COSC 2111 – Data mining (2050)

Assignment 1

Jeyakaran Karnan (s3773303), Sudharsan Seenivasa Raghavan (s3778243)

Date **:** 31st August 2020

Table of contents

1. Classification:

Task 1:

Run number	Classifier	Training error	Cross validation error	Overfitting
1	ZeroR	100%	100%	None
2	OneR	20.56%	25.8245%	None
3	IBK	0.536%	58.7244%	None
4	J48	2.772%	4.16%	None

Table 1: Classifiers with Training and cross validation error

From the above table, IBK classifier performs good with the training set but works bad with cross validation. J48 performs great with two methods. So, we can conclude that J48 works well than any other classifiers used in this task. There is no over fitting occurs in any of the classifier as every classifier has testing accuracy lesser than the training accuracy.

Task 2:

C	M	Training accuracy	Test accuracy
0.25	2	99.81%	99.454%
	3	99.68%	99.454%
	5	99.70%	99.454%
	10	99.3637%	99.064%
1	2	99.894%	99.376%
	3	99.7084%	99.298%
	5	99.7084%	99.454%
	10	99.3637%	99.064%

Table 2: J48 Accuracy

From the above table, it is evident that there is a minor overfitting is there but by increasing the **M** value the overfitting is **gradually decreasing** as it is evident from the table.

Task 3:

SNO	Percentage split	Testing accuracy	Training accuracy
1	66%	99.454%	99.8144%
2	40%	99.469%	99.8144%
3	80%	99.3369%	99.8144%

Table 3: J48 lowering the examples in the dataset

The above table describes the percentage split and its training and testing accuracy. In all the scenario, the training accuracy is greater than testing accuracy. So, we can conclude that there is overfitting. The training accuracy remains stable throughout the run.

Task 4:

K – value	Training accuracy	Test accuracy
1	100%	90.56%

3	95.334%	93.2917%
5	94.3531%	92.7457%
10	93.4783%	92.5117%
20	92.895%	91.7317

Table 4: IBk classifier's accuracy

From the above table, we can conclude the amount of overfitting reduces after the K – value is 10. If we run multiple K- values, we can attain the minimal level of overfitting.

Task 5:

Batch size	Random Tree	Decision stump
1000	99.947%	95.3871%
500	99.947%	95.3871%
250	99.947%	95.3871%

Table 5: Random tree and Decision stump accuracies

Random tree performs better than Decision stump for this dataset throwing nearly 100% accuracy. Even though changing the maximum depth and number of instances, does not make any changes to the accuracy of these classifiers. So, of these two classifiers, we can decide Random Tree is the best.

Task 6:

Classifier	Accuracy
OneR	97.0042%
ZeroR	92.2853%
J48	99.8144%

Table 6: Classifier's accuracy

Of these three classifiers, J48 performs good on this dataset. Since decision tree has easy algorithm and the decision are made by the data itself, this can be more efficient than any other classifiers.

Task 7:

All the analysis has been done for a minimal number of parameters and with very a smaller number of runs. If the runs and the wide range of parameter values has been used, the accuracy would have been attained better.

Task 8:

Classifier	Full set	Reduced set
J48	99.8144%	97.9056%
ZeroR	92.2853%	92.2853%
OneR	97.0042%	97.0042%

Table 7: Reduced set and Full set accuracies

Reduced set does not bring any changes to the accuracy of the ZeroR and OneR classifier. But the J48 classifier's, accuracy got declined after using the reduced set. That may be because, the important attributes needed for the decision tree to make decision might be removed by the attribute selection algorithm.

2. Numeric prediction:

Task 1:

Run number	Classifier	Training error	Cross validation error
1	M5P	12.93	13.6917
2	ZeroR	87.3828	87.6583
3	IBK	0	19.8028

Table 8: Analysis of M5P, ZeroR and IBK

From the above table, ZeroR gives us **maximum** error rate with 87% overall while testing with training data and cross validation. IBK performs great in training data with **0%** error rate.

