Title: Assessment 2: Data Modelling
Predicting cancer patients survival rate

Author: Sharon Tan

Contact details: ymsharon998@gmail.com
Date of report: 11 April 2021

Table of Content

Abstract

The purpose of this assignment is to create a classification model using the “Haberman’s Survival Data Set” that measures breast cancer patients. The model predicts which label a patient falls under, whether a patient is predicted to survive 5 years or longer, or if the patient will survive no longer than 5 years. This is done so by using the K-nearest-neighbour model and the Decision Tree Learning model. Both models were trained and tested against three sets of train and test split data sets. Each model was first trained using commonly industry accepted parameters, such as k-value and max_depth, and then an optimal parameter derived from an optimal graph is used to train each model as a comparison. The one that performed the best was chosen. All the outcomes were assessed on confusion matrix, classification error rate, precision, recall and f1-Score. The best performing KNN model and the best performing Decision Tree Learning model were selected and the final two models were compared against each other. The final chosen model not only performed well on key metrics but also one that was most suited to the purpose of the model. Although the KNN model overall scored better on accuracy and precision, and returned the lowest error rate, the DTL model performed better in the F1-score, which indicates a more balanced model that maintains precision and recall. It is particularly more sensitive than the KNN model in labelling patients with status 2 (patients who may die within 5 years after surgery), which is more useful as it captures more risky patients, despite risking a higher chance of mislabelling. Hence, the Decision Tree Model is deemed to be more appropriate.

Introduction

This assignment will be focusing on data modelling with the classification approach using Python. The chosen data set is the “Haberman’s Survival Data Set” by Haberman (1976), it was donated to the UCI Machine Learning Repository, and is publicly available for use, accessible via this link(https://archive.ics.uci.edu/ml/datasets/haberman's+survival).

According to Haberman (1976), the data set contains cases from study conducted on the survival of patients who had undergone surgery to treat breast cancer, between the year 1958 and 1970. The data set characteristic is multivariate, it contains 3 attributes and it has 306 observations with no missing values present.

The attributes are as follows:

1. Age of patient at the time of operation (numerical)
2. Patient’s year of operation (numerical)
3. No. of positive axillary nodes detected (numerical)
4. The status of patients after the surgery (class)
   - 1 = Survived 5 years or longer
   - 2 = Died within 5 years

The purpose of this data modelling is to create a classification model that can, given the available attributes, best predict if a patient will survive more than 5 years or less than 5 years after surgery. Such predictions will help doctors identify and thus provide priority care to patients who face a higher risk of not surviving longer than 5 years after operation.

To achieve this, two machine learning models will be explored: the k-Nearest-Neighbours (KNN) algorithm and the Decision Tree Learning (DTL) Algorithm. Both models will be split into test and train set at the following ratio:

- 50% for training and 50% for testing;
- 60% for training and 40% for testing;
- 80% for training and 20% for testing;

Each model will be adjusted based on key parameters to optimise performance and its performance will be validated using key metrics: Confusion matrix, classification error rate, Precision, Recall and F1-Score.

The best model will be selected from each group, and a final comparison between the chosen KNN model and the chosen DTL model will be discussed. The final outcome is to select the best performing model assessed on performance accuracy and their ability to generalise unseen data.

Methodology

Data Preparation

I began by loading the data file into the Jupyter notebook, and assigned the correct header to the data set as such: Age", "Year", "No_of_Nodes","Status". I then proceeded to check the data type of each attribute, as well as checking for any typing errors, or impossible values, missing values; and made any necessary conversions, until the data is ready to be used.

Data Exploration

Before proceeding to modelling, I explored each attribute in its entirety as well as the relationships among each other. There are 4 variables in the data set, to which I used different visualisation techniques to represent the data individually. I then explored each attribute with another with relevant visualisation techniques; in total there are 6 pairs of attributes. After exploring each attribute as thoroughly as possible, and after gaining a sufficient understanding of the data set, I proceeded to model the data.

