Assignment #2

SYDE – 675, Winter 2019
Tushar Chopra
tushar.chopra@uwaterloo.ca

Datasets Description

The UCI Machine Learning Repository is a collection of databases, domain theories, and data generators that are used by the machine learning community for the empirical analysis of machine learning algorithms.

Datasets are available on http://archive.ics.uci.edu/ml/datasets.html For this homework assignment, you need to download the datasets "glass" and "Tic-Tac-Toe Endgame" from the above link. The "glass" dataset is categorical and the "Tic-Tac-Toe" dataset is continuous.

Question 1

Design a C4.5 decision tree classifier to classify each dataset mentioned above. Report the accuracy based on the 10-times-10-fold cross validation approach (20% of training set as the validation set for every experiment). Report the mean accuracy and the variance of the accuracy for each experiment.

Explanation:

- **C4.5 decision tree** is used for the classification, which is based on splitting of data via information gain and rule based pruning. For pruning the decision is first converted in the set of rules. Then those rules are pruned by validation set. Pruning is done by randomly removing a node from the path and checking the accuracy of that rule. If removing of that node increases the accuracy of that rule, the process continues recursively until the accuracy is decreased.

- **10 k 10 folds cross validation** is used for the whole process. That is dataset is divided into 10 parts, each time 1 part is used for testing and remaining 9 parts are used for training purpose. This is for 1 time, so for 10 times the process is repeated 10 times and each time the data is shuffled. Implementing this process makes sure there is no bias in training and all possible combinations are tested for the dataset accuracy.

- **Validation set** is 20 percent of training data in each iteration. So that is 20 percent of 90 percent of data(training data) which is equal to 18% of the whole data.

- **The 2 datasets** were Glass and Tic-Tac-Toe taken from UCI machine learning database. The glass data is continuous while the tic-Tac-Toe is symbolic. The Decision tree is coded in generic way that is, it can handle both the datasets and can predict the outputs.

```
Glass Accuracy Mean : 0.6400432900432901
Glass Accuracy Variance : 0.010960111317254176
```

Output of Glass Dataset

```
Tic-Tac-Toe Accuracy Mean : 0.9358344298245613
Tic-Tac-Toe Accuracy Variance : 0.0008504907026
```

Output of Tic-Tac-Toe Dataset

- **Output** of the question 1 is shown above. As the quality of glass dataset is poor, decision tree is able to manage its accuracy to 64% and high variance of 0.01. But for the Tic-Tac-Toe dataset the accuracy is 93.5% with low variance 0.0008.

Question 2

There are two possible sources for class label noise:

a) Contradictory examples. The same sample appears more than once and is labeled with a different classification.

b) Misclassified examples. A sample is labeled with the wrong class. This type of error is common in situations where different classes of data have similar symptoms.

To evaluate the impact of class label noise, you should execute your experiments on both datasets, while various levels of noise are added. Then utilize the designed C4.5 learning algorithm from Question 1 to learn from the noisy datasets and evaluate the impact of class label noise (both Contradictory examples & Misclassified examples).

● Note: when creating the noisy datasets, select L% of training data randomly and change them. (Try 10-times-10-fold cross validation to calculate the accuracy/error for each experiment.)

a) **Plot one figure for each dataset that shows the noise free classification accuracy along with the classification accuracy for the following noise levels: 5%, 10%, and 15%. Plot the two types of noise on one figure.**

Explanation:

- **Contradictory Noise** exists when two or more sample may have exhibit identical behaviours but they classified in different class. This issue can exist in the data due to human error or insufficient information about the data. That is, the class of the data depends upon more number of features than the known number or features.

- **Misclassified Noise** is the error in the data while labelling. This issue can exist in your data while collecting it. There may be error in system or the data collection process is wrong or done carelessly.

- **In this Question** the noise is generated by code and its behaviour on the accuracy is observed.

