Imitating Generative Adversarial Networks with Humans

Viswesh Periyasamy (vperiyasamy@wisc.edu)
Department of Computer Science, 1210 W. Dayton Street
Madison, WI 53706 USA

Abstract

This work extends Generative Adversarial Networks (GANs), a framework for building generative models to capture a data distribution, by substituting the generative model with a human. Specifically, we train a discriminative model to estimate the probability that a sample comes from the true data distribution in the same fashion as the original framework and then give a human subject access to this discriminator for querying generated inputs. The human then summarizes the query results by predicting the data distribution after a set amount of learning rounds. Experiments demonstrate that humans can converge in performance with a small set of queries and show potential for systems in which one or both components of a GAN can be replaced by humans.

Keywords: Generative modeling; Adversarial networks; Concept learning;

Introduction

Generative Modeling

Discriminative models are those which compute a conditional probability of the target Y given an observation x, and have had widespread success over the years. However, generative models (i.e. those that model the conditional probability of the observable X given a target y) had previously shown lackluster results, especially in the field of image processing and computer vision. One reason for this is that classic methods such as Maximum Likelihood Estimation become computationally infeasible in high dimensional settings. Approaches such as Deep Boltzmann machines (Salakhutdinov & Hinton, 2012) combat this by using approximations to the likelihood gradient to maximize log likelihoods, however it would be beneficial to leverage more exact solutions like backpropagation in the context of neural networks.

Generative Adversarial Networks

Goodfellow et al. (2014) present the framework of Generative Adversarial Networks (GANs) to avoid these difficulties. GANs take advantage of two components, both a discriminative and generative network, by pitting them against each other for simultaneous training of the two. Specifically, the discriminative model is trained to output the probability that an example came from the true data distribution, and the generative model uses this probability to tune a mapping from noise to the data distribution. Figure 1 depicts an example of this relationship. This framework, presented in 2014, was one of the first to show breakthrough results in generative modeling and has since been extended in a variety of ways. However, while widely successful, GANs still have several shortcomings and are subject to theoretical limitations shared by models which do not have access to an oracle (Hanneke, Kalai, Kamath, & Tzamos, 2018).

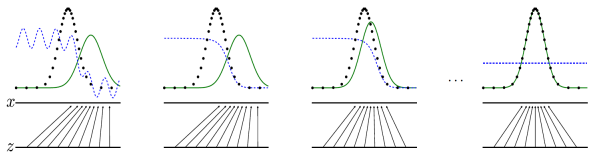

Figure 1: An example of the learning process in a GAN framework where the generated distribution (green) slowly conforms to the true distribution (black) and the discriminator (blue) is unable to differentiate the two. Figure adopted from Goodfellow et al. (2014)

Human Generative Models

One area that remains relatively unexplored is how a GAN framework, or any generative modeling framework, can be modified to leverage humans. Human concept learning is advantageous over machines in a variety of settings. Although the underlying mechanisms are not yet completely understood, humans can learn very rich and flexible representations as opposed to a feature space or set of rules that traditional machine learning models tend to learn. Furthermore, humans have been shown to generalize and learn well from a remarkably small number of examples, whereas machine learning algorithms often need a large data set to produce similar results and in many instances cannot achieve the same performance (Lake, 2014). For these reasons, utilizing humans is an attractive solution to improving the success of machine learning frameworks. This work aims to investigate two main ideas: how the GAN framework can be extended by replacing the generative model with a human, and how the mind navigates concept learning when restricted to a generative process.

Related Work

Previous literature exists which utilizes humans in generative modeling. One such framework, presented by Peterson, Aghi, Suchow, Ku, and Griffiths (2018), captures human category representations within a GAN framework by using a technique known as Markov Chain Monte Carlo with People (MCMCP) (Sanborn & Griffiths, 2007). MCMCP is a procedure which use humans as a valid acceptance function in a Metropolis-Hastings flavor of MCMC. Specifically, the human is presented with images from two categories that are perturbed with noise, and the selected image is fed to a discriminative model whose output is connected to a generative model which sets up the next set of images. Thus, the system is able to approximate high-dimensional deep feature spaces captured by the MCMCP process. The paper also presents results from human distributions which perform competitively

with cutting-edge generative models.

