My Name Biao Feng(myNetID: biaof2)

IE598 MLF F18

Module 6 Homework (Cross validation)

Using the Iris dataset, with 90% for training and 10% for test and the decision tree model that you submitted for Module 2:

Part 1: Random test train splits

Run in-sample and out-of-sample accuracy for 10 different samples by changing random_state from 1 to 10 in sequence.

Display the individual scores, calculate the mean and standard deviation of the set. Report in a table format.

	score_1	score_2	score_3	score_4	score_5	score_6	score_7	\
train	0.962963	0.955556	0.962963	0.955556	0.955556	0.962963	0.955556	
test	0.933333	0.933333	0.933333	1.000000	1.000000	0.933333	1.000000	

	score_8	score_9	score_10	mean	std
train	0.955556	0.955556	0.955556	0.957778	0.003395
test	1.000000	1.000000	1.000000	0.973333	0.032660

Part 2: Cross validation

Now rerun your model using cross_val_scores with k-fold CV (k=10).

Report the individual fold accuracy scores, the mean CV score and the standard deviation of the folds. Now run the out-of-sample accuracy score. Report in a table format.

	score_1	score_2	score_3	score_4	score_5	score_6	score_7	score_8	\
0	1.0	0.933333	1.0	1.0	0.933333	1.0	0.833333	1.0	

	score_9	score_10	cv_mean	cv_test	out_of_sample
0	0.916667	0.833333	0.945	0.06414	0.866667

Part 3: Conclusions

Write a short paragraph summarizing your findings. Which method of measuring accuracy provides the best estimate of how a model will do against unseen data? Which one is more efficient to run?

From the graph, it is apparently that the train_test_split is more useful for the unseen data since mean test scores are higher than out of sample score in cross validation.

Cross validation is more efficient to run since we need to write a for loop for train_test_split to set the random_state.

Part 4: Appendix

Link to github repo

https://github.com/johnfeng123/Biao-Feng