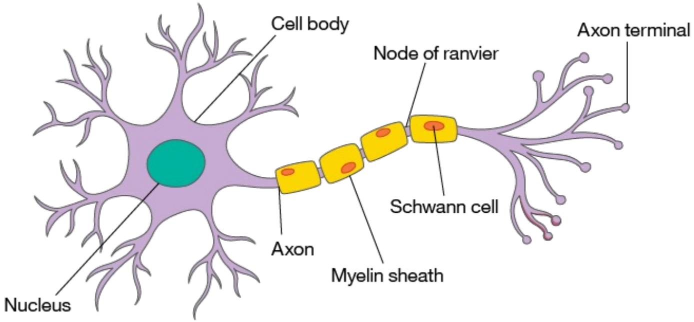
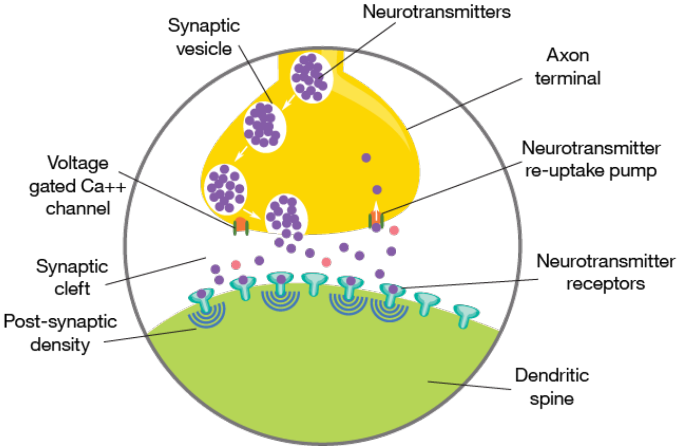
Impacts of AI: COMP3800-03
Deep Learning Explained?
Wentworth Institute of Technology

From biological neural networks to artificial neural networks

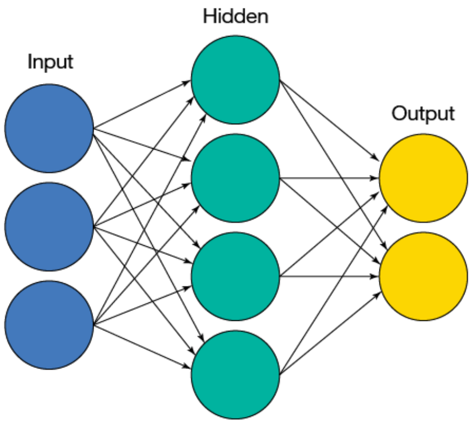
Biological neuron



Biological synapse

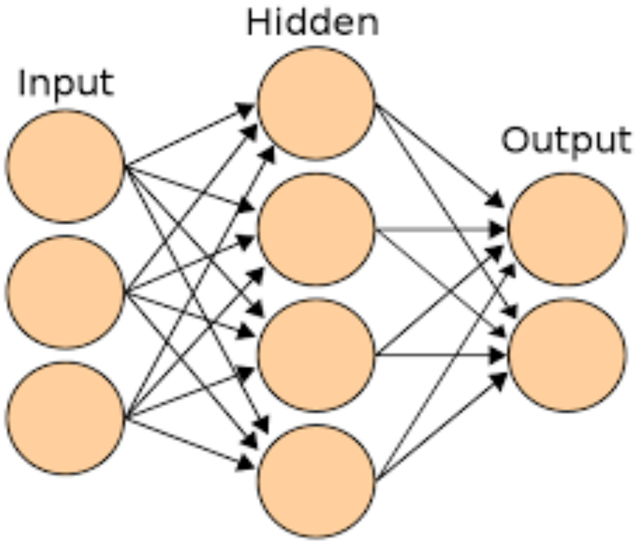


Artificial neural network



Brief History

- Research stalled in the 1950s because they couldn't make them with existing computers and mathematical models.
- In the late 1970s/early 1980s, improvements in computing speed and the development of the backpropagation algorithm reignited interest.
- By the mid 1980s, the multilayer perceptron with backpropagation emerges as a usable general-purpose machine learning mechanism.



