Writing Neural Network Code: Introduction to TensorFlow, Hands-On

Class 1: A Brief History of Artificial Neural Networks

May 11, 2020

Charles J. Lord, PE
President, Consultant, Trainer
Blue Ridge Advanced Design and Automation

DesignNews

Blue Ridge Advanced Design and Automation
Asheville, North Carolina

CEC CONTINUING EDUCATION CENTER

Presented by:
Digi-Key ELECTRONICS

This Week's Agenda

5/11 A Brief History of Artificial Neural Networks

5/12 Neural Network Simulation and Programming

5/13 TensorFlow Hands-On Part 1: Hello World!

5/14 TensorFlow Hands-On Part 2: Defining and Building Your Network

5/15 TensorFlow Hands-On Part 3: Teaching and Testing and Conclusion

This Week's Agenda

5/11 A Brief History of Artificial Neural Networks

5/12 Neural Network Simulation and Programming

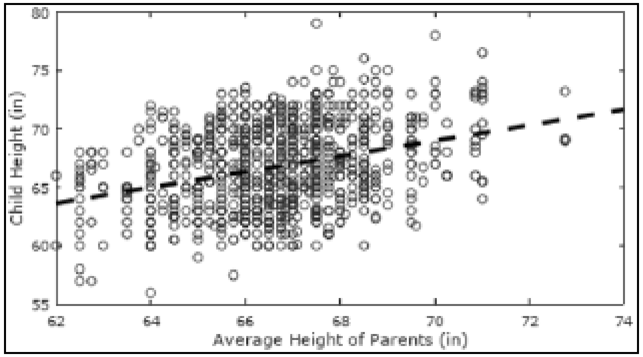
5/13 TensorFlow Hands-On Part 1: Hello World!

5/14 TensorFlow Hands-On Part 2: Defining and Building Your Network

5/15 TensorFlow Hands-On Part 3: Teaching and Testing and Conclusion

AI is "Hot" – why do I want it?

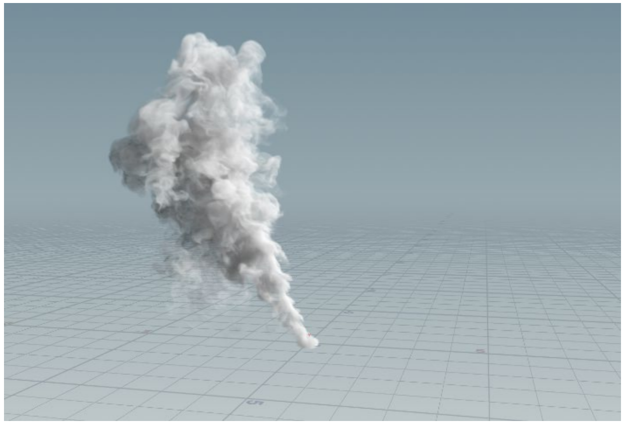
- The more front-end processing that can be done BEFORE sending data over the I(I)ot, the better.
- A lot of data is difficult to easily process or distinguish via simple mathematical means. We often need to apply regression or ' best case' matching.

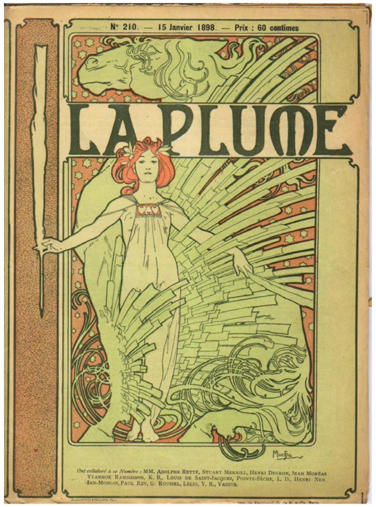


Perception

- We can look at input data in many different ways – we as humans take that data in as what we ***perceive*** of that data.
- Example – if I speak the word “plume” your perception of that sound depends on many factors – but essentially uses (roughly) the same set of neurons in each person. It is what you have LEARNED about that sound.

