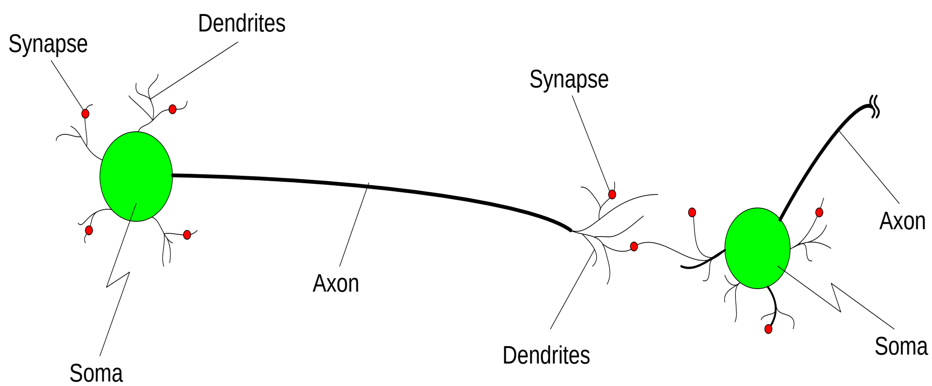
Chapter 6:

Techniques for Predictive Modeling

Neural Network Concepts

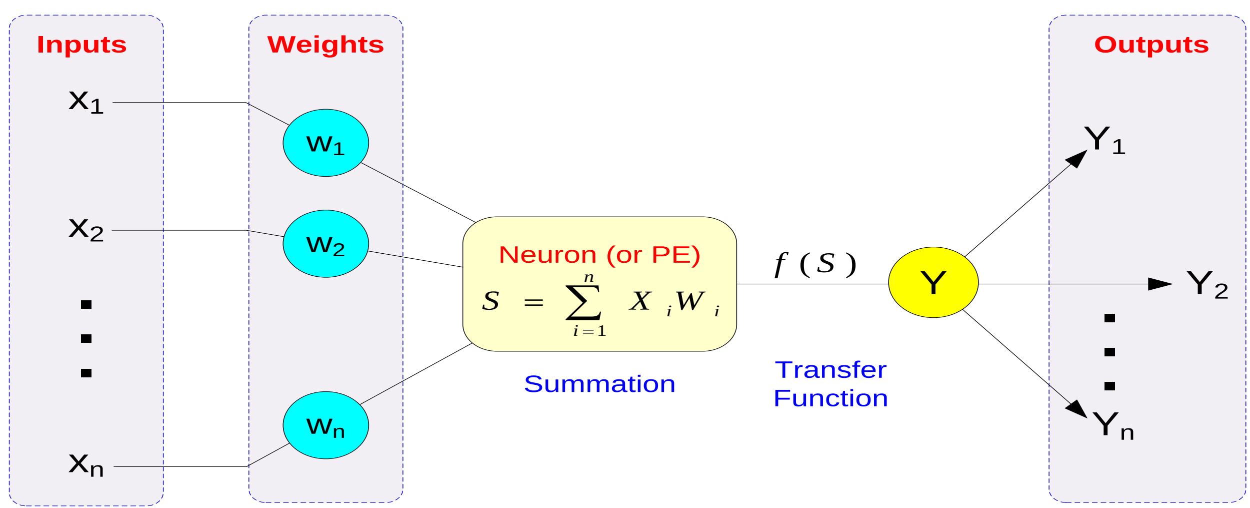
- Neural networks (NN): a brain metaphor for information processing
- Neural computing
- Artificial neural network (ANN)
- Many uses for ANN for
  - pattern recognition, forecasting, prediction, and classification
- Many application areas
  - finance, marketing, manufacturing, operations, information systems, and so on

Biological Neural Networks



- Two interconnected brain cells (neurons)

