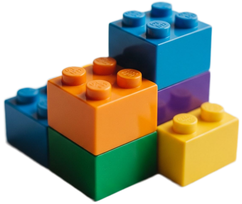
HZDR HELMHOLTZ ZENTRUM DRESDEN ROSSENDORF HELMHOLTZ AI

Neural Network Building Blocks

Maleeha Hassan
Helmholtz AI



AI vs ML vs DL

Artificial Intelligent

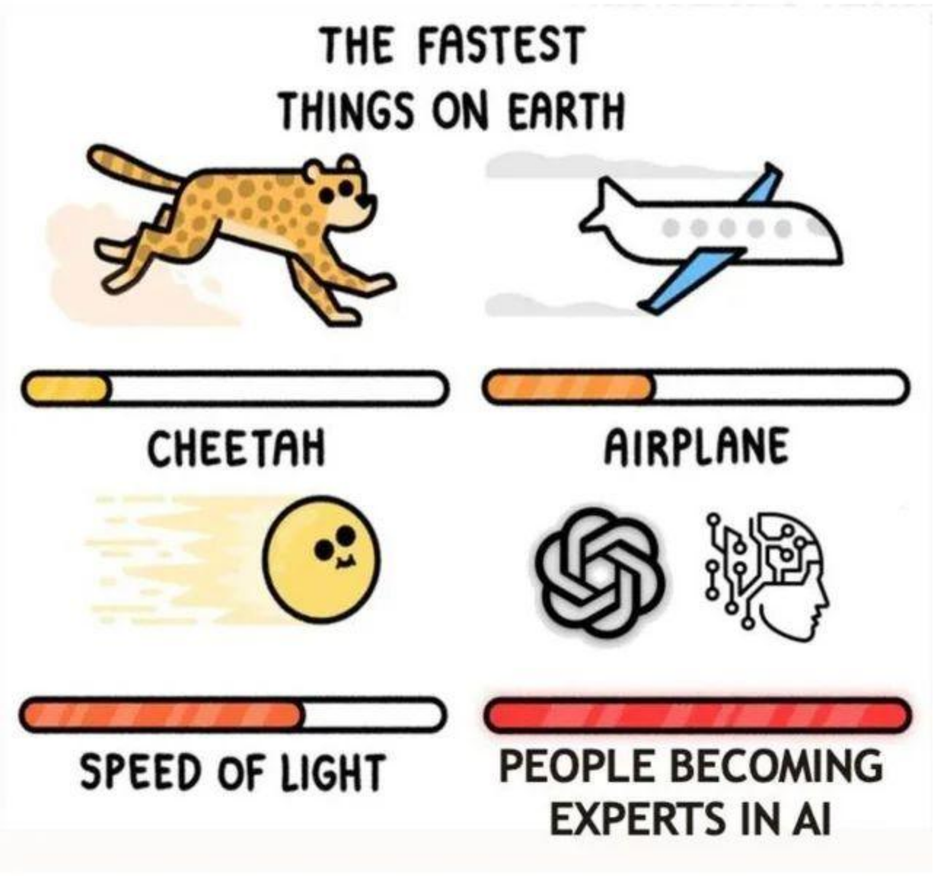
- Creation of machines that can mimic human intelligence