Data Modelling

I defined the target variable, which is 'Status', and then set to include all attributes in my data set as the data features. For both models, I conducted three sets of testing by splitting the train and test data set with the following ratio:

- 50% for training and 50% for testing;
- 60% for training and 40% for testing;
- 80% for training and 20% for testing;

i) KNN Model

Select k-value classifier
At the start of each training set, I determine the k-value based on a common method of using the square root of the training set, i.e. k = sqrt(n) (SINGH, 2018).

Modelling
I then fit the classifier to the train data set using fit(), and tested the mode by applying it against the test data set *(X_test)* to predict outcome, and then compared it against the actual outcome, *y_test*. To assess performances, I used a confusion matrix and the classification report to return key metrics such as Classification Error Rate, Precision, Recall, and F1-Score. To improve my models, I used a graph to plot error rates ranging from 1% to 40% against k - value to determine the optimal k-value. Once an optimal k-value is identified, I retrained the model and compared the results between the method using sqrt(n) and one using optimal graph. I repeated this with the remaining two train and test sets: 60% and 80% split, and picked the best model.

ii) Decision Tree Learning Model

Select decision tree classifier
For the parameters in the DTL model, I decided to focus on the max_depth. I first established the baseline for max_depth using 3, as it is more commonly used in the industry

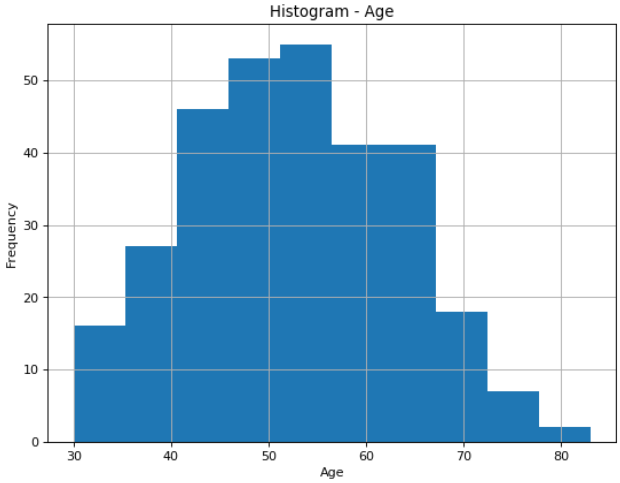
Modelling

Then I trained the model using fit() and tested it against the test data set and then compared it against the actual outcome, *y_test*. Similarly, I assessed its performance based on Confusion Matrix, Classification Error Rate, Precision, Recall, and F1-Score. I then seek to improve my model by using a graph to plot error rate of (1% - 40%) against max_depth to determine the optimal max_depth. I then retrained the model with the optimal max_depth and compared the results against the baseline of max_depth 3. I repeated this with the remaining two train and test sets: 60% and 80% split, and picked the best model out of each set.

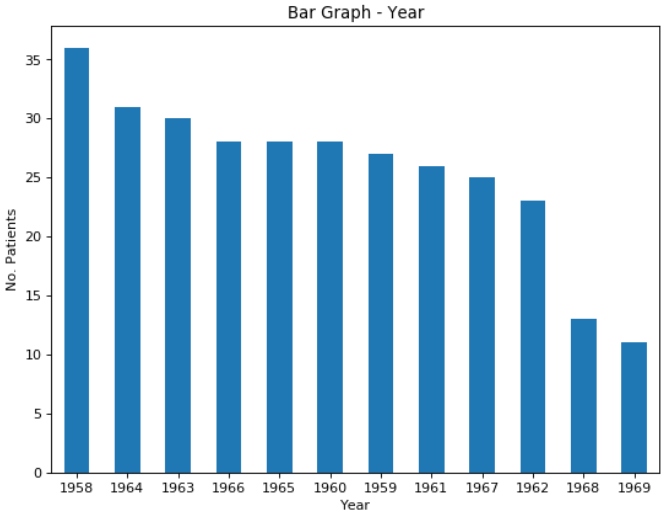
Results

Data Exploration

After exploring each individual attribute, the histogram of age shows that the patient's age in the sample is normally distributed between 30 and 83, with the mean, median and mode age of 52.

