**Machine Learning HW2**

* 1. No we cannot, Logistic regression is a discriminative classifier, that estimates P(Y|X), not P(X|Y). In order to calculate parameters of the Gaussian Naïve Bayes, we need P(Y|X).
  2. By choosing a cut-point c, and classify those observations with a fitted probability above c as positive and those at or below it as negative. For this particular cut-off, we can estimate the sensitivity by the proportion of observations with Y=1 which have a predicted probability above c, and similarly we can estimate specificity by the proportion of Y=0 observations with a predicted probability at or below c. ROC curve is simply a plot of the values of sensitivity against one minus specificity, as the value of the cut-point c is increased from 0 through to 1.

We can calculate AUC using the formula for the area of a trapezoid. (As the area under ROC for two consecutive cut-points is a trapezoid.)

The AUC can also be seen as a concordance measure(c-index). If we take all possible pairs of observations where one is normal(Y=1) and the other is abnormal(Y=0), we can calculate how frequently it’s the abnormal one that has the highest (most ‘abnormal-looking’) test result (if they have the same value, we count that this as ‘half a victory’), R-code for this:

o = outer (abnorm, norm, "-")

mean((o>0) + .5\*(o==0))

* 1. When C goes to Infinity, SVM becomes a hard-margin and error rate increases duo to over-fitting.

2.2) **Tested parameters on validation data:**

max iteration: 100 learning rate: 1

max iteration: 1100 learning rate: 0.1

max iteration: 2100 learning rate: 0.01

max iteration: 3100 learning rate: 0.001

max iteration: 4100 learning rate: 0.0001

max iteration: 5100 learning rate: 1e-05

max iteration: 6100 learning rate: 1e-06

max iteration: 7100 learning rate: 1e-07

max iteration: 8100 learning rate: 1e-08

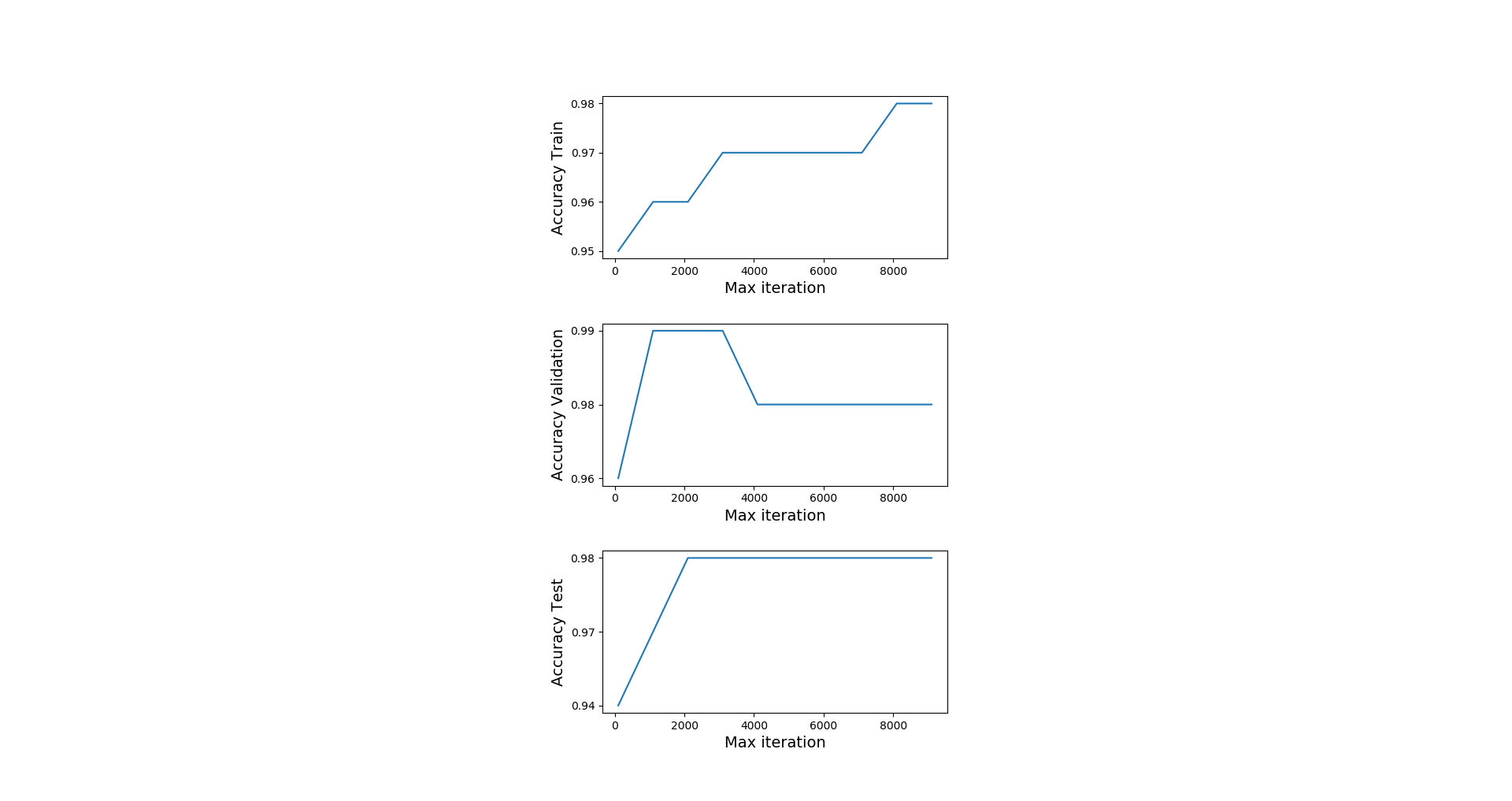
max iteration: 9100 learning rate: 1e-09

