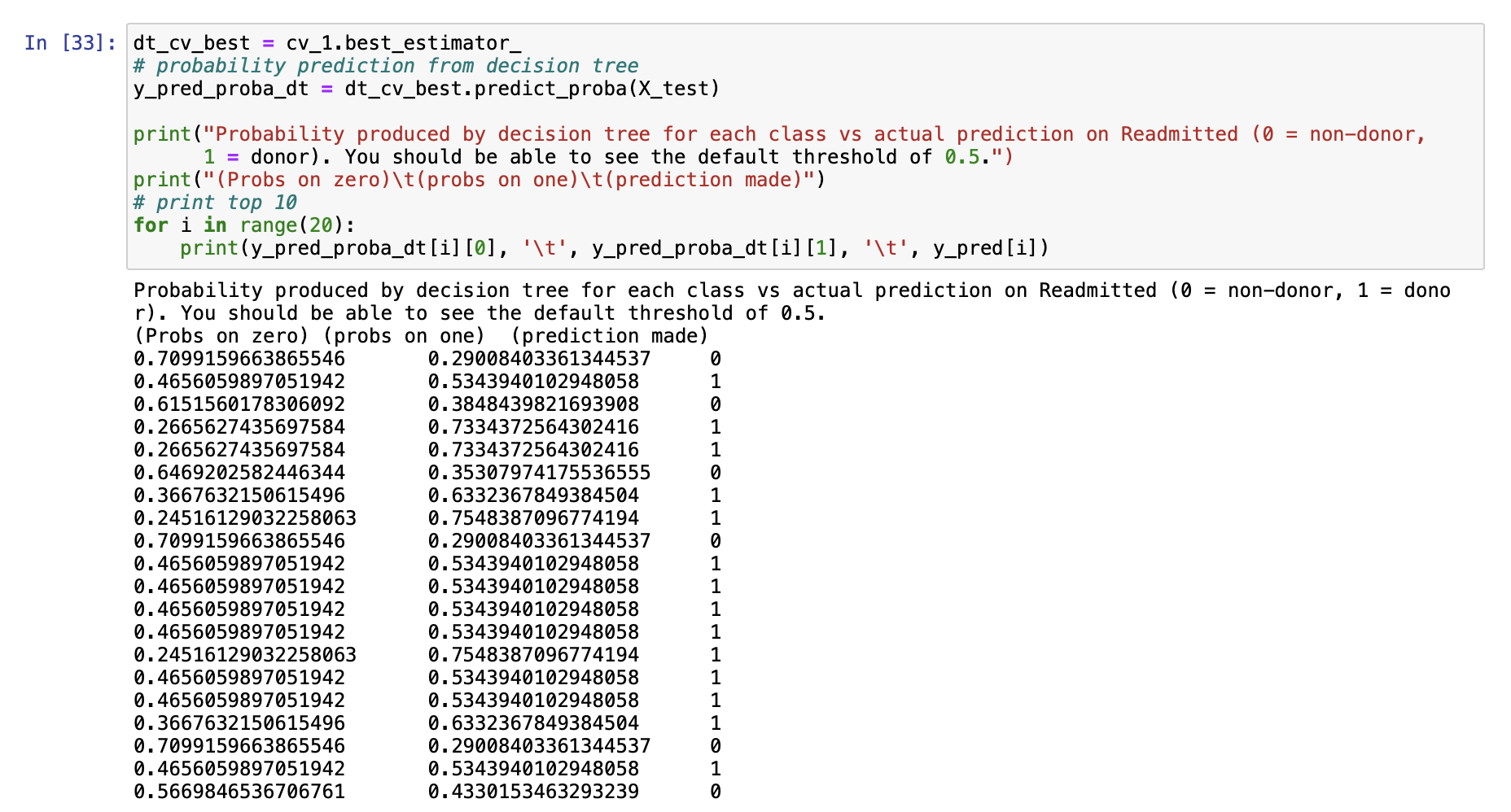
There is no sign of model overfitting as the test accuracy is higher than the train accuracy.

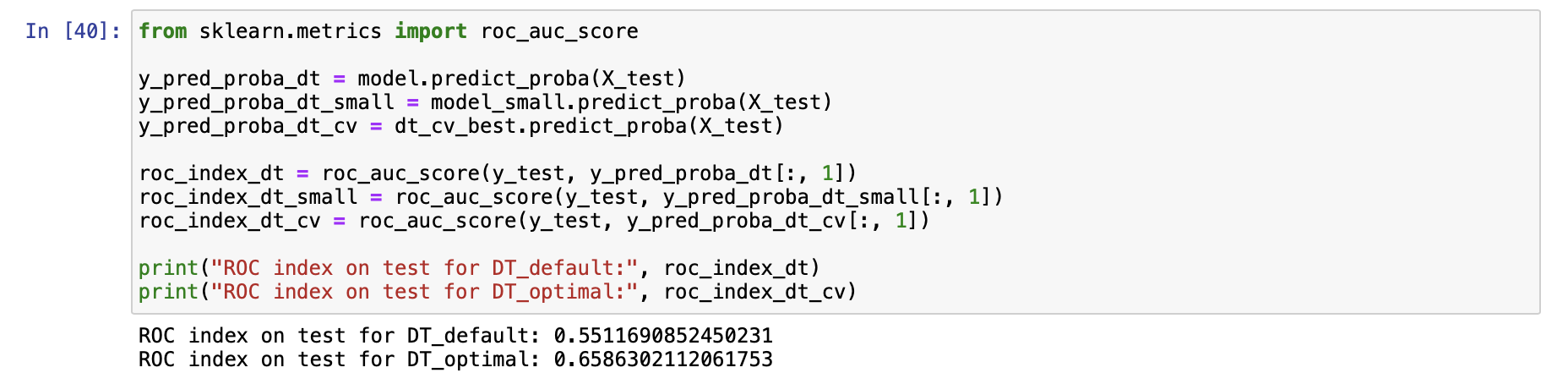
**4. What differences do you observe between these two decision tree models (with and without fine-tuning)? How do they compare performance-wise?**

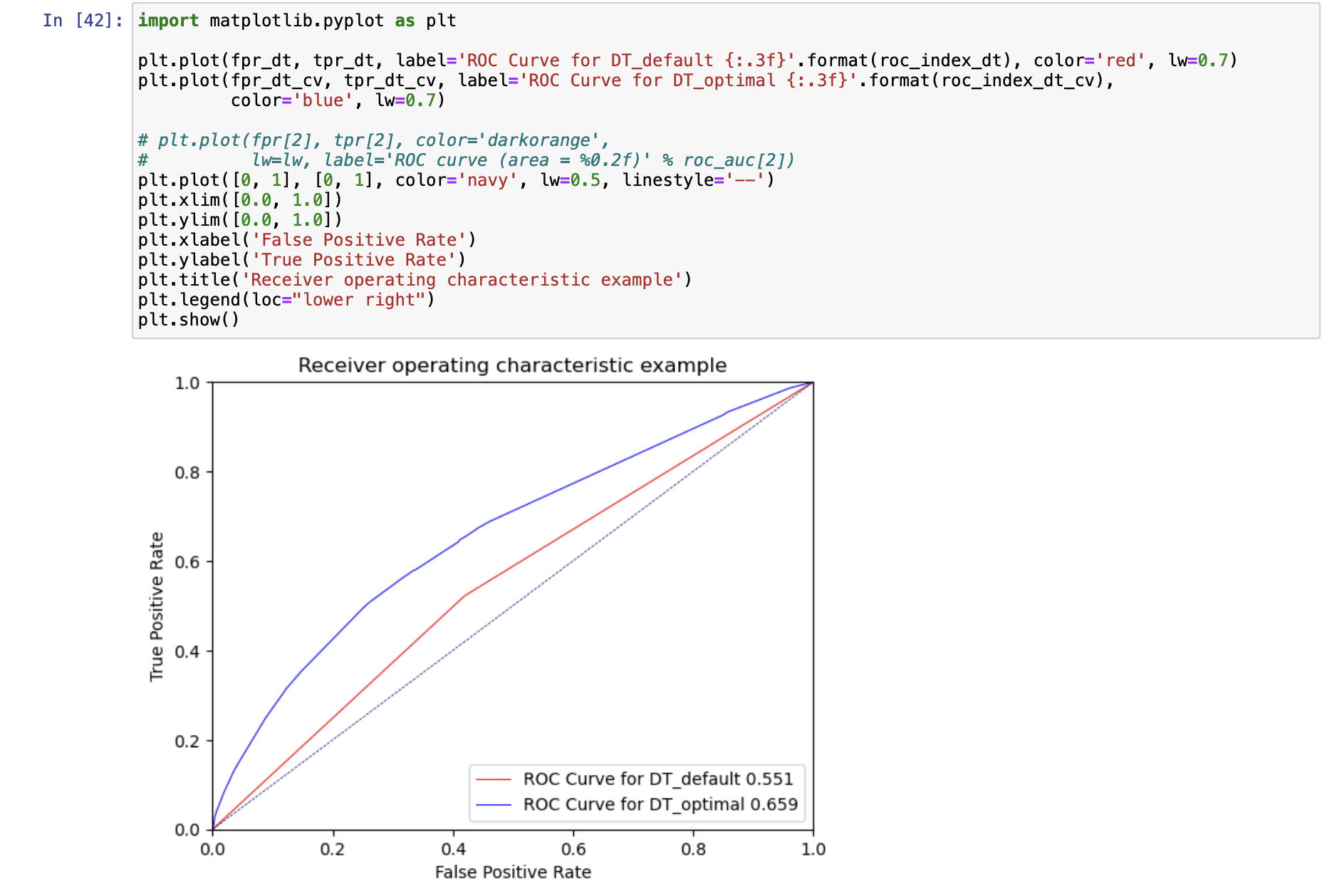
|  |  |
| --- | --- |
| **Default Model** | **Optimal Model** |
|  |  |

The default and the optimised decision tree models have different performance metrics. The precision, recall, and F1-score for both classes are distinct. The training accuracy of the default model is quite high (nearing 1). This implies that the model fits the training data almost perfectly. However, a high training accuracy may indicate overfitting. In contrast, the optimal model displays a lower training accuracy, potentially due to its lower proneness to overfitting. In terms of test accuracy, the optimal model outperforms the default model as it achieves a higher accuracy of around 0.63. The macro and weighted averages for precision, recall, and F1-score of the default and optimal models also differ. The comparison of the default and optimised decision tree models' performances reveals that the optimization approach did result in model performance improvements. The optimal model has a higher precision for both classes 0 and 1 compared to the default model. This suggests that after fine-tuning, the optimal model’s capacity for accurately predicting both positive and negative has increased. The optimal model has a higher recall for class 0. However, the recall of the default model is slightly better than the optional model for class 1. Nonetheless, the optimal model excels with higher F1 score for both class 0 and class 1. This demonstrates that after fine-tuning, the mean of precision and recall has improved.

**Produce the ROC curve for both DTs. Explain why those changes may have happened.**







DT\_default and DT\_optimal both have ROC values of 0.551 and 0.659, respectively. This indicates that the DT\_optimal model outperforms the DT\_default model by a considerable amount. The DT\_optimal model's higher ROC value suggests that it can distinguish between positive and negative classes more effectively than the DT\_default model. As a result, the DT\_optimal model performs better at classifying the classes than the DT\_default model, according to the ROC values.

The increase of the ROC value from 0.551 to 0.659 also indicates a significant improvement in the performance of the model. An example for this improvement is due to reduced overfitting. By achieving the ideal balance between both variance and bias, the DT\_optimal model may have been developed to reduce overfitting. When a model overfits and begins to show noise in the training data rather than the predicted pattern, this is known as overfitting. It's likely that by fine-tuning the model, the complexity was decreased, resulting in higher performance on data that was not observed. Data preprocessing techniques were done to contribute a more accurate and reliable model.

