IA pràctica 3

GEI

Grau en Enginyeria Informàtica

3er Curs GEI
Grup 2

15/01/2023

Sebastian Jitaru
Gerard Llubes

Treepredict

1.1 Implementation

1.1.1 Recursive construction of the decision tree

It starts by checking if the dataset is empty, if so it returns an empty decision node.It then calculates the current score of the dataset using the score function.It then iterates over all the columns in the dataset and for each column it generates a list of possible values.For each value it divides the dataset and calculates the information gain by comparing the current score and the score of the divided sets. It updates the best gain, best criteria and best set if it finds a better split. It then creates the sub branches by recursively calling the buildtree function on the best sets. If it doesn't find a better split, it returns a DecisionNode with the results of the unique counts of the dataset.

1.1.2 Iterative construction of the decision tree

The iterative version of the previous method

1.1.3 Classify method

Ittakes in an observation and decision tree as input.It checks if the tree has results, if so it returns the results.If not it checks the value of the observation at the column index in the tree and chooses the branch to follow and recursively call the classify function on the chosen branch until it reaches the leaf node.

1.1.3 Tree pruning

For each pair of leaves with the same parent, the method tests if joining them decreases the impurity below the threshold.If the impurity is below the threshold then the method joins the leaves and converts the parent to a leaf.The method repeats this process for each pair of leaves until it cannot prune more of them

1.2 Decision trees

1.2.1 Test performance

get_accuracy:
It iterates over the dataset and for each row, it compares the true class label with the predicted class label. Then it returns the accuracy as the ratio of number of correctly classified instances to the total number of instances in the dataset.

1.2.2 Cross-validation

This function performs k-fold cross-validation on a dataset by initializing a random number generator, dividing the dataset into k equal subsets, building decision trees using train sets, pruning the tree using the threshold, calculating accuracy and returning aggregate performance by applying an aggregation function on accuracy list

1.2.3 best threshold

We modify the main function to loop over a range of threshold values and perform cross-validation for each value. Then choose the threshold value that gives the highest accuracy.

```
sebas@sebas-ws:~/Desktop/Global.nou/Studies/Uni/3r/1rQ/IA/Practiques/IA_Prac3_Llubes_Jitaru$ python3 evaluation.py
Threshold: 0.1, Accuracy: 0.95
Threshold: 0.3, Accuracy: 0.95
Threshold: 0.5, Accuracy: 0.9916666666666668
Threshold: 0.7, Accuracy: 1.0
Best threshold: 0.7 with accuracy: 1.0
{'setosa': 50, 'versicolor': 50, 'virginica': 50}
```

Clusters

1.3.1. T9 - Total distance. 0.5 points

In this modification, after the main loop of the function, a variable sum_distances is initialized to 0 and for each cluster, for each point in the cluster, the distance between the point and the centroid of the cluster is calculated using the distance function provided, and added to the sum_distances. Finally, the method returns a tuple with the clusters and the sum_distances.

1.3.2. T11 - Restarting policies. 2 points

Adding this piece of code at the end of the kcluster method will take into account the number of executions and it will keep as the result(best_config) the best configuration

```
    for i in range(k):
        for j in bestmatches[i]:
            sum_distances += distance(clusters[i],rows[j])
    if sum_distances < best_sum_distances:
        best_config = (clusters, bestmatches)
        best_sum_distances = sum_distances
return best_config
```

1.4.1. T10 - Distance in function of `k`. 0.5 points

Show the total distance in function of different values of `k`

```
sebas@sebas-ws:~/Desktop/Global.hdd/Studies/Uni/3r/1rQ/IA/Practiques/Material PR3$ python3 clusters.py
k: 2, Total Distance: 65.54255982749848
k: 3, Total Distance: 59.93955655179541
k: 4, Total Distance: 58.593628923913556
k: 5, Total Distance: 54.97491317667806
k: 6, Total Distance: 54.97092428274785
k: 7, Total Distance: 53.01733925809409
k: 8, Total Distance: 49.937432815248314
k: 9, Total Distance: 52.027797626227255
k: 10, Total Distance: 50.81166224600485
```