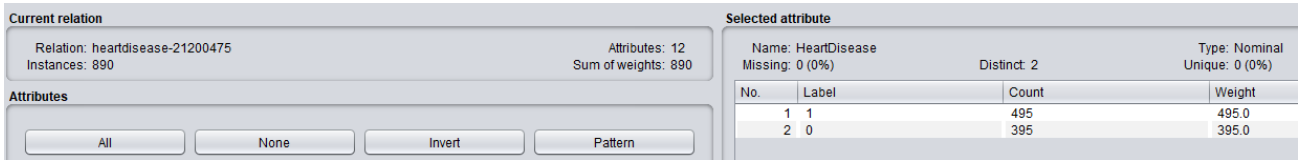
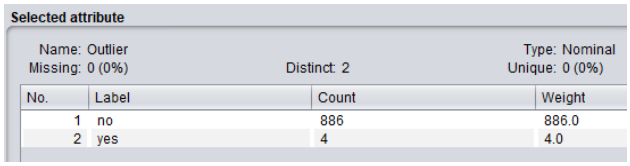
All 6 questions have been answered (Page 1-10). Page 11 consist of some additional explorations on Weka

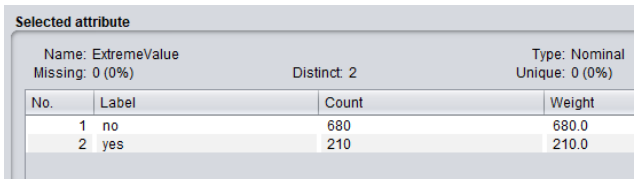
Answers for Question 1.1

1) No missing values were found in any of the features or target class variable
2) Outliers and extreme values were removed using interquartile range filter
3) I used “normalize” method in Weka to scale all numeric columns between scale 0 to 1

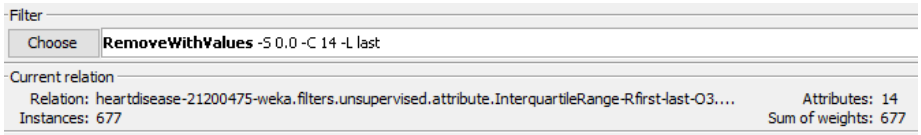
Before removing outlier and extreme values

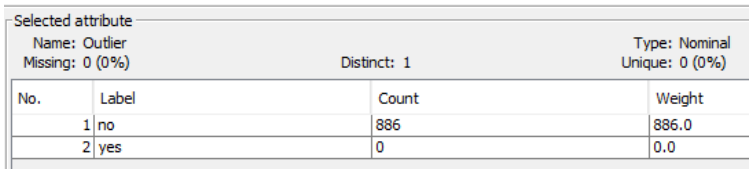


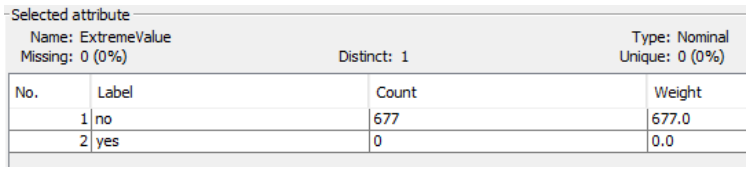




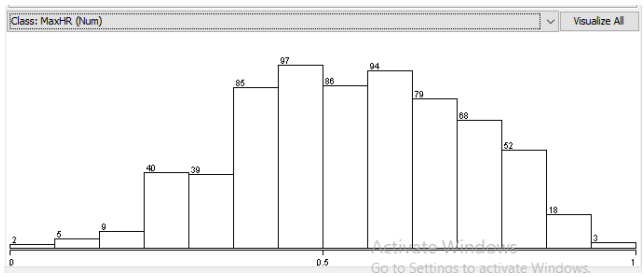
After removing outlier and extreme values







After normalization all numeric features



Answers for Question 1.2

I used 3 classifiers - KNN, Decision Trees, Naïve Bayes
I could find 100% accuracy (screenshot below) in all 3 models using k = 5 fold cross validation.

