

# Machine Learning: Supervised Classification

Mr. Michaud

Marist School

Computational Perception and Artificial Intelligence

# Types of Machine Learning Problems

- Supervised Learning
  - Training Algorithms on Data sets with Attributes and Given Classifications
    - Decision Trees
    - Support Vector Machines
    - Artificial Neural Networks
    - K-Nearest Neighbor (Like Sum of Squared Distances)
- Unsupervised Learning: Grouping Data without Classifications
  - Clustering Algorithms
  - Dimension Reduction
- Reinforcement Learning

# Focus of this Assignment: Supervised Learning with Artificial Neural Network

- Given a Data Set  $D$ 
  - Attributes:  $X$
  - Classification  $Y$
- In each Instance of data in  $D$ :
  - $X_i$  are the attributes
    - $X_1, X_2, X_3 \dots X_i$
    - $Y$  is the “Classification” or Label

$$D \rightarrow \{X_i, Y\}$$

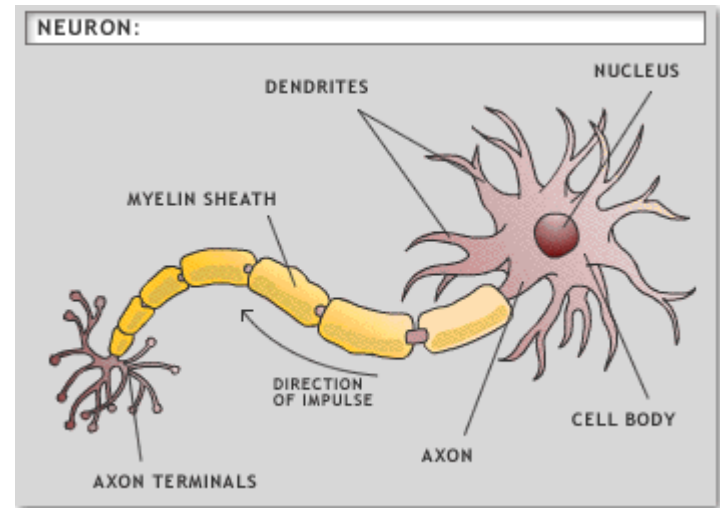
	$X_1$	$X_2$	$X_3$	$X_4$	$Y$
	A	B	C	D	E
1	Sepal Length	Sepal Width	Petal Length	Petal Width	Species
2	5.1	3.5	1.4	0.2	setosa
3	4.9	3	1.4	0.2	setosa
4	4.7	3.2	1.3	0.2	setosa
5	4.6	3.1	1.5	0.2	setosa
6	5	3.6	1.4	0.2	setosa
7	7	3.2	4.7	1.4	versicolor
8	6.4	3.2	4.5	1.5	versicolor
9	6.9	3.1	4.9	1.5	versicolor
10	5.5	2.3	4	1.3	versicolor
11	6.5	2.8	4.6	1.5	versicolor
12	6.9	3.2	5.7	2.3	virginica
13	5.6	2.8	4.9	2	virginica
14	7.7	2.8	6.7	2	virginica
15	6.3	2.7	4.9	1.8	virginica
16	6.7	3.3	5.7	2.1	virginica

# Goal of Supervised Learning

- Given a Data set  $D$  with Attributes  $X_i$  and Classification  $Y$ 
  - Build a Learner that will accurately predict the classification  $Y$  for a given set of attributes  $X_i$
- We build this Learner by training with a data set  $D$  with thousands of samples and creating a Hypothesis ( $H$ ).
- Then with Hypothesis  $H$  we can pass in a set of Attributes  $X_i$  and get a prediction  $Y$

