Deep Learning & Engineering Application

1. Introduction

JIDONG J. YANG, PH.D., P.E.
COLLEGE OF ENGINEERING
UNIVERSITY OF GEORGIA

Self-Introduction

Your name, department/school, and research area/focus.

How many years of experience in data-related work? Any programming experience?

Have you taken any data science or machine learning courses before?

Why do you want to take this course?

Learning from “Machine Learning”

- Selectively take information (Too much noise is bad for learning)
- One thing at a time (Attention Mechanism)
- Organize information (Convolutional Neural Networks)
- Review periodically (Reinforcement Learning)

Learning from “Human Learning History”

Common sense (intuition) is a good thing in engineering practices.

Not all intuitions are trustworthy, especially in science. Many times in human history, counter-intuitions lead to breakthrough discoveries.

Counter-intuitions are often the doors to deeper understanding of something we don’t know yet. Don’t miss out these opportunities.

How to learn “Deep Learning”

Practice & Persevere

"Essentially, all models are wrong, but some are useful"

- George E.P. Box (1919 – 2013)

All learning depends on imposing inductive bias. When that bias agrees with reality, we get sample-efficient models that generalize well to unseen data.

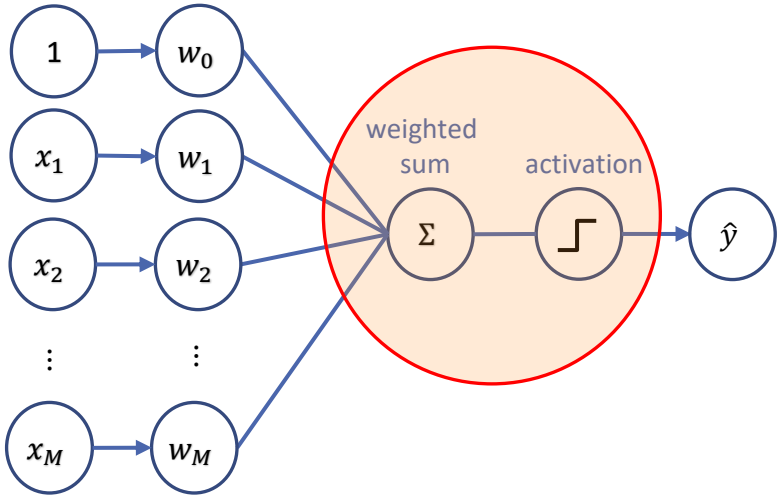
Introduction

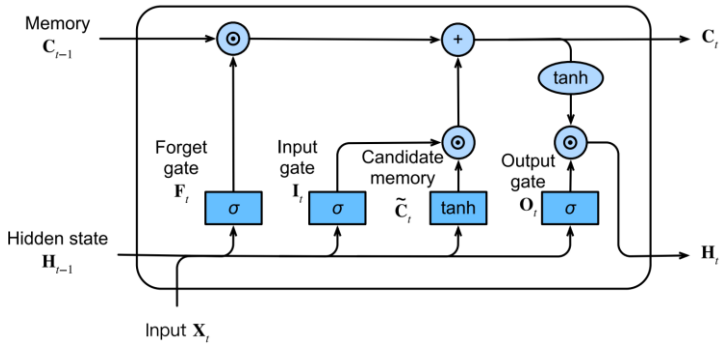
Building Blocks

- Neurons
- Units/Cells
- Layers
- Blocks

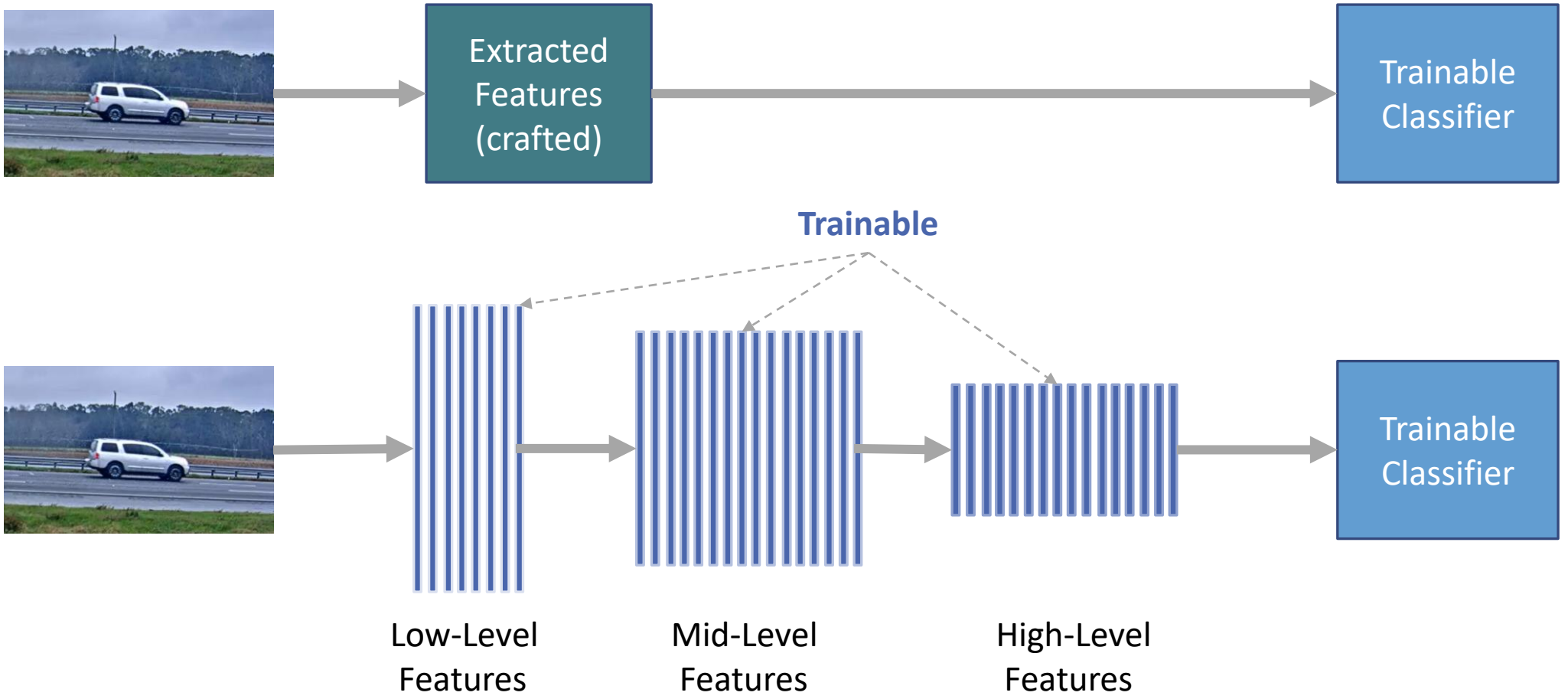
Graphs

- Feed-forward (e.g., Fully-connected, locally-connected, skip connection)
- Recurrent (e.g., RNN, LSTM, GRU, LTC)
- Intra-layer (e.g., self-attention)
- etc.