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               1.000    ?        1.000      1.000    1.000      ?        ?         1.000     no
               ?        0.000    ?          ?        ?          ?        ?         ?         yes
Weighted Avg.  1.000    ?        1.000      1.000    1.000      ?        ?         1.000

=== Confusion Matrix ===

   a   b   <-- classified as
 677   0 |   a = no
   0   0 |   b = yes
```

My initial thought was that I overfitted the model.
Hence, I resorted to redo step 1.1 but with outlier removal + normalization (with the exclusion of extreme values removal)

Accuracy metrics post redoing the step 1.1

I chose the Decision Tree model because of higher accuracy scores compared to other models and importantly higher recall rate in Class 1. Reason - For prediction of heart disease, it is necessary to reduce Type 2 error in Class 1 prediction *(i.e., Not predicting the heart disease whereas one has heart disease).*

KNN has least average accuracy and recall rates followed by Naïve Bayes.

Also, for k = 4 cross validation and 70-30% train test split validation methodologies, the Accuracy and recall rates are higher for Decision Tree model.

	70-30% train test split			k=4 cross validation			k=5 cross validation		
	Precision	Recall	Accuracy	Precision	Recall	Accuracy	Precision	Recall	Accuracy
Naïve Bayes	0.869	0.868	86.8421	0.863	0.863	86.3431	0.854	0.854	85.4402
kNN	0.85	0.85	84.9624	0.832	0.832	83.1828	0.816	0.816	81.6027
Decision Tree (J48)	0.876	0.876	87.594	0.866	0.866	86.5688	0.855	0.856	85.553

Class 1

	70-30% train test split		k=4 cross validation		k=5 cross validation	
	Precision	Recall	Precision	Recall	Precision	Recall
Naïve Bayes	0.893	0.876	0.873	0.882	0.865	0.874
kNN	0.869	0.869	0.848	0.85	0.839	0.827
Decision Tree (J48)	0.885	0.902	0.866	0.896	0.857	0.888

Class 0

	70-30% train test split		k=4 cross validation		k=5 cross validation	
	Precision	Recall	Precision	Recall	Precision	Recall
Naïve Bayes	0.836	0.858	0.851	0.84	0.841	0.83
kNN	0.823	0.823	0.81	0.81	0.788	0.802
Decision Tree (J48)	0.864	0.841	0.865	0.827	0.854	0.815

Answers for Q1.3

(i) Reporting of feature subsets for both filter and wrapper techniques - *screenshots attached below in blue and green sections*

Proposal for subset of features in filter method - ***Please refer to the section below in blue***

Proposal for subset of features in wrapper method - ***Please refer to the section below in green and the notes***

For wrapper method feature selection for one Decision Tree, one Naïve Bayes, and one kNN classifier –
Please refer to section - Comparison among wrapper methods for Naïve Bayes, kNN and Decision Tree
Decision Tree – 8 features, Naïve Bayes – 5 features, kNN – 1 feature

(ii) Differences between filter and wrapper method include:

- Filter method did not give conclusive evidence of best k=4 features as multiple filter methods suggested different ranking order and values. However, wrapper methods (forward selection, backward elimination, best fit) suggested 8 features with merit of best subset being >0.8 in each of the classifiers.
- Filter method did not recognize the classification model I used. However, wrapper method allowed me to use that flexibility.
- Any bias would be removed in wrapper method due to the selection of classification algorithm (Decision Tree with 8 features in my case) when compared against filter method.

(iii) Evaluation measures used for Filter method –> Information Gain > 50% of max Information Gain
Evaluation measures used for Wrapper method –> Top k =8 features

Justification - Feature selection suggestion for kNN (1 feature) and Naïve Bayes (5 features) would underfit the model. Accuracy and recall metrics for Decision Tree was higher compared to kNN and Naïve Bayes.
Hence, I chose to take top k=8 features for Decision trees in wrapper method and for filter method - Information Gain > 50% of max Information Gain as the best way to proceed.

Feature subsets selected in Filter method:

1) Classifier Attribute – with 'Decision Tree': ST_Slope, ChestPainType, ExerciseAngina, OldPeak
2) Information Gain: ST_Slope, ChestPainType, ExerciseAngina, OldPeak
3) Relief algorithm: ST_Slope, ChestPainType, RestingECG, Sex

Reason:
1) Top 4 ranked features have IG > 50% of max IG
2) Classifier Attribute and Information Gain approaches have resulted in same ranking output for top 4 features

Additionally, I chose to also experiment with "Relief algorithm" because it works on both discrete and continuous data.

Relief Algorithm