Predicting your grades using neural nets

X	hrs study	hrs sleep	Grade	y
	7	5	78	
	6	8	93	
	8	2	67	
	5	5	?	$\hat{y}$

Supervised learning

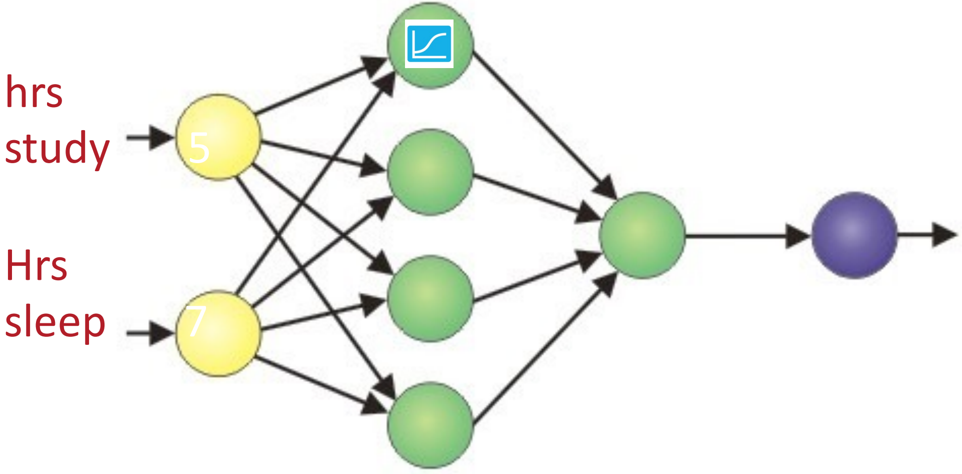
This is a regression problem

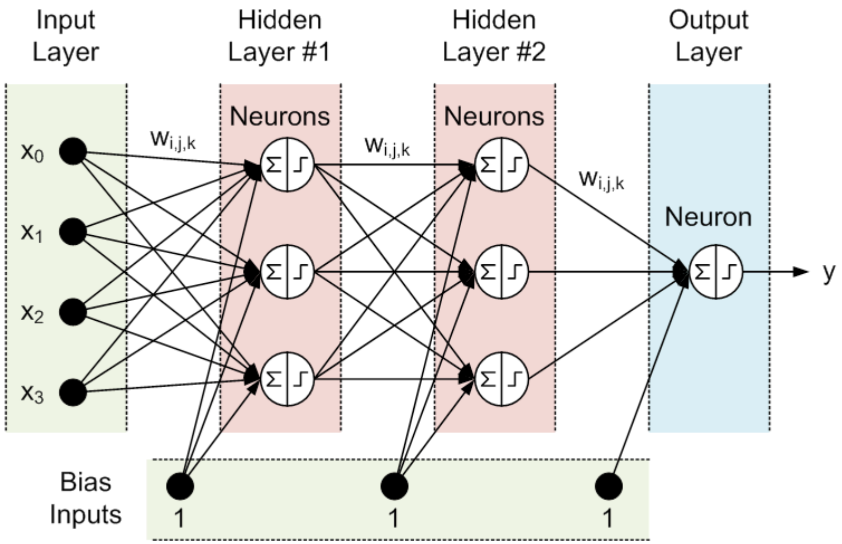
Not a classification problem

$$X_{norm} = x / \max(x)$$

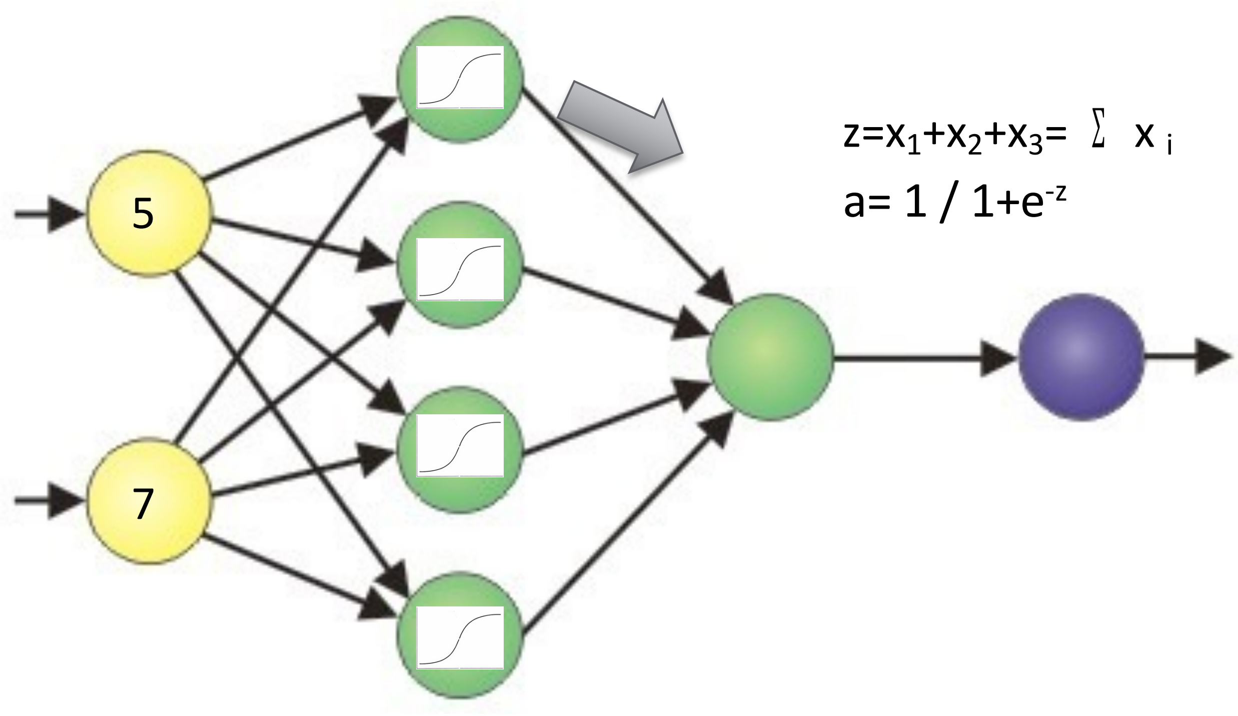
$$Y_{norm} = y / \max(y)$$

Basic structure of a neural net



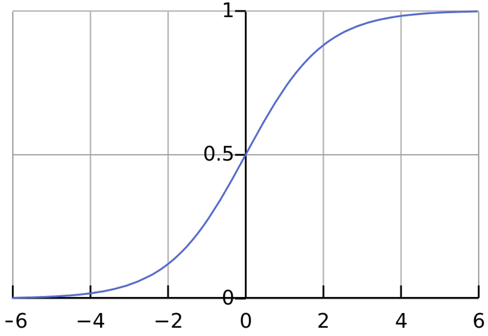


Sigmoid Operation

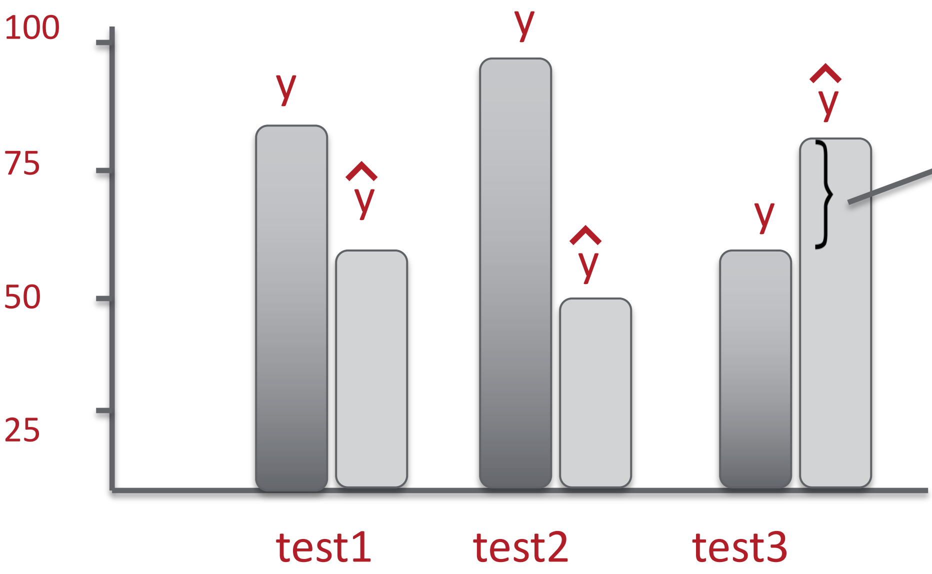


$z = x_1 + x_2 + x_3 = \sum x_i$

$a = 1 / 1 + e^{-z}$



Training neural network = minimizing cost function

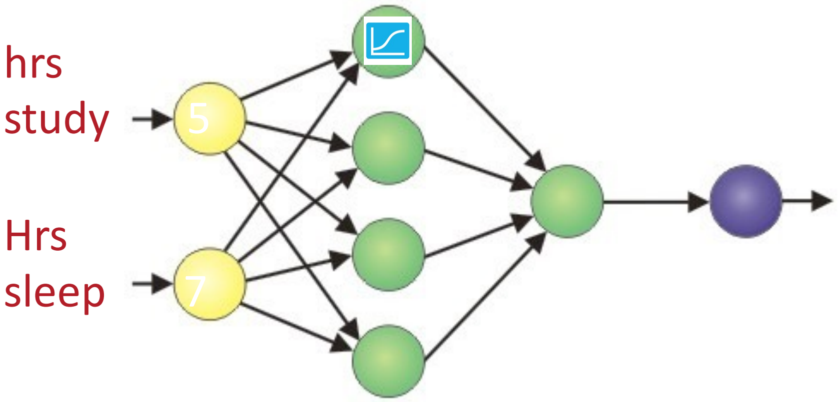


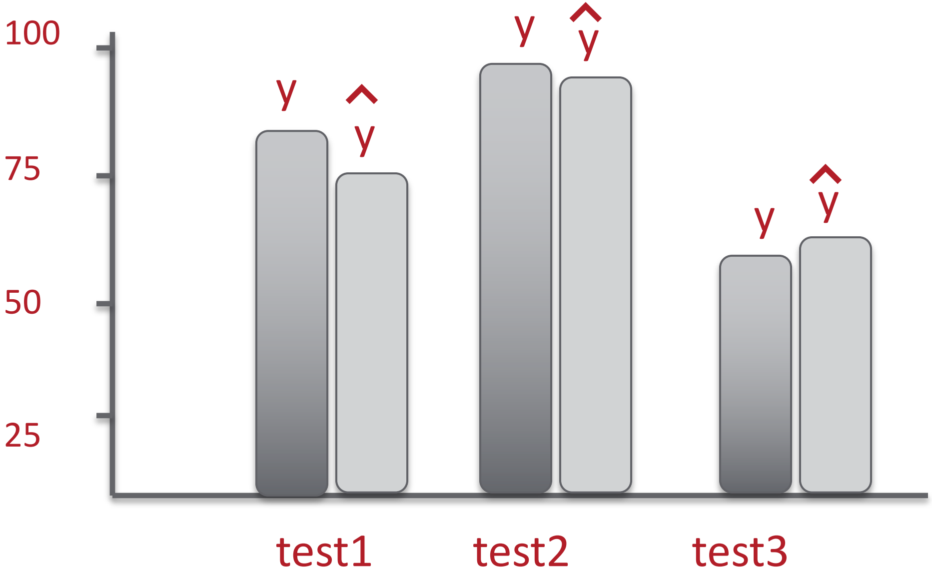
$e = y - \hat{y}$

$j = cost = e_1^2 + e_2^2 + e_3^2$

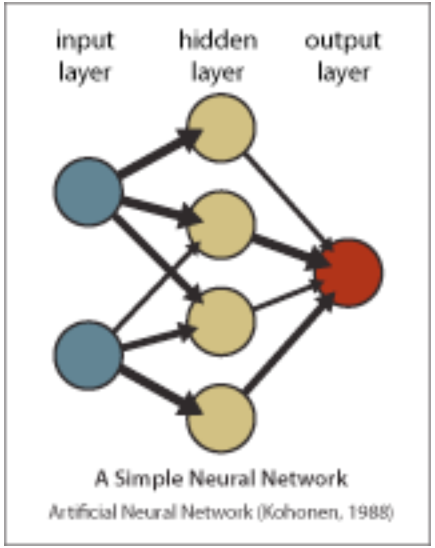
$j = \sum \frac{1}{2} (y - y_{hat})^2$

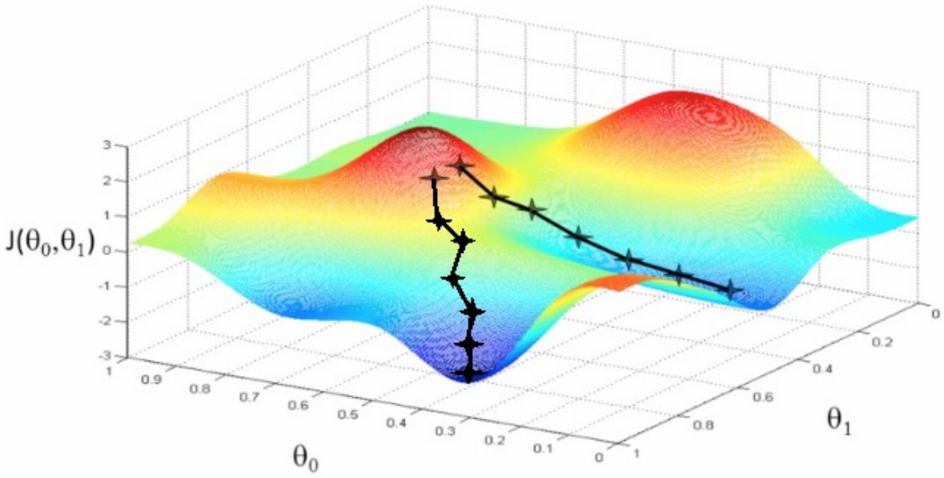
Backpropagation



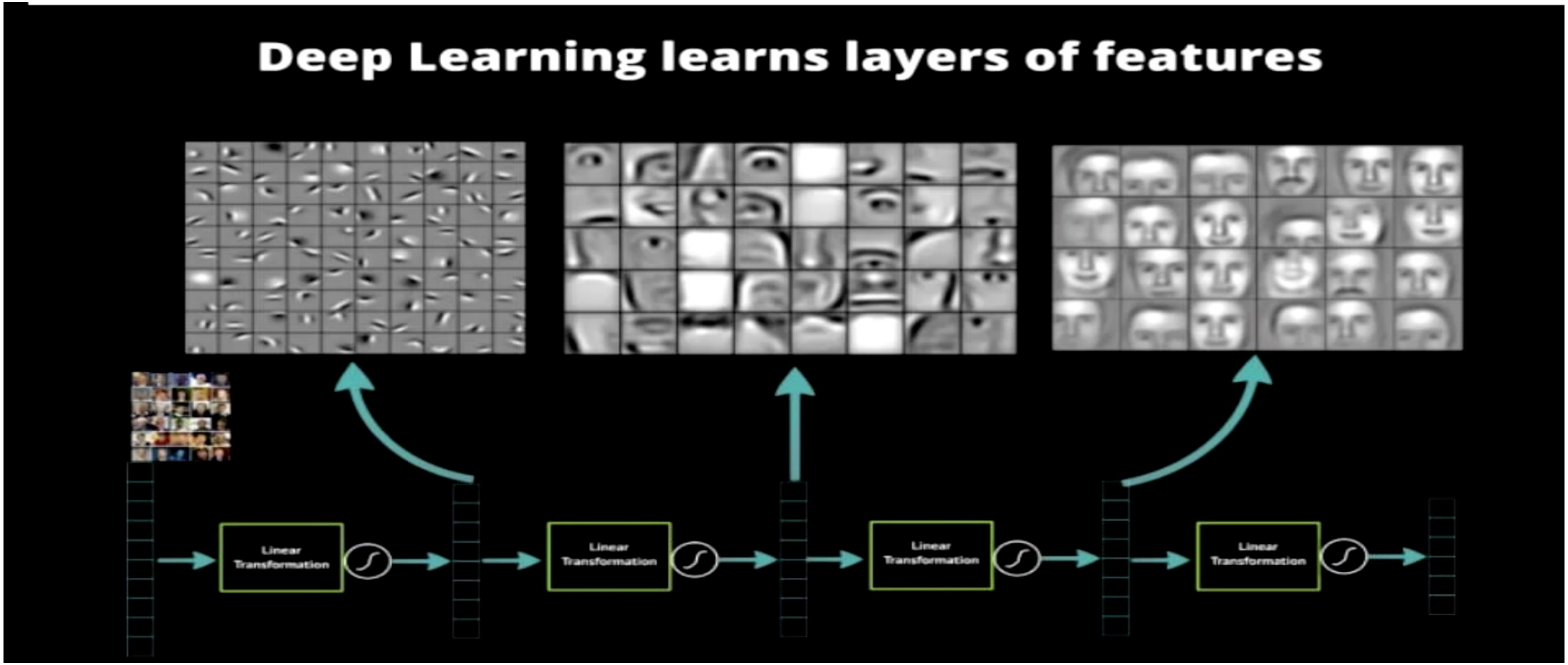


About neural networks





Deep learning is about hidden layers



Brute Statistics versus Artificial Neural Network

