Homework #2 – Regularization and Optimization Techniques for Deep Learning

Ivan Liuliaev

Problem 1

Question 1 - Explain intuitively why L2 regularization is known as weight decay. Then explain how it is related to early stopping

L2 regularization adds a penalty that causes each weight to slightly shrink during each update, hence the calling.
Early stopping is a similar concept serving the same purpose (fighting overfit). But it does it differently - it stops training before the weights have a chance to become very large.

Question 2 - Explain intuitively why L1 regularization leads to more parameter sparsity (i.e., more parameter values are small and close to zero) than L2 regularization

L1 is able to “zero-out” distorting/useless weights, while L2 can’t (it only can get them close to zero, but not completely remove them from the “equation”)

Question 3 - What is the main advantage of using Algorithms 7.2 or 7.3 to train a model compared to using Algorithm 7.1?

Algorithms 7.2 and 7.3 use a validation set to determine the optimal training duration or overfitting point, while 7.1 is a basic early stopping algorithm. Using a two-phase approach usually tends to lead to a better generalization (lower final variance).

Problem 2

Question 1 - Explain the pros and cons of small and large minibatch sizes in Algorithm 8.1.

Small minibatch
Pros: More frequent updates, which sometimes helps explore the loss surface and escape local minima.
Cons: Noisier gradient estimates that can make training less stable.

Large minibatch
Pros: More accurate gradient estimates and smoother convergence.
Cons: Fewer updates per epoch, which slows down progress and reduces exploration.

Question 2 - Explain how the momentum term in Algorithm 8.2 helps overcome plateau regions and speed up learning in regions where the non-zero gradients are roughly constant.

It accumulates a running average of past gradients.

Question 3 - Explain when and how the Adam algorithm in Algorithm 8.7 gives better estimation of the momentum term than the basic momentum algorithm in Algorithm 8.2.

When: when handling noisy or sparse gradients.
How: keeping track of both the first (mean) and second (variance) moments of gradients and using bias correction to adjust these estimates.

Question 4 - Explain how the learning rates are adaptive in Algorithms 8.4 and 8.5.

AdaGrad (8.4): Accumulates the squared gradients over time and divides the learning rate by the square root of this sum, reducing the learning rate for parameters with high gradient magnitudes.
RMSProp (8.5): uses an exponentially decaying average of squared gradients to prevent the learning rate from shrinking too much. It also adapts the learning rate for each parameter, making training more balanced.

Problem 3

Question 1 - What are the expected side effects of dropout, where additional noise is introduced by randomly removing neurons during training?

Slower convergence and less stable training.

Question 2 - Dropout is often regarded as an efficient way to implement bagging. Identify and explain the differences between bagging and dropout.

Bagging trains many separate models on different subsets of data and then averages their predictions, while Dropout simulates an ensemble within one network by randomly dropping neurons during training.

Question 3 - What are the expected side effects of batch normalization, where normalization reduces the expressive power of the involved layers?

It limits the range of activations and how flexible each layer can be in representing complex patterns.

Question 4 - While it is well defined how to perform normalization during the training phase on a minibatch, what to do at the test time is not clearly defined. Explain expected undesirable side effects.

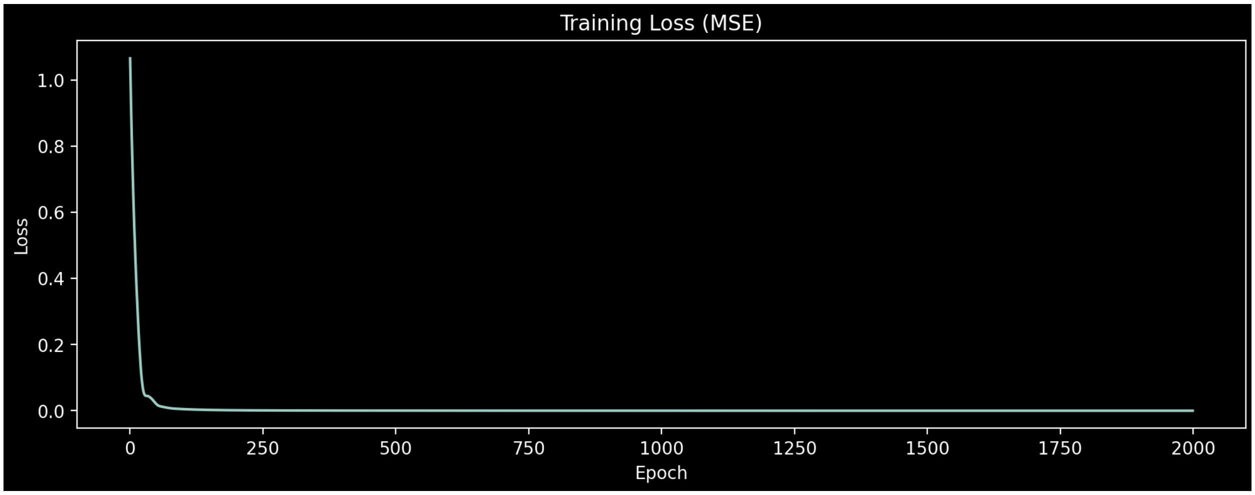
At test time, the network typically uses fixed mean and variance instead of batch statistics. If these fixed values differ from the true distribution of test data, the normalization might misrepresent activations which will lead to reduced performance or unexpected behavior.