Processing Information in ANN



- A single neuron (processing element – PE) with inputs and outputs

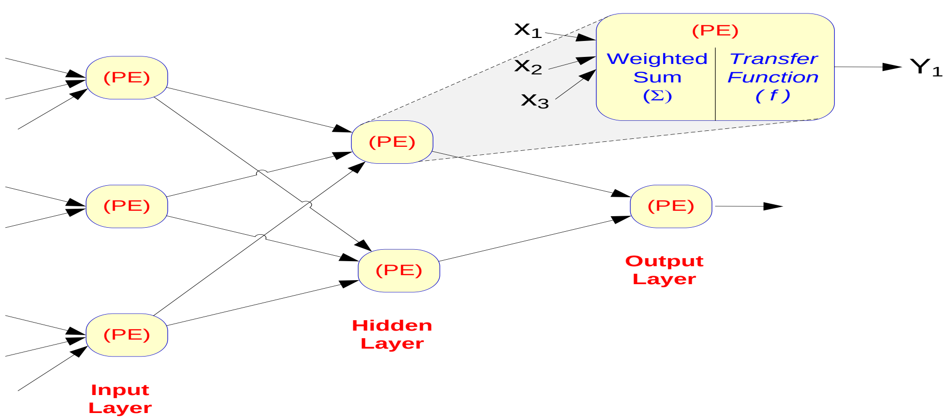
Biology Analogy

Biological versus	Artificial NNs
Soma	Node
Dendrites	Input
Axon	Output
Synapse	Weight
Slow	Fast
Many neurons (10^9)	Few neurons (~100s)

Elements of ANN

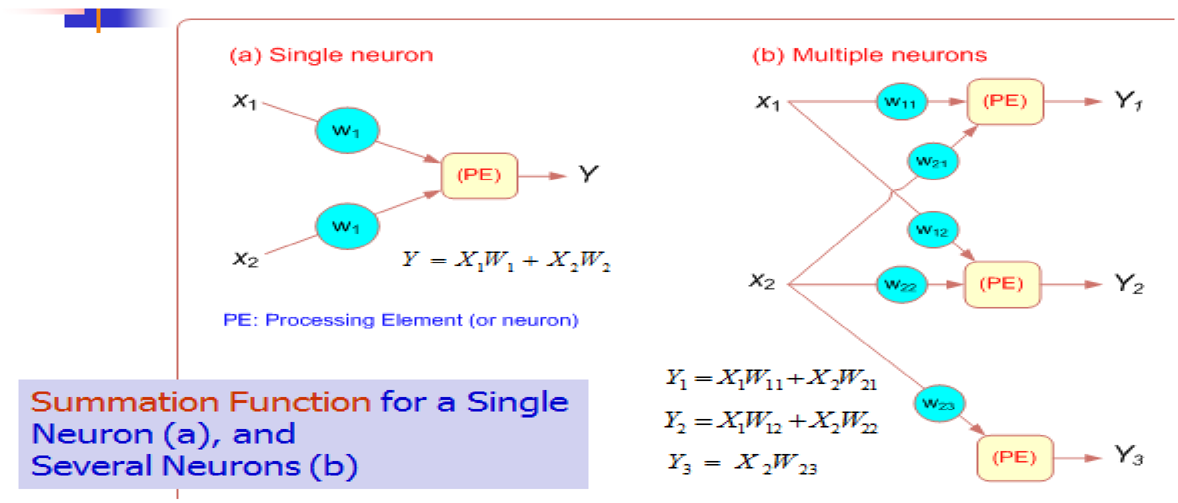
- Processing element (PE)
- Network architecture
  - Hidden layers
  - Parallel processing
- Network information processing
  - Inputs
  - Outputs
  - Connection weights
  - Summation function

Elements of ANN



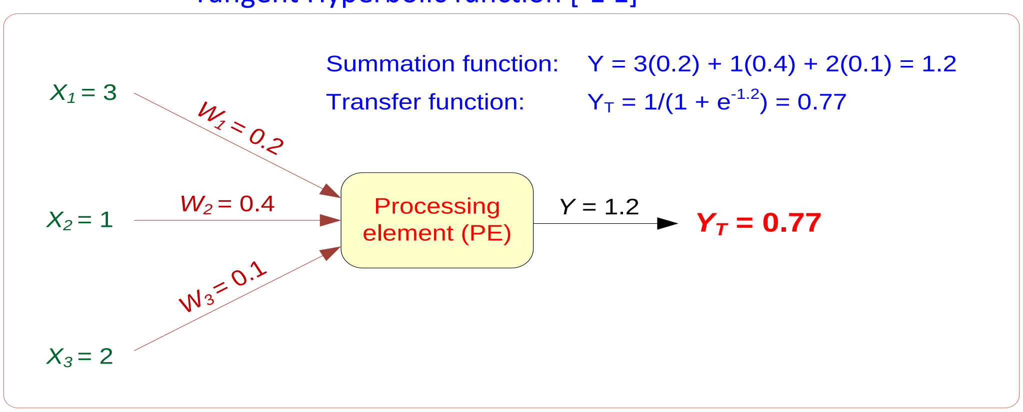
Neural Network with One Hidden Layer

Elements of ANN



Elements of ANN

- Transformation (Transfer) Function
  - Linear function
  - Sigmoid (logical activation) function [0 1]
  - Tangent Hyperbolic function [-1 1]

