Linear Statistical Modeling Methods with SAS

Decision Trees, Random Forests and Neural Networks

Xuemao Zhang
East Stroudsburg University

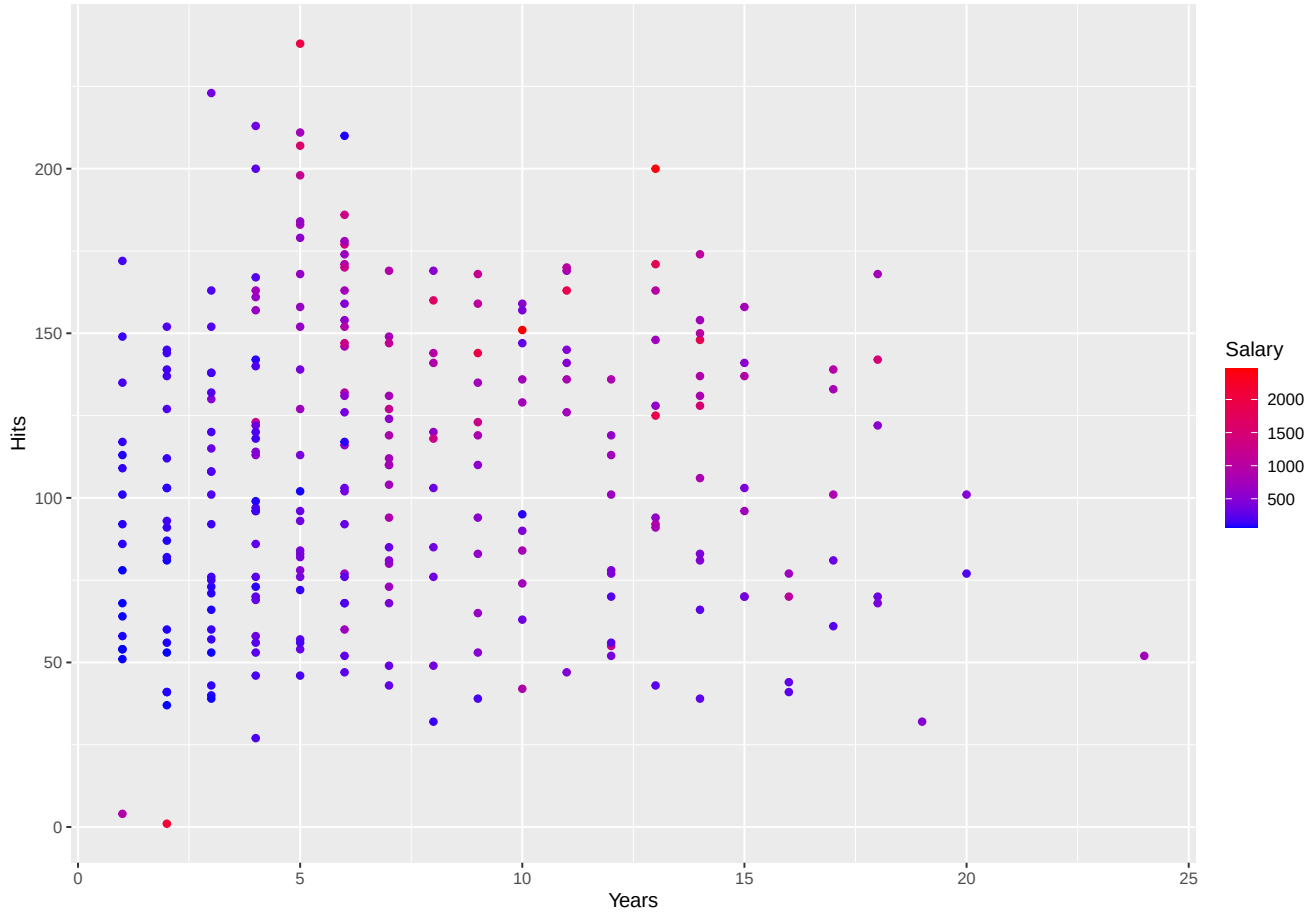
April 24, 2024

Outline

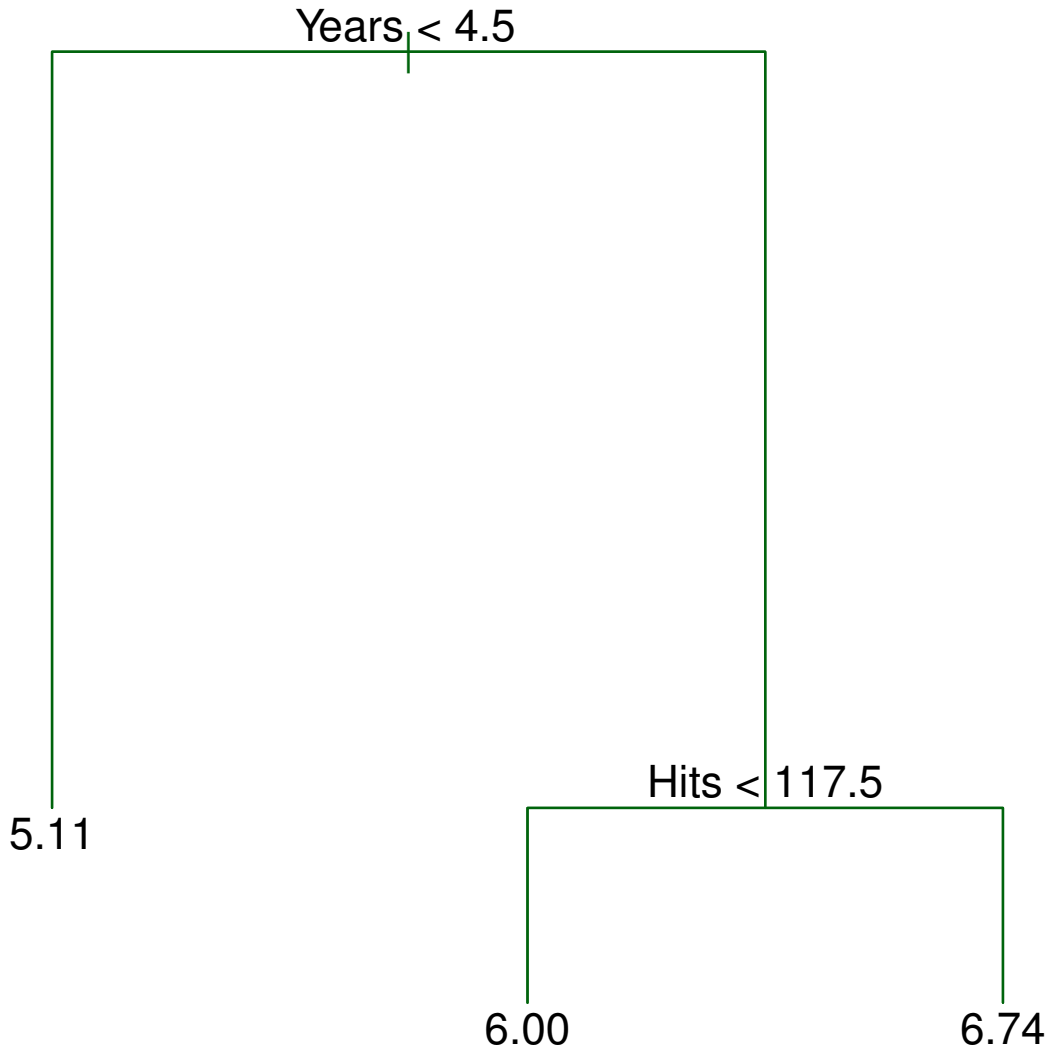
- Decision Trees
- Random Forests
- Neural Networks/Deep Learning
  - We introduce the basic idea of the three methods before we check machine learning in SAS Viya

