1. Generative Adversarial Networks

* Yann LeCun, described GANs as the most interesting idea in ML in the last 10 years
* LeCun – LeNet architecture of CNN
* Ian Goodfellow – inventor of GANs

1. Theory of GANs

* A new and interesting way of combining things we already know how to build
* Common pattern in DL
* Neural networks – logistic regressions chained together
* CNNs and RNNs are just neural networks with shared weights

1. What are GANs for?

* Main use case is to generate data (specifically, images)
* GANs are really good at this compared to previous models

1. How do GANs work?

* A system of 2 neural networks: generator and discriminator

A close-up of a person's face

Description automatically generated

* Intuition: Counterfeiter vs shop owner

1. The ingredients of a deep learning model

* We need a model (the neural network, or in GANs case, 2 of them)
* Objective / loss function to minimize
* Do gradient descent

1. The loss functions

* 2 loss functions for the generator and discriminator
* The discriminator must classify real and fake images -> binary classification
* Binary crossentropy
* Generator loss: Freeze the discriminator layers so only the generator is trained

A diagram of a frozen storage

Description automatically generated

* Loss function is still the binary crossentropy
* Except we switch the labels
* Output probability of real given fake image (target is always 1)

1. More details

* Input into the generator network?

A diagram of a diagram

Description automatically generated

* Just noise!
* Z ~ N(0, 1) – a random vector of size 100 (for instance)
* These vectors come from the latent space

1. Visualizing the latent space

A close-up of a graph

Description automatically generated

* Generator is like the opposite of a feature transformer / embedding
* Image -> vector vs. vector -> image

1. Pseudocode

A screenshot of a computer code

Description automatically generated

A white background with black text

Description automatically generated

1. Loss goes down, accuracy goes up?

* Makes sense if we are doing classification or regression >< not what we are doing
* The discriminator and generator are constantly trying to one-up each other

1. Summary

* GAN is a system of 2 neural networks: generator and discriminator
* Learn in tandem
* Discriminator loss: binary crossentropy
* Generator loss: binary crossentropy on the combined model
* Freeze the discriminator in the combined model
* Flip the labels (images from generator are labeled as real)
* Generators input is noise, sampled from latent space
* Latent space is like the opposite of feature transformation/embedding
* Generator: vector/feature/embedding -> image
* Loss per iteration shouldn’t necessarily decrease over time