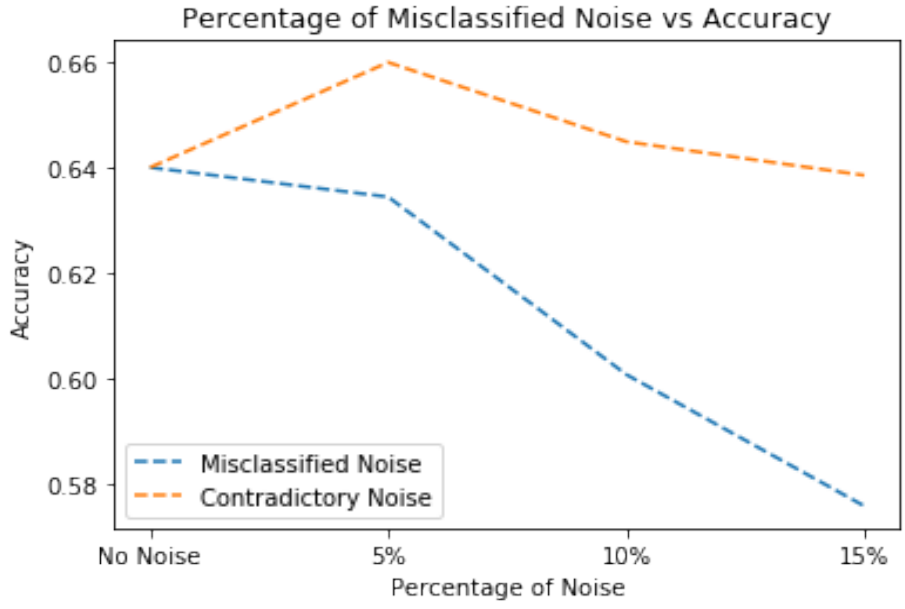
- **Glass Dataset's** Output is shown below.

```
For Glass Dataset, Misclassified Noise...

For No Noise 0% : Mean is 0.6400432900432901  and Variance is 0.010960
111317254176
For No Noise 5% : Mean is 0.6344588744588744  and Variance is 0.009743
574427015986
For No Noise 10% : Mean is 0.6007575757575758  and Variance is 0.00974
3574427015986
For No Noise 15% : Mean is 0.5758008658008658  and Variance is 0.01103
0244935439741
```

```
For Glass Dataset, Contradictoy Noise...

For No Noise 0% : Mean is 0.6400432900432901  and Variance is 0.010960
111317254176
For No Noise 5% : Mean is 0.659978354978355  and Variance is 0.0131809
67373175166
For No Noise 10% : Mean is 0.6449350649350647  and Variance is 0.01318
0967373175166
For No Noise 15% : Mean is 0.6385497835497836  and Variance is 0.01084
4577031914694
```



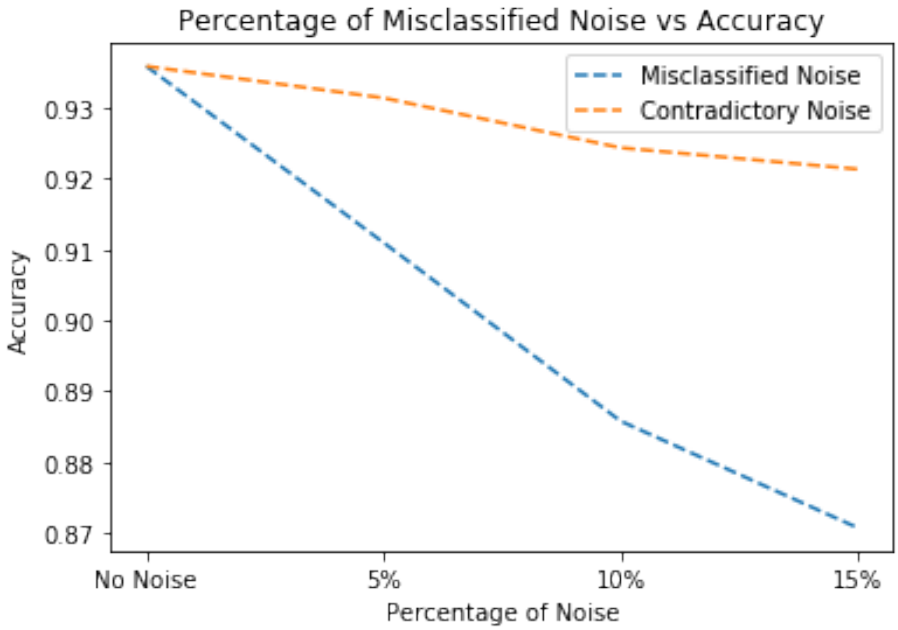
- Tic-Tac-Toe Dataset's Output is shown below.

```
For Tic-Tac-Toe Dataset, Misclassified Noise...

For No Noise 0% : Mean is 0.9358344298245613  and Variance is 0.0008
504907026681285
For No Noise 5% : Mean is 0.9109594298245615  and Variance is 0.0015
275462870787166
For No Noise 10% : Mean is 0.885777412280702  and Variance is 0.0015
275462870787166
For No Noise 15% : Mean is 0.8707236842105263  and Variance is 0.001
0063154047399202
```

```
For Tic-Tac-Toe Dataset, Contradictoy Noise...

For No Noise 0% : Mean is 0.9358344298245613  and Variance is 0.0008
504907026681285
For No Noise 5% : Mean is 0.9313695175438598  and Variance is 0.0009
300528467893965
For No Noise 10% : Mean is 0.9243366228070176  and Variance is 0.000
9300528467893965
For No Noise 15% : Mean is 0.9213201754385968  and Variance is 0.000
7565751769775313
```