Decision Trees

- Decision trees can be applied to both regression and classification problems.
- We first consider a regression problem using the Baseball salary data `Hitters`. how would you stratify it?



Decision Trees

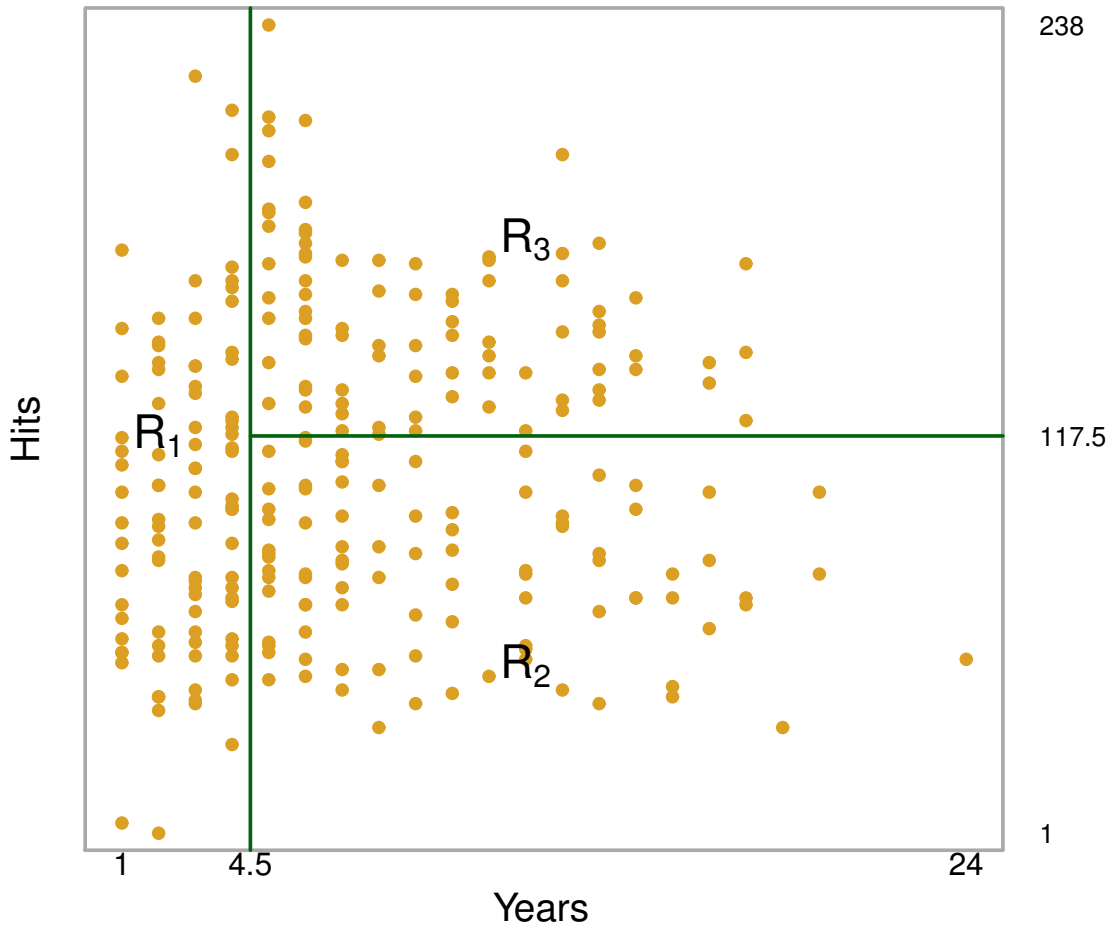


Decision Trees

- For the Hitters data, a regression tree for predicting the log salary of a baseball player, based on the number of years that he has played in the major leagues and the number of hits that he made in the previous year.
- At a given internal node, the label (of the form $X_j < t_k$) indicates the left-hand branch emanating from that split, and the right-hand branch corresponds to $X_j \geq t_k$. For instance, the split at the top of the tree results in two large branches. The left-hand branch corresponds to `Years`<4.5, and the right-hand branch corresponds to `Years`$>=4.5$.
- The tree has two internal nodes and three terminal nodes, or leaves. The number in each leaf is the mean of the response for the observations that fall there.