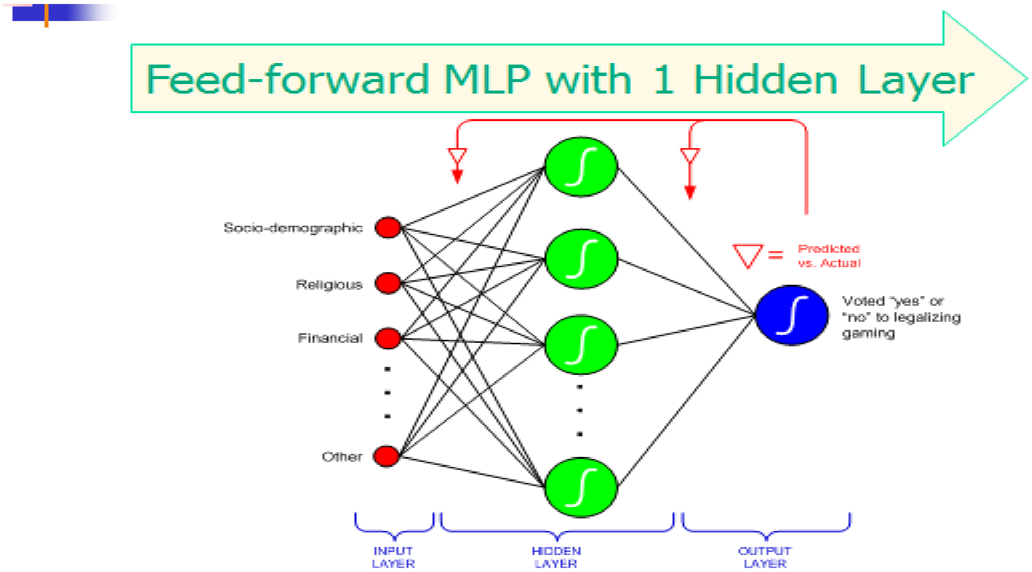


Neural Network Architectures

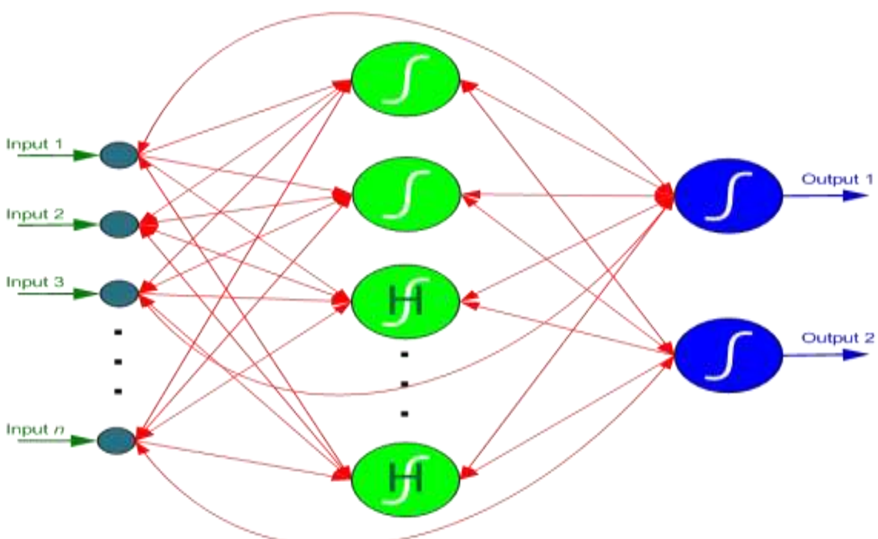
- Architecture of a neural network is driven by the task it is intended to address
  - Classification, regression, clustering, general optimization, association, ….
- Most popular architecture: Feedforward, multi-layered perceptron with backpropagation learning algorithm
  - Used for both classification and regression type problems
- Others – Recurrent, self-organizing feature maps, Hopfield networks, …

