Danbury AI

Neural Networks in Python for Beginners

Neural networks provide a powerful framework for modeling complicated data. From image recognition to language modeling, you can find high performance neural architectures available for building models of your data. Join us in this introductory workshop where we will show how to build neural networks in Python easily with the Keras api.

June 2018

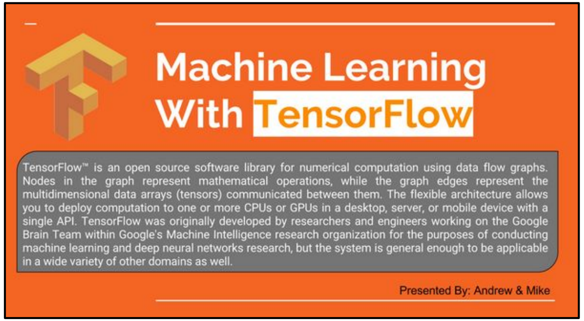
Andrew@kexp.io

ABOUT DANBURY AI - *Mastering AI Together*

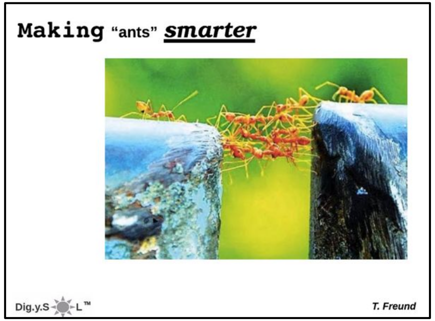
Danbury AI is a public AI meetup group hosted by the Danbury Hackerspace that aims to stimulate discussion in the vast field of artificial intelligence and bring together locals that foster a passion for the field. We hold our in person meetings on the first Tuesday of every month at which we host a variety of presenters, discussions, and workshops. We maintain a lively web presence via slack so the conversation never stops.

"If you want to go fast, go alone; If you want to go far, go together."

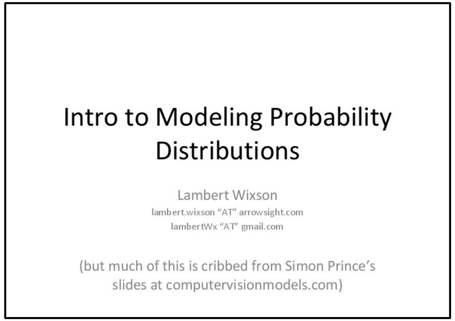
- Heavily inspired by initiatives like Open AI.
- We hold monthly **in-person meetings** at the Danbury Hackerspace.
- We aim to conduct **study groups** and other initiatives of group learning. We have a wide variety of members from all reaches of industry.
- We have a group chat room and **forum on slack for sharing resources and coordination.**
- Main website: https://www.meetup.com/DanburyAI/
- Join our bimonthly AI newsletter: **https://bit.ly/2y1aY2o**



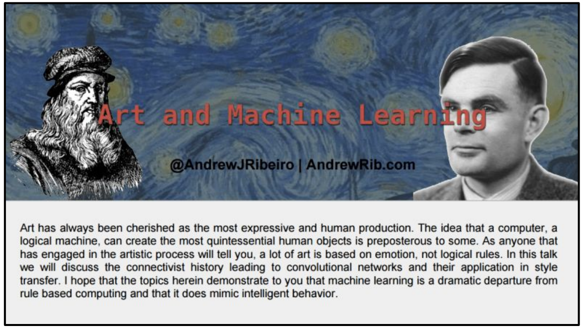
Slides | Repo | Workshop



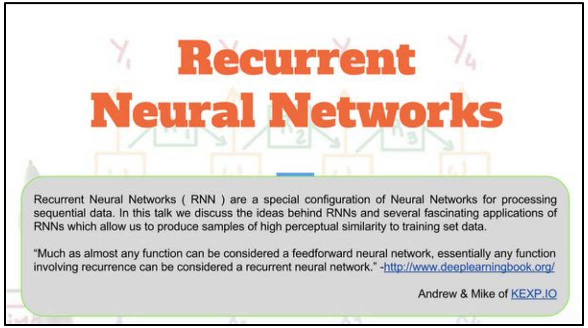
Slides



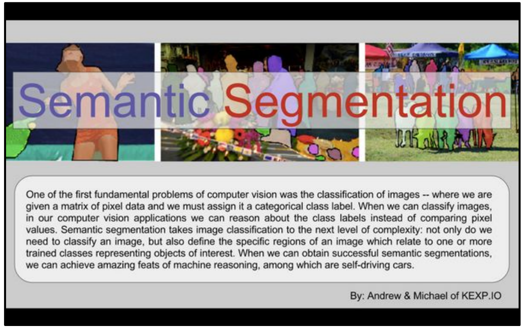
Slides



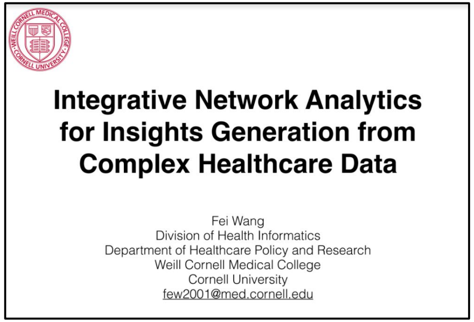
Slides



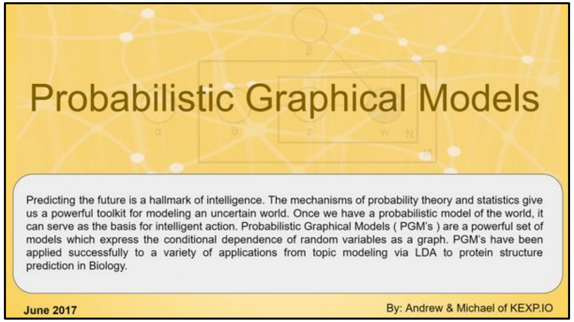
Slides



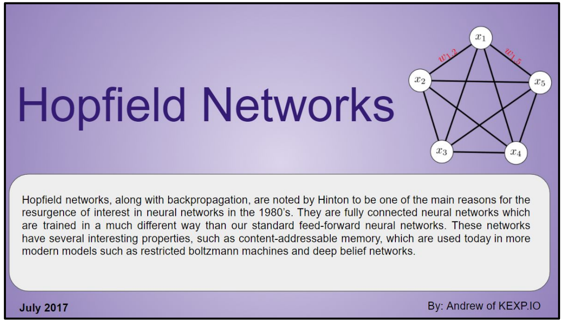
Slides



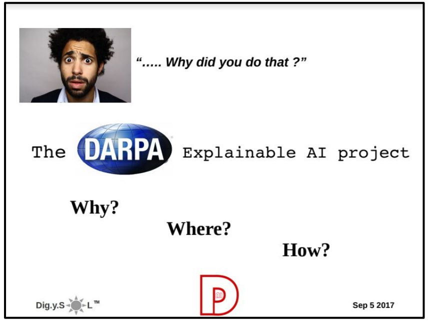
[Slides]



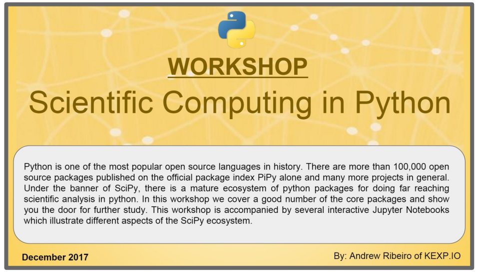
[Slides]



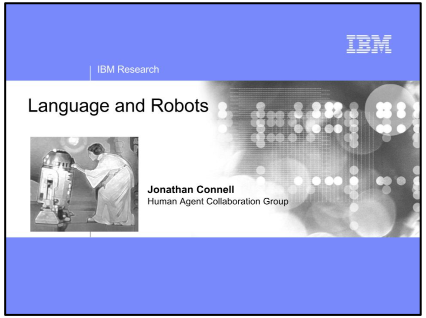
[Slides]



[Slides]



[Repo | Slides]



[Slides | Video]

About Me

- Co-Founder of [Knowledge-Exploration Systems.]
- Co-Organizer of [Danbury AI] **April 2016**
- Studied Computer Science at [WCSU]
- Main interests
  - Machine Learning
  - Neural Networks
  - Natural Language Processing
  - Adaptive Learning Systems
  - Collaborative Technology
  - Theoretical Computer Science
  - Mathematics (Linear Alg, Analysis, Statistics, Foundations)

Introductions

Welcome to Danbury AI! Let's introduce ourselves. Here are some things we'd like to know:

- Name?
- What is your experience with AI?
- Areas of interest?
- Current projects?
- What would you like to get out of this meetup?

OUTLINE

- **Introduction**
    - Audience and Key Workshop Takeaways
    - Statistics, Machine Learning, and Data Science
    - Common Machine Learning Tasks
- **Neural Networks Overview**
    - Cognitive Science and AI
    - Neurons
    - Neurons in Animals
    - Artificial Neurons
    - Networks
    - Neural Networks
    - Origin of Neural Networks
    - Components of a Neural Network
    - Stages of Training a Neural Network
    - Types of Neural Networks
    - Deep Learning
- **The Kaggle Platform**
    - About Kaggle
    - Anatomy of a Competition
    - Working with Kernels
- **Neural Networks in Python**
    - Neural Network Frameworks
    - Hands On
        - Titanic
        - MNIST
        - SVHN
        - Toxicity
- **Deep Learning Applications**
    - Image-to-Image Translation
    - Image Synthesis
    - Image Segmentation
    - Style Transfer
    - Sequence Modeling
    - Audio Synthesis

KEXP

Introduction

Audience and Key Workshop Takeaways

Beginner

“I don’t know much, but I’m interested!”

Workshop Takeaways

- Awareness of broad topics in machine learning and neural networks.
- What resources are available to get started.

Developer

“I’m a programmer, but don’t work in a mathematical field.”

Workshop Takeaways

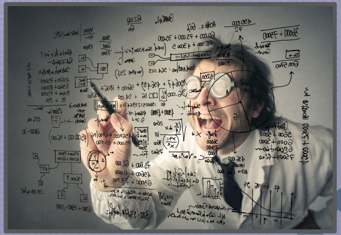
- What python libraries are available for doing data science and working with neural networks and the basics of how to use some of them.
- When you need to turn to machine learning to solve an engineering problem.

Data Scientist

“I’m a mathematical programmer and develop models, but have little experience with neural networks.”

Workshop Takeaways

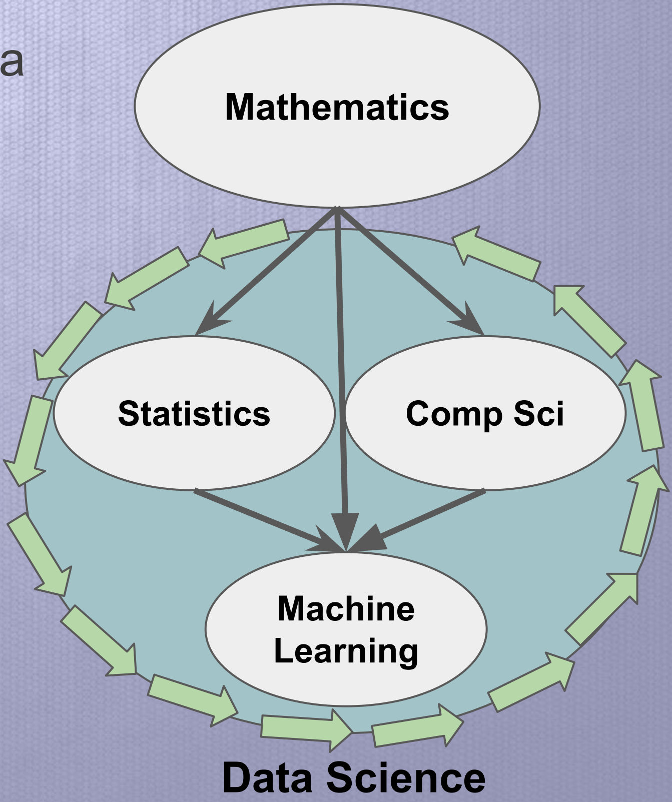
- When neural networks are more effective than traditional models.
- How to evaluate the performance of neural networks and intuition of their hyperparameters.



Statistics, Machine Learning, and Data Science

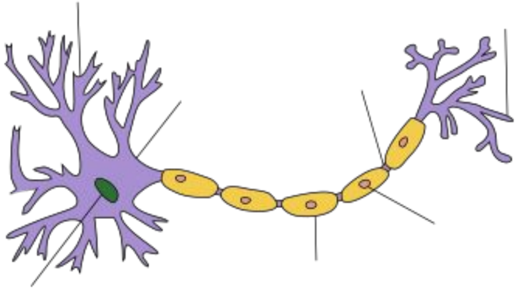
Statistics, Machine Learning, and Data Science have a huge overlap in what they aim to do, produce value from data, but people tend to ascribe different approaches and foci to each of these fields.