Plume





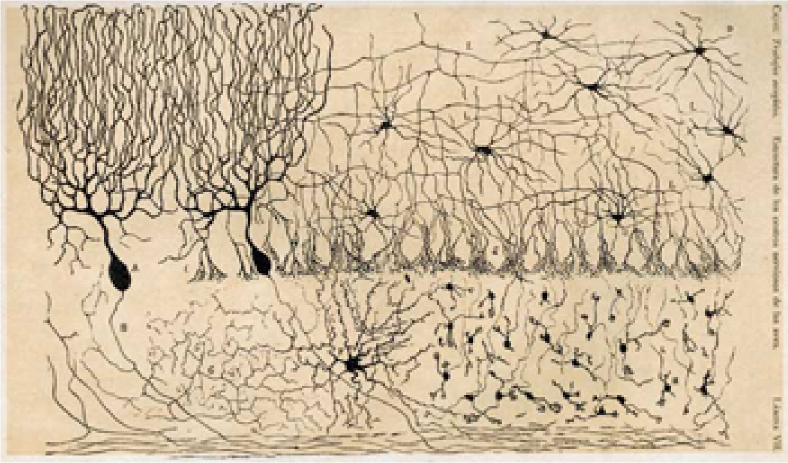
Question 1 – What other things might “plume” mean?

Presented by:

Modelling the Brain

- The concept of modelling the function of the human brain for forming complex traits is not a new idea. Back in 1894, Francis Galton first proposed the use of statistical regression to study inherited traits. This was the beginning of the study of how one could use mathematics to form learning patterns to recognize things not exactly like a known pattern or sequence. From this research came the concepts of normal curves, correlation, and standard deviation.

Ramón y Cajal 1905



- Building on the theory that the nervous system was made up of individual cells, Santiago Ramón y Cajal did extensive work in examining the brain cells of chickens.
- These cells were coined the term “neuron” by Santiago Ramón y Cajal

Modelling the Neuron

- Galton’s work was brought into greater focus when in 1943 McCulloch and Pitts made a model of the neuron and created an early model of the artificial neuron.
- This was then expanded by Frank Rosenblatt in 1957 to what he called the perceptron, which was then arrayed in layers to recognize images.
- This led to the US Department of Naval Research’s Mark 1 Perceptron, the first physical neural network computer.

Mark I Perceptron 1958-1969

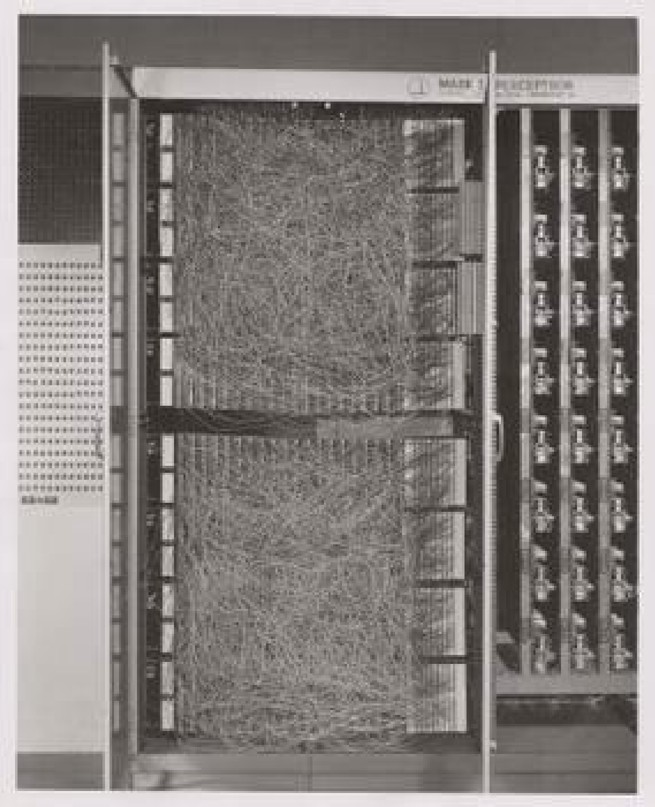
- Basic Threshold Function

$$f(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{w} \cdot \mathbf{x} + b > 0, \\ 0 & \text{otherwise} \end{cases}$$

