Image credit: Tega Brain, Julian Oliver, and Bengt Sjölén, *Asunder* (2019-20)

Carlos Castellanos | School of Interactive Games & Media | Rochester Institute of Technology

Current Topics in Interactive Development - IGME 480

RIT

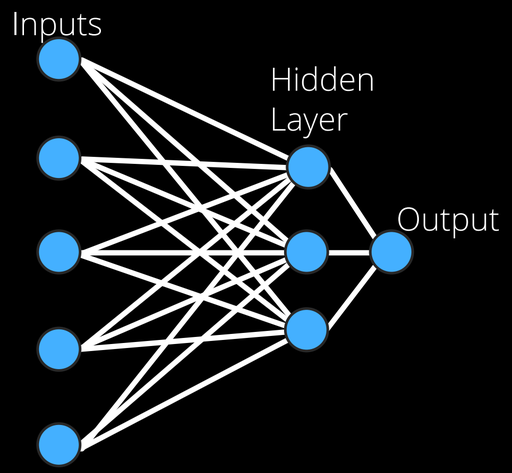
Outline of Today's Lecture

- Introduction to Deep Learning
- Convolutional Neural Networks
- Generative Models

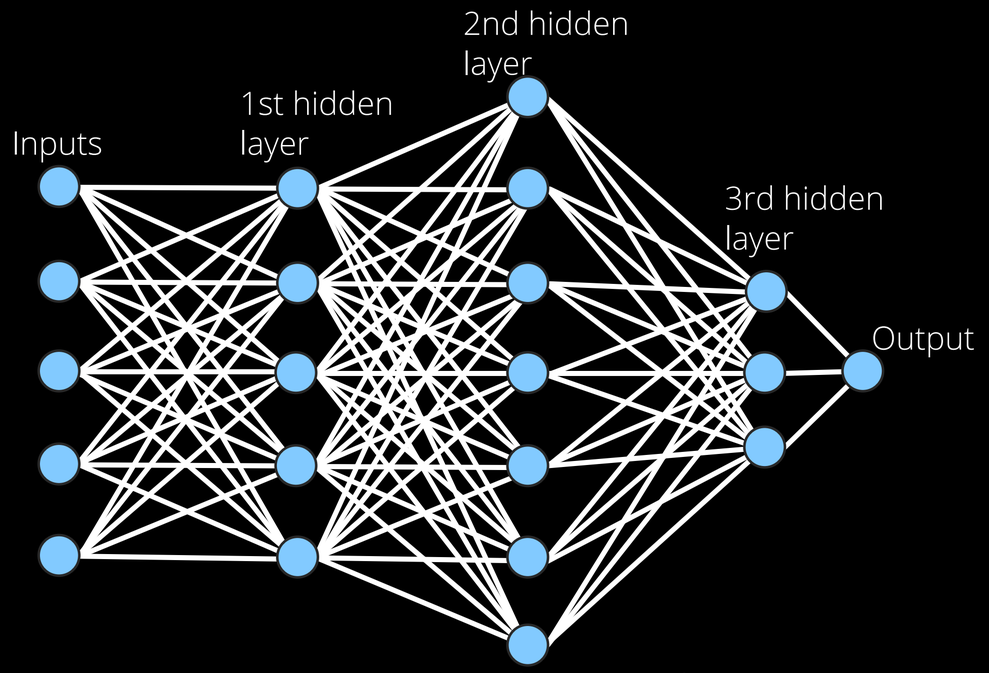
Machine Learning

Introduction to Deep Learning

- The “Deep” in Deep Learning is basically referring to neural networks with many more layers and nodes than “traditional” NNs.



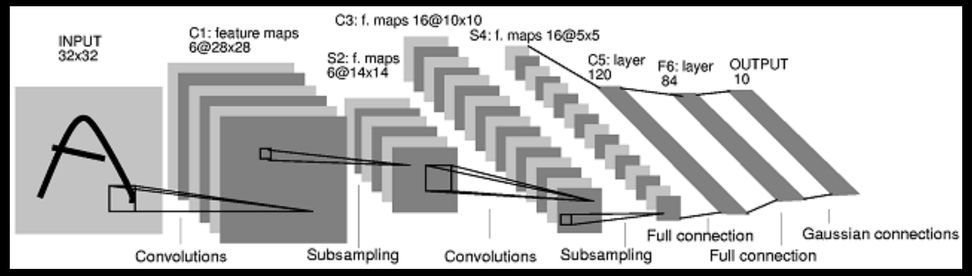
Neural Network



Deep Neural Network

Machine Learning

Introduction to Deep Learning

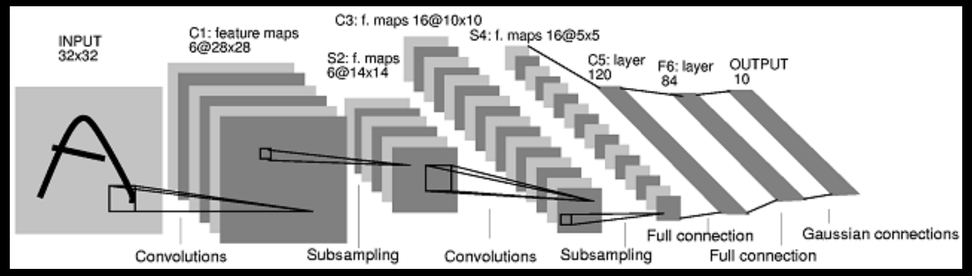


http://yann.lecun.com/exdb/publis/pdf/lecun-99.pdf

- Convolutional Neural Networks (or ConvNets or CNNs)
  - NNs that employ a mathematical operation called convolution, which is a filter (like in Photoshop) to an input that results in an activation. Repeated application of the same filter (or "kernel") creates a feature map that summarizes the presence of detected features (e.g. location of eyes on a face) in the input.
  - CNNs are not categorically different from the neural networks we have seen so far. The functionality is the same as earlier NNs (based upon perceptrons), and improve upon them by introducing a new type of layer, called a convolutional layer.
  - Based on how neurons in the visual cortices of mammals are organized.
  - Convolutional layers are the major building blocks used in convolutional neural networks and form the heart of deep learning.

