EE660 – Spring 2024 — Homework 5

Altman (Gujia) Ma

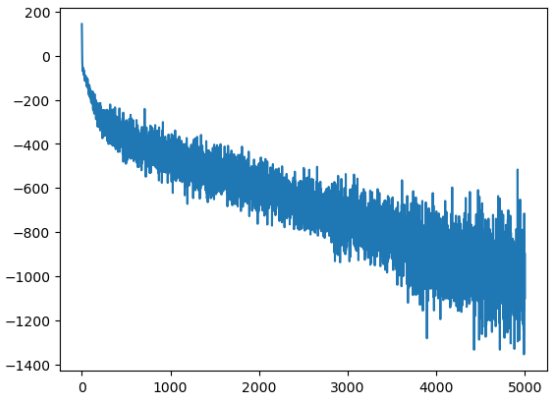
Collaborators: n/a In this Homework, I will apply DSM and EBM on all generators. Also I tried VAE and DDPM, but met problems when doing training so I gave up. However, the jupyter notebooks for the failed trails are also submitted.

1. Denosing Score Mathinc Method

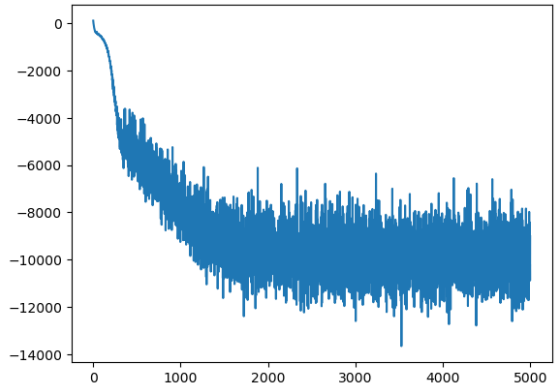
In this part, I applied DSM on all four of the generators. The training process is recorded in four Jupyter notebooks:
Chekerbooks in `denoising_score_matching.ipynb`, PinWheel in `denoising_score_matching_pin.ipynb`, Spiral in `denoising_score_matching_Spin.ipynb` and Gaussian Mixtures in `denoising_score_matching_Gaussian.ipynb`.
I applied the same superparameters for all four datasets:

```
The MLP has 2 hidden layers with 64 nerons on each layer, and 2 on output layer.
Activation: Swish Function.
Learning Rate: 0.001.
Optimizer: Adam.
Batch size: 128.
Number of Epoches: 5000,
Noise Level: 0.1.
```

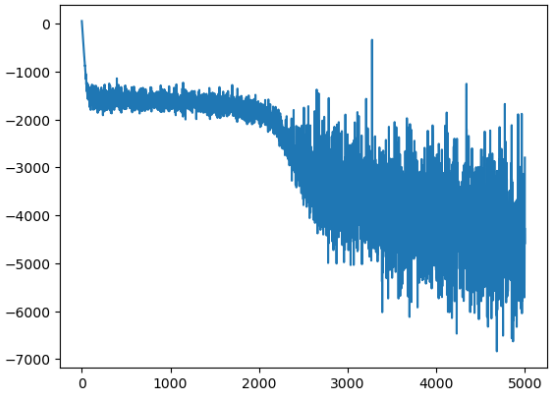
The training losses while training process in four datasets are listed below.
From the training losses functions, we can see that the training on Checker and Gaussian dataset goes very well: The loss keeps decreasing and get a low enough number. However, the cases is not as good in Spiral and PinWheel cases.



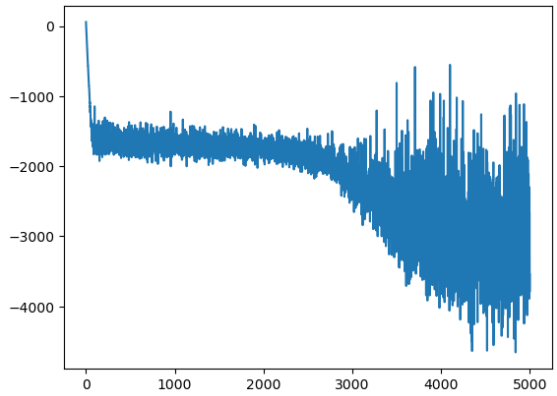
(a) DSM training Loss function on Checker data



(b) DSM training Loss function on Gaussian data



(c) DSM training Loss function on PinWheel data



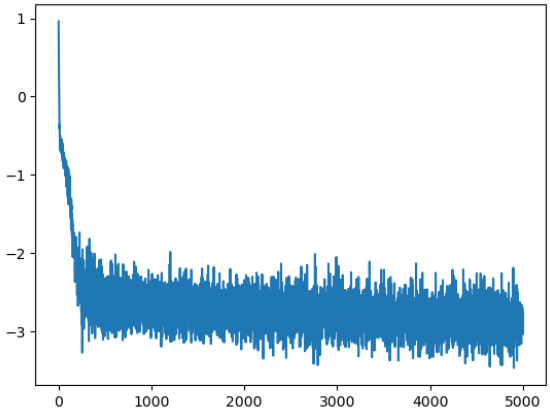
(d) DSM training Loss function on Spiral data