Decision Trees

Overall, the tree stratifies or segments the players into three regions of predictor space: $R_1 = \{X | Years < 4.5\}$, $R_2 = \{X | Years >= 4.5, Hits < 117.5\}$, and $R_3 = \{X | Years >= 4.5, Hits >= 117.5\}$.



Decision Trees

- `Years` is the most important factor in determining Salary, and players with less experience earn lower salaries than more experienced players.
- Given that a player is less experienced, the number of `Hits` that he made in the previous year seems to play little role in his `Salary`.
- But among players who have been in the major leagues for five or more years, the number of `Hits` made in the previous year does affect `Salary`, and players who made more `Hits` last year tend to have higher salaries.
- Surely an over-simplification, but compared to a regression model, it is easy to display, interpret and explain

Decision Trees

Terminology for Trees

- In keeping with the tree analogy, the regions R_1, R_2, and R_3 are known as *terminal nodes*
  - **Predictions**: For every observation that falls into the region $R_j, j = 1, 2, 3$, we make the same prediction.
- Decision trees are typically drawn *upside down*, in the sense that the leaves are at the bottom of the tree.
- The points along the tree where the predictor space is split are referred to as *internal nodes*
  - In the hitters tree, the two internal nodes are indicated by the text `Years`<4.5 and `Hits`<117.5.

Decision Trees

Classification Trees

- Very similar to a regression tree, except that it is used to predict a qualitative response rather than a quantitative one.
- For a classification tree, we predict that each observation belongs to the **most commonly occurring class** of training observations in the region to which it belongs.

Decision Trees

Advantages and Disadvantages of Trees

- Trees are very easy to explain to people. In fact, they are even easier to explain than linear regression!
- Trees can be displayed graphically, and are easily interpreted even by a non-expert (especially if they are small).
- Trees can easily handle qualitative predictors without the need to create dummy/categorical variables.
- Unfortunately, trees generally do not have the same level of predictive accuracy as some of the other regression and classification approaches we have seen.
  - However, by aggregating many decision trees, the predictive performance of trees can be substantially improved.

Bagging

- Bagging (Bootstrap aggregation) is a general-purpose procedure for reducing the variance of a machine learning method
- It is particularly useful and frequently used in the context of decision trees.
- Averaging a set of observations reduces variance.
  - Recall variance of sample means in CLT.
  - But this is not practical because we generally do not have access to multiple training sets.
  - Instead, we can **bootstrap**, by taking **repeated samples** from a **single training data set**.
- The technique of *Bagging* can be applied to regression trees and classification trees.

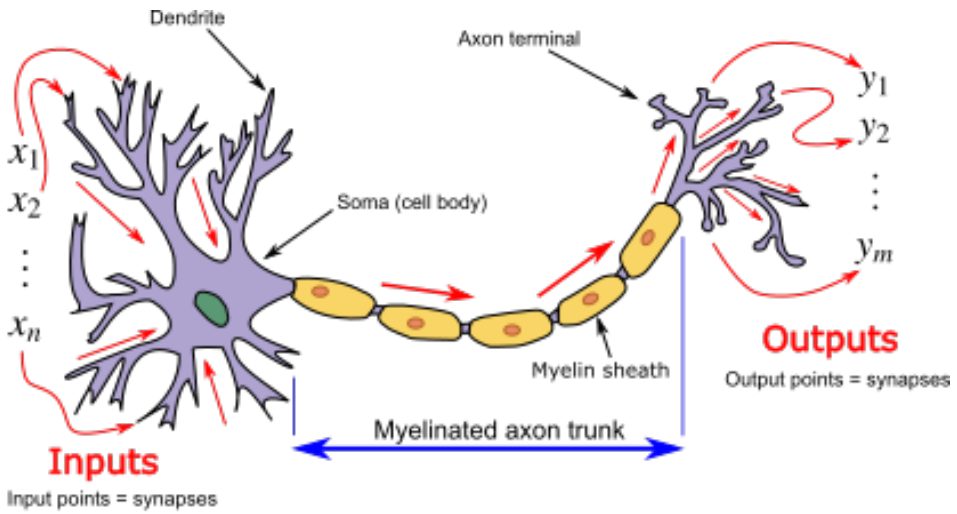
Random Forests

- **Random forests** provide an improvement over bagged trees by way of a small tweak that de-correlates the trees. This reduces the variance when we average the trees.
- As in bagging, we build a number of decision trees on *bootstrapped training samples*.
- But when building these decision trees, each time a split in a tree is considered, a **random selection of m predictors** is chosen as split candidates from the **full set of p predictors**. The split is allowed to use only one of those m predictors.
  - A fresh selection of m predictors is taken at each split, and typically we choose $m \approx \sqrt{p}$.