```
Ranked attributes:
 0.1767   3 ChestPainType
 0.1692  11 ST_Slope
 0.1068   7 RestingECG
 0.0857   2 Sex
 0.0552   5 Cholesterol
 0.0437   9 ExerciseAngina
 0.0379   6 FastingBS
 0.0379  14 ExtremeValue
 0.0258   1 Age
 0.0228   8 MaxHR
 0.0203  10 Oldpeak
 0.02     4 RestingBP
 0       13 Outlier

Selected attributes: 3,11,7,2,5,9,6,14,1,8,10,4,13 : 13
```

Classifier Attribute

```
Ranked attributes:
 0.2585  11 ST_Slope
 0.2077   3 ChestPainType
 0.1749   9 ExerciseAngina
 0.1462  10 Oldpeak
 0.1117   8 MaxHR
 0.1005   2 Sex
 0.0649   1 Age
 0.0305  14 ExtremeValue
 0.0305   6 FastingBS
 0.0299   5 Cholesterol
 0        4 RestingBP
 0       13 Outlier
 0        7 RestingECG

Selected attributes: 11,3,9,10,8,2,1,14,6,5,4,13,7 : 13
```

Info Gain - full dataset

```
Ranked attributes:
 0.2994   11 ST_Slope
 0.22856   3 ChestPainType
 0.18837   9 ExerciseAngina
 0.15841  10 Oldpeak
 0.12639   8 MaxHR
 0.08652   5 Cholesterol
 0.06861   2 Sex
 0.06789   1 Age
 0.05718  14 ExtremeValue
 0.05718   6 FastingBS
 0.01398   4 RestingBP
 0.0096    7 RestingECG
 0        13 Outlier

Selected attributes: 11,3,9,10,8,5,2,1,14,6,4,7,13 : 13
```

Info Gain - 6 fold

average merit	average rank	attribute
0.3 +- 0.014	1 +- 0	11 ST_Slope
0.229 +- 0.017	2 +- 0	3 ChestPainType
0.189 +- 0.013	3 +- 0	9 ExerciseAngina
0.156 +- 0.009	4 +- 0	10 Oldpeak
0.128 +- 0.005	5 +- 0	8 MaxHR
0.088 +- 0.007	6 +- 0	5 Cholesterol
0.069 +- 0.003	7.5 +- 0.76	2 Sex
0.064 +- 0.008	8.2 +- 0.9	1 Age
0.057 +- 0.006	8.7 +- 0.75	14 ExtremeValue
0.057 +- 0.006	9.7 +- 0.75	6 FastingBS
0.01 +- 0.003	11.3 +- 0.47	7 RestingECG
0.006 +- 0.008	11.7 +- 0.47	4 RestingBP
0 +- 0	13 +- 0	13 Outlier

Feature subsets selected in Wrapper method:

For wrapper method:
Decision Tree - Sex, ChestPainType, RestingBP, Cholestrol, FastingBS, ExerciseAngina, OldPeak, ST_Slope
kNN – ST_Slope
Naïve Bayes - Sex, RestingBP, Cholestrol, ExerciseAngina, ST_Slope

Considering both forward sequential selection and backward elimination, below were top 8 features.
Sex, ChestPainType, RestingBP, Cholestrol, FastingBS, ExerciseAngina, OldPeak, ST_Slope

I choose to select the above 8 out of total 11 original features (excluding target class) based on best of 2 approaches
1) Top ranked k=8 features
2) Ranking > 40% of max Information Gain

Notes:

1) RestingECG - although it appeared in Backward Elimination did not appear in forward selection, was ranked lower on Information Gain aspect. Hence, it was excluded in above final feature selection list.
2) Similarly other features like Age, MaxHR were ranked significantly lower in Information gain, hence excluded from feature selection.
3) Feature - ExtremeValue is excluded since it did not add any value to target class label - HeartDisease
4) I experimented on 3rd option of "Best Fit + Wrapper subset evaluation" which also gave 8 best features and were matching the final feature selection list. It added more as conclusive evidence.

Greedy stepwise (forward selection) & Wrapper subset evaluation gave 8 features

```
=== Attribute Selection on all input data ===

Search Method:
        Greedy Stepwise (forwards).
        Start set: no attributes
        Merit of best subset found:    0.867

Attribute Subset Evaluator (supervised, Class (nominal): 12 HeartDisease):
        Wrapper Subset Evaluator
        Learning scheme: weka.classifiers.trees.J48
        Scheme options: -C 0.25 -M 2
        Subset evaluation: classification accuracy
        Number of folds for accuracy estimation: 5

Selected attributes: 2,3,4,5,6,9,10,11 : 8
                     Sex
                     ChestPainType
                     RestingBP
                     Cholesterol
                     FastingBS
                     ExerciseAngina
                     Oldpeak
                     ST_Slope
```

Greedy stepwise (backwards elimination) & Wrapper subset evaluation gave 9 features

```
=== Attribute Selection on all input data ===

Search Method:
        Greedy Stepwise (backwards).
        Start set: all attributes
        Merit of best subset found:    0.871

Attribute Subset Evaluator (supervised, Class (nominal): 12 HeartDisease):
        Wrapper Subset Evaluator
        Learning scheme: weka.classifiers.trees.J48
        Scheme options: -C 0.25 -M 2
        Subset evaluation: classification accuracy
        Number of folds for accuracy estimation: 5

Selected attributes: 2,3,4,5,6,7,8,9,11,14 : 10
                     Sex
                     ChestPainType
                     RestingBP
                     Cholesterol
                     FastingBS
                     RestingECG
                     MaxHR
                     ExerciseAngina
                     ST_Slope
                     ExtremeValue
```

Best Fit & Wrapper subset evaluation also gave 8 features (additional exploration)

```
=== Attribute Selection on all input data ===

Search Method:
        Best first.
        Start set: no attributes
        Search direction: forward
        Stale search after 5 node expansions
        Total number of subsets evaluated: 93
        Merit of best subset found:    0.867

Attribute Subset Evaluator (supervised, Class (nominal): 12 HeartDisease):
        Wrapper Subset Evaluator
        Learning scheme: weka.classifiers.trees.J48
        Scheme options: -C 0.25 -M 2
        Subset evaluation: classification accuracy
        Number of folds for accuracy estimation: 5

Selected attributes: 2,3,4,5,6,9,10,11 : 8
                     Sex
                     ChestPainType
                     RestingBP
                     Cholesterol
                     FastingBS
                     ExerciseAngina
                     Oldpeak
                     ST_Slope
```

Comparison among wrapper methods for Naïve Bayes, kNN and Decision Tree

Naïve Bayes

```
=== Attribute Selection on all input data ===

Search Method:
        Greedy Stepwise (forwards).
        Start set: no attributes
        Merit of best subset found:    0.865

Attribute Subset Evaluator (supervised, Class (nominal): 12 HeartDisease):
        Wrapper Subset Evaluator
        Learning scheme: weka.classifiers.bayes.NaiveBayes
        Scheme options:
        Subset evaluation: classification accuracy
        Number of folds for accuracy estimation: 5

