>>> CS 7840 Soft Computing

>>> Machine Learning Algorithms

Name: Dr. Sanghui Han

Date: January 2, 2022

>>> Agenda

1. Neural Network Introduction
2. Neural Network
3. Components
4. Clustering Methods
5. Chaotic Methods
6. State-of-the-Art

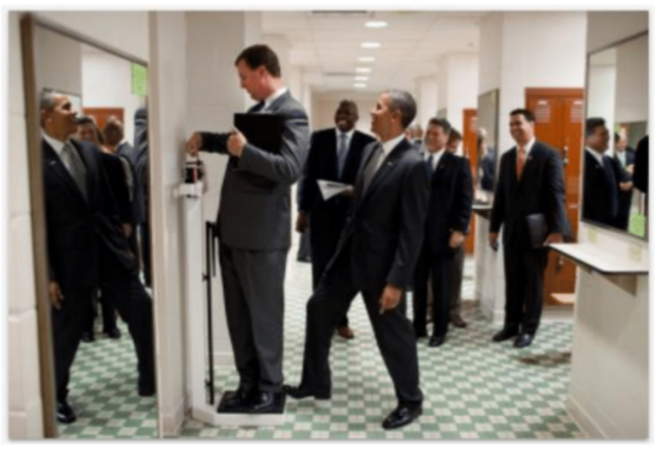
>>> Introduction

* Human brains consist of billions of neurons that allows us to learn and label an object we have not seen before based on its similarity to something we have.
* Automated image recognition is one of the primary methods of enabling Artificial Intelligence (AI) to process what is seen like human do.
* Computer vision is a very difficult problem for AI.
* The recent improvements in this area and its popularity has been due to a few key reasons:
    * Advances in Data Storage and Sharing
    * Increased Computing Power
    * Enhanced Algorithms

>>> Chunky Giraffe

What do you see in the picture?

>>> Humor and Mirrors



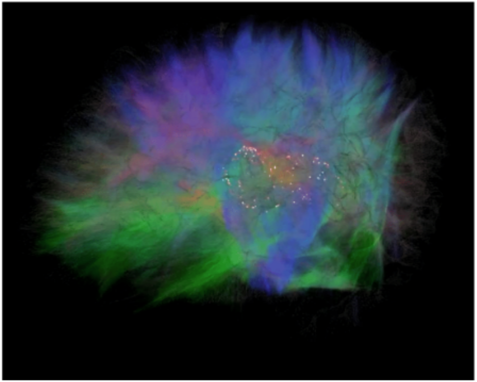
Interpret this image:
Mirrors, 3D Structure, Physics, Sparse Information, Humor,
What's on people's minds?, What happens next?

>>> Cat or Monkey

What does it take for a computer to interpret this image correctly?

>>> Neurons vs. Artificial Neural Network

Biological and Artificial Neural Networks



Human Brain

- Thalamocortical system:
  3 million neurons
  476 million synapses
- Full brain:
  100 billion neurons
  1,000 trillion synapses

Artificial Neural Network

- ResNet-152:
  60 million synapses

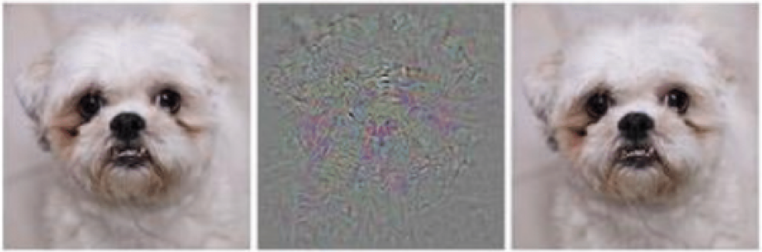
Human brains have ~10,000,000 times synapses than artificial neural networks.

>>> Vision is Old

Visual perception: 540,000,000 years of data

Bipedal movement: 230,000,000 years of data

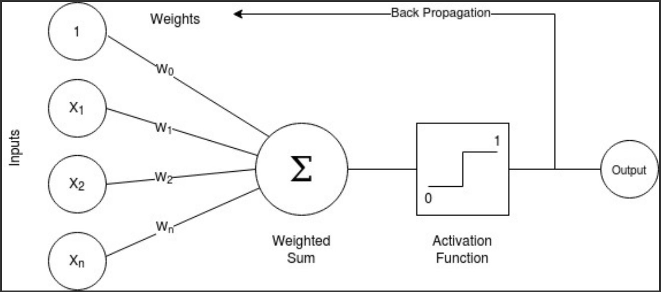
Abstract thought: 100,000 years of data



Prediction: **Dog** + Distortion Prediction: **Ostrich**

>>> Perceptron

* A perceptron is the first algorithmically described neural network.
* Frank Rosenblatt was a psychologist who inspired scientists to mimic the way the brain processes information using neural networks in 1958.
* A perceptron consists of multiple inputs and weights. The sum of the inputs and weights induces a local field to produce an output.