Regression takes the data and tries to find the result that minimizes prediction mistakes, maximizing what is called goodness of fit.

A physicist, an engineer and a statistician go on a hunting trip….

Being precisely perfect on average can mean being actually wrong each time. Regression can keep missing several feet to the left or several feet to the right. Even if it averages out to be the correct answer, regression can mean never actually hitting the target.

Unlike regression, **machine learning** predictions might be wrong on average, but when the prediction miss, they often don't miss by much. Statisticians describe this as allowing some bias in exchange for reducing variance.

Inventing a new machine learning method involves proving that it works better in practice. In contrast, inventing a regression method requires first proving that it works in theory, it requires the articulation of a hypothesis.

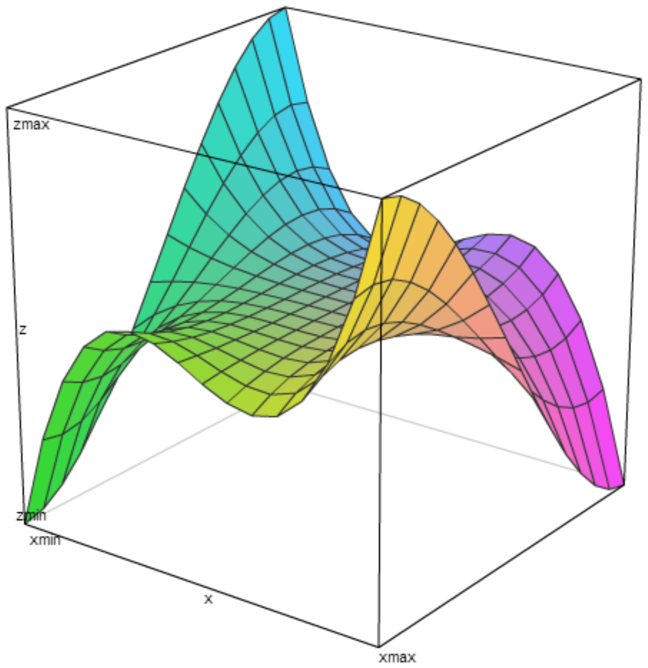
Machine learning has less need to specify in advance what goes into the model and can accommodate the equivalent of much more complex models with many more interactions between variables.

