To run program and recreate experiments:

1) treeCoverClassifiers is the "main" file

2) The first parameter in preProcessData() controls the file, the second controls PCA

a) Original dataset has 11 columns after preprocessing, so PCA can range from 1-11. Anything outside that range will result in PCA being skipped (which may be purposeful to turn PCA off)

3) The first parameter in kFoldCrossValidate\_standardClassifiers controls the number of folds, the fourth controls the model (in string form). The comment above this function details which strings are aceptable. Any other string will cause the function to return and the program to gracefully terminate prematurely.

4) The first parameter in kFoldCrossValidate\_neuralNetwork also controls the number of folds.

This sample code will run Decision Tree classifier on Wilderness Area 1, a subset of the dataset, with no PCA and 10 folds of cross validation. As the code runs, the words ‘fold …” will appear in the console to let the user know how far along program is (which fold it’s on at the moment).

% Preprocess the data. Collapses binary columns, normalizes data. Returns

% as many columns as specified by the second 'numPrincipalComponents'

% argument to preProcessData.

[data, labels] = preProcessData('covtype\_wildernessArea1.csv', 12);

%--------------------------------------------------------------------------

% Train standard classifiers and get their accuracy/results.

%--------------------------------------------------------------------------

% Train a Naive Bayes ('naiveBayes'), Decision Tree ('decisionTree'), or

% K-Nearest Neighbor ('knn') classifier and get the accuracy/results

[trainingTime, predictionTime, accuracy, confusionMatrix, order ] = ...

kFoldCrossValidate\_standardClassifiers(10, data, labels, 'knn');

showResults(trainingTime, predictionTime, accuracy, confusionMatrix, order);

This is the result:

