PROJECT 1

SUDHEER NADELLA

U93511802

Use J48 (decision tree learning algorithms) and load the Iris data (in the files). Choose J48 as the classifier. Use default options. For test options use the training set. What accuracy do you get?

By selecting “use training set” for test options, I got 98% accuracy. There are only 3 Incorrectly classified Instances.

Does analysis allow for any conclusions about this classifier applied to this data set?

Since we only got 3 incorrectly classified instances with high accuracy, the model seems to be overfitting and there may be some bias.

Now try a 10-fold cross validation. What accuracy do you get?

For a 10-fold cross validation, the accuracy is 96% and we got 6 incorrectly classified instances.

Now change the minnumobj to 1. That controls the minimum number of examples at a leaf. What do you get? Why?

The accuracy after changing the minnumobj to 1 is 94.6667% and got 8 incorrectly classified instances. By changing the minnumobj to 1, we are making the number of leafs to 1 so that it makes the selection rule more specific and that eventually results in more number of rules and the tree overfits.

Now change minnumbobj back to 2. Click on more options and change the seed to 23. Do a 10-fold cross validation. What accuracy do you get? . Did the seed matter in this accuracy?

After setting the minnumobj back to 2, and changing the seed value to 23, the accuracy I got for a 10-fold cross validation is 95.333%. Yes, the accuracy is slightly affected by the seed value. Since changing the randomseed value changes the random sampling, If random sampling value is more, then the sampling rule is more general.

Change the seed back to 1.

Now go to select attributes and apply Principal Components. Do all 4 original features have coefficients that are distinct from 0? Looking at the first principal component what seem to be the most important features and why

Eigenvectors

V1	V2	
-0.5224	0.3723	sepallength
0.2634	0.9256	sepalwidth
-0.5813	0.0211	petallength
-0.5656	0.0654	petalwidth

Ranked attributes:

0.2723 1 -0.581(petallength)-0.566(petalwidth)-0.522(sepallength)+0.263(sepalwidth)

0.042 2 0.926(sepalwidth)+0.372(sepallength)+0.065(petalwidth)+0.021(petallength)

Yes, all the coefficients have feature that are distinct from 0. The most important feature in the first principal component is “petal length” (0.581)

Now go to preprocess and delete the features having to do with petal. What is your accuracy?

The accuracy after removing petallength and petalwidth is 72.6667% with 10-fold cross validation

80.6667% with Use training set

Reload and delete the features having to do with sepal. What is your accuracy?

The accuracy after removing sepallength and sepalwidth is 96% with 10-fold cross validation

98% with Use training set.

Did the weights for Principal components agree with the above results (explain)?

Yeah, since the principal components are the petal features and accuracy after removing sepal features is 96%, It agrees with the above results.

Now let’s cluster the Iris data. Load it (remember to remove the classes), choose EM clustering. Cluster and visualize in petal-length, petal-width space. How many clusters were found? Can you conclude anything from the number?

I can observe 5 clusters, in my observation some of the samples belonging to same class are different from the rest and are being clustered separately from the rest of the samples of that class.

Next use EM and set it to 3 clusters. Visualize it. Visualize the raw data all in petal-length, petal-width space. Can you see any errors and if so, describe them (e.g. between what classes).

After setting it to 3 clusters, I found nearly 14 errors. Many points of iris-virginica and iris-versicolor are misclassified

Clustered Instances		0 1 2 ← assigned to cluster	
0	64 (43%)	0 50 0 | Iris-setosa	Cluster 0 ← Iris-versicolor
1	50 (33%)	50 0 0 | Iris-versicolor	Cluster 1 ← Iris-setosa
2	36 (24%)	14 0 36 | Iris-virginica	Cluster 2 ← Iris-virginica

The actual Iris-versicolor instances are 50 but 14 of iris-virginica instances are also considered as versicolor.

Use simple K-means for clustering and set it to 3 clusters (remember remove classes) with the rest of the parameters as default. How many errors in your result?

With K means clustering and the number of clusters as 3, I found 17 errors.

Clustered Instances		0 1 2 ← assigned to cluster	
0	61 (43%)	0 50 0 | Iris-setosa	Cluster 0 ← Iris-versicolor
1	50 (33%)	47 0 3 | Iris-versicolor	Cluster 1 ← Iris-setosa
2	39 (24%)	14 0 36 | Iris-virginica	Cluster 2 ← Iris-virginica

Actual iris-versicolor instances are 47 but 14 of virginica are also considered as versicolor : 14

Actual iris-virginica instances are 36 but 3 of iris-versicolor are also considered as virginica : 3

How does this compare with EM and 3 clusters? You need to visualize the original Iris data with classes.

K means with 3 clusters

Clustered Instances		0 1 2 ← assigned to cluster
0	61 (43%)	0 50 0 | Iris-setosa
1	50 (33%)	47 0 3 | Iris-versicolor
2	39 (24%)	14 0 36 | Iris-virginica

EM with 3 clusters

Clustered Instances		0 1 2 ← assigned to cluster
0	64 (43%)	0 50 0 | Iris-setosa
1	50 (33%)	50 0 0 | Iris-versicolor
2	36 (24%)	14 0 36 | Iris-virginica

Cluster	K means with 3 clusters	EM with 3 cluster
0 : Iris-versicolor	61	64
1 : Iris-setosa	50	50
2 : Iris-virginica	39	36

When K-means compared with EM both with 3 clusters, number of instances of iris-versicolor and iris-virginica have changed but iris-setosa remained constant.