2023

Data Mining Coursework

AUTHOR: B924007

22COP529 DATA MINING

Part 1

The Indian Liver Patient Dataset (ILPD) contains 10 attribute measures for 583 patients, as well as whether the patient does or does not have liver disease (class).

Based on the raw dataset, the following cleaning procedures have been carried out:

1) Firstly, column headers were added to the dataset, as it was unclear what each column was representing, nor was it possible for WEKA to understand the dataset in its raw form. This was done on excel by adding the names of each column above data.

A	B	C	D	E	F	G	H	I	J	K
Age	Gender	TB	DB	Alkphos	Sgpt	Sgot	TP	ALB	A/G	Class
65	Female	0.7	0.1	187	16	18	6.8	3.3	0.9	1
62	Male	10.9	5.5	699	64	100	7.5	3.2	0.74	1

Figure 1 - Column Headers added into the Dataset.

2) One of the most important aspects of data cleaning is the handling of missing data. Missing data is dangerous as it can lead to bias in results. There are several ways to handle missing data such as removal of entire row in the dataset, imputing the missing values, or using statistical techniques to estimate the missing value. The chosen method is highly dependent on the context of the data and the size of the data set.

   Using the 'find' feature in Excel, 10 missing data points were identified in the dataset. As the dataset contains sensitive data used to make important decisions in determining if a patient has liver disease or not, it is dangerous to impute the missing values by adding a mean or median value to them. If imputed data is incorrect or biased, it can lead to incorrect diagnoses, treatment decisions, and outcomes. Therefore, rows in ILPD which contained a missing data point were removed from the dataset, as this is the safer approach to handling the missing data in this context. This was done on Excel using 'Find' to identify a missing value, then deleting the entire row.

145	45	Female	3.5	1.5	189	63	87	5.6	2.9	1	Liver_disease
146	65	Male	0.8	0.2	201	18		5.4	2.9	1.1	No_liver_disease
147	66	Female	2.9	1.3	168	21	38	5.5	1.8	0.4	Liver_disease

Figure 2 - Example of a row with a missing value

3) Handling categorial attributes and classes was also carried out, so that nominal data such as Gender (Female/Male) and Class (No_liver_disease, Liver_Disease) are converted into 0 and 1 respectively. This is done as many ML algorithms require numeric inputs as they work with mathematical equations. This conversion allows algorithms to process the data more easily, an example being making use of Euclidean distance, which is a measure between two points in n-dimensional space, commonly used in clustering and classification algorithms. The conversion of nominal to numeric data was done on Excel using 'Find & Replace' to assign values of 0 and 1 to Gender and Class values.

Class
1
1
1
1
1
1
1
1
0

Figure 3 - Class converted to Numeric format.

4) While handling outliers is a typical task in cleaning data, outliers and irregular data points are not necessarily errors in medical data, as they can provide an explanation for an event that can significantly impact a patient such as a spontaneous spike in a measure that could cause a patient's life to be in danger. It is difficult to determine which outlier is an error that should be removed, and which outlier has an explanation and should be kept. For this reason, outliers have not been removed from the dataset as an outlier is not necessarily an error in medical context, hence cannot be confidently removed from the dataset unless a medical expert can confirm the value to be a certain outlier. Similarly, there are duplicate values in the dataset too, however these will be kept in as we cannot be 100% sure these are duplicate rows, or if the 2 different patients just have the same data. Hence, it is safer to keep all the data (including outliers and duplicates).

Part 2

(a) Methodology & Data/Tools used and justification.

Using linear regression and 3 other ranking method algorithms, it is possible to determine which attributes are more significant than others in determining the class.

Firstly, the clean data was standardised using normalisation. The raw data is measured with different units and scales, making it difficult to analyse. Normalising the data set makes it so each attribute has a common scale range (0 to 1). This is particularly useful in linear regression as it helps improve model accuracy and reduce any bias. Normalisation is done on WEKA using the 'Normalise' filter under the 'Pre-process' tab.

