
PLURALSIGHT

Introduction to Artificial Intelligence

Welcome!

Greg Swanson
Instructor, Pluralsight

Agenda

- Today:
  - Unsupervised Learning
  - Deep Learning with Neural Networks
- The rest of the week:
  - Reinforcement Learning
  - Applications and Capstone Project



Today's Schedule

9:00 am - 12:00 pm PT Deep Learning with Neural Networks I

- Lecture
- Lab

9:50 am - 10:00 am PT Break

Lunch 12:00 - 1 PM PT

1:00 - 2:50 PM PT Deep Learning with Neural Networks II

- Lecture
- Lab

Break 2:50 - 3:00 pm PT

3:00 - 5:00 PM PT Intro to Reinforcement Learning

- Lecture
- Lab

How we're going to work together

- You'll have a copy of the course materials shortly
  - We'll be using Jupyter notebooks again

- You'll be following along in the notebook and..
  - doing coding exercises/labs inside the notebook as well

Today's Key Learning Objectives

- Develop intuition for how a Neural Network is trained and makes predictions
- Be able to build an example neural network with keras
- Define what reinforcement learning and its terminology
- Practice a reinforcement learning approach for a contextual bandits problem

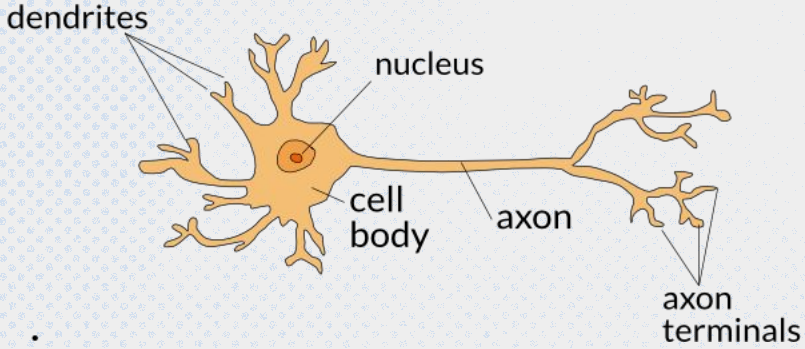
Neural Networks

Fast Facts

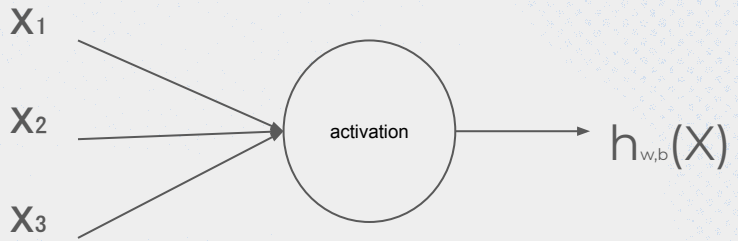
- Neural Nets have been around since the 1940s, but were not heavily used for their first 60 years
- Advancements in computer hardware & software for working with neural nets make them the pre-imminent machine learning approach today
- Applied to almost any machine learning task
- Can outperform humans for certain tasks previously thought impossible for a computer to do

Why are they called Neural Networks?

- While neural networks were first proposed by mathematicians, they get their name from their passing similarity to a neuron
- Neurons take multiple inputs, run them through an activation function, and pass an output:

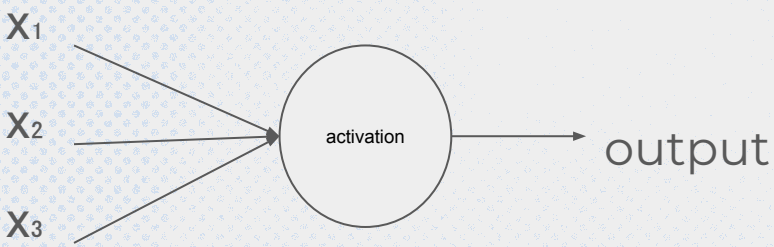


Perceptron



The Perceptron looks familiar, doesn't it?

