SOCIETY OF ACTUARIES

Article from
Predictive Analytics and Futurism
August 2019
Issue 21

Semi-Supervised Learning With Generative Adversarial Networks

By Jeff Heaton

In the world of machine learning, supervised and unsupervised are the two premier methodologies usually discussed. Most problems, and the models used to deal with them, are either classified as supervised or unsupervised. However, there are other types of models beyond supervised and unsupervised. Models that support semi-supervised and reinforcement learning are two model types that have lately been gaining considerable traction. In this two-part article series, we will look at semi-supervised learning. This article will begin by introducing semi-supervised learning and the generative adversarial network (GAN). The GAN, which is usually shown in conjunction with image rendering, will be demonstrated to have insurance industry applications. The second article will provide a more technical implementation of a semi-supervised GAN, using Keras and TensorFlow, for health care data.

AN INTRODUCTION TO THE GENERATIVE ADVERSARIAL NETWORKS

GANs have received a great deal of publicity lately for their ability to generate realistic human faces. The Python package called StyleGAN, available from nVidia, makes for a great introduction to face generation with GANs. Setting up an environment to run StyleGAN can be challenging. It requires a current nVidia GPU, along with a Linux operating system. I suggest using the free Google CoLab (free) for GAN experimentation. Google also provides a free K80 GPU, which is more than sufficient to begin experimenting with GANs. I provide a video that contains detailed instructions for setting up CoLab with GANs.

Figure 1 shows a face that I generated using nVidia StyleGAN. It looks quite realistic. At first glance anyway. If you know what to look for, you can probably spot a fake human face. Does anything about Figure 1 look out of the ordinary?

Any sort of accessory is usually one of the first giveaways. In this case, the earrings are a dead giveaway. Ears are also often nonsymmetrical. Often a GAN-generated face will have two completely different earrings and often two very differently shaped ears. The background is usually a giveaway as well. The background of a GAN-generated image is typically surreal looking. It looks natural, but you are never quite sure what you are looking at. Linear projections that begin on one side of the face often do not align to what is behind the other side. For this image, the background is not that surreal, but I am also not entirely sure what I would classify the background as either. There are other common giveaways as well, particularly if the image is high enough resolution.

Figure 1
Real or Fake?

A GAN accomplishes this generative capability by learning the underlying distributions of source data. The source data does not need to be images. GANs can learn from nearly any sort of data. Consider a GAN that might be trained on medical data. The GAN would quickly learn the distributions of the input data. Input columns such as gender, blood pressure, height, weight, age and even various lab tests could all have their distributions learned by the GAN. With these distributions learned, the GAN could now learn to generate fake medical records just like it creates fake images. If portraits were available together with medical data, the GAN could theoretically learn to generate the medical records behind the impostor faces that the GAN is generating.

However, the GAN does not just learn the individual distributions of the source data. The GAN also learns the conditional distributions and other correlations among the input features. This is why the faces appear so realistic. The GAN matches human face characteristics in ways that we are used to perceiving in real humans. The GAN would perform similarly when attempting to generate impostor medical records. Correlations among age and blood pressure, for example, would be considered. This would add to the realism of fake medical records.

How Do GANs Work?

Though most coverage of GANs in the media has been for image recognition, there is nothing about a GAN that directly ties it to image processing. Likewise, though GANs are most often demonstrated with deep learning, there is nothing that ties a GAN to neural networks as the underlying model. When a GAN is used for computer image processing, it is most often used with a convolutional neural network (CNN). This makes a great deal of sense, because CNNs are among the most state-of-the-art models for image processing.

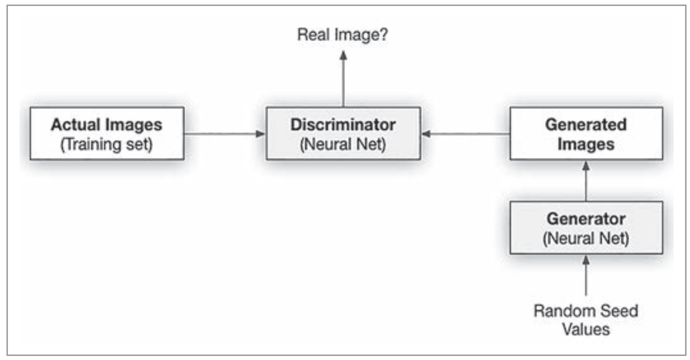
The name generative adversarial network describes the algorithm well. Training of a GAN is essentially an arms race between two neural networks. The GAN algorithm is a zero-sum noncooperative game.

