Assignment 1 - Report

Pietro Alovisi

The Implementation

The implementation is done in Matlab, following the suggested project structure. I achieved good computational performance by using vectorized operations. In particular I used the following matrices:

$$\mathbf{X} = \begin{pmatrix} \mathbf{x_1} & \mathbf{x_2} & ... & \mathbf{x_N} \end{pmatrix}$$
$$\mathbf{Y} = \begin{pmatrix} \mathbf{y_1} & \mathbf{y_2} & ... & \mathbf{y_N} \end{pmatrix}$$

Where $\mathbf{x_i}$ and $\mathbf{y_i}$ are the data column vectors. Then the gradient equations become:

$$\mathbf{P} = softmax(\mathbf{W} \cdot \mathbf{X} + \mathbf{b} \cdot \begin{bmatrix} 1 & 1 & ... & 1 \end{bmatrix})$$
$$\frac{\partial \mathbf{J}}{\partial \mathbf{W}} = \frac{1}{|B|}(\mathbf{P} - \mathbf{Y}) \cdot \mathbf{X^T} + 2\lambda \cdot \mathbf{W}$$
$$\frac{\partial \mathbf{J}}{\partial \mathbf{b}} = \frac{1}{|B|}(\mathbf{P} - \mathbf{Y})$$

To chech the correctness of the gradients I computed the *Frobenious Norm*$(\|\cdot\|_{\mathbf{F}})$ on the differences of the computed and true gradients with $h = 0.001$, and that value was in the order of 10^{-9} bor both.

Results

In the newxt subsections are presented the resutls for the different experiments.

Experiment 1

λ	eta	batches	epochs	Performance
0	0.1	100	40	$23.1\% \pm 4\%$

In Figure 1 is the loss evolution through the epochs, and in Figure 2 the ficture version of the rows of the weight matrix $\mathbf{W}$.

The loss curve is very wiggly because the learning rate is too high, this causes long jumps in the loss function landscape that can result in a worst network performance. This fact along with the lack of regularization causes the performance to have a high variance. The loss of regularization causes also the noise in the learned prototypes, because the parameter λ controls the smoothing of the values in $\mathbf{W}$, and therefore the smoothing of the images.

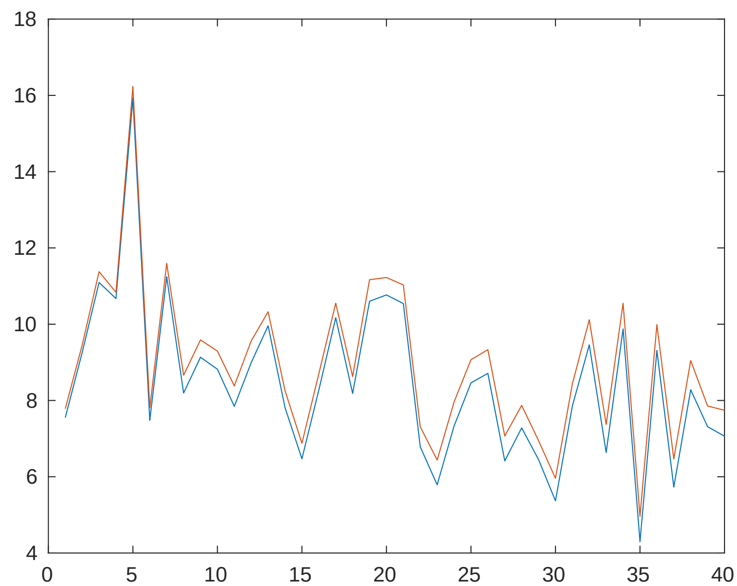

Figure 1: Loss curve for Experiment 1, blue test data and in orage the validation data.

Experiment 2

λ	eta	batches	epochs	Performance
0	0.01	100	40	$36.8\% \pm 1\%$

The results are show in Figure 3 and 4. Compared to the previous experiment the learning rate has been decreased, and we can see the result in the smoothness of the loss curve wich means a better, more stable set of final parameters $\mathbf{W}$ and $\mathbf{b}$. This also reflects in the performance of the network and its low variance. On the loss plot we can see the difference between test and validation dataset which increases over epochs. We don't have signs of overfitting because the linear

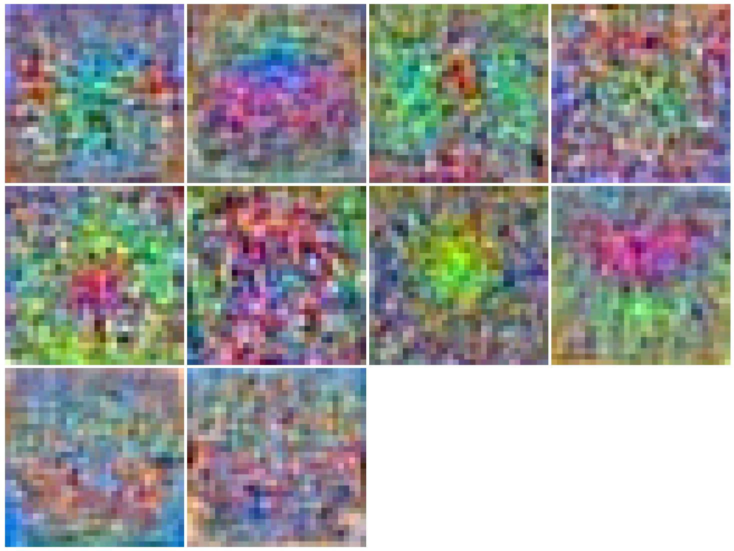

Figure 2: Learned prototypes in Experiment 1.

model is too simple to overfit with so much data. The learned prototypes are mroe clear now, but still the images are not smooth.