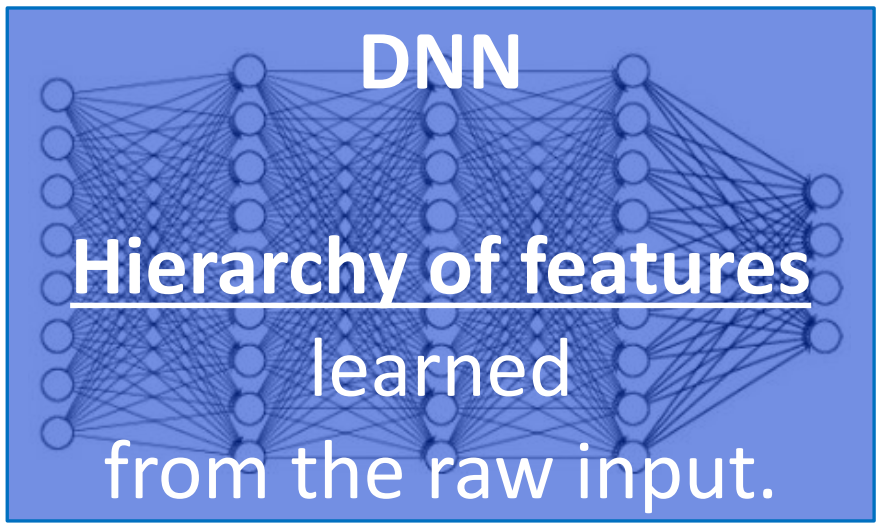


Traditional Machine Learning vs. Deep Learning



Core Ideas of Deep Learning

Raw signals (e.g., image, waveform, text, …)



Object classification

Object detection

Image labeling

Gesture recognition

Voice recognition

Translation

Etc.

Why going deep?

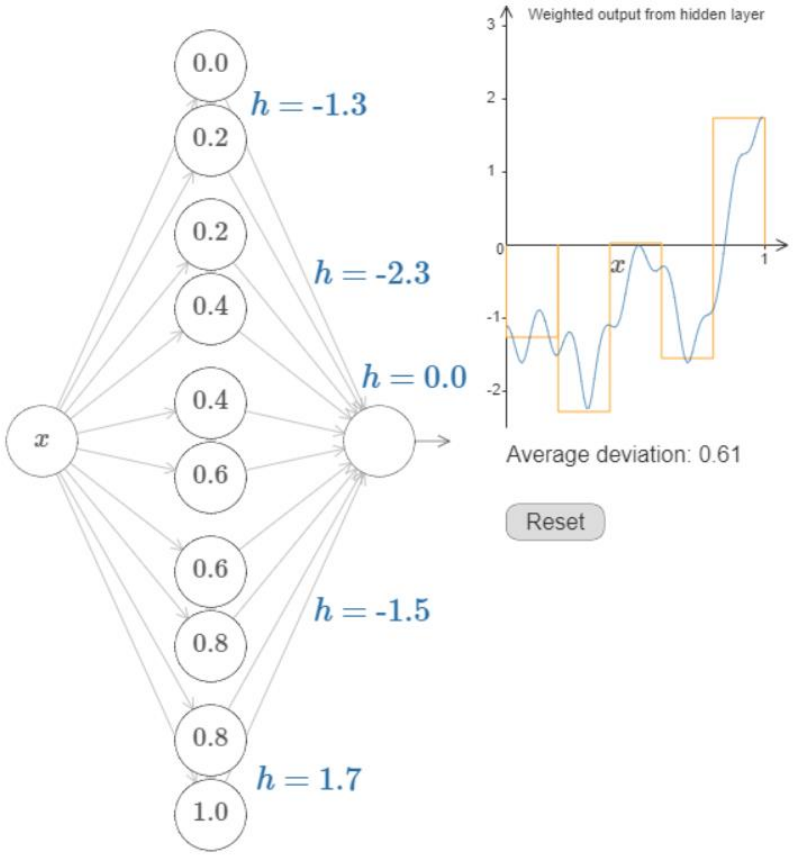
Neural network models have been well studied w.r.t. their widths and depths.

- Universal approximation theorems
- Functional expressivity

Infinite wide neural networks are practically infeasible to train. (resources demanding).

Keep finite width, but go deeper? Trade-off: space (width) vs. depth (time)

Deep neural network architectures are implementable with available computational resources.



http://neuralnetworksanddeeplearning.com/chap4.html

The deeper the better?

It will take quite amount of time and energy to train a decent DNN.