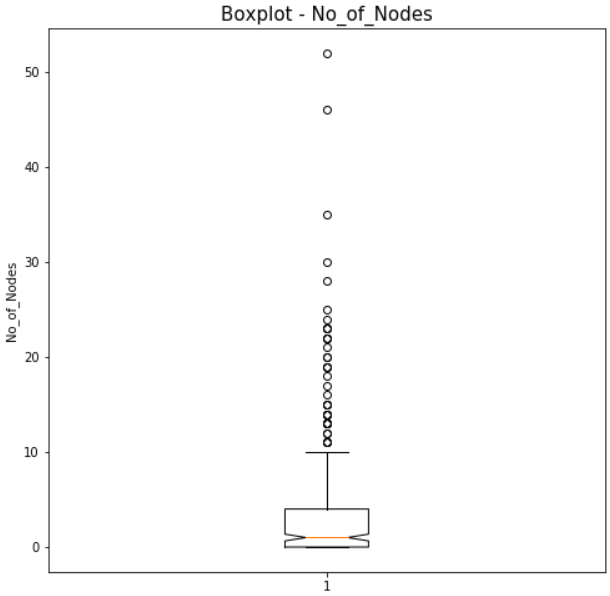


The bar graph for ‘Year’, shows that in the span of 12 years, at the start of the study in 1958, it had the highest record number of patients, and by 1969 the number of patients in the study were the lowest.

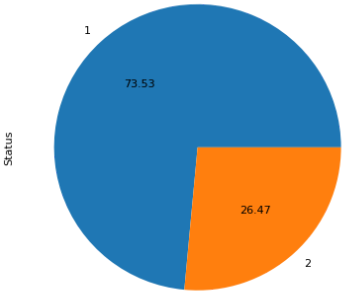


The box plot against the number of positive axillary nodes detected in patients (illustrated below) shows that most patients have between 0 to 10 number of positive axillary nodes detected, while about 20 of

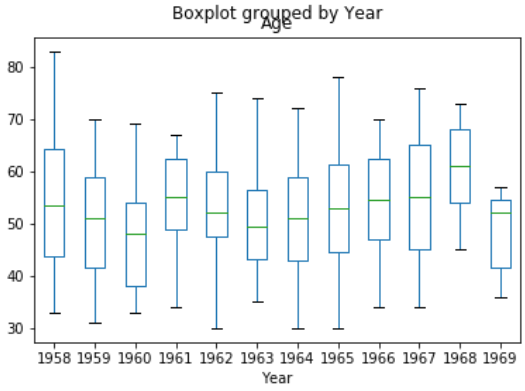
them have exceedingly high number of positive axillary nodes found in them. I decide to keep the outliers in the modelling as it is important to maintain a realistic sample as it is reflected in the real medical world where outliers are to be expected.

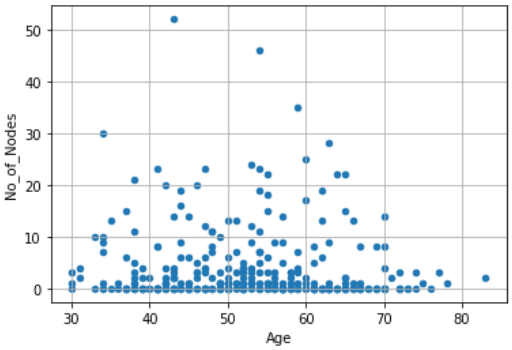


Lastly, a pie chart, shown below, was used to visualise the proportion of patients who survived longer than 5 years and those who survived less than 5 years. As shown here, among the 306 patients in the data set, about 74% of them lived for longer than 5 years after the surgery.

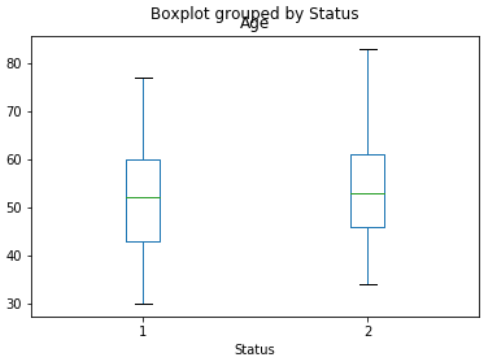


Explore relationships between attributes (Selected attributes)