Selected attributes: 2,4,5,9,11 : 5
                     Sex
                     RestingBP
                     Cholesterol
                     ExerciseAngina
                     ST_Slope
```

kNN

```
=== Attribute Selection on all input data ===

Search Method:
        Greedy Stepwise (forwards).
        Start set: no attributes
        Merit of best subset found:    0.814

Attribute Subset Evaluator (supervised, Class (nominal): 12 HeartDisease):
        Wrapper Subset Evaluator
        Learning scheme: weka.classifiers.lazy.IBk
        Scheme options: -K 1 -W 0 -A weka.core.neighboursearch.LinearNNSearch -A "weka.core.EuclideanDistance -R first-last"
        Subset evaluation: classification accuracy
        Number of folds for accuracy estimation: 5

Selected attributes: 11 : 1
                     ST_Slope
```

Decision Tree

```
=== Attribute Selection on all input data ===

Search Method:
        Greedy Stepwise (forwards).
        Start set: no attributes
        Merit of best subset found:    0.867

Attribute Subset Evaluator (supervised, Class (nominal): 12 HeartDisease):
        Wrapper Subset Evaluator
        Learning scheme: weka.classifiers.trees.J48
        Scheme options: -C 0.25 -M 2
        Subset evaluation: classification accuracy
        Number of folds for accuracy estimation: 5