Introduction to Neural Networks/Deep Learning

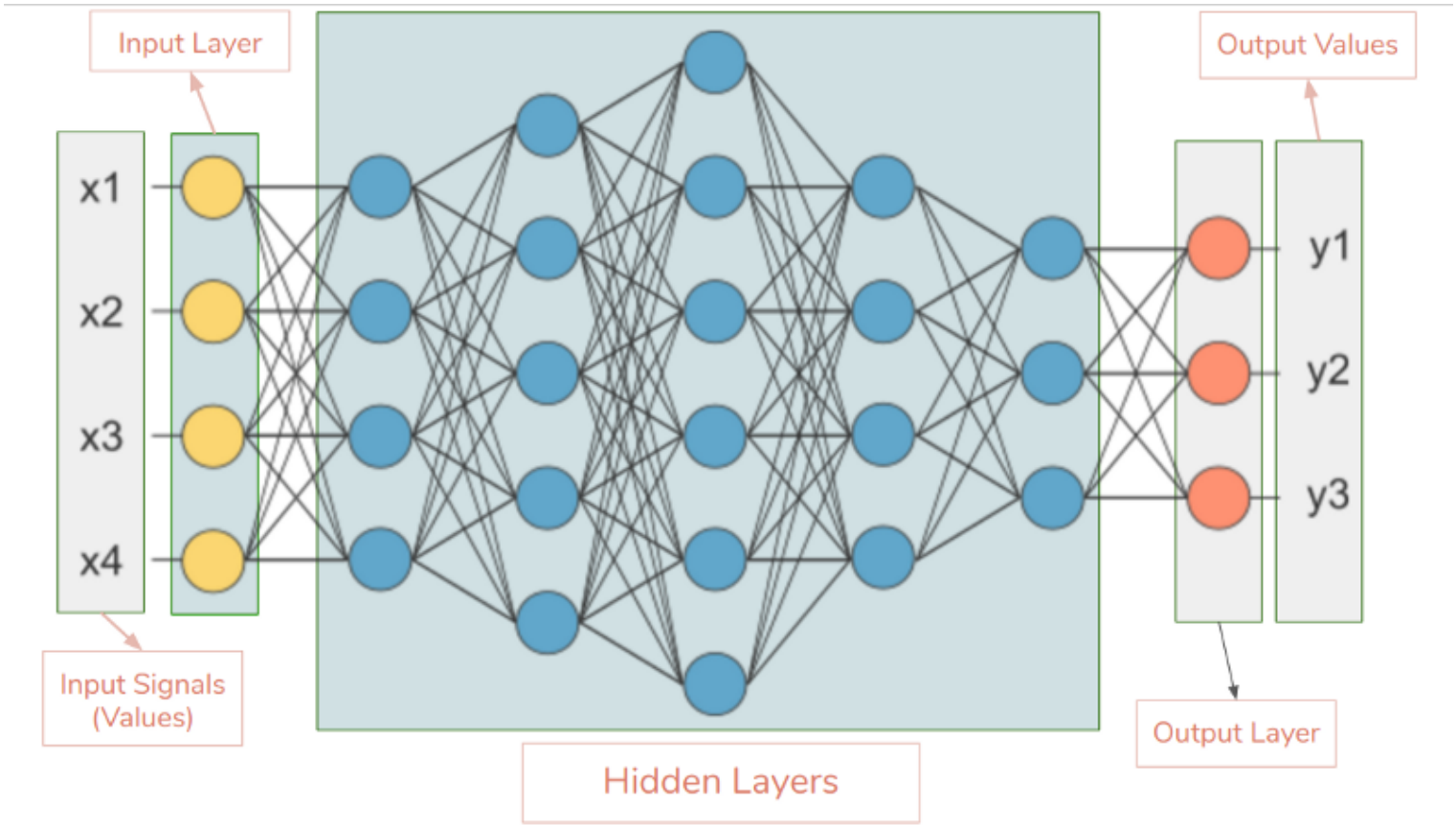
A Biological Neuron: In human brain, billions of neurons interact with each other.

- Dendrite: Receives signals from other neurons
- Soma: Processes the information
- Axon: Transmits the output of this neuron
- Synapse: output points



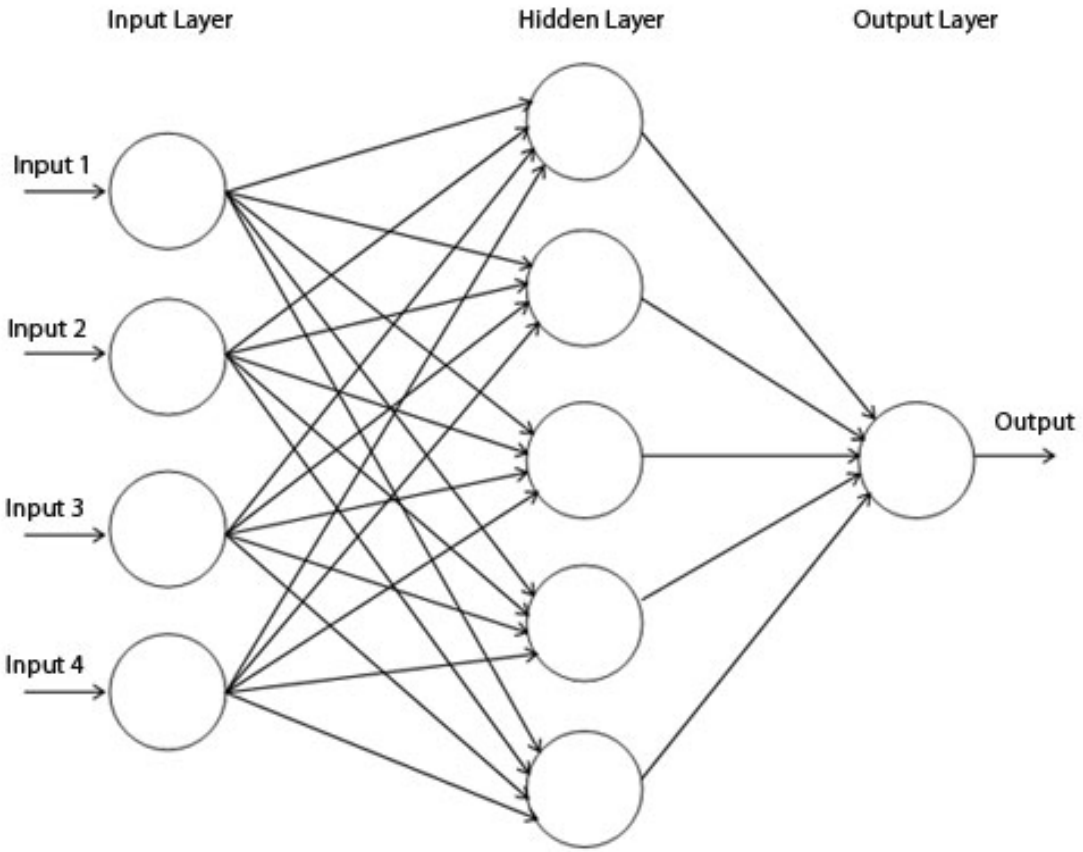
Introduction to Neural Networks/Deep Learning

- Idea is to replicate neurons in brain through Artificial Neuron.
- These artificial neurons interact with each other.