- Where
  w · x is the dot product

$$\sum_{i=1}^{m} w_i x_i$$

- Did not last long – too hard to train / program

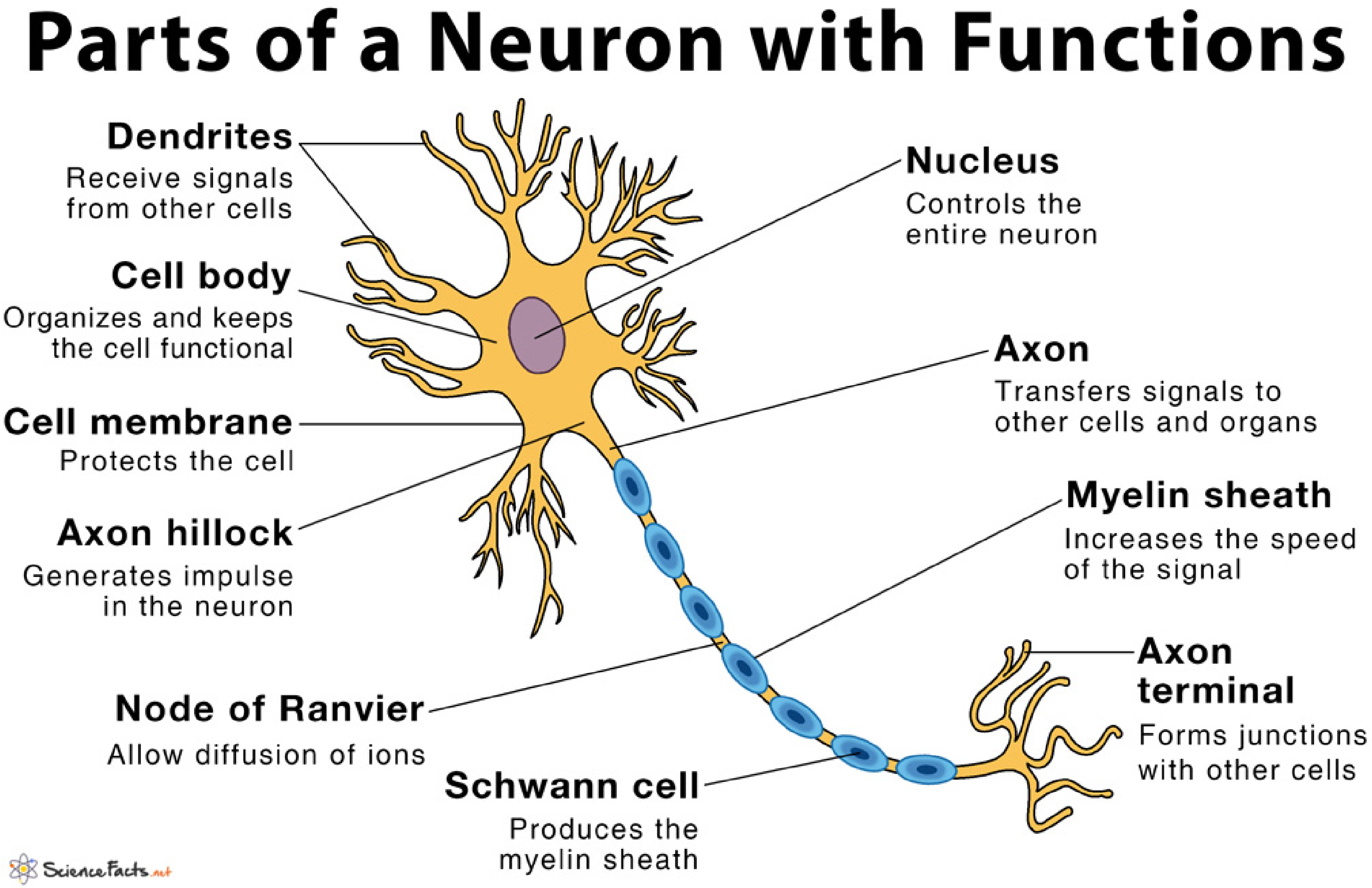


Enter Backpropogation

- Paul Werbos found how to train a network through backpropagation in 1974, but this was still computationally intensive. Even in 1991, a National Institute on Aging research program for predicting fall behavior from camera images required a Cray supercomputer to effectively operate, making the concept impractical at the time. It wasn't until the end of the millennium when effective methods were discovered to make NN practical.

