Homework 2 for CSI 431
Due: Nov 5 at 23:59:59

All homeworks are individual assignments. This means: write your own solutions and do not copy code/solutions from peers or online. Should academic dishonesty be detected, the proper reporting protocols will be invoked (see Syllabus for details).

Instructions: Submit two files. One should be a write-up of all solutions and observations, as *Solution.pdf*. The second should be an archive *Code.zip* containing code and any relevant results files.

1. **[100+ pts.] Classifier Evaluation:** This assignment will involve comparison of the LDA, Decision Tree, and SVM (linear kernel) classifiers as implemented in scikit-learn. Use the example Demo script we discussed (in Blackboard) in class for code examples of how to create classifiers. Work with the datasets attached to the assignment. The data comes from breast cancer diagnosis where each sample (30 features) is labeled by a diagnose: either M (malignant) or B (benign)[recorded in the 31-st column in the datasets].

   In all cases we will compare the classifiers based on the average F-measure in 10-fold cross validation of the training set provided. *Note: The demo script does not show you how to do cross-validation in scikit-learn, you need to learn about the cross_validate function from the library and employ it to do the analysis (`http://scikit-learn.org/stable/modules/cross_validation.html`)*

   (a) **[30 pts.] Configure SVM:** Learn the optimal parameter C of the linear SVM for this dataset. To do this, plot the average F-measure in 10-fold cross validation for linear SVMs with values of $C = \{0.01, 0.1, 1, 10, 100\}$. On the x axis of this figure you should have the values of C and on the y axis the corresponding average F measure. *Use cancer-data-train.csv for this analysis.*

   Discuss your observations. Is smaller or larger margin better for this dataset (need to explain which C values are likely to produce smaller v.s. larger values and then which end up being better in cross-validation.)

   (b) **[30 pts.] Configure the Decision Tree:** We will consider two kinds of decision trees based on the split criterion: Gini Index (call these trees DT-gini) and Information gain (Call these trees DT-ig) (see the *criterion* parameter for DT initialization:`http://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html`).

   For both DT-gini and DT-ig we will identify the best size of the tree. To do that, we will allow up to k leaf nodes for each of the decision trees (see *max_leaf_nodes* parameter in `http://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html`). Train DT-gini and DT-ig trees of maximum leaf-nodes: $k = \{2, 5, 10, 20\}$ and plot their average F-measure in 10-fold cross validation as a function of k. On the x axis of this figure you should have the values of k and on the y axis the corresponding average F measure. You will have two curves: one for DT-gini and one for DT-ig. *Use cancer-data-train.csv for this analysis.*

   Discuss your observations from the figures. Does larger tree mean better F-measure? Which criterion is better?

   (c) **[40 pts.] Compare classifiers:** Choose the best setting of C you identify from part (a), best size of the trees for DT-ig and DT-gini (Note, the optimal size might be different for the two criteria) from part (b). We will also include LDA in this comparison.

   Use the full cancer-data-train.csv for training. Train each of the above classifiers with the identified optimal parameters: SVM, DT-ig, DT-gini and LDA. *Use cancer-data-test.csv* to compute the corresponding confusion matrices for each classifier and plot: Average class precision, average class recall and average class F-measure. Plot these values in 3 bar charts (one for precision, one for recall and one for F-measure). Each plot should have 4 bars (one per classifier). *Hint: You have examples of how to compute some of these measures in the demo script. Need to first train with the first half, then obtain predictions for he second half of the data and then compare the predicted and actual classes to get a statistic.*

   Discuss your findings. Which are the best classifiers when you consider the different metrics? Is there a single winner for this dataset.

   (d) **Extra Credit - Learn how to use a new classifier:** Consider a classifier that is implemented in scikit-learn but we have not discussed in class: RandomForestClassifier. Learn how to create and train it in scikit-learn and add it to the comparison from the previous part (c). Explain the results.