Assignment - 2

Prerak Patel - 000825410

# Data Description

Data set name:

Wisconsin Breast Cancer

Source: <http://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29>

Features:

Features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image. Below mentioned are the features in the data set used which has feature’s title and range of that feature.

|  |  |
| --- | --- |
| 1. Radius | 1. Texture |
| 1. Perimeter | 1. Area |
| 1. Smoothness | 1. Compactness |
| 1. Concavity | 1. Concave points |
| 1. Symmetry | 1. Fractal dimension |
| 1. Radius error | 1. Texture error |
| 1. Perimeter error | 1. Area error |
| 1. Smoothness error | 1. Compactness error |
| 1. Concavity error | 1. Concave points error |
| 1. Symmetry error | 1. Fractal dimension error |
| 1. Worst radius | 1. Worst texture |
| 1. Worst perimeter | 1. Worst area |
| 1. Worst smoothness | 1. Worst compactness |
| 1. Worst concavity | 1. Worst concave points |
| 1. Worst symmetry | 1. Worst fractal dimension |

Description of classification task, plus some statistics:

Breast Cancer Data set is classified into two parts.

1. Benign
2. Malignant

There are total 569 data sets.

Out of which 357 are classified as “BENIGN” and rest of the 212 are “MALIGNANT”

# The Tests

## Part 1: k-Nearest Neighbour

Below test results computed by changing two K values 3 & 5 and also by normalizing data with Euclidean Distance, Minkowski Distance and Manhattan Distance.

Training and testing split are shuffled every single time before the run and then separated as 80% training data and 20% testing data.

After running all the algorithms for 500 times then the best performer comes out to be K=5 Euclidean Distance Normalized, K=5 Manhattan Distance Normalized. Whereas averaging at 94.9, worst performer is K=3 Minkowski Distance at 92.2 average.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Average | Run 1 | Run 2 | Run 3 | Run 4 | Run 5 |
| K=3, Euclidean Distance | 93.2 | 95.6 | 93.9 | 94.7 | 90.4 | 91.2 |
| K=5 Euclidean Distance | 94.2 | 95.6 | 94.7 | 93.9 | 91.2 | 95.6 |
| K=3 Minkowski distance | 92.3 | 92.1 | 94.7 | 90.4 | 92.1 | 92.1 |
| K=5 Minkowski distance | 93.2 | 93.0 | 93.9 | 93.0 | 93.0 | 93.0 |
| K=3 Manhattan Distance | 93.2 | 95.6 | 93.0 | 93.9 | 89.5 | 93.9 |
| K=5 Manhattan Distance | 92.8 | 91.2 | 90.4 | 94.7 | 84.7 | 93.0 |
| K=3, Euclidean, Normalized | 92.6 | 95.6 | 94.7 | 91.2 | 94.7 | 86.8 |
| K=5, Euclidean, Normalized | 94.7 | 94.7 | 97.4 | 93.0 | 93.9 | 94.7 |
| K=3, Minkowski, Normalized | 91.6 | 91.2 | 86.0 | 91.2 | 98.2 | 91.2 |
| K=5, Minkowski, Normalized | 94.2 | 94.7 | 93.9 | 96.5 | 94.7 | 91.2 |
| K=3, Manhattan, Normalized | 95.1 | 93.9 | 98.2 | 94.7 | 98.2 | 90.4 |
| K=5, Manhattan, Normalized | 94.7 | 93.0 | 93.0 | 94.7 | 97.4 | 95.6 |

Text, letter

Description automatically generated

## Part 2: Decision Trees

Below test results are computed by changing criterion, min\_samples\_split, max\_features, max\_leaf\_nodes and max\_depth.

After running all the decision tree algorithms for 500 times best result were provided by criterion = entropy & max\_depth=5 together at 93.4 and worst performer was criterion = entropy, max\_features = 2 at 92.3 average.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Average | Run 1 | Run 2 | Run 3 | Run 4 | Run 5 |
| Default with no args | 93.2 | 94.7 | 93.9 | 92.1 | 91.2 | 93.9 |
| criterion = entropy | 93.5 | 96.5 | 94.7 | 93.0 | 91.2 | 92.1 |
| criterion = entropy, min\_samples\_split = 30 | 92.3 | 92.1 | 92.1 | 90.4 | 94.7 | 92.1 |
| criterion = entropy, max\_features = 2 | 93.2 | 92.1 | 94.7 | 90.4 | 93.0 | 95.6 |
| criterion = entropy, max\_leaf\_nodes = 10, min\_samples\_split = 30 | 92.8 | 87.7 | 95.6 | 93.9 | 93.9 | 93.0 |
| criterion = entropy, max\_depth=5 | 92.6 | 88.6 | 95.6 | 94.7 | 93.9 | 90.4 |
| criterion = "entropy", max\_leaf\_nodes = 20,  max\_depth = 4 | 93.2 | 91.2 | 93.9 | 94.7 | 93.0 | 93.0 |
| criterion = entropy,  max\_leaf\_nodes = 15 | 93.9 | 95.6 | 93.0 | 95.6 | 95.6 | 89.5 |

Text

Description automatically generated

# Discussion

1. **Part 1: Are there clear winners or losers for kNN? Give some solid ideas for why some versions might be better or not better than others. Were you surprised by any of the results? Do the results make sense to you? Why or why not? Be as specific as you can and reference the properties of your data.**

* In the kNN algorithm there are no clear winners or losers as the results for all the algorithms lies between 93.0 to 95.0.
* Clearly normalized data performed better than the unnormalized but not by whole lot, as the gap was not that huge to make a noticeable difference.
* I was not surprised by any particular results, but I was surprised by kNN algorithm in general as the success ratio was able to get about 94%.
* Results did made sense to me as the data sets in Breast Cancer are not overlapping on each other a lot so it’s easy to classify using kNN algorithm such as Euclidean Normalized with k = 5.

1. **Part 2: Are there clear winners or losers for Decision Trees? How did the Decision Trees perform vs. kNN? Give some solid ideas for why some versions were better and why Decision trees or kNN were better overall. Were you surprised by any of the results? Do the results make sense to you? Why or why not? Be as specific as you can and reference the properties of your data.**

* There were no clear winners or losers for Decision Trees as the spread for all the different classifiers lies in between 92.0 to 94.0 which is very tight to label winner or loser.
* Performance of Decision Trees resembles to kNN algorithms as both gives almost same accuracy in labelling the test results.
* Best version of Decision Tree was criterion = entropy & max\_depth=5, as truncating the max\_depth helped not overfitting the training data and scoring more on the testing data. Overall, kNN algorithm performed better than the Decision Tree for most of the algorithms.
* I was not surprised by the outcomes of Decision Tree algorithms as the results did made sense to me.
* The tree.dot pdf was helpful to understand the classification of the decision tree algorithm and the classification was following the precise information passed in the constructor of the Decision Tree Classifier.
* 

# Future Work

1. **If** **you had more time, what more could you explore? Think about how you could modify the data or the kNN and Decision Tree algorithms…**

* If I had more time I would have tried the same algorithms kNN and Decision Tree algorithms to different data sets to see how effective is these algorithms and then average results from different data sets to derive definite outcome for both kNN and Decision Tree algorithms.