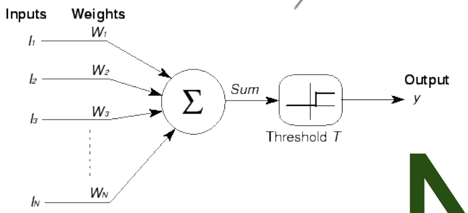
- Statistics is a branch of mathematics and is generally more deductive and mathematically rigorous than the other disciplines.
- Machine Learning is more experiment driven and is primarily focused on solving practical problems that are not easily analysed deductively.
- Data Science is a newer term that essentially refers to “anyone that uses code to analyse data.”

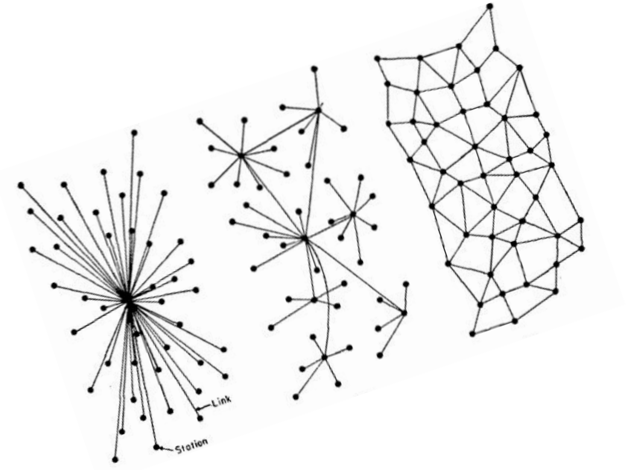


Common Machine Learning Tasks

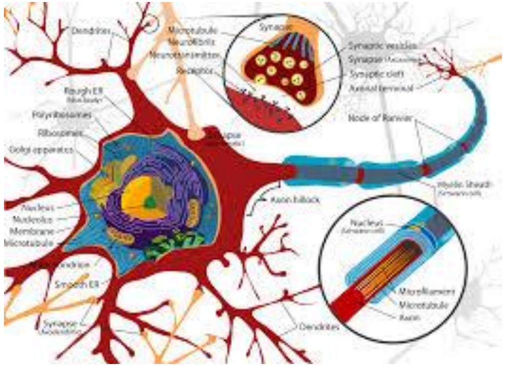
- **Classification**: Given a dataset where every sample has a feature vector and *class label*, produce a model that can predict the *class label* of a new feature vector not seen during training.
  - Example: given the pixels of an image, determine if it is a picture of a hotdog or not hotdog.
  - Neural Network: Network with a sigmoid output for binary classification, and softmax for multi-class classification.
- **Regression**: Like classification, but we are predicting *real values* instead of *class labels*.
  - Example: given the biometric measurements of a person, determine how long it will take them to run a mile.
  - Neural Network: Network with a linear output.
- **Dimensionality Reduction**: Given a dataset where every sample has a feature vector, transform each feature vector into a smaller feature vector while minimizing some reconstruction error.
  - Example: compression of files, unsupervised feature learning.
  - Neural Network: Autoencoders.

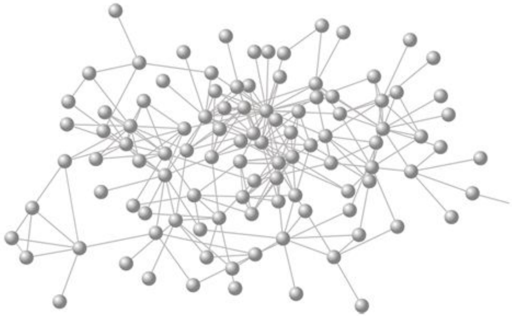






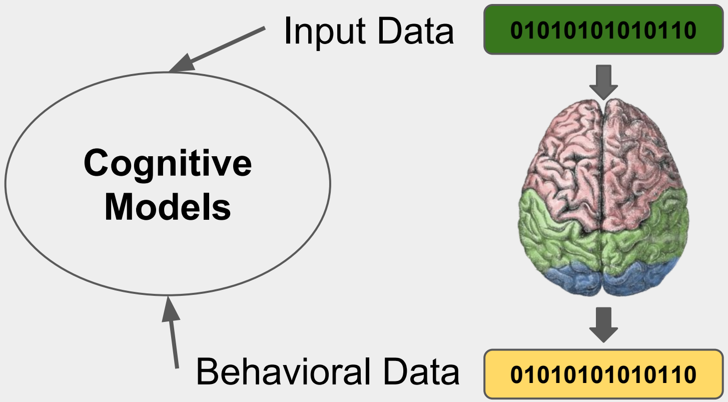
Neural Network





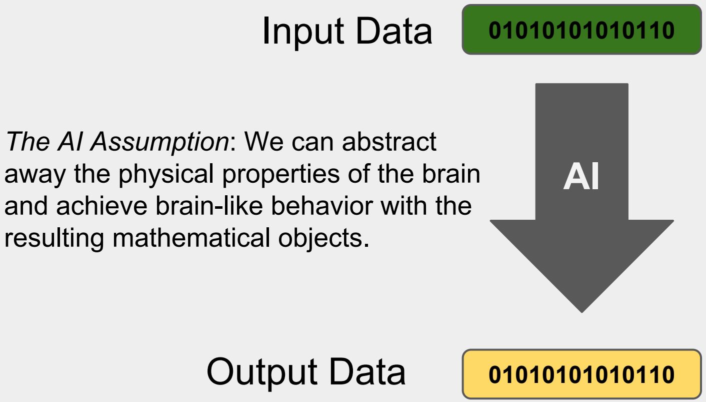
Cognitive Science

We can understand the mechanisms of the brain by studying how it processes information.



Artificial Intelligence

We can understand the mechanisms of the brain by studying information processing in general.



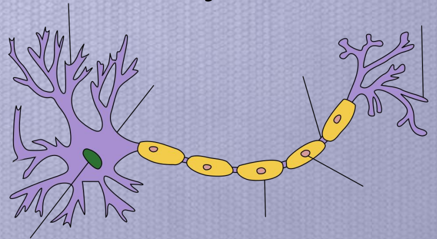
Both fields make the assumption that the brain is fundamentally an *information processing* machine.

Andrew Ribeiro 2018

Neurons

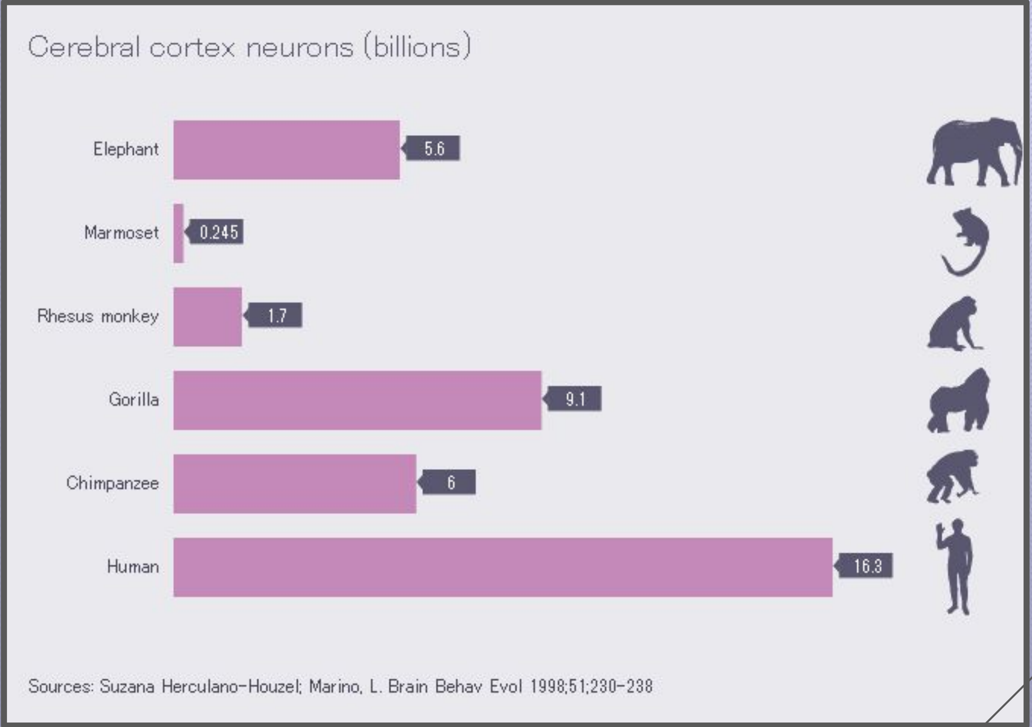
The brain has about 86 billion of these.

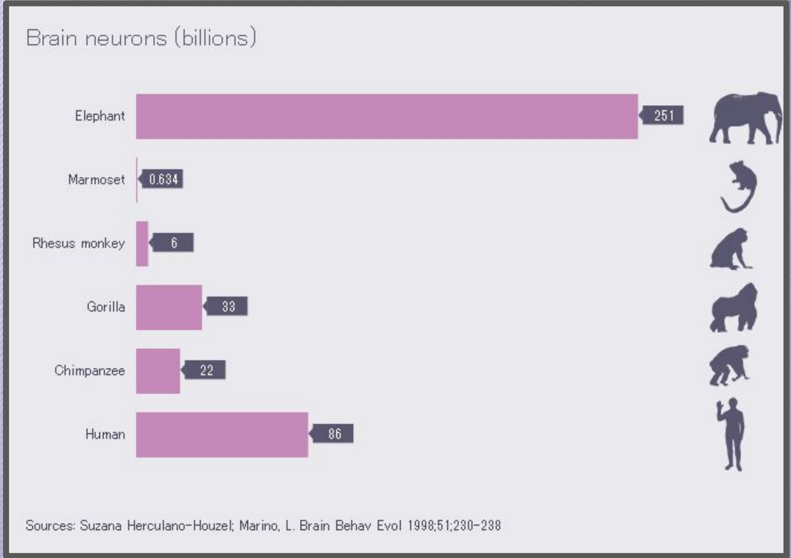
- Biological neurons are the core building block of the nervous system. Our nervous system is responsible for coordinating all the signals that keep different parts of our body working together as a whole.
- Neurons become specialized for different tasks. (i.e. sensory and motor)
- Neurons have three main components:
    - The Cell Body (soma)
    - Axon - Sending Signals
    - Dendrites - Receiving Signals

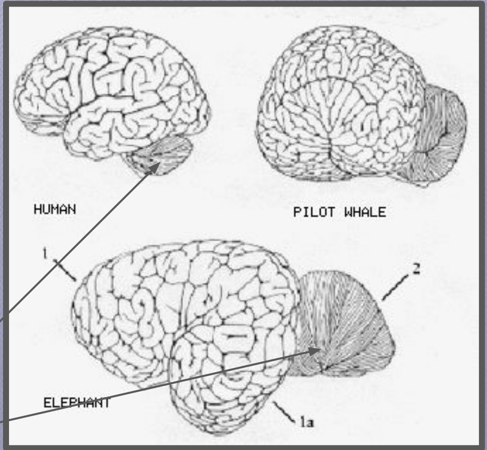


- Neurons communicate with chemicals called Neurotransmitters and by electrical signals.
- Artificial Neurons are a mathematical model of the basic computational functions of a neuron. They do not come close to the true complexity of biological neurons; however, they are still amazingly useful.

