Aaron Shabanian

Machine Learning

Assignment 5

Assignment 5 Writeup

Introduction

In this assignment there were three main parts. First, I made a Convolution Neural Network on the MNIST dataset. Second, I made a conditional generative adversarial network , CGAN, for MNIST data augmentation. Third, I made a conditional variational autoencoder, CVAE and compared it to the CGAN in determining which worked better.

Background

As we learned earlier, Convolutional Neural Networks are great models to use to classify images and find certain features in them. They are good at this because of their use of weight sharing. Weight sharing in CNNs is done by the use of a filter that goes through the entire picture. GANs help us generate pictures, audio or even data from some input. Conditional GANs specifically generate media that is conditioned in a certain class. A Conditional VAE is similar to a CGAN in that they both generate images but they do so in different ways. A VAE uses an encoder and a decoder to generate images while a GAN uses a generator and a discriminator to generate new images.

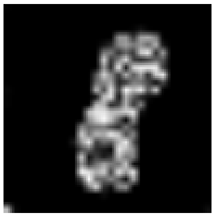
Findings

Convolutional Neural Network

When running the convolutional neural network on the MNIST dataset I got extremely high accuracy of around 98.57% on the test data and 99.86% on the training data. Based off of this, I could conclude that the model was only a tiny bit overfit since the training results are only above 1% higher than the testing accuracy

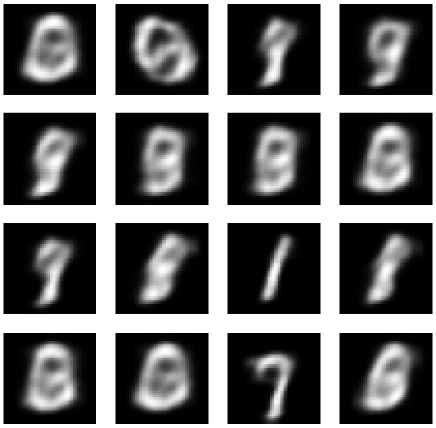
Conditional Generative Adversarial Network

When using the CGAN on the MNIST dataset I was able to see both images that looked like real numbers and those that didn't. Some of the numbers did look a little pixelated as well. It was interesting to see how it would sometimes generate half a number or even combine 2 different numbers together. A majority of the time it would generate actual numbers too.



Conditional Variational Autoencoder

The result of the conditional variational autoencoder was a little different than that of the CGAN. As can be seen in the image, instead of being pixelated images and half images, these images look more blurry than those from the CGAN. It also looks like it is printing a lot of 8s as that's the combination of all other numbers. The blurriness did get better with every epoch but the frequency of 8s is still a problem with this model.



Comparison and Conclusion

In conclusion, I could see that both models have their problems. While the CGAN was generating a lot of pixelated images, the CVAE had a lot of blurriness problems and bias towards one number. It was still impressive to see how images of numbers were able to be generated with just these inputs, even if some of them of them don’t look accurate.