Model	Hardware	Power (W)	Hours	kWh·PUE	CO_2e	Cloud compute cost
$\text{Transformer}_{base}$	P100x8	1415.78	12	27	26	$41–$140
Transformer_{big}	P100x8	1515.43	84	201	192	$289–$981
ELMo	P100x3	517.66	336	275	262	$433–$1472
BERT_{base}	V100x64	12,041.51	79	1507	1438	$3751–$12,571
BERT_{base}	TPUv2x16	—	96	—	—	$2074–$6912
NAS	P100x8	1515.43	274,120	656,347	626,155	$942,973–$3,201,722
NAS	TPUv2x1	—	32,623	—	—	$44,055–$146,848
GPT-2	TPUv3x32	—	168	—	—	$12,902–$43,008

Table 3: Estimated cost of training a model in terms of CO_2 emissions (lbs) and cloud compute cost (USD).[7] Power and carbon footprint are omitted for TPUs due to lack of public information on power draw for this hardware.

Strubell E., Ganesh A., and McCallum A., Energy and Policy Considerations for Deep Learning in NLP, 2019

https://arxiv.org/pdf/1906.02243.pdf

Biological Root

Hubel & Wiesel (1962)

- Simple cells detect local features.
- Complex cells “pool” the outputs of simple cells.

The visual cortex does pattern recognition in a hierarchical manner. Neurons in front of our retina compress the input and the signal travels from our eyes to our brain. After this, the image gets processed in stages and certain neurons get activated for certain categories.

Visual process is essentially a feed forward process. Fast recognition (perception) can be done without recurrent connections.

CNN History

AlexNet (2012) convinced the computer vision (CV) community that CNNs really work.

Over the years, the number of layers used has been increasing: LeNet – 7, AlexNet – 12, VGG – 19, ResNet – 50, etc.

Trade-off between the complexity of the model and its accuracy.

How to compress the networks to make the computations faster?

Compositional Hierarchy in Representation

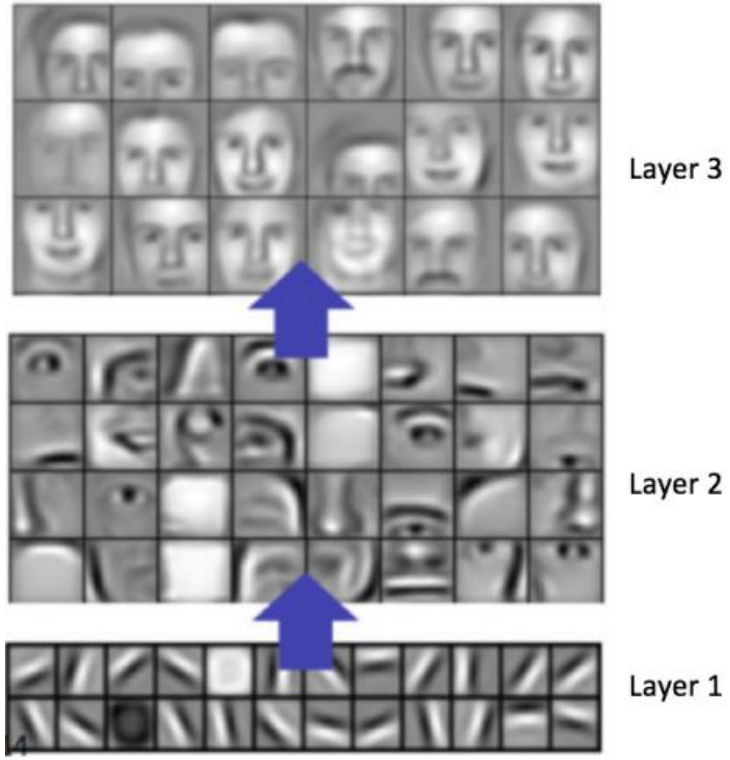
Data (image, text, speech) are compositional in nature.

Deep networks exploit the compositional structure of natural data.