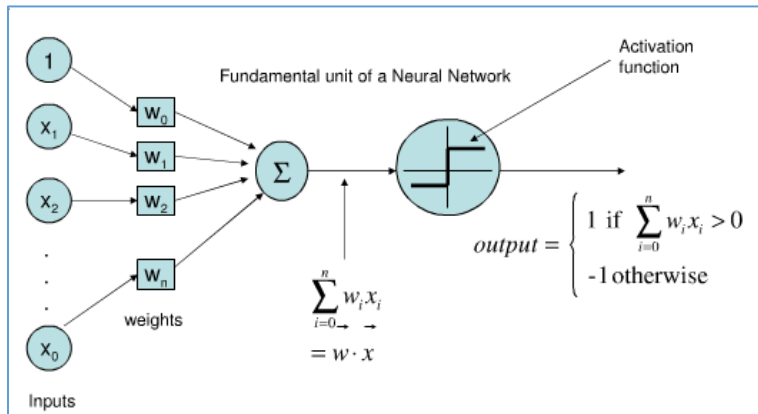
# Learner: Artificial Neural Network (ANN)

- Models the interaction of neurons in a biological system.
- Brain cells:
  - Thousands of connections to other brain cells via dendrites.
  - Thought happens (modelled) through signals passed along the connections between neurons

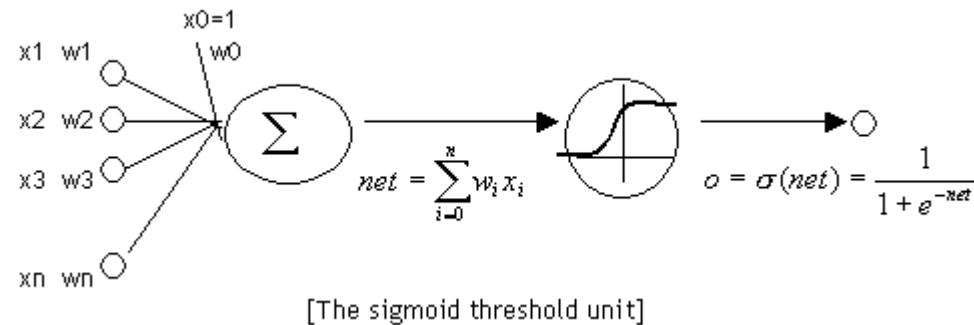


# Perceptron: The “neuron” of an ANN

- Perceptron is the mathematical model of a neuron in an Artificial Neural Network.
- Two types of Perceptrons:
  - Binary
  - Sigmoid

