**W207 Final Project Baseline**

Class Section: 2019 Summer - Monday 6:30 PM

Project Team: Adam Sohn, Curtis Lin, Erik Hou, Youzhi (Chloe) Wu

# Selected Dataset

Forest Cover Type Dataset (https://www.kaggle.com/c/forest-cover-type-prediction/data)

# Problem Description

Need to have a good problem description in baseline submission. To predict forest canopy cover type

Need to show good understanding of data, show some EDA on data

# Data Preparation

EDA

Data clean up

Data transformation

# Overall Approach

Ensemble of algorithms:

* **Feature Selection**: Random Forest Regressor to rank features by their score (<https://blog.datadive.net/selecting-good-features-part-iii-random-forests/>)

# Preliminary Results (Accuracy) - No Feature Eng

Random Forest 0.8466

Support Vector Model 0.7824

k-Nearest Neighbors 0.7822

Decision Tree 0.7751

Logistic Regression 0.6629

Gaussian NB 0.5933

Adaboost w/ Decision Tree 0.4190

# **Questions for Yacov**

Sensible Method:

Note from office hours:

* Base submission goal is to demonstrate good understanding of problem, not code focussed.
* Seek & include domain knowledge to prompt feature engineering
* Presentation may mention many classifiers, yet focus report on the optimal classifier. Discuss why optimal classifier is optimal.
* Feature Engineer multiple classifiers to learn magnitude of possible improvements. A suboptimal classifier can be used as a learning vehicle for understanding features and feature engineering.
* Treatment for binary vs. non-binary variables is needed. This could be using enemble methods for multiple models.

1. What is the recommended number of classifiers to include in the final result for error analysis and optimization?
2. Is the project’s goal just to build one classifier that works well? Or we are also expected to dive deep into the comparison of different classifiers.a
3. What the general process for choosing classifiers should be like?
   1. Do we just use the same data (same transformation) on different classifiers, pick one that performs the best and optimize?
   2. Or, we try to tailor-engineer the features for classifiers first, compare the performance. Then pick the best performing one to optimize?
4. When comparing results from multiple classifiers, how do we know which classifier has best ROI potential for further enhancements?

Feature Engineering:

1. For soil type 7, all values are zero. Should this feature be included in the model?
2. Since feature engineering has not been covered in detailed yet, what are some useful feature engineering techniques besides scaling data.