>>> Neural Network

* Neural networks have multiple layers that process inputs to make predictions.
* Different activation paths across the layers of a neural network can still lead to the same prediction. For example, a number in the MNIST dataset, different variations of the number 6 will have different activation paths, but produce the same output.

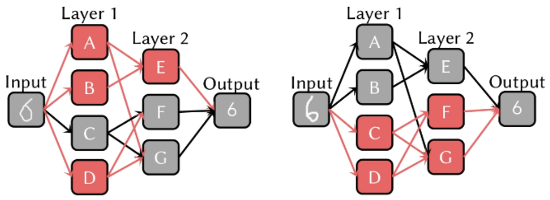


Figure 7. An example neural network fed with two different inputs but producing the same output. Activated neurons are in red.

>>> Components of a Neural Network

A typical Neural Network has four components:

* The Input Layer processes the pixels in an image. Each pixel is processed as a "neuron", but a convolution layer can identify features from a combination of pixels.
* The Hidden Layers transforms the input pixels that enter the network to increase their resemblance to other images with known labels. The number of neurons should be proportional to the number of pixels in the image.
* The Output Layer is the final prediction and consists of one neuron if there is one possible outcome. It has as many neurons as possible outcomes.
* The Loss Layer exists when a neural network is being trained. This layer gives feedback on whether the input was identified correctly or not to reinforce pathways that lead to the correct answer or re-calibrate their activation criteria.

>>> Activation Rules

* Neurons need to be activated in order to generate a prediction, and this is governed by an activation rule.
* The activation rule is fine-tuned during training to assign varying strengths to different associations, known as weights.
* A neuron is activated if it exceeds a certain threshold, which is determined by the sum of all the weights in a layer.

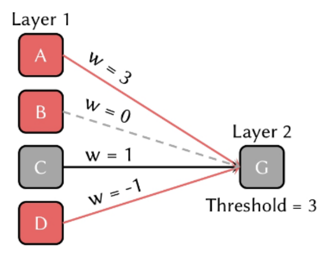


Figure 8. Example of a neuron's activation rule.

Was Neuron G activated?

>>> Activation Functions

* The induced local field of a neuron is defined as

$$v = \sum_{j=1}^{m} w_j x_j + b \tag{1}$$

  Here w_j is the weight at input node j, x_j is the input node, and b is the bias from the fixed input.
* The Threshold Function also referred to as a Heaviside functions can be expressed as

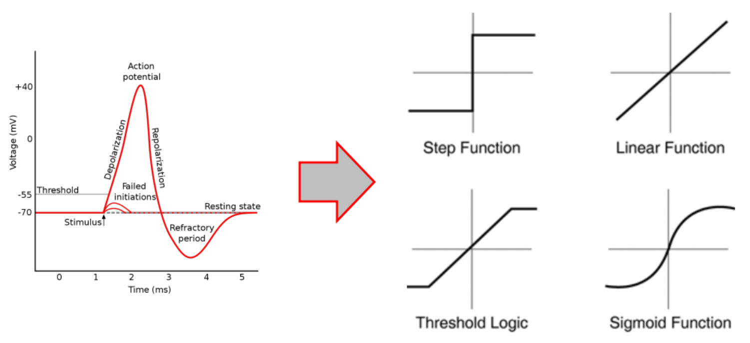
$$\phi(v) = \begin{cases} 1 & \text{if } v \geq 0 \\ 0 & \text{if } v < 0 \end{cases} \tag{2}$$

* The Sigmoid Function is "S" shaped instead of the "Step" in a Threshold Function.

>>> Activation Functions

* What type of activation functions did we use in Homework 1?
* An example of a Sigmoid Activation function is the Logistic Function.

$$\phi(v) = \frac{1}{1 + e^{-av}} \tag{3}$$



>>> Stochastic Model of Neuron

* Whether or not a neuron is activated can be probabilistic, but it has only two states - "on" or "off".
* Let $P(v)$ denote the probability of a neuron firing where v is the induced local field of the neuron.

$$P(v) = \begin{cases} +1 \texttt{ with probability} & P(v) \\ -1 \texttt{ with probability} & 1 - P(v) \end{cases} \tag{4}$$

* The standard choice for $P(v)$ is a sigmoid function

$$P(v) = \frac{1}{1 + e^{-v/T}} \tag{5}$$

where T is a pseudotemperature used to control the noise level and the uncertainty in firing. It controls the fluctuations of the synapses firing.