Machine Learning

Convolutional Neural Networks



http://yann.lecun.com/exdb/publis/pdf/lecun-99.pdf

- Convolutional Neural Networks (or ConvNets or CNNs)
  - NNs that employ a mathematical operation called convolution, which is a filter (like in Photoshop) to an input that results in an activation. Repeated application of the same filter (or "kernel") creates a feature map that summarizes the presence of detected features (e.g. location of eyes on a face) in the input.
  - CNNs are not categorically different from the neural networks we have seen so far. The functionality is the same as earlier NNs (based upon perceptrons), and improve upon them by introducing a new type of layer, called a convolutional layer.
  - Based on how neurons in the visual cortices of mammals are organized.
  - Convolutional layers are the major building blocks used in convolutional neural networks and form the heart of deep learning.

Machine Learning

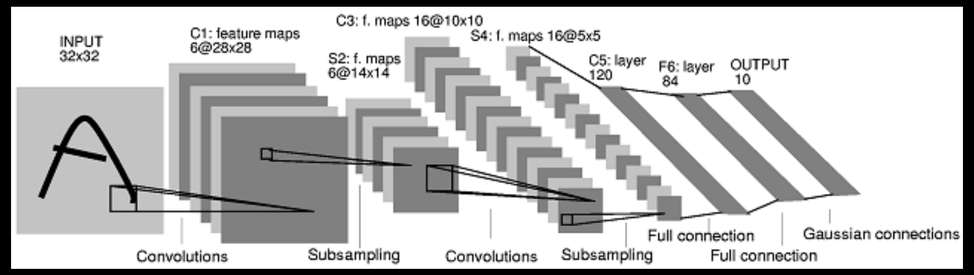
Convolutional Neural Networks

- Most commonly applied to analyzing visual imagery (computer vision), but also used in audio and text
- No need to select feature primitives (the CNN learns to extract features itself)
- The pixel information *is* the feature primitives (e.g. a 32 x 32 image = 1024 inputs to a CNN)
- Unlike earlier algorithms, CNNs can operate directly on a raw image and need little or no preprocessing.

Machine Learning

Convolutional Neural Networks

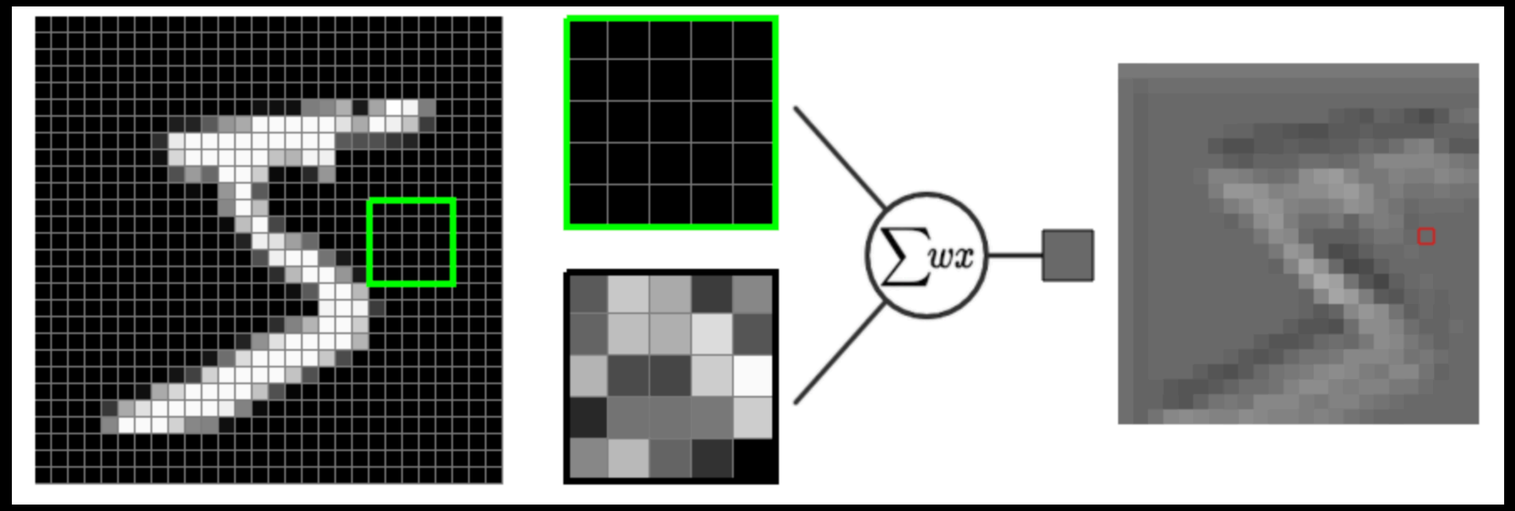
- So the major difference between a CNN and previous NNs is that each neuron is connected to the outputs of only a limited number of neurons from the previous layer (determined by the kernel size and pooling). This is a major improvement for computer vision tasks, as previous NNs interpreted their weights in such a way that created a “template” or a kind of average of the training set. This is good to a point, but it does not fare well on complicated visual information.



