# DAV 6150 Module 11 Assignment

## *<u>Decision Trees & Random Forests</u>*

**<u>*** You may work in small groups of no more than four (4) people for this Assignment ***</u>**

We've learned that decision trees and random forest models can both be very effective when applied to classification problems, and that random forests can often improve upon the performance of a single individual tree by constructing a large number of individual decision trees via bootstrap aggregation and using their collective output to arrive at a predicted data value. While the performance of a random forest model is usually expected to be better than that of a single decision tree, there is an obvious complexity vs. performance tradeoff we must assess when deciding whether to implement a single decision tree vs. a random forest: random forests are much more computationally complex and generally more difficult to explain/interpret than are individual decision trees.

Your task for the **Module 11 Assignment** is to compare the performance of a decision tree vs. a random forest. The data set you will be working with is subset of the dataset we first engaged with for Project 1. A data dictionary describing the attributes contained within the data file is provided below.

| Data Set Attribute | Description |
|---|---|
| report_school_year | Indicates school year for which high school graduation info is being reported |
| aggregation_index | Numeric code identifying manner in which high school graduation data has been aggregated |
| aggregation_type | Text description of how high school graduation data has been aggregated |
| nrc_code | Numeric code identifying "needs / resource capacity", which is an indicator of the type of school district |
| nrc_desc | Text description of the type of school district |
| county_code | Numeric code for county name |
| county_name | Full name of applicable NY State county |
| nyc_ind | Indicates whether or not the school district resides within the borders of NYC |
| membership_desc | Indicates school year in which students first enrolled in High School |
| subgroup_code | Numeric code identifying student subgrouping |
| subgroup_name | Text description of student subgrouping. Note that a student may belong to **<u>MORE THAN ONE</u>** subgrouping (e.g., "Female", "Hispanic", "Not English Language Learner", etc.) |
| enroll_cnt | How many students of the indicated subgrouping were enrolled during the given school year |
| grad_cnt | How many enrolled students of the indicated subgrouping graduated at the end of the given school year |
| grad_pct | What percentage of enrolled students of the indicated subgrouping graduated at the end for the given school year |
| reg_cnt | How many enrolled students of the indicated subgrouping were awarded a "Regents" diploma |
| reg_pct | What percentage of enrolled students of the indicated subgrouping were awarded a "Regents" diploma |
| dropout_cnt | How many enrolled students of the indicated subgrouping discontinued their high school enrollment during the school year |
| dropout_pct | What percentage of enrolled students of the indicated subgrouping discontinued their high school enrollment during the school year |

As you will recall, the dataset is comprised of more than 73,000 observations, each of which pertains to a particular NY State school district and associated subgroupings/categorizations of high school students who had been enrolled for at least 4 years as of the end of the 2018-2019 school year. The response variable you will be modeling will be a categorical indicator variable derived from the dataset's **reg_pct** attribute. This new indicator variable (which you will need to create) will be comprised of three possible values:

A. "**low**": indicates that the percentage of regents diplomas awarded for a given school district / student subgrouping is less than ½ of the median percentage of all regent diplomas awarded (i.e., across all school district / student subgroupings);

B. "**medium**": indicates that the percentage of regents diplomas awarded for a given school district / student subgrouping is between 0.5 * the median percentage of all regent diplomas awarded (i.e., across all school district / student subgroupings) and 1.5 * the median percentage of all regent diplomas awarded (i.e., across all school district / student subgroupings), i.e., (0.5 * median percentage) < percentage of regents diplomas awarded for a given school district <= (1.5 * median percentage)

C. "**high**": indicates that the percentage of regents diplomas awarded for a given school district / student subgrouping exceeds 1.5 * the median percentage of all regent diplomas awarded (i.e., across all school district / student subgroupings).

As such, your decision tree and random forest models should be designed for purposes of predicting which of the three required new indicator values is most likely to apply to a given observation.

Get started on the Assignment as follows:

1) Load the provided **M11_Data.csv** file to your DAV 6150 Github Repository.
2) Then, using a Jupyter Notebook, read the data set from your Github repository and load it into a Pandas dataframe. Ensure your data attributes are properly labeled within the data frame.
3) Using your Python skills, perform some basic exploratory data analysis (EDA) to ensure you understand the nature of each of the variables (including the response variable). (NOTE: If you already have a high-quality EDA from Project 1, you may incorporate the relevant components of it here. If your Project 1 EDA was flawed for any of the attributes that reappear in the M11 data set, you should repeat the EDA work and address any relevant shortfalls identified in your Project 1. Note that any uncorrected flaws will result in corresponding point deductions for this assignment).