Neural Network Architectures

Feed-Forward Neural Networks



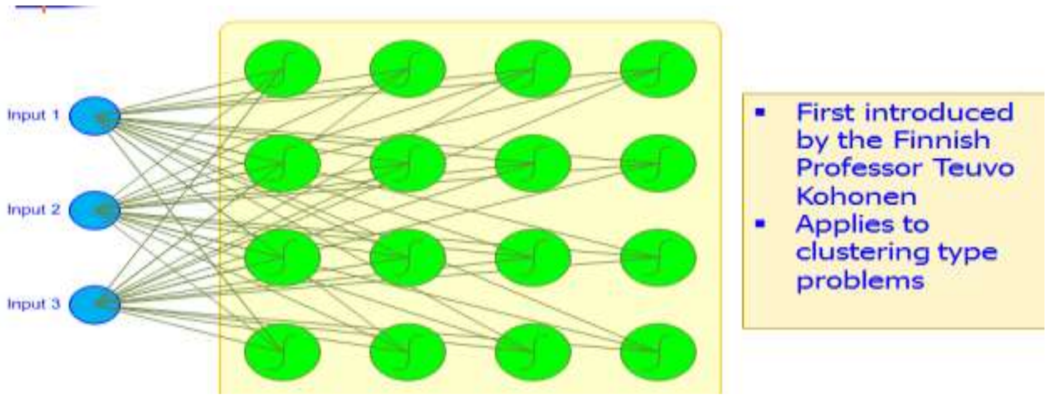
Neural Network Architectures Recurrent Neural Networks



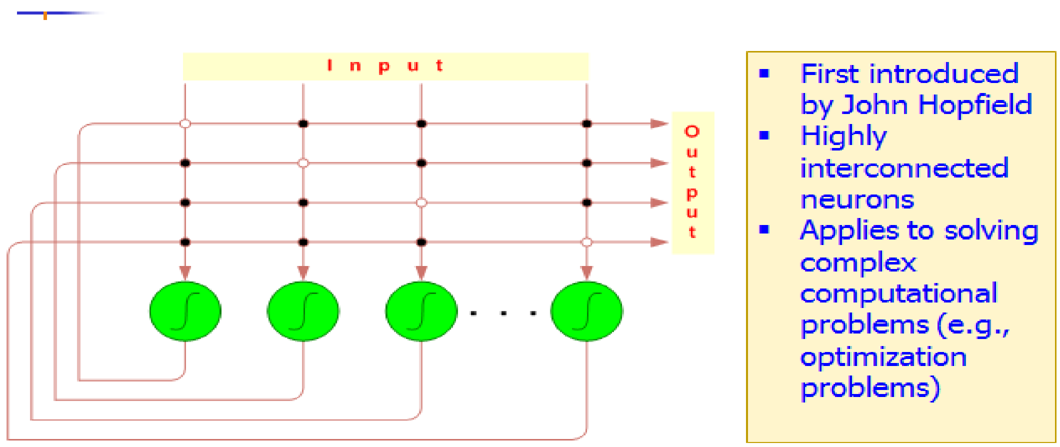
*H: indicates a "hidden" neuron without a target output

Other Popular ANN Paradigms

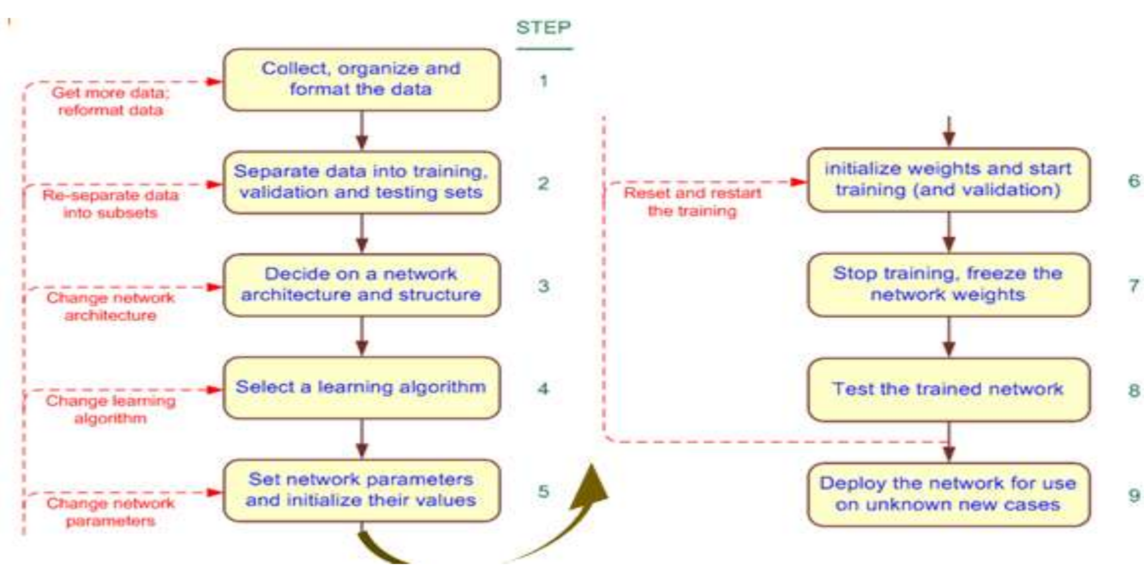
Self-Organizing Maps (SOM)