http://yann.lecun.com/exdb/publis/pdf/lecun-99.pdf

Machine Learning

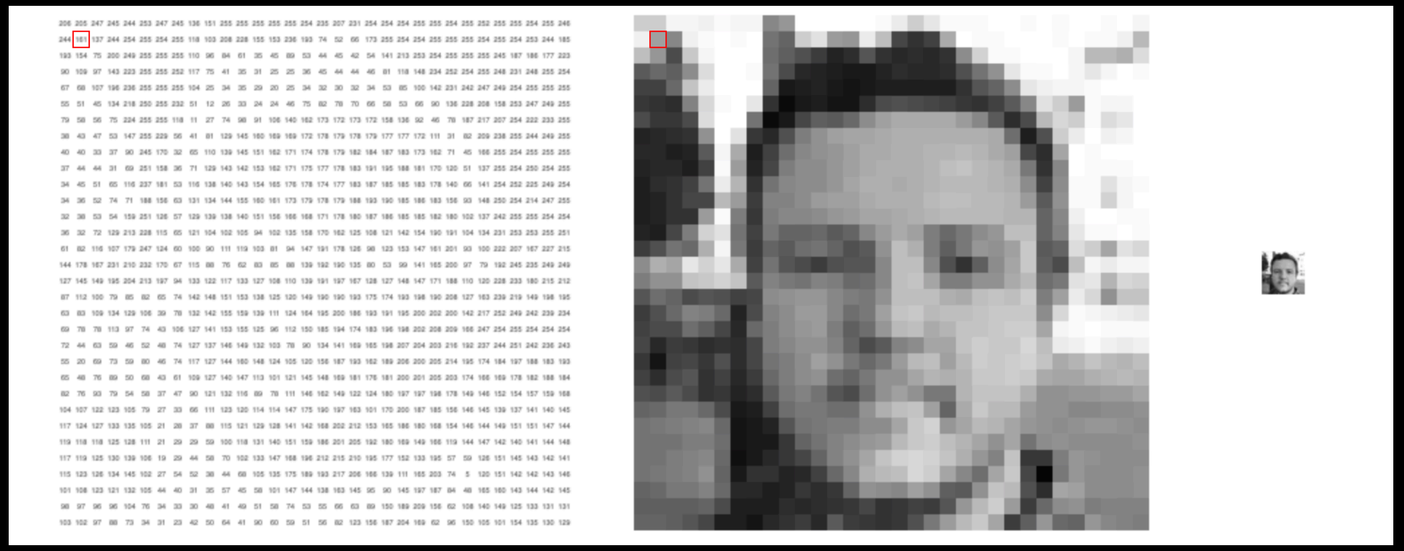
Convolutional Neural Networks



https://ml4a.github.io/demos/convolution/

Machine Learning

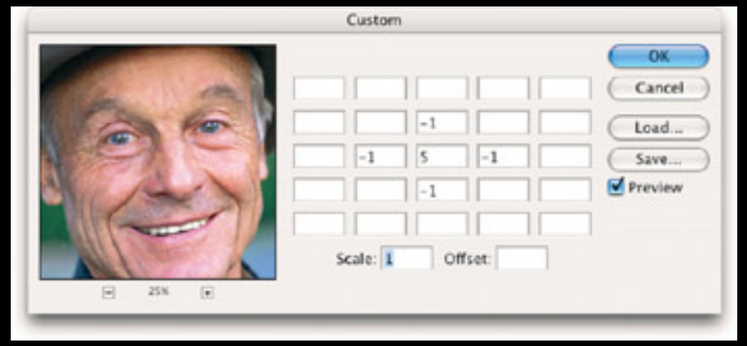
Convolutional Neural Networks



https://setosa.io/ev/image-kernels/

Machine Learning

Convolutional Neural Networks

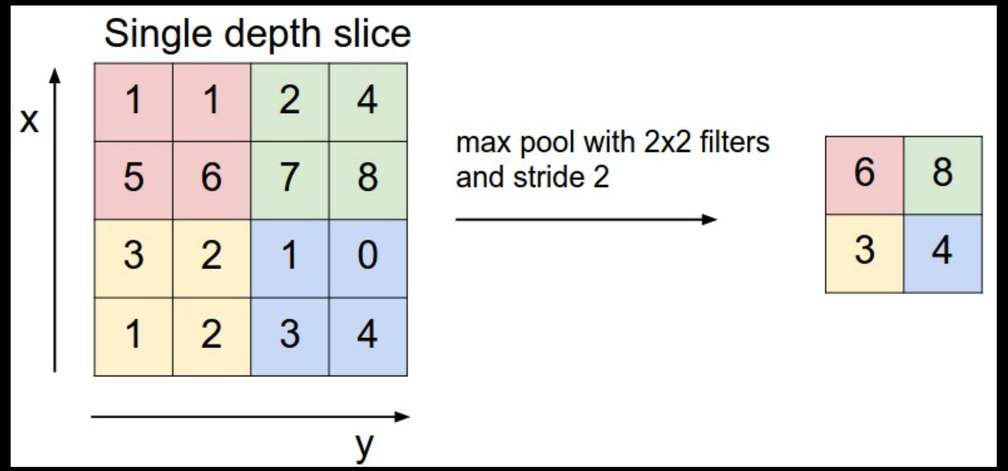


Photoshop Custom Filter Tool

Machine Learning

Convolutional Neural Networks

- Types of layers in a CNN:
  - Convolutional layers: a filter that passes over the image, scanning a few pixels at a time and creating a feature map that summarizes the presence of detected features.
  - Pooling layers: this is a form of downsampling; reduces the amount of information in each feature obtained in the convolutional layer while maintaining the most important information; also helps reduce overfitting