b) How do you explain the effect of noise on the C4.5 method?

Explanation:

- The outputs are shown in part (a) of this question.

- Misclassified Noise. The data is modified by adding different amount of noise [5%, 10%, 15%]. As observed and expected the accuracy is decreased when the noise is added into the data. The degradation of the performance of classification is directly depends on the amount of noise in the data.
- For every 5 percent of the noise the accuracy is decreased roughly 1.5 to 2 percent for the Tic-Tac-Toe dataset and 2 to 3 percent for Glass dataset.
- The higher drop in accuracy in Glass dataset is expected as the dataset is not clean in itself than accuracy drop in Tic-Tac-Toe dataset.
- With no noise the Glass dataset has accuracy of 64% which is dropped to 57%(approx) after adding 15% of noise. On the other hand, Tic-Tac-Toe is dropped to 87% at 15% noise from the 93.5%(approx) without noise.

- Contradictory Noise. The data is modified by adding different amount of noise [5%, 10%, 15%]. As observed the accuracy is decreased when the noise is added into the data. But the behaviour is erratic for Glass dataset and that's because of poor data quality in Glass dataset.
- For Tic-Tac-Toe these is a drop in accuracy, but it hampers the performance lesser than misclassified noise.
- In general, degradation of the performance of classification is observed here as well and is directly depends the amount of noise in the data.
- The impact on accuracy after adding Contradictory noise is lesser than Misclassified noise. The possible explanation of this behaviour can be is, your most of the training set is correct only new erroneous samples are added. It means major portion of the samples are intact compare to misclassified noise. In misclassified noise your big chunk of data is corrupted, higher percentage of bad data compare to contradictory noise.

Question 3

Design a feature selection algorithm to find the best features for classifying the Mnist dataset. Implement a bidirectional search algorithm using the provided objective function as the measure for your search algorithm.
Use the first 10000 samples of training set in the Mnist dataset for feature selection and training set for kNN approach. Use Euclidean distance to calculate Inter-class.

The objective function should be based on this equation:

$$J = \text{Inter Class distance}$$

a) Select the set of {10, 50, 150, 392} features based on the implemented feature selection approach and report the accuracy on the test set of MNIST based on kNN with k = 3. Note: you can take advantage of data structure tricks to speed up the efficiency of kNN algorithm.

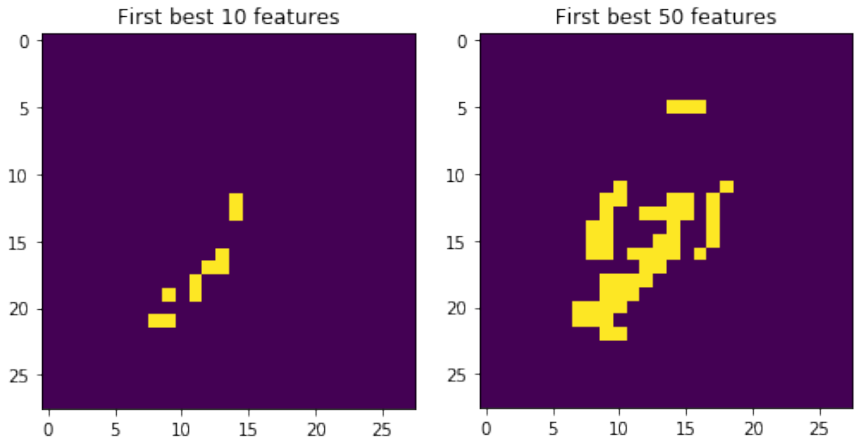
Explanation:

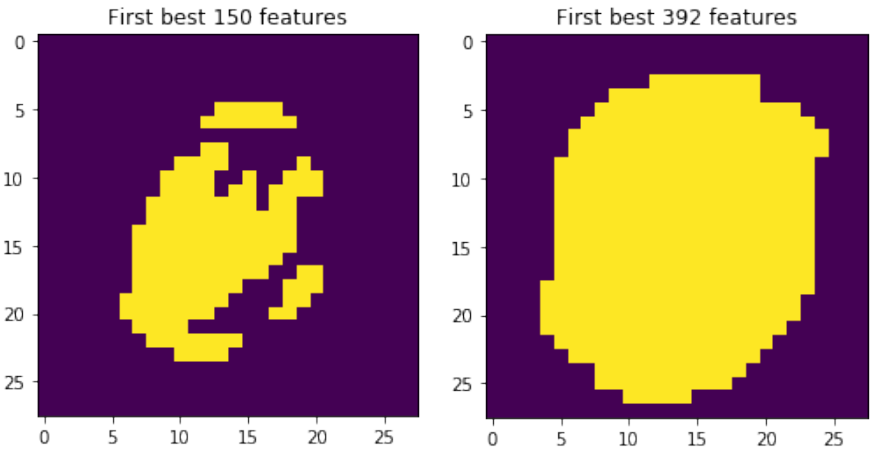
```
For d = 10 Accuracy is 0.5815
For d = 50 Accuracy is 0.8165
For d = 150 Accuracy is 0.9135
For d = 392 Accuracy is 0.9365
```

- The interclass distance is the distance between the mean of the classes. For calculating distance, we used Euclidian distance. In this question, we have selected different number of features based and sort by interclass distance. Means, the feature which has maximum distance (J_max) is selected first then second best then so on.
- Bidirectional search is used to get best features. First a feature with higher value of J_max is selected and then stored in the empty list followed by its deletion from the main dataset. Second, feature which has the minimum distance is removed as well from the dataset. This process is repeated until we get our desired number of features.
- This main advantage of bidirectional search is, it works from both sides if the dataset. That is selected the best feature(Sequential Forward Search) and removing the worst one(Sequential Backword Search).
- The output of different number of features or dimensions is shown above. The accuracy improves when we select more number of features than fewer one.
- Important catch in this is that, when d=150 the accuracy is 91.35% which is 2% less than d=392. It means after selecting best certain number of features from a dataset the accuracy doesn't improves because the feature which have higher interclass distance are selected first. And these features are sufficient enough of differentiate the data and classify new sample.

b) Visualize the selected features for each set in {10, 50, 150, 392} by a zero 2-D plane where the selected features are pixels set to a value of 1. Compare the 4 different planes.

Explanation:





- The output of the best features is shown above for various values of d. All these fours images confirm that the most of data is in the centre of the image, while the pixels around the corner of the image are mostly zero.
- As we move from d=10 to d=392 we can see that centre part of the image is becoming more important or the pixels in these region have more interclass distance than the pixels in the corner of the image.
- That's why SFS choose the pixels in the middle most for the classification and the SBS choose the pixel around the corner and remove those.

c) Apply LDA on the dataset and report the accuracy based on kNN with k =3. Compare the achieved accuracy by the reported accuracies in part (a). Note: you need to implement LDA method by yourself.

Explanation:

```
Accuracy : 0.903
```

- The accuracy of the Linear Discriminant Analysis with KNN is shown above which is 90% for the values of k=3 and d=9(number of classes - 1).
- LDA gives a projection which maximises the separation between multiple classes.
- Compare to Interclass distance we can see the LDA is performing better with lower number of dimensions. It best selects the features which can be used to identity or clarify the data.
- On the other hand, in interclass distance, we need mean of each class and then we calculate Euclidian distance. We took mean which best represents a class but it may not be accurate because of the presence of the outliers.
- For interclass distance we need the labels of the data to calculate the distance and get the best features. On the other hand LDA doesn't care about this, it uses eigenvalues and Eigen vectors to get to know about the best projection from where the data can be classified at the best.
- LDA wins in features selection over interclass distance. It using 9 features out of 784 which is approximately 1% of the data to get 90% accuracy. On the other hand Interclass distance is only able to get 58% accuracy for first 10 features. Even after using 50 features interclass distance feature selection is only able get accuracy of 80%.