The linear regression models output provides an equation with constants that are comparable between attributes. The greater a constants absolute value (regardless of positive or negative), the more significant effect it has to the class determination. Linear regression was applied to the model using WEKA -> Classify -> Classifier -> Functions -> Linear regression (No attribute selection).

Using WEKA -> Select Attributes -> Search Method -> Ranker, the first chosen attribute evaluator is 'ClassifierAttributeEval' which evaluates the worth of an attribute by using a user-specified classifier. The chosen classifier here is kept as the default 'ZeroR'. Running this test gives a list of ranked attributes. Similarly, the second attribute evaluator tested is 'ReliefAttributeEval' which also generates a list of ranked attributes. This evaluator method determines the worth of an attribute by repeatedly sampling an instance to consider the value of an attribute for the nearest instance of the same and different class. The third ranking test is the 'CorrelationAttributeEval' which evaluates the worth of an attribute by measuring its correlation with 'class'.

(b) Experimental Results

Linear regression Ranking:

```
Linear Regression Model

Class =

      0.2642 * Age +
      0.0609 * Gender +
      0.8214 * TB +
     -0.0837 * DB +
      0.4651 * Alkphos +
      0.4682 * Sgpt +
      0.4494 * Sgot +
     -0.4381 * TP +
     -0.1064 * ALB +
     -0.7628 * A/G +
      0.5343

Time taken to build model: 0 seconds

=== Cross-validation ===
=== Summary ===

Correlation coefficient                  0.1372
Mean absolute error                      0.3967
Root mean squared error                  0.4964
Relative absolute error                 97.3625 %
Root relative squared error            109.9398 %
Total Number of Instances              573
```

Classifier Attribute Eval Ranking:

```
Ranked attributes:
  0  10 A/G
  0   3 TB
  0   2 Gender
  0   9 ALB
  0   4 DB
  0   5 Alkphos
  0   6 Sgpt
  0   7 Sgot
  0   8 TP
  0   1 Age

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

Relief Attribute Eval Ranking:

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

Search Method:
        Attribute ranking.

Attribute Evaluator (supervised, Class (numeric): 11 Class):
        ReliefF Ranking Filter
        Instances sampled: all
        Number of nearest neighbours (k): 10
        Equal influence nearest neighbours

Ranked attributes:
 0.0018823  10 A/G
 0.00001     8 TP
 0           2 Gender
-0.0025474   1 Age
-0.0057777   4 DB
-0.0067309   9 ALB
-0.008947    6 Sgpt
-0.0103802   7 Sgot
-0.0135975   3 TB
-0.0139739   5 Alkphos

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

Correlation Attribute Eval Ranking:

```
Attribute Evaluator (supervised, Class (numeric): 11 Class):
        Correlation Ranking Filter
Ranked attributes:
 0.2158   3 TB
 0.1879   5 Alkphos
 0.1514   4 DB
 0.15     7 Sgot
 0.1364   1 Age
 0.1282   6 Sgpt
 0.0881   2 Gender
-0.0727   8 TP
-0.0994  10 A/G
-0.1561   9 ALB

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

(c) Results Analysis

Based on the linear regression model, TB, A/G, Sgpt are the most significant factors and ALB, DB, Gender are the least significant factors. Classifier and Relief models have many similarities in ranking, although have TB and TP in opposite positions. Most the algorithms have similar attributes towards the bottom of the ranks such as ALB, TP, Sgot. All four algorithms have very different ranks, making it difficult to find a pattern through which the most significant attributes can be determined. Therefore, the most suitable attribute ranking algorithm is the one with the most similarities to the others. It is evident that that since 'ClassifierAttributeEval' and 'ReliefAttributeEval' produce a very similar ranking order, the chosen rank system to move forward with is one of these two. Comparing these two to 'Linear regression' and 'CorrelationAttributeEval', ClassifierAttributeEval is the best ranking of the features as it has more similarities than 'ReliefAttributeEval'. Some of these similarities include: TB has a high rank, Alkphos is relatively high, and TP is a lower rank (all of these similarities are not shared with 'ReliefAttributeEval').

Table 1- Attribute Rankings (Best[Top] to Worst[Bottom]) using different ranking algorithms.

Linear Regression	***ClassifierAttributeEval***	***ReliefAttributeEval***	***CorrelationAttributeEval***
TB	A/G	A/G	TB
A/G	TB	TP	Alkphos
Sgpt	Gender	Gender	DB
Alkphos	ALB	Age	Sgot
Sgot	DB	DB	Age
TP	Alkphos	ALB	Sgpt
Age	Sgpt	Sgpt	Gender
ALB	Sgot	Sgot	TP
DB	TP	TB	A/G
Gender	Age	Alkphos	ALB

Part 3

(a) Methodology & Data/Tools used and justification.

The clean data set is slightly altered to change class back to a nominal data type (0 -> No Liver disease, 1 -> Liver Disease) as classifiers such as Naïve Bayes, J48, JRip are not available on WEKA if the class is numeric. A t-test (using 'Experimenter' on Weka) cannot be carried out without nominal classes either.

The two chosen classifiers to apply to the original data set using 10-fold-cross-validation are Naïve Bayes and Multilayer Perceptron (MLP).

Naïve bayes is selected as it is a fast and easy classification method that outperforms other classifiers, while not even needing a significantly large training dataset. MLP is selected as a classifier, as it is an algorithm capable of solving complex non-linear problems, handling large data, and can make quick predictions after training.

The modified dataset contains fewer attributes compared to the original dataset. Using the ranking order in part **(i)** chosen using 'ClassifierAttributeEval', the bottom 3 ranking attributes are removed. Hence the modified dataset will contain 7 attributes, where Sgot, TP, and Age have been excluded.

(b) Experimental Results

Original Dataset: Naïve Bayes

```
=== Summary ===

Correctly Classified Instances         335               58.4642 %
Incorrectly Classified Instances       238               41.5358 %
Kappa statistic                          0.2727
Mean absolute error                      0.4145
Root mean squared error                  0.606
Relative absolute error                101.7304 %
Root relative squared error            134.3159 %
Total Number of Instances              573

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.444    0.061    0.948      0.444    0.605      0.366    0.698     0.873     Liver_disease
                 0.939    0.556    0.402      0.939    0.563      0.366    0.699     0.392     No_liver_disease
Weighted Avg.    0.585    0.202    0.793      0.585    0.593      0.366    0.698     0.736

=== Confusion Matrix ===

   a   b   <-- classified as
 182 228 |   a = Liver_disease
  10 153 |   b = No_liver_disease
```

Original Dataset: Multilayer Perceptron

```
=== Summary ===

Correctly Classified Instances         394               68.7609 %
Incorrectly Classified Instances       179               31.2391 %
Kappa statistic                          0.0391
Mean absolute error                      0.3541
Root mean squared error                  0.4367
Relative absolute error                 86.9029 %
Root relative squared error             96.7925 %
Total Number of Instances              573

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.915    0.883    0.723      0.915    0.807      0.048    0.679     0.850     Liver_disease
                 0.117    0.085    0.352      0.117    0.175      0.048    0.679     0.383     No_liver_disease
Weighted Avg.    0.688    0.656    0.617      0.688    0.627      0.048    0.679     0.717

=== Confusion Matrix ===

   a   b   <-- classified as
 375  35 |   a = Liver_disease
 144  19 |   b = No_liver_disease
```

Modified Dataset: Naïve Bayes

```
=== Summary ===

Correctly Classified Instances         330               57.5916 %
Incorrectly Classified Instances       243               42.4084 %
Kappa statistic                          0.2545
Mean absolute error                      0.4182
Root mean squared error                  0.5866
Relative absolute error                102.6408 %
Root relative squared error            130.0126 %
Total Number of Instances              573

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.441    0.086    0.928      0.441    0.598      0.339    0.698     0.873     Liver_disease
                 0.914    0.559    0.394      0.914    0.551      0.339    0.698     0.394     No_liver_disease
Weighted Avg.    0.576    0.220    0.776      0.576    0.585      0.339    0.698     0.737

=== Confusion Matrix ===

   a   b   <-- classified as
 181 229 |   a = Liver_disease
  14 149 |   b = No_liver_disease
```

