**Instructions**

In this problem, you will apply different classification methods. You will use a Rock dataset where you will use 19 different rock features to predict the rock category. The data you need are included in these two files: 1) [aggregateRockData.xlsx](https://iu.instructure.com/courses/2201574/files/167078715?wrap=1)[Download aggregateRockData.xlsx](https://iu.instructure.com/courses/2201574/files/167078715/download?download_frd=1)you will only use 2nd column that contains the rock category number (1 = Igneous, 2 = Metamorphic, 3 = Sedimentary) - that will be the label. 2) [norm540.txt](https://iu.instructure.com/courses/2201574/files/167078714?wrap=1)[Download norm540.txt](https://iu.instructure.com/courses/2201574/files/167078714/download?download_frd=1)you will only use columns 4 to 22 - those will be the attributes (features). See this website for a detailed description of the dataset: [https://osf.io/cvwu9/wiki/Data%20File%20Descriptions/. Links to an external site.](https://osf.io/cvwu9/wiki/Data%20File%20Descriptions/)

We will use only the first 480 rows (so ignore rows 481 to 720).

**Answer the questions below directly in your Jupyter Notebook, using Markdown cells**. Be sure to clearly indicate that your comment is an answer to a particular question.

1. Display the statistical values for each of the attributes, along with visualizations (e.g., histogram) of the distributions for each attribute. Are there any attributes that might require special treatment? If so, what special treatment might they require? **[2 points]**
2. Analyze and discuss the relationships between the data attributes and between the data attributes and labels. This involves computing the Pearson Correlation Coefficient (PCC) and generating scatter plots. **[3 points]**
3. For training data, use token numbers 1-10, for validation 11 to 13, and for testing 14 to 16 (each of the 30 rock subtypes has 16 token numbers). **[2 points]**
4. Train different classifiers and tweak the hyperparameters to improve performance (you can use the grid search if you want or manually try different values). Report training, validation and testing performance (classification accuracy, precision, recall and F1 score) and discuss the impact of the hyperparameters (use markdown cells in Jupyter Notebook to clearly indicate each solution):
   1. Multinomial Logistic Regression (softmax regression); hyperparameters to explore: C, solver, max number of iterations. **[10 points]**
   2. Support vector machines (make sure to try using kernels); hyperparameters to explore: C, kernel, degree of polynomial kernel, gamma.  **[10 points]**
   3. Random Forest classifier (also analyze feature importance); hyperparameters to explore: the number of trees, max depth, the minimum number of samples required to split an internal node, the minimum number of samples required to be at a leaf node. **[10 points]**
5. Combine your classifiers into an ensemble and try to outperform each individual classifier on the validation set. Once you have found a good one, try it on the test set. Describe and discuss your findings. **[8 points]**
6. Is your method better than a human? Test that by taking human data from [trialData.csv](https://iu.instructure.com/courses/2201574/files/170264991?wrap=1) Compute human accuracy on train and test data (use only rocks with numbers 1 to 480 and note that *Block number* 1-3 is training, number 4 is test). How does it compare to the accuracy of your best model? **[2 points]**Then, compute the average human accuracy for each of the 480 rocks and the standard deviation of accuracy for each of the 480 rocks (regardless of whether they are train or test rocks). Make a plot with the x-axis showing rock numbers (1 to 480) and the y-axis showing average human accuracy (values between 0 and 1) and standard deviation for each of 480 rocks. **[2 points]** Then, add the accuracy of your model for each of those rocks (1 for correct classification and 0 for incorrect classification) on the same plot (in a different color). What do you observe - is your model making similar errors as humans?**[1 point]**