Bidirectional Gradient Flows for Deep Generative Learning

This repository is the demo implementation of Bidirectional Gradient Flows for Deep Generative Learning.

Requirements

To install requirements:

```
pip install -r requirements.txt
```

Training

To train BGF on toy examples, run this command:

```
python demo_toys.py --outf 'Results/demo_toys' --nEpoch 100
```

To train BGF on simulations, run this command:

```
python train_simulation.py --outf 'Results/simulation' --nEpoch 100
```

To train BGF on MNIST and FashionMNIST dataset and get the evaluation results , run this command:

```
python train.py --outf 'Results/MNIST' --nz 128 --dataset mnist --nEpoch 100
python train.py --outf 'Results/FashionMNIST' --dataset fashionmnist --nz 128 --nEpoch 100
```

Evaluation

```
To evaluate BGF on MNIST and FashionMNIST using Inception score, run:
python eval.py  --dataset mnist --nz 128 --netG ./Results/MNIST/checkpoint/KL-mnist-50-ckpt.t7 --resnet
python eval.py  --dataset fashionmnist --nz 128 --netG ./Results/FashionMNIST/checkpoint/KL-fashionmnis
```

Pre-trained Models

BGF does not adopt the pre-trained models on MNIST and FashionMNIST. However, to save the time and expenses of training, we provide the trained models that can avoid the afresh training in folder Trained

Results

Our model BGF achieves the following performance on :

[Image performance on MNIST, FashionMNIST, CIFAR10 and CelebA]

Dataset	Inception score	FID
MNIST	9.37	2.47
FashionMNIST	7.52	9.24
CIFAR10	7.63	22.34
CelebA	NA	9.63

With the afresh training, results can be obtained by running

```
python train.py --outf 'Results/MNIST' --nz 128 --dataset mnist --nEpoch 100
python train.py --outf 'Results/FashionMNIST' --dataset fashionmnist --nz 128 --nEpoch 100
```