- **Discriminator neural network.** Given an input image, predict the probability that the image is real.
- **Generator neural network.** Given an input random vector, generate a realistic-looking image that will fool the discriminator.

The beauty of this approach is that it is unsupervised. Source images are needed, but they do not need to be labeled in any way. No human intervention is needed. If a human were needed

to rate each of the generated images, it would take a considerable amount of time to train such a network. Rather, the two neural networks are learning from each other. The discriminator learns to discriminate better by evaluating what the generator produces. Likewise, the generator also ups its game trying to fool the discriminator. The process is summarized in Figure 2.

Figure 2
Generative Adversarial Network (GAN)



Once training is complete, the discriminator is usually discarded, as you now have a generator that can generate data that is limited only by the number of random seed vectors that you are willing to generate.

GANs FOR SEMI-SUPERVISED LEARNING

When I first saw GAN technology, I was captivated by the amazingly realistic-looking images that they produced. However, I did not see a great deal of practical use to my job as a data scientist working for a life reinsurer. Ultimately, once I saw their application to semi-supervised learning, I began to evaluate this new technology.

Semi-supervised is useful when your data are only partially labeled. Maybe you have a large number of medical records but underwriting decisions or mortality experience on a small number. For mortality experience, semi-supervised would be cases where you do not know how long the individual ultimately lived. Instances where some individuals are still alive and only a handful have died would not be considered semi-supervised, as you do not have labels for all the data. Also, such situations might better lend themselves to survivor modeling.

Semi-supervised learning is very biologically plausible. Children see a wide number of human beings in their early childhood. Very young children see these samples of humans well before they know ages and genders. This foundational knowledge is very useful when they begin to receive labels and learn to discern ages and genders with increasing accuracy. This is somewhat how GANs are applied to semi-supervised learning.

This article presented GANs and showed how they can be extended to implement semi-supervised learning.

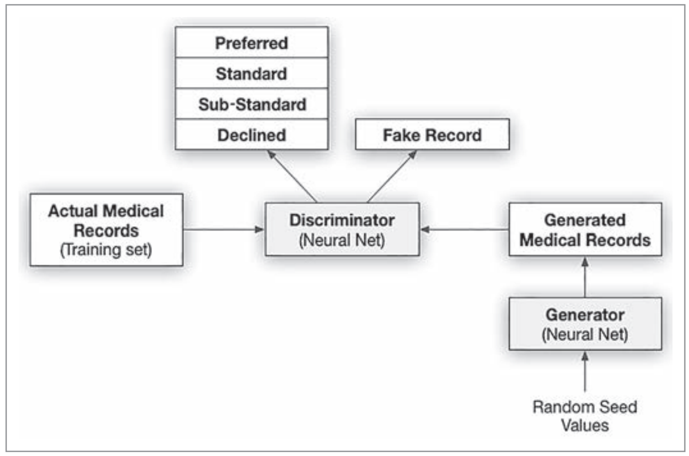
Traditionally, GANs produce a useful generator, leaving a discriminator that is discarded. Semi-supervised GANs flip this entirely. The generator is used only to train the best possible discriminator that will be the ultimate model artifact from the training exercise. The discriminator for a semi-supervised GAN will not simply classify between real or fake. The semi-supervised GAN will classify into a set number of classes plus an additional class for fake records. This is used to train the generator to produce more likely fake data. However, in the process, the discriminator is getting very good at not only classifying but also detecting fake records. Once training is done, the fake record classification can be used as an anomaly detection mechanism should anything be classified as fake in production. This process, for an insurance underwriting system, is shown in Figure 3.

Figure 1 shows how to perform semi-supervised training for classification. For regression, the neural network would simply have two outputs. The first output neuron would be the regression prediction. The second output neuron would present the probability that the input data was fake.

NEXT STEPS

This article presented GANs and showed how they can be extended to implement semi-supervised learning, which can be extremely beneficial when your dataset is only partially labeled.

Figure 3
Semi-Supervised Learning



The next article in this two-part series will show a technical implementation using Keras. This will show how to perform predictions of partially labeled medical data. ■

Jeff Heaton, Ph.D., is vice president and data scientist at RGA Reinsurance Company Inc. He can be reached at *jheaton@rgare.com*.

REFERENCES

1 Using Google CoLab for GANS. *GitHub.com*, *https://github.com/jeffheaton/present/blob/master/youtube/style_gan.ipynb* (accessed June 10, 2019).

2 nVidia StyleGAN. *GitHub.com*, *https://github.com/NVlabs/stylegan* (accessed June 10, 2019).