Task 2:

Batch size	M5P							
	M = 1		M = 2		M = 10		M = 20	
	Time taken	Error rate	Time taken	Error rate	Time taken	Error rate	Time taken	Error rate
1000	0.01	13.7425	0.01	13.7425	0.01	15.9042	0	18.6526
500	0	13.7425	0.01	13.7425	0.01	15.9042	0	18.6526
250	0.01	13.7425	0.01	13.7425	0.01	15.9042	0.01	18.6526
150	0.02	13.7425	0.01	13.7425	0.01	15.9042	0.02	18.6526
100	0.01	13.7425	0.01	13.7425	0	15.9042	0.01	18.6526

Table 9: M5P analysis

Batch size	IBK							
	K = 1		K = 2		K = 10		K = 20	
	Time taken	Error rate	Time taken	Error rate	Time taken	Error rate	Time taken	Error rate
1000	0	19.8028	0	28.2504	0	45.1931	0	40.5361
500	0	19.8028	0	28.2504	0	45.1931	0	40.5361
250	0	19.8028	0	28.2504	0	45.1931	0	40.5361
150	0	19.8028	0	28.2504	0	45.1931	0	40.5361
100	0	19.8028	0	28.2504	0	45.1931	0	40.5361

Table 10: IBK analysis

From the table 9 and table 10, the changes in parameters **Batch size, M** and **K** is analyzed. **M5P algorithm and IBK algorithm** increases the error rate while increasing the **M** and **K** value, respectively. So, maintaining the **M and K** value as low as possible will increase the predictive accuracy. **Minor Overfitting** occurs in all levels of parameters in both algorithms.

Task 3:

Batch size	Linear Regression			
	M5 method		Greedy method	
	Time taken	Error rate	Time taken	Error rate
1000	0	28.4042	0	29.0213
500	0	28.4042	0	29.0213
250	0	28.4042	0	29.0213
150	0	28.4042	0	29.0213
100	0	28.4042	0	29.0213

Table 11: Analysis using Linear Regression

Random Forest				
K	Seed	Time taken	Error rate	Overfitting
1	1	0.01	7.5295	None
	3	0.02	7.3076	None
	5	0.02	7.1681	None
	10	0.02	7.3282	None
	50	0.03	6.9683	None
4	1	0.03	5.6534	None
	3	0.03	5.3288	None
	5	0.03	5.3952	None
	10	0.03	5.2404	None
	50	0.03	5.3951	None

Table 12: Analysis using Random Forest

From the above two tables, it is evident that changing the seed value increases the accuracy in Random Forest while Linear Regression does not affect much due to Batch size. The overfitting does not happen as the testing accuracy is lesser than the training accuracy.

Task 4: Of all these algorithm, M5P algorithm performs good at this dataset with the best value of **m = 5** and Batch size between **100 and 1000** as we tested.

3. Clustering:

Clustering is the task of dividing the data into different set of groups, where the data points in the same group represent the same activity with the other data points in the same group and different activity with the data points in the different group.

Task 1: K – means clustering has been ran for the given data for different **k (cluster)** values – **(1, 2, 3, 4, 5, 10, 20)**. The sum of squared errors, which is calculated as the difference between each observation and its cluster's mean. This value should be very less possible so that there is less variance among the data points. When the cluster is kept as 5, the value becomes **21.5 (a huge dip).** So, we can postulate that the cluster for this dataset is **5**.

Task 2: Lets fix the K-value as 5 and change the random seed. Selecting the centroid (**Random seed**) in clustering becomes crucial as this will impact the algorithm in many ways. Error value seems sensitive to the random seed. For five consecutive iterations starting at random seed 9, the error value decreases until random seed is 11, and henceforth increases until random seed is set to 13.

Task 3: When the EM algorithm is run on the data with its default parameters, the model was built in **18.83** seconds and the output included 6 clusters with a log likelihood of **-11.87**. It can be increased to get better results.

Task 4: The dataset is been normalized to bring out the range between 0 and 1. Since the "sex" attribute is nominal, it cannot be normalized. After normalizing the log likelihood is been increased tremendously showing the value of **0.515**.