Neurons in Animals







SOURCE

SOURCE

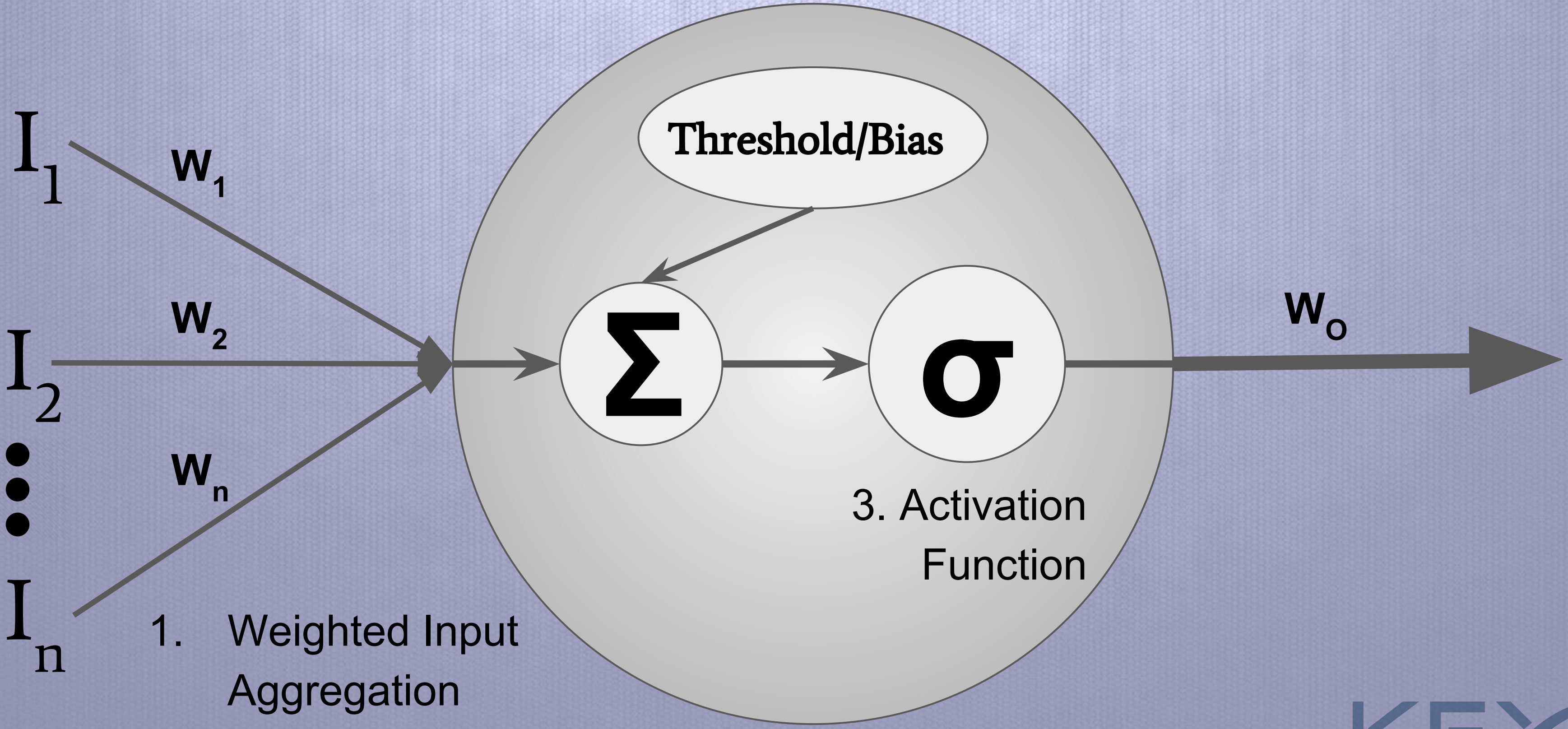
Why aren't elephants running the world?

Cerebrum vs Cerebellum

In linear algebra:
$\sigma(I^T W + B)$

Artificial Neurons

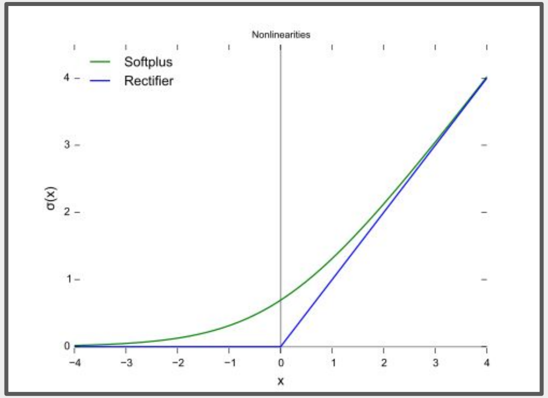
2. Threshold/Bias



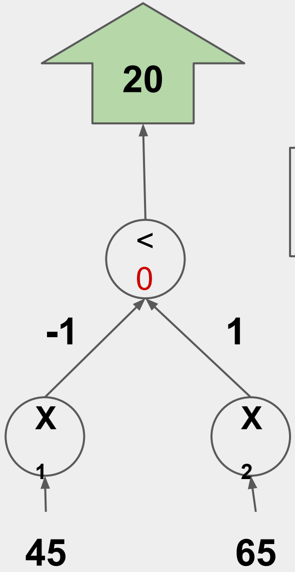
The Less Than Difference Neuron

$$f((x_1, x_2) \in \mathbb{R}^2) \begin{cases} 0 & x_1 \geq x_2 \\ x_2 - x_1 & x_1 < x_2 \end{cases}$$

The Activation Function: RELU



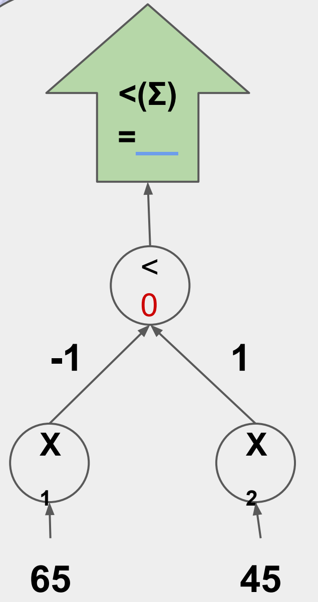
$$< (n \in \mathbb{R}) \begin{cases} 0 & n \leq 0 \\ n & n > 0 \end{cases}$$



H = -1*45 + 1*65 + 0 = 20
<(H) = **20**

Handout: Perform a forward propagation on this neuron.

1. Calculate the weighted value of X_1.
2. Calculate the weighted value of X_2.
3. Calculate the sum of the weighted values plus the node bias.
4. Compute the activation function on the result of the calculations prior.



1.) L = X_1 * Weight = ______

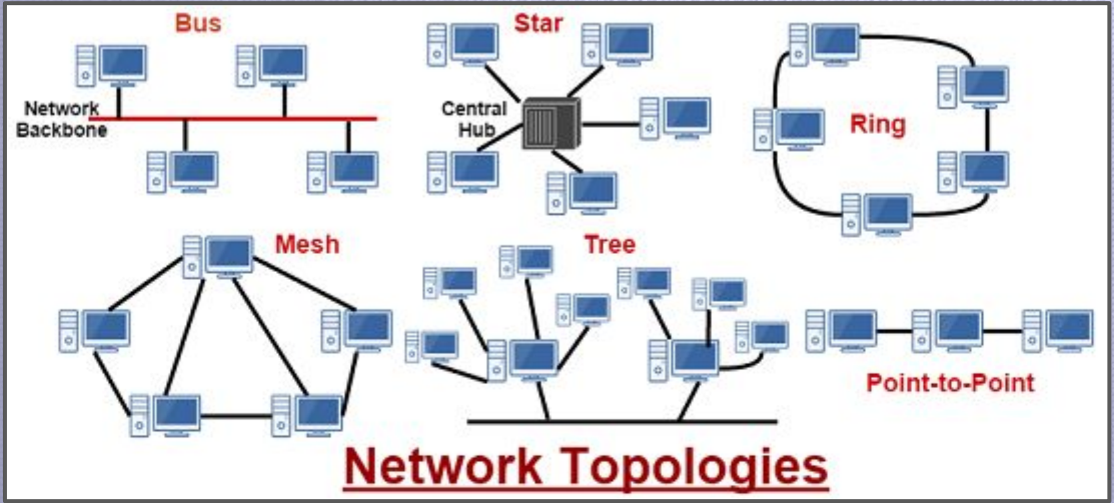
2.) R = X_2 * Weight = ______

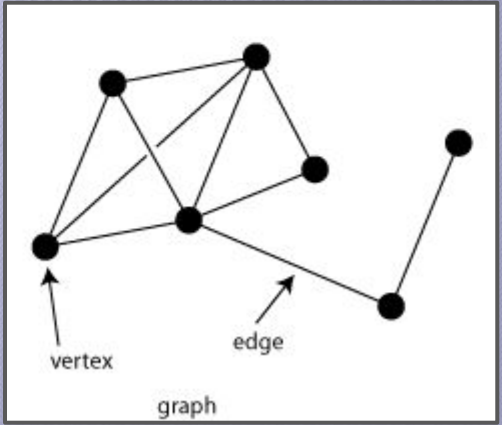
3.) Σ = L + R + **0** = ______

4.) $<(\Sigma)$ = ______

$$< (n \in \mathbb{R}) \begin{cases} 0 & n \leq 0 \\ n & n > 0 \end{cases}$$

Networks

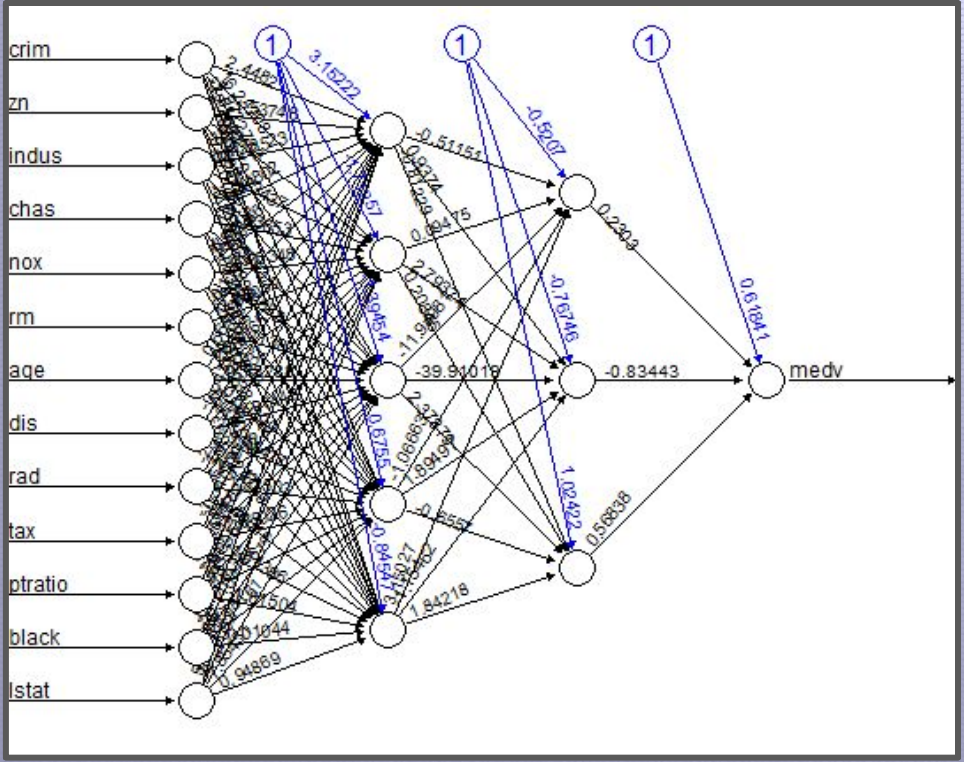




Example: computer networks

- Networks describe the connectivity of distinct things.
  - Nodes: The entities in the network. (Neurons)
  - Edges: The connections between the entities (Axons)
- Networks are modeled by mathematical structures called graphs.

Neural Networks



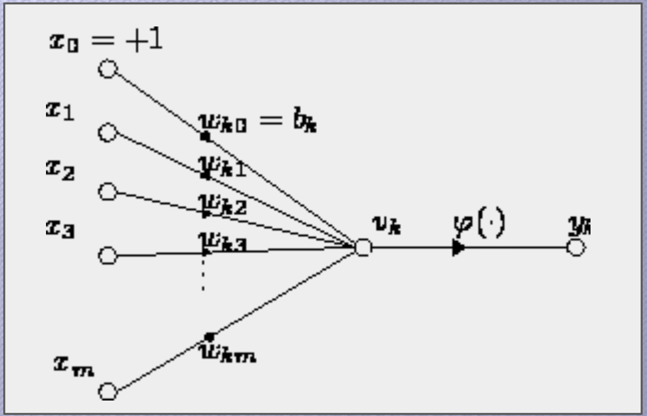
[SOURCE](#)

- Neural Networks are networks of neurons.
- These networks can be trained on data to produce useful representations that enable accurate predictions.
- Neurons become specialized during training. And groups of neurons take on a shared representation (localization).
- Neural networks provide a link between optimization and knowledge representation.
- Different neural architectures are suitable for different types of data modeling tasks.

“Either the universe is composable or God exists.”

-I heard Yann LeCun paraphrase this quote

Origin of Neural Networks

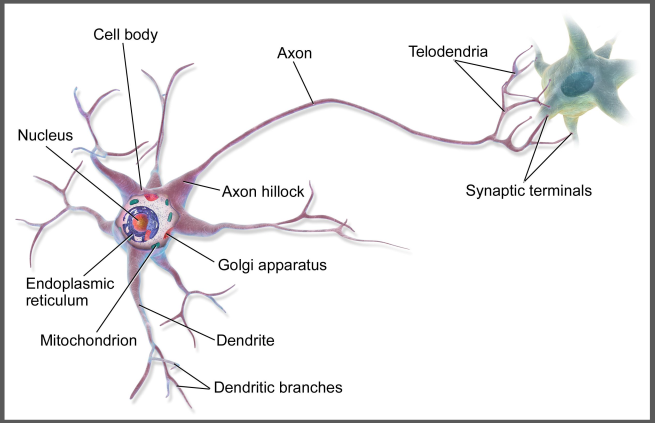


The Modern Connectionist Timeline

- 1943: Threshold Logic (McCulloch and Pitts)
- 1954: Hebbian Networks (Wesley A. Clark)
- 1958: Perceptrons (Frank Rosenblatt)
- 1969: AI Winter (Minsky and Papert)
- 1974: Multi-Layer Perceptrons and Backpropagation (Werbos)
- 1990: Convolutional Neural Networks (LeCun first runaway success)
- 1997: Long Short-Term Memory Networks (Hochreiter & Schmidhuber)
- 2014: Generative Adversarial Networks (Goodfellow et al.)