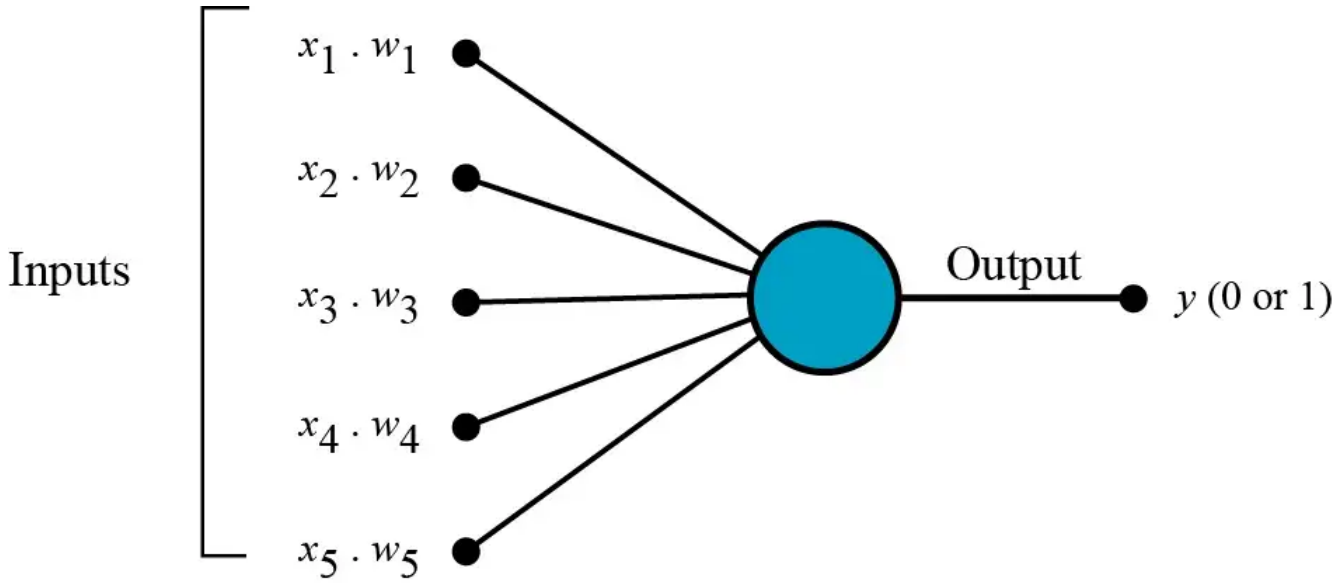


Introduction to Neural Networks/Deep Learning

- 1950's: *Perceptron*, first neuron was developed by Rosenblatt: Single layer neural network.

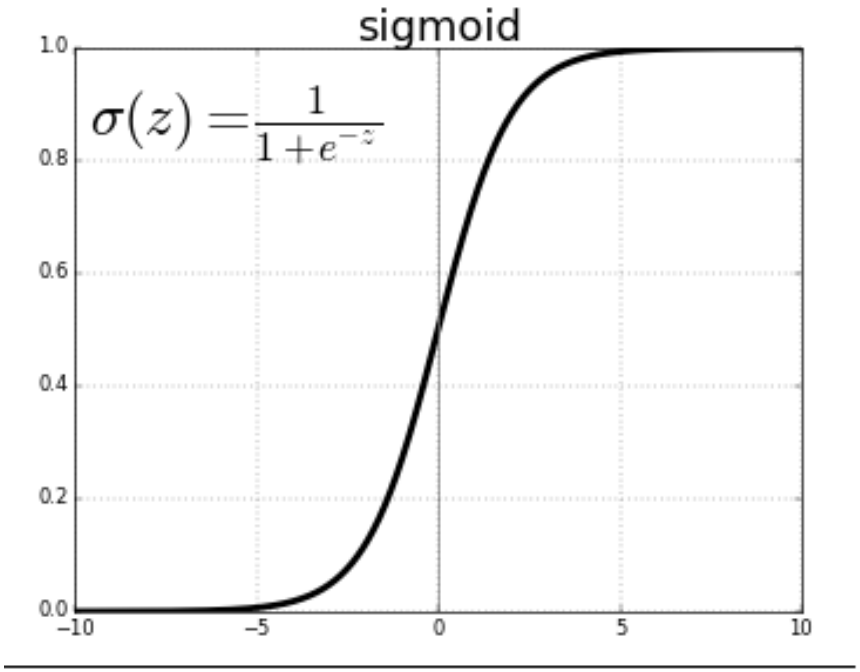


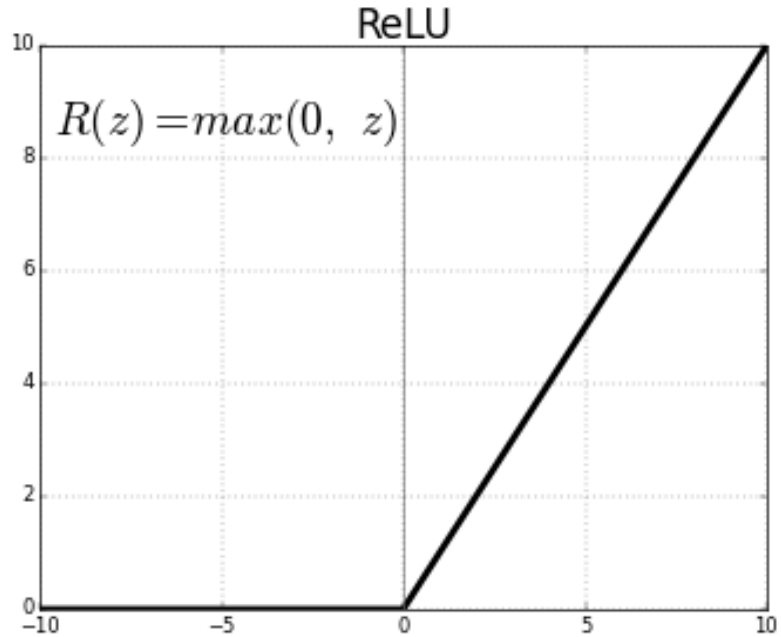
Introduction to Neural Networks/Deep Learning



