BAN 5753, Exercise SVM (10 Points)

You must do it alone (it is not a group activity)

The administration of a large private university requested that the Office of Enrollment Management and the Office of Institutional Research work together *to identify prospective students who would most likely enroll as new freshmen in the next Fall semester*. There is one target variable **TARGET(Enroll)** in the dataset **ENROLLMENT\_DATA** and the rest can be thought as input variables for this exercise. The variable descriptions are below. Ignore IRSCHOOL variable for any analysis



For all model comparisons, use Validation Misclassification as the criteria to select best model.

1. Use a Data Partition node (70 for training, 30 for validation and random seed of 12345). Attach an Impute node (use default imputation is SAS but create unique indicator for capturing missing value replacement and change its role to Input). Now, build a logistic regression model to Predict **Target(Enroll)**. Use Stepwise (not Forward) selection and use defaults for selection. Report results from this model and comment on how well does this model work? (**1 point**)
2. Add a Decision Tree node after the data partition node and build the best decision tree model you can. Explain this tree model. Compare your model to the one in #1. Which is better? (**1 point**)
3. Now run another Logistic regression (add after Impute node) with backward elimination (with probability of entry as 1 and probability of stay as 0.05). Connect several SVM nodes with this backward elimination node and use the different functions as shown in class. Comment on which SVM seems to perform the best. Explain the best performing SVM model and compare its performance with the model in #1 and #2. (**2 points**)
4. Now add two versions of Random Forest models as demonstrated in the lecture using same settings as in the lecture after partition node. Which of the two Random Forest model did better?. Did this model perform better or worse compared to model in #1 - #3? Explain. (**2 points**)
5. Use Gradient Boosting as a model – add after partition node. Play with the 3 options mentioned din the lecture to come up with the best model. How well did this model perform compared to models in #1-#4. Comment. (**1 point**)
6. Repeat #1-#5 using your preferred open source codes. You may use *Python* in answering this question. *Please note that you may need to impute before splitting data in Python to make it easier on you (but theoretically speaking, it is not the correct approach!). Also, if you are using sklearn package, use L1/L2 regularization instead of stepwise option*. Gradient Boosting and Random Forest in Python does not have same set of options as SAS EM. So, you need to figure out what the common options and use these options to build the models Python. Make sure you mention what options you used in your write up. Also, the compare and contrast you results from open-source with those from SAS? What are similar and what are different between these results? How do these two differ in terms of ease of use in analyzing and comparing multiple models? Comment.
   1. Turn-in written answers in a Word document for Q1-Q6. (**1 point**)
   2. But also include a self-explainable Jupyter notebook with comments (if using Python). We will run your codes with the data and if the codes don’t run, you will not get credits. You will also not get full credits, if codes do not have comments. (**2 points**)

Deliverables:

As you complete the exercise, create a short report in Microsoft Word and in this report answer the questions in the exercise description. Copy and paste supporting documents/diagrams/screenshots as needed to justify your answer. Make sure you print your name, section number, student ID# on the report and turn-in the report as communicated by your instructor.

*For R/Python part of this assignment, turn-in your codes via separate txt file or Jupyter note book. The codes must be executable and have comments to get full-credits.*