COMP47460

Assignment 1

Judith Smolenski, 22204071

1.1 - Data Cleaning and Prep

Examining the data for missing values, normalisation and removal that needed to be done to prep the data for further work.

- There were no missing values throughout all of the features, so none needed to be replaced.
- Using max-min normalisation I made sure that all the numerical feature values were updated to be on the same scale to allow for meaningful comparison between them.

At this beginning step, although some features already seemed not to be promising in terms of returning valuable insights about our target feature, I chose to keep all of them and complete the feature selection at the later stage of task 1.4. All features have distinct values and no missing data, therefore they do contain some information which may have an effect on the classification in the next step.

1.2 - Classification with All Features

k-NN

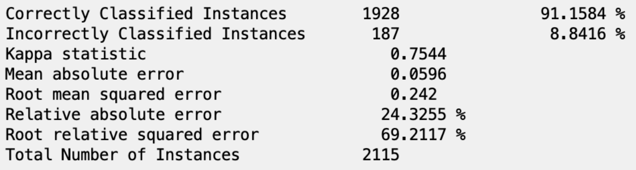
3-fold: Decent results and accuracy with 91.1584% correctly classified instances. Some of the error rates are cause for concern, specifically as this is a medical context and errors at any point can have major side effects with the relative absolute error being 24.3255% and root relative squared error at 69.2117%. The high dimension of the current dataset probably has an effect on these results too and using 3 as the value of k may return better results after feature selection.

10-fold: Returns very slightly better results overall with 91.4421% correctly classified instances. However the difference to the 3-fold cross validation is almost negligible with error rates differing only by around 1% each.

20-fold: Does not alter the results much with 91.773% correct classifications and high error rates.

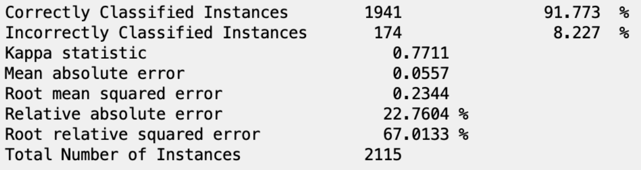
Confusion Matrices for all three models also show little difference in the performance, however all of them are consistent in that they perform better when classifying the "Normal" class and have trouble when classifying the "pathological" and "suspect" classes. This makes sense as the two latter classes are less common in the dataset. The models have a bias towards the majority set.

Comparison of 3-fold and 20-fold:



```
Correctly Classified Instances        1928               91.1584 %
Incorrectly Classified Instances       187                8.8416 %
Kappa statistic                          0.7544
Mean absolute error                      0.0596
Root mean squared error                  0.242
Relative absolute error                 24.3255 %
Root relative squared error             69.2117 %
Total Number of Instances             2115
```

3-fold



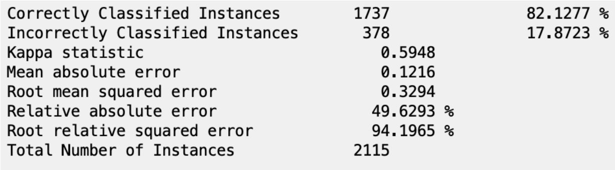
```
Correctly Classified Instances        1941               91.773  %
Incorrectly Classified Instances       174                8.227  %
Kappa statistic                          0.7711
Mean absolute error                      0.0557
Root mean squared error                  0.2344
Relative absolute error                 22.7604 %
Root relative squared error             67.0133 %
Total Number of Instances             2115
```

20-fold

Naive Bayes

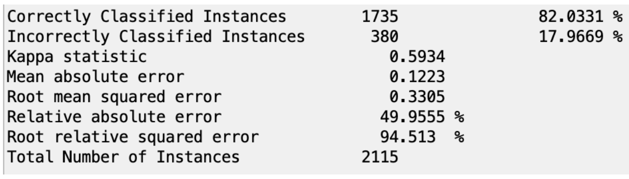
3-fold: Overall accuracy at 82.13% which is worse than any of the k-NN models. Again the different classes of our target feature return different accuracy results with the majority class "Normal" having a precision of 97.8%, the "Pathological" class being at a low 58.2% and "Suspect" at 48.4%. The high numbers of false negatives in the prediction of both the pathological and suspect classes make this model not suited for medical use where there is more lenience for false positives than false negatives.