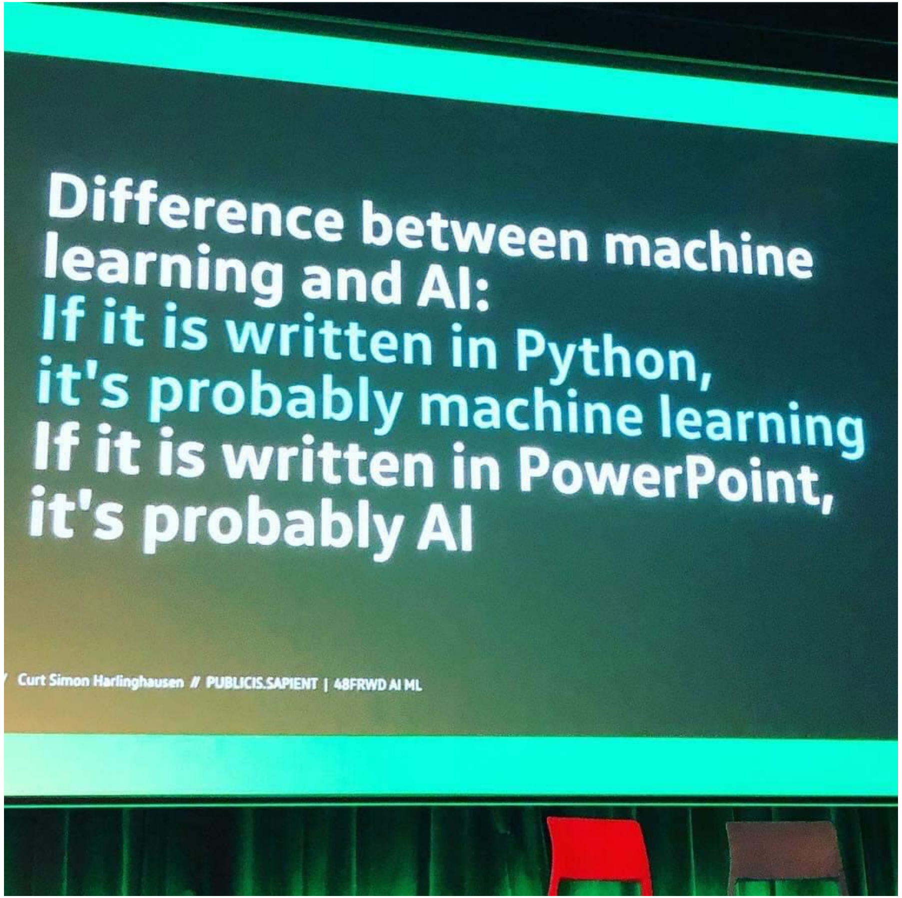
Machine Learning

- Allows machines to learn from data without explicitly being programmed

Deep Learning

- Uses artificial NNs to train models on big data


THE FASTEST
THINGS ON EARTH
CHEETAH
AIRPLANE
SPEED OF LIGHT
PEOPLE BECOMING
EXPERTS IN AI


Difference between machine learning and AI:
If it is written in Python, it's probably machine learning
If it is written in PowerPoint, it's probably AI
Curt Simon Harlinghausen // PUBLICIS.SAPIENT | 48FRWD AI ML

Goal

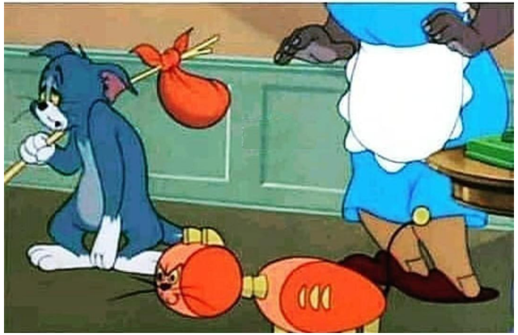
- Creating an algorithm to achieve a certain task for example image recognition
- Historically one had to invent algorithms
- Nowadays one trains a generalized algorithm that is already invented and implemented