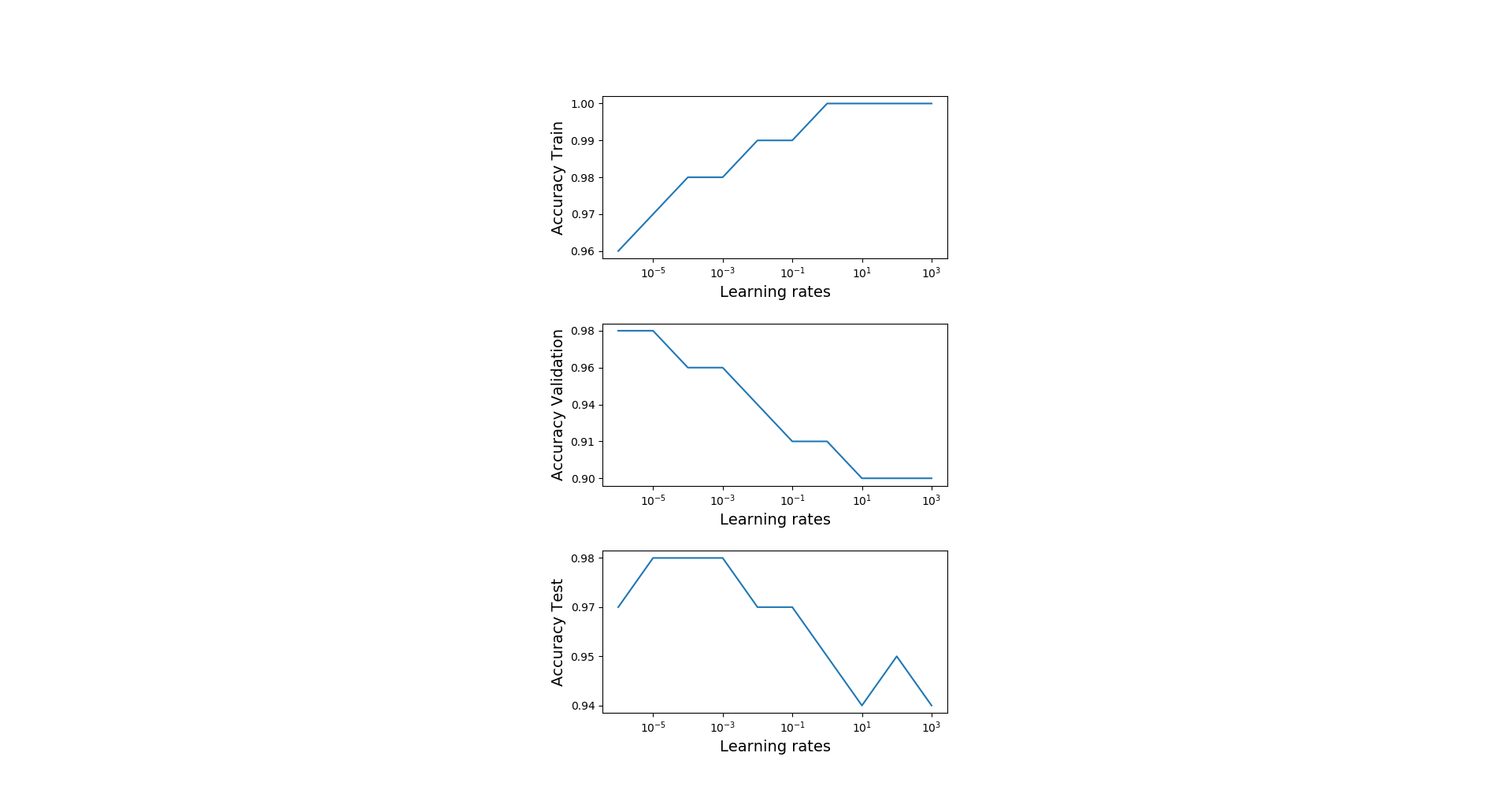
max iteration: 10100 learning rate: 1e-10

*Tuned parameters: max iteration: 5100 learning rate: 1e-05*

*Accuracy on test data using tuned parameters: 0.98*

2.3)





Tuned max iteration was 5100 and learning rate was 1e-05. On the training data fixed learning rate, as iteration number increases accuracy is increases also and after 8000 iterations it goes to %98. On the validation data, it gets %99 accuracy around 2000 iterations and on the test data it gets %98 accuracy after 2000 iterations.

On the training data fixed max iteration rate, as learning rate increases accuracy is increases also and after 1 it goes to %100. On the validation data, as learning rate increases accuracy is decreases and it gets %98 accuracy around 10-5. On the test data it gets %98 accuracy around learning rate 10-5. One observation is, accuracy is inversely proportioned to learning rate on validation and test data. On the other hand, accuracy is proportioned to learning rate on training data. Also, tuned learning rate 10-5 is doing very well on test and validation. Tuned max iteration also very good on test data and doing fair on validation.