Introduction to Neural Networks/Deep Learning

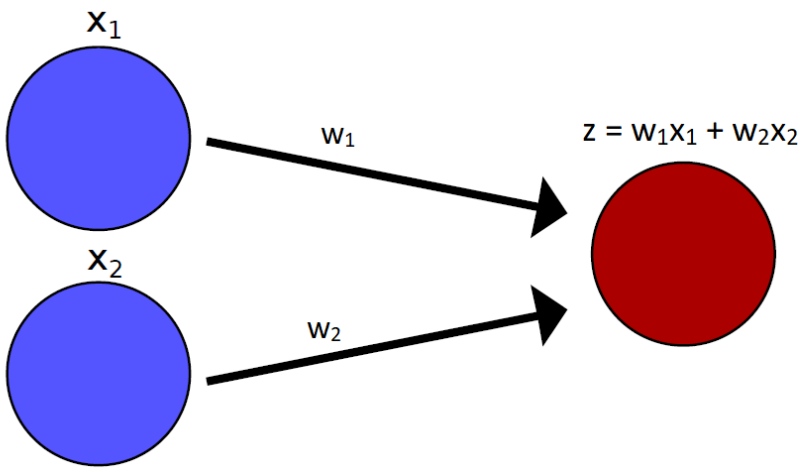
- How does a *Perceptron* work?
- Apply an **Activation Function** S to the weighted sum, and the output y is resulted from a Sigmoid function or ReLU (a step function):





Introduction to Neural Networks/Deep Learning

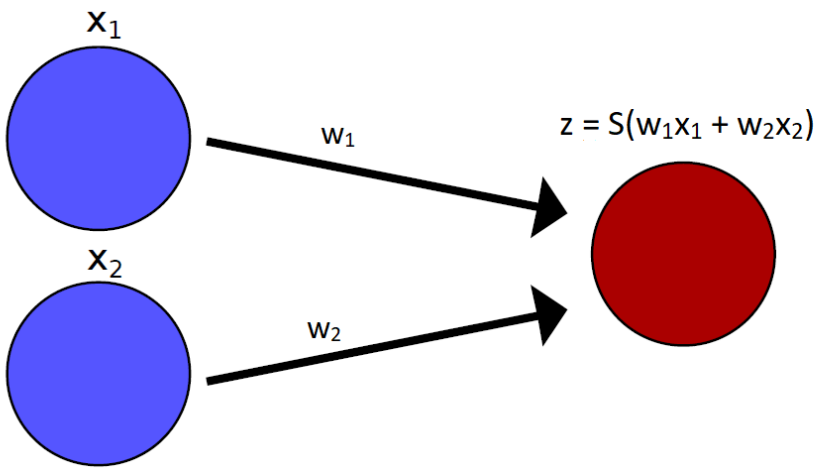
- A *Perceptron* without an **Activation Function**:



$$f_{w_1,w_2}(x_1, x_2) = w_1 x_1 + w_2 x_2$$

Introduction to Neural Networks/Deep Learning

- The **Activation Function** generally is nonlinear.



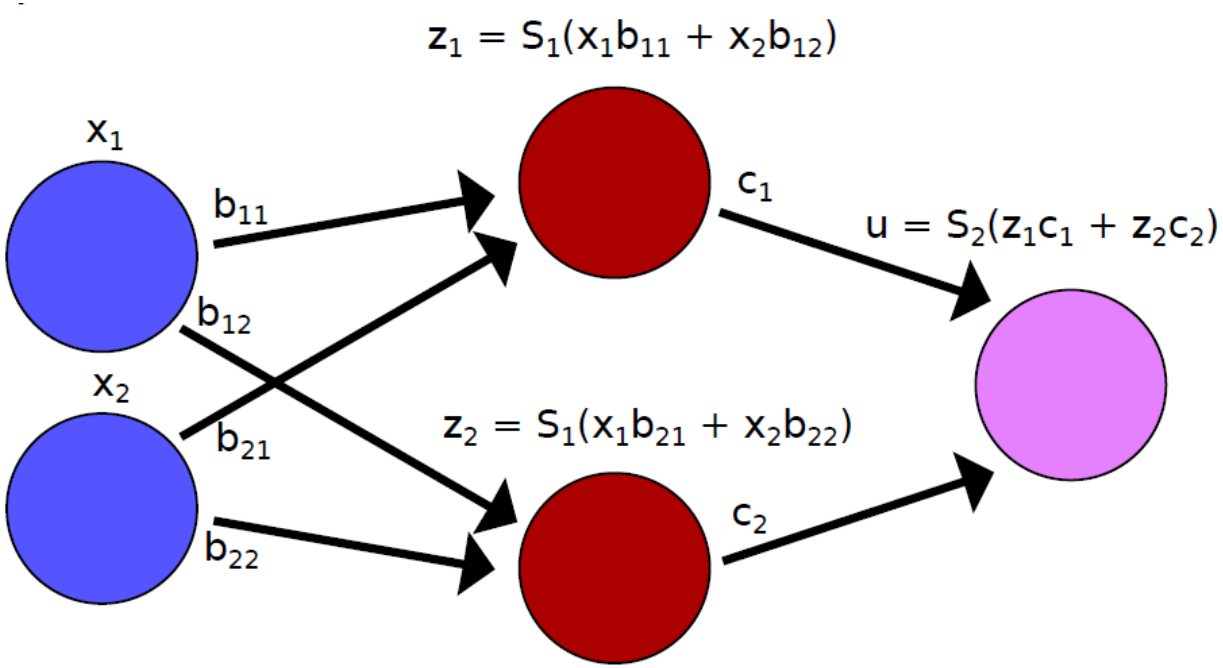
$$f_{w_1,w_2}(x_1, x_2) = S(w_1x_1 + w_2x_2)$$