https://cs231n.github.io/convolutional-networks/

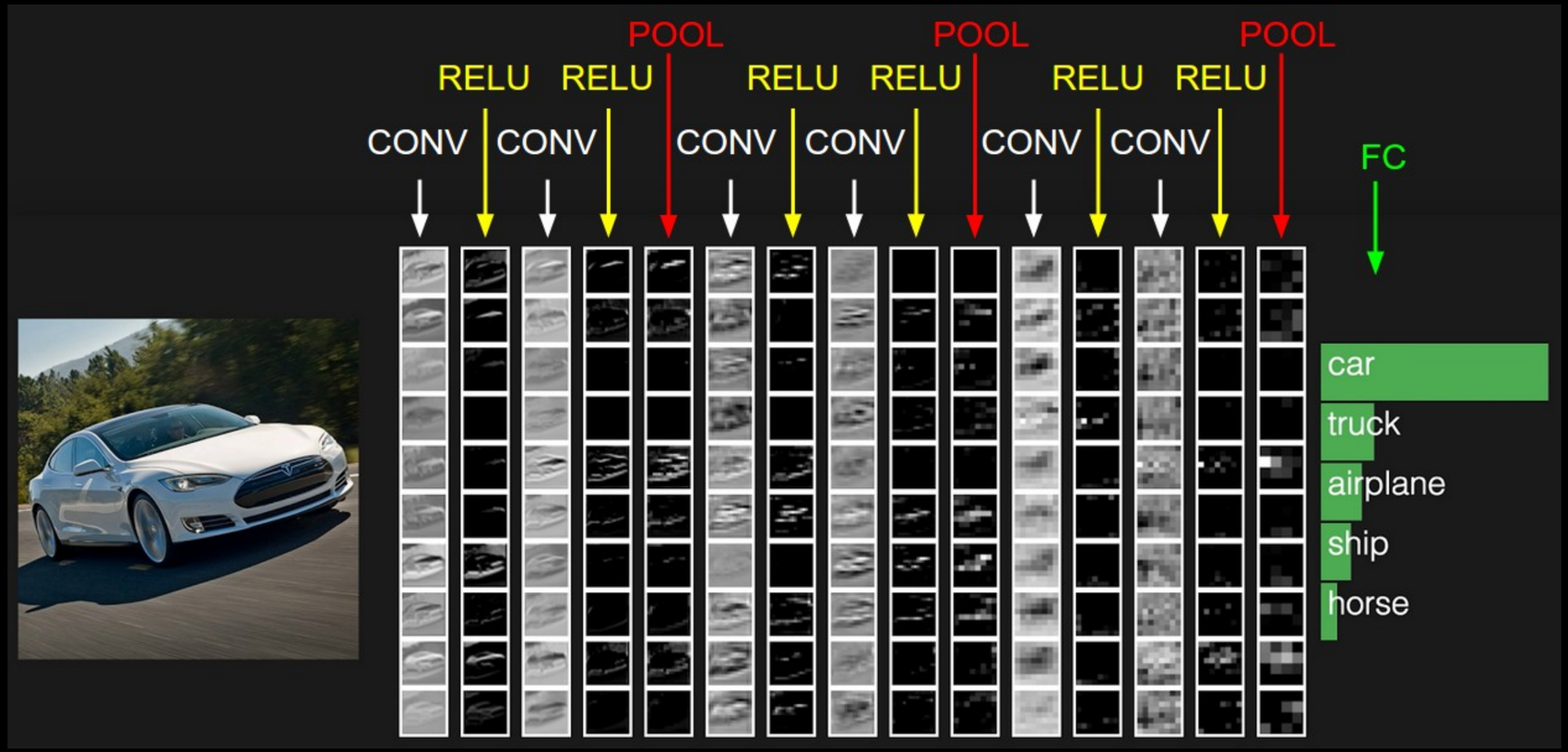
Machine Learning

Convolutional Neural Networks

- Types of layers in a CNN (continued):
  - Dense (fully connected) layers: full connections to all activations in the previous layer (just like in regular NNs).
    - Fully connected input layer (flattened): takes the output of the previous layers, “flattens” them and turns them into a single vector that can be an input for the next stage.
    - Fully connected output layer: gives the final probabilities for each label.
  - Batch Normalization layers: keeps weights of the network within a reasonable range of values (bad if they get too big)
  - Dropout layers: occasionally drop a neuron (sets output to zero); helps prevent overfitting

Machine Learning
Convolutional Neural Networks

- Typical CNN architectures:
  - Input
  - Conv2D
  - BatchNormalization
  - Activation (e.g. LeakyReLU)
  - Conv2D
  - BatchNormalization
  - Activation (e.g. LeakyReLU)
  - Pool
  - Conv2D
  - BatchNormalization
  - Activation (e.g. LeakyReLU)
  - Flatten
  - Dense
  - BatchNormalization
  - Activation (e.g. LeakyReLU)
  - Dropout
  - Dense
  - Activation (softmax)



https://cs231n.github.io/convolutional-networks/

“Computers will understand sarcasm before Americans do.”

Geoffrey Hinton

Machine Learning

Generative Models

- A Generative Model describes, in terms of probabilities, how a dataset is generated. By sampling from this model, one can generate new data that has a certain likelihood or similarity to that dataset.


This person does not exist


This person does not exist


This person does not exist

Machine Learning

Generative Models

- https://thispersondoesnotexist.com/
- https://www.thiswaifudoesnotexist.net/
- https://thiscatdoesnotexist.com/
- https://thisxdoesnotexist.com/

Machine Learning
Generative Models

- **Supervised Learning**
  - Data: (x, y) x is dataset, y is label
  - Goal: Learn a function to map x -> y
  - Examples: Classification, regression, object detection, etc.

- **Unsupervised learning**
  - Data: x is only data, no labels
  - Goal: Learn some underlying hidden structure of the data
  - Examples: Clustering, dimensionality reduction, feature learning, etc.

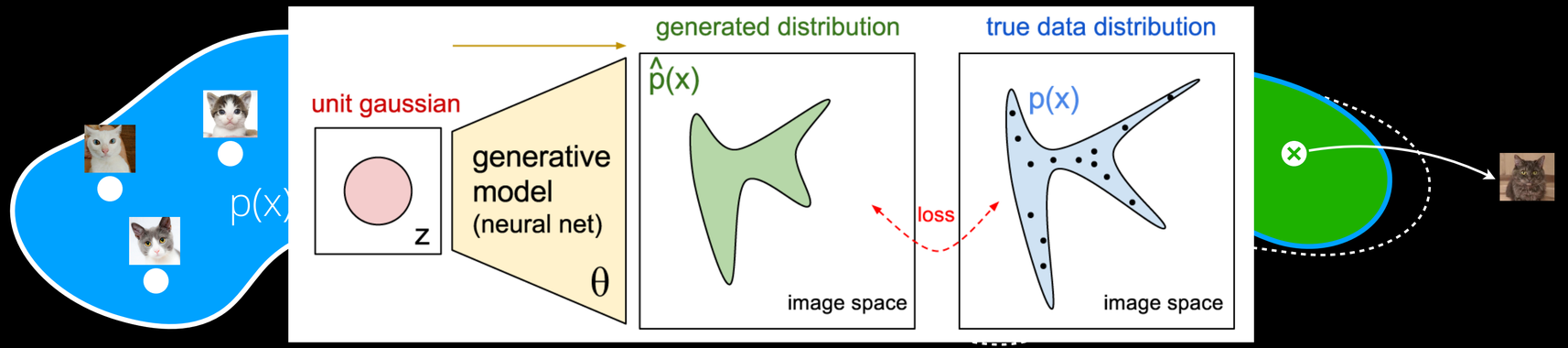
training data is cheap/faster

holy grail: understand structure of perception

Machine Learning

Generative Models

- A Generative Model describes, in terms of probabilities, how a dataset is generated. By sampling from this model, one can generate new data that has a certain likelihood or similarity to that dataset.
- Goal is to generate new data (e.g. images) from the same probability distribution (in an unsupervised manner)