Another approach by Hwang, Azernikov, Efros, and Yu (2018) uses the GAN architecture in the classic way but relies on human generated designs to train on. In the dental industry, designing crowns takes years of human training to build accurate and functional crowns which is extremely expensive. Instead, they build a GAN system that trains on human generated designs to capture the intricacies of an experienced technician.

While both of these approaches incorporate humans in some fashion to a GAN framework, the humans do not partake in the actual learning of the models. Consequently, the models are unable to directly take advantage of the human mind to improve the generated output; this is the main problem this work aims to address.

Approach

Minimax Optimization

As mentioned above, a GAN framework is comprised of two models: a generator G that captures the data distribution by mapping input noise to the observable feature space X, and a discriminator D which estimates the probability that a sample came from the true distribution. These two models compete objectively in what is known in game theory as **Minimax**. Minimax is a decision rule in which the objective is to *minimize* the possible loss for a worst case (i.e. *maximum* loss) scenario. This can be summarized by the following value function:

$$\bar{v_i} = \min_{a_{-i}} \max_{a_i} v_i(a_i, a_{-i}) \tag{1}$$

where a_i represents the action by player i and v_i is the player's corresponding value function. Intuitively, this can be thought of as a player maximizing their value, given the minimum possible value their opponent could force.

Adversarial Networks Framed as Minimax

In the context of GANs, we first define the generator's distribution over the data x as p_g. Furthermore, we define a prior on the input noise z as $p_z(z)$. $G(z;\theta_g)$ represents the mapping from noise to the observable feature space with some parameters θ_g. Conversely, $D(x;\theta_d)$ symbolizes the probability that a sample x came from the data rather than p_g. The Minimax problem can then be reframed as follows:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{(x)}}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1-D(G(z)))] \tag{2}$$

In this scenario, we are training the models to maximize the probability of assigning the correct label to the input ($D(x)$) while simultaneously minimizing $1-D(G(z))$.

Rectangular Concepts as Data Distributions

In order to build a cohesive experiment with human and machine models, distributions in the form of rectangle regions were chosen. This choice was made for a number of reasons; first and foremost, translating images between humans and machines presents several logistical difficulties. A typical GAN framework set up for image generation will require thousands or millions of iterations and examples in order to get good performance, which is not feasible for a human subject to participate in. Additionally, it would be extremely difficult for a human to transfer the output probability of the discriminator into meaningful tuning of their concept distribution for an image.

On the other hand, using a one-dimensional target distribution would be too simple for both parties of the system. For example, having some target range that the discriminator learns to accept would be fairly easy, and a human could systematically query numbers to determine the range - this would not lend itself to meaningful results.

A rectangular distribution, specifically a square region of a grid where points inside are labeled as true and points falling outside are labeled as false, provides enough complexity for both systems to engage in active learning. Rectangle are commonly used to illustrate models of concept learning, as shown by Mitchell (2017). Furthermore, although they are 4-dimensional (two points representing the top left and bottom right corners), they can be translated to a visual representation for humans.

Experiment Details

The discriminative model is a neural network which takes in a set of four points in two-dimensional Euclidean space in the form of a flattened vector of eight values, which is then fed into two densely connected layers of 16 nodes each. Each layer uses a Rectified Linear Unit (ReLU) activation function, and the final layer feeds into a Softmax activation to output a probability representing whether the input came from the true distribution or not. Finally, binary cross entropy is used as a loss function and the Adaptive Moment Estimation (ADAM) algorithm is used for optimization.

The experiment is laid out as follows:

1. A target rectangle is randomly generated in a 100x100 pixel grid

2. The discriminator alternates between training on "good" points (all four points lie within the boundary) and random sets of "bad" points (anywhere on the grid)

3. The human repeatedly queries the discriminator by selecting sets of four points and receiving a probability evaluated on that input

4. The human finally draws their interpretation of the target rectangle (generative distribution) and compares it to the true rectangle (target distribution)