>>> Association

* A method for finding how items, people, ect. are associated with each other is using Association Rules.
* There are three measures used to identify associations:
  1. Support - This indicates how frequently an itemset appears within the dataset. Itemsets can also contain multiple items that are related and it can be used to calculate the support for all the individual items in the set. A support threshold can be chosen to identify frequent items.
  2. Confidence - This indicates how frequently one item appears when another is present. One drawback of this measure is that it might misrepresent the importance of an association.
  3. Lift - This indicates how frequently two items appear together while accounting for how frequently each appear on their own. A lift value greater than one implies two items are related, and a value less than one means they are not.

>>> Association Example

* Using the three measures to identify associations, we can identify associations as well as infer their significance whether it is coincidence or meaningful.

Transaction 1	🍎 🍺 🍚 🍗
Transaction 2	🍎 🍺 🍚
Transaction 3	🍎 🍺
Transaction 4	🍎 🍐
Transaction 5	🍼 🍺 🍚 🍗
Transaction 6	🍼 🍺 🍚
Transaction 7	🍼 🍺
Transaction 8	🍼 🍐

Table 1. Example transactions.

$$\text{Support}\ \{🍎\} = \frac{4}{8}$$

Figure 1. Support measure.

$$\text{Confidence}\ \{🍎 \rightarrow 🍺\} = \frac{\text{Support}\ \{🍎, 🍺\}}{\text{Support}\ \{🍎\}}$$

Figure 2. Confidence measure.

$$\text{Lift}\ \{🍎 \rightarrow 🍺\} = \frac{\text{Support}\ \{🍎, 🍺\}}{\text{Support}\ \{🍎\} \times \text{Support}\ \{🍺\}}$$

Figure 3. Lift measure.

>>> Apriori Principle

* The possible configurations for association grows exponentially making this an intractable problem for datasets with more than a few items.
* Using the apriori principle, we can get a more efficient solution. This method finds itemsets with high support, confidence and lift.
* The support is calculated and iterated from a single set to larger sets. Then using the support values, we find associations with high confidence and high lift values.
* Even though the apriori principle accelerates the process, it is still computationally expensive.
* Another limitation is that we don't know if the associations are spurious and are happening by change, especially when the dataset is big.
* However, association rules are intuitive and so remain a popular method for identifying patterns.

>>> Social Network Analysis

* Another method for examining relationships and associations is using the Social Network Analysis technique.
* A graph can represent a network. It has nodes and edges, and each edge can have a weight.
* Figure 1 shows a graph of four people in a social network. Each person is a node, and each line is an edge. The thicker lines have higher weights representing relationship strength.

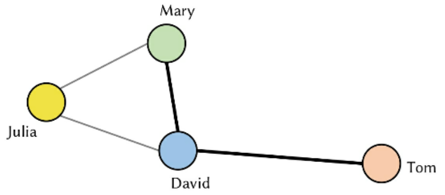


Figure 1. Simple friendship network. Closer friendships are represented by thicker edges.

>>> Network Example

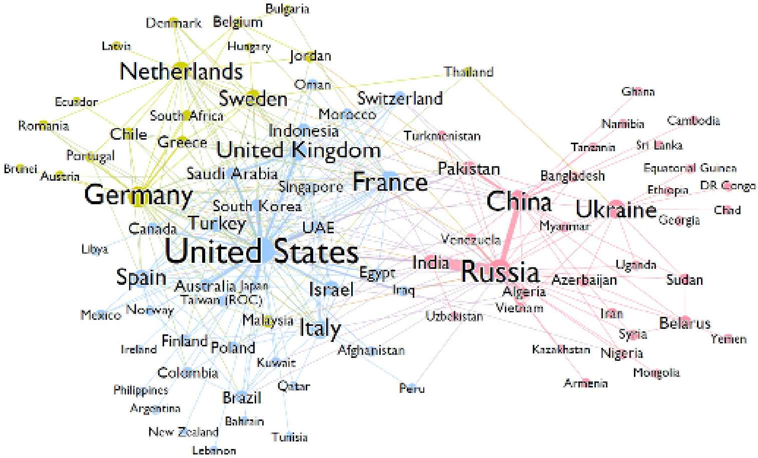


Figure 2. Network of countries based on weapons trade.