Some examples of AI

AlphaFold

Accelerating breakthroughs in biology with AI

The first victim of
Artificial Intelligence



SCIENCE

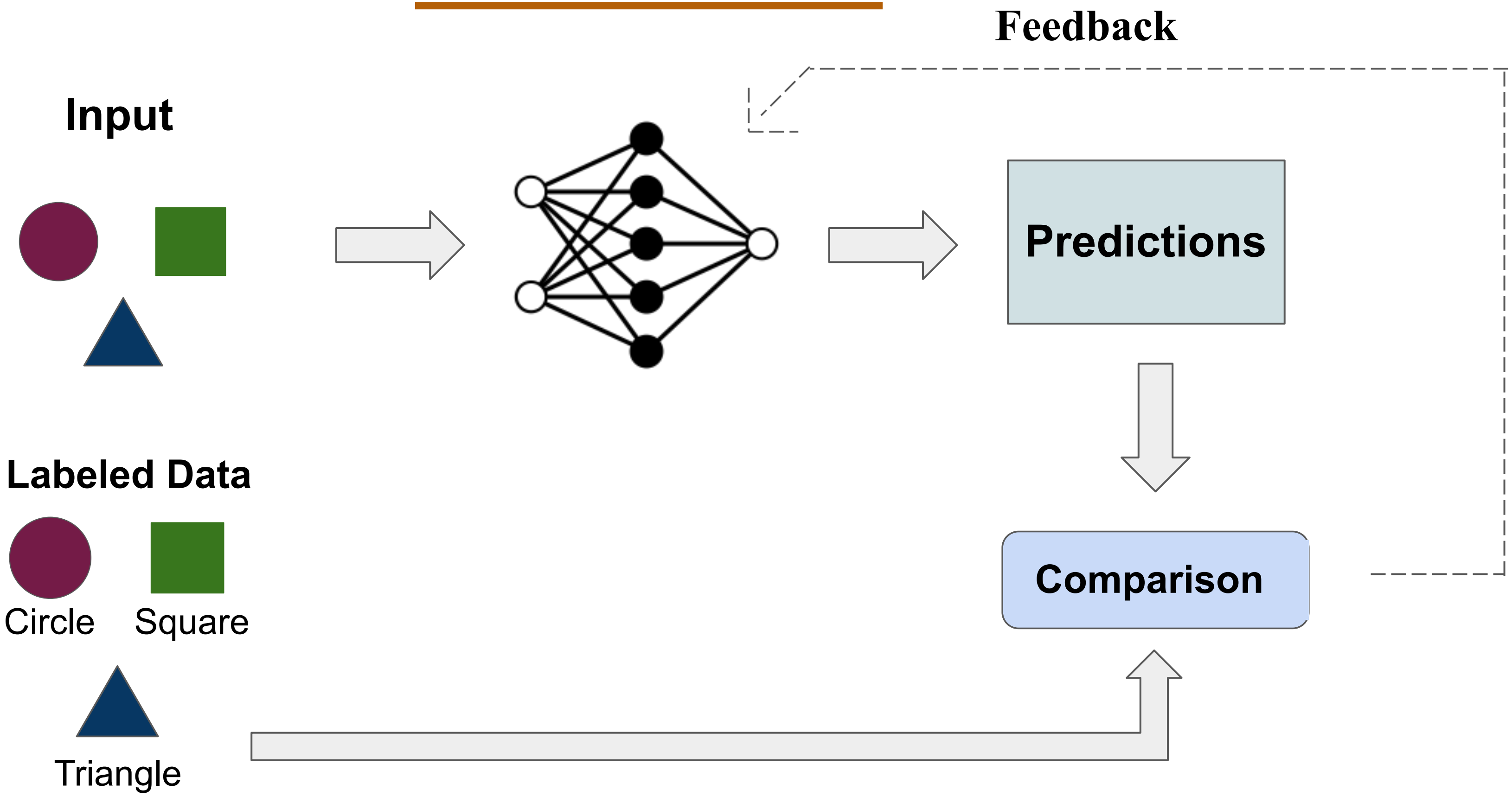
GraphCast: AI model for faster and more accurate global weather forecasting

DINOv3: Revolutionary Self-Supervised Vision Model

Machine Learning Vocabulary

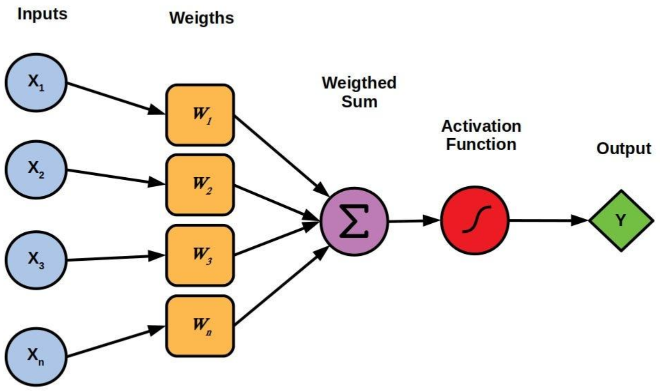
Supervised Learning	Unsupervised Learning	Reinforcement Learning
Learns a mapping from labeled inputs to target outputs.	Discovers structure in unlabeled data.	Learns decisions by interacting with an environment to maximize reward.
Classification, regression.	Clustering, dimensionality reduction.	Control, game playing, robotics.
Labeled examples (X,y).	Unlabeled examples (X).	No fixed dataset; experiences of states, actions, rewards.
Generalizes from labels to predict outputs on new data.	Groups, compresses, or reveals latent structure.	Learns a policy via trial-and-error.

Supervised learning

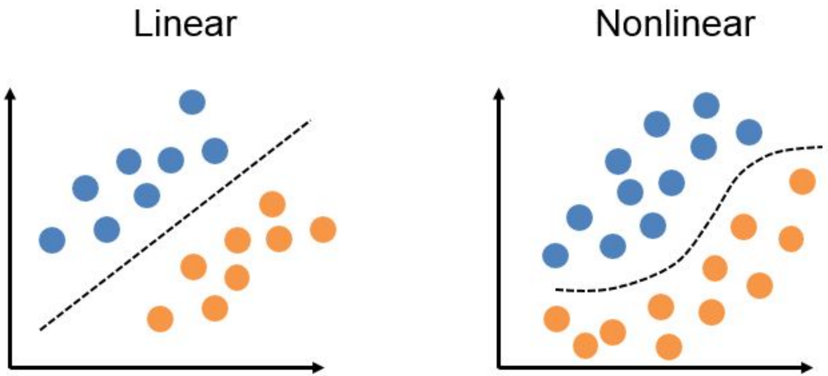


The Perceptron by Rosenblatt

A Perceptron is the simplest type of artificial neuron, introduced by Frank Rosenblatt in 1958.