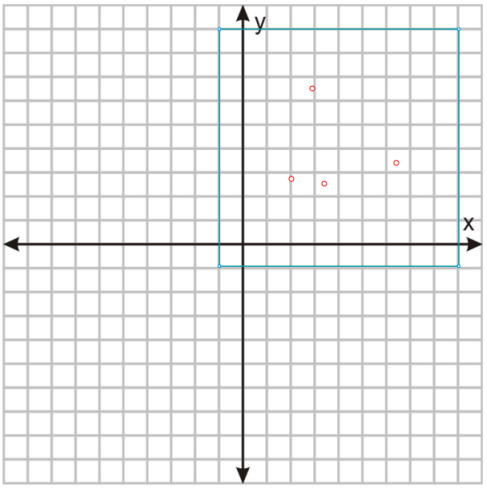

Figure 2: The coordinate grid presented to human subjects for querying and guessing the target distribution.

Since a positive example is defined as all points lying within the target distribution, alternating between good and bad examples was necessary for the discriminator training because randomly generated examples almost never contained all points within the boundary. Additionally, in order to simplify the experiment and control for variability, target rectangles were restricted to 50x50 pixel squares on the grid. Figure 2 shows the grid and point interface used in the experiments.

Results

Experiments were conducted by varying the number of iterations allowed before forcing the human subject to guess the target distribution. Since the target concept can be defined by two points, error can be defined by a residual vector between the coordinates making up the target concept. Error was measured in three ways: Euclidean distance (3); Root Mean Square error (4); Relative normalized error (5).

If $\mathbf{a}$ is our vector for the actual target concept and $\mathbf{g}$ is our guessed concept, the residual vector is $\mathbf{r} = \mathbf{a} - \mathbf{g}$. Then the error measures are given by the following equations:

$$E_{euc} = \|\mathbf{r}\|_2 = \sqrt{\sum_{i=1}^{n} r_i^2} \tag{3}$$

$$E_{RMS} = \sqrt{\frac{\sum_{i=1}^{n} r_i^2}{n}} \tag{4}$$

$$E_{rel} = \frac{\|\mathbf{r}\|_2}{\|\mathbf{a}\|_2} = \frac{\sqrt{\sum_{i=1}^{n} r_i^2}}{\sqrt{\sum_{i=1}^{n} a_i^2}} \tag{5}$$

Figures 3 and 4 shows the error measures of target concepts guessed by the human subject as the number of iterations per experiment grows. Experiments were conducted by increasing the number of iterations from 5 to 100, in intervals of 5, for 20 total experiments. Additionally, Figure 5 shows a plot of the discriminator output probabilities over the course of one experiment with 100 queries.

Lastly, to investigate the measured concept error as a function of iterations per experiment, Figure 6 shows the relative normalized error over 10 experiments, each with an increasing number of iterations before guessing the distribution.

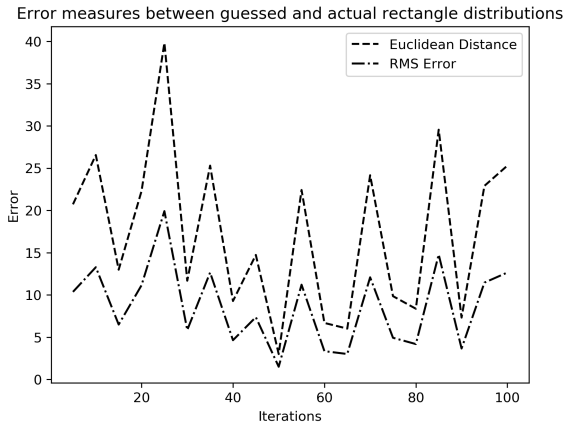

Figure 3: Euclidean distance and RMS Error of generated target concept varied over number of iterations.

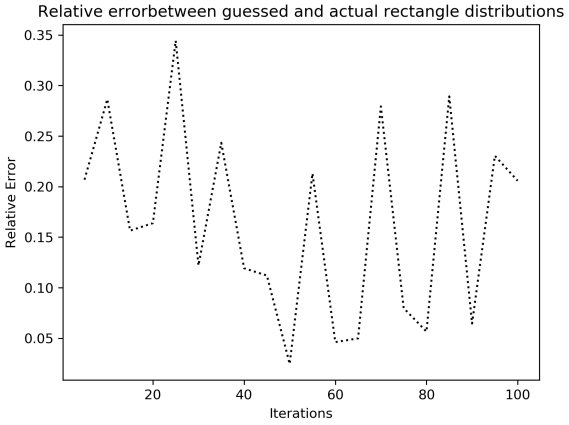

Figure 4: Normalized error of generated target concept varied over number of iterations.