>>> Louvain Method and PageRank Algorithm

* The Louvain Method is a way to identify clusters in a network. It experiments with different clustering configurations to maximize the strength of edges and minimize those of nodes in different clusters.
* The edges represent interactions and the Louvain Method identifies cluster regions where they are concentrated. It is simple and efficient and so it is popular for network clustering.
* Another method for identifying clusters is the PageRank Algorithm.
* The number, source, and strength of links are all accounted for in the PageRank Algorithm.
* For an example such as the weapons trade network, we can identify a country's dominance using this method.

>>> Support Vector Machines

* Support Vector Machines (SVM) is a prediction technique that derives an optimal classification boundary between two groups.
* Peripheral data points in one group that are closest to the points from the other group are identified and the boundary is drawn between these points. These points are called support vectors.
* This speeds up computation since only the boundary points are considered for decisions.

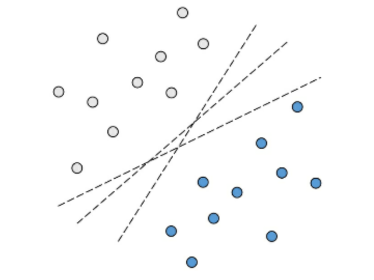


Figure 2. Multiple ways to separate two groups.

>>> SVM Example

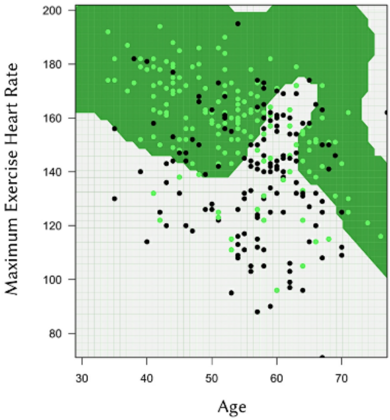


Figure 1. Using SVM to predict the presence of heart disease. The dark green region represents the profile of healthy adults, while the gray region represents the profile of heart disease patients. The light green and black points represent healthy adults and heart disease patients respectively.

>>> SVM Limitations

* The reliance on a small set of points means the decisions means the decision boundary is more sensitive to the positions of these points.
* Most data don't have clean boundaries and data from each group often overlap.
* To overcome this problem, a buffer zone is created by tuning a cost parameter that determines the degree of tolerance for classification errors.

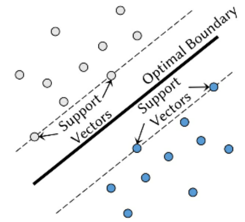


Figure 3. Optimal boundary is located in the middle of peripheral data points from opposing groups.

>>> SVM Advantage

* An advantage of SVM is that it can account for curved patterns in the data using the kernel trick.
* The data is projected onto a higher dimension where it can be separated using a straight line.
* The straight line is easy to compute and easily translated into a curved line when projected back down into a lower dimension.
* SVM's ability to manipulate data in higher dimensions makes it popular.

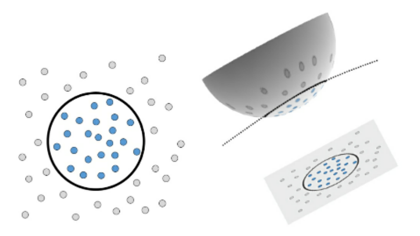


Figure 4. A circle of blue points on a 2-dimensional sheet could be delineated using a straight line when projected onto a 3-dimensional sphere.

>>> Decision Tree

* A decision tree traverses through a set data by recursive partitioning which repeatedly separates the data into two groups.
* The main idea behind this algorithm is that data following the same path are likely to be similar.
* Decision trees are popular because they are easy to interpret and the presence of non-significant variables do not affect results.
* Binary questions tend to divide data points around central values, it is robust against outliers.
* However, since the trees are grown by splitting the data into homogenous groups, a slight change could trigger a change in the split that propagates through. It is also prone to overfitting.

>>> Decision Tree Example

* Sometimes effective splits early on in the process does not lead to the best predictions. Less effective splits initially leads to better predictions subsequently.
* To overcome this limitation, we can diversity the tree as we grow by combining predictions from different trees to get better results with more stability and accuracy.

