Generative Adversarial Nets

Authors: Ian J. Goodfellow , Jean Pouget-Abadie , Mehdi Mirza, Bing Xu, David Warde-Farley,Sherjil Ozair , Aaron Courville, Yoshua Bengio.
Publisher: Conference in Neural Information Processing Systems
Year: 2014

1. **Abstract:**
"We propose a new framework for estimating generative models via an adversarial process, in which we simultaneously train two models: a generative model G that captures the data distribution, and a discriminative model D that estimates the probability that a sample came from the training data rather than G. The training procedure for G is to maximize the probability of D making a mistake. This framework corresponds to a minimax two-player game. In the space of arbitrary functions G and D, a unique solution exists, with G recovering the training data distribution and D equal to 2 1 everywhere. In the case where G and D are defined by multilayer perceptrons, the entire system can be trained with backpropagation. There is no need for any Markov chains or unrolled approximate inference networks during either training or generation of samples. Experiments demonstrate the potential of the framework through qualitative and quantitative evaluation of the generated samples."

2. **Introduction:**
   - In the scope of many methods and models of deep learning, deep generative models had been deserted because of the difficulty of using the piecewise linear units and of approximating inflexible probabilistic computations leading to the best probability evaluation.
   - In generative models, a model is created such that it is an opposing model which acquire the data distribution and the discriminative model is the model that minimizes the output of the generative models from calculating the probability that the sample is from the training data captured by G.
   - It is analogous to the minimax algorithm where G and D, in which the G tries to create the data such that increasing the probability that D will be misestimating and D will be try to maximize the probability to find the data from the G.
   - The total procedure is done by the multilayer perceptrons in both the models and these models will create an advantage of abandoning the markov models.

3. **Related Works:**
   - The models such as Deep Boltzmann machines had been successful in the deep generative models and require many number of approximations to represent the likelihood

- To replace these difficulties, generative stochastic networks, a type of generative machines, are created which would train the data through backpropagation instead of numerous approximations.
- Noise Contrast Estimation(NSE) had been another process of generative network, which discriminate data of noise distribution from the previously trained data, seems to be informal competition between the generative and discriminative models.
- Predictability minimization had said to be the other approach which had been more relevant than the former approaches. In this the hidden unit of one network is predicted based on the values of all other hidden units. The current approach differs in the training norms.
- The adversarial examples had also been sometimes thought to be generative adversarial networks, but they are not even relevant as adversarial examples are not a method of training data.

4. Content and Approach:

- As said before the models G and D plays the game with a function called as V(G,D), where G is a differentiable function with respect to θ_g. And the D is the second layer of multi-perceptrons as the function of $D(x:\theta_d)$.
- The function is represented as:
    - $\min_G \max_D V(D, G) = E_{x\sim p}$ data (x) [log D(x)] + $E_{z\sim p\,(z)}$ [log(1 − D(G(z)))].
- In the practical approach, we apply the function as iterative process. We iterate D over some predefined k steps and one step of G, otherwise it leads to overfitting of D.
- On the other hand we see while iterating , the model has to be converged with the probability normal distribution. For this purpose, initially we get a distribution of D, which is not converged and the comparison between the domain x and domain z is taken to check the probability of estimation as shown in the figure.
- The algorithm is given to define the process theoretically, in which initially for k steps a sample of z and x are taken for the both models. The minibatch with domain z is taken with the prior noise probability of $p_g(z)$.
- Then the ascending of gradient descent is taken place with an equation shown below. After iterating over for the ‘k’ times, the iteration is done on the generative model for one time with same initial batch and prior, but this try to predict the minimum of gradient descent.
- Finally an equilibrium point is reached where the curve converges and the fixed point is generated, and based on that probability the output is generated.
- The framework creates an advantage of dropouts and use of noise in the intermediary levels of the process.

5. **Results:**
   - The experiments had been created on the fundamental datasets such as MNIST, CIFAR-10 and TFD(Toronto Face Database). The dropouts were done on the discriminator networks.
   - Despite of some of it's limitations, the researchers had used the gaussians function for the validation which estimates the likelihood, had been best method so far.
   - The samples had been random draws, but not chosen by the trainers. The results were also nearby neighbouring training examples based on the probabilities.
   - The results had been compared with the log mean and the variance of the datasets. In case of MNIST datasets, Generative nets had performed significantly resulting in high mean and also with high variance comparatively with DBN, stacked CAE and deep GSN.
   - In the case of TFD, the generative nets created a low variance of the data and almost as high mean as Stacked CAE which had outperformed in the case.

6. **Conclusion and Future work:**
   - There had been considerable pros and cons of the procedure to be noted. The cons were that during the phase of training, the model D has to be coordinated with the G model.
   - The Boltzmann machine used also should updated along with the learning steps. Moreover, these can be camouflaged by the pros such as that the markov models can be avoided.
   - The adversarial networks seems to be working even with the degenerate distributions, whereas the markov chains needed at least the blurred distribution to mix up the between the nodes.
   - Various future works based on the applications and the process phases were described as a table by the authors which had to be done further such as need of inference during the training, requiring Markov chains in some applications of sampling etc.,