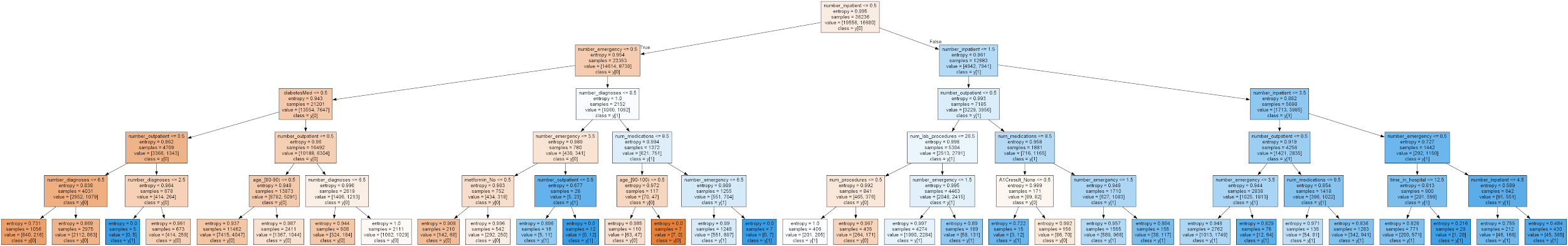
**5. From the better model, can you identify which patients could potentially be "readmitted"? Can you provide the general characteristics of those patients?**

To identify the potential readmission, visualisation of the decision tree is made.

****

The optimal model is as pasted in the [next page.](https://drive.google.com/file/d/1Fs-8opwFIAaHmsMrCFN1W0mU0DlBfD2U/view?usp=share_link)

Referring to the decision tree, we can identify the readmitted patient by going through the tree nodes. For instance, a patient with the num\_inpatient of more than 1.5, will have the possibility of readmission while a patient with num\_inpatient less than 0.5, number\_emergency less than 0.5, diabetes\_med less than 0.5, number\_outpatient less than 0.5, number\_diagnoses less than 6.5 will not be readmitted. In the optimal model’s visualisation tree, input variables include number\_inpatient, number\_outpatient, number\_emergency, number\_diagnoses, num\_lab\_procedures, num\_procedures, num\_medications, time\_in\_hospital, metformin (“No”), age, diabetesMed, A1C\_result (“No”). The general characteristic can be shown - the higher the number\_inpatient, number\_outpatient, number\_emergency, number\_diagnoses, num\_medications, age, along with diabetesMed being “No”, metformin not “No” and A1C\_result being “No” will result in the possibility of readmission.

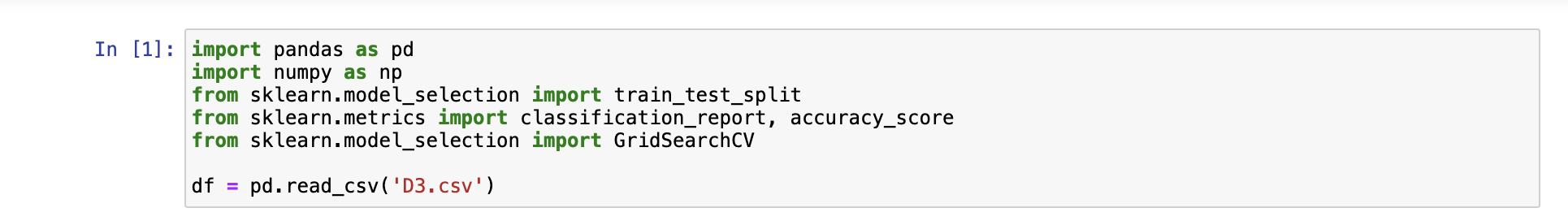
**Decision Tree**

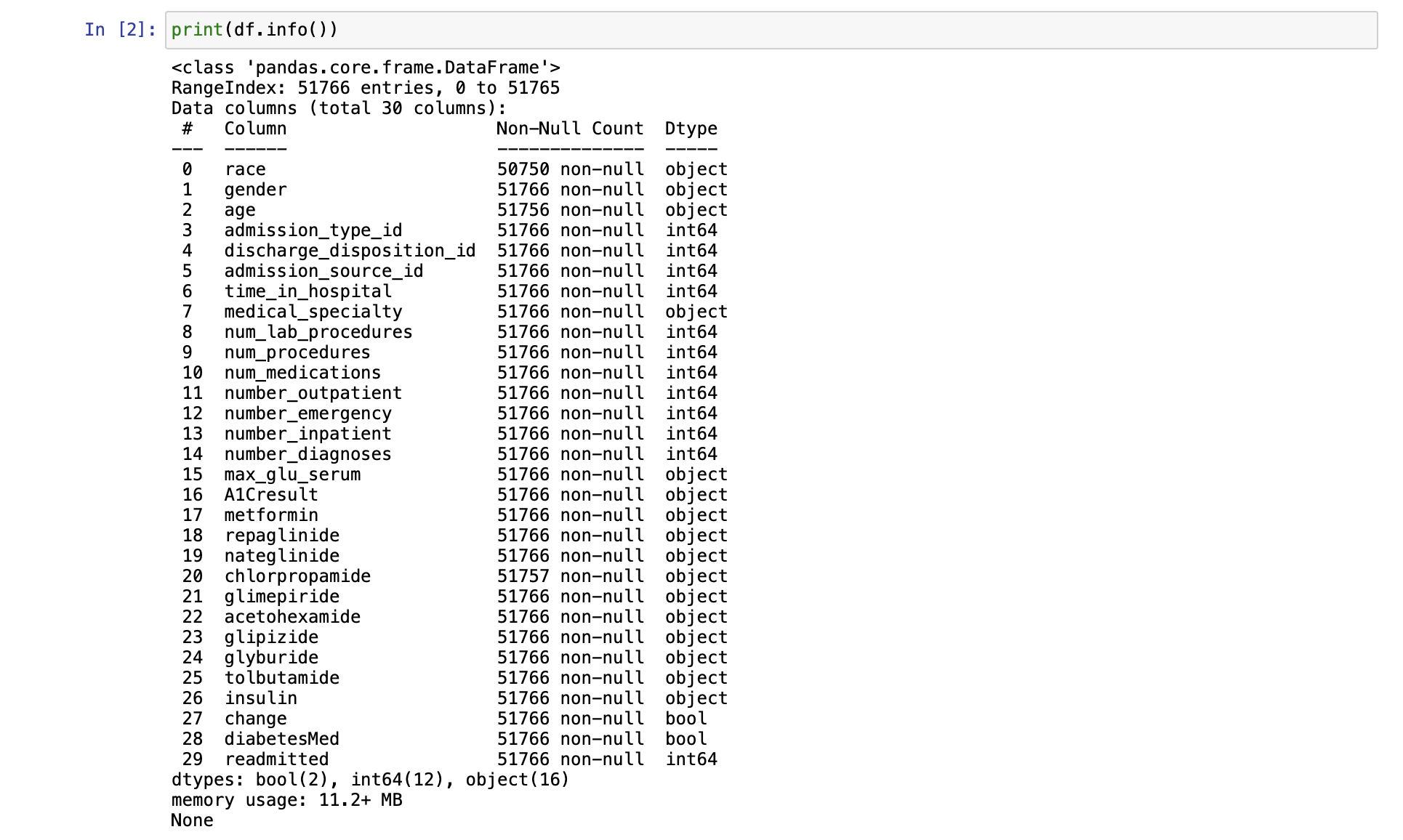
Kindly click the link here for a clearer visualisation: [optimalmodel\_viz.png](https://drive.google.com/file/d/1Fs-8opwFIAaHmsMrCFN1W0mU0DlBfD2U/view?usp=share_link)

**Case 4: Predictive modelling using Regression**

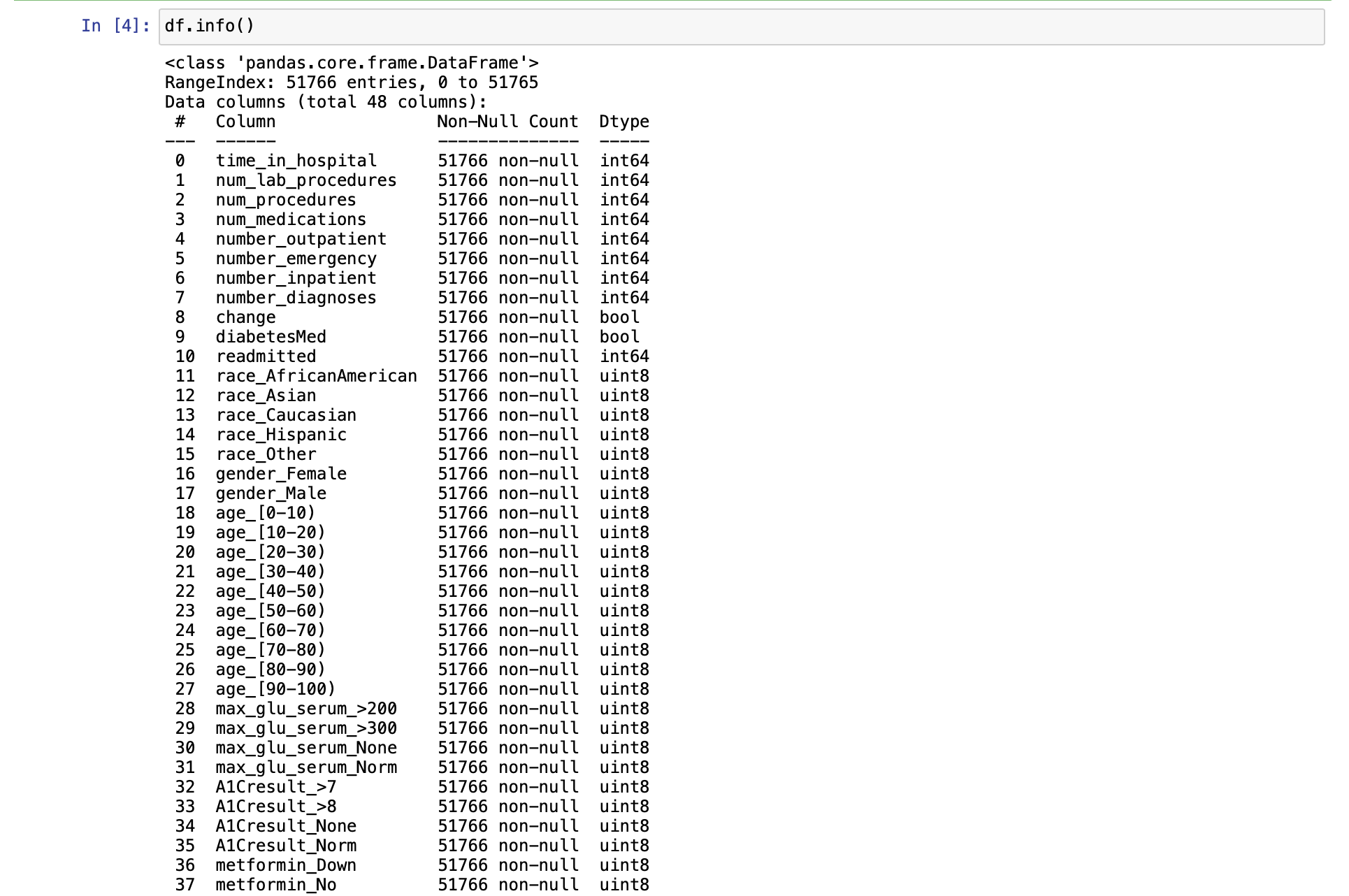
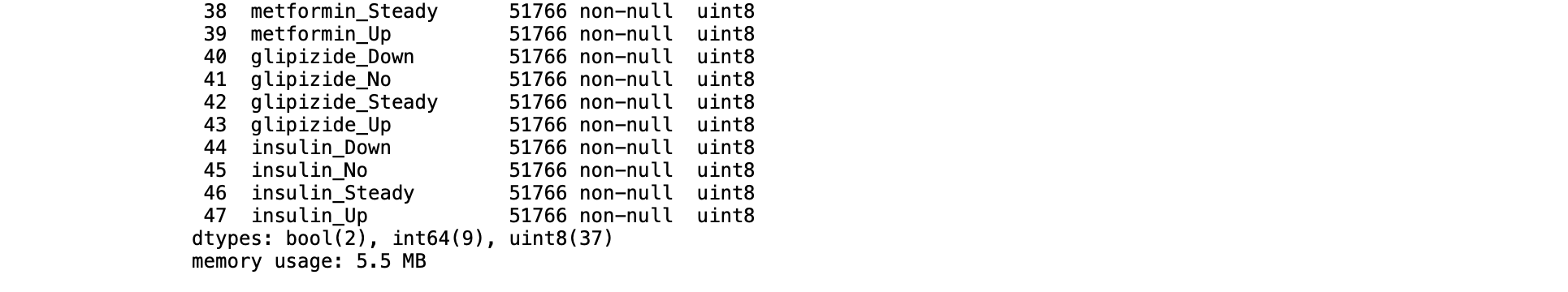
**1. What pre-processing was required on the dataset before regression modeling? What distribution split between training and test datasets have you used?**

After importing the data, similar pre-processing from the decision tree is applied for the regression preparation. The Unknown/Invalid values in gender are replaced ith NaN. Then, the missing values in gender, race, age are replaced with mode. As shown previously, columns with highly skewed distribution that contribute no information are dropped. The categorical data is then encoded, and inputs are converted to a numpy matrix. The Distribution split used is 70% of training data, 30% of test data.