These excerpts are taken from the book **Prediction Machines** by Ajay Agrawal, Josh Gans and AVI Goldfarb

Gradient Descent

Gradient descent is algorithm that attempts to “roll down hill” (or minimize a function).

In AI, it's typically used to minimize the error function.



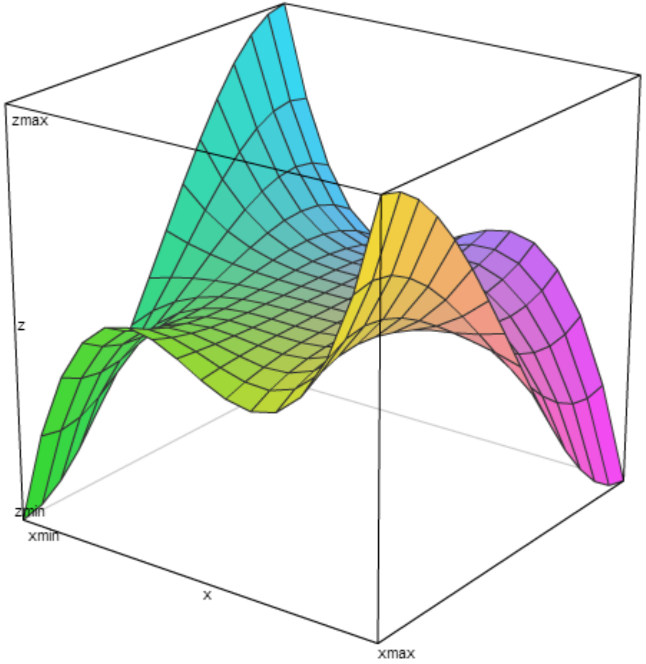
Gradient Descent

Examples of gradient descent algorithms –

Logistic regression, MLP with backpropagation, genetic algorithms, many others.

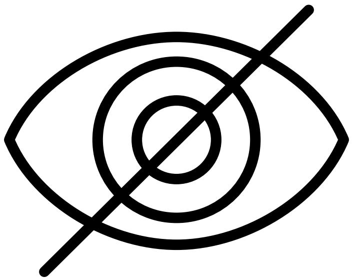
MLPs vs. Logistic regression

- Logistic regression and MLPs perform similarly.
- MLPs can be a little more flexible but are more prone to over fitting.
- MLPs are a true black box.



Gradient Descent

Gradient descent is blind!



It has no (or very little) memory of the space it has traversed. It's a bit like a blind mouse trying to make his way down a mountain.

Common problems:

- Getting stuck in local minima
- No guarantee of convergence (roaming)

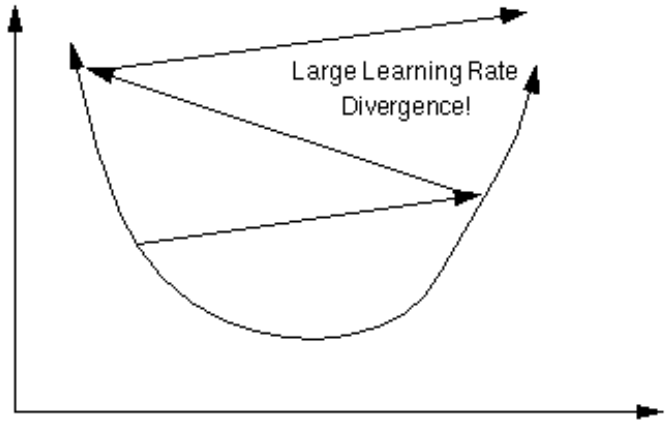
There are strategies to mitigate these problems, but nothing to solve them.

Gradient Descent – **Learning Rate**

- The learning rate in gradient descent is basically how big of a "step" the algorithm takes.

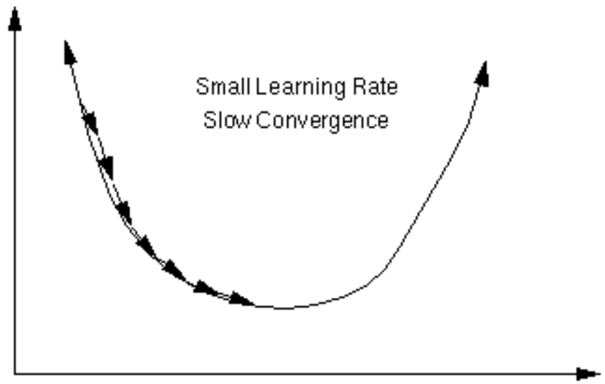
- **The appropriate rate depends on the data and solution space. But problems can arise if the rate is too large or too small.**

- **If the learning rate is too large, the algorithm might step over the global minimum.**

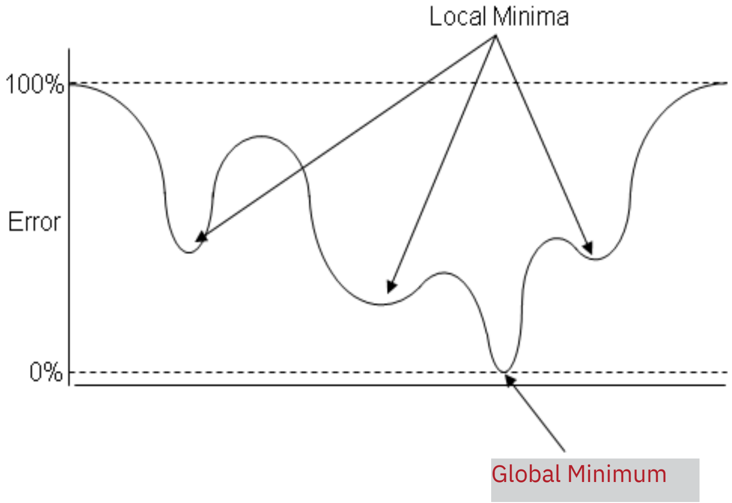


Gradient Descent – **Learning Rate**

- If the learning rate is too small, then the algorithm might take a long time to converge.



Or get stuck in a local minima.



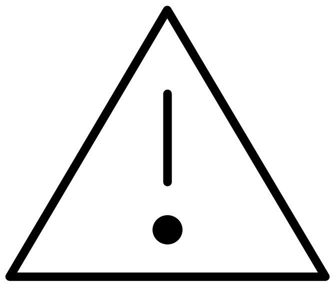
Gradient Descent – **Error Rate**

- The error rate is the level of acceptable error in the system.

- Gradient descent stops when an acceptable error level is achieved. The lowest error rate you can get isn't necessarily a good thing.

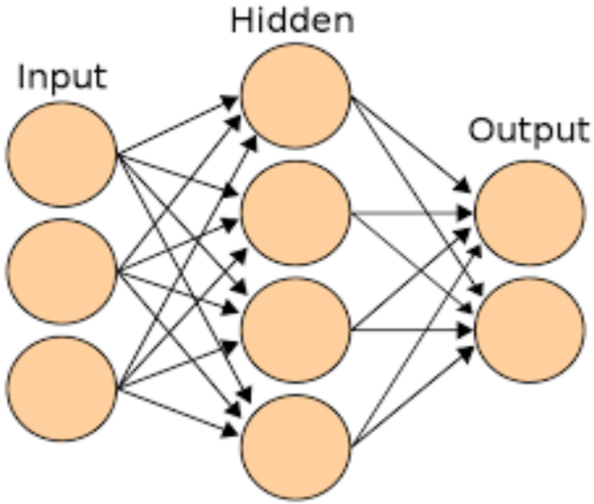
- **A very low error rate may force the model to memorize the training set too closely (over fitting).**

- Finding an appropriate error rate for a particular problem domain
  and training set will take some trial and error.



Multilayer Perceptrons – **Topology**

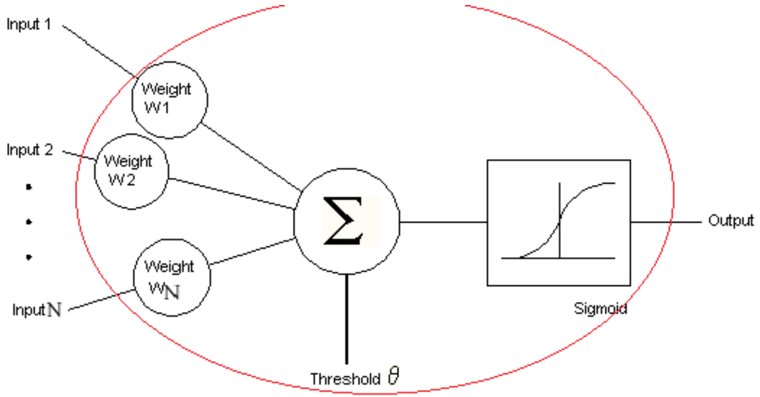
- MLPs are directed acyclic graphs made of perceptrons and weighted connections.
- Every node in the previous layer is connected to the following layer.



A basic MLP has 1 input layer, N hidden layers (usually 1), and 1 output layer. MLPs are said to be “feed forward” networks.

Multilayer Perceptrons - **Construction**

- The basic building block in a neural network is called a **Perceptron**. This is neural network analog to biological neuron.
- A perceptron takes an input and produces an output value that is governed by an activation function.
- All of the inputs have weights unique to their source. The weighted sum of these inputs is fed to the activation function.

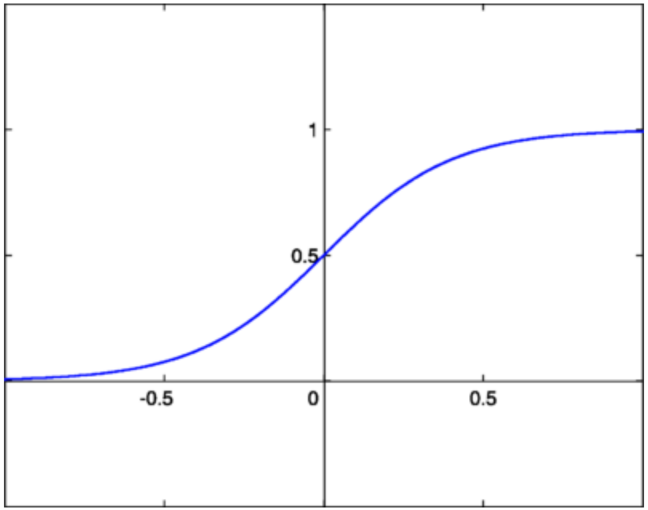


Multilayer Perceptrons – **Activation Function**

- The most common activation function is a sigmoid function. The logistic function (0 to 1) and hyperbolic tangent (-1 to 1) are the most commonly used.

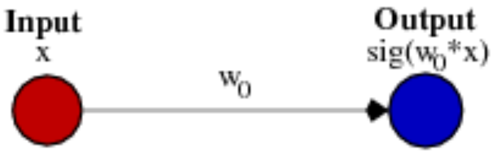
Why a sigmoid?

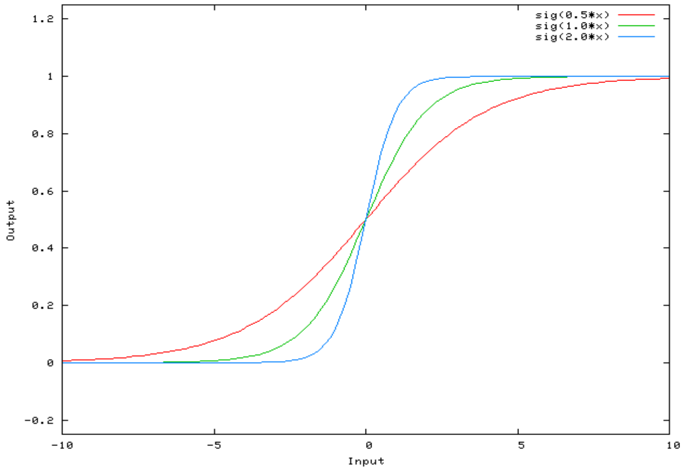
- Produces and on/off value to mimic biological function.
- An unbounded function would produce huge values which would make it impossible for gradient descent to work.



Multilayer Perceptrons – **Bias Nodes**

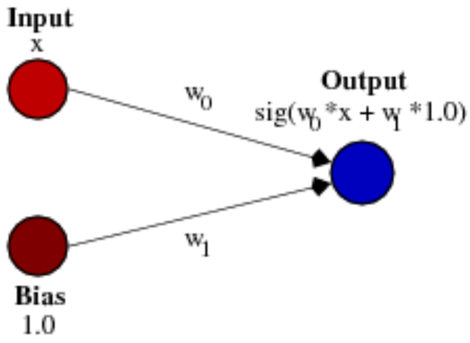
- Most networks use an additional node at each layer called a bias node. The bias node is a constant value that isn't connected other layers.
- Bias nodes allow a perceptron more flexibility with respect to where it's activation point lives. **More flexibility = More capacity to learn!** Consider the following unbiased example.

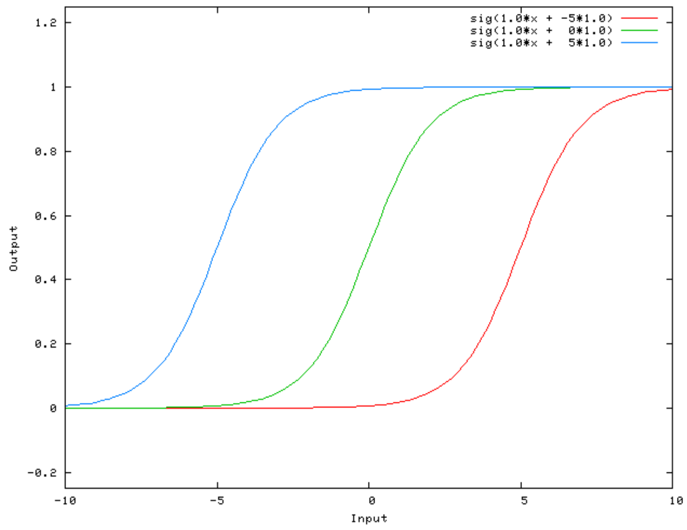




Multilayer Perceptrons – **Bias Nodes**

- Now observe how bias can shift the function left or right instead of just increasing the steepness.





Multilayer Perceptrons – **Bias Nodes**

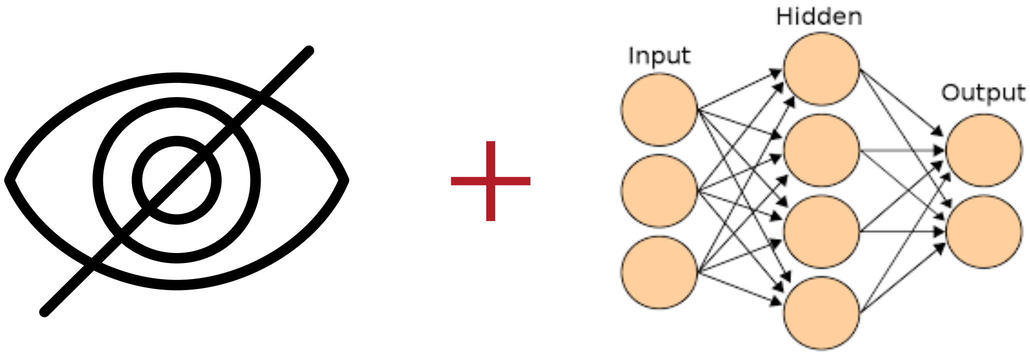
- The weights of bias nodes are adjusted by training just like the weights of normal input nodes are.

You can technically build a network with no bias. But most models use it by default and you're likely better off with bias.