Task 5: All the given three parameters are changed invariably to test the results. The variance **(standard deviation)** is set as **2** initially, we got only one cluster with very minimum likelihood which is not advisable. So, the variance is set as low and minLogLikelihoodImprovementCV is also set as low as possible. Then, we get high performance with 5 clusters and likelihood closer to zero. So, changing these parameters will also have the huge impact on the performance.

Task 6: By clicking on the visualize tab in the Weka tool, we can generally conclude that the data has mostly **2 or 3** clusters for every pair of the attributes. But by varying the number of clusters, we can improve the performance of the algorithm.

Task 7: K means uses Euclidean distance while calculating the distance between two data points and the **EM** uses statistical methods. Different algorithms are preferred for different scenarios and in this case, K means can be preferred over the other because it takes lesser time to build the model.

Task 8: Some common nuggets observed from the algorithms run are, most of the clusters shows us that the females are having the higher **TSH** and **TT4** levels.

4. Association Finding

Task 1: The labels in the **supermarket1-small.arff** are represented as **f, t** and **supermarket2-small.arff** are represented as **?, t** which are the labels represented as true and false.

Task 2: The **supermarket1-small.arff** data is been loaded**.** 30 attributes were selected by using the **CFS subset evaluation** and **Greedy stepwise** search method. Those attributes are selected in preprocessing step and ran through apriori algorithm with default parameters. The results give us some useful associations from the supermarket data. Some significant findings were, "It is **99%** confident that people who don't buy pork won't buy trim pork also". So, that we can keep any one product in the supermarket.

Task 3: When using the **confidence** metric, the associations with higher confidence among the data can be shown with the minimum metric provided. When using the **Lift** metric, the associations are getting more wilder with more itemset. Changing the **Leverage** is not encouraged, as it should be maintained low to getting clarity in the association. **Conviction** will also increase the itemset with complex associations.

Task 4: The **supermarket2-small.arff** data is been loaded**.** All the attributes are selected in this case and ran through apriori algorithm. The results give us some useful associations from the supermarket data. Some significant findings were, "It is **86%** confident that people who buy frozen foods and fruit will buy bread and cake also". So, that we can keep both the product in the supermarket.

Task 5: Minimum **confidence metric** provides us the high confidence the dataset has on that association. Changing the **Lift** metric, makes the associations wilder. **Leverage** should be maintained low.

Task 7: When applying the minimum conviction parameter as **1.2**, we can find many useful and complex associations that are useful to the business. **99%** of the people who are not interested in buying pork does not buy desserts. We can conclude that most people who are not having pork, they are also not having desserts.

```
43. puddings-deserts=f department124=f 142 ==> trim pork=f 141    conf:(0.99) lift:(1.01) lev:(0.01) [1] < conv:(1.18)>
```

Task 8: The dataset has association rules that are predominantly in **false**. It is **100%** sure that the people who do not have Thyroxine globulin deficiency do not have hypopituitary condition.

References:

- Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2017). Data mining: Practical machine learning tools and techniques. Amsterdam: Morgan Kaufmann.
- Finding a balance: What are the challenges of ethical data mining. (2019, April 26). Retrieved August 31, 2020, from https://www.information-age.com/data-mining-123481736/
- www.recruiter.com. (n.d.). Ethical Data Mining: How Doing the Right Thing Is Good for Business. Retrieved August 31, 2020, from https://www.recruiter.com/i/ethical-data-mining-how-doing-the-right-thing-is-good-for-business/

Split up of tasks in the group:

Part – 1 (Classification)	– Majorly done by **Jeyakaran Karnan**
Part – 2 (Numeric Prediction)	– Majorly done by **Jeyakaran Karnan**
Part – 3 (Clustering)	– Majorly done by **Sudharsan Seenivasa Raghavan**
Part – 4 (Association finding)	- Majorly done by **Sudharsan Seenivasa Raghavan**
Part - 5 (Presentation)	- Content by **Sudharsan Seenivasa Raghavan** and **Jeyakaran Karnan**, Recording by **Sudharsan Seenivasa Raghavan**