**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 following:**

**a. Explain why you chose that model.**

Out of the two models, we opted for the default model, as it has higher test accuracy of 62.87% as compared to the GridSearchCV, with test accuracy of 62.86%.

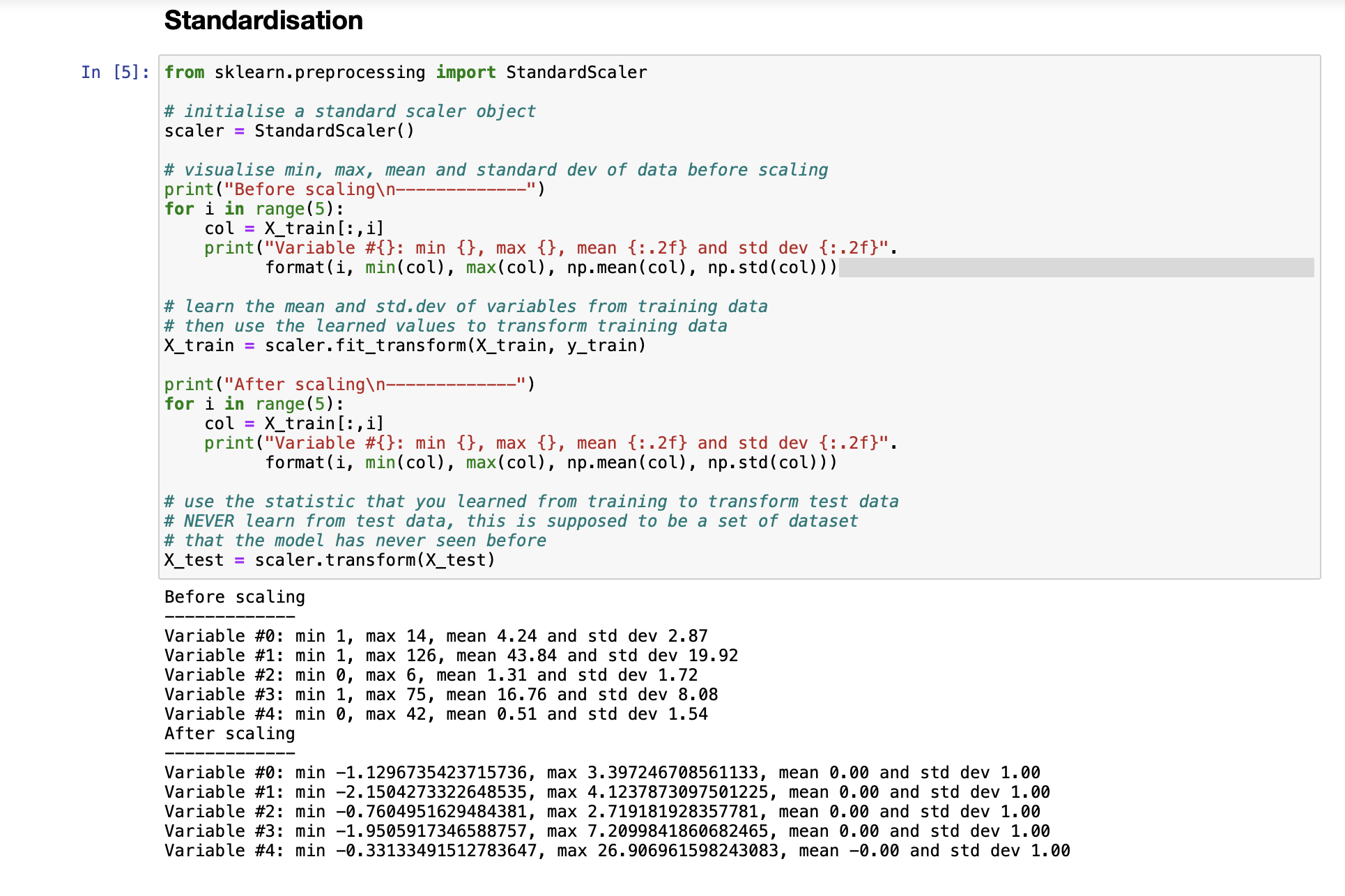
|  |  |
| --- | --- |
| **DefaultModel** | **GridSearchCV** |
|  |  |

**b. Name the regression function used.**

The regression function used is Logistic Regression, as our target variable - readmission is a categorical data type.

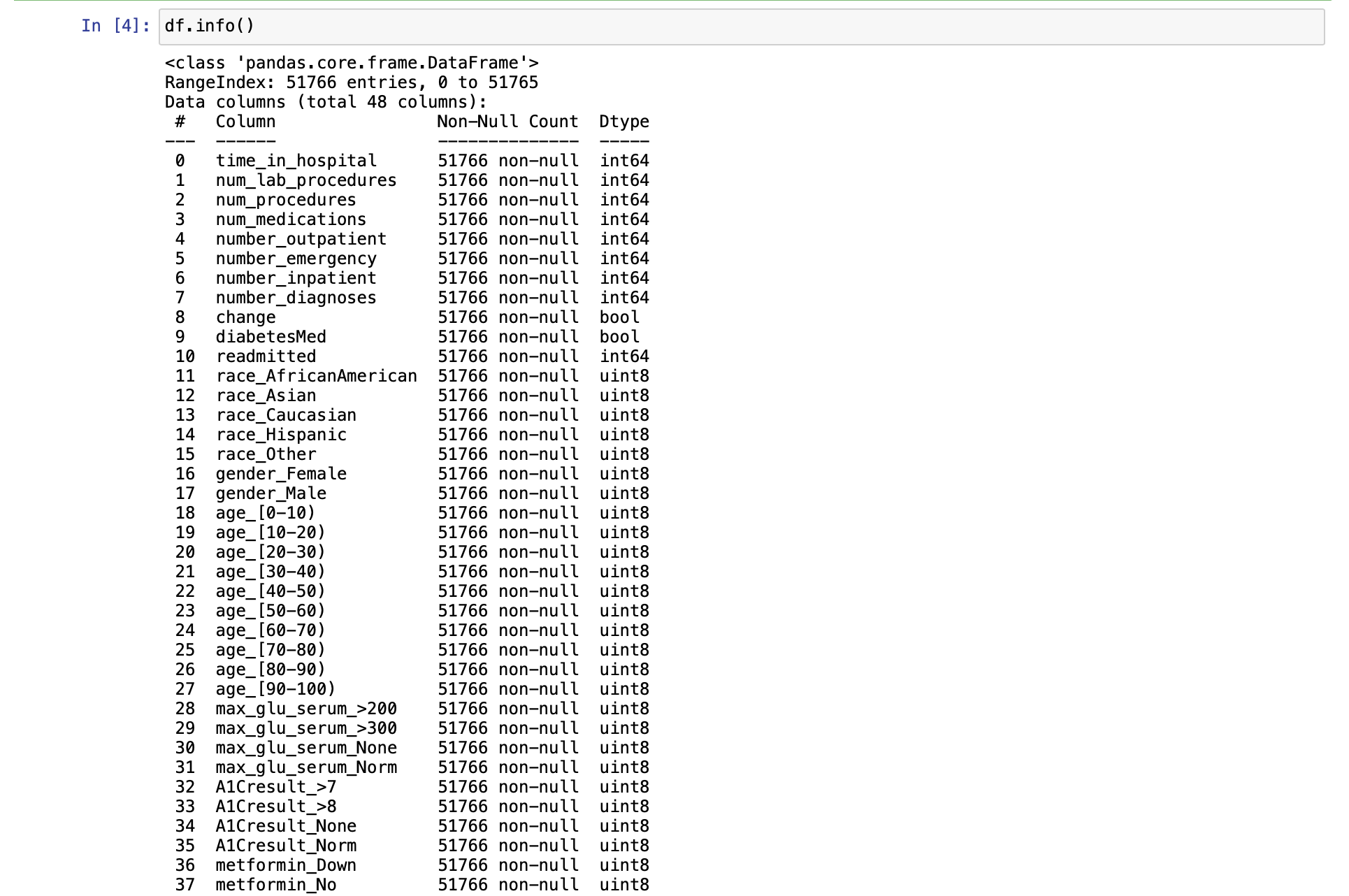
**c. Did you apply standardization of variables? Why would you standardize the variables for regression mining?**

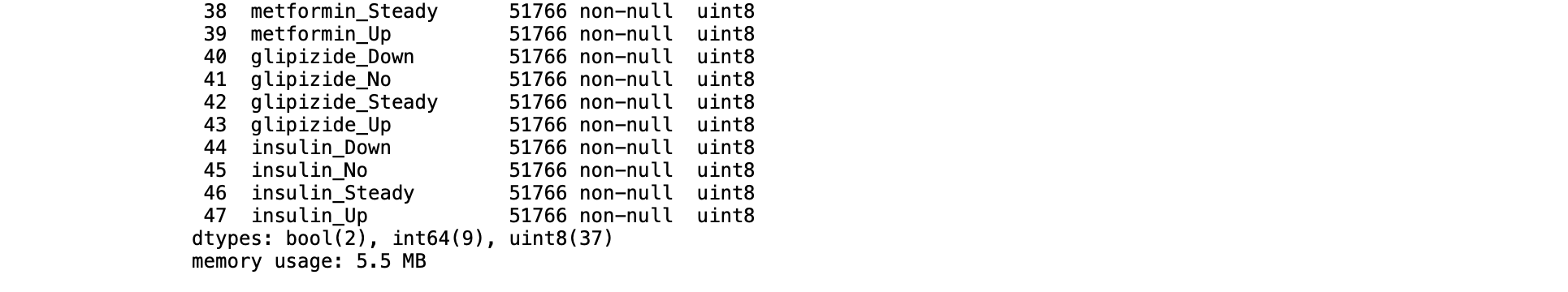
Yes, we applied variable standardisation in our analysis to handle the challenge involving input variables with varying scales as differences in the scale of input variables tend to be sensitive to regression models. Also, standardisation brings additional benefits in improving the training process of regression models. It helps improve the performance of gradient descent, an optimization algorithm used in training regression models.



**d. Report the variables included in the regression model.**

The following variables are included in the regression model:



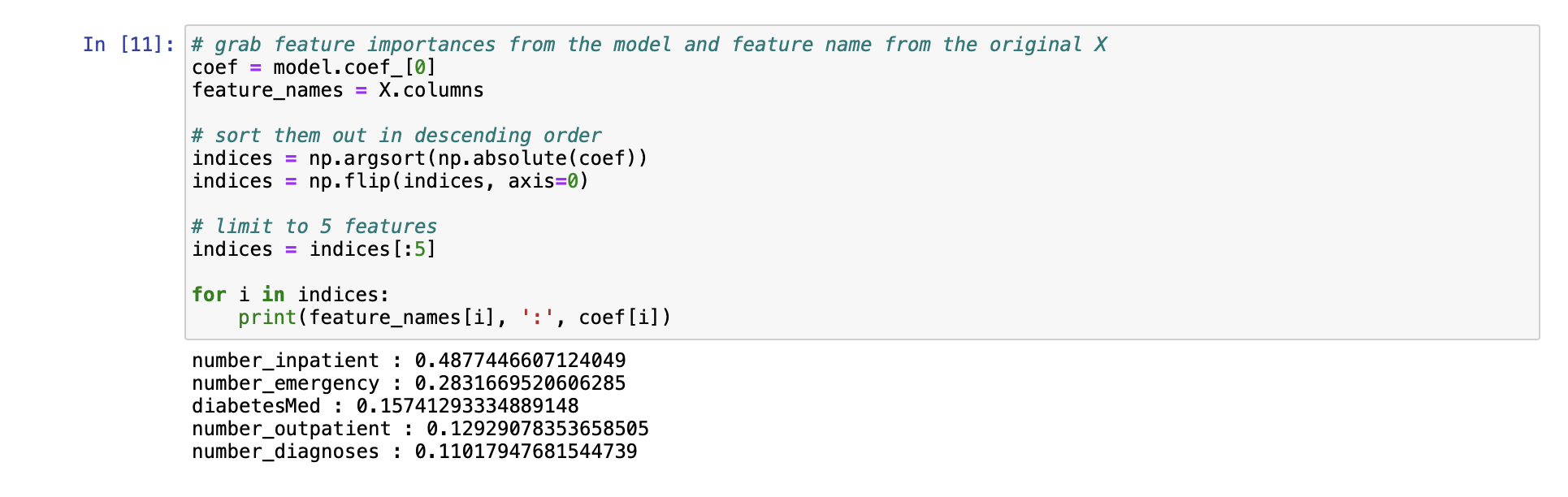


**e. Report the top-5 important variables (in the order) in the model.**

Based on the coefficient values, the top five important variables in the model are as follows:

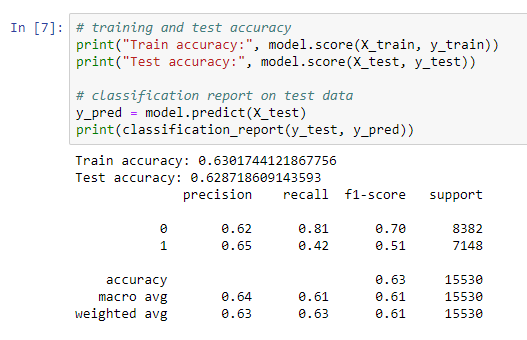
* number\_inpatient
* number\_emergency
* diabetesMed
* number\_outpatient
* number\_diagnoses

These variables have the highest absolute coefficient values, suggesting that they have the most influence on the logistic regression model’s prediction.