Comparing different sizes of cross-validation:



```
Correctly Classified Instances        1737               82.1277 %
Incorrectly Classified Instances       378               17.8723 %
Kappa statistic                          0.5948
Mean absolute error                      0.1216
Root mean squared error                  0.3294
Relative absolute error                 49.6293 %
Root relative squared error             94.1965 %
Total Number of Instances             2115
```

3-fold



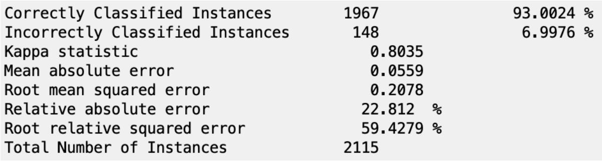
```
Correctly Classified Instances        1735               82.0331 %
Incorrectly Classified Instances       380               17.9669 %
Kappa statistic                          0.5934
Mean absolute error                      0.1223
Root mean squared error                  0.3305
Relative absolute error                 49.9555 %
Root relative squared error             94.513  %
Total Number of Instances             2115
```

20-fold

Decision Tree

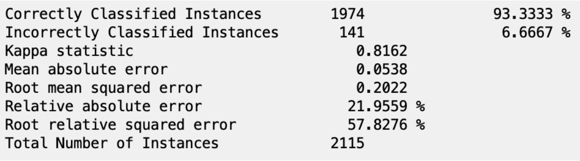
With the data being the way it currently is, with all features still being used, the decision tree returned the best overall accuracy results. At 3-fold cross validation the accuracy was 93.0024%. The error rates were still high, most likely due to the large dimensionality of the dataset, and the mis-classification was very similar to the other models tested above.

Comparison of 3-fold and 20-fold:



```
Correctly Classified Instances        1967               93.0024 %
Incorrectly Classified Instances       148                6.9976 %
Kappa statistic                          0.8035
Mean absolute error                      0.0559
Root mean squared error                  0.2078
Relative absolute error                 22.812  %
Root relative squared error             59.4279 %
Total Number of Instances             2115
```

3-fold



```
Correctly Classified Instances        1974               93.3333 %
Incorrectly Classified Instances       141                6.6667 %
Kappa statistic                          0.8162
Mean absolute error                      0.0538
Root mean squared error                  0.2022
Relative absolute error                 21.9559 %
Root relative squared error             57.8276 %
Total Number of Instances             2115
```

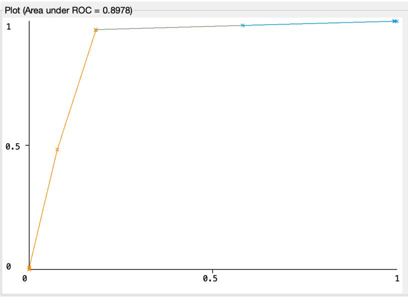
20-fold

Overall, at this stage of the project, the data is too cluttered by all the features still present for the models to be able to make better predictions on our target feature than what is seen above. The high error rate and especially the high rates of false negatives when predicting the minority classes would cause issues if these models were to be used in a medical setting. The three model types and their accuracy measures do not differ substantially here. The decision trees could be useful for dimension reduction steps as they show high and low entropy features when selecting their splits.

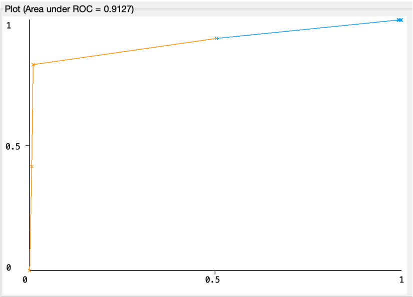
1.3 - ROC Curves

NB. For plotting the curves I chose the 10-fold classification models for their average results. As results did not vary much overall though, the choice has little effect on the output. I checked the other k-fold validation results and they confirmed this suspicion.