Selected attributes: 2,3,4,5,6,9,10,11 : 8
                     Sex
                     ChestPainType
                     RestingBP
                     Cholesterol
                     FastingBS
                     ExerciseAngina
                     Oldpeak
                     ST_Slope
```

Answers for Q1.4

Evaluate the performance of various classifiers (including at least one Decision Tree, one Naïve Bayes and one k-NN classifier) on your dataset using the feature subset(s) identified in 1.3.i. and evaluation measures identified in 1.3.iii.

		70-30% split				k=6 fold			
Filter method (4 features)		Class 1		Class 0		Class 1		Class 0	
		Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
	NB	0.864	0.869	0.821	0.814	0.85	0.866	0.829	0.81
	kNN	0.846	0.863	0.809	0.788	0.819	0.837	0.791	0.769
	J48	0.84	0.895	0.845	0.77	0.821	0.892	0.849	0.756
Wrapper method (8 features)		Class 1		Class 0		Class 1		Class 0	
		Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
	NB	0.894	0.882	0.843	0.858	0.873	0.89	0.859	0.838
	kNN	0.838	0.843	0.786	0.779	0.824	0.801	0.76	0.787
	J48	0.891	0.856	0.815	0.858	0.869	0.886	0.854	0.832

For Filter method:

Features selected include - ST_Slope, ChestPainType, ExerciseAngina, OldPeak

Result – I used 4 features to evaluate
Decision Trees gives highest recall rates (around 89%) for Class 1
Naïve Bayes gives highest recall rates (around 81%) for Class 0
However, if we go with combination of both precision and recall, Naïve Bayes would be the best solution

For wrapper method:

Decision Tree - Sex, ChestPainType, RestingBP, Cholestrol, FastingBS, ExerciseAngina, OldPeak, ST_Slope
kNN – ST_Slope
Naïve Bayes - Sex, RestingBP, Cholestrol, ExerciseAngina, ST_Slope

Result – I used all 8 features from to explore and evaluate
Naïve Bayes gives highest precision (88%) and recall rates (85%) for both Class 1 and Class 0
If we go with combination of both precision and recall, Naïve Bayes would be the best solution

Explore the effect of different parameter settings on these classifiers.

Below 4 parameters were changed for each of the 3 classifiers in Wrapper technique.

1) Naïve Bayes
useKernelEstimator = True as against FALSE previously, the metrics degraded on Precision and Recall for Class 1 and 0

useKernelEstimator = False				useKernelEstimator = True			
Class 1		Class 0		Class 1		Class 0	
Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
0.894	0.882	0.843	0.858	0.893	0.876	0.836	0.858

2) Naïve Bayes
useSupervisedDescritization = True and debug = True as against FALSE previously, recall improved for Class 1

useSupervisedDescritization = False and debug = False				useSupervisedDescritization = True and debug = True			
Class 1		Class 0		Class 1		Class 0	
Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
0.894	0.882	0.843	0.858	0.891	0.908	0.873	0.85

3) kNN
kNN = 4 and crossValidate = True → high improvement seen in both precision and recall for both classes 0 & 1

kNN = 1 and crossValidate = False				kNN = 4 and crossValidate = True			