all possible images of a dataset

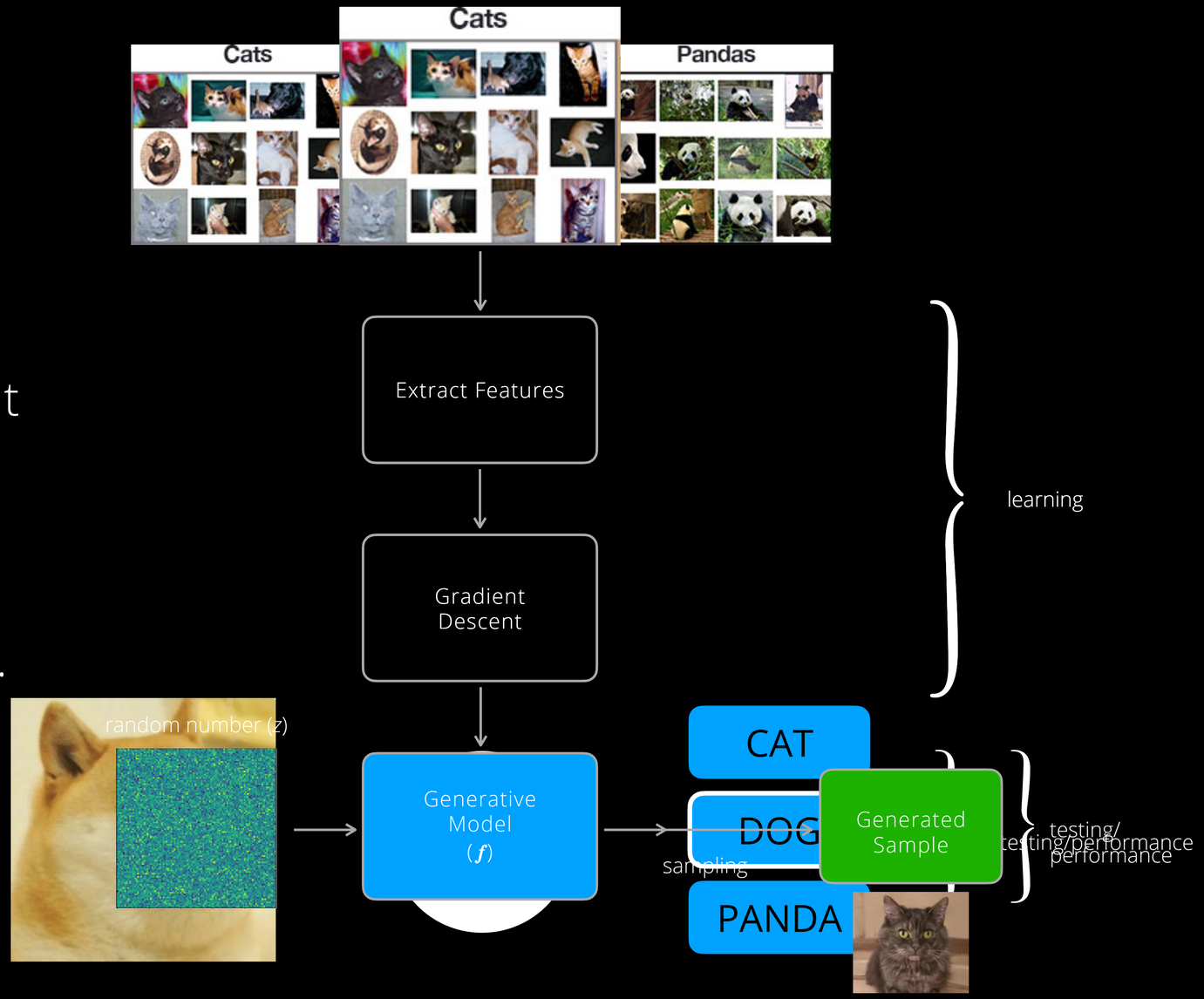
learns those possibilities and samples from them