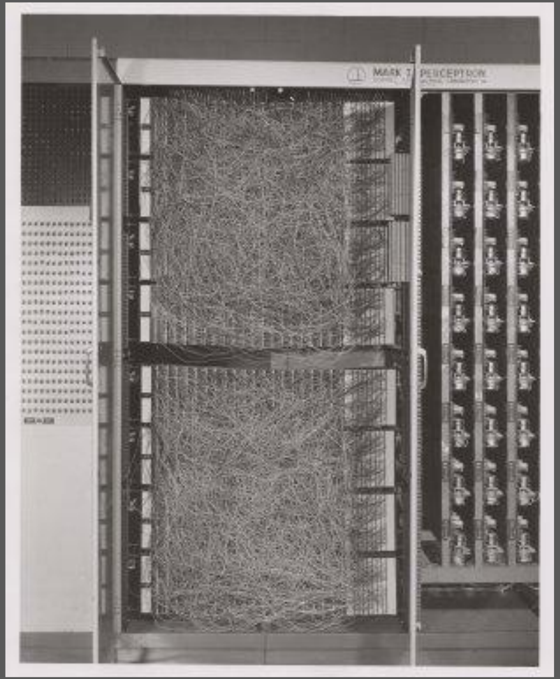
***An incomplete history**

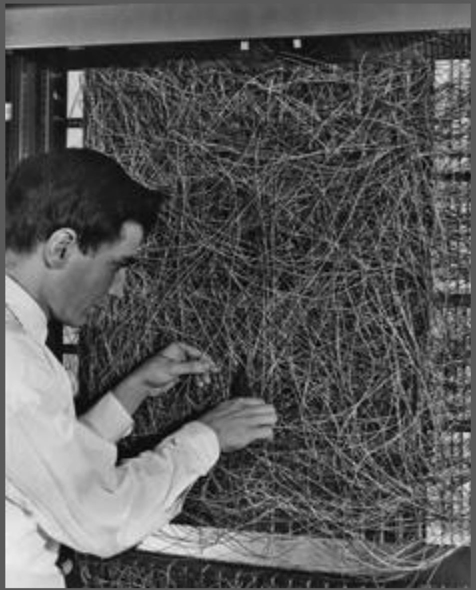
KEXP



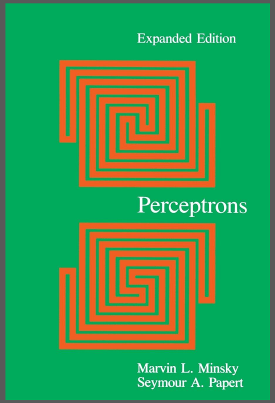
The Neuron

Biological Inspiration





Mark 1 Perceptron
Frank Rosenblatt



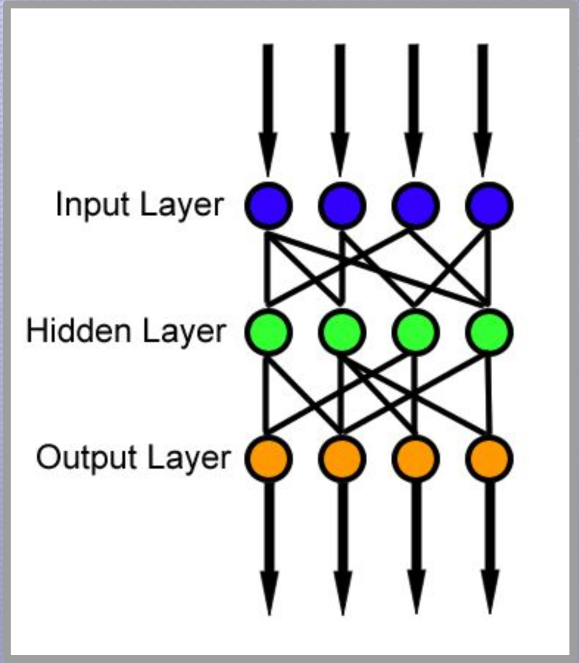
Harbingers of the AI Winter

Components of a Neural Network

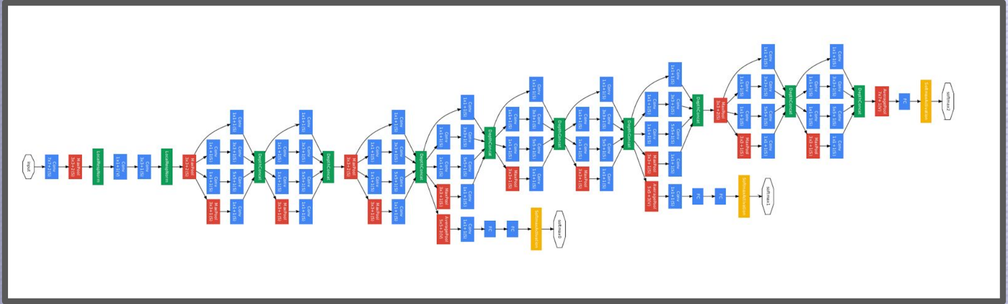
- **The Network Architecture**
  - How many neurons and layers a network has. How the neurons/layers are connected (Feedforward, Recurrent, Etc). What types of activations functions the neurons have.
- **The Objective/Loss function**
  - Given the last layer of a neural network and some targets for the output of the network, how well did the network perform?
- **The Network Gradient**
  - Backpropagation of errors from the cost function backwards through the network. This is really the chain rule of calculus.
- **The Optimization Algorithm**
  - Gradient Descent iteratively uses the network gradient, computed by backpropagation, to adjust the weights of the network. Adjusting these weights is called training or learning.

Stages of Training a Neural Network

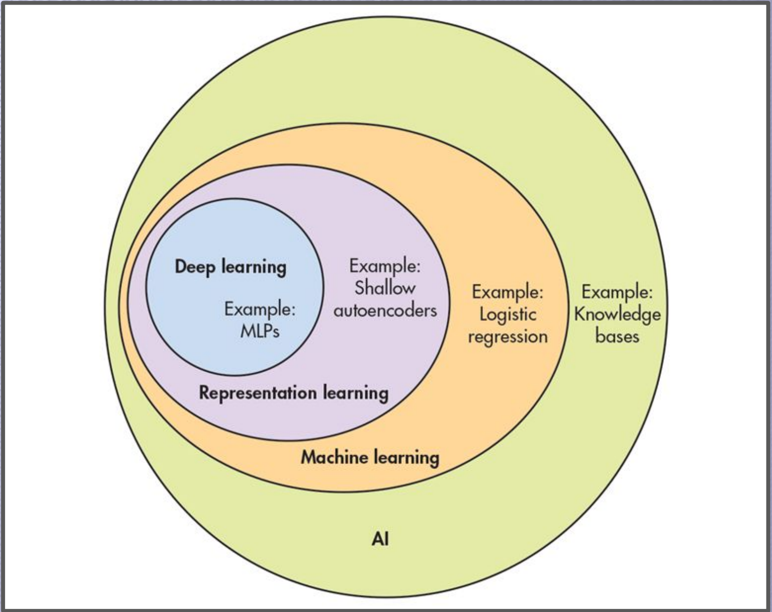
1. Data Preparation and Initialization
2. Training Loop - Training Set
    a. Forward Propagation of Training Batch
    b. Backpropagation of Errors on Training Targets
3. Evaluation
    a. Test Set
        i. Generalization of Training to the Test Set
    b. Validation Set
        i. Hyperparameter Optimization



Deep Learning

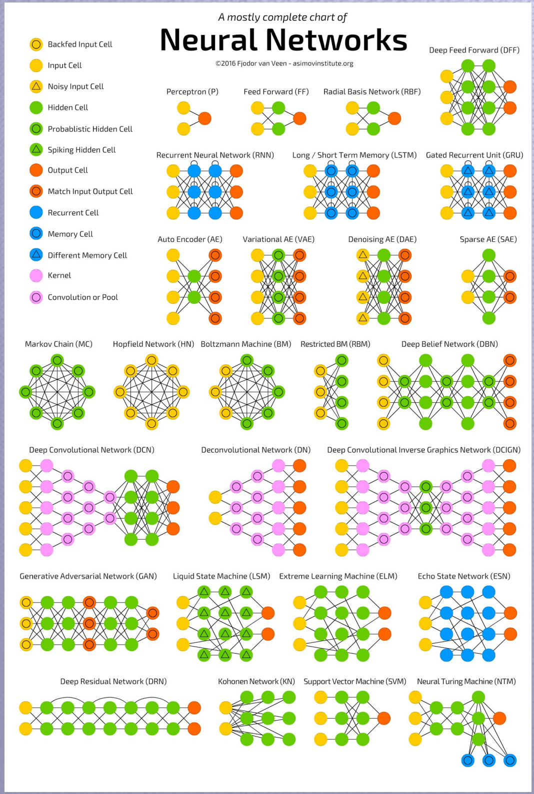


GoogleNet



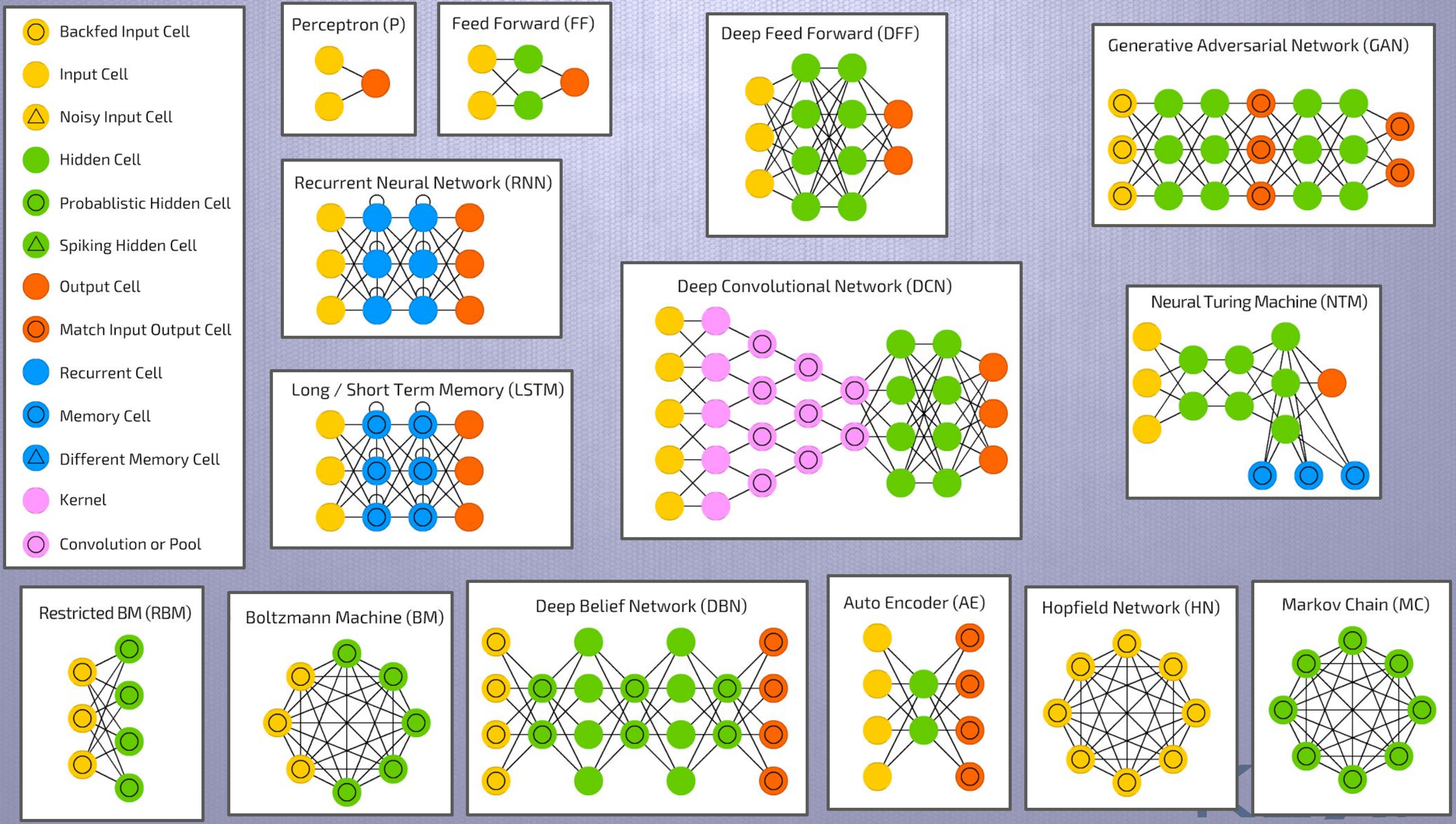
- Deep Learning refers to the observation that neural networks with many layers -- this is the deep part,-- are easier to train and generalize better than shallow neural networks.
- The term is also used to encapsulate modern neural network research.