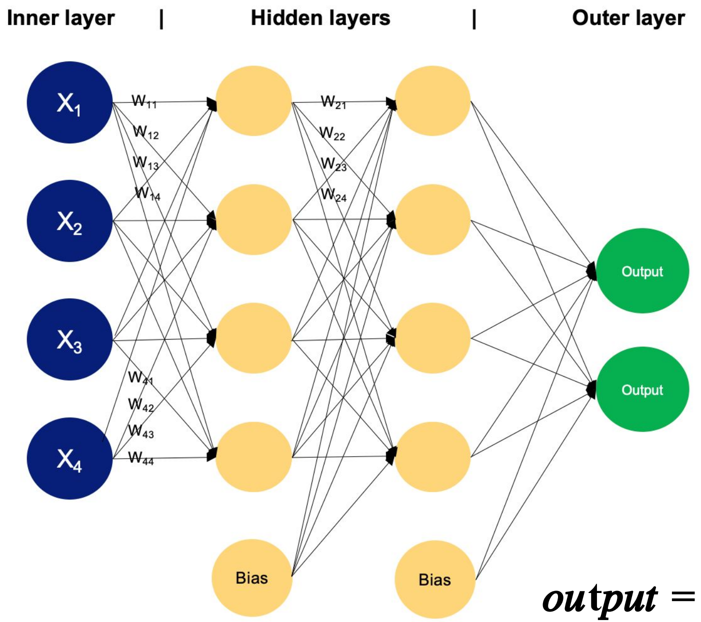
credits: starship-knowledge.com



https://github.com/jtsulliv/ML-from-scratch/blob/master/Neural-Networks/perceptron.ipynb

MultiLayer Perceptron

A multilayer perceptron (MLP) has multiple layers, including one or more hidden layers, allowing it to learn and represent more complex.