   Your EDA writeup should include any insights you are able to derive from your statistical analysis of the attributes and the accompanying exploratory graphics you create (e.g., bar plots, box plots, histograms, line plots, etc.). You should also try to identify some preliminary predictive inferences, e.g., do any of the explanatory variables appear to be relatively more "predictive" of the response variable? There are a variety of ways you can potentially identify such relationships between the explanatory variables and the response variable. It is up to you as the data science practitioner to decide how you go about your EDA, including selecting appropriate statistical metrics to be calculated + which types of exploratory graphics to make use of. Your goal should be to provide an EDA that is thorough and succinct without it being so detailed that a reader will lose interest in it.

4) As the first step of your Data Preparation work, you ***<u>MUST</u>*** create a new categorical indicator variable derived from the content of the **reg_pct** attribute using the approach described above. Using the results of your EDA, create a new indicator variable named "**reg_pct_level**" having the three possible categorizations described above (i.e., "**low", "medium**", and "**high**")

   Ensure that an appropriate "**reg_pct_level**" value is calculated for every observation contained within the data set.

   Once you have created the **reg_pct_level** indicator, **be sure to remove the "reg_pct" and "reg_cnt" attributes from your dataframe**. This must be done to eliminate the collinearity that will result from the addition of the "**reg_pct_level**" indicator to your collection of attributes.

5) Within your Prepped Data Review, be sure to analyze the distribution of the newly created "**reg_pct_level**" indicator value. What does your analysis tells us about the distribution of this newly created indicator variable?

6) Using your Python skills, apply your knowledge of feature selection and dimensionality reduction to the prospective explanatory variables to identify variables that you believe will prove to be relatively useful within your models. Your work here should reflect some of the knowledge you have gained via your EDA work. While selecting your features, be sure to consider the tradeoff between model performance and model simplification, e.g., if you are reducing the complexity of your model, are you sacrificing too much in the way of accuracy (or some other performance measure)? The ways in which you implement your feature selection and/or dimensionality reduction decisions are up to you as a data science practitioner to determine: will you use filtering methods? PCA? Stepwise search? etc. It is up to you to decide upon your own preferred approach. Be sure to include an explanatory narrative that justifies your decision making process.

7) After splitting the data into training and testing subsets, use the training subset to construct **<u>at least two different decision tree models</u>** and **<u>at least two different random forest models</u>** using different combinations of the explanatory variables (or the same variables if they have been transformed via different transformation methods). **Your models must each include at least four (4) explanatory variables.**

8) After training your various models, decide how you will select the "best" classification model from those you have constructed. For example, are you willing to select a model with slightly lower performance if it is easier to interpret or less complicated to implement? What metrics will you use to compare/contrast your models? Evaluate the performance of your models via cross validation using the training data set. Then apply your preferred model to the testing subset and assess how well it performs on that previously unseen data.

**<u>Your deliverable for this Assignment</u>** is your Jupyter Notebook. It should contain a combination of Python code cells and explanatory narratives contained within properly formatted Markdown cells. The Notebook should contain (at a minimum) the following sections (including the relevant Python code for each section):

1) **Introduction (5 Points)**: Summarize the problem + explain the steps you plan to take to address the problem

2) **Exploratory Data Analysis (15 Points)**: Explain + present your EDA work including any conclusions you draw from your analysis, including any preliminary predictive inferences. This section should include any Python code used for the EDA.

3) **Data Preparation (10 Points)**: Describe + show the steps you have taken to address the data integrity + usability issues you identified in your EDA, including any feature engineering techniques you have applied to the data set. This section should include any Python code used for Data Preparation.

4) **Prepped Data Review (5 Points)**: Explain + present your post-Data Prep EDA analysis. This section should include any Python code used for re-running your EDA on the variables adjusted during your Data Preparation work.

5) **Decision Tree + Random Forest Modeling (40 Points)**: Explain + present your decision tree and random forest modeling work, including your feature selection / dimensionality reduction decisions and the process by which you selected the hyperparameters for your models. This section should include any Python code used for feature selection, dimensionality reduction, and model building.

6) **Select Models (15 Points)**: Explain your model selection criteria. Identify your preferred model. Compare / contrast its performance with that of your other models. Discuss why you've selected that specific model as your preferred model. Apply your preferred model to the testing subset and discuss your results. Did your preferred model perform as well as expected? Be sure include any Python code used as part of your model selection work and to frame your discussion within the context of the classification performance metrics you have derived from the models.

7) **Conclusions (10 Points)**

**Your Jupyter Notebook deliverable should be similar to that of a publication-quality / professional caliber document and should include clearly labeled graphics, high-quality formatting, clearly defined section and sub-section headers, and be free of spelling and grammar errors. Furthermore, your Pythion code should include succinct explanatory comments.**

Upload your Jupyter Notebook within the provided M11 Assignment Canvas submission portal. Be sure to save your Notebook using the following nomenclature: **first initial_last name_M11_assn**" (e.g., J_Smith_M11_assn). ***Small groups should identity all group members at the start of the Jupyter Notebook and each team member should submit their own copy of the team's work within Canvas.***