Software Design Project 2 Report

In this project, I was tasked with doing an exploratory analysis of a specific dataset and preparing it for predictive analysis of an intelligent system. The goal is to make the best learning model with supervised learning techniques to predict the recurrence of breast cancer for patients. Seeing as this is a real-world question/difficult to get and AI continues to advance it would be great to get a model with good enough accuracy to help!!

Data Preprocessing

For this dataset, I did the usual beginning preparations such as taking care of any duplicated columns, and null/invalid values. Most of it went relatively smoothly except for the invalid values, the “node-caps” column had. There were 8 values and since most of the columns were object types it wasn’t as simple as getting the mean so I had to do some manual calculations to find which value suited which cell. I had noticed a small similarity between the “irradiat” column to the “node-caps” column so I got some ratios in comparison and felt it fit to use as a foundation for assigning the values!! Whereas with the “breast-quad” column I didn’t know if there was a reliable way to assign the invalid value to something and seeing as there was only 1 invalid value I dropped that cell.

Uni/Multivariate Analysis

Before I did one-hot encoding since there is A LOT of potential to do so, I first did some univariate analysis on a couple of the columns. I felt it necessary to know the age range and have a histogram representing it since one of the most important details to pay attention to with certain diseases is the age at which they were diagnosed, most at risk, etc. Once I plotted the ages I saw that the majority of patients are diagnosed from ages 40-60 and very uncommonly diagnosed in patients of 20-30 years old. I also did a histogram plot on the breast quadrant with specifications to which breast each quadrant had a ratio to get an idea of the tumor location trends.

I did one boxplot before having done any one-hot encoding and it was only for the degree malignance which is important but I felt as though other columns may have better suited the box plot. I then did some one-hot encoding to every column besides “age”, “tumor-size”, “inv-nodes”, and “deg-malig”.

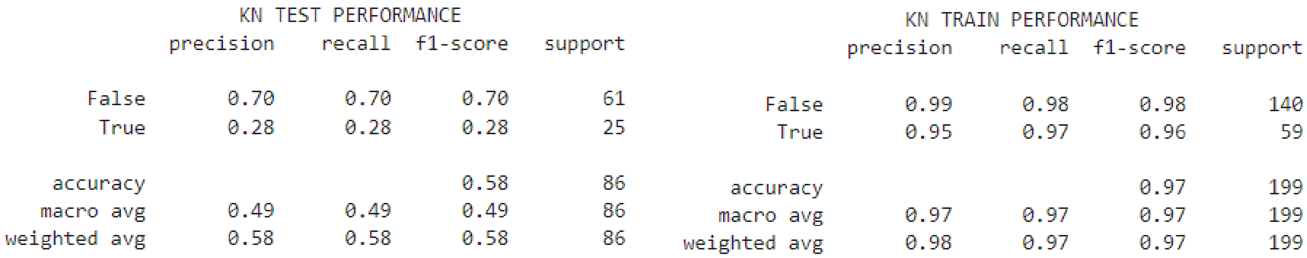
To these columns (besides “deg-malig”) I decided to generalize them and convert them to integer data types, seeing as they were ranges of numbers I had to do a bit of element manipulation. I actually didn’t know how to go about this at first so I asked chat gpt

“How could I generalize this list into a list of integers that keeps the same range? ['30-39' '40-49' '60-69' '50-59' '70-79' '20-29']”

This gave me a function that essentially splits the elements from the hyphen (-) and gets the average of the 2 numbers in the range. I then applied this function to all 3 columns and made sure to change their data type to an integer. Finally, with all my data preprocessing done, I made a multivariate heatmap plot to compare the similarities of ALL the columns to one another.

Building & Assessing Models

In this section, I was to train and test my data and choose 3 supervised learning techniques to see the classification results of the different techniques. I started off by splitting my data with 70/30 train-test, using the one-hot encoded “class” column as my dependent variable since I want to predict the recurrence of breast cancer. Then using the rest of the columns as the independent variables, the more the merrier as they say. I chose the K-Nearest Neighbor, Random Forest, and Decision Trees Classifiers. The point of using these classifiers is to predict the recurrence in future patients so I want to mainly pay attention to the false negatives since it'd be more detrimental to falsely identify non-recurrence patients than not.



```
                    KN TEST PERFORMANCE
              precision    recall  f1-score   support

       False       0.70      0.70      0.70        61
        True       0.28      0.28      0.28        25

    accuracy                           0.58        86
   macro avg       0.49      0.49      0.49        86
weighted avg       0.58      0.58      0.58        86
```

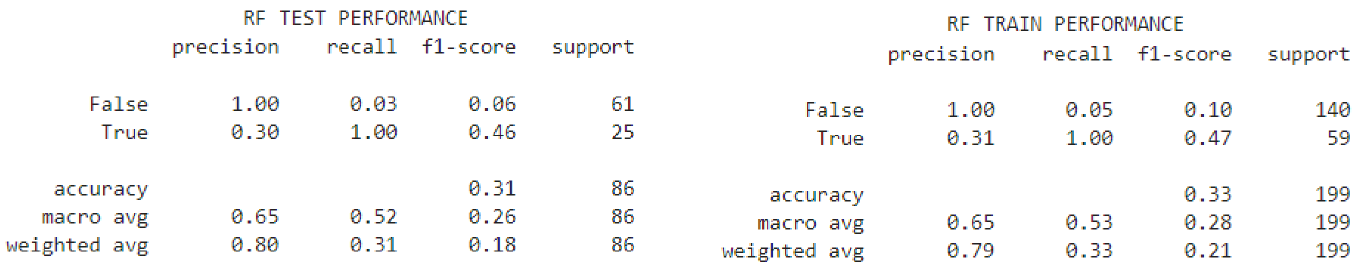
```
                    KN TRAIN PERFORMANCE
              precision    recall  f1-score   support

       False       0.99      0.98      0.98       140
        True       0.95      0.97      0.96        59

    accuracy                           0.97       199
   macro avg       0.97      0.97      0.97       199
weighted avg       0.98      0.97      0.97       199
```