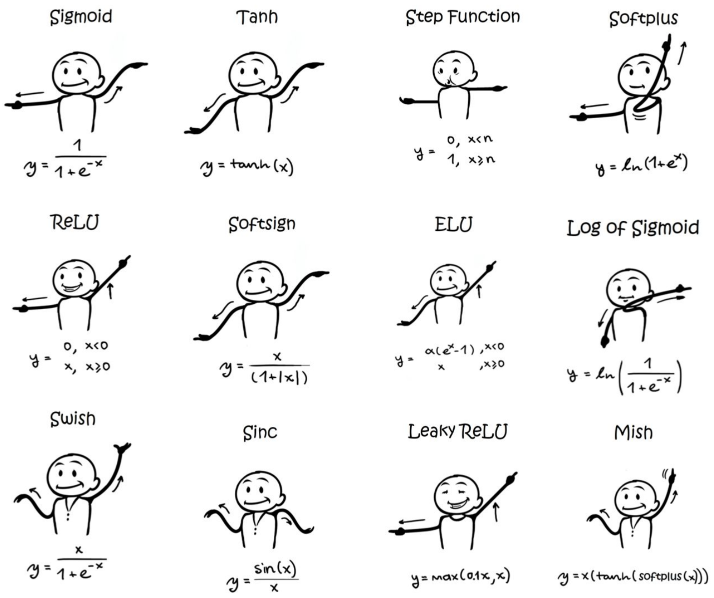


$$output = Activation\ (\sum(xi*wi)+bias)$$

Activation Functions

Activation Functions

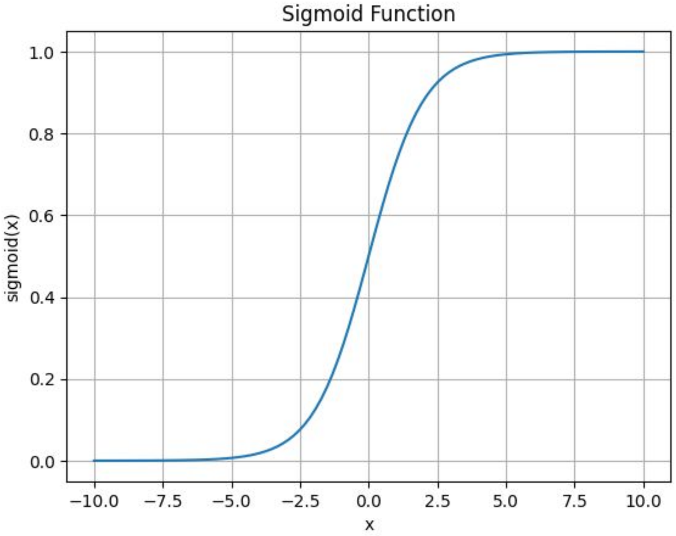
- Activation function decides whether a neuron should be activated by calculating the weighted sum of inputs and adding a bias term.
- Activation functions introduce non-linearity to neural networks.



https://sefiks.com/2020/02/02/dance-moves-of-deep-learning-activation-functions

Sigmoid Activation Function

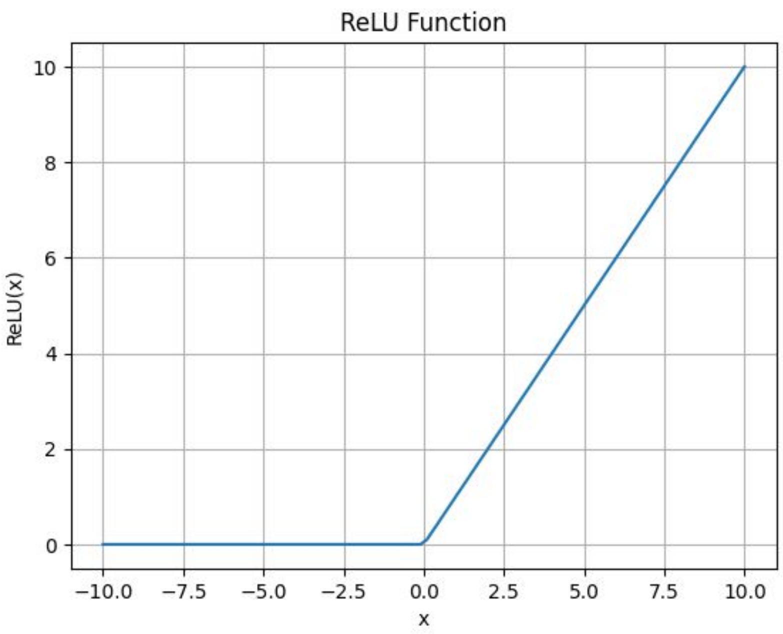
- values are between 0 and 1.
- Example: Output = 0.85 → 85% chance this is a cat.
- The derivative of sigmoid is $\sigma'(x) = \sigma(x) \cdot (1 - \sigma(x))$.
- Mainly used in binary classification.
- Has vanishing gradient problem (network stops learning when inputs are large or too small).



$\sigma(x) = 1 / (1 + e^{-x})$

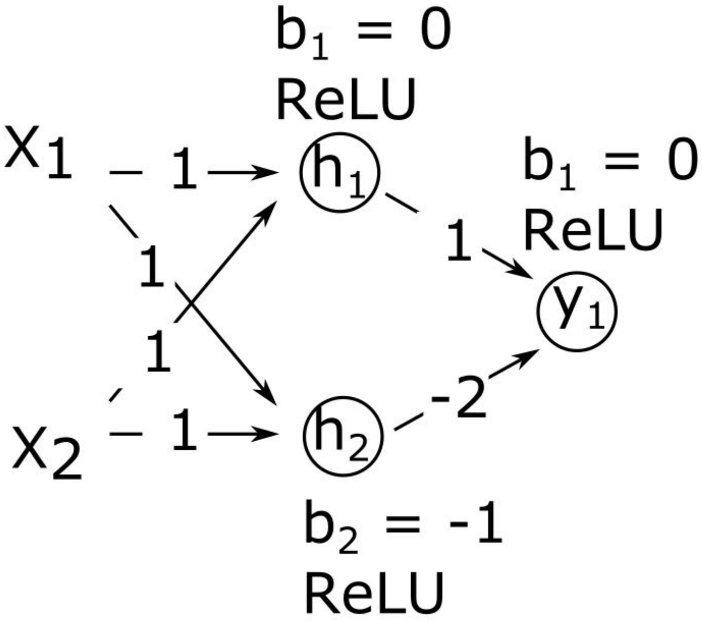
Rectified Linear Units (ReLU)

- If the function receives any negative input, it returns 0; however, if the function receives any positive value x, it returns that value.
- Doesn’t saturate for positive values which avoids vanishing gradient problem.
- Fast to compute (no exponentials).
- Dying ReLU problem → if a neuron always gets negative inputs, it outputs 0 forever and becomes dead.
- Leaky ReLU fixes the problem by allowing a small curve when x<0.



$$f(x) = \max(0,x)$$

NEURAL NETWORK CALCULATIONS

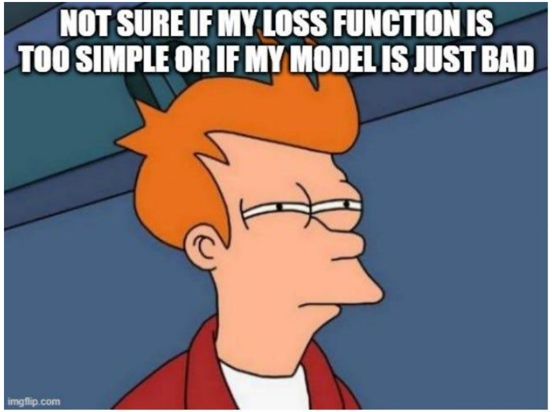


x1	x2	y
0	0	..
0	1	..
1	0	..
1	1	..

Now that we've seen how activation functions turn numbers into meaningful predictions — *How do we measure whether those predictions are correct?*

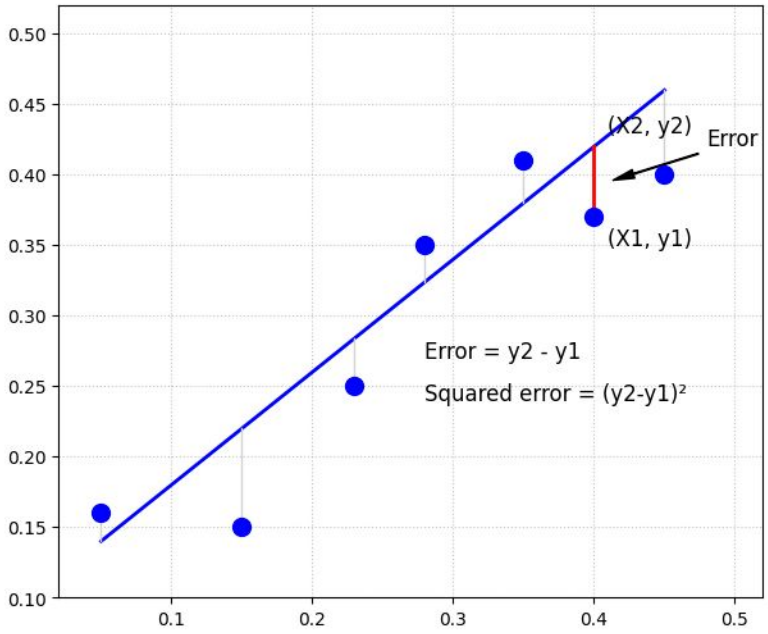
Loss Function

- A loss function (also called a cost function or error function) measures how wrong the model's prediction is compared to the true answer.
- Input is prediction + true label and output is a single number (loss).
- Provides the *learning signal* (error) for backpropagation.
- The goal of training is to minimize loss.
- The choice of loss function depends on the task at hand.



Loss Function

- **Regression losses**
  - Mean Absolute Error (L1)
    - MAE = (1/n) * Σ|actual - predicted|
  - Mean Squared Error (L2)
    - MSE = (1/n) * Σ|actual - predicted|^2
- **Classification losses**
  - Binary Cross Entropy (BCE)
    - Loss = - (y * log(p) + (1 - y) * log(1 - p))
    - - (1/N) * Σ [y_i * log(p_i) + (1 - y_i) * log(1 - p_i)]
  - Categorical Class Entropy (CCE)



Helping Materials

- Youtube Channels
  - 3blue1brown
  - StatQuest
  - Digital Sreeni

- Websites and Blogs
  - Introduction to Deep Learning
  - Understanding Deep Learning
  - https://gombru.github.io/2018/05/23/cross_entropy_loss/
  - Neural Networks, Manifolds, and Topology -- colah's blog
  - ConvNetJS demo: Classify toy 2D data
  - Neural networks: Activation functions | Machine Learning | Google for Developers

Hands On

https://github.com/maleehahassan/NNBuildingBlocksTeachingPt1