

1 Summary

Previously I was able to generate some samples using GAN, but I wasn't able to do substantial improvements on the model by tuning the hyperparameters so I have started working on generating samples using VAE. I have tried reading the VAE paper [1] but couldn't make a lot of sense out of it so I re-watched the lecture videos and also looked into some YouTube videos about VAE to understand the mathematics behind it. From what I can gather, the VAE would have three parts

1. **An encoder:** which encodes the input samples into a latent space in a lower dimension
2. **A decoder:** which takes the input from the latent space and generates an output
3. **Loss function:** a way to calculate loss which will be used to run the optimizer

In this setting, the VAE network would learn the underlying distribution of the data and would be able to sample new data from that distribution using the trained decoder.

After finding out about the high level concept of the VAE, I have started implementing the VAE in `python` using `pytorch`. I feel that I can learn about the VAEs in detail by implementing it and running into some issues which require me to understand some concept in detail.

Instead of starting from scratch, I am taking the VAE implementation given in the `pytorch` examples [2] as a reference. I will be reusing the batch generator that I implemented when implementing the GAN. Although I have started implementing the VAE, I don't have any demonstrable artifacts right now.

2 Next Steps

As most of my time was spent on reading about VAEs and understanding how they work. My next step is to implement a working prototype of VAE to generate handwritten digits. I have sketched out the architecture of the VAE and I implementing the same using `pytorch`. Once I am done implementing the VAE, I would fine tune the hyperparameters of both the VAE and the GAN to generate the best samples and then I would compare the results from both to draw some conclusions.

References

- [1] <https://arxiv.org/pdf/1312.6114.pdf>
- [2] <https://github.com/pytorch/examples/tree/main/vae>