Class 1		Class 0		Class 1		Class 0	
Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
0.838	0.843	0.786	0.779	0.892	0.915	0.881	0.85

4) J48 – Decision Tree
reducedErrorPrunning = True → improvement seen in recall for class 1, but degrade in precision
Degrade seen in recall for class 0, but significant improvement in precision (by 10%)

reducedErrorPrunning = False				reducedErrorPrunning = True			
Class 1		Class 0		Class 1		Class 0	
Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
0.891	0.856	0.815	0.858	0.872	0.935	0.902	0.814

Describe the evaluation procedure that you used in detail.

The focus was to increase the recall to reduce Type 2 error (Not predicting the heart disease when one has it)

Initially only 4 features were selected in the Filter method based on Information Gain evaluation.
Although ranking was obtained for all 13 features, it was difficult to select which are the best features due to their closeness in ranking index numbers.

Post application of classification algorithms in Wrapper method, a combination of best of below 2 evaluation strategies were used.
1) Top k=8 features
2) Information Gain > 50% of max of IG.

Additionally, after hyper parameter optimization, I discovered certain new strategies to improvements recall factor in

1) J48 – decision tree using "reducedErrorPrunning = True"
2) kNN using "kNN = 4 and crossValidate = True"

Answers for Q1.5

To what extent are these results in line with or different from what you learnt about these classifiers in your lectures? For example, is your accuracy higher or lower on the dataset with reduced number of features as compared to the original dataset?

Below are the accuracy metrics before and after feature selection:

	Accuracy Metrics			
		70-30% split	k=4 cross validation	k=6 cross validation
Wrapper method (8 features)	Naïve Bayes	88.34	86.79	86.68
	kNN	81.57	80.92	79.45
	J48	85.71	87.02	86.23
		70-30% split	k=4 cross validation	k=6 cross validation
Before feature selection	Naïve Bayes	86.84	86.34	85.44
	kNN	84.96	83.18	81.60
	J48	87.59	86.57	85.55

For 70-30% split in train – test,
-accuracy improved for Naïve Bayes by 2%
-accuracy reduced for kNN by 3%
-accuracy reduced for J48 by 2%

For k=6 fold cross validation,
-accuracy improved for Naïve Bayes by 1%
-accuracy reduced for kNN by 3%
-accuracy improved for J48 by 2%

The results are good according to me since maximum reduction of accuracy goes up to only 3% and improvement goes up to 2% post feature selection for some classfiers.
Reason - We were successfully able to remove 3 irrelevant features from our original dataset and retain top 8 features.
For some classifiers like Naïve Bayes and J-48 we were able to improve model accuracy by 2%

Is the relative performance of different classifiers and configuration settings in line with your expectation?

Yes, the relative performance of different classifiers is in line with my expectation since they hover around 80-85% accuracy levels with improvement in recall factors.

