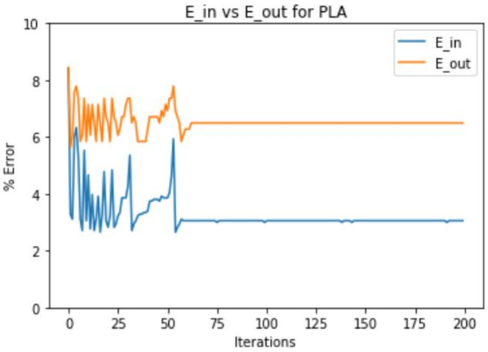
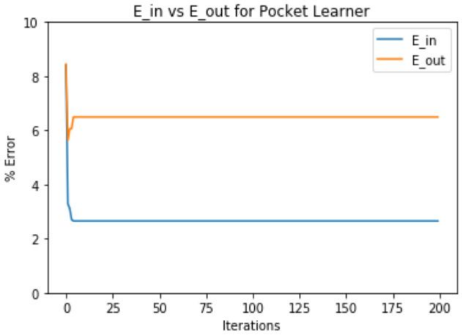
Name - Anantvir Singh Romana
UDID - 702455052

(b)

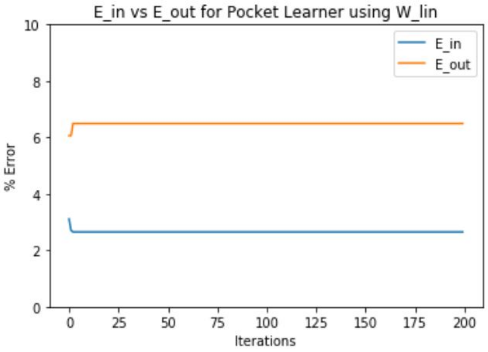


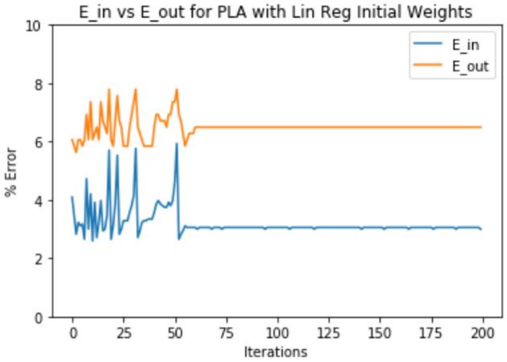


- Error on Training Data for Pocket Learner: 2.6497
- Error on Testing Data for Pocket Learner: 6.4935
- Error on Training Data for PLA was 3.0529
- Error on Testing Data for PLA was 6.4935

Hence by using Pocket Learning Algorithm, error on training data decreases, but there is no change in error on testing data. This suggests that Pocket Learner improves on training data i.e classifies all points until a minimum error is reached but does not generalize equally well out of sample. This might vary depending upon the dataset we get. Usually it’s more probable for Pocket Learner to perform better than PLA.

(d) E in vs E out for Pocket and PLA using Linear Regression Weights



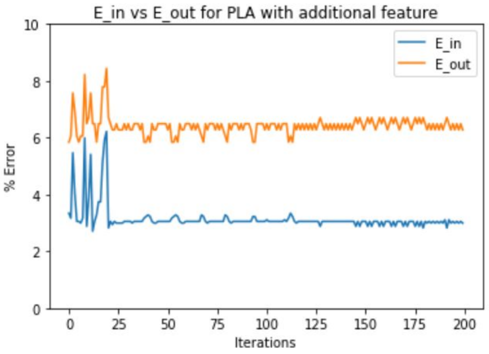


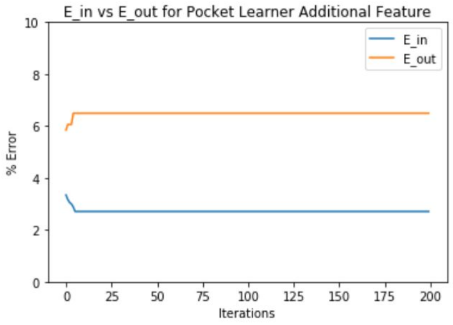
E in with PLA = 2.9953
E_out with PLA = 6.4935

Hence using Linear Regression weights E in decreases but E_out remains the same.

(e) The additional feature I have chosen is square of symmetry. Because this data is not linearly separable, so if we have a higher dimensional feature, then it might help make the data more sparse and hence more separable(not 100% separable). Hypothesis set will become larger and there is more chance for better classification. This higher dimensional feature can be quadratic in simplest form which can be (symmetry)^2 or (intensity)^2 or (symmetry)*(intensity) etc.
I tried all 3 and got best results using square of symmetry.
In PLA both E in and E out decrease with this additional feature

(f) Repeat (a) **Plots of E in vs E out for PLA and Pocket Learner with 3 features**





For PLA:

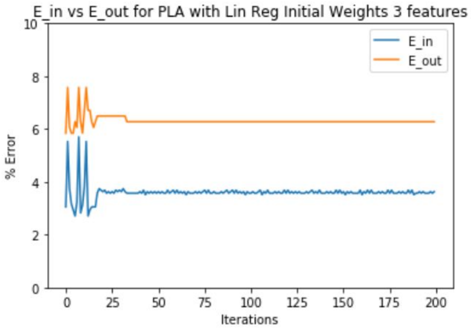
- Error on Training Data with 3 features = 2.9953
- Error on Testing Data with 3 features = 6.2770

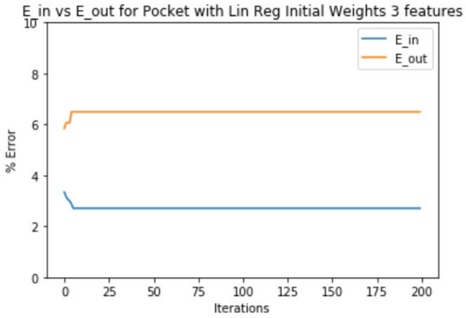
For Pocket Learner:

- Error on Training Data with 3 features = 2.7073
- Error on Testing Data with 3 features = 6.4935

Hence using Linear Regression weights, the training error decreases but there testing error increases. Hence out of sample generalization is poor in Pocket Learner as compared to PLA. For this specific problem Pocket should perform better because it draws a line which separates both classes with least error and when new examples arrive they should also have the same intensity and symmetry as the training examples because people write the digits in the same way. But since Pocket Learner's error increases, it might be because the new handwritten digits might be written in a very bad handwriting, because of which an example which should have high intensity and lower symmetry can have comparatively high symmetry and low intensity. Example a badly written 7 is testing data can resemble the digit 1.
7 should have high intensity than 1 and lower symmetry than 1, but badly written 7 can resemble 1 and hence can have high symmetry and low intensity and our Pocket Learner will miss-classify it.

(f) Repeat (d)





E in for Pocket is better than PLA but PLA generalizes better i.e E out for PLA is less than Pocket.

E_out PLA = 6.277

E_out Pocket= 6.4935