Experiment 3

λ	eta	batches	epochs	Performance
0.1	0.01	100	40	$34.3\% \pm 0.6\%$

The results are show in Figure 5 and 6. By increasing the regularization term we get smoother prototypes, and also a more stable performance. By looking at the loss plot, we can see how it quickly becomes steady for both validation and test data.

The more stable results are obtained through the regularization at the expenses of the performance, because regulariziing constraints the parameter space a little.

Experiment 4

λ	eta	batches	epochs	Performance
1	0.01	100	40	$21.5\% \pm 0.5\%$

The results are show in Figure 3 and 4. Here we can immediately tell the value for λ is too high, because the performance has gone too low and the loss saturates almos timmediately. What network is doing is trying to decrease more the regularization term than the term due to the croos-entropy loss.

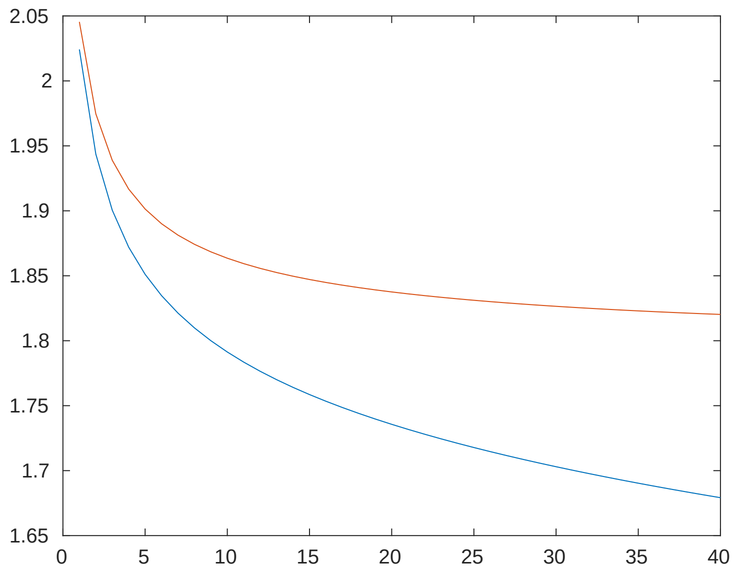

Figure 3: Loss curve for Experiment 2, blue test data and in orage the validation data.

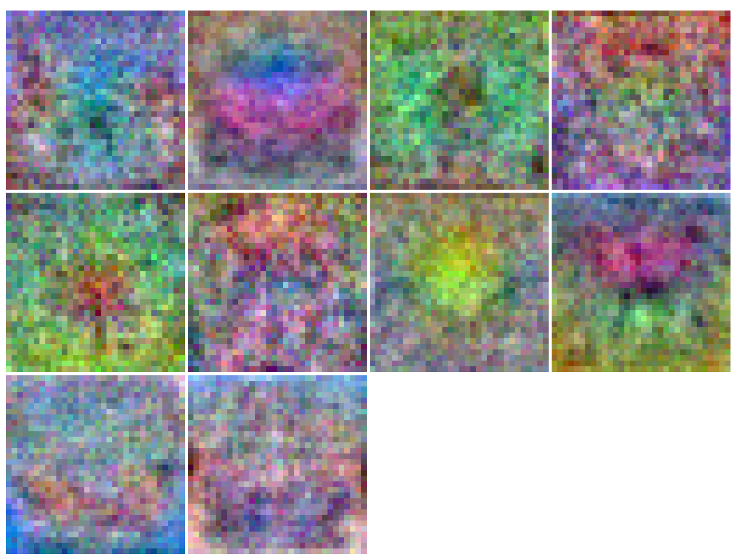

Figure 4: Learned prototypes in Experiment 2.

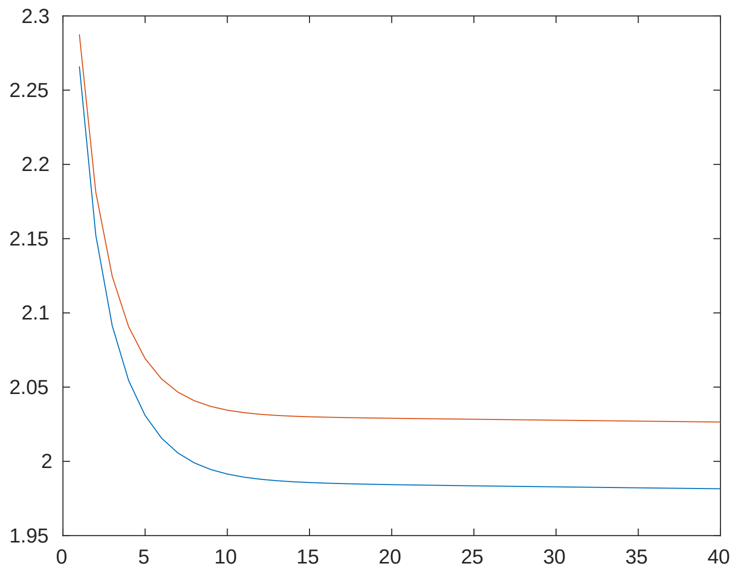

Figure 5: Loss curve for Experiment 3, blue test data and in orage the validation data.