- For the binary classification problem, the single perceptron looks very similar to our old friend, logistic regression:



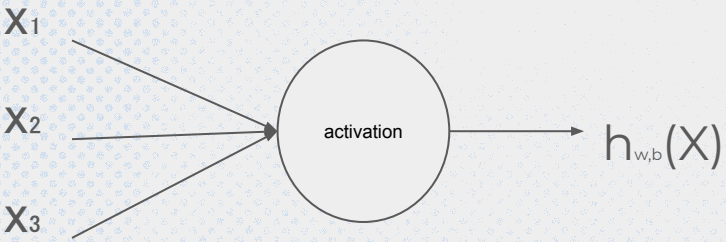
Perceptron

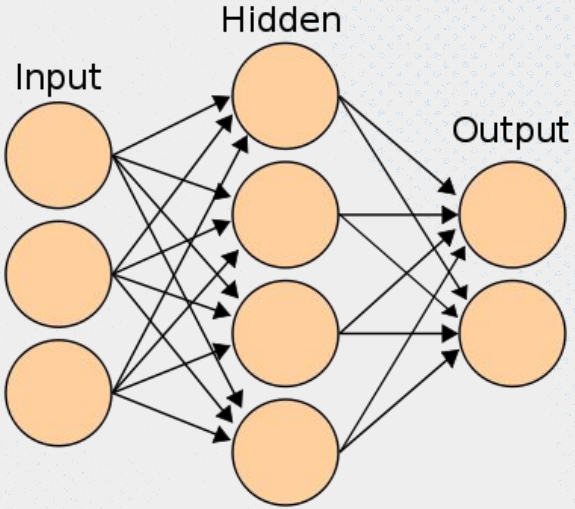
output
- 1 if $\frac{1}{1+e^{-(ax_1+bx_2+c)}} \geq 0.5$
- 0 if $\frac{1}{1+e^{-(ax_1+bx_2+c)}} < 0.5$

Network implies more than one, right?

- Similar to our brain, neural networks are made up of many neurons connected to each other in various ways
- Each Neural Network has at least 3 layers: input, hidden, and output
- You can choose # layers

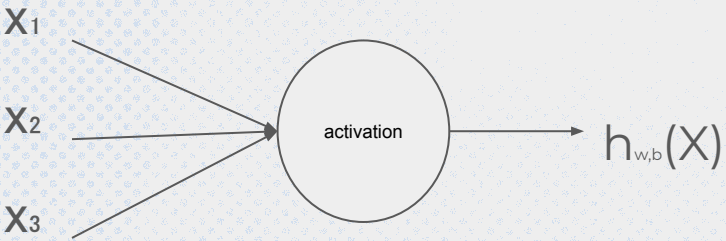
  & # perceptrons



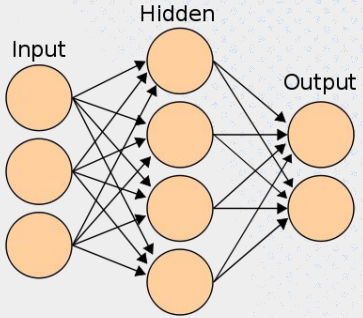


Layers, Nodes, and Capacity

- Capacity is the term for how complex a pattern or relationship a given model can express.
- It's closely related to the bias-variance tradeoff in that a model with low capacity could have high bias and be underfitting data and underperforming in the evaluation step
- Which model would have greater learning capacity?