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

The default model has a training accuracy of around 63.02% and a test accuracy of around 62.86%. Precision, recall, and F1-scores were computed for two classes (0 and 1), Class 0 showed higher recall (81%) but lower precision (42%) with an F1-score of 0.70. In contrast, class 1 had lower recall (42%) but higher precision (65%), yielding an F1-score of 0.51.

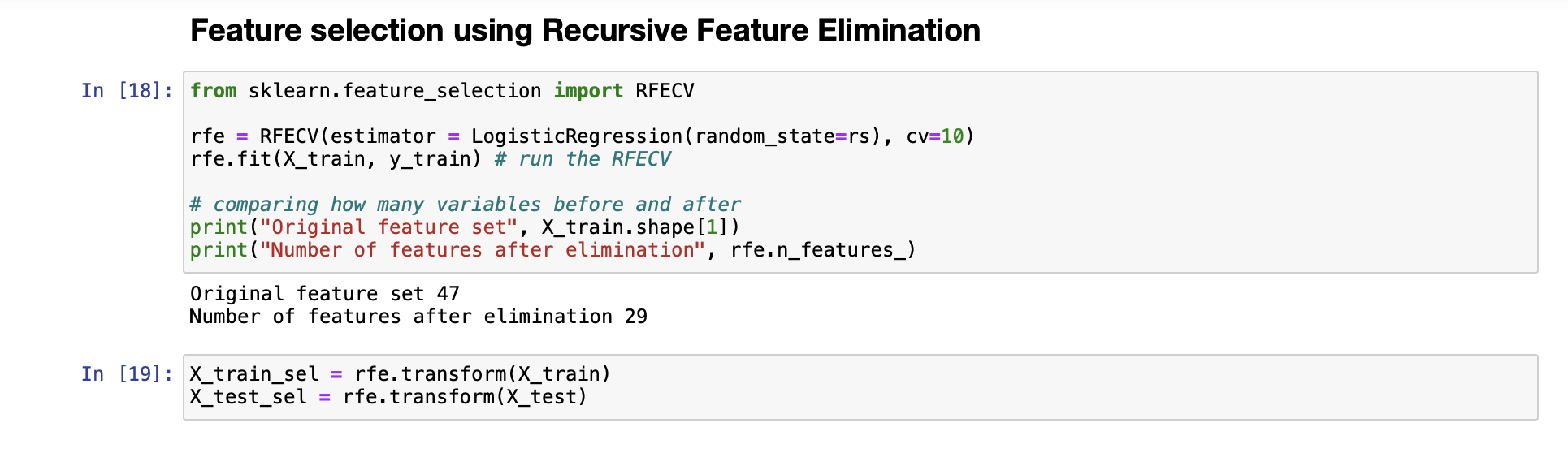


**g. Report any sign of overfitting in this model.**

Though train accuracy is slightly higher than test accuracy, slight overfitting is observed but it can be ignored due to the small difference.

**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 following:**

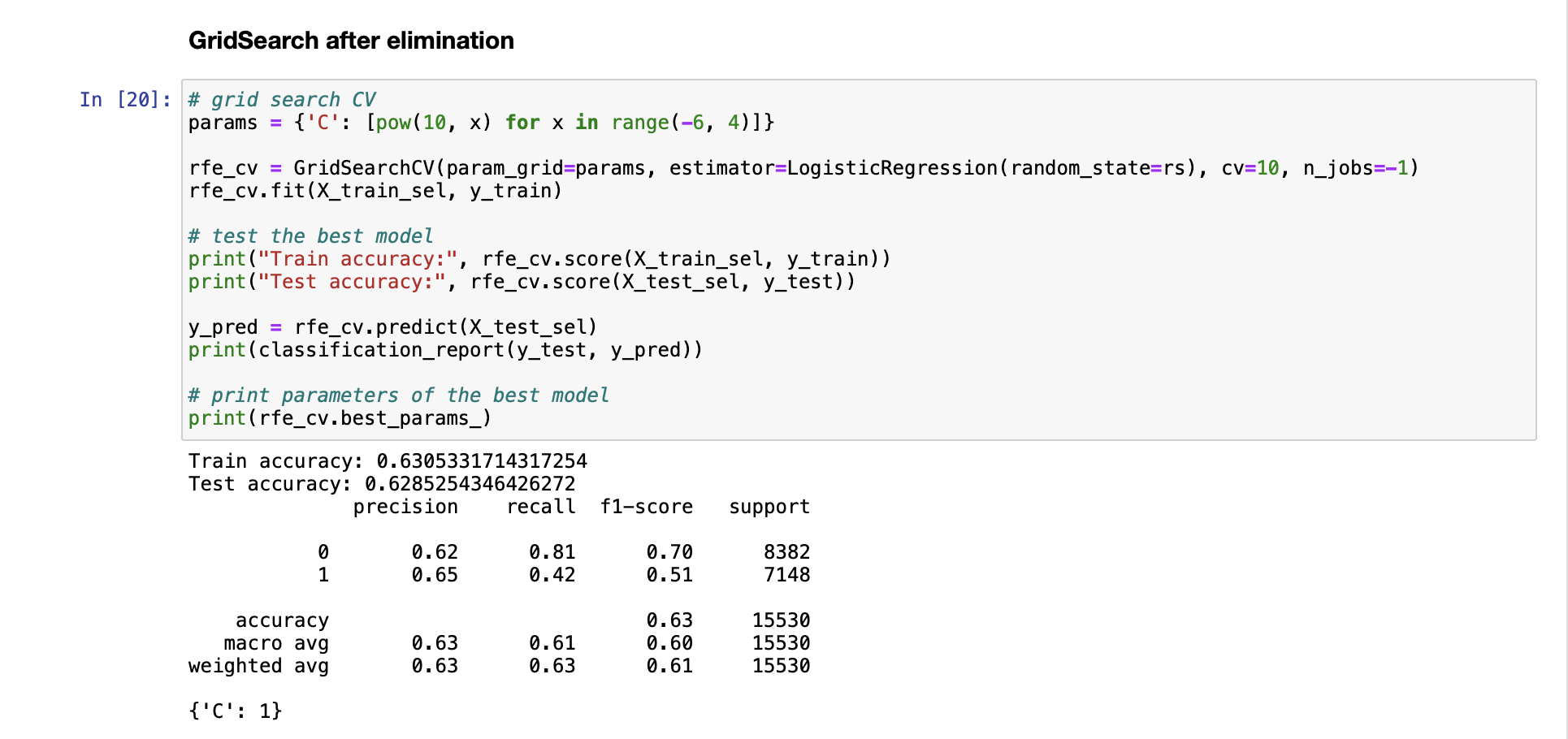
**a. Was dimensionality reduction found useful in identifying a good feature set for building an accurate model?**



|  |  |
| --- | --- |
| **Model Types** | **Test and Training Accuracy** |
| **Default** |  |
| **GridSearchCV** |  |
| **GridSearchCV + Recursive Feature Elimination** |  |

After tuning the model with GridSearchCV and with the addition of Recursive Feature Elimination, the model’s accuracy did not improve. In addition, the train and test accuracy, recall precision and F1-score remained almost similar in comparison with the default model. Though dimensionality reduction did not improve the test accuracy, it successfully identified 29 good features, allowing the model to achieve a similar accuracy score using lesser input variables.

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



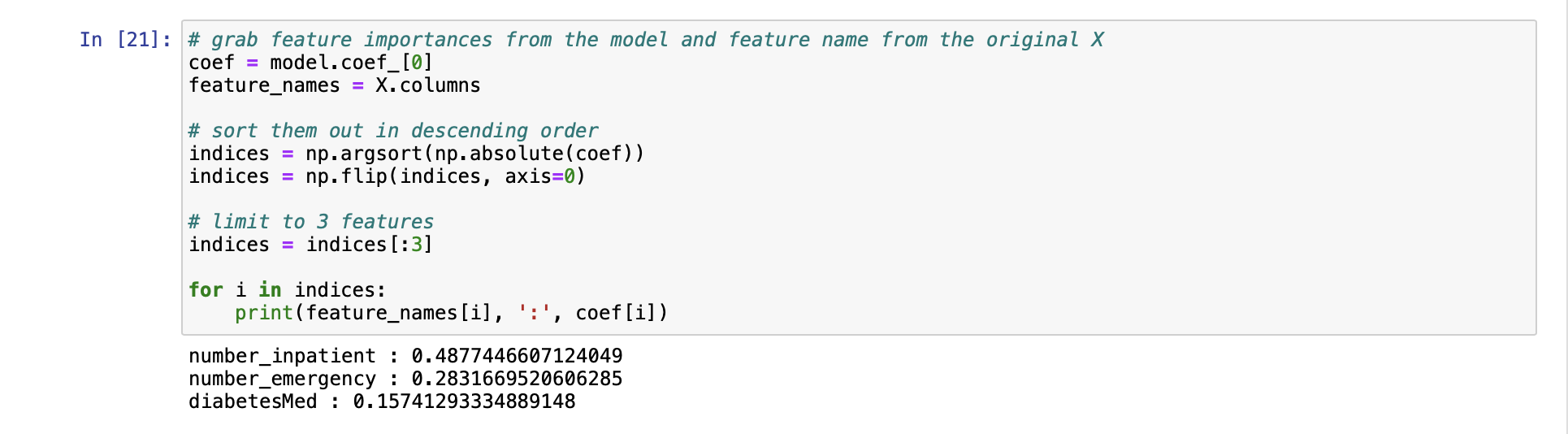
The classification accuracy on the training and test datasets is 63.05% and 62.85% respectively. Class 0 showed higher recall (81%) but lower precision (62%) with an F1-score of 0.70. In contrast, class 1 had lower recall (42%) but higher precision (65%), yielding an F1-score of 0.51, similar to the default model.

**c. Report any sign of overfitting.**

A slight overfitting is observed as train accuracy is higher than test accuracy, but it could be ignored as the values are relatively close.

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

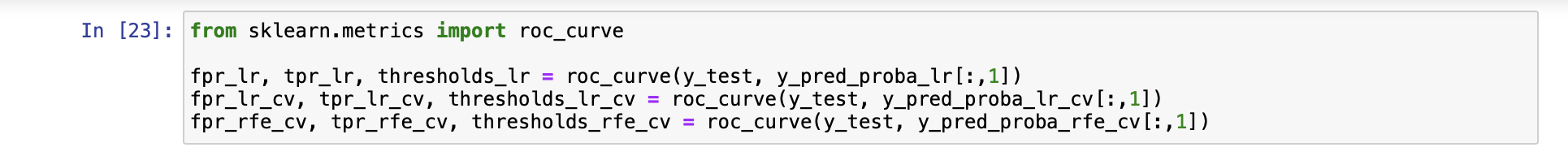
The top 3 important variables are number\_inpatient, number\_emergency and diabetesMed.