Question 2 – Experience with a Cray? What was the 'seat' for?

This is your Brain. (1 billionth)



Parts of a Neuron with Functions

Dendrites
Receive signals from other cells

Cell body
Organizes and keeps the cell functional

Cell membrane
Protects the cell

Axon hillock
Generates impulse in the neuron

Node of Ranvier
Allow diffusion of ions

Schwann cell
Produces the myelin sheath

Nucleus
Controls the entire neuron

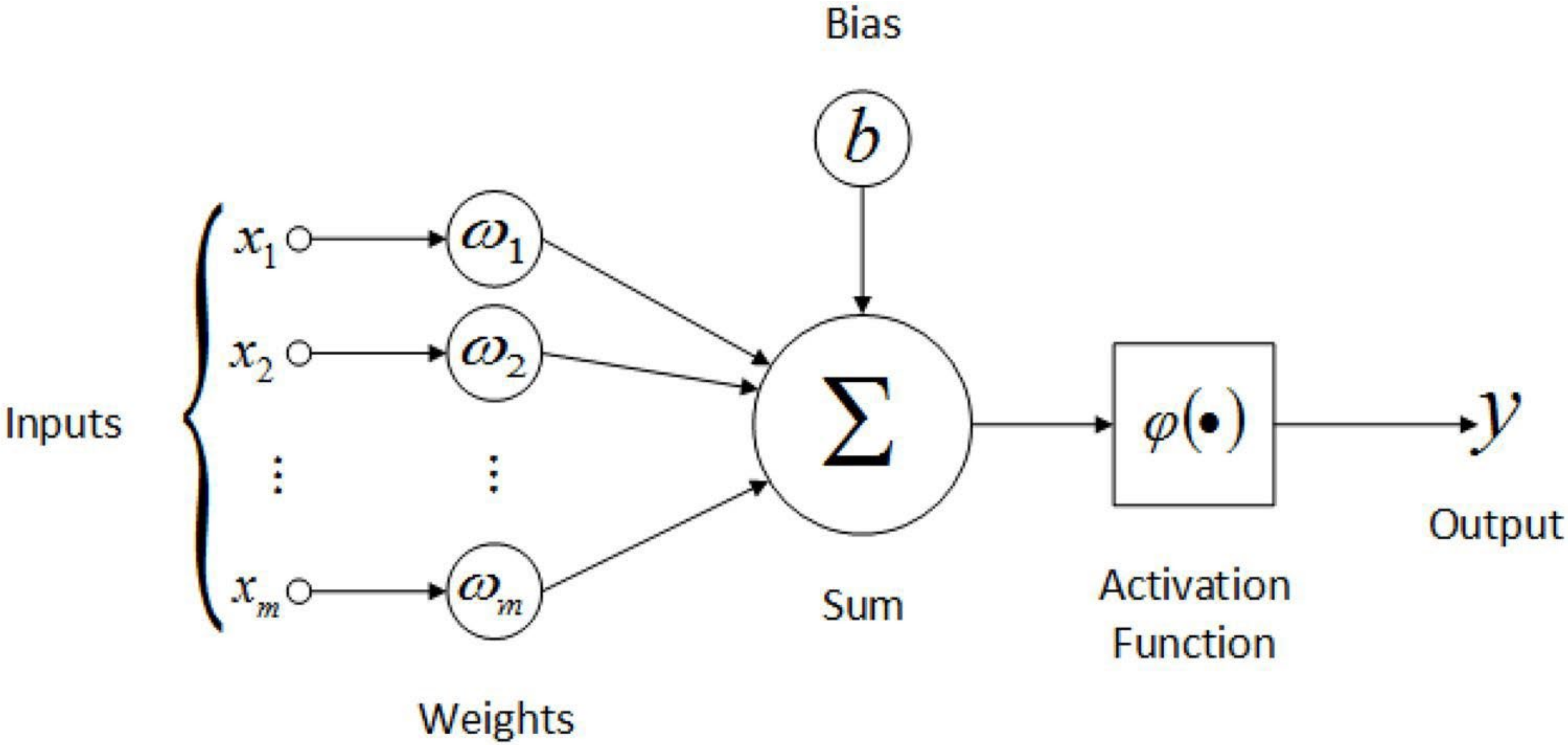
Axon
Transfers signals to other cells and organs

