Getting started with GANs

1 Motivation

There are several reasons why GANs have been a popular area of research in machine learning. One reason is that they can be used to generate new, synthetic data that is similar to a training dataset. This can be useful in a variety of applications, such as generating realistic images, audio, or text, or creating new data for training other machine learning models.

Another reason to study GANs is that they can be used to learn complex, high-dimensional distributions, such as the distribution of images in a dataset. This can be difficult for other types of models, especially when the data is noisy or unstructured. By training a GAN to generate synthetic data, researchers can learn more about the underlying distribution of the data and potentially use that knowledge to improve the performance of other machine learning tasks.

Finally, GANs have also been used for a variety of other tasks, such as data augmentation, domain adaptation, and representation learning. They have the potential to be useful in many different fields and applications, and as a result, there has been a lot of interest in understanding how they work and how to improve their performance.

2 Introduction

In order to understand quantum GANs, it is helpful to have a basic understanding of classical GANs and quantum computing.

Classical GANs, or Generative Adversarial Networks, are a type of machine learning model designed for generating new, synthetic data that is similar to a training dataset and they have been applied to a variety of tasks including image generation, audio synthesis, and text generation. They consist of two parts: a **generator network***, which produces synthetic data***, and a **discriminator network**, which tries to distinguish the synthetic data from real data. The generator and discriminator are trained together in a zero-sum game, with the generator trying to produce data that can fool the discriminator, and the discriminator trying to accurately identify synthetic data. The goal of the GAN is to find a balance between the generator and discriminator, such that the generator is able to produce high-quality synthetic data and the discriminator is able to accurately distinguish synthetic data from real data.

A quantum GAN is a variant of the classical GAN that uses quantum computers and quantum algorithms to train the generator and discriminator networks. The goal is to leverage the power of quantum computers to improve the performance of the GAN, for example by allowing it to generate higher-quality synthetic data or to learn more efficiently.

*The generator network is responsible for generating synthetic data. It takes as input a noise vector and produces a synthetic output that is intended to be similar to the real data in the training dataset. The generator is trained to "fool" the discriminator network by producing synthetic data that is difficult to distinguish from real data.

***Synthetic data is not real data, but it is intended to be similar to the real data in some sense. For example, if the real data is a collection of images, the synthetic data generated by the GAN should also be images that are similar to the real images in some way, such as having the same general structure or style.

**The discriminator network is responsible for distinguishing synthetic data from real data. It takes as input either real data or synthetic data and outputs a probability that the data is real. The discriminator is trained to accurately identify synthetic data, so it must be able to distinguish the synthetic data produced by the generator from real data.

Why there is a need to generate synthetic data?
There are several reasons why it can be useful to generate synthetic data. Some of the main reasons include:

- Increasing the size of the dataset: Synthetic data can be used to augment a small dataset, which can be beneficial for training machine learning models that require a large amount of data.
- Reducing the cost of data collection: Generating synthetic data can be much cheaper and faster than collecting real data, especially when the real data is difficult or expensive to obtain.
- Protecting privacy: Synthetic data can be used to preserve privacy when working with sensitive real data, as the synthetic data does not contain any personal or confidential information.
- Testing and evaluating machine learning models: Synthetic data can be used to test and evaluate machine learning models, for example to measure the generalization error of a model or to compare the performance of different models.

Overall, generating synthetic data can be a useful tool in a variety of situations where real data is not available, is difficult to obtain, or needs to be protected.

3 Underlying Mathematics

The mathematics underlying classical GANs involves a combination of linear algebra, probability theory, and optimization.

The training process for a GAN involves optimizing the parameters of the generator and discriminator networks using stochastic gradient descent (SGD) or a related optimization algorithm. The objective function for the GAN is the sum of the generator and discriminator loss functions, which are defined based on the probabilities output by the discriminator and the structure of the data.

In terms of the mathematics, the generator and discriminator networks are typically implemented as feedforward neural networks, which involve matrix multiplications and element-wise nonlinearities such as the sigmoid or ReLU function. The training process involves calculating gradients of the loss function with respect to the network parameters and using these gradients to update the parameters using SGD or a similar optimization algorithm.