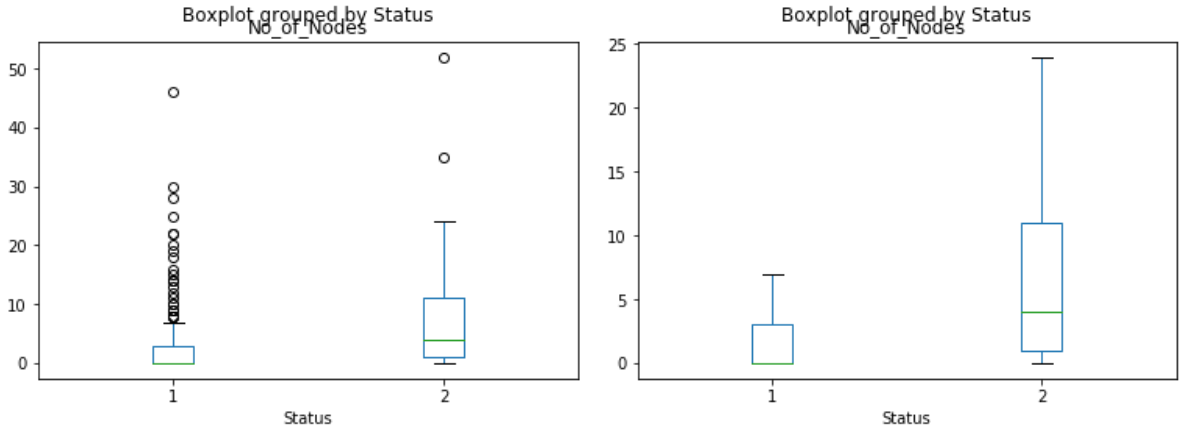




The box plot on the left is used to see if each year there is a consistent spread of age in each year. From above, it can be inferred that the mean age for each year is largely similar, with a slight increase from 1964 up till 1968 before it declined sharply. The scatterplot on the right shows that there is no correlation between a patient's age and the number of positive axillary nodes found. The patient that had the highest number of positive axillary nodes was around 42 years old, where the oldest patient in the study had only. This means age is not a good predictor of the number of positive axillary nodes.



Although it appears from the boxplot above that the mean age for patients with status '1' and those with status '2' are the same or near identical, the box plot shows that the age range for patients who survived 5 years or longer is between 30 and 70+, whereas patients in status 2 is between 35 and 80+, which means those who survived less than 5 years tend to be older than those who survived 5 years or more. Showing that age may contribute to patients' survival outcome.

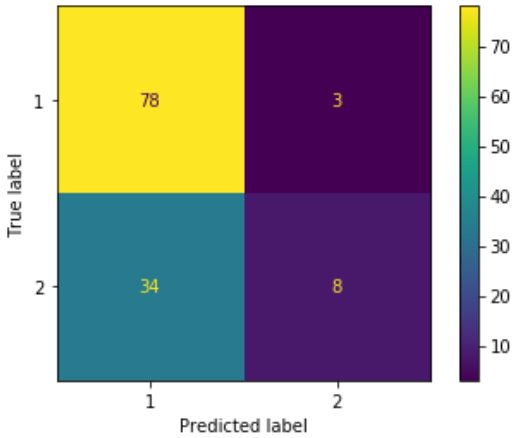


Looking at both boxplots above, left included outliers, and the right excluded, it can be inferred that patients who survived 5 years or longer, tend to have less than 10 positive axillary nodes detected, some with 0 nodes. Whereas patients who survived less than 5 years had at least 1 positive node, and can range up to 20+ nodes. However, there are many outliers detected among patients who survived longer than 5 years despite having a high number of positive nodes. This means although statistically having a lower number of nodes may indicate higher chance of survival, it is not the only factor.

