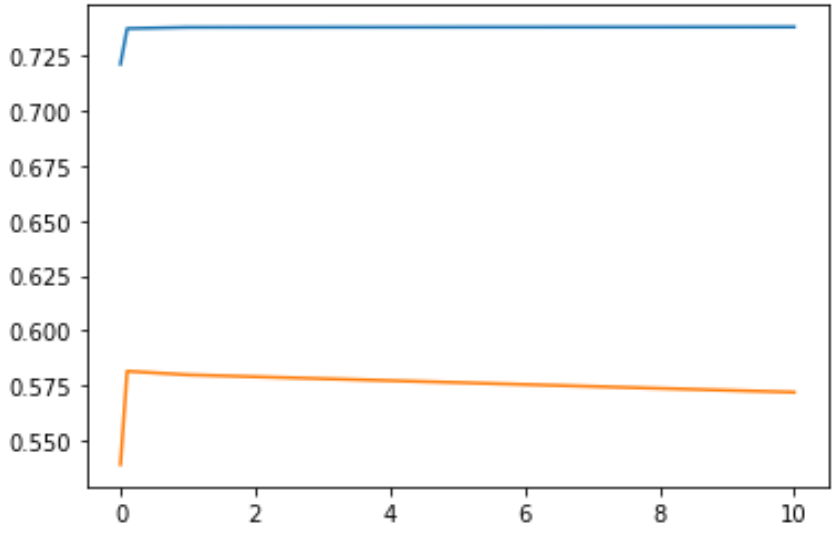
1. My project is aimed at predicting the forest cover type based on the surroundings.

2. I am working on the same dataset, Forest Cover Type Dataset from UCI Machine Learning, as I proposed in Deliverable1. I changed the number of data used for training, validation, and testing. 64% (371,848) of the dataset is used for training, 16% (92,962) for validation, and the rest 20% (116,202) for testing. There are 12 features of attributes, 10 quantitative features and 2 qualitative features. I used Min-Max scalar to normalize the quantitative dataset to use support vector machine effectively.

3. I used LinearSVC from sklearn.svm, since my dataset is very large and LinearSVC works well with large dataset. I split the dataset into three parts, and simply chose the first subset for training, the second for validation, and the third for testing. It might be better if each of the subset would contain similar number of the 7 cover type entries. I might change the training/validation/test splits later. Since the accuracy for training set is not very high, I think the model is underfitting. I used SVC first, and the running time was too long. Therefore, I chose LinearSVC instead. To increase accuracy, I also used KNN.

4. LinearSVC: The accuracy of training set is 73.78% and the accuracy of testing set is 58.0% using default parameters (C=1.0, class_weight=None, dual=True, fit_intercept=True, intercept_scaling=1, loss='squared_hinge', max_iter=1000, multi_class='ovr', penalty='l2', random_state=0, tol=0.0001, verbose=0.) I changed the value of C (0.001, 0.1, 1, 10). C = 0.1 gives the highest accuracy, but this result is still unsatisfying. When C is greater than or equal to 10, I got a ConvergenceWarning.



KNN: The accuracy of training set is 95.9% and the accuracy of validation set is 59.8%. The accuracy of training set is much higher than that given by LinearSVC, but the accuracy of validation set is similar. Given the high accuracy of training set, it is possible that KNN with n_neighbors = 5 is overfitting the training set. Also, KNN took much longer time than LinearSVC to give a prediction.

5. LinearSVC is very fast, however, the accuracy is too low. I will try other SVM to see if I can get higher accuracy. It is possible that a linear kernel is not suitable for this dataset, so LinearSVC cannot give high accuracy. I will try SVC with polynomial kernel to see if I can increase the accuracy of my model. I might also try SDG based method since it is also fast for large dataset.