The training process for a quantum GAN involves optimizing the parameters of the generator and discriminator circuits using a quantum algorithm or classical optimization algorithm. The objective function for the Q-GAN is the sum of the generator and discriminator loss functions, which are defined based on the probabilities output by the discriminator and the structure of the data.

In terms of the mathematics, the generator and discriminator circuits are typically implemented as quantum circuits, which consist of a series of quantum gates that perform quantum-mechanical transformations on quantum bits (qubits). The training process may involve calculating gradients of the loss function with respect to the circuit parameters and using these gradients to update the parameters using a quantum algorithm or classical optimization algorithm.

Overall, the mathematics underlying quantum GANs involves a combination of quantum computing, machine learning, and optimization, which are all central to the field of quantum machine learning.

4 Comparing GANs to other Generative models

GANs were designed to avoid many disadvantages associated with other generative models:

- Generative Adversarial Networks (GANs) have the advantage over other generative models, such as FVBNs, in terms of the way they generate samples. In GANs, the process of generating samples can be done in parallel, meaning that the time it takes to generate the samples is not directly proportional to the dimensionality of the data. This is a significant advantage compared to FVBNs, where the runtime is proportional to the dimensionality of the data, making the process much slower for high-dimensional data. This means that GANs can more efficiently and quickly generate a large number of samples, making them a powerful tool for various applications.

- GANs have a lot of freedom in the design of the generator function, which can be any function that maps from a latent space to the data space. This is a major advantage compared to other generative models like Boltzmann machines, where the generator has to meet specific requirements to allow for efficient sampling, and nonlinear ICA, where the generator has to be invertible and have the same number of dimensions as the data. The flexibility of the generator design in GANs allows for greater experimentation and a greater chance of finding a suitable generator for a given data set.
- No Markov chains are needed in GANs. This is an advantage relative to Boltzmann machines and GSNs. Markov chains are a type of mathematical model for random processes that have a memory of their past state, meaning that the next state depends only on the current state. The absence of Markov chains in GANs means that the model does not rely on this type of memory or dependency, providing a different approach to generating samples. This difference can potentially lead to advantages in terms of computational efficiency or sample quality compared to models that do use Markov chains.
- The GAN framework does not require the use of a variational bound, meaning it does not rely on approximating a target distribution in order to generate samples. This sets GANs apart from Variational Autoencoders (VAEs) which do require a variational bound. Additionally, there are already established model families within the GAN framework that are known to be universal approximators, meaning they can approximate any target distribution. This means that GANs are already known to be asymptotically consistent. On the other hand, while some VAEs are believed to be asymptotically consistent, this has not yet been proven.
- GANs are subjectively regarded as producing better samples than other methods.

5 References

Classical GANs

1. Generative adversarial nets (2014), I. Goodfellow et al.
   For an introduction to classical GANs. This paper provides a high-level overview of the GAN model and describes how it can be used to generate synthetic data.
2. NIPS 2016 Tutorial:Generative Adversarial Networks by Ian Goodfellow
   To learn more about the theory and implementation of classical GANs
3. Wasserstein GAN (2017), M. Arjovsky et al.
4. Improved techniques for training GANs (2016), T. Salimans et al.
5. Unsupervised Representation Learning With Deep Convolutional Generative Adversarial Networks (2015), A. Radford et al.

Quantum GANs

1. Quantum Generative Adversarial Networks by Pierre-Luc Dallaire-Demers, Nathan Killoran.
2. He-Liang Huang et al. Experimental Quantum Generative Adversarial Networks for Image Generation (2020)
3. Quantum GANs-tutorial (pennylane)
4. Anomaly detection with Variational Quantum Generative Adversarial Networks by Daniel Herr, Benjamin Obert, and Matthias Rosenkranz. This paper presents a specific type of VQGAN that uses a variational quantum circuit to generate samples.