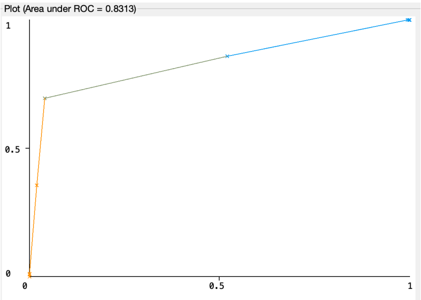
k-NN results:



“Normal”

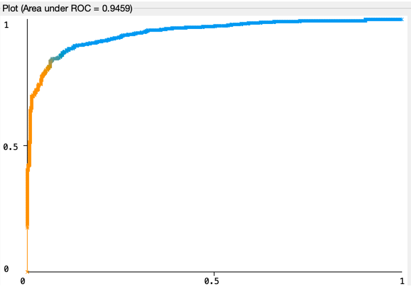


“Pathological”

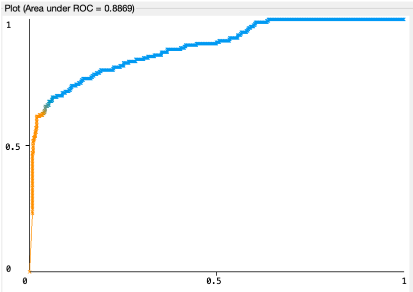


“Suspect”

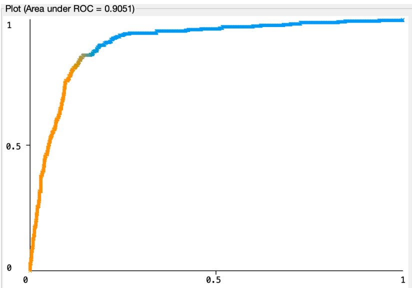
Naive Bayes results:



“Normal”

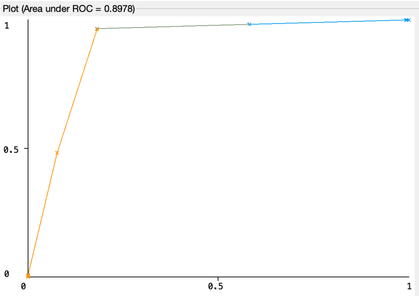


“Pathological”

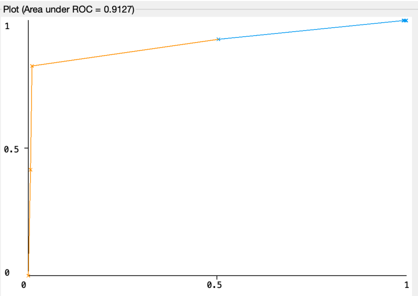


“Suspect"

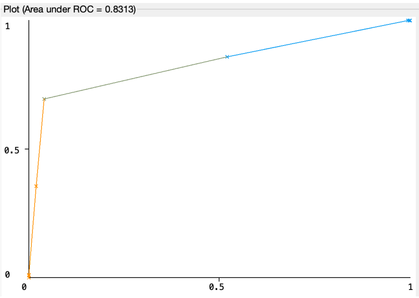
Decision Tree results:



“Normal”



“Pathological"



“Suspect”

For all of the above models we can see that the predictions are pretty good. This lines up with the data gathered in task 1.2. AUC values for all models and target feature classes are in the 0.80s with some in the low 0.90s as well. This shows that overall the models are doing a good job of distinguishing between the different classes and the scores are significantly higher than 0.50 which would be random chance.
The interpretation of these results as being favourable for the models only extends as far as the quality of the data being used. Although everything is clean and normalised and therefore ready to be used, the large number of features as well as the confusion matrices results of false negatives for the cases which are not “normal” suggest that the models need to be improved further by scaling down the feature set being used. This will be done in the next step. But as it stands currently, the ROC and their AUC values suggest there is potential for good prediction use once further data prep has been done.