Modified Dataset: Multilayer Perceptron

```
=== Summary ===

Correctly Classified Instances         399               69.6335 %
Incorrectly Classified Instances       174               30.3665 %
Kappa statistic                         -0.0013
Mean absolute error                      0.3558
Root mean squared error                  0.4307
Relative absolute error                 87.3342 %
Root relative squared error             95.4632 %
Total Number of Instances              573

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.956    0.957    0.715      0.956    0.818      -0.002   0.686     0.858     Liver_disease
                 0.043    0.044    0.280      0.043    0.074      -0.002   0.686     0.388     No_liver_disease
Weighted Avg.    0.696    0.697    0.591      0.696    0.607      -0.002   0.686     0.725

=== Confusion Matrix ===

   a   b   <-- classified as
 392  18 |   a = Liver_disease
 156   7 |   b = No_liver_disease
```

Original Dataset: t-test

Test output

```
Tester:     weka.experiment.PairedCorrectedTTester -G 4,5,6 -D 1 -R 2 -S 0.05 -result-matri
Analysing:  Percent_correct
Datasets:   1
Resultsets: 2
Confidence: 0.05 (two tailed)
Sorted by:  -
Date:       23/02/2023, 10:25


Dataset                   (1) bayes.Na | (2) funct
----------------------------------------------------
'Indian Liver Patient Dat(100)   58.13 |   69.34 v
----------------------------------------------------
                               (v/ /*) |   (1/0/0)


Key:
(1) bayes.NaiveBayes '' 5995231201785697655
(2) functions.MultilayerPerceptron '-L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a' -59906078170
```

Modified Dataset: t-test

Test output

```
Tester:     weka.experiment.PairedCorrectedTTester -G 4,5,6 -D 1 -R 2 -S 0.05 -result-matrix "weka.e
Analysing:  Percent_correct
Datasets:   1
Resultsets: 2
Confidence: 0.05 (two tailed)
Sorted by:  -
Date:       23/02/2023, 10:35


Dataset                   (1) bayes.Na | (2) funct
----------------------------------------------------
'Indian Liver Patient Dat(100)   57.75 |   70.08 v
----------------------------------------------------
                              (v/ /*) |   (1/0/0)


Key:
(1) bayes.NaiveBayes '' 5995231201785697655
(2) functions.MultilayerPerceptron '-L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a' -5990607817048210779
```

(c) Results Analysis

The Naïve Bayes for the original dataset under-represents patients with liver disease. It misclasses 228 Liver disease patients which is extremely dangerous as the data suggests most patients don't have liver disease. It claims that 192 patients have liver disease, when in reality it is 410. This test also has a low accuracy, high error %, and lower FP rate.

MLP for the original dataset has a better accuracy, and correctly classifies 375 liver disease patients, only misclassing 35 patients here. It also misclasses 144 'No_Liver_disease' patients as 'Liver_Disease', yielding a high FP rate. This is alright to an extent, as it is safer to overestimate the number of Liver disease patients than to underestimate it. That being said, a high FP rate is only good in comparison to a low TP rate where many liver disease patients are assumed to be non-liver-disease patients. So where possible, if the number of Liver disease patients is accurate or slightly higher, this is better than a model where it is extremely high, leading to unnecessary medical tests and procedures that waste time and increase costs.

The analysis of Naïve Bayes and MLP follows the same trend for the modified dataset.

MLP for the modified dataset has a greater accuracy and FP rate than the MLP for the original data, suggesting it is the best model of the 4 tests as it correctly classifies more liver disease patients, further confirming that the removed attributes may not be significant contributors to class identification.