Most frameworks will initialize bias (and all weights) to a random value. This changes the “starting location” for gradient descent which is why different runs of the same network with the same data will take a different number of iterations to converge.

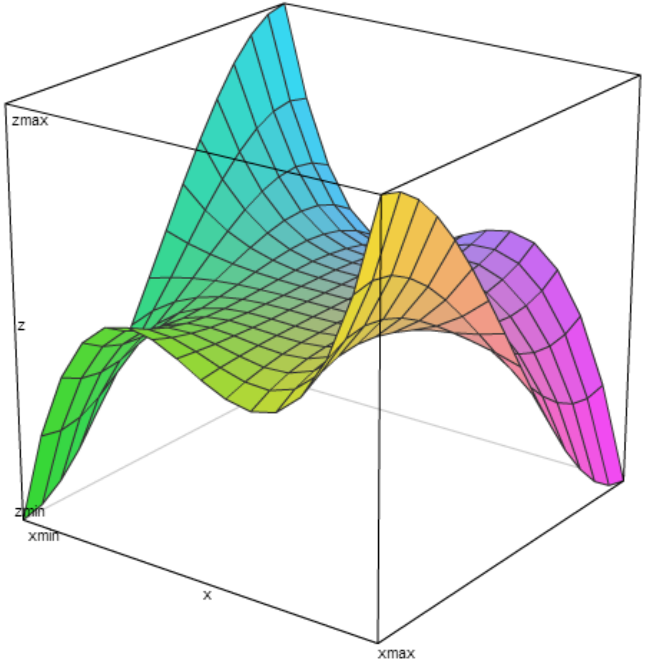
Multilayer Perceptrons – **Backpropagation**

- In a nutshell…



Multilayer Perceptrons – **Backpropagation**

- Revising our original metaphor for gradient descent.

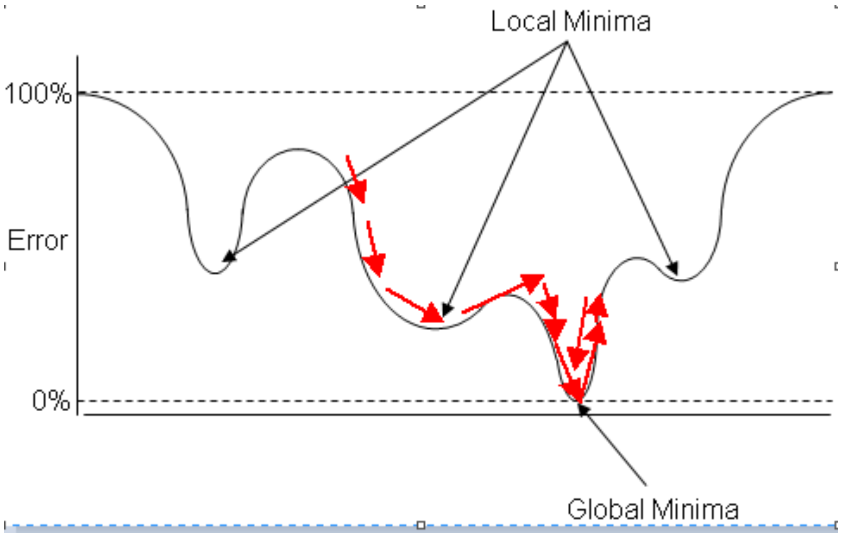


Gradient Descent Improvement: Momentum

- **Many optimizations have been made to the backpropagation algorithm to mitigate some of the known problems.**

Gradient descent with momentum changes the learning rate based on the error adjustments from the last training cycle.

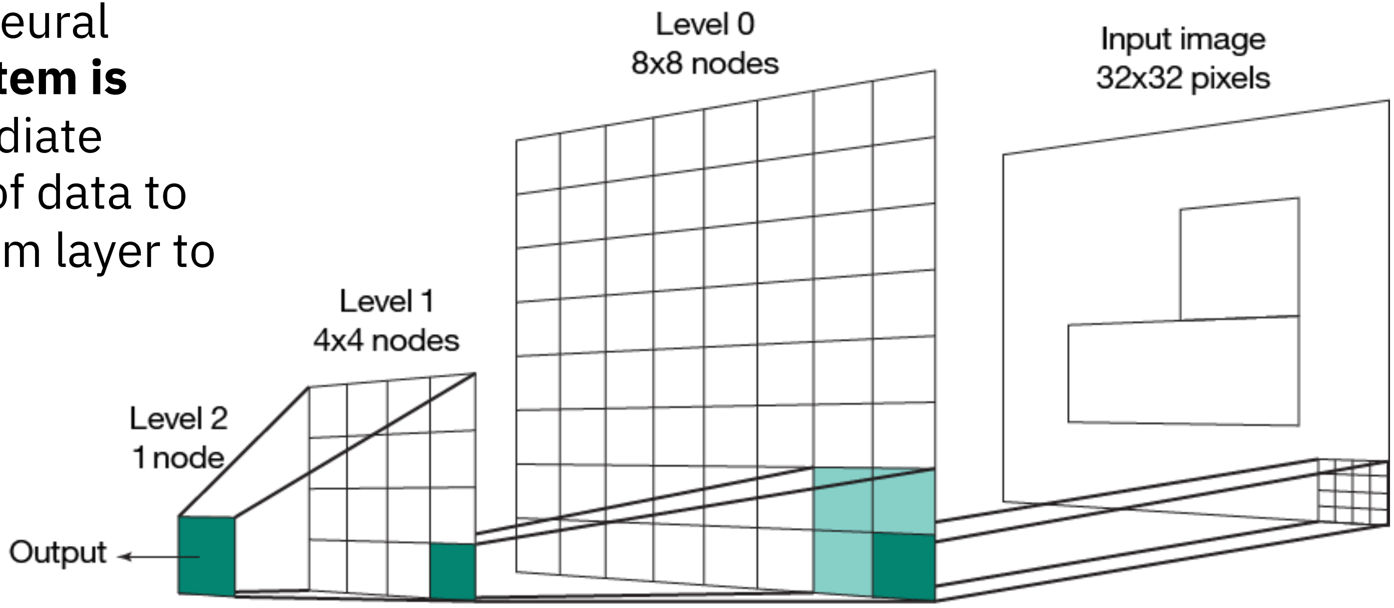
The idea is that if the gradient descent has just fallen off a steep cliff, use the momentum from the fall to push through local minimums.



The difference is in the hidden layer

Stacking of neural network layers for image recognition

When stacking layers and creating a deep neural network, **the system is learning** intermediate representations of data to help a downstream layer to perform better.



HOW DECISION TREES PREDICT AND CLASSIFY

Predict if Nemra will commute to the office

- **Hard to guess under what conditions Nemra will work from home or drive to the office.**
- Let's divide and conquer:
- Split into subsets
- Are the all pure (all yes or all no)
- If yes: stop
- If no: repeat

Day	Weather	Type of calls	Exercise	Commute?
D15	Rain	Video	Walk	?