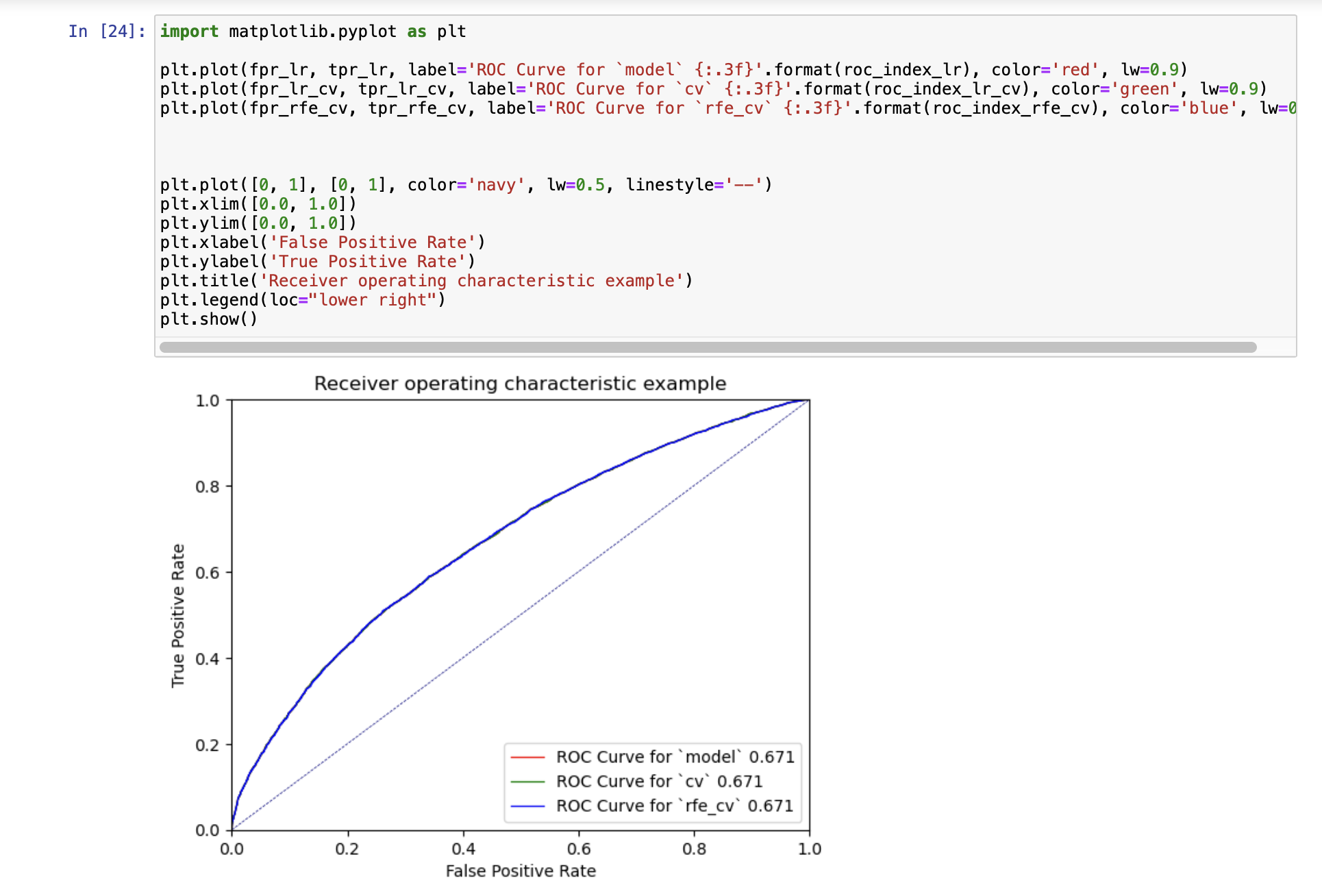


**4. Produce the ROC curve for all different regression models. Using the best regression model, can you identify which patients could potentially be “readmitted”? Can you provide the general characteristics of those patients?**

Based on the ROC curve, all models have a similar area under the curve (0.671). As the default is shown with the highest accuracy - only visible at 5th decimal point, the default model will be used.







Readmission possibility, can be identified using the slope (y-intercept) and model coefficient (x-coefficient). Regression equations can be constructed using the value of slope and model coefficient. With the regression equation, we can calculate the change in the targeted value (Readmission - True) due to the changes in the input values. For instance, if number\_medications increases by 0.021, the readmission probability will increase by log-odds of Y = 1. Probability of 0.5 above for Y=1 indicates that there is a possibility of readmission.

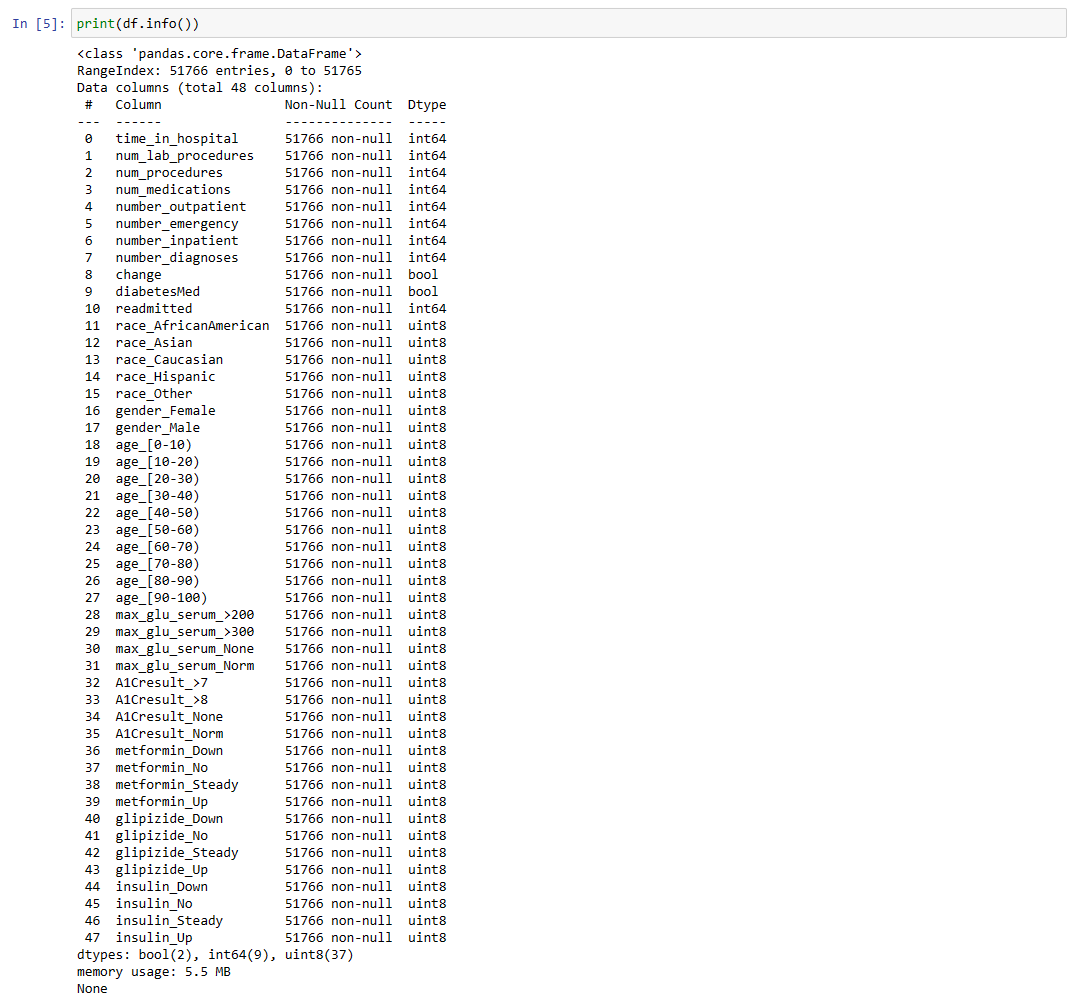
The general characteristics can be viewed from the model coefficients as well. The direction of the coefficients shows a negative/positive relationship with readmission. For instance, the higher the number\_inpatient, the higher the readmission possibility and vice versa for those with negative direction.

|  |  |
| --- | --- |
| **Alpha (Y-Intercept)** | **Beta (Model Coefficients)** |
|  |  |

**Case 5: Predictive modeling using Neural Networks**

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

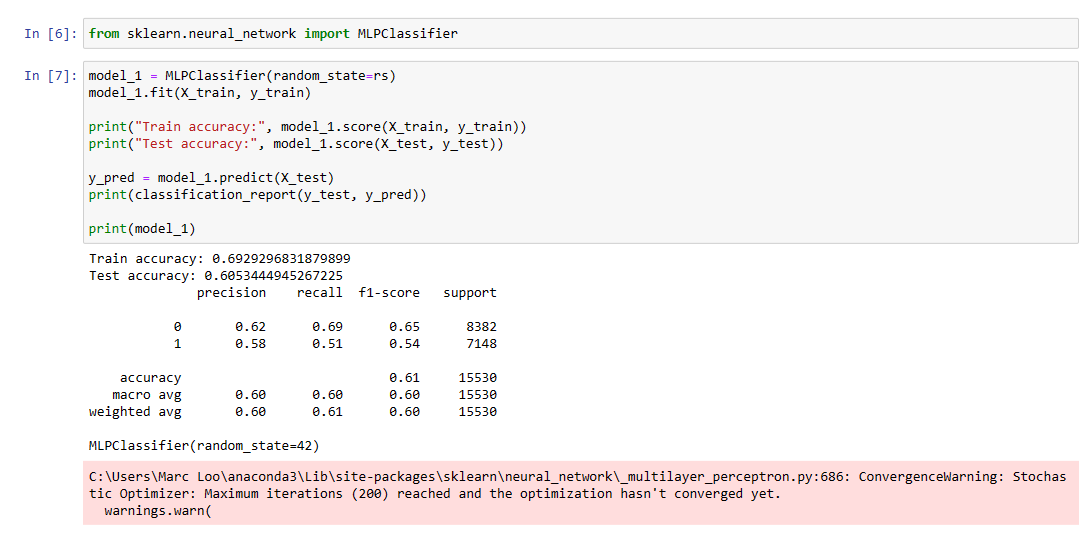
****

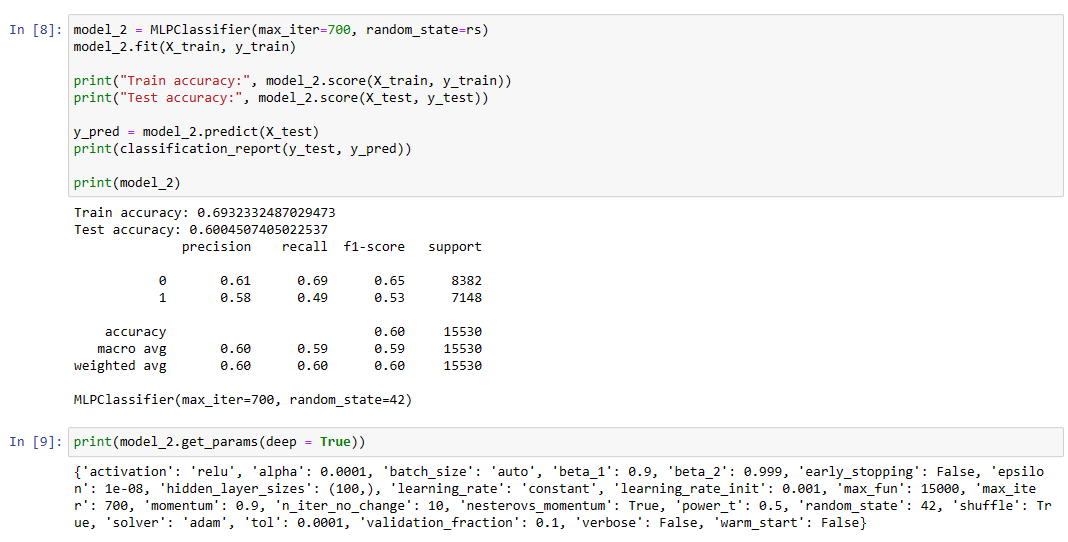
****

After importing the data, similar pre-processing from the decision tree and regression is applied for the neural network preparation. The Unknown/Invalid values in gender are replaced ith NaN. Then, the missing values in gender, race, age are replaced with mode. As shown previously, columns with highly skewed distribution that contribute no information are dropped. The categorical data is then encoded, and inputs are converted to a numpy matrix. The Distribution split used is 70% of training data, 30% of test data.