Types of Neural Networks



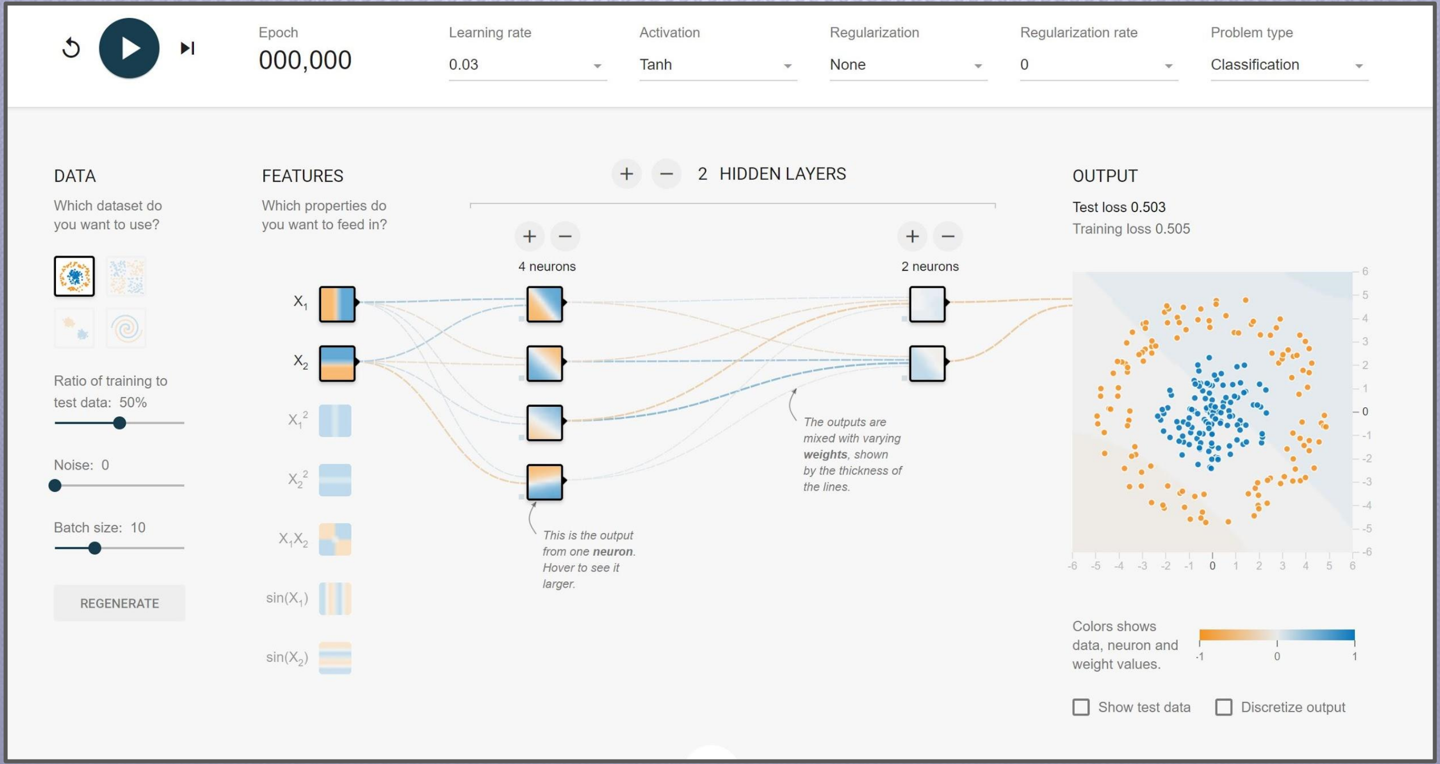
The number of neural network is only limited by our creativity.

Some have well known properties that make them useful for different tasks.



Backfed Input Cell
Input Cell
Noisy Input Cell
Hidden Cell
Probablistic Hidden Cell
Spiking Hidden Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Different Memory Cell
Kernel
Convolution or Pool
Perceptron (P)
Feed Forward (FF)
Deep Feed Forward (DFF)
Generative Adversarial Network (GAN)
Recurrent Neural Network (RNN)
Long / Short Term Memory (LSTM)
Deep Convolutional Network (DCN)
Neural Turing Machine (NTM)
Restricted BM (RBM)
Boltzmann Machine (BM)
Deep Belief Network (DBN)
Auto Encoder (AE)
Hopfield Network (HN)
Markov Chain (MC)

Neural Network Playground



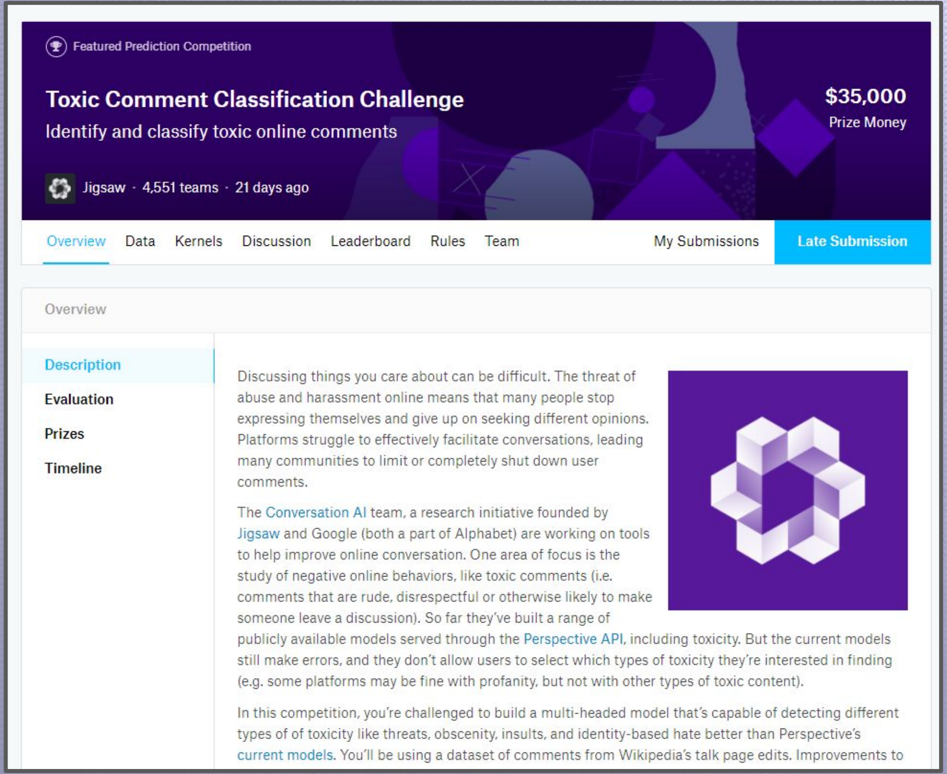
http://playground.tensorflow.org

The Kaggle Platform

About Kaggle

- Kaggle is a competitive data science platform and community for solving real world data driven problems.
- Founded in April 2010, acquired by google in 2017.
- Kaggle has been steadily evolving to become a platform for doing data science.
- Has hosted 200 data science competitions since its inception.

Anatomy of a Competition



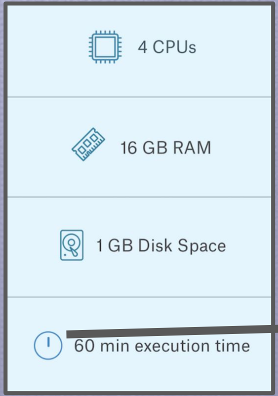
Competition Example

Working with Kernels

[Now With GPU Support](#)

Note: All Kaggle Kernels operate in a sandboxed environment without internet connectivity. This restricts what you can do with them to some degree. See google's [Colaboratory](#) for a more robust hosted notebook environment.

- Kaggle provides a hosted web-based environment for you to do analysis without having to download any data or any tools. Snippets and notebooks produced in this environment are called *Kaggle Kernels*.
- In this section we will go on Kaggle and look at how to work with the Kaggle Kernel system.
    - How to upload data, files, and pre-trained models to Kernels.
    - How to work with multiple data sources.
    - How to submit scores directly from a kernel.
    - How to output files from a Kernel.
    - How to add packages.
    - How to enable GPU support.
- Examples:
    - [Visualizing iMaterialist Data](#)
    - [Recovering the Videos](#)



[6 Hours](#)

Neural Networks in Python

Neural Network Frameworks

- Neural network frameworks allow us to develop neural networks at a conceptual level without worrying about low level details like GPU optimization, utilizing cluster computing, or computing the gradient of the neural network.
- Libraries like Tensorflow are very powerful, but sometimes they can be daunting for newcomers due to their complexity.
- Keras provides an even higher level api to various neural network frameworks and makes it much easier to begin experimenting with your own neural networks.

PYTORCH

KEXP

Hands On https://www.kaggle.com/c/dai-june/

In this workshop we will use neural networks to produce models of four different datasets in order to provide a glimpse into the different ways neural networks can be used.

1. The Titanic Disaster: How to use a *Feed-Forward Neural Network* to predict the survivors of the Titanic based on categorical (gender, ticket class, etc) and quantitative (age, ticket price, etc) features.

2. MNIST: How to use a *Feed-Forward Neural Network* to classify well preprocessed images of handwritten digits.

3. Street View House Numbers: How to use a *Convolutional Feed-Forward Neural Network* to classify noisy real world images of google street view house numbers.

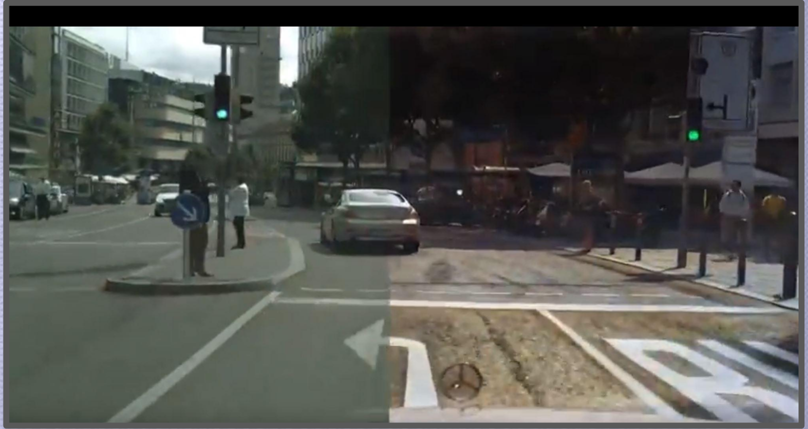
4. Jigsaw Toxic Comment Classification Challenge: How to use a *Long Short-Term Memory Network* to classify wikipedia comments.

We are also hosting an educational kaggle competition with this workshop to help you learn by solving a challenging problem. We will keep our notebooks collected under this competition also. See the link above.