Myelin sheath
Increases the speed of the signal

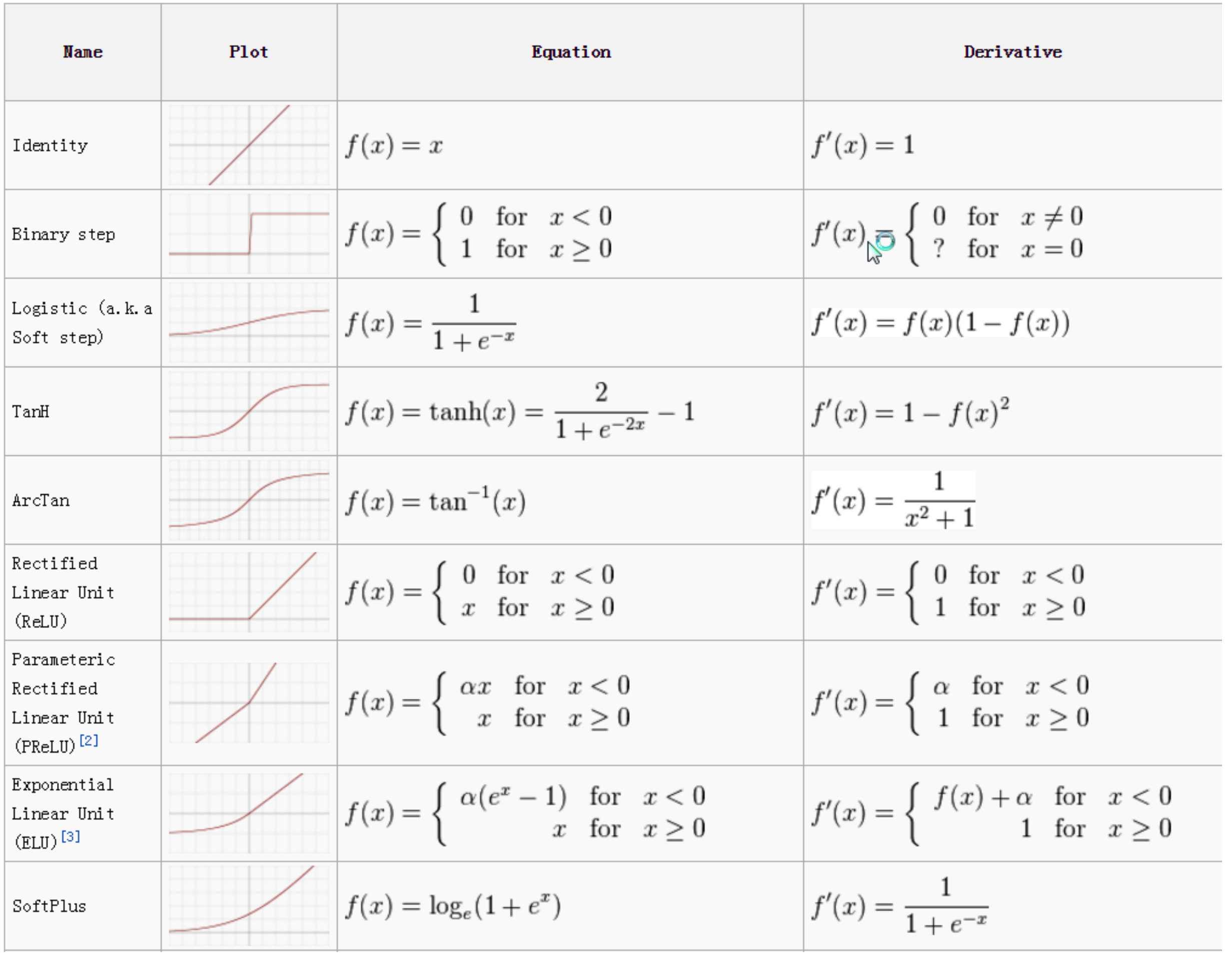
Axon terminal
Forms junctions with other cells