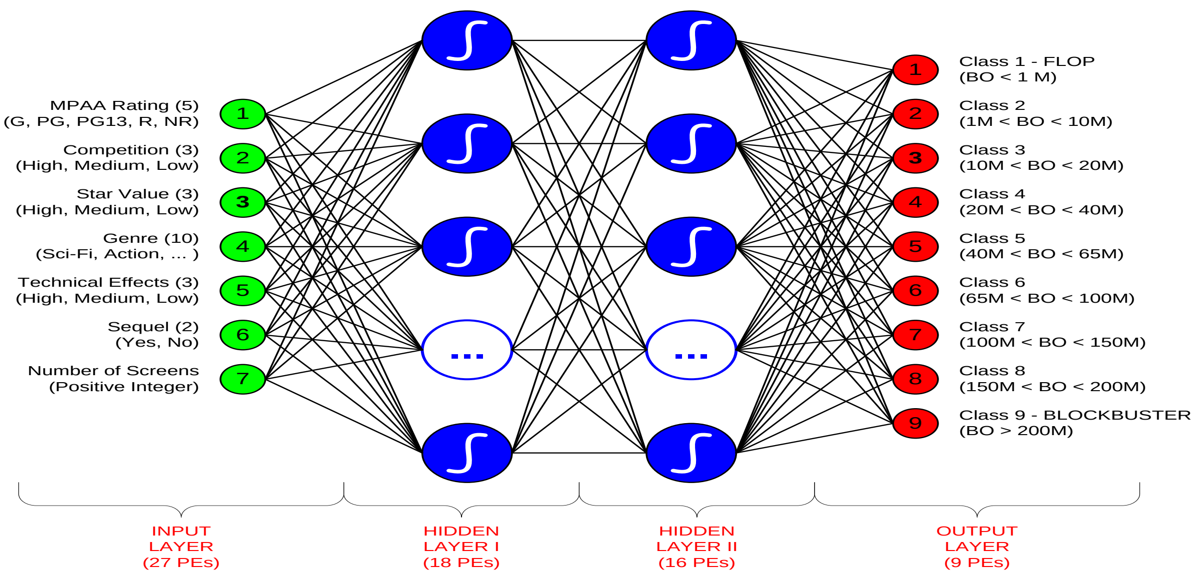
Other Popular ANN Paradigms Hopfield Networks



Development Process of an ANN



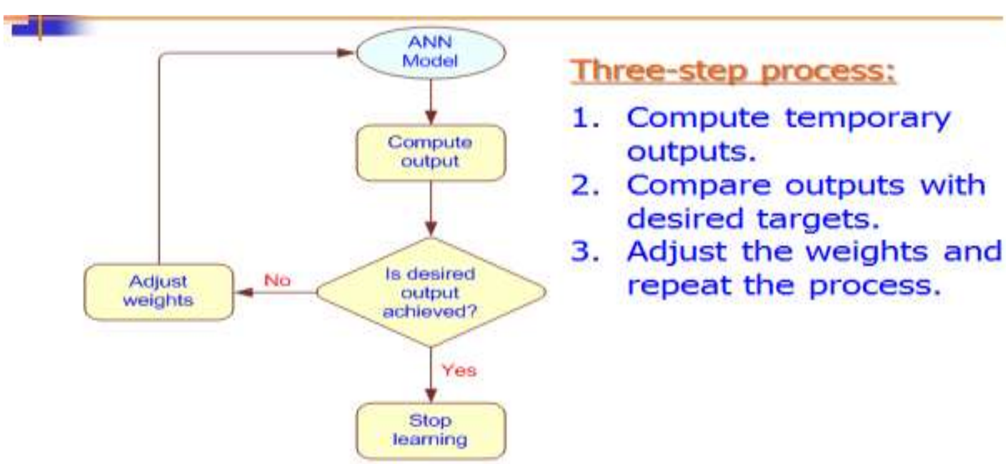
An MLP ANN Structure for the Box-Office Prediction Problem



