Week 7 Assignment

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 Files:
Python - CreditData_RareEvent.xlsx
SAS EM - CreditData_RareEvent.sas7bdat

Part 1 SAS: Create a SAS EM project names "Week 7 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 not contain missing values or outliers. There is no need to "clean" these data.

These data contain 10,500 observations, but only 500 are classified as 'bad' credit. This constitutes a 'rare event' problem since the percentage of 'bad' is less than 10%.

Please build a high-performance decision tree model that best classifies credit applicants as good' or 'bad' credit using the following loss function:

False Negatives (classify someone as 'bad' when they are 'good')
Loss = 0.15 x (Loan Amount)

False Positives (classify someone as 'good' when they are 'bad')
Loss = Loan Amount

For the decision tree, use an hp-tree built using gini impurity. Set the optimum depth to 8 levels.

For Step 1 – selection of the best ratio – use simple majority:minority ratios of 50:50, 60:40, 70:30, 75:25, 80:20, 85:15 and the entire data. For each ratio evaluate 10 randomly constructed datasets. Usually you

would run more than 10, but in the interest of time, we'll use only 10 for this homework exercise.

For Step 2 – ensemble 10 randomly constructed datasets using the best ratio found in Step 1. Each tree will be constructed from a random sample of the optimum ratio and will use gini impurity for splitting and a maximum depth of 8.

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

1. A screen shot of your project diagrams, step 1 and step 2
2. A screen shot or listing of SAS Code used inside one loop and all nodes outside the loop. Should be 3 screenshots.
3. A table showing average loss and MISC for each ratio
4. A description of the total loss and MISC for the ensemble model calculated from the entire dataset, not the smaller ratio dataset.

Part 2 Python: Do the same assignment as Part 1 using Python **with** some modifications. See our notes, chapter 7, for an example.

Use one-hot encoding for the nominal attributes, and scale the interval attributes using 'std' (standardized scaling). Use a decision tree as the base model build upon the 'gini' split criterion and optimize the depth for values between 2 and 20.

For Part 2, the ensemble model, use 100 trees instead of 10.

Prepare a report containing:

1. A listing of your python code.
2. The average loss and misclassification error rate for each ratio evaluated and the optimum depth for that ratio.
3. The total loss from an ensemble model (100 trees instead of 10) using an ensemble model built upon the best ratio and optimum depth.