Introduction to Neural Networks/Deep Learning

- The above simple examples connect NNs back to simple objects: linear or logistic regression without harnessing the power of neural networks.
- 1970's: The Quiet Years Limitations of Perceptron was demonstrated by Minsky and Papert (1969).
- 1980's: Renewed Interest in neural networks: Geoffrey Hinton et. al (1986) proposed a **multilayer** neural network and demonstrated the BP(backpropagation) algorithm.
  - Godfather of Deep learning.
  - At the time, he was PostDoc at UCSD.
  - Emeritus Prof. at University of Toronto; Worked at Google Brain; Chief scientific advisor of the Vector Institute in Toronto.

Introduction to Neural Networks/Deep Learning

- The power of neural networks comes through hidden layers.
- In the following, b_{11}, b_{12}, b_{21} and b_{22} are the weights for the first layer, c_1 and c_2 are the weights for the hidden layer.



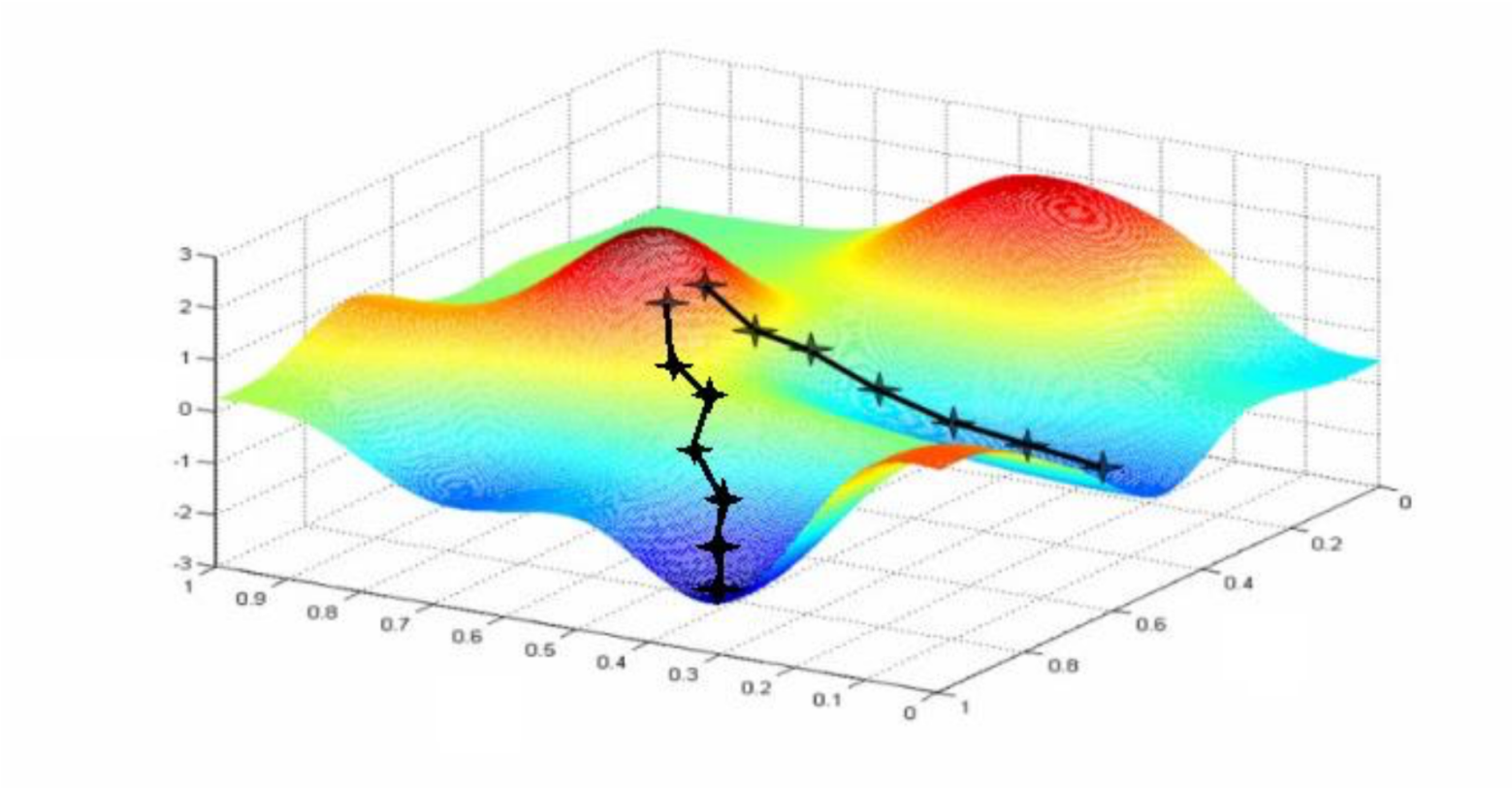
Introduction to Neural Networks/Deep Learning

- The structure of a multilayer NN:
  - Number of hidden layers
  - Number of nodes per layer
  - Types of activation functions
- For a given structure of a NN, let θ denote the vector of coefficient values. We can think of the NN as a **function** $f_{\theta}(\mathbf{x})$
- We choose the θ-value that best fits our data!

$$\hat{\theta} = \arg min_{\theta} \sum_{i=1}^{n} (y_i - f_{\theta}(\mathbf{x_i}))^2$$