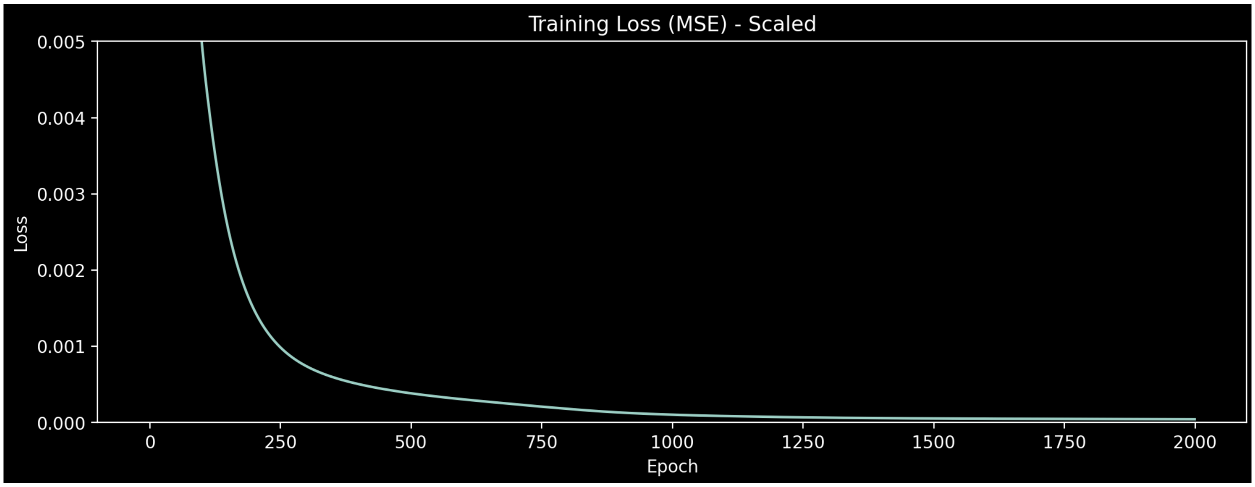
Question 5 - Based on your understanding, explain how dropout and batch normalization should be used jointly.

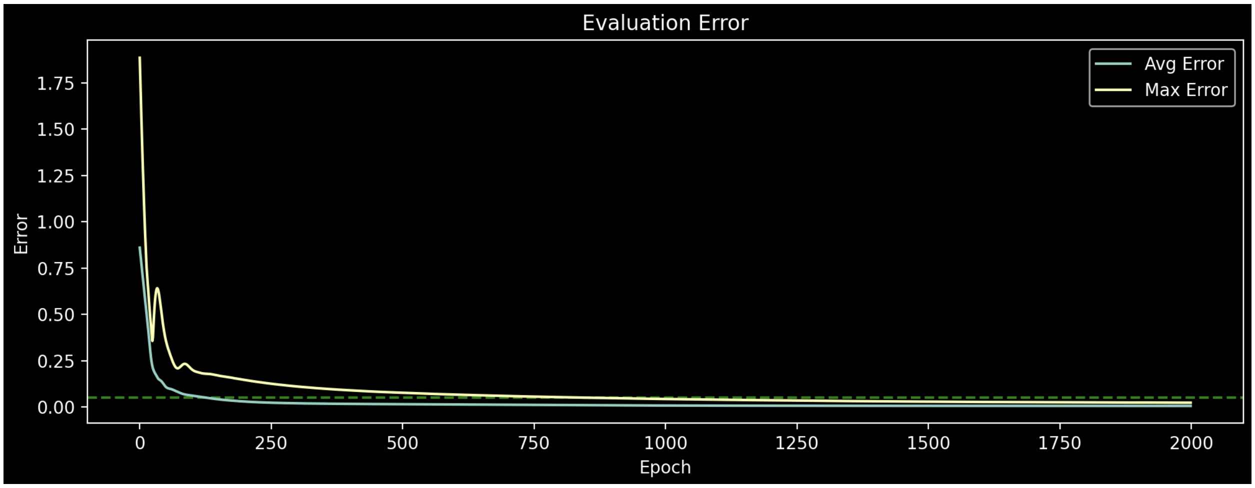
Batch normalization should be applied before dropout because it standardizes activations, and dropout then adds regularization. Also, overusing dropout might undermine the benefits of batch normalization.

