Project 2: Nearest Neighbor

Hung Le, 862063377, hle026@ucr.edu, 12/8/2022

https://github.com/hungle132/nearest-neighbor

Introduction:

In this project, I implement the nearest neighbor algorithm that will take in a textfile of classes and their features. It will then parse the data into a 2D array so we can loop through the array and find each features closest neighbor according to the class. Then I put the vectors into the of the values into the function to run the nearest neighbor algorithm and run forward selection. The function as of now is able to get the best feature for the first one but isn't able to gather the next best feature. I believe it's because of the way I designed the nearest neighbor calculation. The backwards selection would have used the feature test for the part of the code I didn't implement.

Forward Selection:

The forward selection would take in the 2D vector and loop through the feature size and the number of features. I then had each nearest neighbor return the lowest euclidean distance of the current sqrt(val - temp)^2. After I get this value, I ran through the feature and got the class desc of the lowest euclidean value and pushed it to a vector. Then I would do this until we are done looping through the feature. I then push the correct values of the feature onto a vector and compare it with another vector of correct values. Then I get the result of how many class spots were correct so I get the accuracy of the feature for the nearest neighbor. This will give me the best feature of the

class and then we returned as a vector of int where the best feature is the position at the I spot. The highest accuracy is also returned and compared with a variable that holds the highest accuracy and will test if there is a new higher accuracy of the subset.

Backward Selection:

In this algorithm, we would look at each subset of the features and see which one would give us a lower accuracy and remove that certain feature. We will then loop through the 2D array until there exists 2-3 features left that give us a high accuracy of the nearest neighbor. The backward selection should take around the same time for a smaller dataset but should run faster for larger sets because of being able to find the solution a lot faster.

Conclusion:

The two searches are supposed to be nearly similar when running them with small datasets and backward being faster with larger datasets. The code could have been done better to incorporate a subset feature search so it can look through the entire featureset. I probably could have done another function that takes in the feature set and run the euclicean distance with that set in order to get the new accuracys of the features. In any case, the code as of now is able to get the first best feature from the feature list.

Small dataset 1 and Small dataset 96

```
Selection Algorithm for nearest neighbor
1
forward selection
rate = 71.2%
==================================================
rate = 71.6%
==================================================
rate = 74.2%
==================================================
rate = 74.4%
==================================================
rate = 81.8%
==================================================
rate = 74%
==================================================
5
--------------------------------------------
rate = 71.2%
==================================================
rate = 71.6%
==================================================
rate = 74.2%
==================================================
rate = 74.4%
==================================================
rate = 74%
==================================================
5 4
accuracy lowered
--------------------------------------------
rate = 71.2%
==================================================
rate = 71.6%
==================================================
rate = 74.2%
==================================================
rate = 74%
==================================================
5 4 3
accuracy lowered
--------------------------------------------
rate = 71.2%
==================================================
rate = 71.6%
==================================================
rate = 74%
==================================================
5 4 3 6
accuracy lowered
--------------------------------------------
rate = 71.2%
```

```
Selection Algorithm for nearest neighbor
1
forward selection
rate = 87.4%
==================================================
rate = 68.2%
==================================================
rate = 73.4%
==================================================
rate = 71.8%
==================================================
rate = 67.2%
==================================================
rate = 74.6%
==================================================
1
--------------------------------------------
rate = 68.2%
==================================================
rate = 73.4%
==================================================
rate = 71.8%
==================================================
rate = 67.2%
==================================================
rate = 74.6%
==================================================
1 6
accuracy lowered
--------------------------------------------
rate = 68.2%
==================================================
rate = 73.4%
==================================================
rate = 71.8%
==================================================
rate = 67.2%
==================================================
1 6 3
accuracy lowered
--------------------------------------------
rate = 68.2%
==================================================
rate = 71.8%
==================================================
rate = 67.2%
==================================================
1 6 3 4
accuracy lowered
--------------------------------------------
```