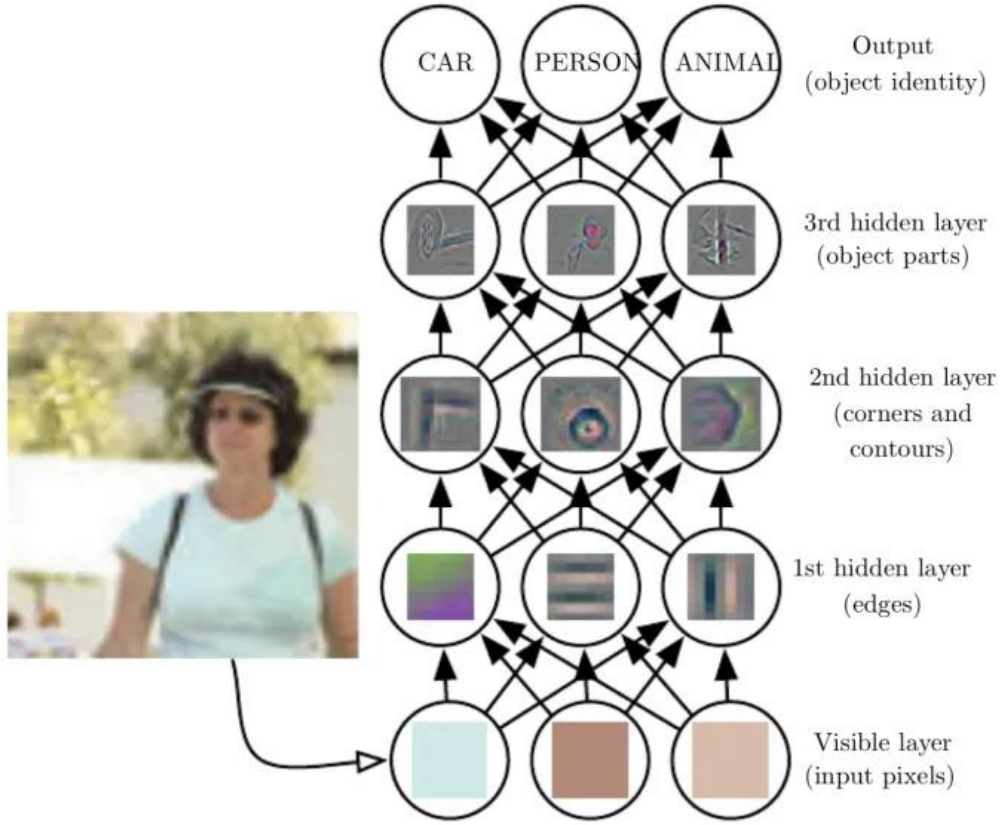
Deep Learning architecture is inspired by mimicking the compositional hierarchy through the depth of many layers, where combinations of objects at one layer in the hierarchy form the objects at the next layer.

The learned features have a hierarchy of representations with increasing level of abstractions.

Hierarchical Feature Representation



Lee et al. (2009). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. https://dl.acm.org/doi/10.1145/1553374.1553453



Goodfellow et al. (2016), http://www.deeplearningbook.org

Deep Learning ~ Representation Learning

What do we mean by "deep"?

- An SVM has basically two layers, which is not considered as "deep".

  $y = \sum_{i=1}^{N} \alpha_i K(X, X_i) = F_1(W_1 F_0(W_0 X))$

  For example, $y = \sum_{i=1}^{N} w_i \cdot \exp\left(-\gamma \|x - x_i\|^2\right)$

- Classification/regression trees are hierarchical, but not considered as "deep" because every tier (layer) of the trees analyzes the same features.

A deep network has a sequence of layers for building a hierarchy of features of increasing complexity.

Neural Tangent Kernel (NTK)

Given an initial parameter vector w_0 and a loss function $L = \sum_i (y_i^*, y_i)$, gradient descent repeatedly modifies the model's parameters w by subtracting the loss's gradient from them, scaled by the learning rate:

$$w_{t+1} = w_t - \epsilon \nabla_w L(w_t)$$

Define a path kernel: $K(x, x') = \int_{c(t)} \underbrace{\nabla_w y(x) \cdot \nabla_w y(x')}_{\text{Neural tangent kernel}} dt$ where, $c(t)$ is the path.