A t-test is used to compare the two classifiers at a 95% significance level. For both the original and modified data, you can confidently say that the MLP test is a better test and is more statistically significant than the Naïve Bayes test as the test shows a 'v' next to the MLP value. Therefore, you can conclude that the MLP is a better classifier than Naïve Bayes for this particular dataset. The t-tests were also carried out at a 99.999% significance level, still showing MLP as the better performing classifier.

Part 4

(a) Methodology & Data/Tools used and justification.

The equal-width bins modified dataset is generated using the 'Discretise' filter under 'Pre-process' on WEKA (where the number of bins is kept default at 10 bins). The better performing classifier in part **(iii)** (Multilayer Perceptron) is applied to this discretised dataset. Running this test using 10-fold cross-validation (same as the previous part) will result in a summary of statistics including a confusion matrix and accuracy and error percentages to be compared.

(b) Experimental Results

Multilayer Perceptron Test using Discretised dataset:

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         381               66.4921 %
Incorrectly Classified Instances       192               33.5079 %
Kappa statistic                          0.1453
Mean absolute error                      0.3612
Root mean squared error                  0.4928
Relative absolute error                 88.6491 %
Root relative squared error            109.2274 %
Total Number of Instances              573

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.790    0.650    0.753      0.790    0.771      0.146    0.638     0.834     Liver_disease
                 0.350    0.210    0.399      0.350    0.373      0.146    0.638     0.393     No_liver_disease
Weighted Avg.    0.665    0.525    0.653      0.665    0.658      0.146    0.638     0.708

=== Confusion Matrix ===

   a   b   <-- classified as
 324  86 |   a = Liver_disease
 106  57 |   b = No_liver_disease
```

(c) Results Analysis

The multilayer perceptron test above was carried out on equal bin width (discretised) data, making use of the original clean dataset with 10 attributes. The figure above shows that 66% of instances were correctly classified. The FP rate was around 0.5 meaning that about half the 'No_Liver_disease' patients were classed as Liver disease patients. The confusion matrix also shows that 86 liver disease patients were misclassed, and 106 'No_Liver_Disease' patients were misclassed. Overall, this is fine as the total number of Liver disease patients in the model was 430, in comparison to the actual number, which was 410, making this a valid model that slightly over-represented the number of patients with liver disease. If the model underrepresented liver disease patients, the model would not be good as this can lead to poor actions being taken, potentially endangering the patients with liver disease.

The MLP performance for this discretised dataset can be compared to the MLP for the original dataset and the MLP for the selective attribute dataset from part **(iii)**. The test on the original dataset had a greater accuracy on correctly classified instances and slightly lower relative errors. It also had a lower mean and relative error, a higher FP rate, ROC, and PRC area, all of which contribute to a better model in this context. However, the confusion matrix of MLP for the original dataset shows that despite more Liver Disease patients being classed correctly, only 19 of 163 'No_Liver_disease' patients were classed correctly. This model is heavily over-representing liver

disease patients, perhaps to an extreme level. While it is not necessarily harmful to assume most patients have liver disease, comparing the values to the actual number of patients with liver disease, it shows that the model may be too skewed in favour of liver disease. A high false positive rate is not always good as it can lead to unnecessary testing, hence the discretised model with a lower FP rate is actually better in this case.

A similar conclusion can be drawn from the selective attribute dataset in part **(iii)** which has an even greater accuracy, but only correctly classes 7 'No Liver disease' patients. In these 2 cases, it seems that the MLP is overfitting the data towards liver disease. However, this overfitting could mean that the model would not be accurate in classing a random new dataset. The discretised model is better despite a lower accuracy %, as it does not overfit the data to the same extremity.

Equal bin width strategy is advantageous (although has downsides) in attempting to distribute data evenly to reduce any bias. Equal-width bins reduce noise in the data, allowing patterns and relationships to form between variables to identify trends and outliers. A skewed 'equal bin width' histogram could suggest that outliers exist at one extreme and should be removed or kept with justification. For example, TB has one extreme value that skews all the data. However, during pre-process, an explanation was given as to why such outliers are kept in the data in these occasions.

Part 5

(a) Methodology & Data/Tools used and justification.

The original dataset is used to carry out k-means clustering. This is found on WEKA -> Cluster -> Simple K means, using the Cluster Mode -> 'Classes to clusters evaluation'. Using this cluster mode, WEKA ignores the class attribute and generates clusters based on attribute data. It assigns classes to the clusters based on the majority value of the class attribute in each cluster. Based on this, the resulting output contains cluster details, a confusion matrix, and the number of incorrectly clustered instances. The number of clusters selected is 2 (equal to number of classes).

(b) Experimental Results (Simple k means clustering)