I was hoping to have a larger dataset. It would be great to understand bias – variance trade off in such larger datasets.

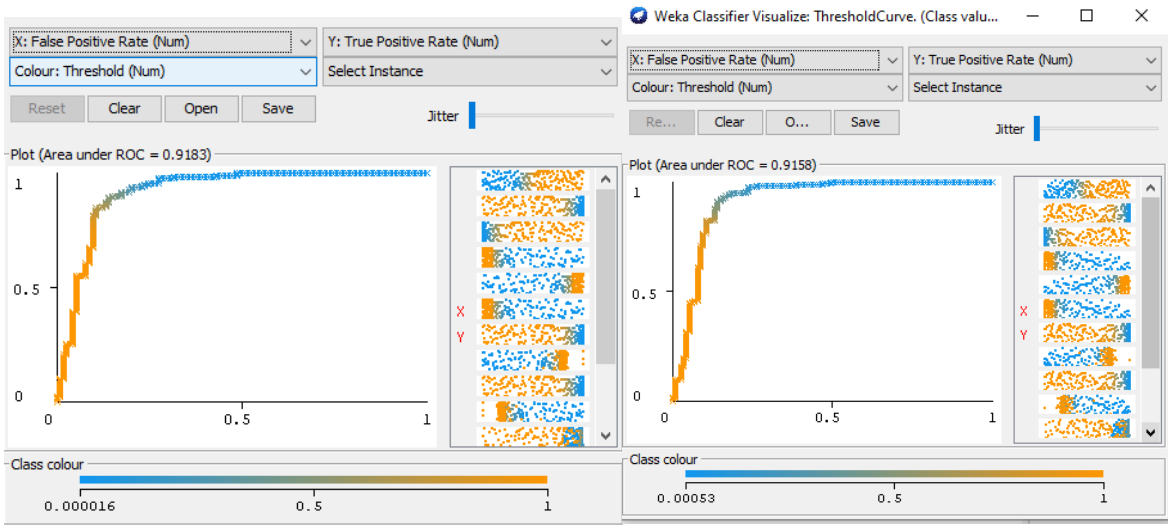
I also would try Random Forest and compare the statistical measures (precision, recall, accuracy, true positive rate, false positive rate etc). Maybe boosting and some ensemble techniques would perhaps better the model accuracy.

Answers for Q1.6

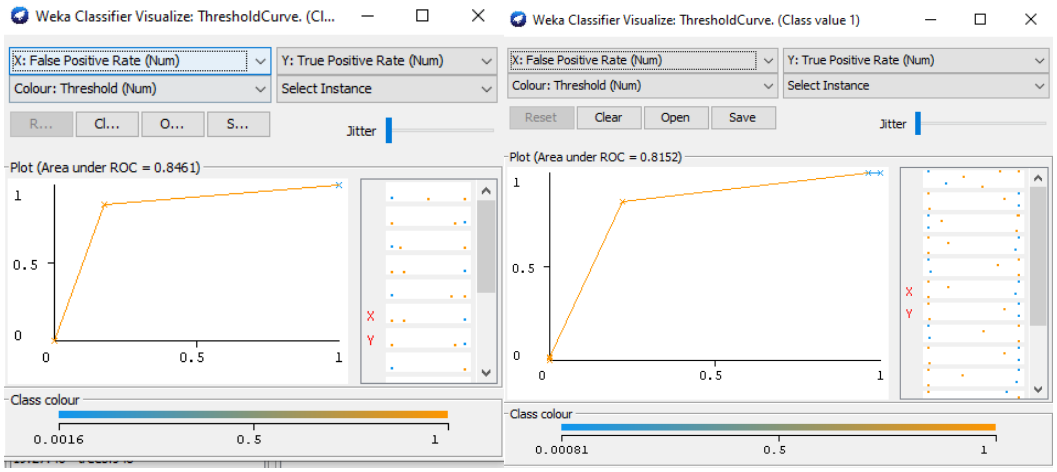
i. Plot the ROC curves for 3 different classification models (Weka does this).

Below are ROC curves for class 1 before feature selection (left side) and after applying feature selection (right side)

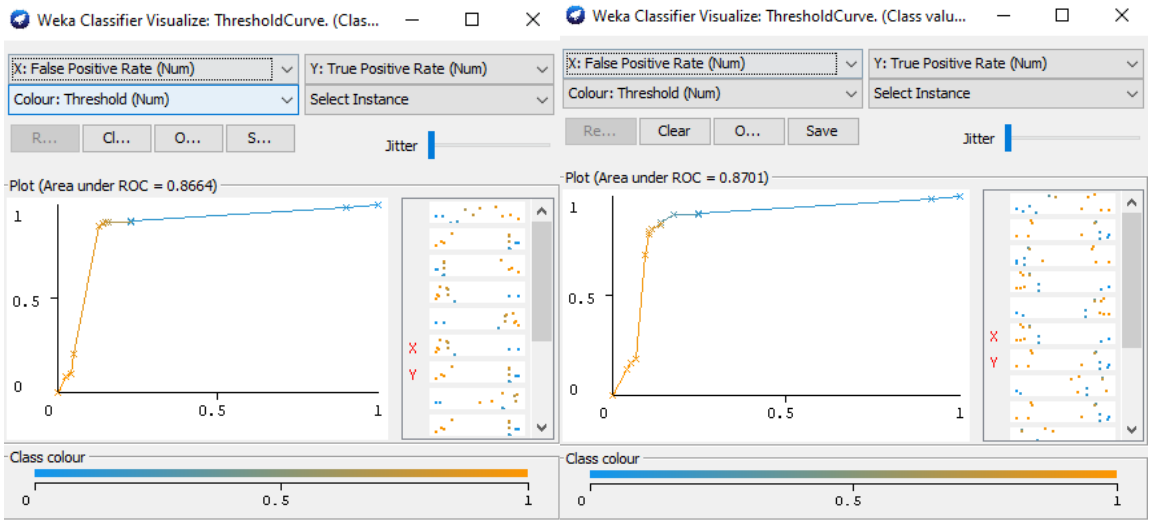
Naïve Bayes – ROC



kNN – ROC



J48 – Decision Tree



ii. What do you learn from these ROC curve? Include the AUC in your discussion.

The closer the ROC curve bends towards Y axis with value = 1, recall / sensitivity factor for predictions will improve significantly in medical devices. Consequently, Area under curve (AUC) would increase when this happens

In a real world, AUC above 0.80/0.85 can be considered a reasonably good metric to certify the model.

When Recall will be higher, it acts a good evidence that Type 2 errors for medical disease predictions in hospitals can be reduced (in this case – Heart Disease).

iii. Which classifier/configuration is best suited for this task? Are you satisfied with the performance?

Naïve Bayes is best suited for this task since AUC for before and after feature selection has remained above 91%.

I am satisfied with the performance since 2 algorithms (Naïve Bayes and J48 – Decision Tree) see slight improvement of 0.2% and 1% respectively in AUC post feature subset selection.

Data Saved

Name	Date modified	Type	Size
heartdisease_21200475.arff	20-10-2021 12:34	ARFF Data File	35 KB
heartdisease_21200475_iq.arff	30-10-2021 00:46	ARFF Data File	41 KB
heartdisease_21200475_iq_outlier_removed.arff	30-10-2021 00:50	ARFF Data File	41 KB
heartdisease_21200475_iq_outlier_and_extreme_values_removed.arff	30-10-2021 00:56	ARFF Data File	32 KB
heartdisease_21200475_normalized.arff	30-10-2021 01:13	ARFF Data File	48 KB
heartdisease_21200475_iq_outlier_removed_and_normalized.arff	30-10-2021 01:32	ARFF Data File	62 KB
heartdisease_21200475_initial_classification.arff	30-10-2021 12:33	ARFF Data File	62 KB
heartdisease_21200475_after_q3.arff	30-10-2021 20:18	ARFF Data File	62 KB
heartdisease_21200475_best_filter.arff	31-10-2021 13:45	ARFF Data File	20 KB
heartdisease_21200475_best_wrapper.arff	31-10-2021 14:03	ARFF Data File	36 KB

Initial Data Visualization

Jitter: While visualizing the dataset, I had lot of dots overlaying each other and it is hard to see what is going on. Hence, I used "Jitter" which added some random noise to the data in the plots.

The purpose of using jitter was to spread out the points clearly and help see what is going on (example: outliers). Below is the screenshot after using jitter.