https://openai.com/blog/generative-models/

Machine Learning

Generative Models

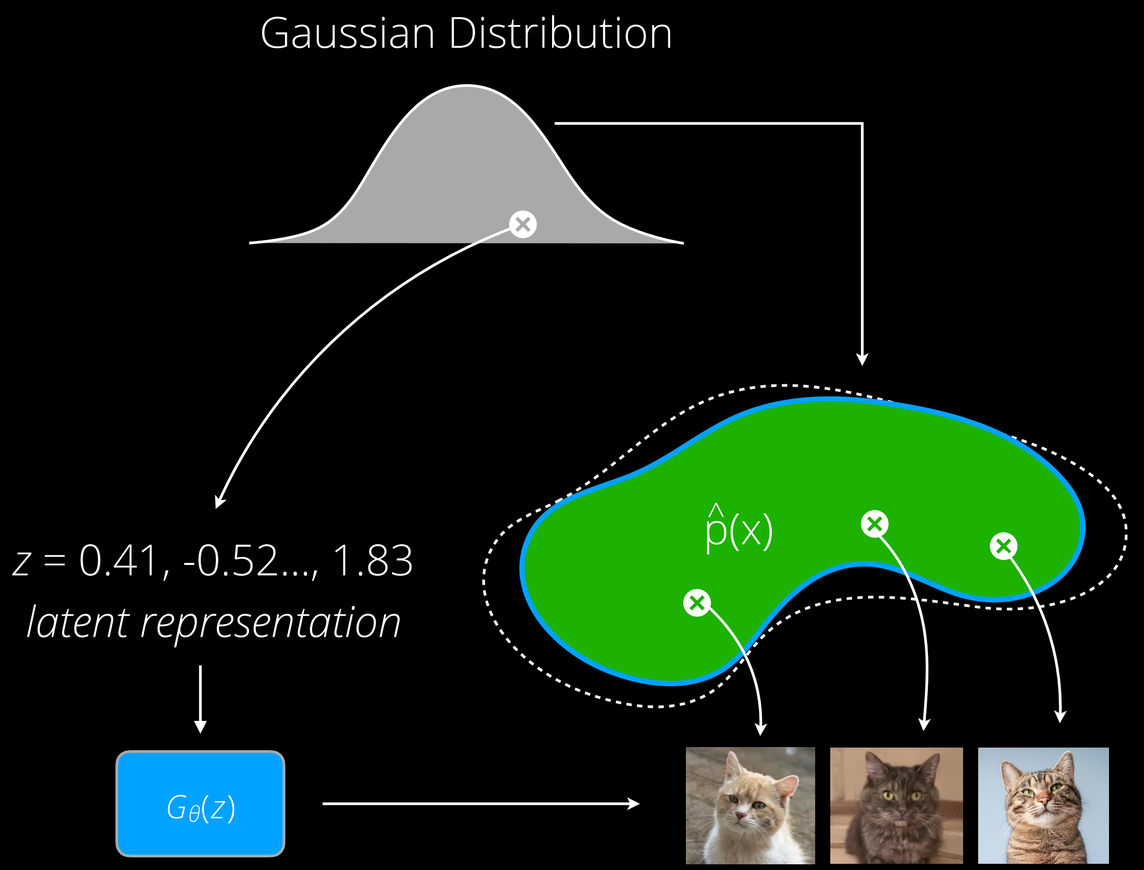
- Generative vs Discriminative Modeling
  - discriminative modeling: synonymous with supervised learning, maps an input to an output using a labeled dataset (classification or regression);
  - generative modeling: tells us the probability that a given observation (i.e. image) is possible, given a particular dataset; probability of observing observation x.



Machine Learning

Generative Models

- Generative models do not sample directly from the high-dimensional space of $\hat{p}(x)$.
- Each observation (e.g. image) in the training set is instead described by sampling from a simple distribution (e.g. a normalized Gaussian) in some low-dimensional latent space representation. A mapping function $G_\theta(z)$ (the model) is then learned that can take a point in the latent space and map it to a point in the original domain.
- Each point, or latent vector (z), is transformed into an image (or whatever datatype you are using) using that mapping function.
- So latent space is really a representation of compressed data (i.e. a low resolution version of the image), to which we apply the mapping function (to e.g. generate a new image similar to what is in the uncompressed dataset)
- n-pixel RGB image: $255^{(3)}$ (16,581375) possible values per pixel
- e.g. for a 32x32 image: $255^{(3*32*32)}$ possible images



Machine Learning

Generative Models

- Two types of Generative Models discussed today
  - Generative Adversarial Network (GANs)
  - Variational Autoencoders (VAEs)