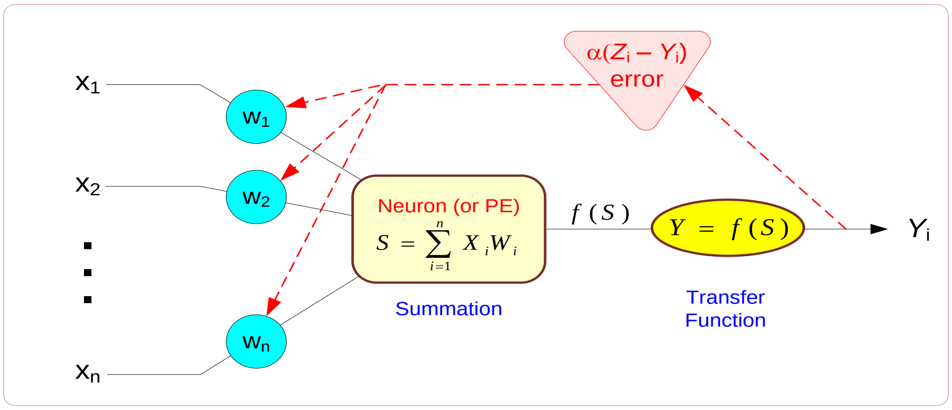
Testing a Trained ANN Model

- Data is split into three parts
  - Training (~60%)
  - Validation (~20%)
  - Testing (~20%)
  - *k*-fold cross validation
  - Less bias
  - Time consuming

AN Learning Process A Supervised Learning Process



Backpropagation Learning



- Backpropagation of Error for a Single Neuron

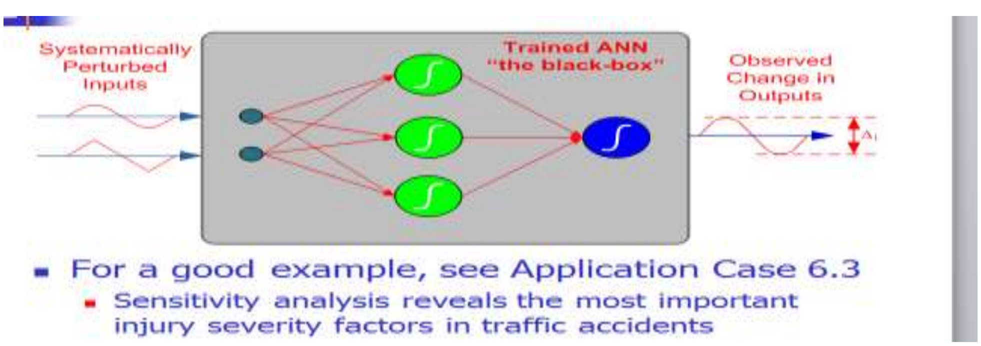
Backpropagation Learning

- The learning algorithm procedure
    1. Initialize weights with random values and set other network parameters
    2. Read in the inputs and the desired outputs
    3. Compute the actual output (by working forward through the layers)
    4. Compute the error (difference between the actual and desired output)
    5. Change the weights by working backward through the hidden layers
    6. Repeat steps 2-5 until weights stabilize

Illuminating The Black Box Sensitivity Analysis on ANN

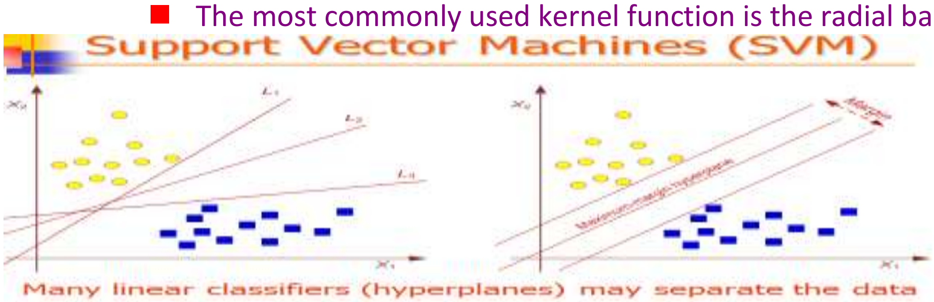
- A common criticism for ANN: The lack of transparency/explainability
- The black-box syndrome!
- Answer: sensitivity analysis
    - Conducted on a trained ANN
    - The inputs are perturbed while the relative change on the output is measured/recorded
    - Results illustrate the relative importance of input variables