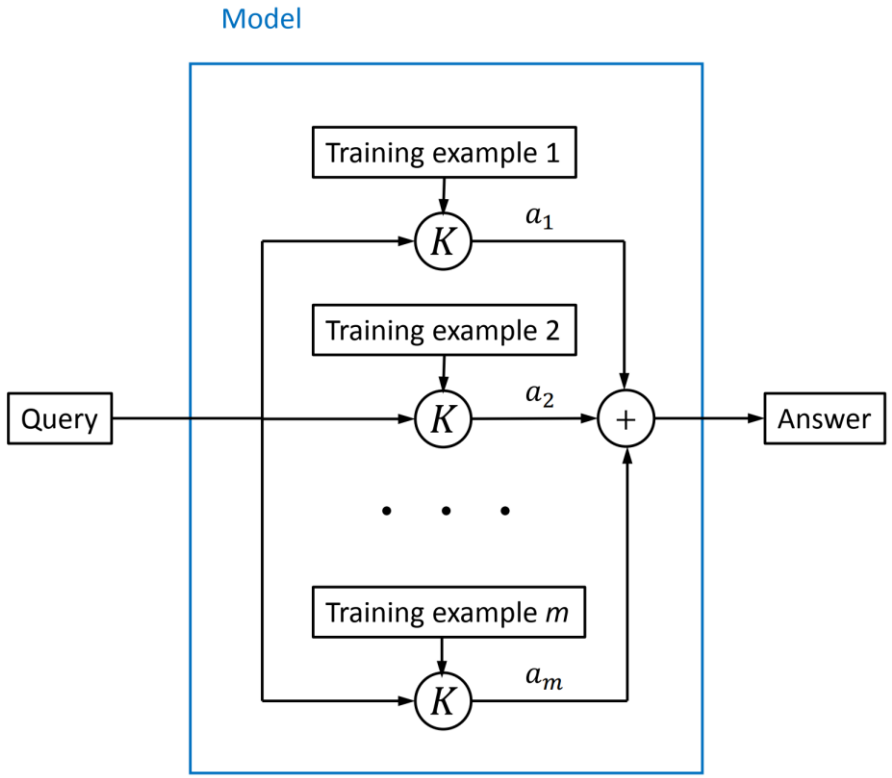
Two data points are similar if the candidate function's derivatives at them are similar, rather than if they are close in the input space.

Jacot et al. (2018) Neural Tangent Kernel: Convergence and Generalization in Neural Networks, https://arxiv.org/abs/1806.07572
Domingos (2020) Every Model Learned by Gradient Descent Is Approximately a Kernel Machine, https://arxiv.org/abs/2012.00152

Interpretability of Deep Networks as a Kernel Machine

In particular, the weights of a deep network have a straightforward interpretation as a superposition of the training examples in gradient space, where each example is represented by the corresponding gradient of the model.

Applying the learned model to a query example is equivalent to simultaneously matching the query with each stored example using the path kernel and outputting a weighted sum of the results.



Domingos (2020) Every Model Learned by Gradient Descent Is Approximately a Kernel Machine, https://arxiv.org/abs/2012.00152

Manifold Hypothesis

Natural data lives in a low-dimensional manifold.

What is the number of all possible images for RGB image of size 1000x1000?

How many these images represent “natural” images?

Natural images are well-structured and constitute a tiny subset of all possible images in the 3,000,000-dimension space.

What is the number of all possible images?

Practicality

Use your own computers (with a decent GPU)

Google Colab: https://colab.research.google.com/notebooks/intro.ipynb?utm_source=scs-index

UGA – Georgia Advanced Computing Resource Center (GACRC): https://gacrc.uga.edu/

Paid cloud computing services (Google Cloud, Amazon Web Services, Microsoft Azure, etc.)