Data Modelling - KNN

After setting the initial k value to sqrt(n) and then comparing it with a different k-value derived from the optimal k-value graph, the best performing KNN model is one with a k-value of 22. The outcome of the model are shown below:

Confusion Matrix (CM):



The CM above shows that the best KNN model has an error rate of 30%, out of 123 instances, the model correctly predicted 86 outcomes.

```
Classification Report (K =22):
              precision    recall  f1-score   support

           1       0.70      0.96      0.81        81
           2       0.73      0.19      0.30        42

    accuracy                           0.70       123
   macro avg       0.71      0.58      0.56       123
weighted avg       0.71      0.70      0.64       123

Compare with k value = 13
Classification Report:
              precision    recall  f1-score   support

           1       0.69      0.89      0.77        81
           2       0.50      0.21      0.30        42

    accuracy                           0.66       123
   macro avg       0.59      0.55      0.54       123
weighted avg       0.62      0.66      0.61       123
```

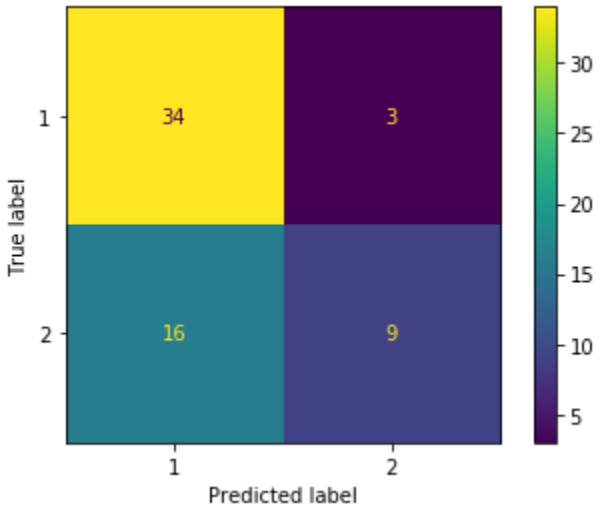
The classification report above shows a comparison between KNN (k=22) and KNN (k=13), it shows that the former outperformed on accuracy (0.70) as well as weighted avg in f1-score (0.64). Furthermore, a

higher k-value also makes the model less susceptible to outliers as it will consider more neighbours before generalising unseen data points (Subramanian, 2019).

Data Modelling - Decision Tree Learning

After training all DTL models with two different approaches, first using a baseline max_depth of 3 and the second using the optimal max_depth derived from the error rate graph, the best performing decision tree model is one with a max_depth of 7.

Confusion Matrix (CM)



The CM above shows that the best DTL model has an error rate of 31%, out of 62 instances, the model correctly predicted 42 outcomes.

```
Classification Report: Decision Tree Model (max_depth = 7
              precision    recall  f1-score   support

           1       0.68      0.92      0.78        37
           2       0.75      0.36      0.49        25

    accuracy                           0.69        62
   macro avg       0.72      0.64      0.63        62
weighted avg       0.71      0.69      0.66        62

Compare with max_depth = 3
Classification Report:
              precision    recall  f1-score   support

           1       0.67      0.92      0.77        37
           2       0.73      0.32      0.44        25

    accuracy                           0.68        62
   macro avg       0.70      0.62      0.61        62
weighted avg       0.69      0.68      0.64        62
```

The classification report above shows that the best DTL model (max_depth =7) outperformed the other with max_depth = 3, on accuracy score, 0.69 and on f1-score, 0.66. Other than precision, models were also checked if it's over or underfitting, although a max_depth of 7 is higher than 3, it is still within reasonable range.

Discussion

Observing the chosen KNN and the chosen DTL models above, both models' average performances can be summarised as below:

Metrics	**KNN (k=22)**	**DTL (max_depth = 7)**
Classification error rate	0.30	0.31
Accuracy score	0.70	0.69
Weighted avg. Precision	0.71	0.71
Weighted avg. Recall	0.70	0.69
Weighted avg. F1-score	0.64	0.66

On average, as shown above the KNN model is slightly more accurate in labelling patients correctly. Both models have the same precision of 0.71, which means 71% of the time both of them can precisely label a patient correctly. KNN has a slightly higher recall than DTL, which makes the KNN model slightly more sensitive and tends to falsely label more patients than the DTL model. The F1-score represents a balanced score that maintains precision and recall, in this comparison the DTL model is a more balanced model than KNN, achieving both good precision and good recall performance.