**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.**

****

****

The parameters of the default setting can be identified using the get\_params method. The activation function is relu, which helps change the linear models into a non-linear model. The max\_iterations used is 700 as initially 200 shows warning, the default hidden layers size is 100 and alpha of 0.0001.

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

For model 2, the classification accuracy on the training and test datasets is 69.32% and 60.05% respectively, which shows overfitting. Class 0 showed higher recall (69%) but lower precision (61%) with an F1-score of 65%. In contrast, class 1 had lower recall (49%) but higher precision (58%), yielding an F1-score of 53%. The weighted average of f1-score is 60%.

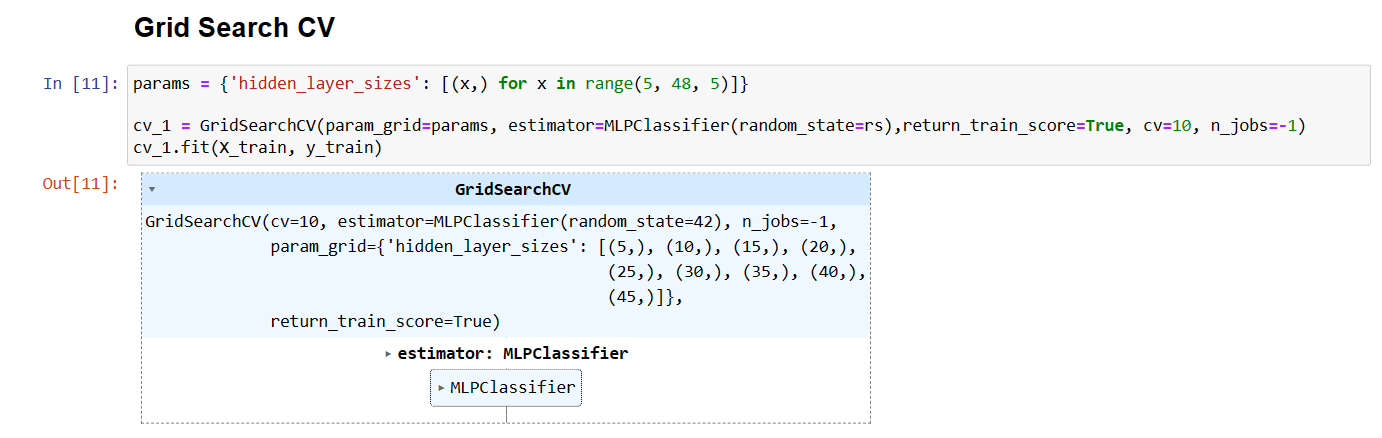
**c. Did the training process converge and result in the best model?**

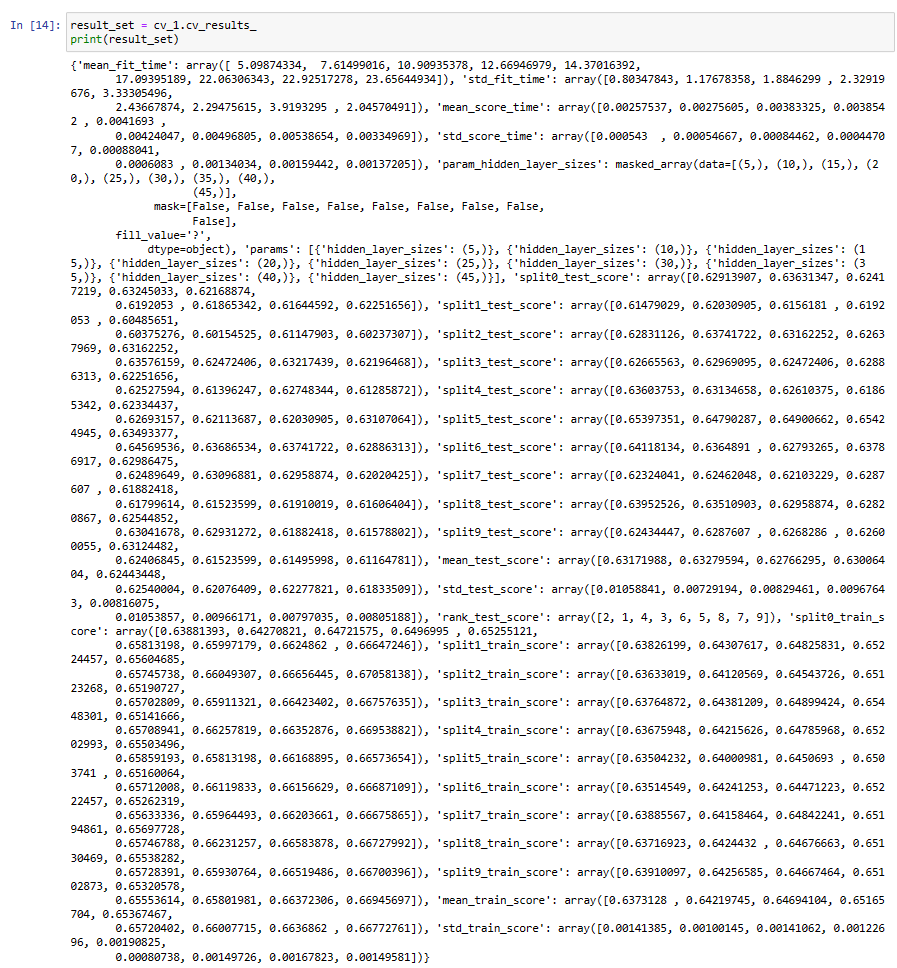
Yes, by setting the maximum iterations of 700, the model has converged and the best model is retrieved.

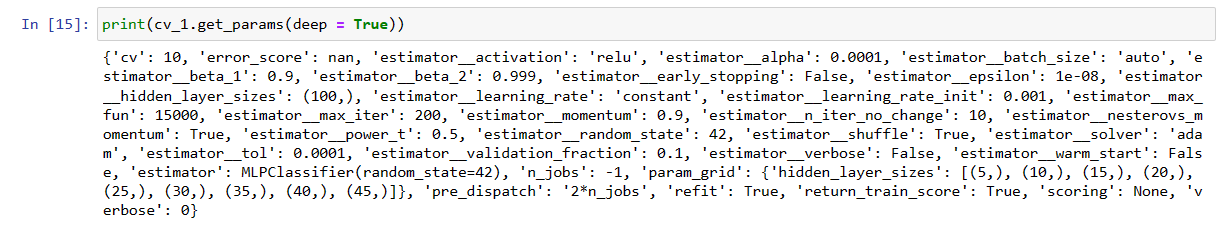
**3. Refine this network by tuning it with GridSearchCV. Report the trained model.**

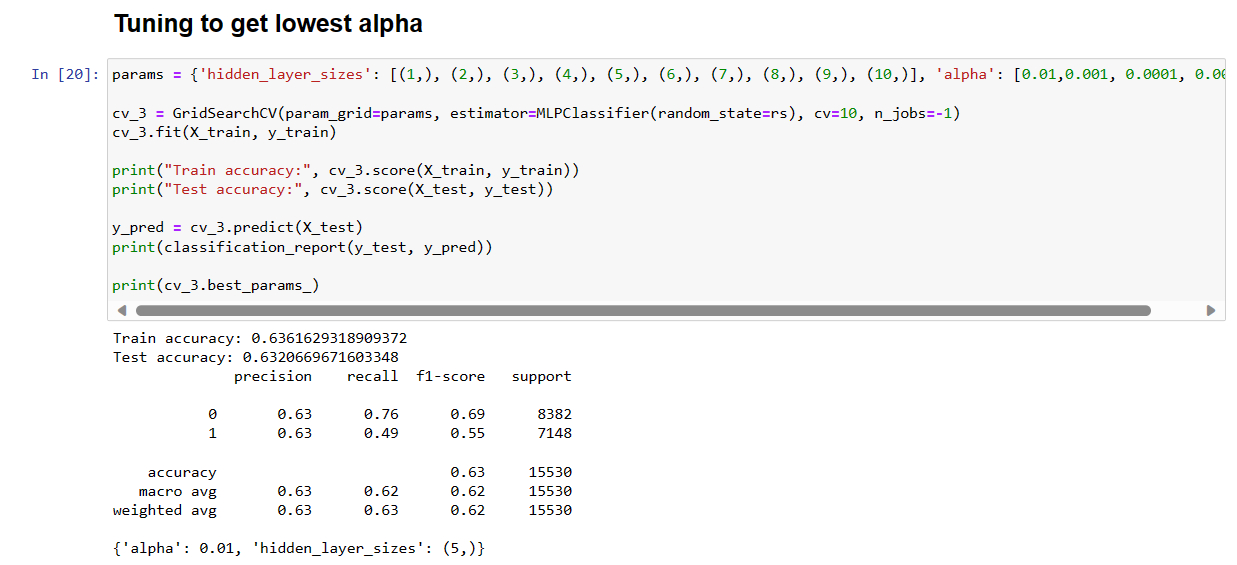
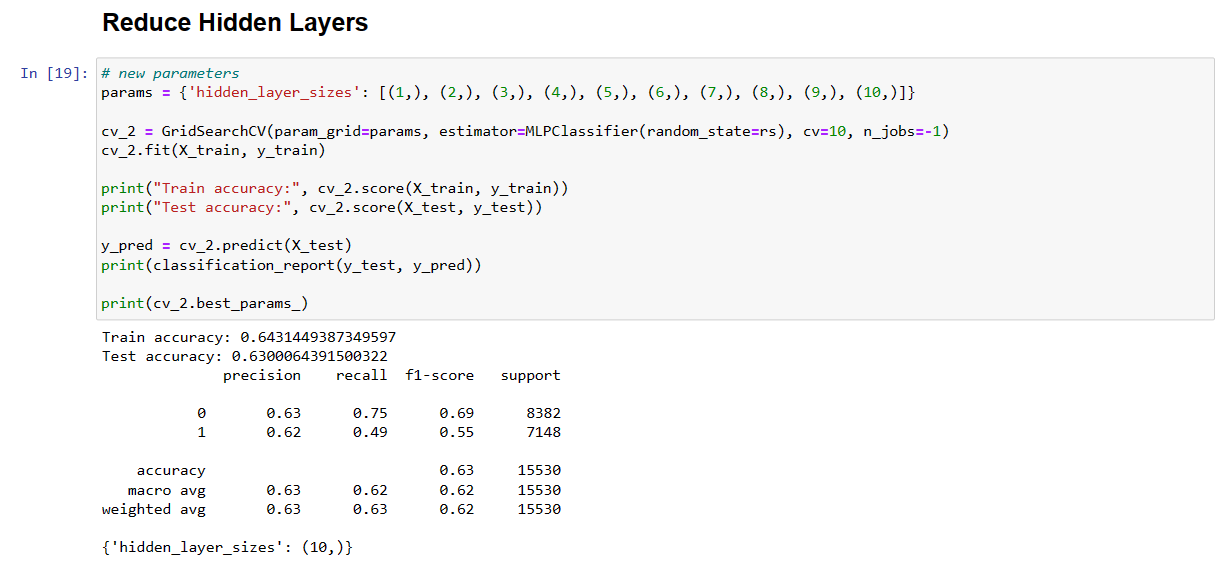
**a. Explain the parameters used in building this model, e.g., network architecture, iterations, activation function, etc.**











The parameters used can be identified by using the get\_params method. The Rectified Linear Unit activation function was utilised in the specific layers of the neural network during the training process. The max\_iterations set is 200 and showed no convergence warnings. With 47 features, we opted to tune one hidden layer of 5 to 48 neurons with increments of 5, in cv\_1. Subsequently, we also tested the hidden layers from 1 - 10 neurons in cv\_2 to see if the optimal hidden layer size remained the same from cv\_1. Lastly, we performed tuning on alpha in cv\_3. Thus, the parameters of hidden layer size and alpha varied in order to find the optimal params, while the activation function, max\_iterations and other network architecture remained default.

