**Topic:** Training Classifiers and Evaluating Them with Python

1. **Task 1:** For the wine data set provided (wineData.csv): (1) open the file and read it to a Panda's data frame, (2) identify Class attribute and perform class mapping, (3) normalize all remaining attributes to (0, 2) range using Min-Max normalization, (4) save the entire data set as wineNormalized.csv file. What can you tell me about this data set? E.g. What is the size of the data set? How many descriptive attributes do you have in your file? Are the classes balanced?

2. **Task 2:** Read the data from your wineNormalized.csv file to the Pandas data frame and split your instances into training (2/3) and testing (1/3) data sets (you will need to perform stratified holdout sampling, as I want you to make sure you have an even-out number of class labels in each of these two sets). Save your training and testing data sets as *.csv files, as you will need them to complete the remaining tasks. In this task you will be working with Decision Trees, so perform experiments that involve trees with different numbers of levels, and different split measures (e.g. Gini index, entropy). Which Decision Tree is the best in your opinion (provide sufficient information about your winning classifier, so I could re-produce it easily) and why? Show your score charts (generated for training and testing sets) to support your recommendation. Finally, visualize the winning tree and include it in your written report.
3. **Task 3:** Now, using exactly the same data (i.e., training and testing data sets, which were generated and saved in Task 2, need to be loaded here), and evaluation methodology as in Task 2, investigate kNN classifiers. Perform experiments that involve at least two different similarity measures, different k values, and different neighbors-weighting scenarios. Which kNN classifier of the ones you investigated is the best in your opinion (provide sufficient information about your winning classifier, so I could re-produce it easily) and why? Show your score charts (generated for training and testing sets) to support your recommendation.
4. **Task 4:** Now, using exactly the same data as in the last two tasks, implement Random Forest to rank all of your descriptive features based on their importance. Present the ranking you were able to generate in the report, and discuss how you could use this knowledge to your advantage. Would it be fair to compare your Random Forest classifier against the winning classifiers from Task 2 and 3? Out of these three classifiers, which algorithm, when run on your data, has a better chance to win this comparison in your opinion and why? How would you implement a fair comparison?
5. **Task 5:** Now, it is time to have some free play! Come up with some ideas to improve your machine learning results achieved in the tasks above,and test them. Try to generate/derive some new descriptive features and take advantage of feature ranking or other types of classifiers to improve your results. Make this a comparable/fair investigation thought, stick to the same training and testing data sets, and be consistent with your evaluation methodology. Were you able to beat the best results from the tasks above? Yes or no (state clearly), and why? Any ideas that you could investigate if you would have more time or more resource