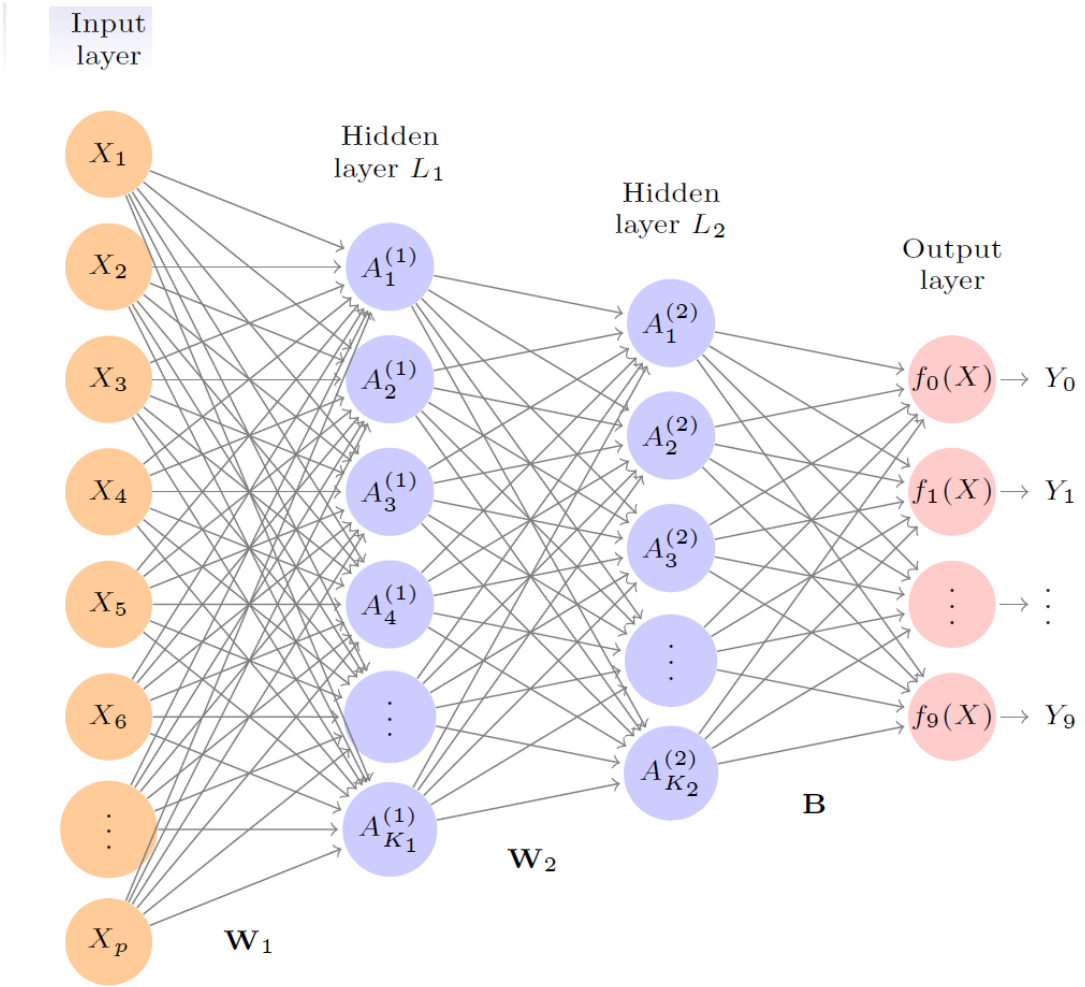
Introduction to Neural Networks/Deep Learning

- The minimizer $\hat{\theta}$ is obtained by the BP algorithm which uses gradient descent!



Introduction to Neural Networks/Deep Learning

- Example: Neural network diagram with two hidden layers and multiple outputs, suitable for the MNIST handwritten-digit problem. The input layer has $p = 784$ units, the two hidden layers $K1 = 256$ and $K2 = 128$ units respectively, and the output layer 10 units. Along with intercepts (referred to as biases in the deep-learning community) this network has 235,146 parameters (referred to as weights).



Introduction to Neural Networks/Deep Learning

- 1989: ConvNet, Yann Lecun came up with CNN (Convolutional Neural Network).
  - Prof. at New York University
  - Chief AI Scientist at Meta
  - Independently discovered BP Algorithm.
  - LeCun received the 2018 Turing Award (often referred to as "Nobel Prize of Computing"), together with Yoshua Bengio and Geoffrey Hinton, for their work on deep learning.
- Research in Neural Networks died between 1990 and 2012 because
  - Required a lot of data.
  - Computation Intensive.

Introduction to Neural Networks/Deep Learning

- It all began in 2012. The algorithm existed since 1990's, so why?
- The two problems solved
  - Lots of Data - with the help of Internet/Mobile Devices
  - Lots of computational power - GPU
- Most common variations of neural network architectures are:
  - Multilayer perceptron(MLP)
  - Convolutional Neural Network(CNN)
  - Recurrent Neural Network(RNN)

Introduction to Neural Networks/Deep Learning

- Multilayer Perceptron:
  - We build most of our fundamental understanding with Multilayer Perceptron(MLP)
- Convolutional Neural Network:
  - We will extend understanding of MLP into CNN.
  - CNN's are typically used in Images/Videos related problems.
  - Can be used to generate/draw images as well.
- Recurrent Neural Network:
  - Typically used to understand sequences, eg speech, text, etc.
  - It can even be used to generate music.
- Tranformer proposed in Attention is all you need (December 2017) is the foundation of Chat GPT(Generative pretrained transformer)

Applications of Artificial Neural Networks

- Visual Recognition
  - Recognize digits
  - Recognize letters, eg. Plate number
  - face recognition

Applications of Artificial Neural Networks

- Photo descriptions

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

Image Captioning

Applications of Artificial Neural Networks

- Speech recognition

AMAZON'S ALEXA GOOGLE'S ASSISTANT APPLE'S SIRI MICROSOFT'S CORTANA

Applications of Artificial Neural Networks

- Self-Driving Cars

Applications of Artificial Neural Networks

- Natural Language Processing
  - summarizing articles
  - machine translation
  - question answering
  - text classification
- Music composition
- Automatic Game Playing
- Applications in healthcare: predictive models, imaging techniques
- For more, see Top 26 Applications of Deep Learning in 2024

License