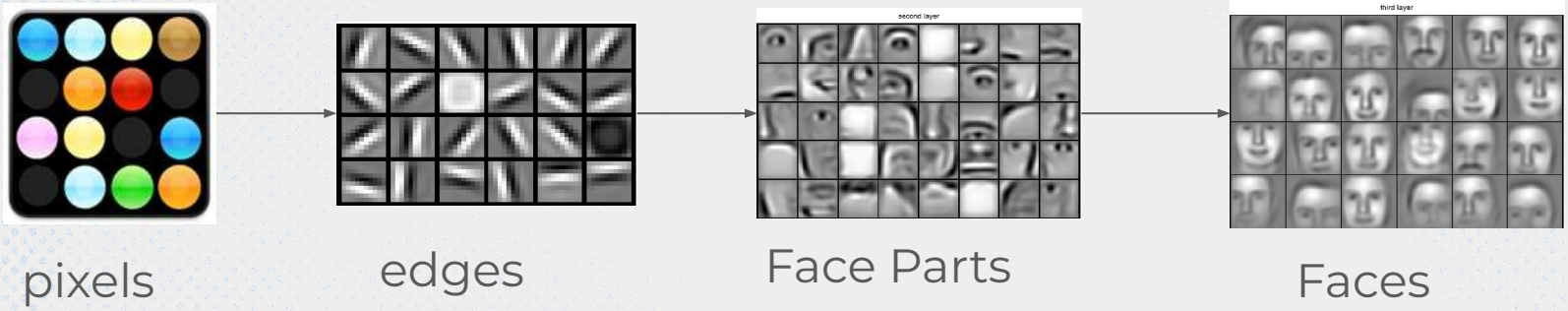


OR



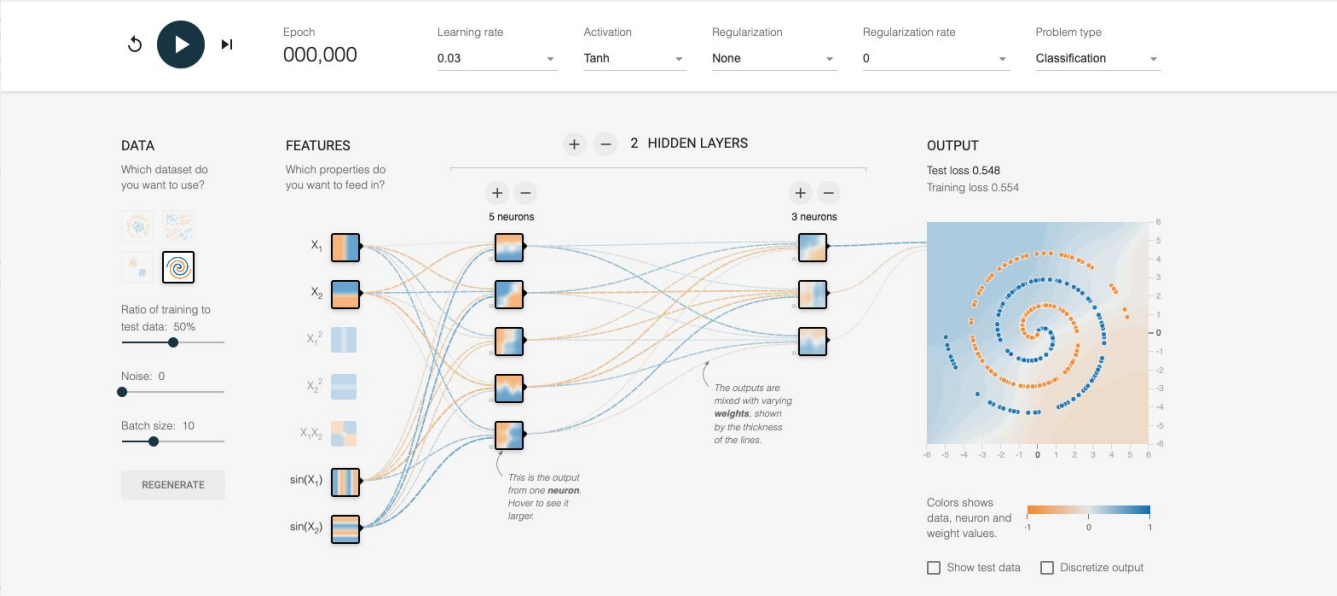
Why add layers?

You can think of the additional layers as finding other useful features from the interactions of your initial input layers



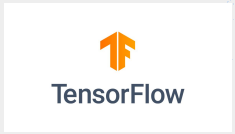
A Visual Example

- Go to [tensorflow playground](#) and illustrate a feed-forward neural network



Common Deep Learning Libraries

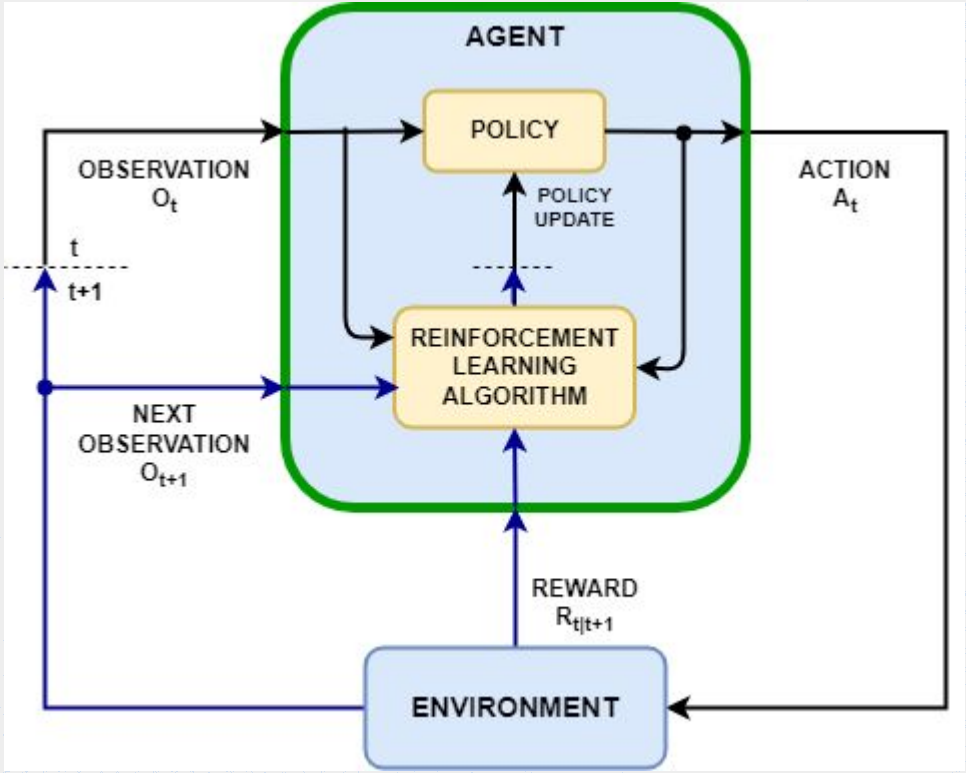
- Pytorch (meta / facebook) and tensorflow (google) are the 2 most popular deep learning frameworks for python
- Keras simplifies development and is therefore useful in our introductory course



