Laishek, Tso (LT359)

Joe Yu (JDY28)

a.2

The result can be so small that the CPU cannot handle, as known as underflow.

a.5

The following accuracy test experiment is done with k = 1

	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Accuracy	0.613	0.639	0.636	0.653	0.664	0.673	0.683	0.689	0.687	0.686

Generally the accuracy increases along with the % training data use, which makes sense, practice makes perfect.

a.6

We tested different values of k with 100% of the training data:

	K=1	K=2	K=0.5	K=99	K=0	K=0.4	K=0.2	K=0.05	K=0.000001
Accuracy	0.686	0.652	0.716	0.455	error	0.723	0.748	0.789	0.823

It looks like as the k approaches 0, the accuracy improves. Which makes sense, because some features for certain labels are supposed to have 0 probability, adding the k manipulate it by giving it some false chances. So setting K to a lower value minimizes the manipulation of the probability, thus increase the probabilities. I think with k = 0.05 is reasonable, because in principle, nothing should get a absolute 0% chance. But for the in class competition, we will do k = 0.00000001 to maximize the accuracy b.1

We have the following features added:

Features[‘loop’] – detects whether the given digit_data contains loop

Features[‘topHeavy’] – detects whether the digit_data is heavy on the top side

Features[‘bottomHeavy’] – the opposite of top heavy

Features[‘leftHeavy’] - detects whether the digit_data is heavy on the left side

Features[‘rightHeavy’] – the opposite of left heavy

The following new features(without basic feature) accuracy test experiment is done with k = 0.05

	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Accuracy	0.3	0.285	0.285	0.285	0.3	0.29	0.3	0.29	0.29	0.29

Without the basic feature, the accuracy remains low, and they are around the same with 29%, as the features we implemented are supplement to the basic feature.

b.2

The following new features and together with the basic feature accuracy test experiment is done with k = 0.05

	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Accuracy	0.749	0.778	0.774	0.782	0.79	0.797	0.799	0.799	0.796	0.801

As expected, with the basic extractor together with the new features, the accuracy boosts with more training data.

With only basic feature, the accuracy = 0.789

With basic feature and 3 new features, the accuracy = 0.801