ScienceFacts.net

Mathematical Model



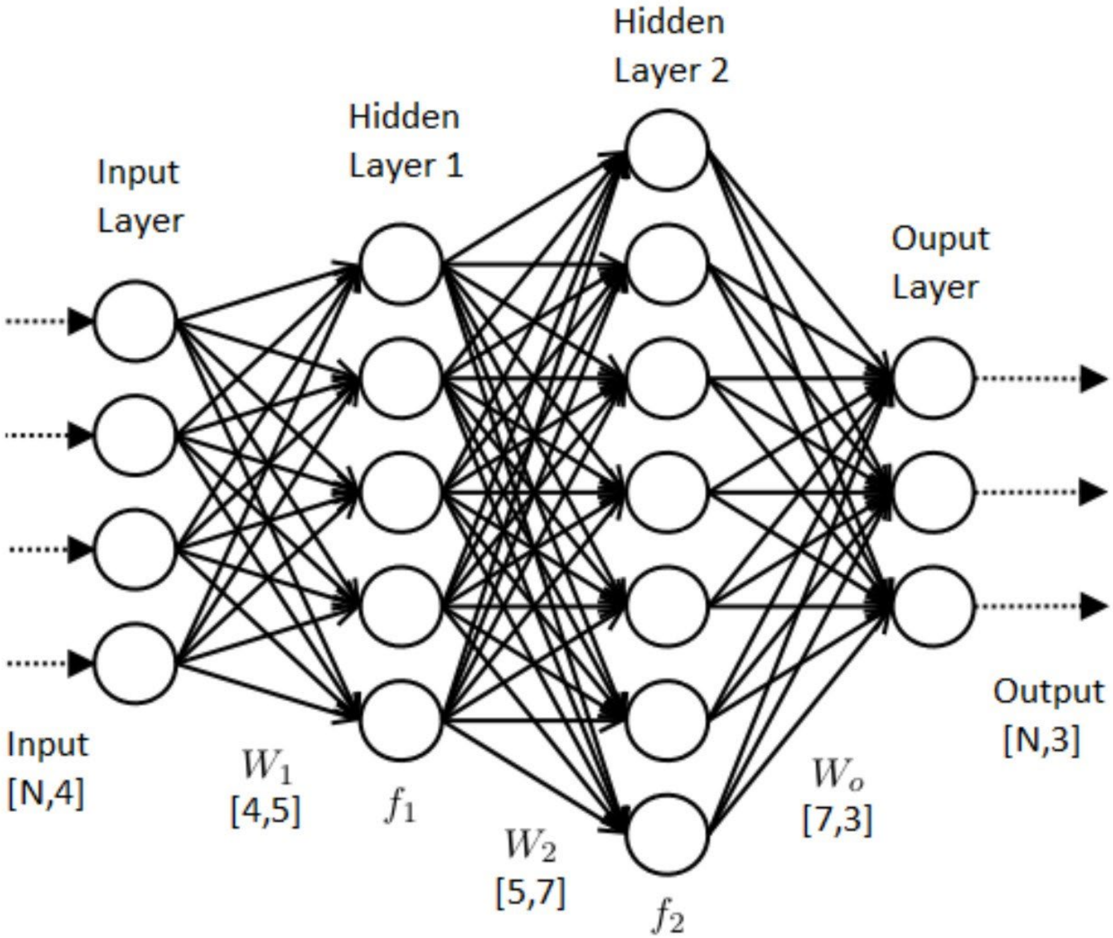
Activation Functions



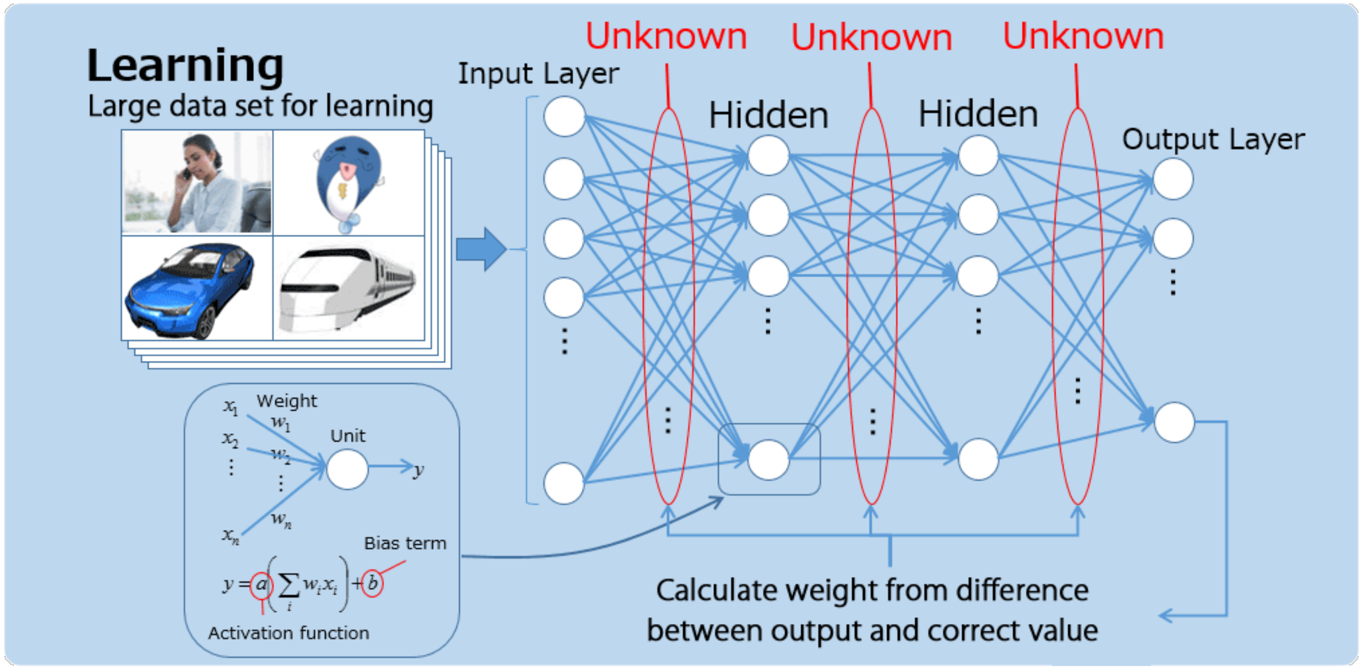
Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Binary step		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1+e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
TanH		$f(x) = \tanh(x) = \frac{2}{1+e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2+1}$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU)[2]		$f(x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Exponential Linear Unit (ELU)[3]		$f(x) = \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1 + e^x)$	$f'(x) = \frac{1}{1+e^{-x}}$

Presented by:

A Neural Network

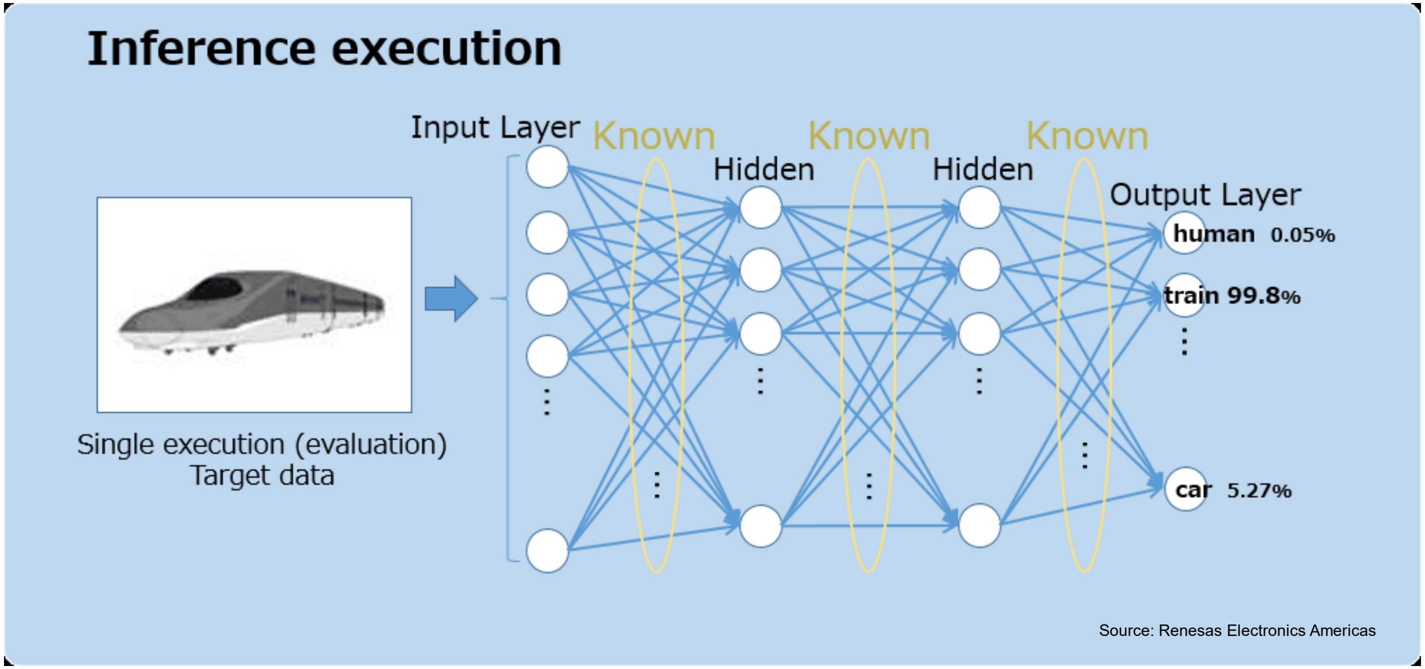


Training a Neural Net: Data and Function



Source: Renesas Electronics Americas

Testing:
Confidence of Finite Answers



Source: Renesas Electronics Americas

Like human learning, outputs can be variable and still contain errors

Architecture

- Step-by step on single core
  - Can be done – certainly not ‘real-time’!
- Step-by-step on multicore
  - Somewhat faster
- Dedicated multicore (often massive)
  - GPU based
  - IP on FPGA

Tools

- Many competing tools exist – some of the more popular
- Caffe
  - One of the oldest in image detection
  - Largest collection of pretrained models*
- Torch
  - Used by many apps including facebook
  - Lua or Python based

Tools

- Keras
  - – Easier to learn
  - – Large active community
- Tensorflow
  - – Growing in popularity
  - – Best suited to larger processors
  - – Keras frontend
  - – uTensor is a scaled-down version supported by ARM

Question 3 – What languages is TensorFlow programmed in?

Tomorrow

- We will install TensorFlow on our computers and also look at the online Google Colaboratory (Colab) notebook environment.
- We will use Google Colab for much of our learning – needs no local computing.

This Week's Agenda

5/11 A Brief History of Artificial Neural Networks

5/12 Neural Network Simulation and Programming

5/13 TensorFlow Hands-On Part 1: Hello World!

5/14 TensorFlow Hands-On Part 2: Defining and Building Your Network

5/15 TensorFlow Hands-On Part 3: Teaching and Testing and Conclusion

Please stick around as I answer your questions!

- Please give me a moment to scroll back through the chat window to find your questions
- I will stay on chat as long as it takes to answer!
- I am available to answer simple questions or to consult (or offer in-house training for your company)
  c.j.lord@ieee.org
  http://www.blueridgetechnc.com
  http://www.linkedin.com/in/charleslord
  Twitter: @charleslord
  https://www.github.com/bradatraining