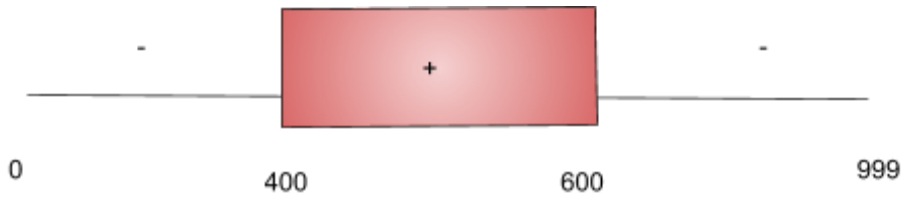


Case 1: no training sample in [400,600]

set all the [1,999] to “-”, so the error would be [400,600]

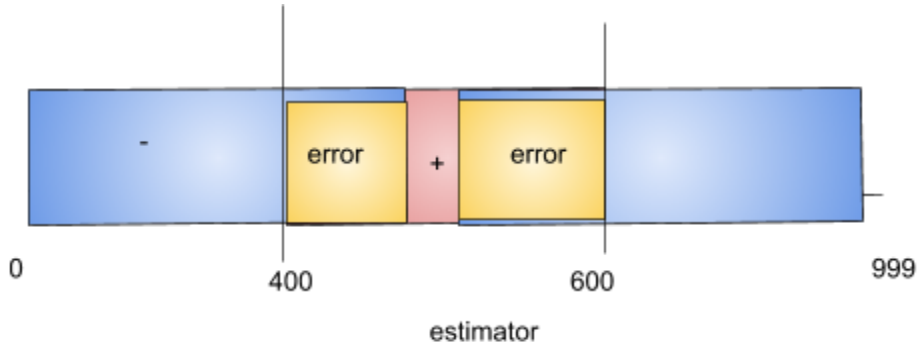
P (error)=[400,600]/[0,999]=0.2

So 0.2 is the maximum error rate.

Case 2: one sample in [400,600]

Set the point to “+”, other samples to “-”

p(error) =([400,600] - 1)/[0,999] = 0.199



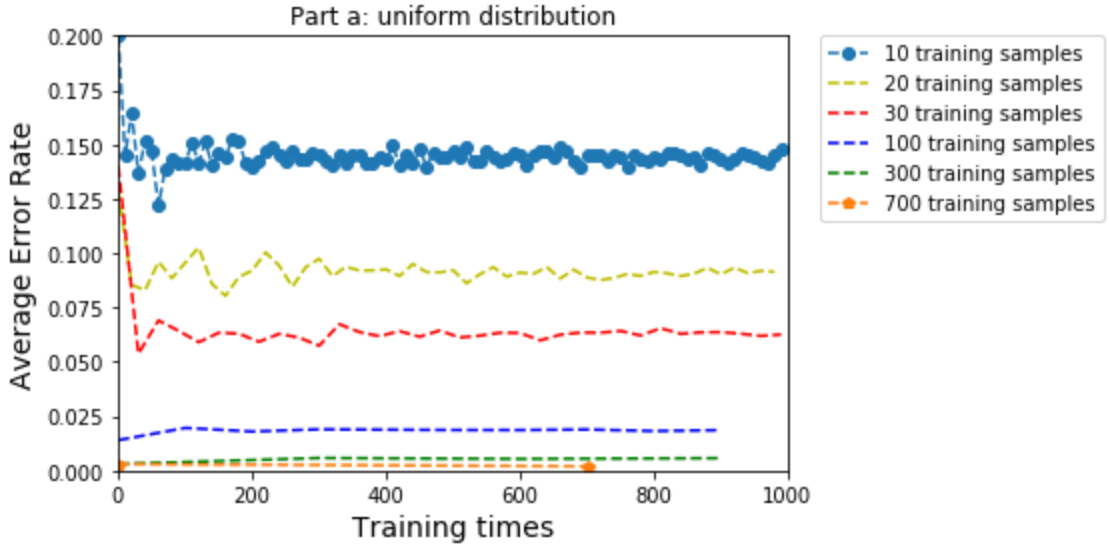
Case 3, more than 2 point in [400,600]

p(error) = ([min - 400] + [600 - max])/[0,999]

Average error rate = sum(error rate)/training times

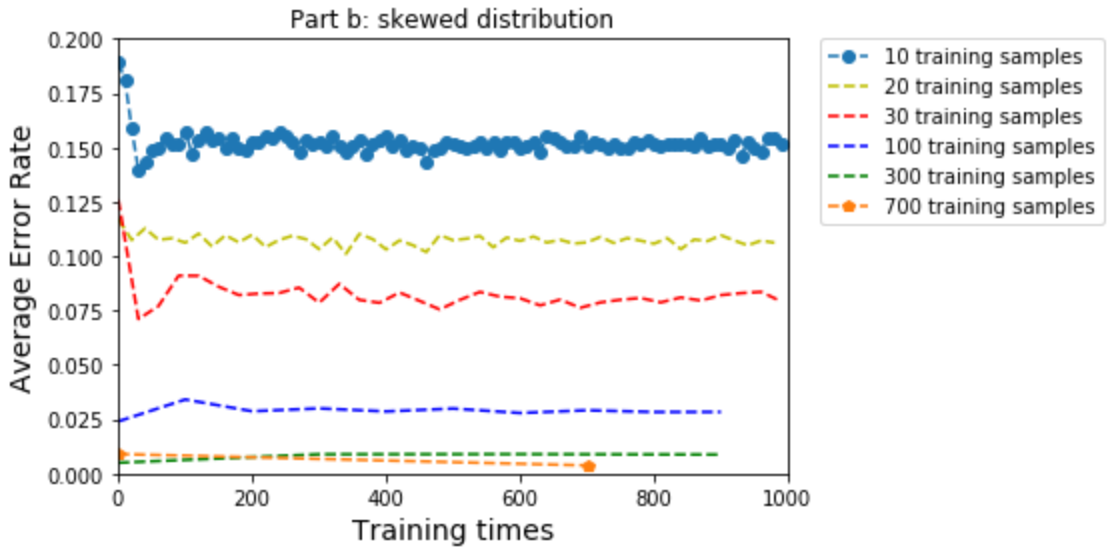
Failure times = 0;
If average > 0.05
Failure ++;

