Assignment 1

# Successfully write the functions to compute the gradient analytically

Completed

# Tests to check against the numerically computed gradient

Calculating for W[0:10, 0:20] the gradient with the provided function “ComputeGradsNumSlow” and comparing with the values I get when calculating W for one epoch.

# Results



Extremely small deviations between the numerical and analytical gradient.

Comparison between W[0:10, 0:20] of the numerical gradient with the analytical updated W[0:10, 0:20] with lambda=0, n\_batch=100, eta=0.001

# Graphs of the loss and the cost function

**lambda=0, n epochs=40, n batch=100, eta=.1**

**A graph with blue and orange lines

Description automatically generatedA graph with blue and orange lines

Description automatically generated**

**lambda=0, n epochs=40, n batch=100, eta=.001**

**A graph of a graph with blue and orange dots

Description automatically generatedA graph of loss and loss

Description automatically generated**

**lambda=.1, n epochs=40, n batch=100, eta=.001**

**A graph of a graph with blue and orange dots

Description automatically generatedA graph of loss and validation

Description automatically generated**

**lambda=1, n epochs=40, n batch=100, eta=.001**

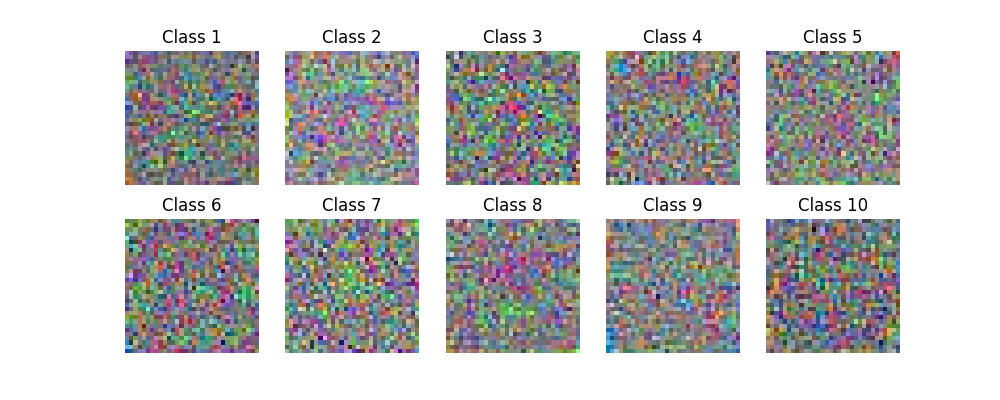
**A graph of a graph with blue and orange dots

Description automatically generatedA graph of loss and validation

Description automatically generated**

# Images representing the learnt weight matrix

**lambda=0, n epochs=40, n batch=100, eta=.1**

****

**lambda=0, n epochs=40, n batch=100, eta=.001**

A group of squares with text

Description automatically generated

**lambda=.1, n epochs=40, n batch=100, eta=.001**

**A group of squares with text

Description automatically generated**

**lambda=1, n epochs=40, n batch=100, eta=.001**

**A group of colorful squares

Description automatically generated**

# Test accuracy

**lambda=0, n epochs=40, n batch=100, eta=.1**

Test Accuracy: 0.2484

**lambda=0, n epochs=40, n batch=100, eta=.001**

Test Accuracy: 0.3769

**lambda=.1, n epochs=40, n batch=100, eta=.001**

Test Accuracy: 0.3796

**lambda=1, n epochs=40, n batch=100, eta=.001**

Test Accuracy: 0.3674

# Effect of increasing the amount of regularization and the importance of the correct learning rate.

A learning rate of 0.001 showed to have a lot better accuracy then with 0.1.

Including a regularization term increased accuracy by a very small margin. However, only when the regularization term was 0.01, if it was 1 accuracy decreased. Also with a regularization term the visualizations of the weights seem slightly more diffuse which could be an indication that less overfitting is happening and the model tries to use more generalized patterns.