Week 4 Assignment

Note: Late homework assignments are not accepted because the solutions are discussed at the start of every class on Tuesday. Your solution to this assignment must be uploaded on eCampus as a PDF file. A common approach is to put the solution into a word document and then save that into a PDF file. Please only submit PDF files.

Assignment: You can complete one of the two parts to this assignment, or both. If you do both parts, you can obtain extra points for this assignment. You are expected to complete at least one part successfully.

Data File: CreditHistory_Clean.xlsx

Part 1: Create a SAS EM project named "Week 4 Homework". In that project read this data file for this assignment. Import the data, ensuring that all attributes have the proper metadata described in the data dictionary. In this case, the target is 'good_bad', a binary target.

These data do no contain missing values or outliers. There is no need to "clean" these data. Use 'gini' as the split criterion rather than the default 'ProbChisq'. Gini is the default used in python.

Use 10-fold cross validation to determine the best tree depth for depths 5, 6, 8, 10, and 12. For each setting of the depth parameter, calculate recall, accuracy, precision and F1. Use these metrics to select the best depth for your decision tree, based upon the non-HP tree.

After you have selected the best depth, evaluate your model using a 70/30 training/validation split. Calculate the same metrics for the validation data.

Part 1: SAS EM Solution Upload (all screen shots must be readable)

1. A screen shot of your project diagram
2. A screen shot or listing of ALL SAS code used in your diagram.
3. A table of the metrics for each of the 10 cross-validation folds
4. Describe which model you selected and why.

5. A table of the same metric for the 70/30 test of your selected model.
6. A screen shot of your tree

Part 2: Do the same assignment as Part 1 using Python.

Use one-hot encoding for the nominal attributes, but do not bother to scale the interval attributes. Instead of running depths described for Part 1, use depths 5, 6, 7, 8, 10, 12, 15, 20 and 25. It's easier in python to vary these depths, and with a small dataset it runs quickly.

Also use the parameter drop=False with the ReplaceImputeEncode() method. This was just added for trees, and specifies not to drop the last column for a nominal feature.

Prepare a report containing:

1. A listing of your python code.
2. A table of the metrics (recall, accuracy, precision and F1) calculated for each of your 10 cross-validation folds.
3. Describe which model you selected from Cross-Validation, and why.
4. A table of the metrics (recall, accuracy, precision and F1) for the 70/30 split using your selected model.
5. A screen shot of your tree