```
Final cluster centroids:
                        Cluster#
Attribute    Full Data          0          1
               (573.0)    (281.0)    (292.0)
=============================================
Age             44.9337    57.5694     32.774
Gender             Male       Male       Male
TB               3.2499     4.1121     2.4202
DB               1.5529     2.0402     1.0839
Alkphos        293.2688   329.7153   258.1952
Sgpt            89.9738     78.395   101.1164
Sgot           109.9808   105.6584   114.1404
TP                6.607     6.1064     7.0887
ALB               3.1414     2.7566     3.5116
A/G               1.0082     0.9529     1.0613



Time taken to build model (full training data) : 0.02 seconds

=== Model and evaluation on training set ===

Clustered Instances

0      281 ( 49%)
1      292 ( 51%)


Class attribute: Class
Classes to Clusters:

   0   1  <-- assigned to cluster
 219 191 | Liver_disease
  62 101 | No_liver_disease

Cluster 0 <-- Liver_disease
Cluster 1 <-- No_liver_disease

Incorrectly clustered instances :       253.0    44.1536 %
```

(c) Results Analysis

The Simple K means cluster algorithm attempts to class each instance into one of the two clusters using classes to clusters evaluation (hence, not using the existing class data).

The k means cluster evaluation above suggests that there is almost an even 50/50 split of instances between the 2 classes where 281 (49%) of instances were of Liver disease patients, and 292 (51%) of the patients did not have liver disease. This is an incorrect evaluation when compared to the available ground truth, as in actuality 573 (71.55%) patients had liver disease and 163 (28.45%) patients did not have liver disease. This immediately shows that the k means clustering algorithm has not done a great job in classifying instances into the correct clusters as 253 instances (44.15%) were classified incorrectly over the clustering test.

An analysis into the accuracy and suitability of this clustering method can be looked into deeper by looking at the resulting confusion matrix produced. The confusion matrix shows that 219 patients were correctly classed as liver patients, but 191 liver disease patients were misclassed as 'No liver disease' patients. Since medical data regarding diagnosing a patient with or without liver disease is extremely sensitive, underestimating the number of liver disease patients by 46% means the model is not very good at all. The model states that only 281 patients have liver disease, when the actual ground truth value using the original clean data set states that there are 410 patients with liver disease. The actual number of patients who did not have liver disease was 163, however, this clustering model increases that number to 292.

To conclude, the model assumes an almost 50/50 split of instances between liver disease and no liver disease. This means that the Simple k means algorithm is not good is classifying instances for this dataset.

Alternatively, using the 'FarthestFirst' clusterer (shown below), 466 (81%) of instances were seen to be Liver disease patients, and 107 (19%) were not liver disease patients. While this is not a perfect model either, it is definitely closer to the factual split from the original clean data of 71%/29%. Farthestfirst also has fewer incorrectly classed instances and a much better confusion matrix in comparison to Simple k means.

Farthest First Clustering

```
Clustered Instances

0      466 ( 81%)
1      107 ( 19%)


Class attribute: Class
Classes to Clusters:

   0   1  <-- assigned to cluster
 343  67 | Liver_disease
 123  40 | No_liver_disease

Cluster 0 <-- Liver_disease
Cluster 1 <-- No_liver_disease

Incorrectly clustered instances :       190.0    33.1588 %
```