Figure 1: DSM training Losses

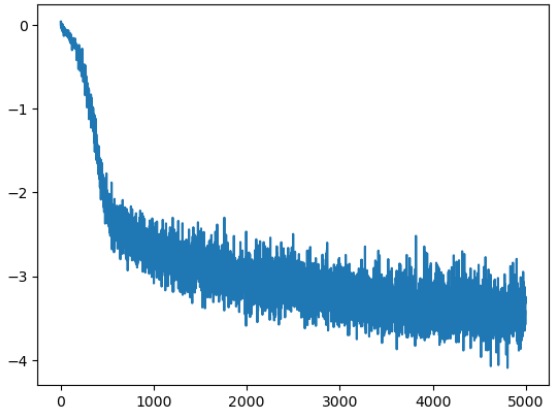
2. Energy Based Model

In this part, I applied DSM on all four of the generators. The training process is recorded in four Jupyter notebooks:
Chekerbooks in `EBM.ipynb`, PinWheel in `EBM_pin.ipynb`, Spiral in `EBM_Spin.ipynb` and Gaussian Mixtures in `EBM_Gaussian.ipynb`.
The super parameters are listed below:

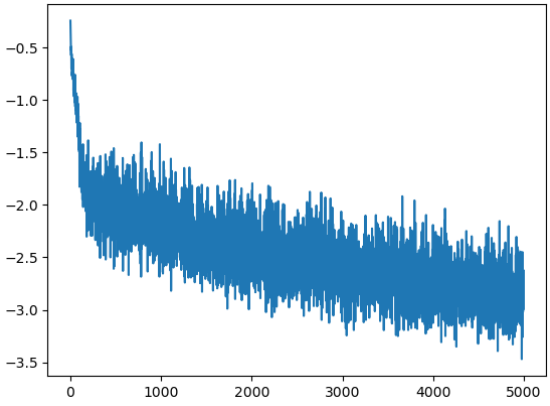
texttt The MLP has 2 hidden layers with 64 nerons on each layer, and 2 on output layer.
Activation: relu.
Learning Rate: 0.001.
Optimizer: Adam.
Batch size: 128.
Step size: 0.01.
Number of Epoches: 5000,
Noise Level: 0.1.
And the training losses are listed below.



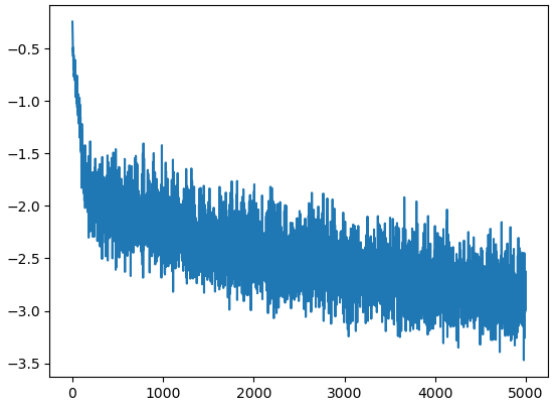
(a) BM training Loss function on Checker data



(b) EBM training Loss function on Gaussian data



(c) EBM training Loss function on PinWheel data



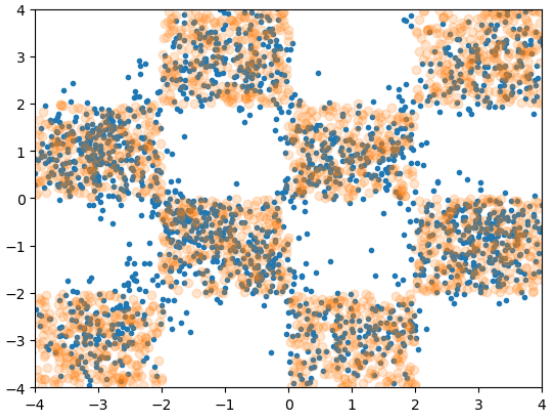
(d) EBM training Loss function on Spiral data

Figure 2: EBM training Losses

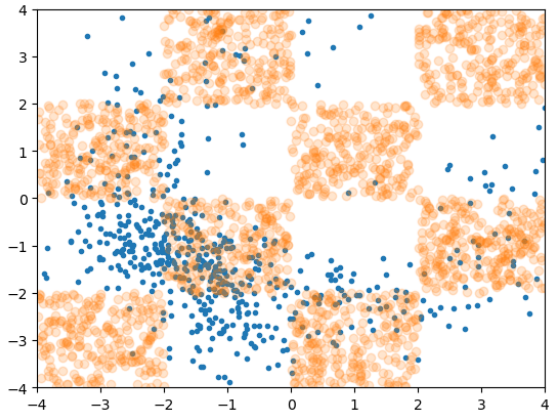
From the training losses, we can see that EBM fail to find a training path with decreasing training losses. The Training losses stopped decreasing at a very high level.

3. Qualitative comparison of model sample quality

The Comparison between Samples generated from the trained sample and from true distributions are shown below.



(a) Samples from DSM traning model with Original Data



(b) Samples from EBM traning model with Original Data

Figure 3: Qualitative Comparison between DSM and EBM Models in Checker Dataset