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

|  |  |
| --- | --- |
| **Model** | **Classification accuracy** |
| **Cv\_1**  **(hidden layers from 5 to 48, increments of 5)** |  |
| **Cv\_2**  **(hidden layers from 1 - 10)** |  |
| **Cv\_3**  **(Lowest alpha)** |  |

The hidden layer of cv\_1 shows an optimal hidden layer of 10. To check if there is a better value with reduced hidden layers, cv\_2 model runs on 1-10 hidden layer sizes. After reducing the hidden layers, the training and test accuracy remained unchanged at 64.31% and 63.00% respectively as the optimal hidden layer size is still 10. The precision, recall and f1-score also remained unchanged.

After tuning to get the lowest alpha, the training accuracy decreased to 63.62% while the test accuracy increased to 63.20%. Class 0 and Class 1 showed similar precision at 63%. Class 0 showed a higher recall and f1-score at 76% and 69% in comparison with Class 1 which showed 49% and 55% respectively. The weighted average of f1-score is 63%. Hence, cv\_3 is the better model.

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

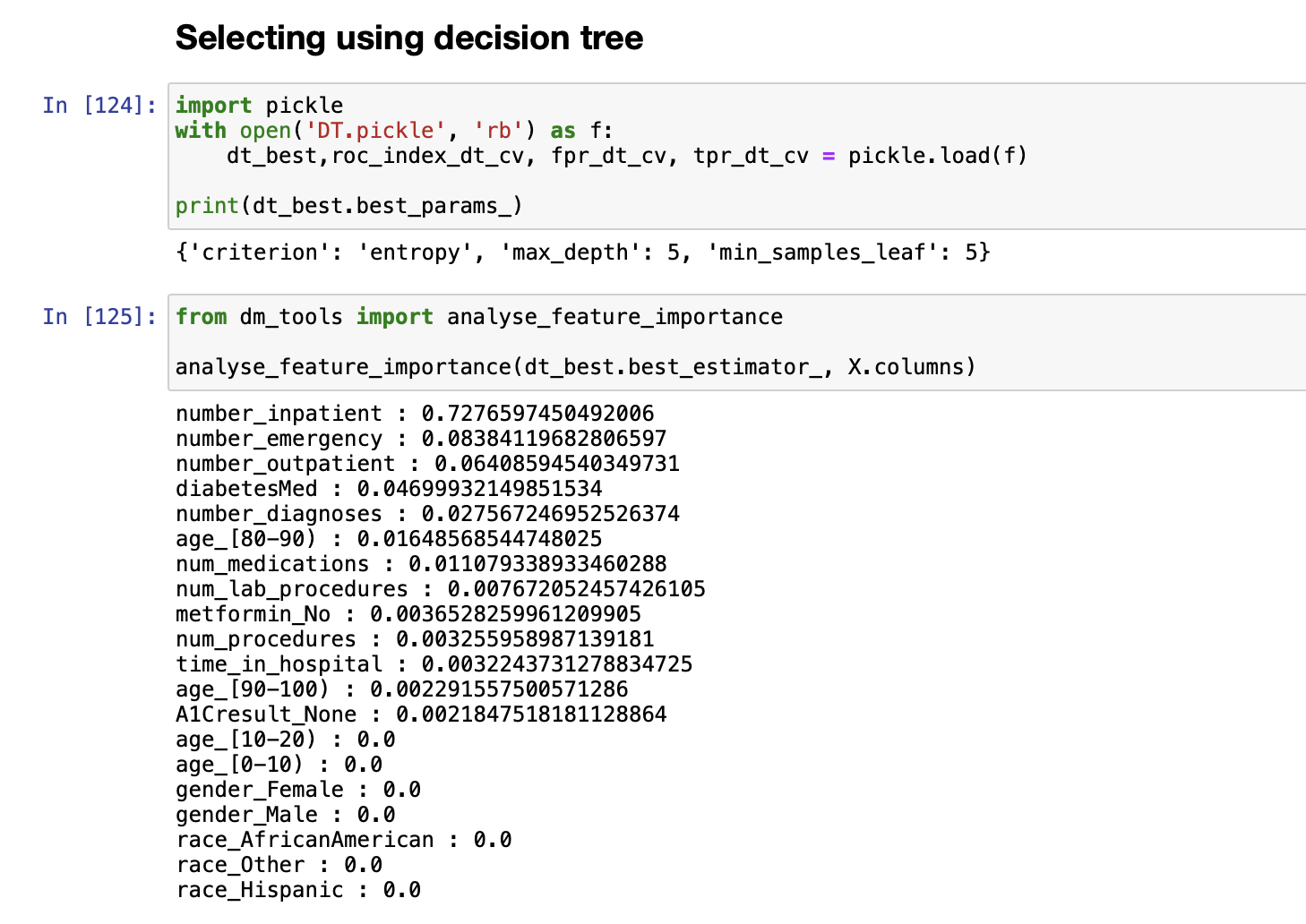
Yes, at maximum iterations of 200, no convergence warning messages occurred throughout the tuning of hidden layers and alpha.

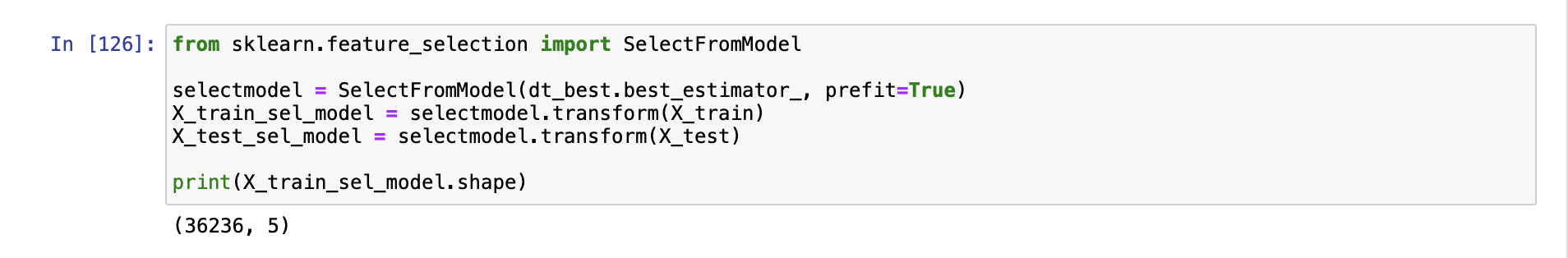
**d. Do you see any sign of over-fitting?**

There is a slight overfitting observed in both hidden layers tuning and alpha tuning as train accuracy is higher than the test accuracy, but it could be ignored as the values are relatively close.

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

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

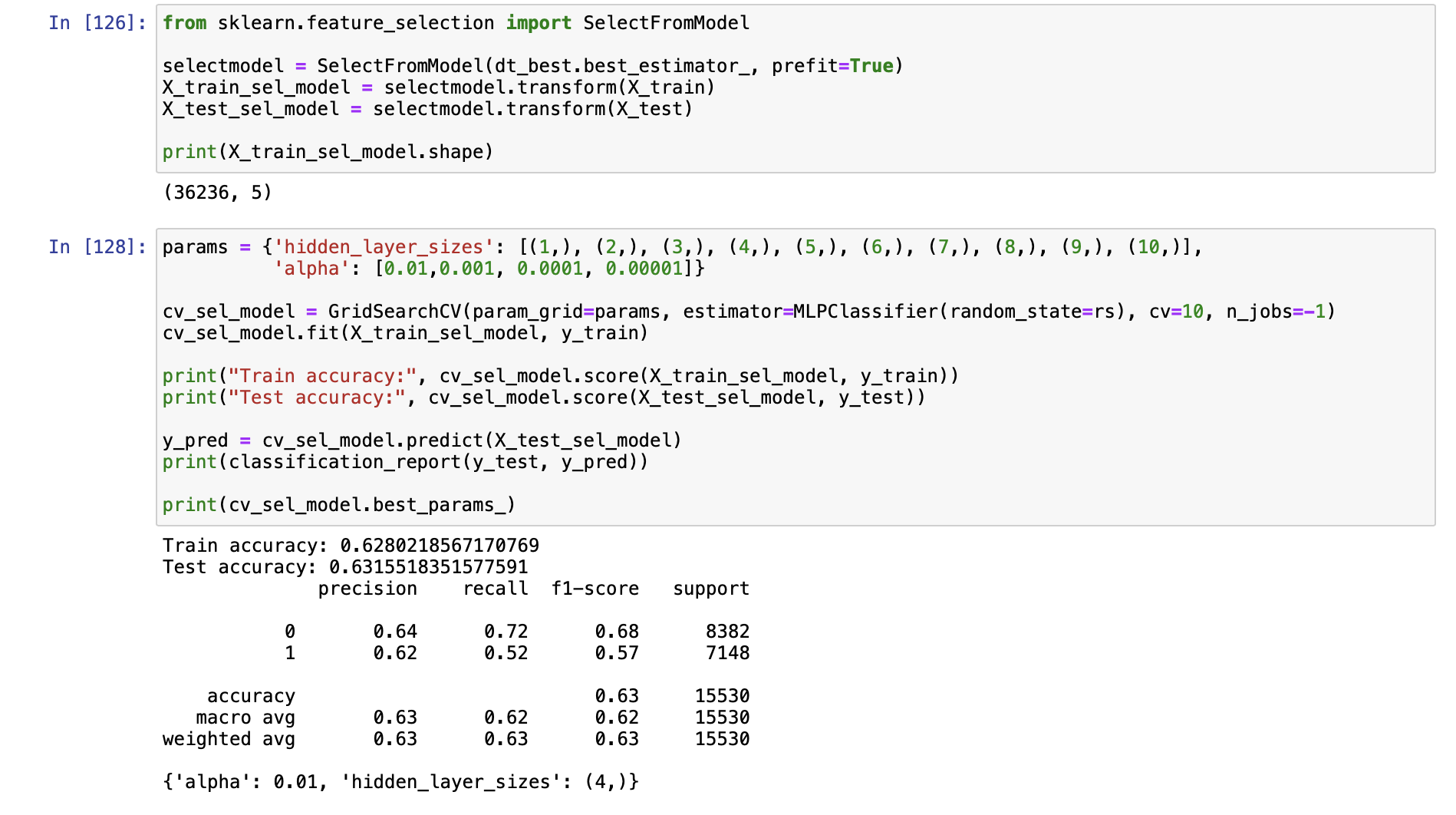
****

****

Feature selection favoured the outcome as it reduced the number of features in our original dataset. This process resulted in the identification of the top 5 important features with their importance scores for our analysis. Regarding the network architecture, the network architecture stayed the same, with a single hidden layer in a Multi-Layer Perceptron (MLP) classifier, and we didn’t change its structure or complexity. As for the inputs used as the network input, the chosen features for classification were those with the highest importance scores following the feature selection process:

* number\_inpatient: 0.7277
* number\_emergency: 0.0838
* number\_outpatient: 0.0641
* diabetesMed: 0.0470
* number\_diagnoses: 0.0276

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

****

The classification accuracy on the training dataset is approximately 62.80%, and on the test dataset, it is approximately 63.16%. In the classification report, for class 0, the model has a precision of approximately 64%, recall of 72%, and an F1-score of about 0.68. For class 1, precision is around 62%, recall is approximately 52%, and the F1-score is about 0.57.