KEXP

Deep Learning Applications

Image-to-Image Translation - GANs



VIDEO

VIDEO

Video

VIDEO

- CycleGan - https://junyanz.github.io/CycleGAN/
- Pix2Pix Demo: https://affinelayer.com/pixsrv/
- Pix2Pix HD: https://tcwang0509.github.io/pix2pixHD/

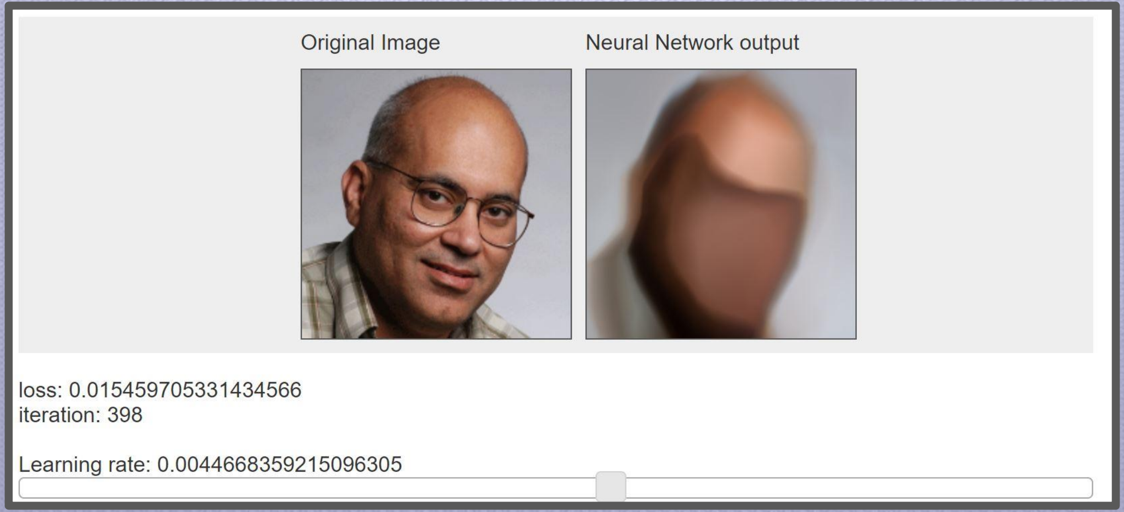
Image Synthesis - GANs

Video | Repo

Video

- https://www.csail.mit.edu/news/creating-videos-future
- http://carlvondrick.com/tinyvideo/

Image Painting - Autoencoders

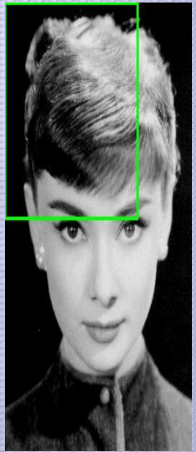


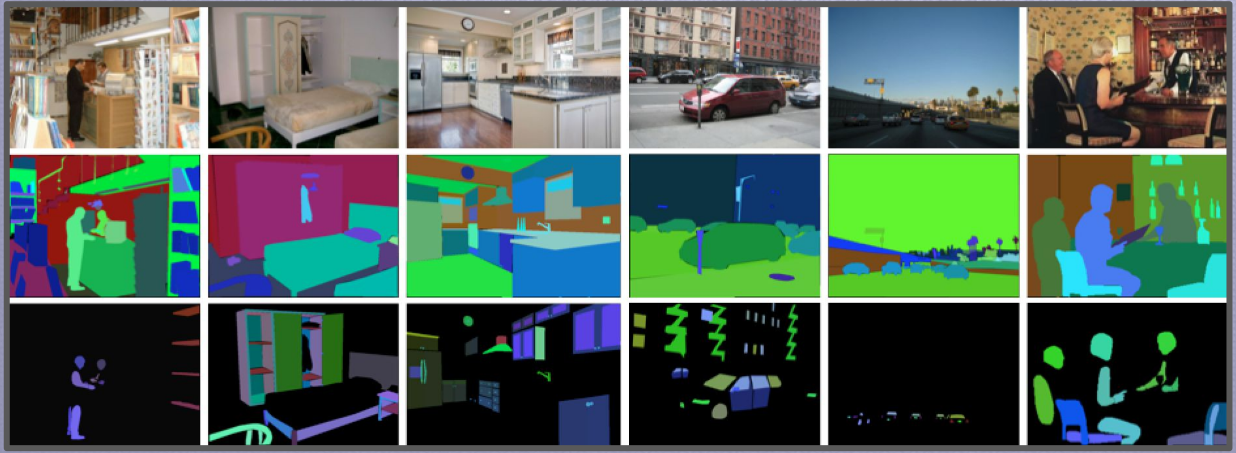
- Input: (x,y) position. Output: (r,g,b) color value.
- The network tries to learn a neural representation that produces the original image. This is similar in nature to an autoencoder.

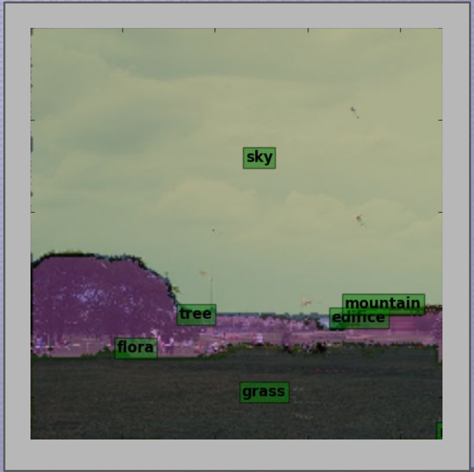
Source: https://cs.stanford.edu/people/karpathy/convnetjs/demo/image_regression.html

Segmentation - CNNs

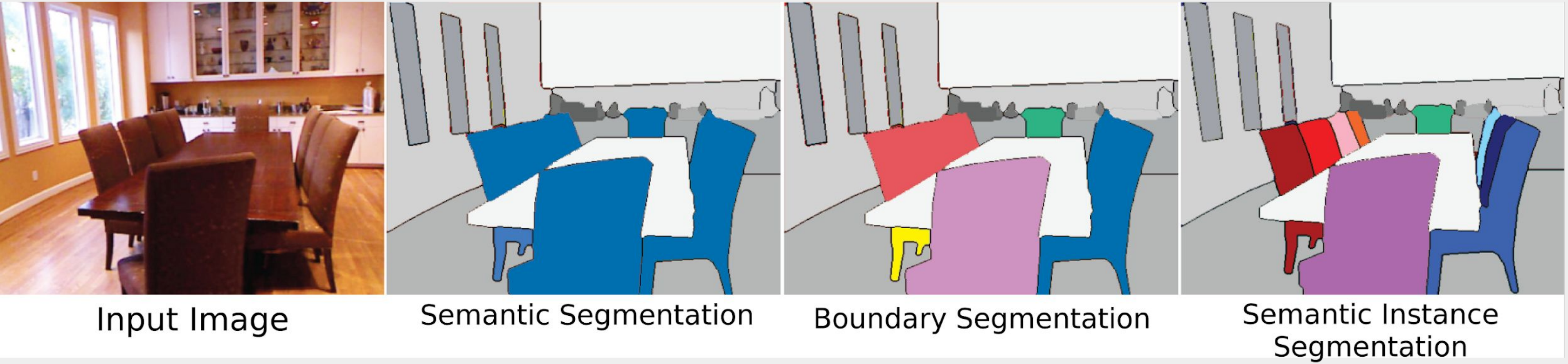
- Semantic Segmentation - Pixel-Wise Labeling
- Boundary Segmentation - Cohesive Regions
- Instance Segmentation - Countable Regions







Object classes: chair, table, window, etc...



Source: http://cs.nyu.edu/~silberman/images/coverage.png

Style Transfer - CNNs

A Neural Algorithm of Artistic Style - https://arxiv.org/abs/1508.06576

content loss: 1.22706e+06
style loss: .659507
total loss: 1.89246e+06

Style

Content

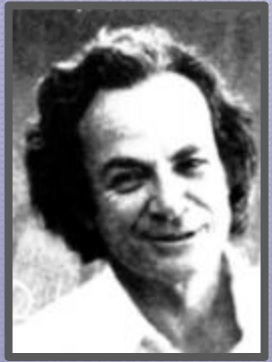
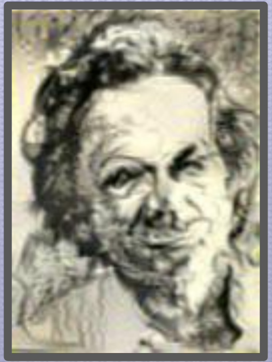
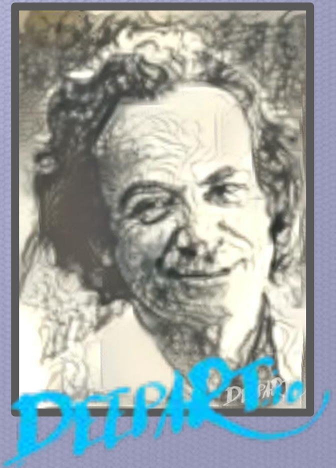


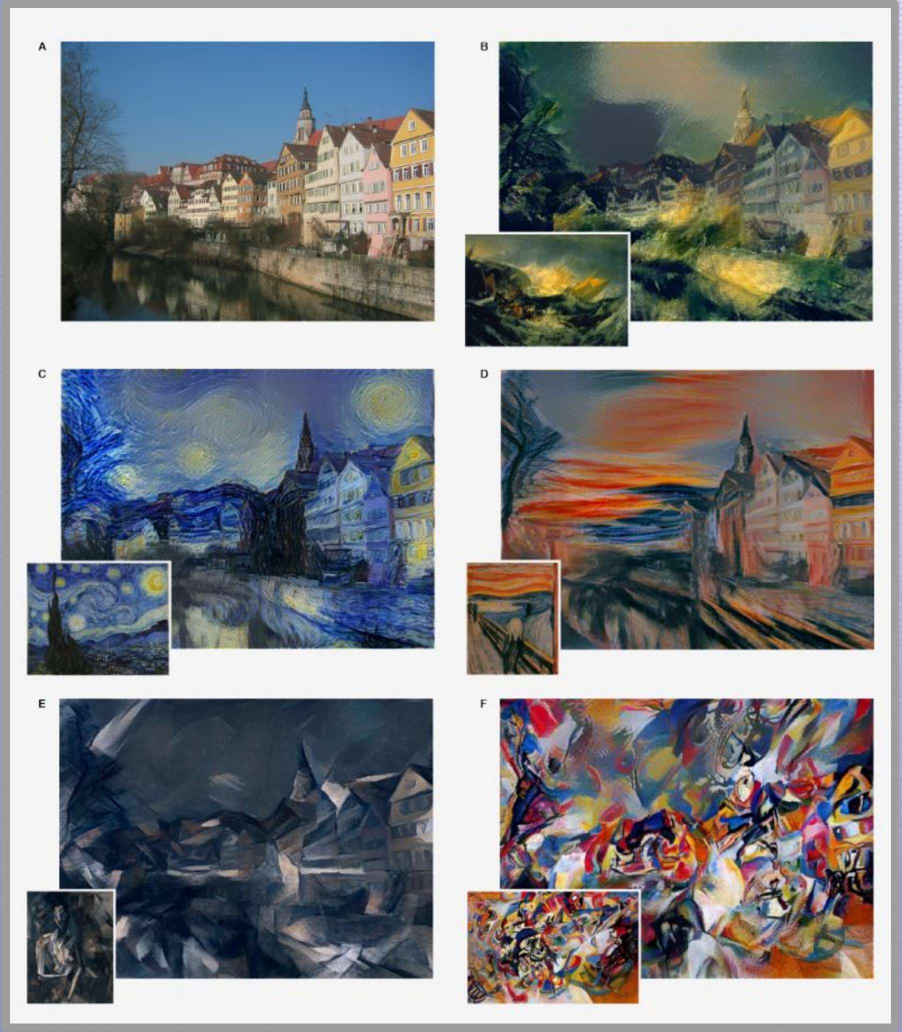
Image Quilting

Neural

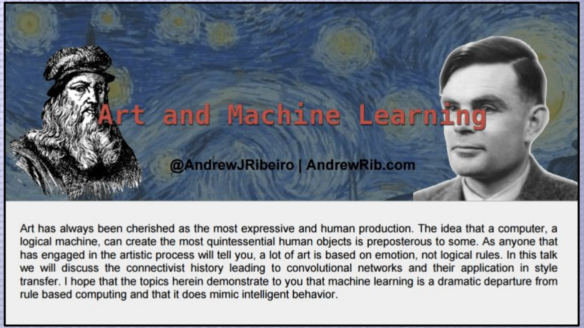




Note: I took a screenshot from the paper to get the style,content, and image quilting result images. They were probably scaled down in the paper so we didn't get great results, but it's still illustrative.



DEMO



Slides

https://www.youtube.com/watch?v=Khuj4ASldmU

KEXP

Sequence Modeling: Recurrent Neural Networks

- Handwriting Generation
- Machine Translation and Language Modeling
- Autocomplete
- Automated Image Captioning W/ Attention Modeling
- Audio Synthesis (Text-to-speech)
- Voice Recognition
- Conversational chat-bots

Handwriting Generation

https://distill.pub/2016/handwriting/

Audio Synthesis - Wavenet

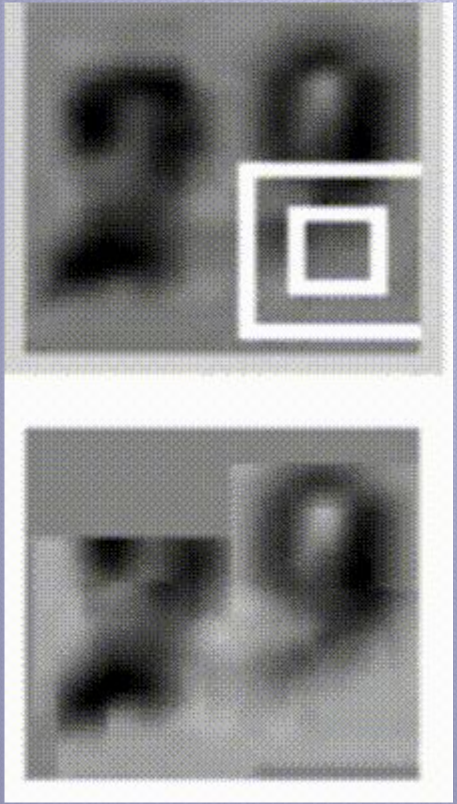
WaveNet: A Generative Model for Raw Audio

This post presents **WaveNet**, a deep generative model of raw audio waveforms. We show that WaveNets are able to generate speech which mimics any human voice and which sounds more natural than the best existing Text-to-Speech systems, reducing the gap with human performance by over 50%.

