Machine Learning – HW2

108062135 呂佳恩

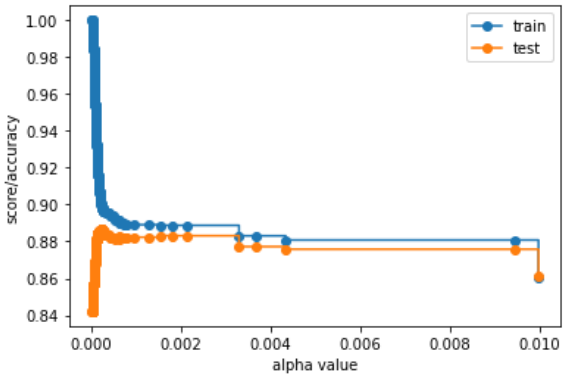
1. The top3 are CMO, mvar12, mvar23 are the top 3.
2. How I built the tree

   For the first part, I followed the instructions and used a recursion to recursively build the left and right part of every node. When it is fully classified/split, it returns the result. Or it returns when the max_depth is reached.

   For the second part, in the sklearn.tree package, I imported the DecisionTreeClassifier, it will automatically build the tree with the provided datas. I also imported the plot_tree package to visualize the tree.

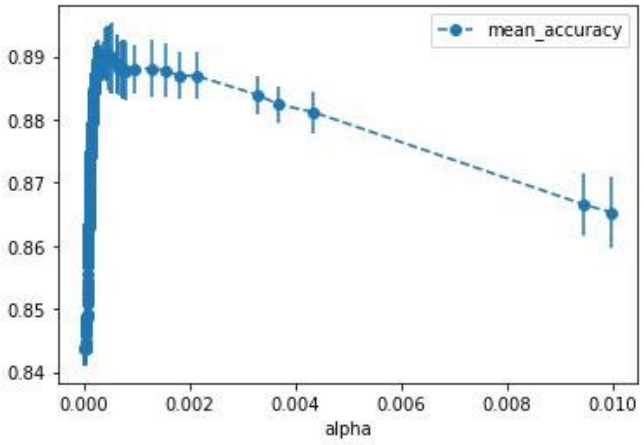
   With the confusion matrix, I could visualize how my model was performing and could tune my model.
3. Improving the tree

   I used the CPP (cost complexity pruning) method of pruning the tree, as we increase the alpha, the more of the tree will be pruned. As we can see from the graph below, as the alpha increases, the accuracy first increases then decreases. This could be explained by saying that when pruning, the model reduces the problem of overfitting. However, after pruning too many leaves, the model becomes too inaccurate since too many features are removed.



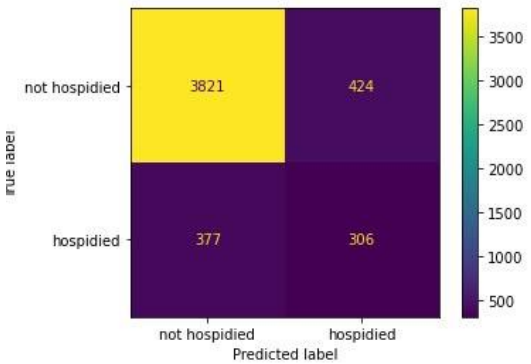
We know that the alpha varies when the training data is changed after doing some test, therefore, we us cross validation to find the alpha with the highest average accuracy.

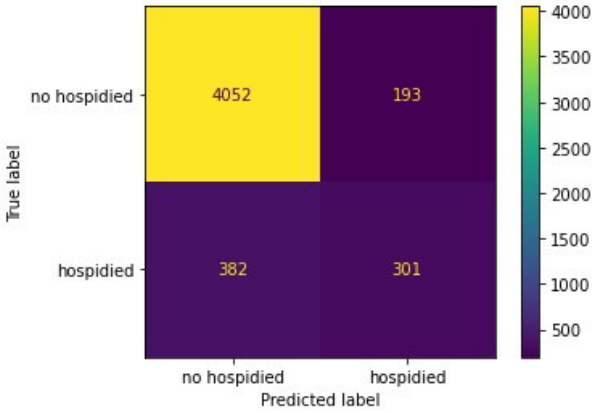
:[111]: <AxesSubplot:xlabel='alpha'>



	alpha	mean_accuracy	std
803	0.000253	0.887993	0.004700
804	0.000257	0.887232	0.003983
805	0.000266	0.887739	0.003614
806	0.000282	0.889109	0.002505
807	0.000282	0.889109	0.002505
808	0.000286	0.889109	0.002721
809	0.000287	0.889109	0.002721
810	0.000290	0.889413	0.002436
811	0.000291	0.889413	0.002436
812	0.000295	0.889413	0.002436
813	0.000295	0.889413	0.002436
814	0.000301	0.889413	0.002436
815	0.000332	0.888652	0.002529
816	0.000334	0.888652	0.002529

As we prune the tree, the model becomes more and more accurate to a degree. The left is the model before pruning. And the right





is after certain degree of pruning.

Preprocess data

I dropped some features that are useless. The patient index provides close to no help when training, it might even make the model worse since the model might try to fit it into the classification. I also dropped the index time since it is also hard to fit in the model.