**c. How many iterations are now needed to train this network?**

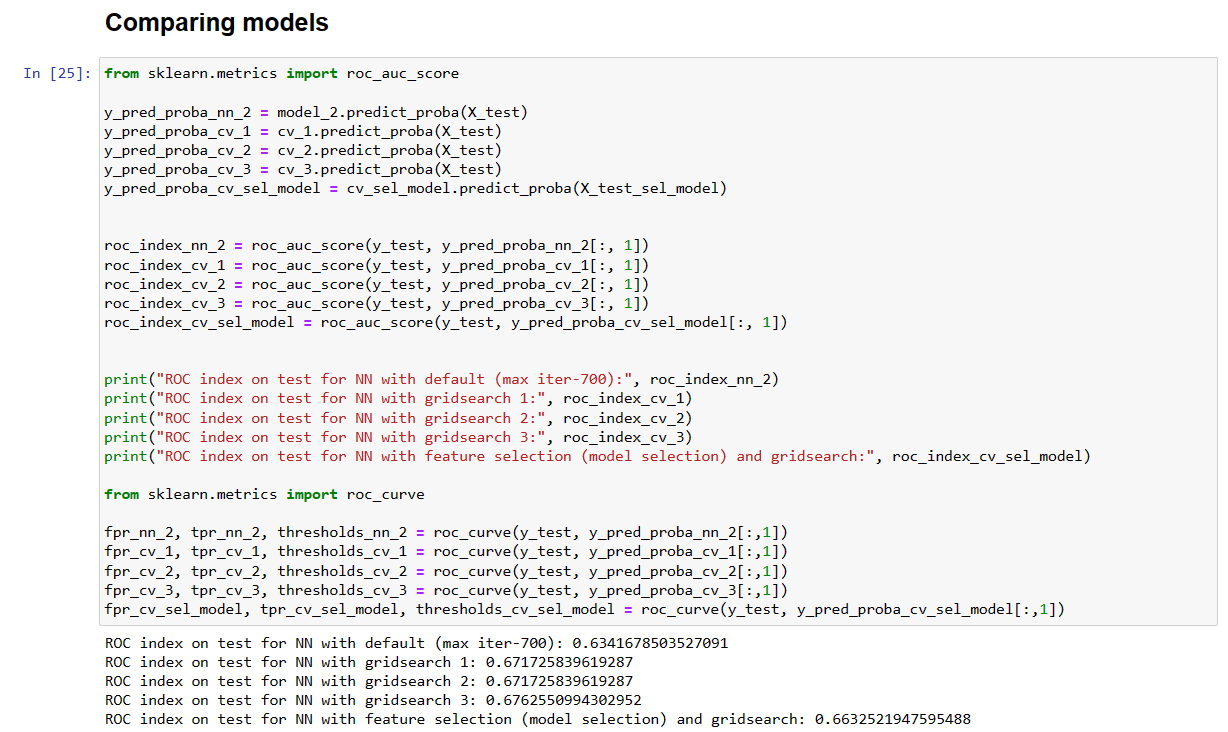
Iterations needed to train this network is 200. Since we didn’t observe any warnings indicating that the maximum iteration limit (200) was reached, we didn’t make any adjustments to it. It seems that the default iteration setting is sufficient for training this network**.**

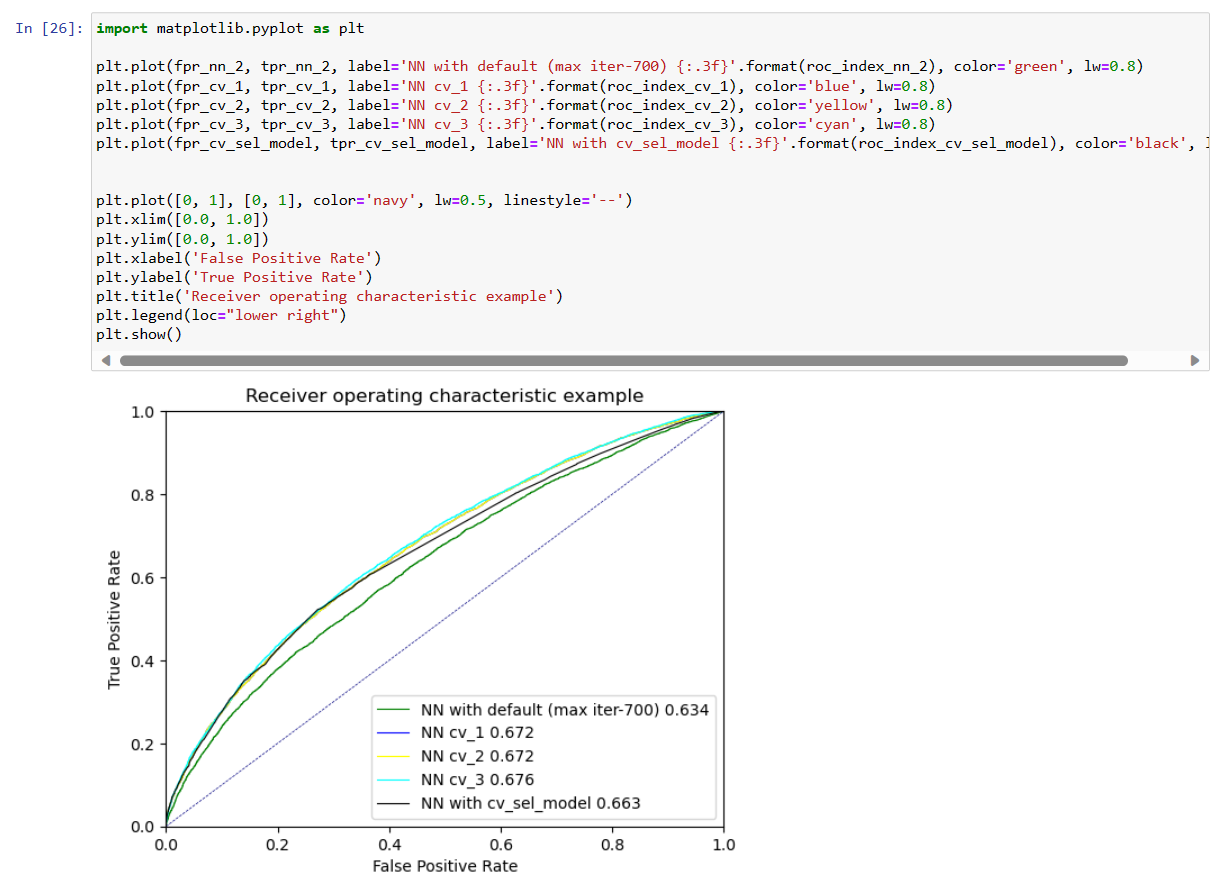
**d. Do you see any sign of over-fitting? Did the training process converge and result in the best model?**

The training accuracy (about 62.80%) and test accuracy (around 63.16%) are relatively close, with the test accuracy being slightly higher, which implies that the model generalises effectively to new data. There is no strong indication of overfitting.

Due to the absence of warnings about reaching the iteration limit, it can be inferred that the training process successfully converged and produced the best model.

**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.**

****

****

After generating the ROC curves for the following neural network models:

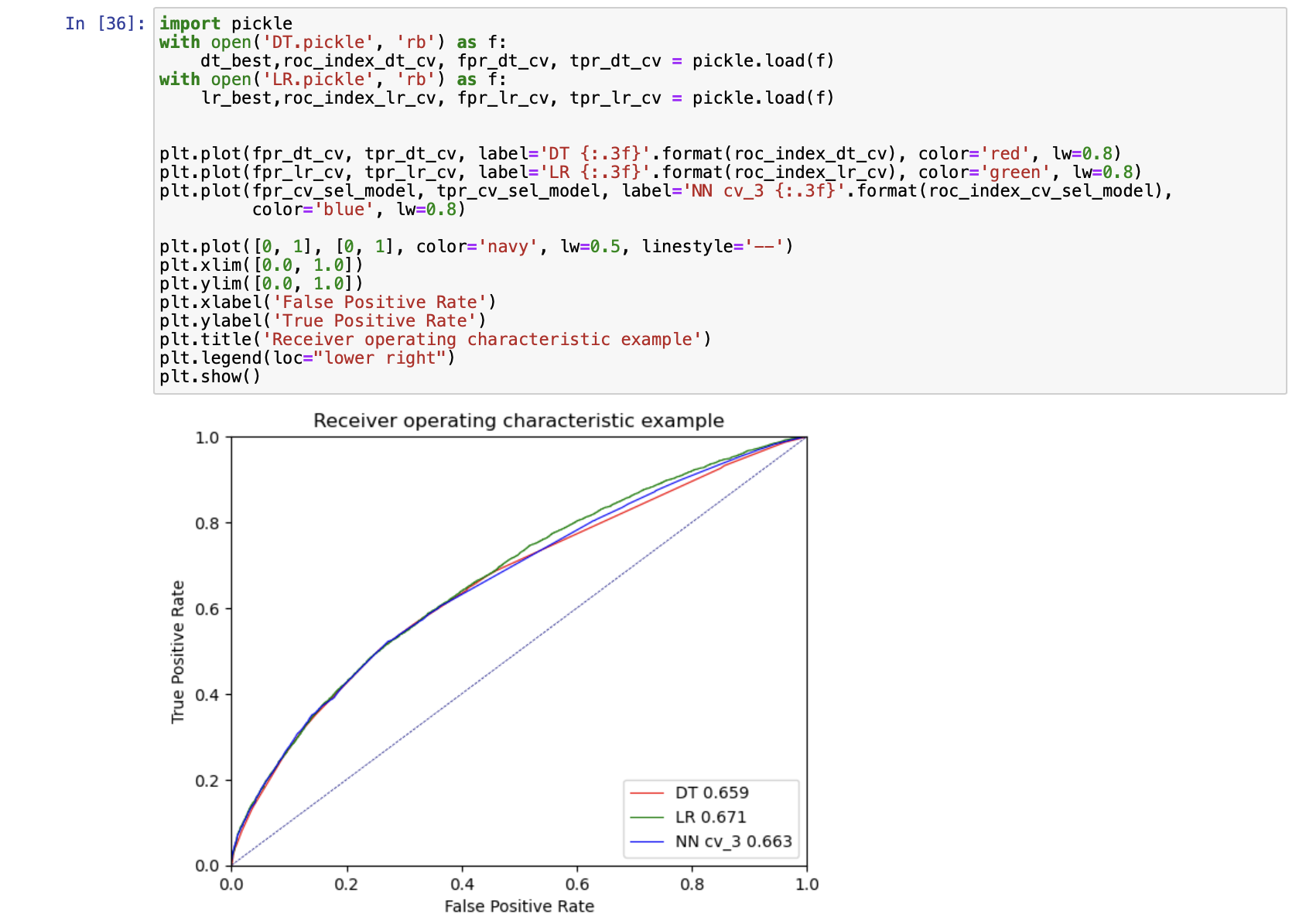
* Neural network default model (model\_2) - (max iteration 700)
* Neural network + grid search (cv\_1)
* Neural network + grid search (cv\_2)
* Neural network + grid search (cv\_3)
* Neural network + feature selection using DT + grid search (cv\_sel\_model)

We observed that among these models, the best-performing neural network on the test dataset is “Neural network + grid search (cv\_3)”, which achieved an ROC index of approximately 0.6763.

However, it is difficult to ascertain general attributes of the patients identified by the neural network model due to its complex and nonlinear operation. Neural networks are frequently referred to as “black box” models because of the lack of transparency in the way they make predictions. Although we are able to determine the importance of features, it can be difficult to understand these important scores in light of the characteristics of the patients. Furthermore, as neural networks usually take into account the interactions between features, it might be challenging to identify specific feature combinations that result in a prediction.

**Final remarks: Decision making**

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

****

|  |  |
| --- | --- |
| **Model** | **Accuracy** |
| **Decision Tree** |  |
| **Logistics Regression** |  |
| **Neural Network** |  |

Based on the ROC chart and accuracy values of each model, we will use the Logistic Regression (LR) model for decision-making. Among the three models, LR has the highest ROC index, indicating the better performance at distinguishing between classes. While LR’s test accuracy is slightly lower at 0.6287 compared to Decision Tree and Neural Network, it still performs reasonably well, and the difference is not significant. Moreover, LR’s training accuracy is comparable to the other models, signifying that it has learned the training data effectively. The LR model balances good ROC performance, high test accuracy, and interpretability. Considering these factors, it is therefore our choice for decision-making.

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

For decision trees, the positives based on the analysis are easy interpretation of the visualised model, the automatic feature selection using the features with best split, no need for standardisation and the low computational cost. However, the overfitting is rather serious on the default model (Around 99%), so we have to perform pruning. Another negative that we saw is the large visualisation size on the default model when we have yet to adjust the tree depth.

For logistic regression, the positives based on the analysis are easy interpretability (using y intercept and model coefficients), low computational cost and the default model has high test accuracy. Moreover, it is also easy to perform tuning with one hyperparameter (C). The negatives of logistic regression is the requirement of cleaning missing data, outliers as compared to decision trees. Though we have the same preprocessing for all 3 models, the decision tree can actually use data with missing values while regression requires cleaning of missing values, outliers as it is not robust.

For neural networks, the positives are the ability to generate both categorical and numerical output, as compared to decision tree and regression. However, there is a high computational cost as it took a long time for the modelling to be processed. Another issue is the black-box nature, where we are unable to interpret the general characteristics of the patients, as compared to decision tree and logistic regression.