Sensitivity Analysis on ANN Models



Support Vector Machines (SVM)

- SVM are among the most popular machine-learning techniques.
- SVM belong to the family of generalized linear models… (capable of representing non-linear relationships in a linear fashion).
- SVM achieve a classification or regression decision based on the value of the linear combination of input features.
- Because of their architectural similarities, SVM are also closely associated with ANN.
- Goal of SVM: to generate mathematical functions that map input variables to desired outputs for classification or regression type prediction problems.
    - First, SVM uses nonlinear kernel functions to transform non-linear relationships among the variables into linearly separable feature spaces.
    - Then, the maximum-margin hyperplanes are constructed to optimally separate different classes from each other based on the training dataset.
- SVM has solid mathematical foundation!
- A hyperplane is a geometric concept used to describe the separation surface between different classes of things.
    - In SVM, two parallel hyperplanes are constructed on each side of the separation space with the aim of maximizing the distance between them.
- A kernel function in SVM uses the kernel trick (a method for using a linear classifier algorithm to solve a nonlinear problem)
    - The most commonly used kernel function is the radial basis function (RBF).

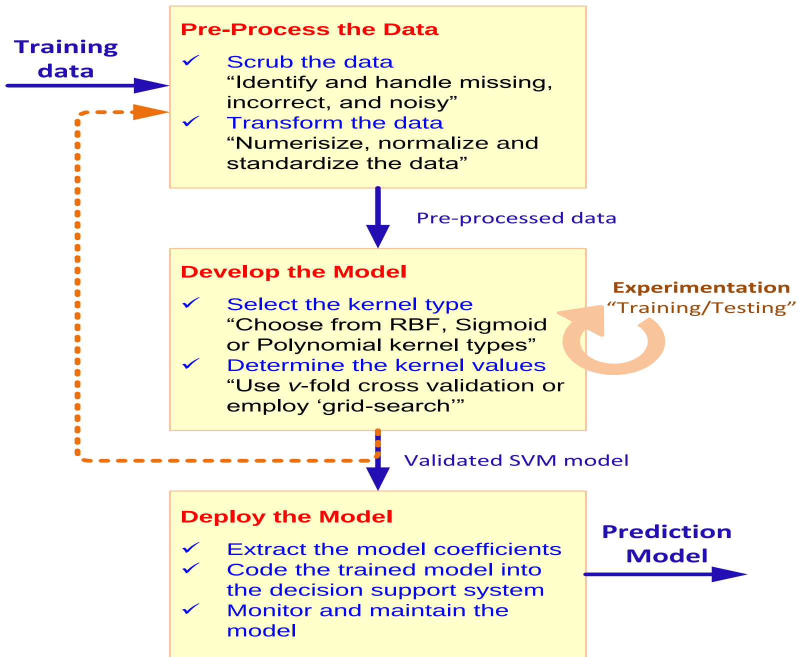


How Does an SVM Work?

- Following a machine-learning process, an SVM learns from the historic cases.
- The Process of Building SVM

1. Preprocess the data
    - Scrub and transform the data.
2. Develop the model.
    - Select the kernel type (RBF is often a natural choice).
    - Determine the kernel parameters for the selected kernel type.
    - If the results are satisfactory, finalize the model; otherwise change the kernel type and/or kernel parameters to achieve the desired accuracy level.
3. Extract and deploy the model.