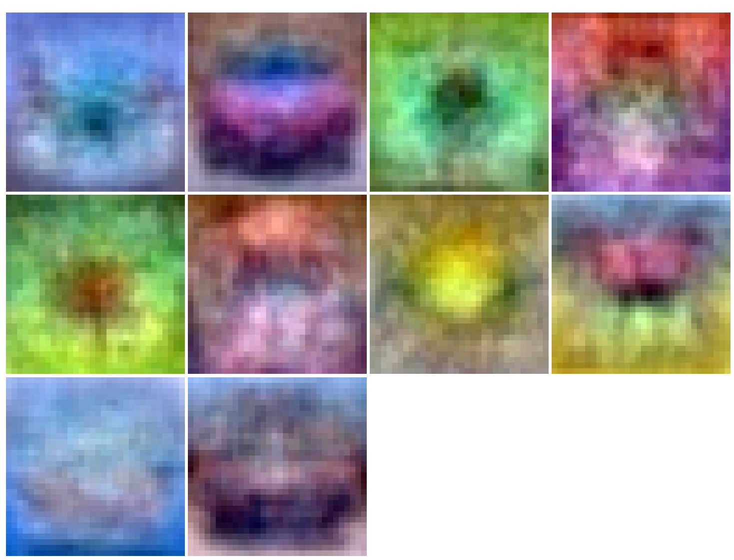

Figure 6: Learned prototypes in Experiment 3.

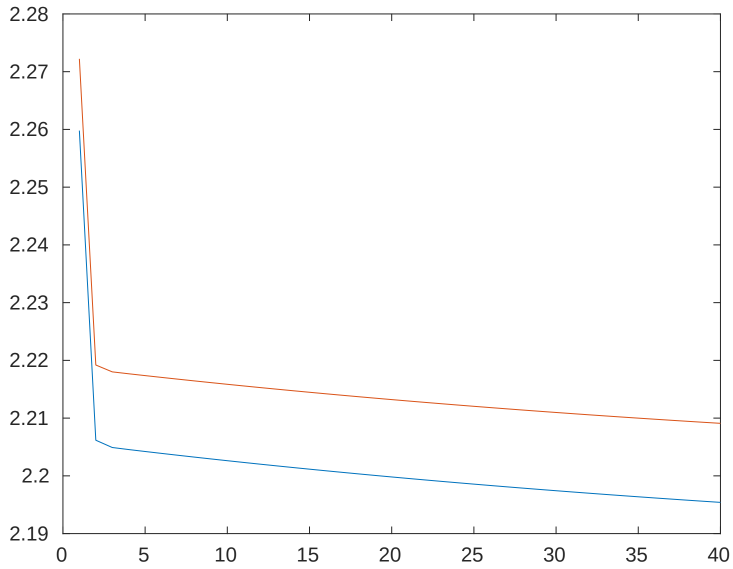

Figure 7: Loss curve for Experiment 4, blue test data and in orage the validation data.

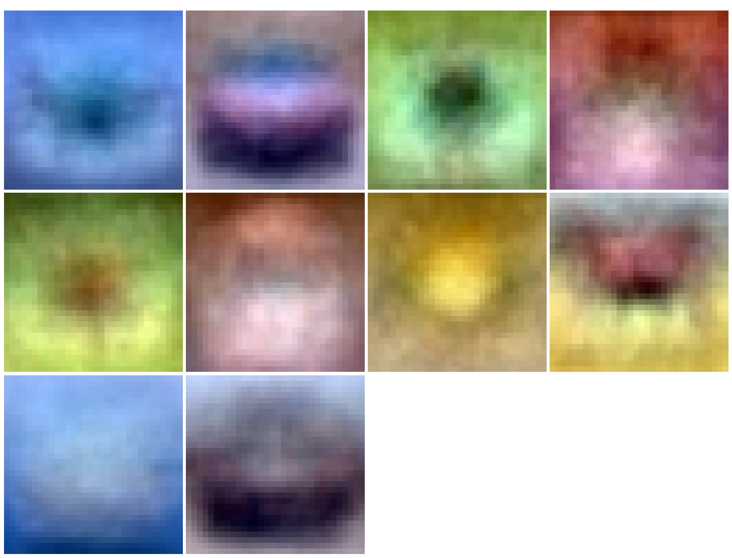

Figure 8: Learned prototypes in Experiment 4.