1.4 and 1.5 - Feature Selection / Top 5 Features

Table of top 7 features returned using different filter/ranking methods:

	Correlation	Information Gain	Wrapper Forwards	Wrapper Backwards
1	abnormal_short_term_variability	mean_value_of_short_term_variability	uterine_contractions	accelerations
2	percentage_of_time_with_abnormal_long_term_variability	percentage_of_time_with_abnormal_long_term_variability	abnormal_short_term_variability	fetal_movement
3	accelerations	abnormal_short_term_variability	mean_value_of_short_term_variability	uterine_contractions
4	prolongued_decelerations	histogram_mean	percentage_of_time_with_abnormal_long_term_variability	light_decelerations
5	baseline_value	histogram_variance	histogram_max	abnormal_short_term_variability

	Correlation	Information Gain	Wrapper Forwards	Wrapper Backwards
6	uterine_contractions	histogram_mode	histogram_number_of_peaks	percentage_of_time_with_abnormal_long_term_variability
7	mean_value_of_short_term_variability	accelerations	histogram_mean	histogram_max

Correlation: Using Pearson's correlation coefficient results in feature selection outputs dependent on that features correlation to the target feature. The top five features that were returned to me in weka using this technique had high correlation values from 0.4558 to 0.2476 (where 1 is the highest possible value and -1 the lowest). After the listed features correlation results were below 0.2 and therefore probably less valuable for predictions.

Information Gain: Ranking features based on entropy using the information gain filter resulted in a different collection of top 5 features with only the second place on matching the first filter, and only one other feature (ranked 3) being common to both. While the two methods do not rank the features returned by the other as completely irrelevant, they are given mid-range values. Information Gain is ranked from 1 to 0, and the top features returned ranged from 0.30218 to 0.19794 after which features received below 0.2.

Wrapper Backwards: This method took a while to complete due to its complexity and higher computational cost. Again, it returned a new collection of top features of which only two were not returned in the top seven for the previous methods. It is worth noting that this method returned a list of 7 total features as being most valuable, not more.

Wrapper Forwards: This method again took some time and resulted in a list of nine top features of which I listed the top 7. Of these there were again two new ones added to the total collection of features returned by the different methods.

Common Features:
While there was some variance in the features returned by each filtering/ranking method, several features were returned by all or at least 3 out of 4. These seem to be promising given that they are consistently returned as being valuable in the prediction of the target feature. In no particular order the common features for all are:

abnormal_short_term_variability
percentage_of_time_with_abnormal_long_term_variability

And the features returned by three out of four methods are:

accelerations
uterine_contractions
mean_value_of_short_term_variability

1.6 - Classification with Top 5 Features only

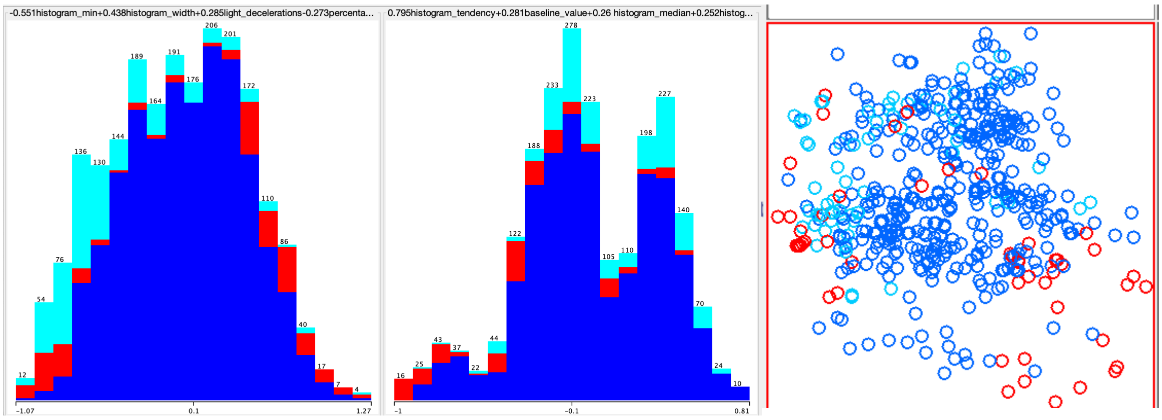
Using the top five features of the different feature groups created in task 1.5, ML models trained on each one returned the following results in terms of accuracy %:

Features Used:	Correlation	Info Gain	Wrapper Forwards	Wrapper Backwards	Common
k-NN	90.7801%	91.0165%	89.409%	86.7139%	90.0709%
Naive Bayes	84.3499%	82.7896%	81.0402%	78.0142%	79.9054%
Decision Tree	92.9551%	92.3404%	91.4421%	88.8416%	90.591%

While there is some variance in the accuracy results returned by the different reduced datasets, overall the differences are not very substantial. Overall the Decision Tree model returned the best of the three model results regardless of which dataset it was working with, and out of all datasets it was the correlation one that returned the highest results. When comparing results across the different feature sets both of the sets using features returned by the wrapper functions performed the worst by a small margin, and the common top features collection did not have great results either. The decision tree models capability to handle non-linear relationships well as well as representing feature interaction successfully likely resulted in its overall better performance.

In order to be able to compute more accurate predictions, a useful step might be to derive new features based on the original 21 so that their data can be used without compromising the capabilities of a machine learning model by having too many features to work with, rather than simply dropping them entirely and losing the data they provide. Also given the data using only 5 features may not be optimal for this specific dataset, as the wrapper backwards search identified a minimum of seven features as being the most valuable, and the wrapper forwards search an even higher count at nine.

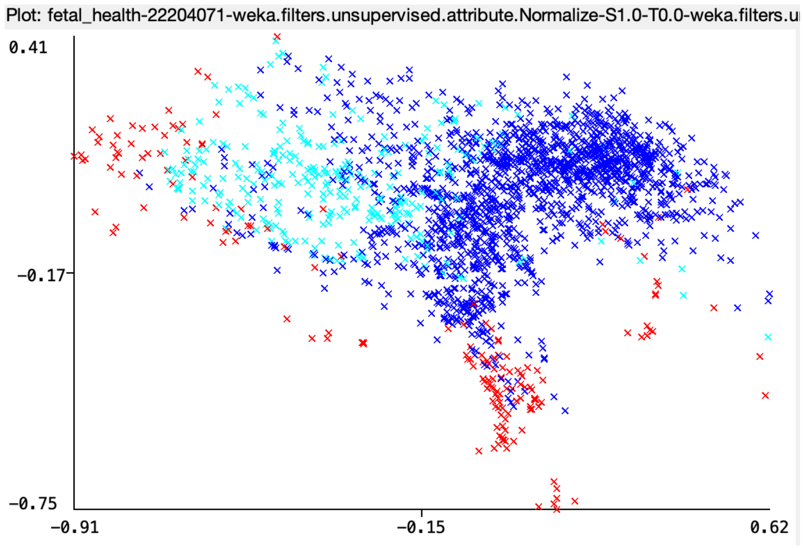
1.7 - Principle Components



2 PCA from original set

Using PCA to reduce the dimensionality of the original dataset results in combined features and data which keeps the variance as intact as possible.

The Eigenvalues of the features returned by PCA range from 0.32918 to 0.01146. Even the highest of these is not particularly great but only so in comparison with the worst. This indicates that perhaps these PCAs are not ideal. The results of applying k-NN, Naive Bayes and Decision Tree models to the set of data reduced by using these PCAs also indicate an unsuitability of this method applied in this way to the dataset we are working with. Accuracy is low compared with the previously trained models, with k-NN at 84.3499%, Naive Bayes at 81.7021% and Decision Trees at 84.1135%.



2 PCA from InfoGain set

Overall the extremely reduced dimensionality seems to not be a great solution for this dataset as it combines too many differently significant values when applied to the original 21 feature dataset.

When applying the same PCA filter to the most successful dataset from the previous task, the InfoGain collection of features, the model predictions are slightly better than outlined in the step above, and the slightly more distinguished target feature results can be seen when looking at the following visualisation of the further reduced dataset as well.