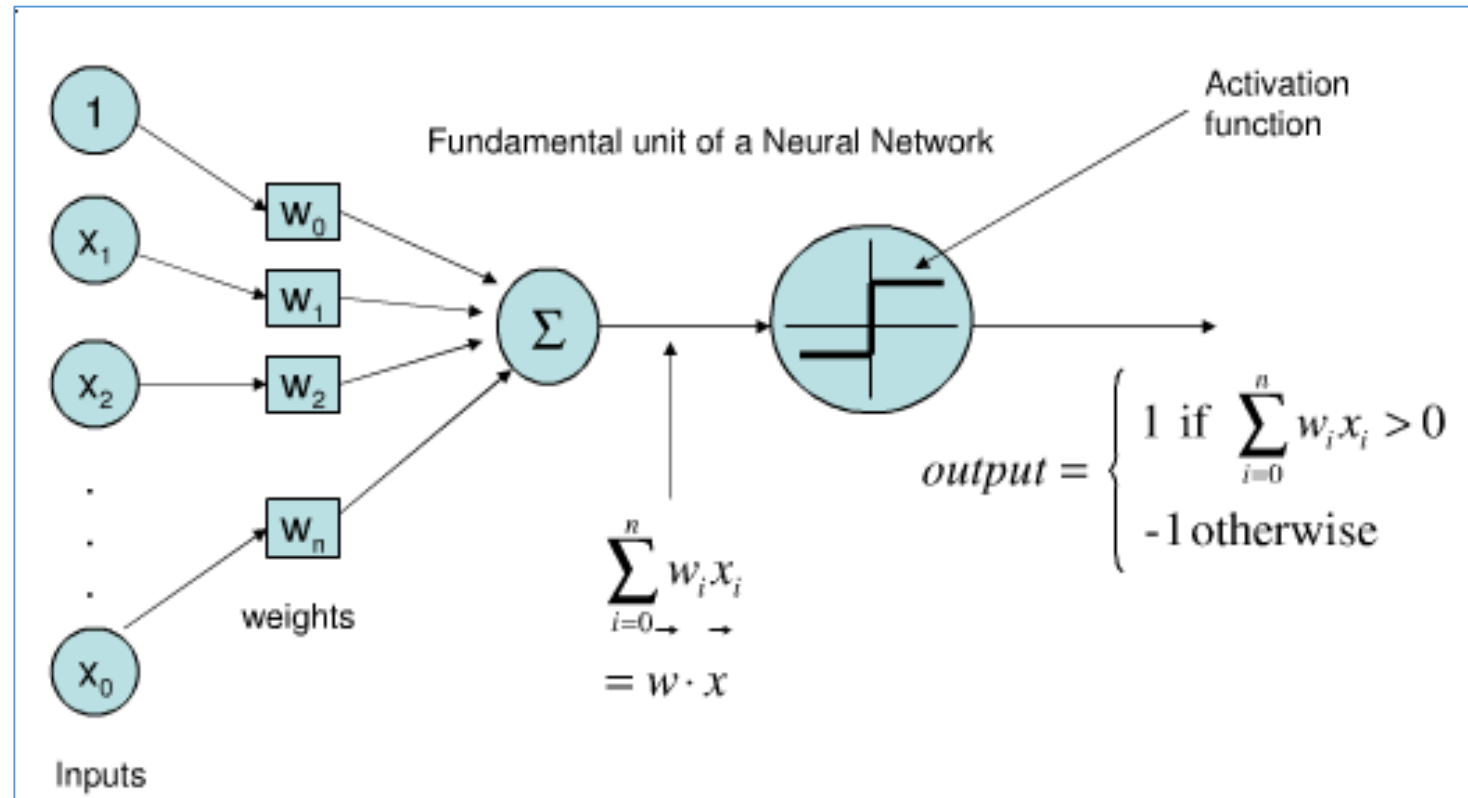


Binary

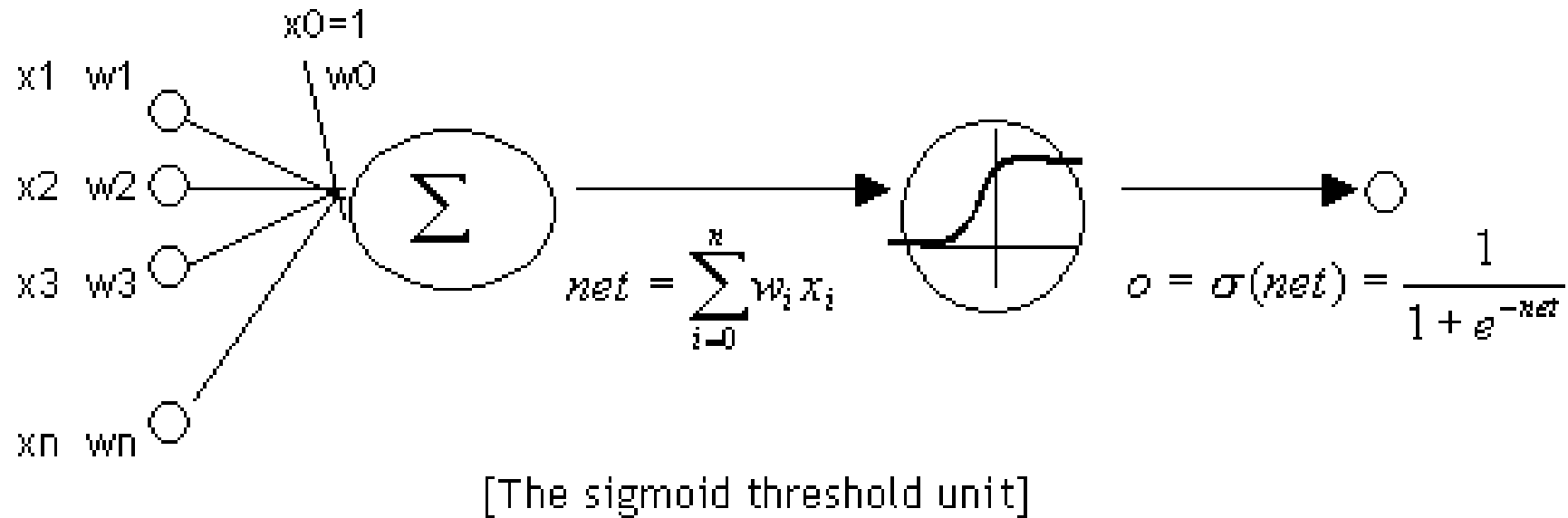


Sigmoid

# Binary Perceptron

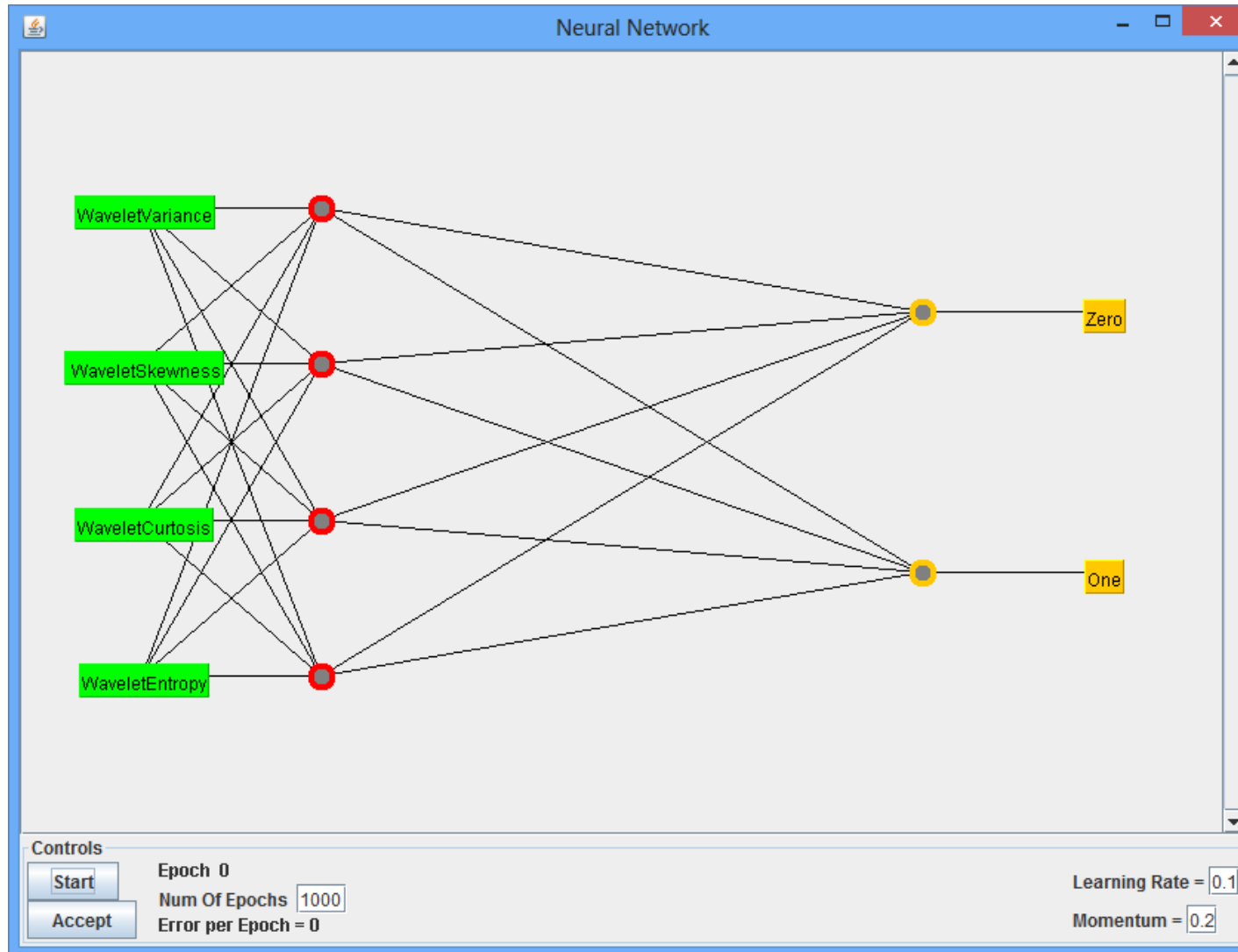