The Process of Building an SVM



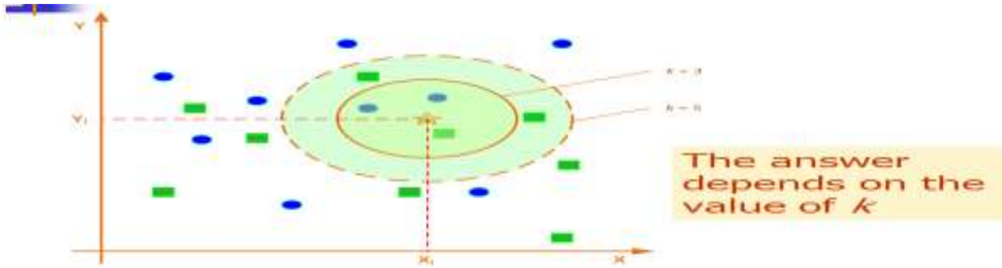
SVM Applications

- SVMs are the most widely used kernel-learning algorithms for wide range of classification and regression problems
- SVMs represent the state-of-the-art by virtue of their excellent generalization performance, superior prediction power, ease of use, and rigorous theoretical foundation
- Most comparative studies show its superiority in both regression and classification type prediction problems.
- SVM versus ANN?

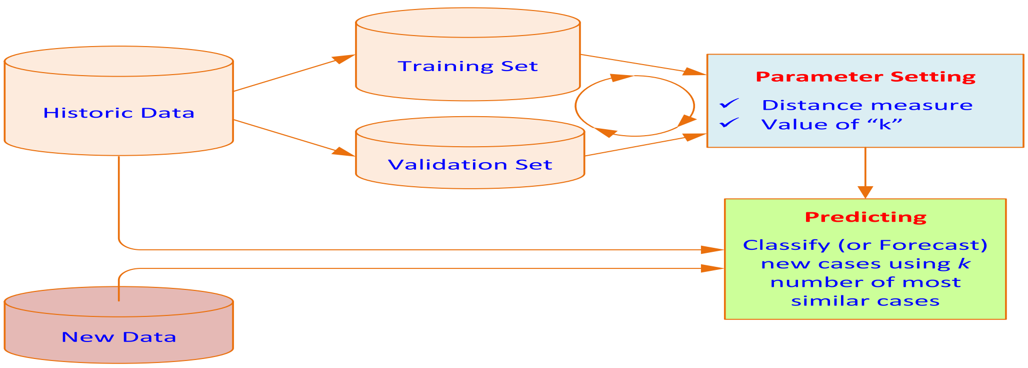
k-Nearest Neighbor Method (k-NN)

- ANNs and SVMs → time-demanding, computationally intensive iterative derivations
- *k*-NN is a simplistic and logical prediction method, that produces <u>very competitive</u> results
- *k*-NN is a prediction method for classification as well as regression types (similar to ANN & SVM)
- *k*-NN is a type of instance-based learning (or lazy learning) – most of the work takes place at the time of prediction (not at modeling)
- *k* : the number of neighbors used

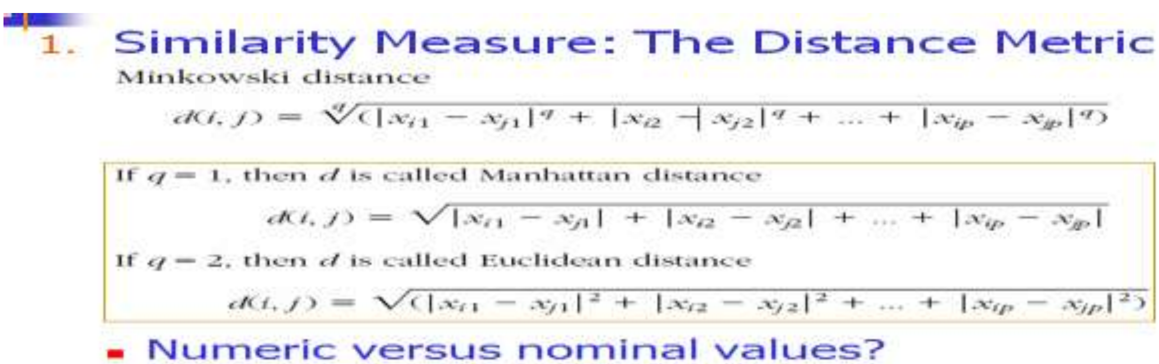
k-Nearest Neighbor Method (k-NN)



The Process of *k*-NN Method



k-NN Model Parameter



k-NN Model Parameter

2. Number of Neighbors (the value of *k*)
   - The best value depends on the data
   - Larger values reduce the effect of noise but also make boundaries between classes less distinct
   - An "optimal" value can be found heuristically
- Cross Validation is often used to determine the best value for k and the distance measure