Discussion

Error Measures Under Experiment Conditions

As Figures 3 and 4 show, error did not show a consistent trend of increasing or decreasing as the number of iterations grew. In fact, the error seems to spike back up after every 10 iterations or so, while crashing down in between. The relative normalized error also shows this trend, meaning it's not just exaggerated by large values for the concept boundaries.

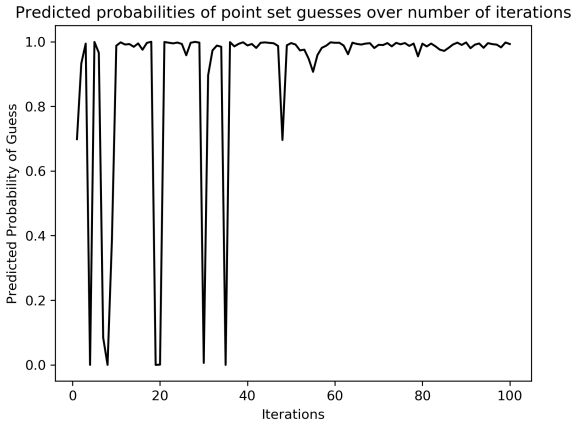

Figure 5: Discriminator output probabilities over 100 queries in one experiment.

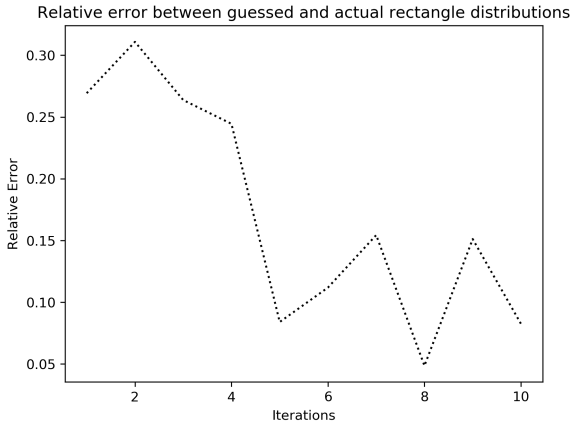

Figure 6: Normalized error of generated target concept varied over number of iterations from 10 experiments.

One explanation for this might be that past a certain number of iterations, it might not be beneficial to continue querying the discriminator and may actually harm the learning process. However, since the error also shows very low values on alternating experiments, another explanation might be that inherently humans make better predictions after an even number of queries. Consequently, adding an odd number of queries after an even number (e.g. 55 or 85) may throw the human off from their internalized concept. This study makes no theoretical claims to back up this suspicion, although many studies in the literature exist to support that humans prefer even or whole chunks.

One anecdotal note from the participant in this study is that beyond a certain number of iterations, previous query results were somewhat forgotten and the subject found themselves re-querying the same spots repeatedly to remember how the discriminator responded. This would provide further support that past a certain number of iterations, more queries are not beneficial and might even make results worse with plethora of data to process. It might be an overload of information for the human mind, but one way to combat this would be to give the subject access to all previous queries and results - still, this study acknowledges that it would be difficult for a human to process hundreds of query results when making a decision.

In order to investigate further, experiments were also run by varying the number of iterations per experiment from 1 to 10 to get a fine-grained analysis of how the number of iterations played a role when feedback was sparse. Figure 6 depicts a general downward (albeit noisy) error trajectory which settles close to 0.10 compared to the initial error around 0.30. This tells us that before 5 iterations the performance is quite unpredictable, however after 5 iterations the human subject has had enough queries to solidify the concept in their mind enough to restrict the error to some range. Intuitively this makes sense, as after 4 queries, each of the four quadrants can generally be ruled out (target concepts took up 25% of the the grid) and then further queries allow for fine tuning of the distribution.