Failure probability = failure times/ training times



(b)

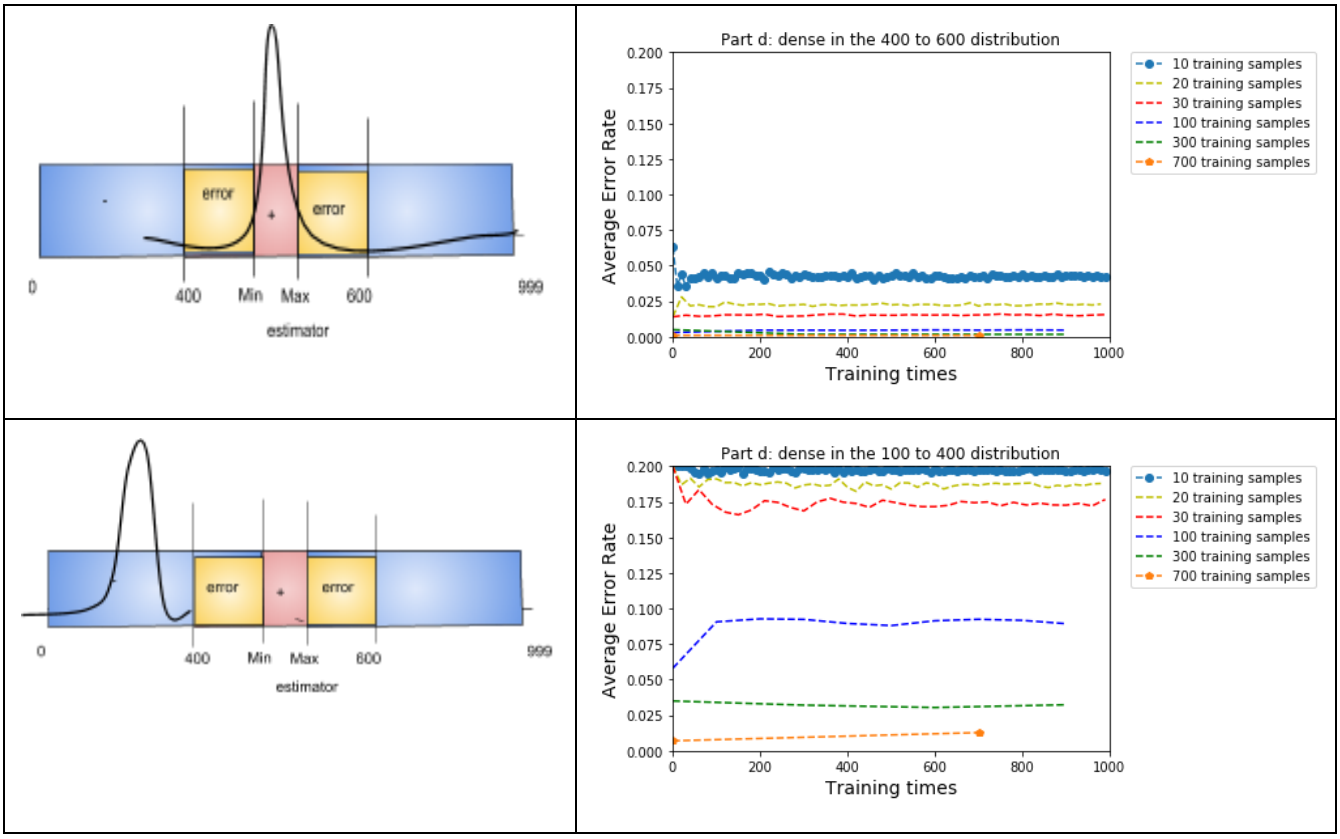
Generate skewed data by pick 80% from number < 500, 20% from 500-1000.



(c)

It seems like when both the training and test data are generated from skewed distribution, the failure probability and average error rate are similar to that from an uniform distribution. Both algorithms' hypotheses (uniform and skewed distribution) are similar accurate, and change with a similar trend when sample size changes.

（d) The test distribution with the most cases in [400, small_threhold) and (large_threhold, 600] is the worst. You can make 100% error when you know the value of small_threhold and large_threhold, and they are not equal to 400, 600, respectively.



As shown above, if the distribution of the sample dense (80%) in the range of [400,600], the error of the training show dramatically lower in all the sample size. If the distribution of the sample dense(80%) in the range of [1,400], the error of the training remarkably higher.

3.