Day	Weather	Type of calls	Exercise	Commutes
D1	Sunny	Video Conference	Walk	No
D2	Sunny	Video Conference	Gym	No
D3	Overcast	Video Conference	Walk	Yes
D4	Rain	Video Conference	Walk	Yes
D5	Rain	Telephone	Walk	Yes
D6	Rain	Telephone	Gym	No
D7	Overcast	Telephone	Gym	Yes
D8	Sunny	Video Conference	Walk	No
D9	Sunny	Telephone	Walk	Yes
D10	Rain	Telephone	Walk	Yes
D11	Sunny	Telephone	Gym	Yes
D12	Overcast	Video Conference	Gym	Yes
D13	Overcast	Telephone	Walk	Yes
D14	Rain	Video Conference	Gym	No

The Decision Tree

9 yes / 5 no

Weather

Sunny

Day	Weather	Type of calls	Exercise
D1	Sunny	Video	Walk
D2	Sunny	Video	Gym
D8	Sunny	Video	Walk
D9	Sunny	Telephone	Walk
D11	Sunny	Telephone	Gym

2 yes / 3 no
Split further

Overcast

Day	Weather	Type of calls	Exercise
D3	Overcast	Video	Walk
D7	Overcast	Telephone	Gym
D12	Overcast	Video	Gym
D13	Overcast	Telephone	Walk

4 yes / 0 no
Pure subset

Rain

Day	Weather	Type of calls	Exercise
D4	Rain	Video	Walk
D5	Rain	Telephone	Walk
D6	Rain	Telephone	Gym
D10	Rain	Telephone	Walk
D14	Rain	Video	Gym

3 yes / 2 no
Split further

If Entropy, then branch further

9 yes / 5 no

Weather

Overcast

Day	Weather	Type of calls	Exercise
D3	Overcast	Video	Walk
D7	Overcast	Telephone	Gym
D12	Overcast	Video	Gym
D13	Overcast	Telephone	Walk

Sunny

Type of calls

Video

Day	Type of calls	Exercise
D1	Video	Walk
D2	Video	Gym
D8	Video	Walk

Telephone

Day	Type of calls	Exercise
D9	Telephone	Walk
D11	Telephone	Gym

Rain

Exercise

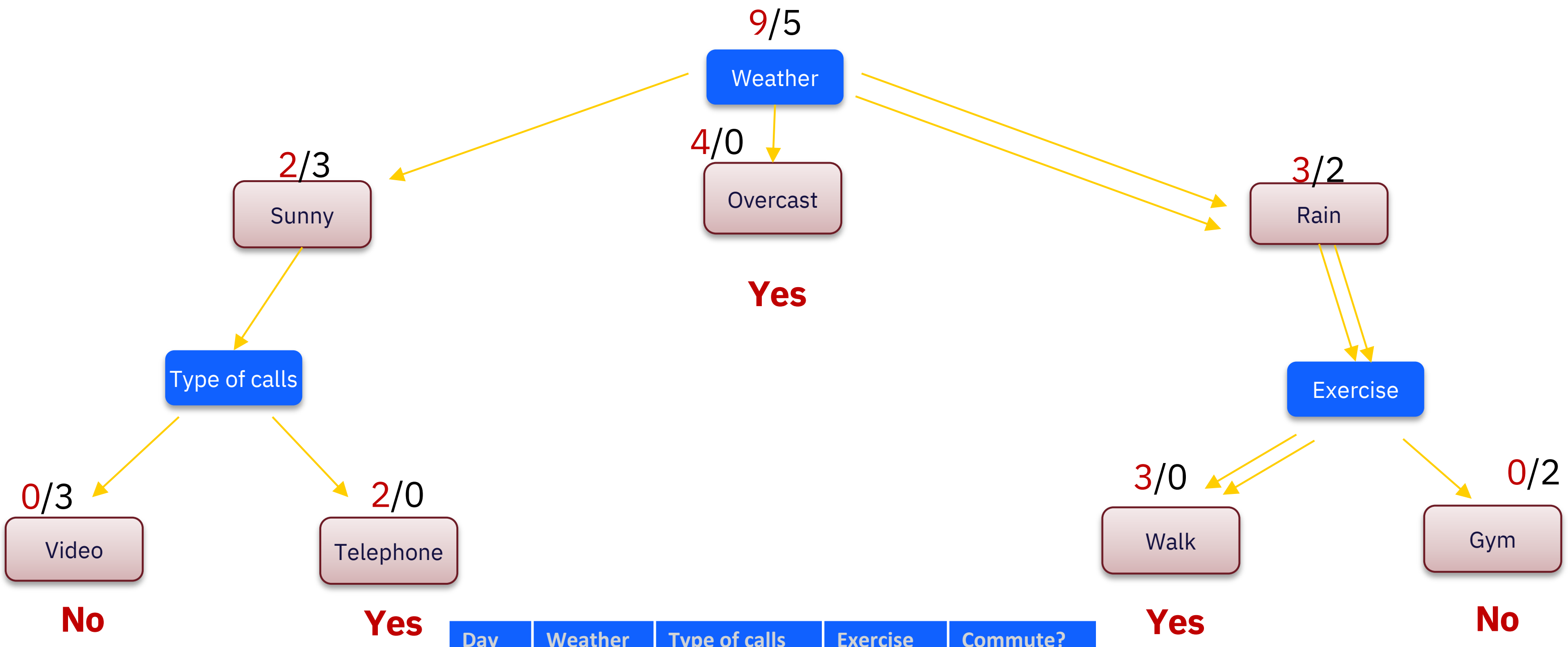
Walk

Day	Type of calls	Exercise
D4	Video	Walk
D5	Telephone	Walk
D10	Telephone	Walk

Gym

Day	Type of calls	Exercise
D6	Telephone	Gym
D14	Video	Gym

The Prediction



New data: Day 15

Day	Weather	Type of calls	Exercise	Commute?
D15	Rain	Video	Walk	? **Yes**

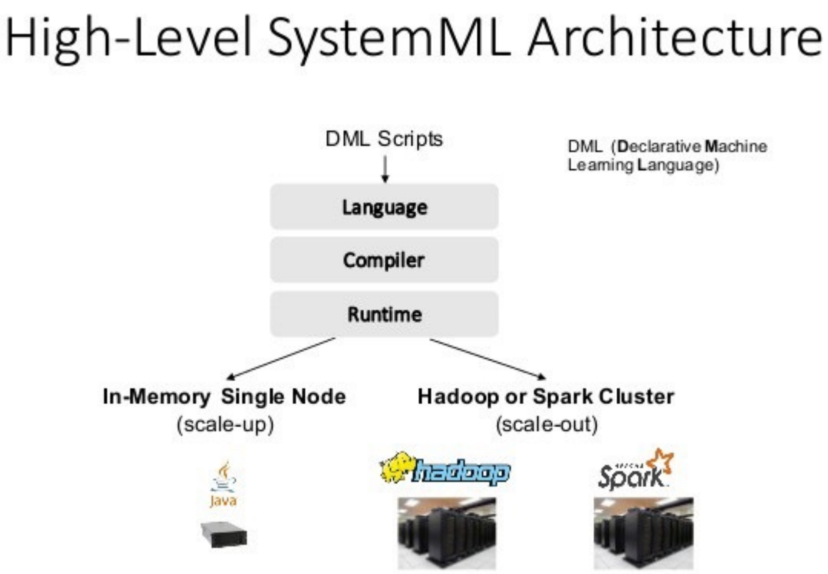
Deep Learning Ecosystem

Deep learning ecosystem

Platform as a Service Providers:
Deep learning services included as part of PaaS solutions. Technologies like IBM Cloud, Microsoft Azure, Amazon AWS or Google Developer Cloud.

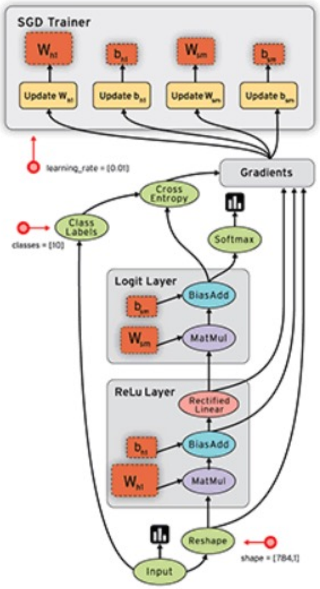
Deep Learning Frameworks:
Libraries and programming models that enable the fundamental constructs to build deep learning applications.

Deep learning framework: Apache SystemML (ML)



- The Apache SystemML language, Declarative Machine Learning (DML), includes linear algebra primitives, statistical functions, and ML-specific constructs that make it easier and more natural to express ML algorithms.
- DML significantly increases the productivity of data scientists by providing full flexibility in expressing custom analytics as well as data independence from the underlying input formats and physical data representations.