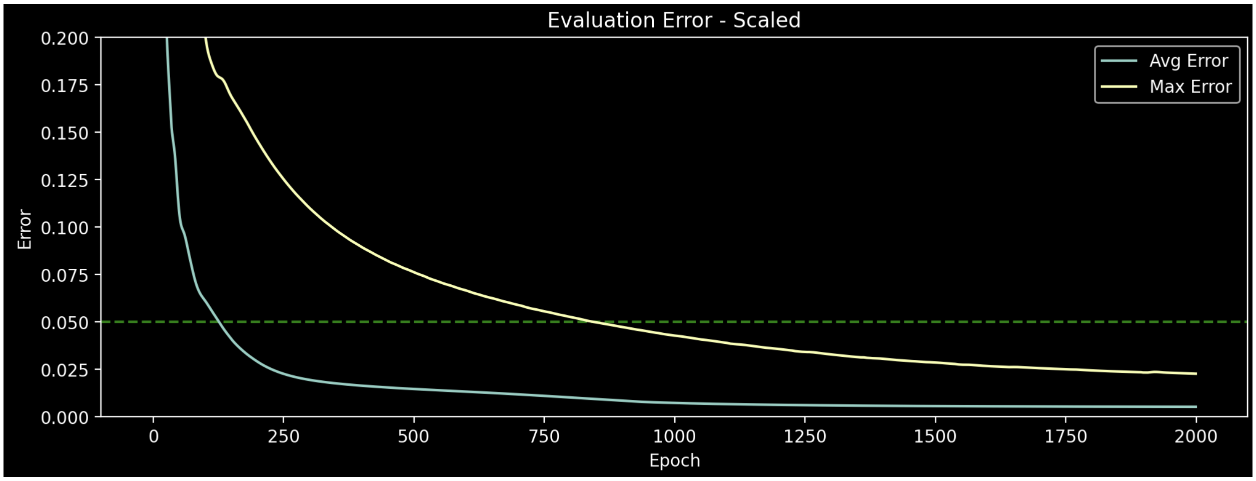
Problem 4

Question 1

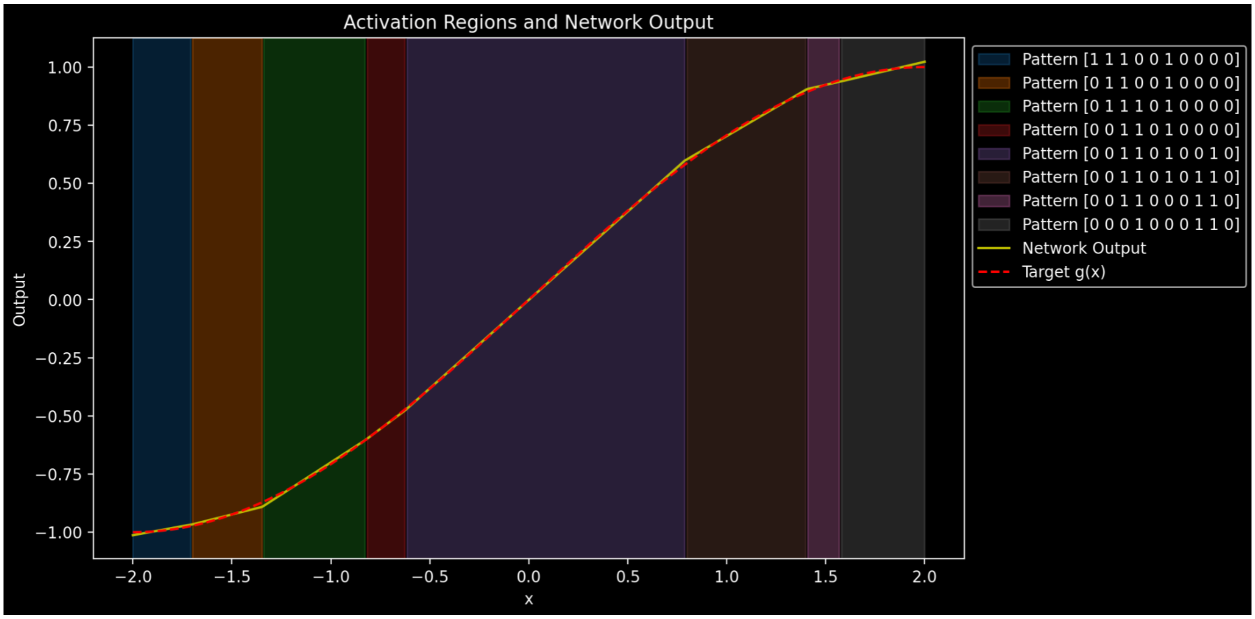


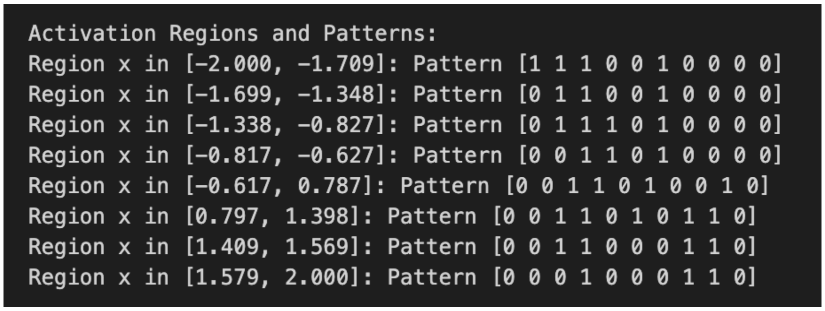



Evaluation Error
Avg Error
Max Error
Error
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
0
250
500
750
1000
1250
1500
1750
2000
Epoch


Evaluation Error - Scaled
Avg Error
Max Error
Error
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
0.200
0
250
500
750
1000
1250
1500
1750
2000
Epoch

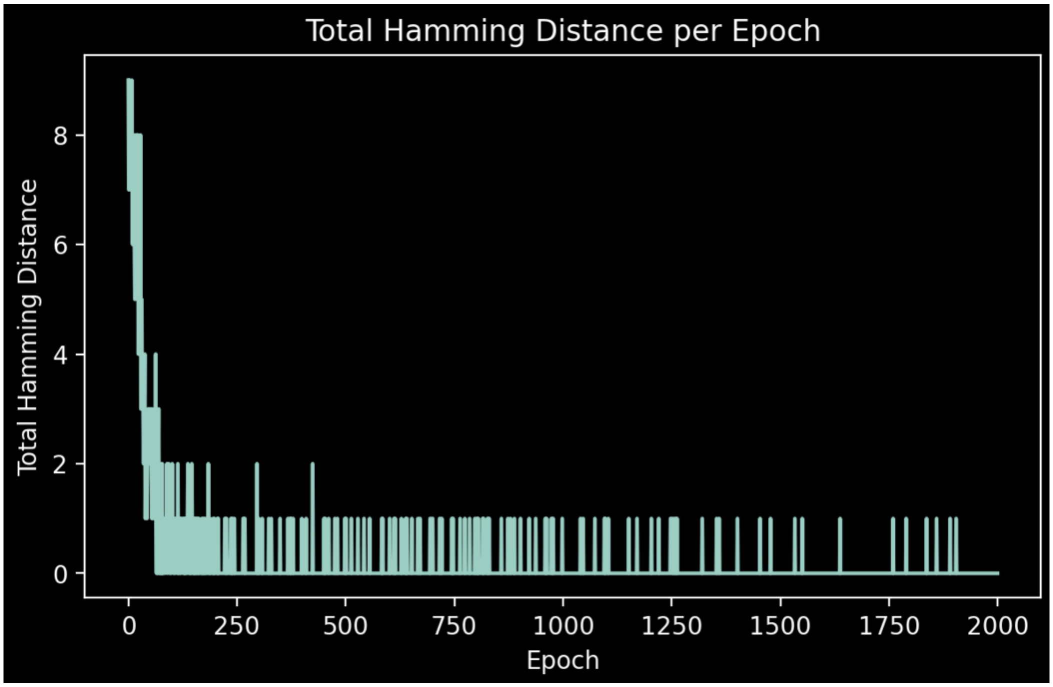
Question 2





```
Activation Regions and Patterns:
Region x in [-2.000, -1.709]: Pattern [1 1 1 0 0 1 0 0 0 0]
Region x in [-1.699, -1.348]: Pattern [0 1 1 0 0 1 0 0 0 0]
Region x in [-1.338, -0.827]: Pattern [0 1 1 1 0 1 0 0 0 0]
Region x in [-0.817, -0.627]: Pattern [0 0 1 1 0 1 0 0 0 0]
Region x in [-0.617, 0.787]: Pattern [0 0 1 1 0 1 0 0 1 0]
Region x in [0.797, 1.398]: Pattern [0 0 1 1 0 1 0 1 1 0]
Region x in [1.409, 1.569]: Pattern [0 0 1 1 0 0 0 1 1 0]
Region x in [1.579, 2.000]: Pattern [0 0 0 1 0 0 0 1 1 0]
```

Question 3



Question - Is the overall trend consistent with your expectation?

Yes, it is consistent with the expectations.

Early in training, the network experiences significant changes in activation patterns as it adjusts its weights. Once the network converges, Hamming distances are mostly near 0 or 1. Low late-training mean indicates that the model has reached a stable state, which is what is expected.