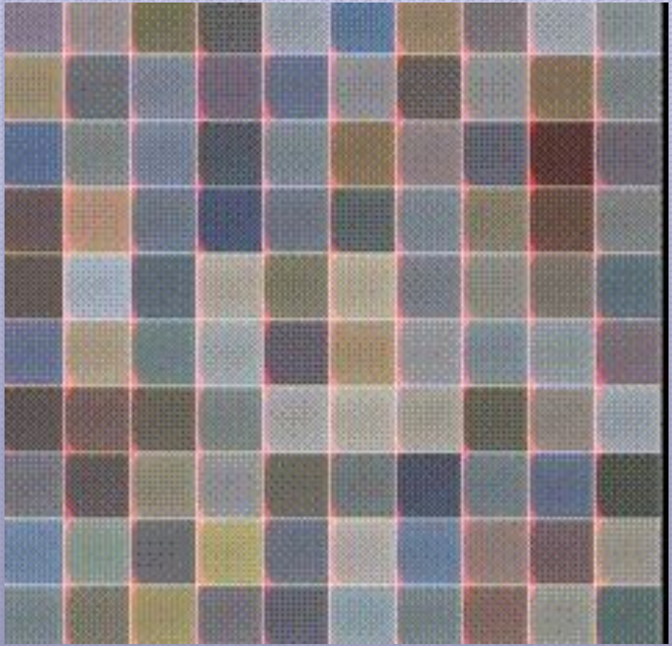
We also demonstrate that the same network can be used to synthesize other audio signals such as music, and present some striking samples of automatically generated piano pieces.

https://deepmind.com/blog/wavenet-generative-model-raw-audio/

Applications of RNN: Visual Attention



RNN learns to read house numbers.



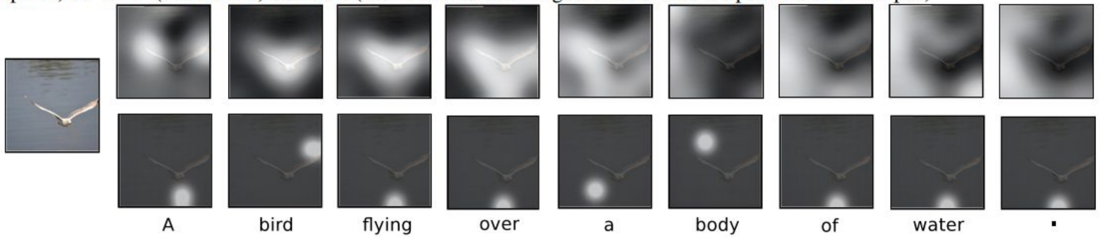
RNN learns to paint house numbers.

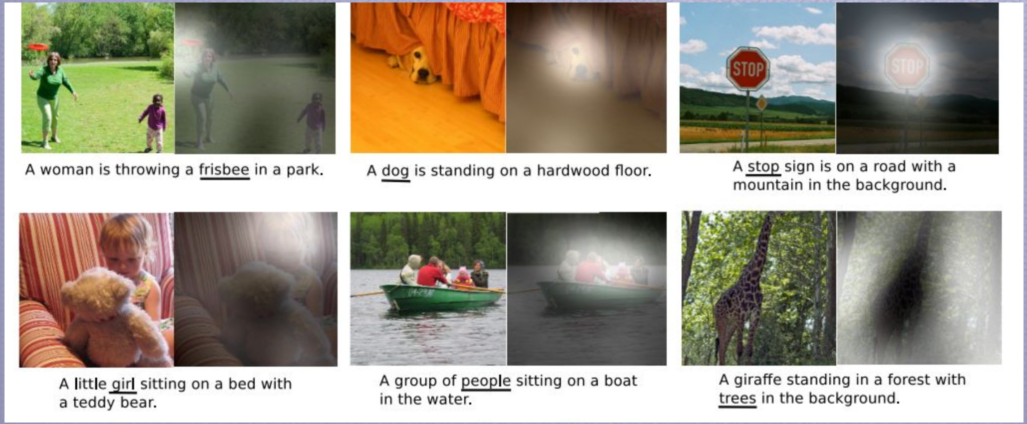
- The way humans perceive an image can be considered a time series problem; namely, where our focus changes over time.
- There has been research done on connecting RNN and CNN for image captioning. The idea here is that an RNN figures out the attention pattern and a CNN figures out what the objects are. This is referred to as an ensemble method.
- Attention is considered to be one of the big uses of RNN.

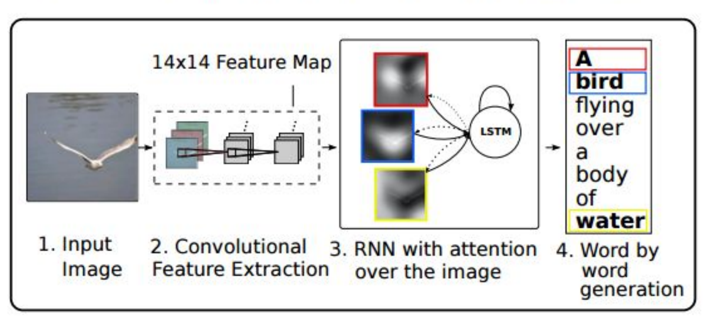
Source.

Applications of RNN: Image Captioning

Figure 2. Attention over time. As the model generates each word, its attention changes to reflect the relevant parts of the image. "soft" (top row) vs "hard" (bottom row) attention. (Note that both models generated the same captions in this example.)







[Source]

Applications of RNN: Language Modeling

PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.

Shakespeare

For $\bigoplus_{n=1,\dots,m}$ where $\mathcal{L}_{m_\bullet} = 0$, hence we can find a closed subset $\mathcal{H}$ in $\mathcal{H}$ and any sets $\mathcal{F}$ on X, U is a closed immersion of S, then $U \to T$ is a separated algebraic space.

Proof. Proof of (1). It also start we get

$$S = \mathrm{Spec}(R) = U \times_X U \times_X U$$

and the comparicoly in the fibre product covering we have to prove the lemma generated by $\coprod Z \times_U U \to V$. Consider the maps M along the set of points Sch_{fppf} and $U \to U$ is the fibre category of S in U in Section, ?? and the fact that any U affine, see Morphisms, Lemma ??. Hence we obtain a scheme S and any open subset $W \subset U$ in $Sh(G)$ such that $\mathrm{Spec}(R') \to S$ is smooth or an

$$U = \bigcup U_i \times_{S_i} U_i$$

which has a nonzero morphism we may assume that f_i is of finite presentation over S. We claim that $\mathcal{O}_{X,x}$ is a scheme where $x, x', s'' \in S'$ such that $\mathcal{O}_{X,x'} \to \mathcal{O}'_{X',x'}$ is separated. By Algebra, Lemma ?? we can define a map of complexes $\mathrm{GL}_{S'}(x'/S'')$ and we win. □

To prove study we see that $\mathcal{F}|_U$ is a covering of $\mathcal{X}'$, and $\mathcal{T}_i$ is an object of $\mathcal{F}_{X/S}$ for $i > 0$ and $\mathcal{F}_p$ exists and let $\mathcal{F}_i$ be a presheaf of $\mathcal{O}_X$-modules on $\mathcal{C}$ as a $\mathcal{F}$-module. In particular $\mathcal{F} = U/\mathcal{F}$ we have to show that

$$\widetilde{M}^\bullet = \mathcal{I}^\bullet \otimes_{\mathrm{Spec}(k)} \mathcal{O}_{S,s} - i_X^{-1}\mathcal{F})$$

is a unique morphism of algebraic stacks. Note that

$$\mathrm{Arrows} = (Sch/S)^{opp}_{fppf}, (Sch/S)_{fppf}$$

and

$$V = \Gamma(S, \mathcal{O}) \longmapsto (U, \mathrm{Spec}(A))$$

is an open subset of X. Thus U is affine. This is a continuous map of X is the inverse, the groupoid scheme S.

Proof. See discussion of sheaves of sets. □

The result for prove any open covering follows from the less of Example ??. It may replace S by $X_{spaces,étale}$ which gives an open subspace of X and T equal to S_{Zar}, see Descent, Lemma ??. Namely, by Lemma ?? we see that R is geometrically regular over S.

Lemma 0.1. *Assume (3) and (3) by the construction in the description.*

Suppose $X = \lim |X|$ (by the formal open covering X and a single map $\underline{Proj}_X(\mathcal{A}) = \mathrm{Spec}(B)$ over U compatible with the complex

$$Set(\mathcal{A}) = \Gamma(X, \mathcal{O}_{X,\mathcal{O}_X}).$$

When in this case of to show that $\mathcal{Q} \to \mathcal{C}_{Z/X}$ is stable under the following result in the second conditions of (1), and (3). This finishes the proof. By Definition ?? (without element is when the closed subschemes are catenary. If T is surjective we may assume that T is connected with residue fields of S. Moreover there exists a closed subspace $Z \subset X$ of X where U in X' is proper (some defining as a closed subset of the uniqueness it suffices to check the fact that the following theorem

(1) f is locally of finite type. Since $S = \mathrm{Spec}(R)$ and $Y = \mathrm{Spec}(R)$.

Proof. This is form all sheaves of sheaves on X. But given a scheme U and a surjective étale morphism $U \to X$. Let $U \cap U = \coprod_{i=1,\dots,n} U_i$ be the scheme X over S at the schemes $X_i \to X$ and $U = \lim_i X_i$. □

The following lemma surjective restrocomposes of this implies that $\mathcal{F}_{x_0} = \mathcal{F}_{x_0} = \mathcal{F}_{X,\dots,0}$.

Lemma 0.2. *Let X be a locally Noetherian scheme over S, $E = \mathcal{F}_{X/S}$. Set $\mathcal{I} = \mathcal{J}_1 \subset \mathcal{I}'_n$. Since $\mathcal{I}^n \subset \mathcal{I}^n$ are nonzero over $i_0 \leq \mathfrak{p}$ is a subset of $\mathcal{J}_{n,0} \circ \overline{A}_2$ works.*

Lemma 0.3. *In Situation ??. Hence we may assume $\mathfrak{q}' = 0$.*

Proof. We will use the property we see that $\mathfrak{p}$ is the mext functor (??). On the other hand, by Lemma ?? we see that

$$D(\mathcal{O}_{X'}) = \mathcal{O}_X(D)$$

where K is an F-algebra where δ_{n+1} is a scheme over S. □

Latex

Source

RNN Shakespeare

- 3-layer RNN with 512 hidden nodes on each layer.
- Trained on all the works of Shakespeare concatenated into a single (4.4MB) file.
- Source: http://karpathy.github.io/2015/05/21/rnn-effectiveness/

PANDARUS:
Alas, I think he shall be come approached and the day
When little srain would be attain'd into being never fed,
And who is but a chain and subjects of his death,
I should not sleep.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO:
Well, your wit is in the care of side and that.

Second Lord:
They would be ruled after this chamber, and
my fair nues begun out of the fact, to be conveyed,
Whose noble souls I'll have the heart of the wars.

Clown:
Come, sir, I will make did behold your worship.

VIOLA:
I'll drink it.

Harry Potter: Written by Artificial Intelligence

LSTM Trained on First Four Harry Potter Books

“The Malfoys!” said Hermione.

Harry was watching him. He looked like Madame Maxime. When she strode up the wrong staircase to visit himself.

“I’m afraid I’ve definitely been suspended from power, no chance—indeed?” said Snape. He put his head back behind them and read groups as they crossed a corner and fluttered down onto their ink lamp, and picked up his spoon. The doorbell rang. It was a lot cleaner down in London.

Hermione yelled. The party must be thrown by Krum, of course.

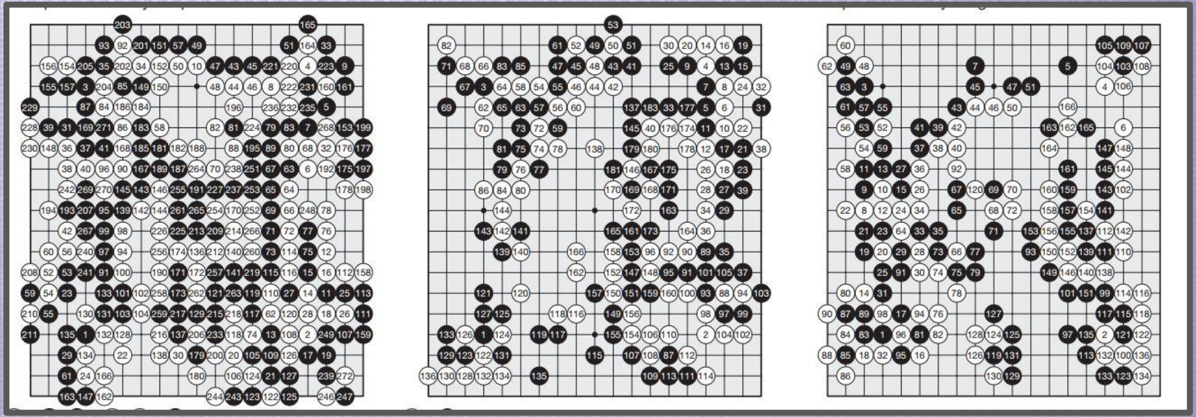
Harry collected fingers once more, with Malfoy. “Why, didn’t she never tell me. …” She vanished. And then, Ron, Harry noticed, was nearly right.