# Sigmoid Perceptron

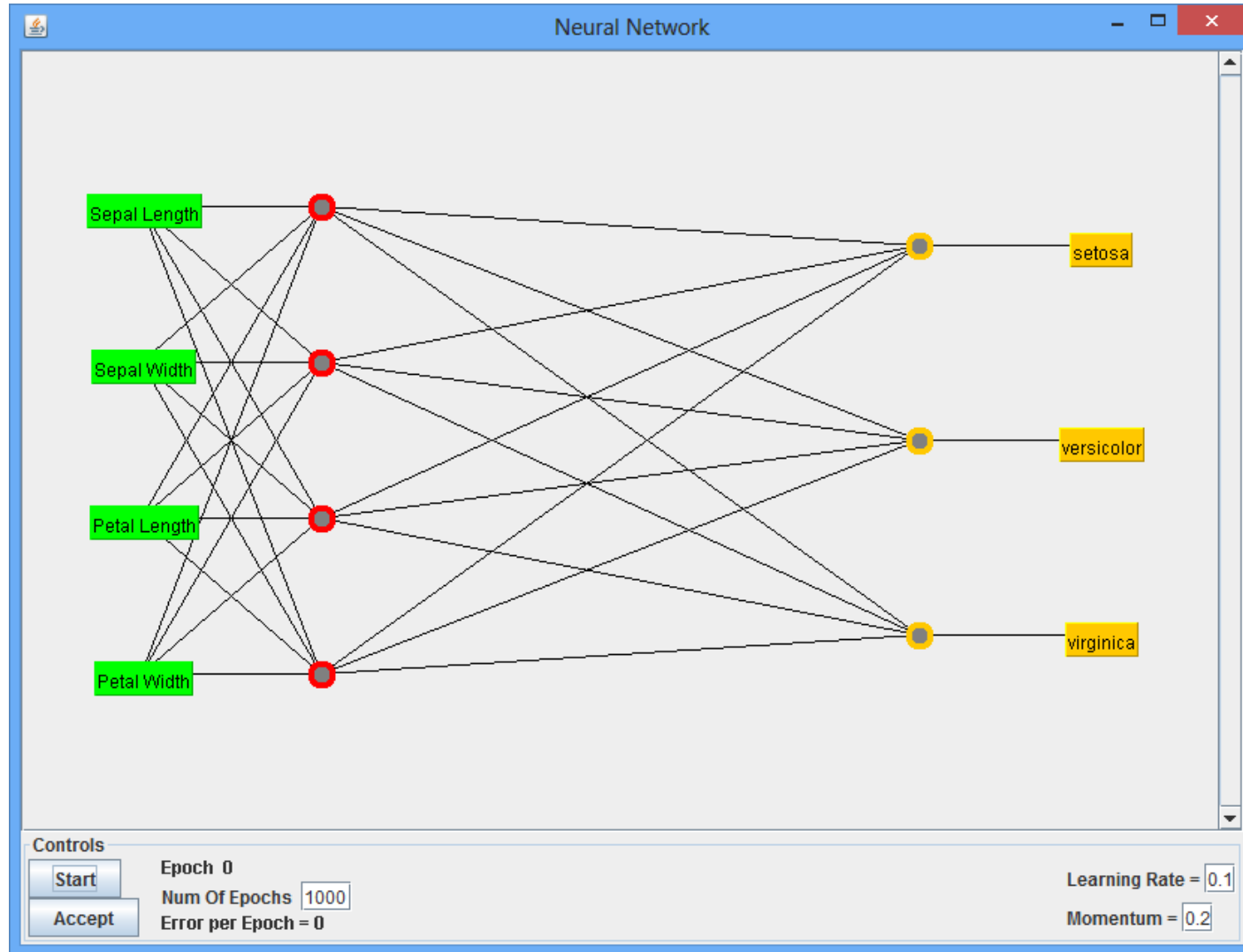




# Diagram of Small Neural Network



# Another Example . . .



# Training equations for Perceptrons

Basic Idea:

$$w_i^m = w_i^{m-1} + \alpha (y_j - f(x_j))$$

**Where:**

w -> weight at connection of perception

$\alpha$  -> Learning Rate – like the slope of the learning

y -> Expected (correct) output given the training example

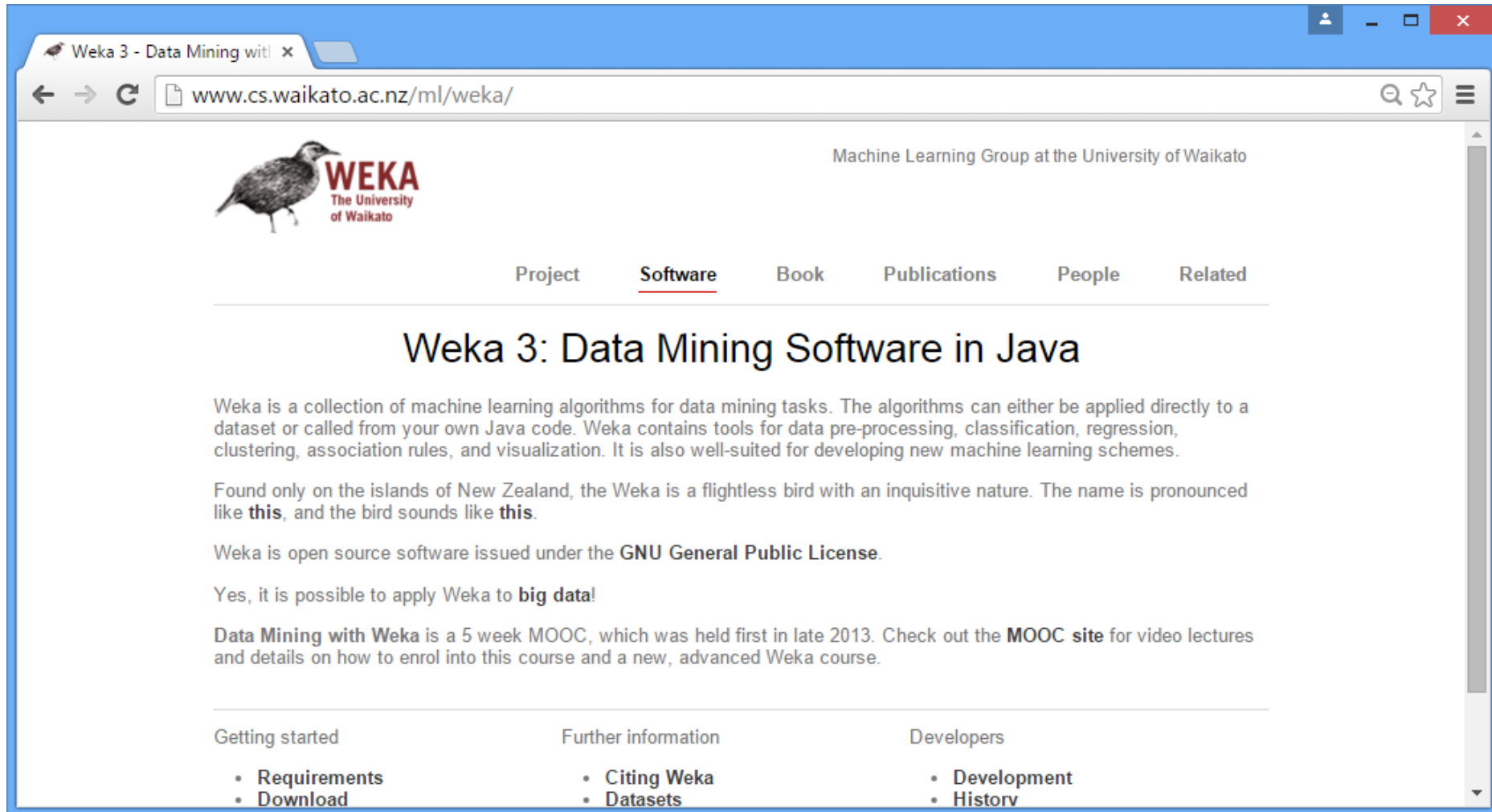
f(x) -> Output actual from the neural network