Reinforcement Learning

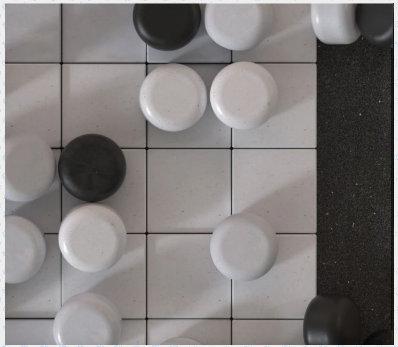
Reinforcement Learning: an area of machine learning that trains an agent to take a suitable action to maximize reward in a particular situation

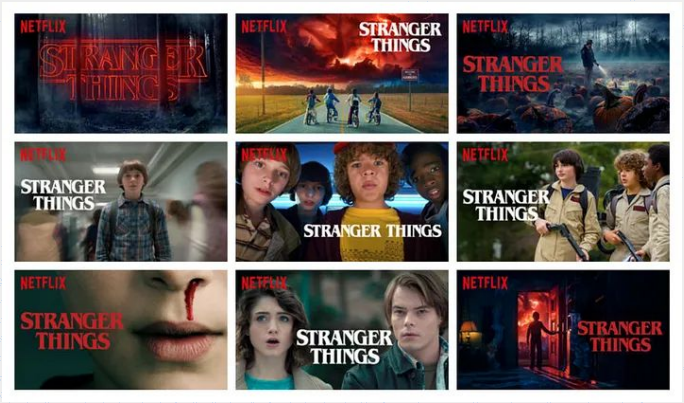
Agents?

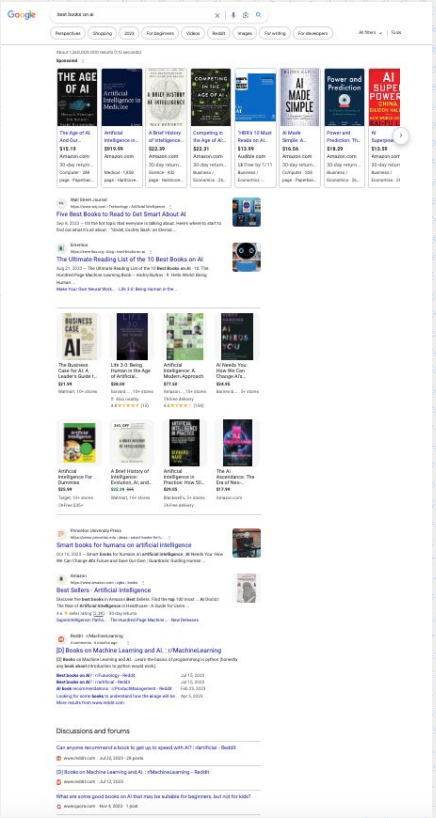


Examples of reinforcement learning

- Netflix Thumbnail Optimization
- New web pages in google search
- Alpha go







Today's Key Learning Objectives

- Develop intuition for how a Neural Network works
- Be able to build an example neural network with keras
- Define reinforcement learning and its terminology
- Practice a reinforcement learning approach for a contextual bandits problem

Questions?

Thank you!

If you have any additional questions, please ask!If