Deep learning framework: TensorFlow



- Google’s TensorFlow deep learning framework was developed originally by the Google Brain Team for conducting research in machine learning and deep neural networks.
- **The framework’s name is derived from the fact that it uses data flow graphs, where nodes represent a computation and edges represent the flow of information — in Tensor form — from one node to another.**

Deep learning framework: Torch

- Torch was based upon the scripting language Lua, which was designed to be portable, fast, extensible, and easy to use with an easy-to-use syntax.
- **Torch features many community-contributed packages, giving Torch a versatile range of support and functionality.**

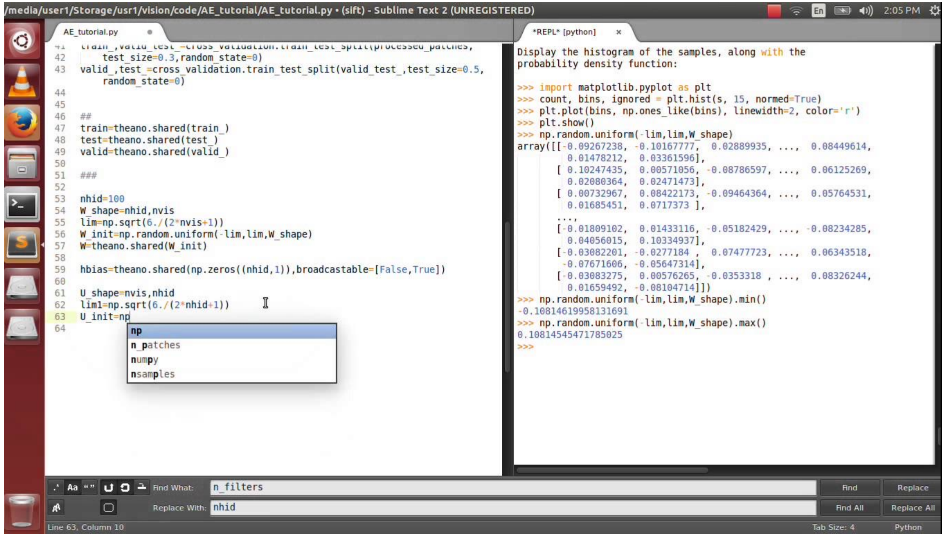
Simple NN in Torch

```
-- create closure to evaluate f(X) -- and df/dX
local feval = function(x)
   -- get new parameters
   if x ~= parameters then
      parameters:copy(x)
   end

   -- reset gradients
   gradParameters:zero()

   -- f is the average of all criterions
   local f = 0
```

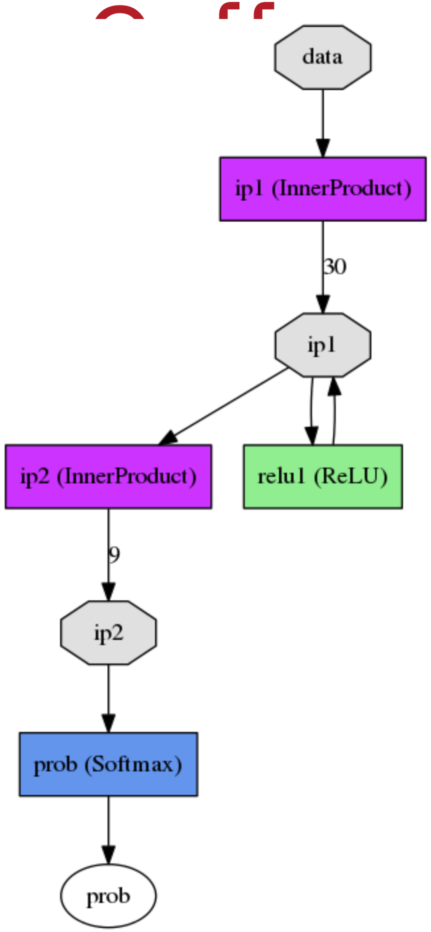
Deep learning framework: Theano



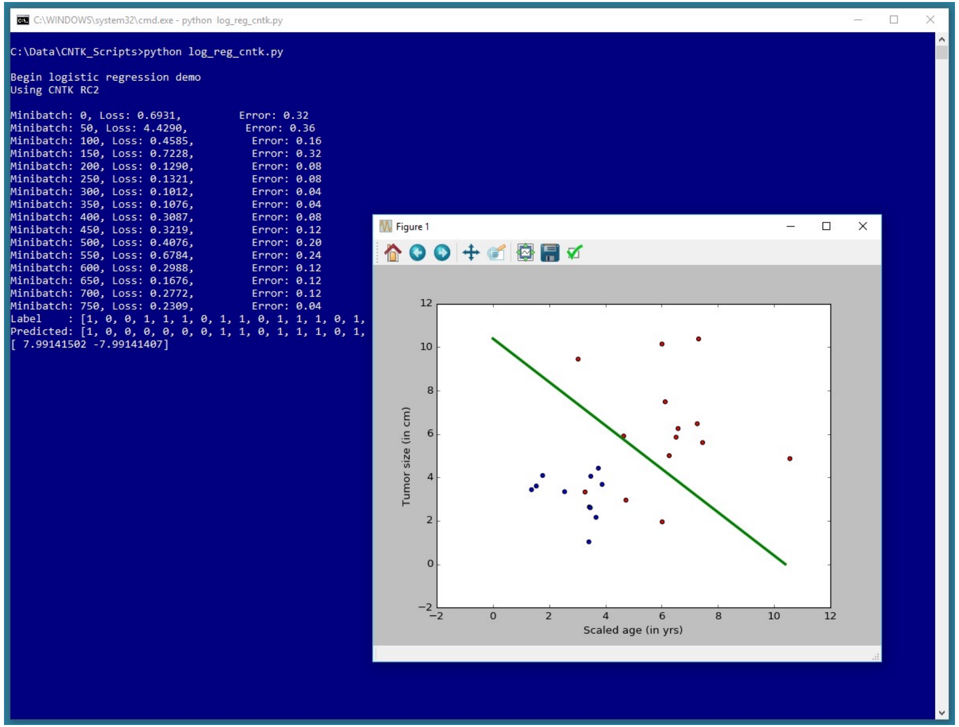
- Very popular within the academic research community, [Theano]() is considered grand-daddy of deep-learning frameworks, which is written in Python.
- **Theano is a library that handles multidimensional arrays, like Numpy. Used with other libraries, it is well suited to data exploration and intended for research.**

Deep learning framewor

- Caffe is a well-known and widely used machine-vision library that ported Matlab's implementation of fast convolutional nets to C and C++. Caffe was developed at the Berkeley Vision and Learning Center (BVLC).
- **Caffe is useful for performing image analysis (Convolutional Neural Networks, or CNNs) and regional analysis within images using convolutional neural networks (Regions with Convolutional Neural Networks, or RCNNs).**



Deep learning framework: CNTK



- [CNTK](#) is Microsoft’s open-source deep-learning framework. The acronym stands for “Computational Network Toolkit.”
- **While CNTK appears to have a [permissive license](#), it has not adopted one of the more conventional licenses, such as ASF 2.0, BSD or MIT.**

ROLES INTEGRATED ENVIROMENT

[7]
You need framework for **Machine Learning** and likely framework for **Deep Learning**
Scikit, TensorFlow, Keras, Torc, Theano, etc.

[6]
Choose your scientific computing and statistic packages:
SciPy, NumPy are widely utilized

[5]
Choose your visualization and plotting tools:
Matplotlib, PixieDust are top market trends

[4]
Choose your data munging libraries and tools:
Pandas is a very flexible library under the python framework

[3]
Choose your programming language:
python is most versatile, R and **Scala** are mostly specialized statistical packages

[2]
Now you need a development environment.
- Get **Jupyter Notebooks** (julia+python+R)
- Get it from **Anaconda** (www.continiuum.io)

[1]
So you have your collected data:
Is it structured, semi-structured, unstructured, mix?

Text editor	MS Excel	DB2, MS SQL	Hadoop	MongoDB	Watson Studio	Sentiment
CSV	xls	RDBMS	DFS	JSON	images	text