The new framework of Generative Adversial Nets is aimed towards replicating the probability distribution of the training data (and still the generated data is very different from original training data). The framework involves training two models simultaneously: a generative model G and a discriminative model D. The generative model G captures the probability distribution of training data and tries to replicate it. The discriminative model D outputs a probability (0-1) that the given data is from the training data rather than G. The training procedure for G is such that it propels D towards making a mistake. It is like a minimax two player game. Competition in this game drives both the models to improve themselves until generated data can't be differentiated from the original data. Back propagation is sufficient for this model and there is no need of Markov chains, thus it is computationally efficient.

We define input noise p(z) and map it to G (z, Θ), where Θ is parameter of generative model G. Similarly, we also define a second model D (x, Θ). We train D to maximise its probability of detecting training data. Simultaneously, we train G to minimise log(1-D(G(z))). In this way, two-player minimax game is played with value function V (G, D):

V (G, D) = E [log D(x)] + E [log(1-D(G(z)))] //E-expectation

Optimizing D in the inner loop is computationally expensive, so we implement k steps of minibatch stochastic gradient descent of D and one step of G (momentum was used in original research, however we can use any other gradient descent optimizer). The training objective for D can be interpreted as maximizing the log-likelihood for estimation for conditional probability P(Y = y|x), where Y indicates whether x comes from training data(with y = 1) or from generative model(with y = 0). We prove that the algorithm converges with probability distribution of generative model being equal to the probability distribution of original training data and output of D becoming ½ for all by using Kullback-Leibler divergence and Jenson-Shannon divergence.

The model was trained on a range of datasets including MNIST, the Toronto Face Database and CIFAR-10. ReLU and sigmoid activations was used for generator nets while discriminator nets used maxout activations alongwith dropout for regularization. We estimate performance of our model using Gaussian Parzen window and it proved at par(if not better) with other generative models. There are certain advantages and disadvantages of adversial nets compared to other generative frameworks. Major disadvantage is that D must be synchronized well with G to avoid “the Helvetica scenario” in which generative model produces data very similar to that of training data (which can be done by many other computationally cheap methods). Major advantage is that Markov chains are never used, only backprop is used to obtain gradients which makes the algorithm computationally efficient. Other advantage is that input training data is not directly fed to generative model rather gradients from discriminator are used to update the generative model. This gives statistical advantage and ensures that the generated data is not directly copied from training data but is very different from training data while maintaining the similar probability distribution. (despite theoretical guarantees, the practical model is not perfect. Nonetheless, the performance of the model in practice suggests that they are a reasonable model.)

\*\*\*\*\*\*SUMMARY ENDS\*\*\*\*\*\*\*

P.S. – Please try to upload the papers at least 3-4 days before discussion. For a beginner in deep learning like me, it is very difficult to understand a lot of things and have to go through reference papers and internet to understand the paper clearly (but can’t do because of the lack of time). Since it is open for all, I feel that there can be more beginners like me. I hope this will be sorted. Thanks.