With this all being said, this still shows that humans are able to converge to a learned target distribution relatively quickly (under 10 examples) which supports the hypothesis that humans can generalize effectively from a few examples.

Human Learning Trajectory

Another aspect of the generative process that was probed was the actual learning process of the human. Examining Figure 5 shows the plot of query probabilities returned by the discriminator for each iteration over one 100-iteration experiment. We see that in the first 50 iterations, there are about 5 valleys which represent guesses that are extremely low, and then the probabilities taper upward and converge very close to one for the rest of the experiment.

The first insight to gain from this is that it enforces the theory that it is more informative to learn from negative examples first than to continually receive positive examples. As shown by the trajectory, the initial spikes help rule out certain areas of the grid from being part of the target concept. Then, the probability remains increasingly closer to 1 until it finally converges.

We can also conclude that the discriminative model has either learned the concept extremely well from its training, or is just so sensitive that it outputs almost binary values. If the latter is true, it might be easier for the human subject to just receive a binary output and make their decision based on that. As the experiment currently stands, a probability value might not give enough intuition to the human subject in how to tune their current internalized concept - the plot supports the idea that the subject could have randomly placed points, memorized which random sets returned high or low values, and then guessed based on that. In fact, a continuous probability value might even confuse the subject more by offering "false" guidance when one or two points fell out of the boundaries. However, the subject noted intermediate probabilities in the range of 0.25 to 0.75 were *perceived* to be helpful in tuning their distribution.

Future Work

This study lays the foundation for constructing a GAN framework using human subjects as generative models. However, many questions remain to be explored in this context.

In the domain of rectangle concepts, one question is how many query points are adequate for both a generative and discriminative model to internalize the target concepts. This study chooses four arbitrarily, since although rectangles are made up of four corners, the subject never queried by constructing four-point rectangles. It remains to be seen whether increasing or decreasing the number of points would help improve performance.

Second, further studies are necessary to understand how humans can interpret a scalar probability in order to fine tune their next query. Experiments where the value is thresholded to a binary output would help bring more conclusive results in this aspect.

Lastly, this study did not compare results to a traditional GAN framework with machines in both components. To extend this study, future work could build a generative model which would be able to "guess" a target concept instead of just learning what small point set (possibly bunched around one point) would land the highest probability. One way to do this would be to enforce some sparsity constraint around the points so they are spaced out and draw a rectangle around that. A comparative analysis could then conclude whether humans can outperform machines with a much smaller dataset, or that humans can outperform machines until a certain number of examples have been queried.

Appendix

All code and materials can be found at the following repository:

`https://github.com/vperiyasamy/human-gan`

References

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). Generative adversarial nets. *NIPS'14 Proceedings of the 27th International Conference on Neural Information Processing Systems*, *27*, 26722680. Retrieved from `https://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf`

Hanneke, S., Kalai, A., Kamath, G., & Tzamos, C. (2018). Actively avoiding nonsense in generative models. *CoRR*, *abs/1802.07229*. Retrieved from `http://arxiv.org/abs/1802.07229`

Hwang, J., Azernikov, S., Efros, A. A., & Yu, S. X. (2018). Learning beyond human expertise with generative models for dental restorations. *CoRR*, *abs/1804.00064*. Retrieved from `http://arxiv.org/abs/1804.00064`

Lake, B. M. (2014). *Towards more human-like concept learning in machines: compositionality, causality, and learning-to-learn*. Unpublished doctoral dissertation.

Mitchell, T. M. (2017). *Machine learning*. McGraw Hill.

Peterson, J., Aghi, K., Suchow, J., Ku, A., & Griffiths, T. (2018, Jan). Sampling from object and scene representations using deep feature spaces. *Journal of Vision*, *18*(10), 403. doi: 10.1167/18.10.403

Salakhutdinov, R., & Hinton, G. (2012). An efficient learning procedure for deep boltzmann machines. *Neural Computation*, *24*(8), 19672006. doi: 10.1162/neco_a_00311

Sanborn, A. N., & Griffiths, T. L. (2007). Markov chain monte carlo with people. *NIPS 2007 Proceedings*. Retrieved from `https://papers.nips.cc/paper/3214-markov-chain-monte-carlo-with-people.pdf`