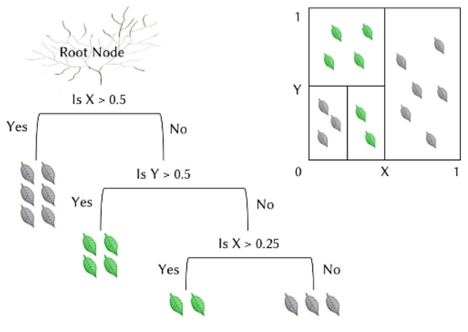


Figure 4. Partitioning of data points by a decision tree visualized in a scatterplot.

>>> Diversifying Trees

* One method for diversifying trees is to choose different combinations of binary questions at random to grow multiple trees.
* The predictions from the different trees are then aggregated. This technique is known as building a random forest.
* Another method is to strategically select binary questions so that the prediction accuracy for each subsequent tree improves incrementally.
* A weighted average of the predictions from all the trees is then taken to get the result. This technique is known as gradient boosting.
* While diversifying trees lead to more accurate predictions, it also makes them more complicated and difficult to visualize.

>>> Random Forests

* Combining models of different strengths and weaknesses to reinforce accurate predictions and cancel out wrong predictions is a methods known as ensembling.
* A random forest is an ensemble of decision trees that can predict complex phenomenon.
* An ensemble combines predictions from many different models by majority voting or averaging.

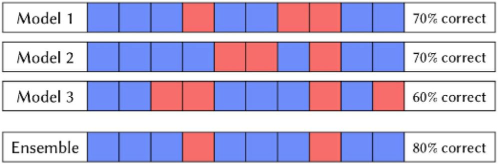


Figure 5. Example of three individual models attempting to predict ten outputs of either blue or red. The correct predictions were blue for all ten outputs. An ensemble formed by majority voting based on the three individual models yielded the highest prediction accuracy of 80%.

>>> Bootstrap Aggregating

* For ensembling to work, the models need to be uncorrelated. A way to generate uncorrelated decision trees is bootstrap aggregating.
* Bootstrap aggregating creates thousands of decision trees that are adequately different from each other.
* The predictor variables allowed for selection at each split are restricted. Only three of the predictor variable in Figure 6 are allowed at each node.

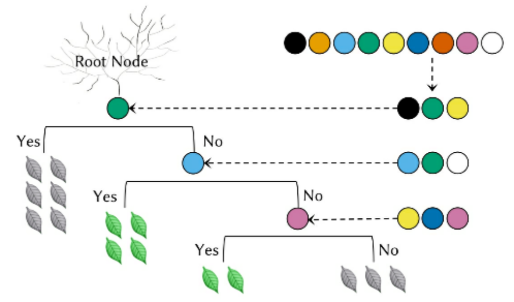


Figure 6. Creating a decision tree via bootstrap aggregating.

>>> Bootstrap Aggregating Continued

* To minimize correlation, a tree is generated from a random subset of the training data, using a random subset of predictor variables.
* This allows us to grow trees that are dissimilar, but retain ceratain predictive powers.
* Restricting the possible predictors for each split generates dissimlar trees to prevent overfitting.
* Another way to reduce overfitting is increasing the number of trees in the random forest which results in a model that is robust yet accurate.
* Since random forests are greatly dependent on the variables at each node, the system is chaotic and difficult to explain or interpret and the process becomes a black box.
* Random forests are widely used because they are easy to implement especially when accuracy is more important than interpretability.

>>> Image Classification

* EfficientNets were released 2 years ago and was considered SOTA because of their scalability which increased training speed. Google released EfficientNetV2 with improvement in training speed and accuracy.
* The main foundation of better performing networks is achieving better performance with a lower number of parameters.
* EfficientNetV2 uses the concept of progressive learning which means that although the image sizes are originally small when the training starts, they increase in size progressively.
* Code is available at: https://github.com/google/automl/efficientnetv2
* Open access paper is referenced: Link to paper

>>> Image Segementation

* U-Nets have been quite powerful in image segmentation.
* A recent trend to optimize U-Nets is to place SOTA CNNs as the U-Net encoder/backbone.
* Since EfficientNets is one of the main SOTA image classification models, U-Net’ s encoder was replaced with an EfficientNet and it showed impressive results.
* Open access paper is referenced: Link to paper

>>> Object Detection

* YOLO an acronym for 'You only look once'.
* It is an object detection algorithm that divides images into a grid system.
* Each cell in the grid is responsible for detecting objects within itself.
* YOLO is one of the most famous object detection algorithms due to its speed and accuracy.
* Paper an code is available at: dhttps://github.com/hyz-xmaster/VarifocalNet