Figure 1: K-Nearest Neighbor Classification Report

When using KN Neighbor, grid search showed that the best parameters would be 1, and after running a classification report with this it doesn’t seem the case. There's a very mediocre accuracy of 58% on the test performance, and even lower of recall score for True values. I want a higher true value percent but when comparing the test to the training performance its almost day and night with the difference in accuracy and recall. I get a very high value of 97% for accuracy AND recall. With such a large difference in performance between testing and training, I can only assume the model is overfitting the data which isn't always bad but with this specific prediction it's very important.



```
              RF TEST PERFORMANCE
              precision    recall  f1-score   support

       False       1.00      0.03      0.06        61
        True       0.30      1.00      0.46        25

    accuracy                           0.31        86
   macro avg       0.65      0.52      0.26        86
weighted avg       0.80      0.31      0.18        86
```

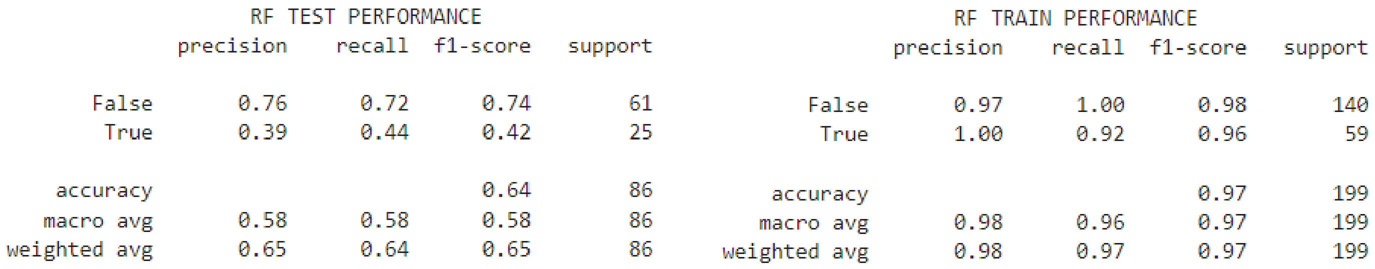
```
              RF TRAIN PERFORMANCE
              precision    recall  f1-score   support

       False       1.00      0.05      0.10       140
        True       0.31      1.00      0.47        59

    accuracy                           0.33       199
   macro avg       0.65      0.53      0.28       199
weighted avg       0.79      0.33      0.21       199
```

Figure 2: Random Forest Classification Report

As for Random Forest, I was pleasantly surprised. The test data showed a recall of 100% for true values which means I didn't get any false negatives which is exactly what I want for identifying recurrence!! Although the accuracy was pretty low at 31% which on its own is relatively disappointing. When compared to the training report I was a little delighted to see that despite the accuracy still being generally low (33%) the false positives were still 0 and since the accuracy didn't change too much I can assume the fitting was fine. So far I think the Random Forest performed better taking everything into account over the KN Neighbor classifier.



```
              RF TEST PERFORMANCE
              precision    recall  f1-score   support

       False       0.76      0.72      0.74        61
        True       0.39      0.44      0.42        25

    accuracy                           0.64        86
   macro avg       0.58      0.58      0.58        86
weighted avg       0.65      0.64      0.65        86
```

```
              RF TRAIN PERFORMANCE
              precision    recall  f1-score   support

       False       0.97      1.00      0.98       140
        True       1.00      0.92      0.96        59

    accuracy                           0.97       199
   macro avg       0.98      0.96      0.97       199
weighted avg       0.98      0.97      0.97       199
```

Figure 3: Decision Tree Classification Report

Finally with the Decision Trees I was a little disappointed seeing as I expected this classifier to do the best for this specific dataset since it's going from if-statement to if-statement. I realize now that this isn't necessarily the most accurate for this dataset seeing as there are not many concrete ways to identify recurrence, unlike other diseases/illnesses. Nonetheless, the classification report it seems very similar to the KNN report, we have a mediocre but the highest accuracy (64%) for testing but a higher percentage for false negatives which is an increase we don't want. After comparing it to the training performance it seems to be another case of overfitting just like the KNN, the jump in accuracy is too suspicious even then there is a 92% of false negatives which is the biggest problem!!

Conclusion

As mentioned at the beginning my goal is to predict the recurrence of breast cancer in patients, when dealing with intelligent systems and predicting information based on data it is very important to understand the impact the predictions can make. Of course, every prediction being focused on is important but when dealing with the prices of cars vs the recurrence of a disease in patients one bears a heavier burden in accurate predictions. For our specific dataset/scenario we want to focus on the recall seeing as false negatives are more impacting than not. With this in mind knowing how each model performs I would choose the Random Forest classifier over the 2 others purely for the lack of false negatives!! With this being said if there is a classifier with higher accuracy and similar if not better recall percentage that'd be ideal but if only it were that simple.