“Now, be off,” said Sirius, “I can’t trace a new voice.”

Source: https://medium.com/deep-writing/harry-potter-written-by-artificial-intelligence-8a9431803da6

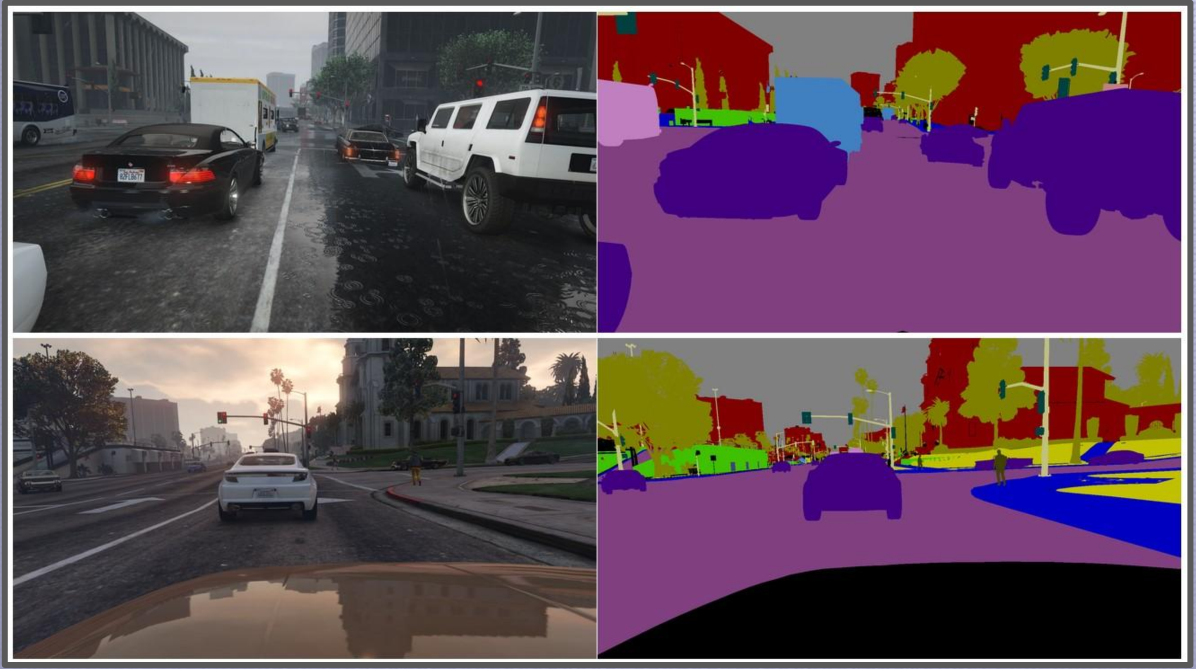
AlphaGo

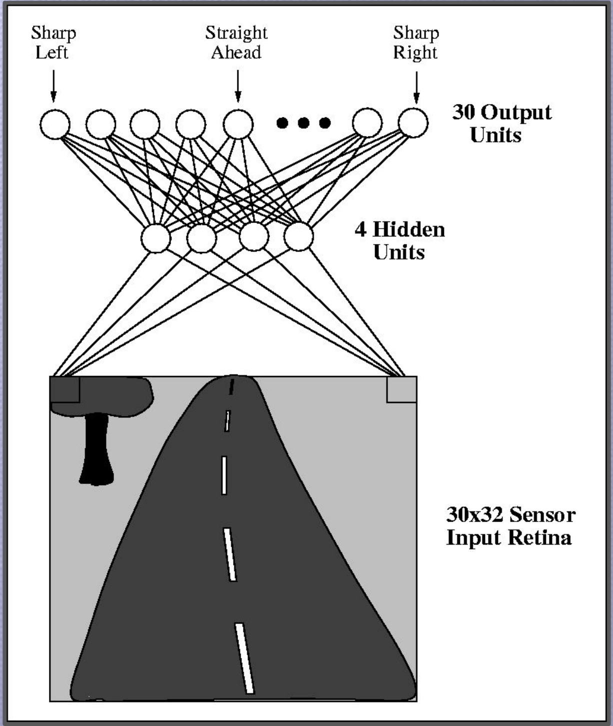
- Created by DeepMind, AlphaGo has become the first superhuman Go player in history.
- The ancient game of Go is combinatorially explosive. A computer simply cannot use brute force to calculate the best moves.



- “AlphaGo used two deep neural networks:
  - A policy network that outputs move probabilities.
  - A value network that outputs a position evaluation.
  - The policy network was trained initially by supervised learning to accurately predict human expert moves, and was subsequently refined by policy-gradient reinforcement learning”
- The knowledge AlphaGo learned about the game acts as a set of heuristics through the combinatorial space of possible plays.

Autonomous Vehicles





AV simulator in Unity | LIDAR gives us much more data

T-shirt Question

Andrew Ng | What do you call a group of variables that go to the gym? - View this email in your browser AI is the new electricity Machine Lea | 5:03 pm

What do you call a group of variables that go to the gym?

Thanks for coming!

Discussion & Wrap Up

Questions for me?

Questions for you:

- Can you see AI being used in your industry? How so?
- How much covered here did you know already?
- What about AI interests you? Why did you come today?
- What would you like to see in future talks about AI?

Andrewnetwork@gmail.com
Andrew@kexp.io
Twitter: @AndrewJRibeiro
https://github.com/Andrewnetwork

@kexpsocial
https://github.com/k-exp
WWW.KEXP.IO

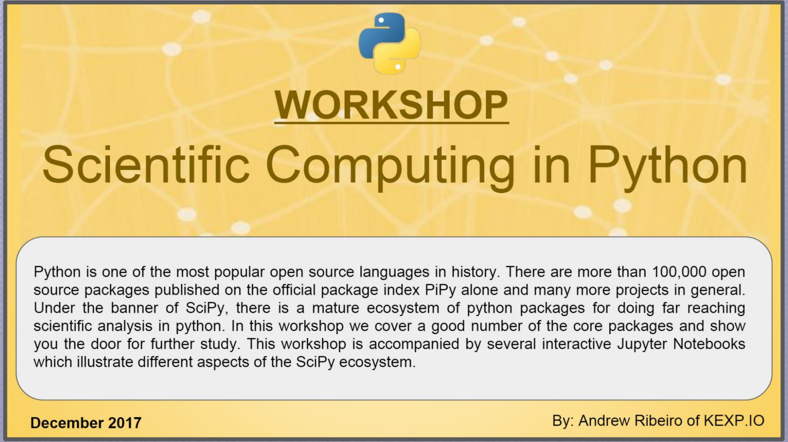
KEXP
KNOWLEDGE EXPLORATION SYSTEMS

By: Andrew Ribeiro of Knowledge-Exploration Systems

Resources and Sources

My SciPy Workshop

- Python is a great multi-paradigm language with tons of libraries for doing AI and scientific computing. Easy FFI also allows for high performance.
- Jupyter Notebooks are awesome! Combines prose with computation.
- Check out my repo for more resources:



https://github.com/Andrewnetwork/WorkshopScipy

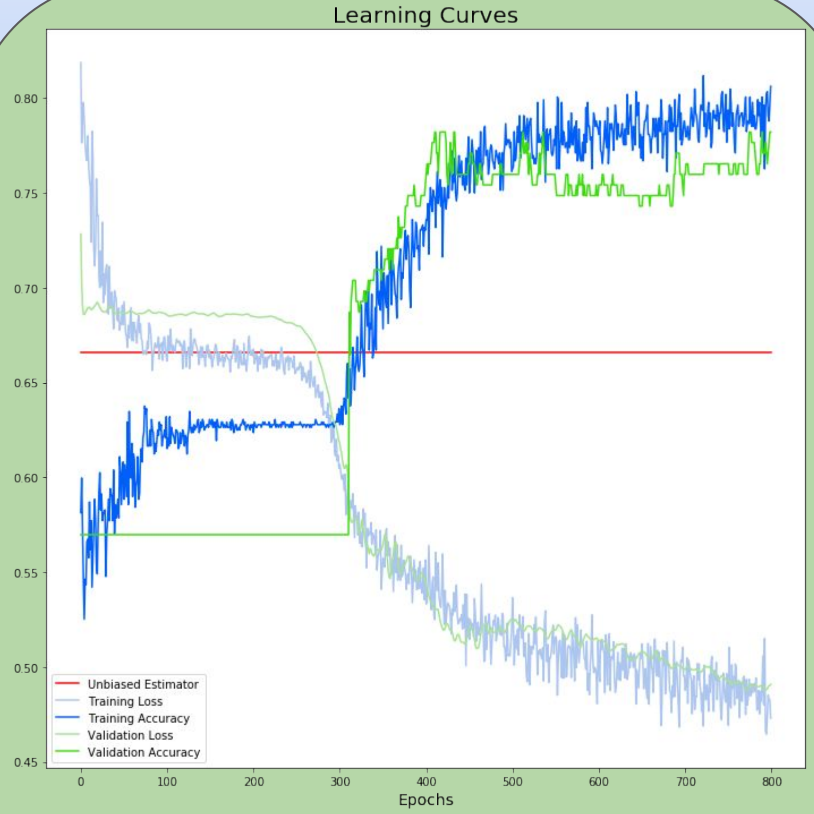
Courses

- (Start here) Machine Learning. Andrew NG. Coursera.
- Artificial Intelligence. MIT. Winston. OCW.
- CS188: Intro to AI. Berkeley. Course Videos.
- CS231n: ConvNets for Visual Recognition. Stanford. Playlist. Course Notes.
- Neural Networks for Machine Learning. Hinton. Coursera.
- Probabilistic Graphical Models. Coursera.

Books

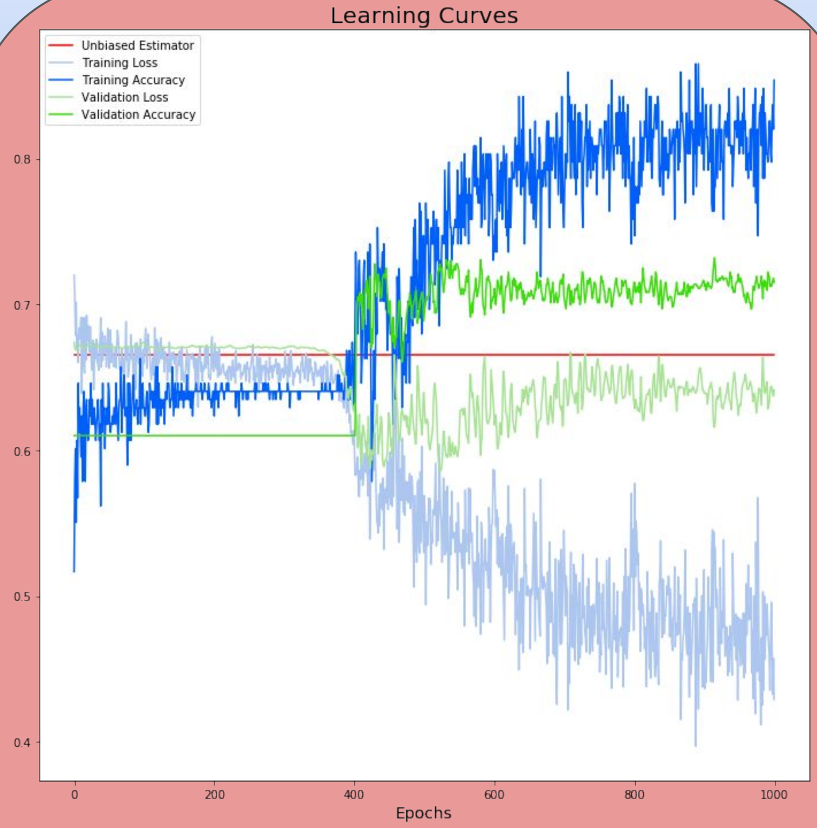
- (Start here) Artificial Intelligence: A Modern Approach. Norvig, Russell. Site.
- The Deep Learning Book. Goodfellow, Bengio,Courville. Free Book.
- Computer Vision: Models, Learning, and Inference. Prince. Free Book.
- Pattern Recognition and Machine Learning. Bishop. Amazon.
- Probabilistic Graphical Models. Koller, Friedman. Site.
- The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Amazon.

HEALTHY



We would expect this model to generalize well.

BAD



Validation and training set metrics diverge.

Andrew Ribeiro 2018