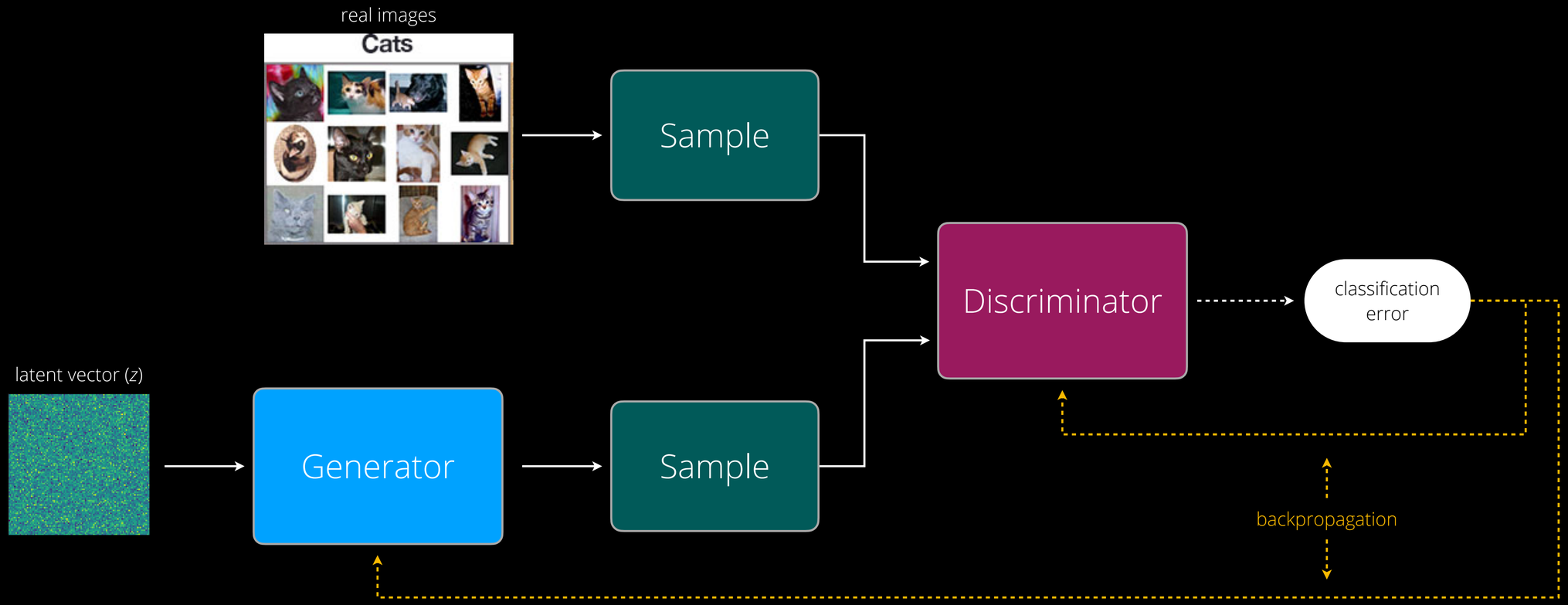
Machine Learning

Generative Models

- Generative Adversarial Network (GANs)
  - Based on an “adversarial” setup where two neural networks: the generator and the discriminator compete to optimize their own objectives
  - The generator tries to generate images that look like they are from the dataset. The discriminator tries to tell apart fake images coming from the generator and real images coming from the dataset. The generator never sees the training data directly.
  - The key to GANs lies in how the training of the two networks occurs. As the generator becomes better at fooling the discriminator, the discriminator must adapt in order to maintain its ability to correctly identify which observations are fake. This pushes the generator to find new ways to fool the discriminator, and so on.

Machine Learning

Generative Models



Generative Adversarial Network

Machine Learning

Generative Models

- Discriminator
  - standard binary classifier (a supervised image classification problem)
  - its training data comes from two sources:
    - real data instances (e.g. real photos of cats); used as positive examples during training.
    - fake data instances created by the generator; used as negative examples during training
- Training the Discriminator
  - receives images randomly selected from the training set (the real images) and some randomly selected from the output of the generator.
  - train the discriminator to learn how to tell the difference between the original and generated images
  - float between 0 -1; real image = 1, generated image = 0 (supervised learning)

Machine Learning

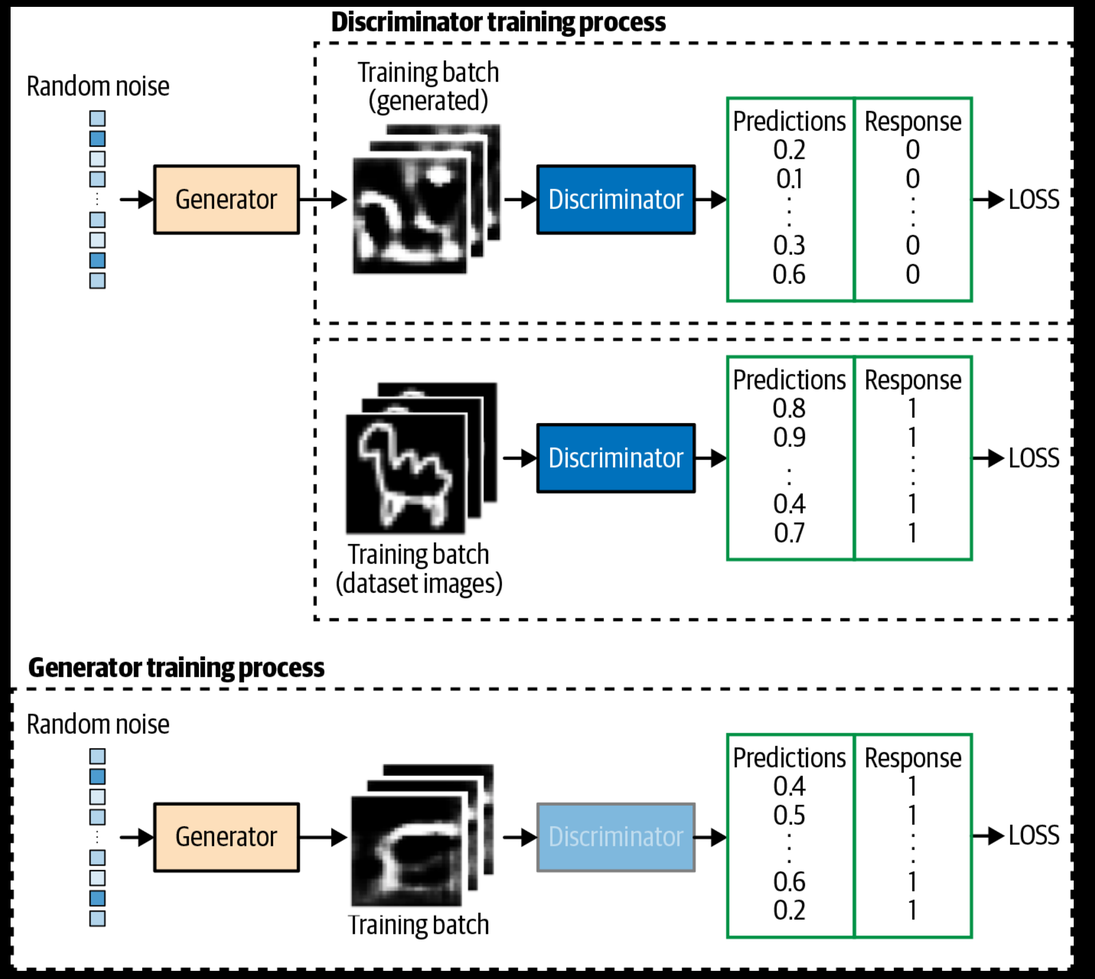
Generative Models

- Generator
  - converts random noise into observations that look as if they are from the original dataset
  - learns to create fake data by incorporating feedback from the discriminator
- Training the Generator
  - use the discriminator to train the generator
  - create samples of random noise (latent vectors) as input and a response which is set to 1 (i.e real)
  - generator loss comes from producing samples that the discriminator classifies as fake
    - e.g.: generate a fake image; generator says 1, discriminator says 0.4; generator loss is 0.6
  - only change generator weights updated when you backpropogate (very important!)

Machine Learning

Generative Models

- GAN Training Process
  - generator and discriminator alternate training
    - each trains for 1 or more epochs, one after the other.



Foster, *Generative Deep Learning*.