$$\text{Error} = (y_j - f(x_j))$$

# So – how does this work???

- Algorithm called ‘Back Propagation’ that runs the training samples through the neural network thousands of times.
  - You will see this later . . .
  - It is like the Chain Rule in Calculus
- Each time (epoch) the sample is run, the weights adjust (converge) towards the ‘correct’ Hypotheses.
- The converged Hypotheses then is the model for the Neural Network.

# Tool: WEKA Machine Learning Library



The screenshot shows a web browser window with the URL [www.cs.waikato.ac.nz/ml/weka/](http://www.cs.waikato.ac.nz/ml/weka/). The page features the WEKA logo (a bird) and the text "WEKA The University of Waikato". The navigation menu includes "Project", "Software" (underlined), "Book", "Publications", "People", and "Related". The main heading is "Weka 3: Data Mining Software in Java". The text describes Weka as a collection of machine learning algorithms for data mining tasks, applicable to data pre-processing, classification, regression, clustering, association rules, and visualization. It also mentions that Weka is open source software under the GNU General Public License and is used for big data. A MOOC (Massive Open Online Course) is mentioned, which was held first in late 2013. The footer contains three columns of links: "Getting started" (Requirements, Download), "Further information" (Citing Weka, Datasets), and "Developers" (Development, History).

Weka 3 - Data Mining with Weka

www.cs.waikato.ac.nz/ml/weka/

Machine Learning Group at the University of Waikato

WEKA  
The University of Waikato

Project Software Book Publications People Related

## Weka 3: Data Mining Software in Java

Weka is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes.

Found only on the islands of New Zealand, the Weka is a flightless bird with an inquisitive nature. The name is pronounced like **this**, and the bird sounds like **this**.

Weka is open source software issued under the **GNU General Public License**.

Yes, it is possible to apply Weka to **big data**!

**Data Mining with Weka** is a 5 week MOOC, which was held first in late 2013. Check out the **MOOC site** for video lectures and details on how to enrol into this course and a new, advanced Weka course.

Getting started

- Requirements
- Download

Further information

- Citing Weka
- Datasets

Developers

- Development
- History

# Setup: Machine Learning with Java and WEKA

- Download [PS11.zip](#) and extract in your directory
- Open Eclipse and set workspace to PS11
- Create a new Java Project called PS11
- Import the weka.jar file and place on build path
- Create directory “iris\_data” in your PS11 Java Project
- Copy the iris.csv file into the “iris\_data” directory
- (Detailed Directions at: <http://nebomusic.net/perception/PS11.pdf> )

# Procedure for Establishing ANN in Java / WEKA

- Load Training Data
- Load Testing Data (Optional)
- Set Class Index
- Build Neural Network
  - Set Momentum
  - Set Training Rate
  - Set Hidden Layers
- Train Network
- Test Network / Evaluate Network
- Create System using Learner (after training and testing)