To decide which model is appropriate, the purpose of this assignment is worth reiterating; which is to create a model that best predicts if a patient will survive more than 5 years or less than 5 years after surgery. The outcome of prediction can be used to help doctors identify higher risk patients early on, i.e. patients who may not survive within 5 years after surgery.

With the purpose in mind, the chosen model should be one that is more sensitive, one that has a higher recall particularly in labelling patients with status 2, as this group of patients require greater medical attention. It is one that is less precise but captures more patients as it is better to over capture patients than mislabelling a patient that actually needs attention. Looking at the classification reports shown previously, the DTL model has a higher recall in labelling patients with a status 2 (0.36) compared to the KNN model which has a recall score of just 0.19.

It is also worth pointing out that the KNN model has a higher recall in patient 1 (0.96) compared to DTL which is only 0.92, this means the KNN model is more likely to label a patient 1 than 2, although it may be more relevant as most patients do survive longer than 5 years, it is better to mislabel patients 1 as 2 than to mislabel a patient who may die within 5 years as not.

Hence, based on the available information presented here, the appropriate model is the DTL model. If more factors and information are available such as medical history, diet and hospital facilities, they may help improve the models.

Conclusion

This assignment focused on classification data modelling, and used the data set, “Haberman’s Survival Data Set” by Haberman (1976). The purpose of this exercise is to create a classification model that can best predict if a patient will survive more than 5 years or less after undergoing breast cancer surgery. Such prediction modelling can be used to help doctors prioritise critical patients and to prepare early on to maximise the patients chance of survival.

After preparing the data and exploring each attribute, two different types of classification models were selected for training, they are the K-nearest-neighbor Model and the Decision Tree Model. Both models were divided into 3 sets of train and test data sets with varying splits, and were tested against different parameters. The best performing train set from each model was selected and compared against each other.

Overall, the best of both models scored similarly against their test sets. However, for the purpose of filtering out critical patients i.e. patients that may not survive greater than 5 years (Status 2), it is more appropriate to favour the model that can capture more patients that may fall under this category, than one that favours precision and risk excluding them. The KNN model showed good overall accuracy and tended to label patients correctly but it leaves out slightly more patients than the DTL model, resulting in more false negatives among Status 2 patients. The DTL model, on the other hand, scored higher in F1-score, implying a more balanced performance in precision and sensitivity. It is particularly more sensitive in labelling patients as “Status 2”. Despite it resulting in more false positives, this supposedly less than ideal outcome may be favored in this instance as Status 2 patients is the key concern group. For this reason, this makes the DTL Model more useful and practical than the KNN Model, in helping doctors ensure as many patients as possible survive more than 5 years post operation.

References

Chowdary, D. H., 2020. 'Decision Trees Explained With a Practical Example' [Online]
Available at: https://towardsai.net/p/programming/decision-trees-explained-with-a-practical-example-fe47872d3b53
[Accessed 11 April 2021].

Kumar, A., 2020. 'Statistics for Data Science : Box & Whisker Plots'. [Online]
Available at: https://medium.datadriveninvestor.com/statistics-for-data-science-box-whisker-plots-e5ce7e9ff92a
[Accessed 11 April 2021].

Seif, G., 2018.'A Guide to Decision Trees for Machine Learning and Data Science'. [Online]
Available at: https://towardsdatascience.com/a-guide-to-decision-trees-for-machine-learning-and-data-science-fe2607241956
[Accessed 11 April 2021].

SINGH, A., 2018. 'A Simple Introduction to K-Nearest Neighbors Algorithm'. [Online]
Available at: https://www.analyticsvidhya.com/blog/2018/08/k-nearest-neighbor-introduction-regression-python/
[Accessed 11 April 2021].

Subramanian, D., 2019. 'A Simple Introduction to K-Nearest Neighbors Algorithm'. [Online]
Available at: https://towardsdatascience.com/a-simple-introduction-to-k-nearest-neighbors-algorithm-b3519ed98e
[Accessed 11 April 2021].

Lekhtman. A., 2019 'Data Science in Medicine — Precision & Recall or Specificity & Sensitivity?' [Online]
Available at: https://towardsdatascience.com/should-i-look-at-precision-recall-or-specificity-sensitivity-3946158aace1
[Accessed 11 April 2021].