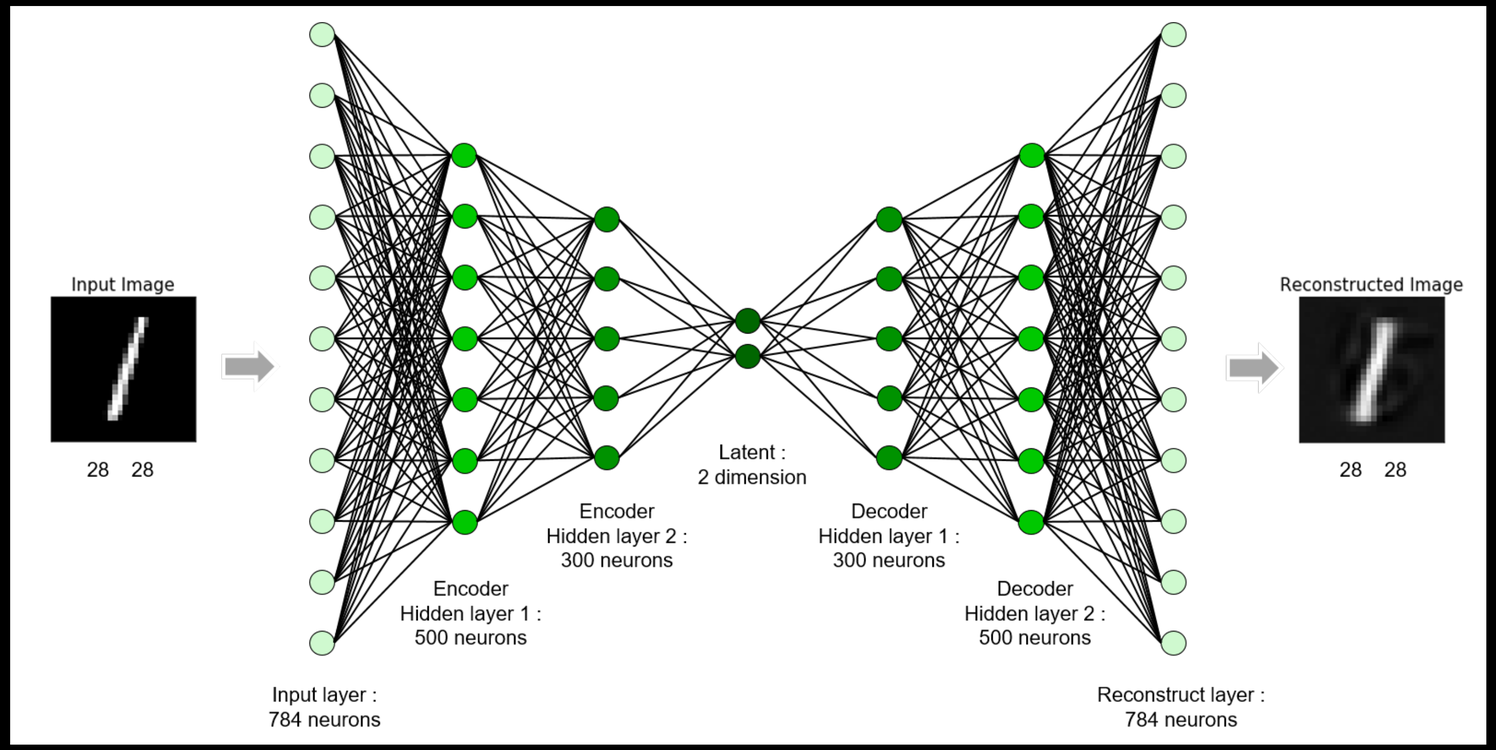
Machine Learning

Generative Models

- Variational Autoencoders (VAEs)
  - The basic idea of an autoencoder is to have an output layer with the same dimensionality as the inputs. The idea is to try to reconstruct each dimension exactly by passing it through the network. (Aggarwal, *Neural Networks and Deep Learning: A Textbook*)
  - a method of dimensionality reduction
  - a fundamental and widely used deep learning architecture for generative modeling

Machine Learning
Generative Models

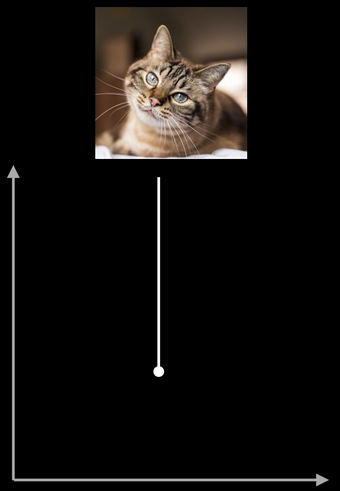
- Structure of an Autoencoder
  - Made up of two parts:
  - encoder: network that compresses high-dimensional input data into a lower-dimensional representation; a single point in latent space
  - decoder: network that decompresses a given representation back to the original domain

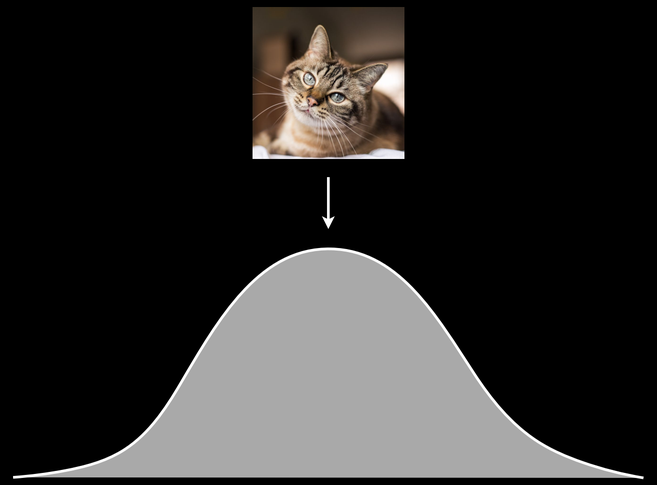


Machine Learning

Generative Models

- a Variational Autoencoder (VAE) improves on the basic autoencoder architecture and is used for generative models in practice
- in a VAE, each image is mapped to a Gaussian distribution around a point in latent space (in an autoencoder it is mapped directly to one point in the latent space)





“Nothing is wholly obvious without becoming enigmatic. Reality itself is too obvious to be true.”

Jean Baudrillard