People and dates:

* Frank Drake – Drake Equation - 1961
* Robin Hanson – Great Filter
* Isaac Asimov – Three laws of Robotics
* Nick Bostrom – Paperclip Maximizer - 2003
* Elaine Rich – Dr O’neals definition of AI
* John Searle – Chinese Room Paradox
* Doug Lenat – Cyc – 1984 to present
* Vernor Vinge – forecasted takeover by singularity by 2030– 1990s
* John von Neumann – First coined singularity – 1958
* Raymond Kurzweil – Singularity predication at 2045 – human level intelligence by 2029
* Stuart Armstrong – study of expert opinion of agi averaged to 2040
* Warren McCulloch & Walter Pitts - Model of neuron - 1943
* Frank Rosenblatt - Perceptron Learning Rule - 1957
* AI first coined – Dartmouth Conference – 1956
* Enrico Fermi & Michael H. Hart – Fermi Paradox
* Freddie – Computer Vision robot

Information:

* Drake Equataion -
  + N – number of communicaftive civilizations in the milky way
  + R – Average rate of star formation in our galaxy
  + F­p – Fraction of stars with planets
  + Ne – fraction of those planets that may be able to support life
  + Fl – fraction of those planets that develop life
  + Fi – fraction of those planets which life is intelligent and civilized
  + Fc – the fraction of those civilizations that have developed communications
  + L – Length of time over which such civilizations release detectable signals
* Three laws of robotics
  1. A robot may not injure a human being or, through inaction, allow a human being to come to harm
  2. A robot must obey the orders given it by human beings except where such orders would conflict with the First Law
  3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Laws.
* Artificial Intelligence – branch of computer science that has as its goal the construction of a general-purpose intelligence – of constructing machines that are capable of doing all of the things which, at the present time, people are better.
  + Computer Vision
  + Computer Speaking
  + Natural Language Translation
  + Game Playing
  + Planning and Reasoning
  + Creativity
  + Driving
* Levels
  + Weak AI behaves intelligently but cannot claim to be conscious
  + Strong AI is truly conscious
  + There is no test to distinguish between them
* Types/Approaches
  + Symbolic AI (good ol’ fashion AI) reaches logical conclusions through hard-coded inductive branches
  + Machine Learning is a way of training an AI to respond to certain scenarios and environments in a desired way – it cannot explain why it reached a conclusion
* Pathways for advances
  + Deep learning Algorithms
  + Massive data collection via internet
  + Better hardware from the emergence of parallel GPUs
* Eras
  + Success Optimism and Growth : 1950s – 1960s
  + Brick Wall in front of progress : 1970s
  + Expert Systems : 1980s
  + Little progress that lays the foundation for future growth : 1990s to 2000s
  + Machine Learning boom : 2005 – now
* Paradigms
  + Supervised learning – preclassified inputs for training
    - Training and testing data
  + Unsupervised learning
    - Given input data and a cost function to be minimized
  + Reinforcement learning
    - The agent performs an action in its environment that generates a cost. The goal is to minimize long-term cost
* Architectures
  + DNN – Neural Network with “many” layers (5-25)
  + CNN – Convolutional networks “squeeze” the input to extract important features
    - Softmax – each layer’s outputs are normalized so that they all add up to 1
    - Maxpool – The output of a cluster of neurons is analyzed so that the neuron with the highest output becomes the input for a single neuron in the next network level
  + RNN – Allow for feedback mechanisms
    - LTSM – Allow for keeping track of memory for problems that may grow during analysis

Perceptrons

* Uses two sets of data for supervised learning
  + Training Data
  + Testing Data
  + Set data should be disjoint
* Guaranteed to converge on perceptron-learnable problems
* Perceptron-learnable problems are linearly separable
* Limitations of the space of problems that can be solved by a perceptron can be overcome by adding multiple layers, but there is no way to determine blame for multiple-layered perceptron networks
  + This is called the credit assignment problem
* Three components
  + Inputs
  + Weights
  + Threshold
* Formula
  + Add together all False Negative vectors to get a sum vector, SumFN
  + Add together all False Positive vectors to get a sum vector, SumFP
  + Compute learning gradient:
  + Compute delta:
  + Compute new weights:

Sigmoid Neurons

* Supervised
* NOT guaranteed to converge, can get stuck
* Can be trained on multiple levels
* Components
  + Inputs
  + Weights
  + Bias (one)
    - These produce the Z
    - Z is passed through the sigmoid function
    - Output is then fed forward
* Formula
  + Output:
    - w is the vector of weights
    - x is the vector of inputs
    - b is the bias of a particular neuron
    - Ultimately this comes out to
    - The formula for the activation of neuron *j* at level *L* is given by
      * + Where *k* is the set of inputs for the neuron
* Stochastic Gradient Descent
  + Randomize training set items
  + Divide into equally sized mini-batches
  + Compute weight gradients and bias gradients over mini batch *i*
  + After completing mini-batch, update weights and biases with:

    - where *η* is the learning rate and *ni* is the size of the mini-batch
  + If mini-batches remain, return to step 3
  + If stopping criteria have not been met, return to step 1
* An iteration over all mini-batches is an *epoch*
* Back-propagation – Make a backwards pass through the network to compute the bias gradient and weight gradient for every neuron
  + Calculating the Bias gradient of node j on the final layer
    - * Where *y* is the correct value of node *j* and *a* is the actual output
  + Calculating the bias gradient of node *j* on layer *l* from the layer in front, *l+1*
    - * Where W is the vector of weights from node *j* to its successors, and it is dotted with the vector of bias gradients of the successors
  + Calculating the weight gradient of a neuron occurs as follows: