



MASTER OF SCIENCE  
IN ENGINEERING

# Multimodal Processing, Recognition and Interaction

**Unsupervised & Semi-Supervised Learning**

Simon Ruffieux

Elena Mugellini, Stefano Carrino, Omar Abou Khaled

# Summary

- Introduction
- **Trends**
- **Unsupervised Learning**
  - Introduction
  - Clustering
  - K-Means Clustering Algorithm
  - Self Organizing Maps
- **Semi-Supervised Learning**
  - Introduction
  - Self-training
  - Co-training
  - Active Learning
- **Overview**
  - Supervised vs Unsupervised vs Semi-supervised

# Introduction

So far, in the learning techniques considered, a training example consisted of a set of attributes (or features) and a class/label attached to it

- What if we do **not** have the **labels** of the training samples?
- What if we have **too much data** to label ?
- What if we are **too lazy to label** ? (often true)
- What if we **do not know the possible labels** or we only want to **explore** the data space ?



# Trends

## **BIG DATA**

# Trends

Interest over time ⓘ Google Trends for “Big Data” ↗



Big Data refers to a large collection of data (often collected from Internet or customers indirectly)

**Goal:** Big Data aims at getting an **economical advantage** from the **quantitative analysis** of **internal and external data**



Source: [www.octo.com](http://www.octo.com)

Where does the data come from ?  
Large amount of unlabeled data !



# Trends

- Big Data
  - Most of the time not clearly labelled or just impossible to manually label.
- How to find mechanism to classify ?
  - Pay people to label ?
    - See Mechanical Turk systems
    - Crowd sourcing solutions
  - Unsupervised learning
  - Semi-supervised learning

# Trends

## Example

**Idea:** Send coupons to customer according to their needs

**Goal:** Determine if a woman is pregnant based on her shopping data

*“Take a fictional Target shopper named Jenny Ward, who is 23, lives in Atlanta and in March bought cocoa-butter lotion, a purse large enough to double as a diaper bag, zinc and magnesium supplements and scent-free cotton. There’s, say, an 87 percent chance that she’s pregnant and that her delivery date is sometime in late August.” (NYT)*

**Result:** Predictions were too accurate and scared the customers. They had the impression to be spied on.

Source: <http://www.nytimes.com/2012/02/19/magazine/shopping-habits.html>



# Unsupervised learning

# Unsupervised Learning – Introduction

- The data has **no target attribute or label!**
- We want to **explore the data** to find some **intrinsic structure** in them
- Find the most “**natural**” **division between classes** present in the data.

# U.L. – Some Methods

- **Clustering**
  - **K-Means clustering**
- Principal Component Analysis (PCA) (ANN)
- Expectation-Maximization algorithm
- Competitive Learning (ANN)
- Kohonen's Neural Network – **Self Organizing Maps** (ANN)
- Etc.

# U.L. – Clustering

## Reminder - What is Clustering ?

- Input
  - Training samples  $\{x_1, \dots, x_m\} \in \mathbb{R}^n$
  - No labels are given !!

### Goal

**Grouping** input samples into classes of similar objects (“clusters”)

- High intra-class similarity
- Low inter-class similarity

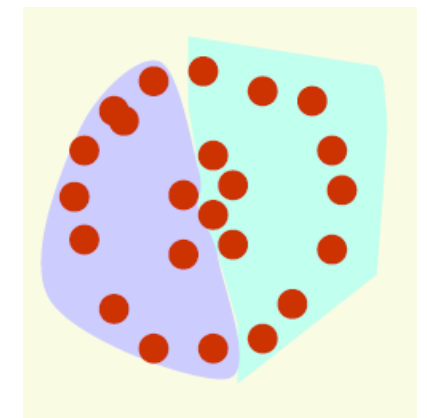
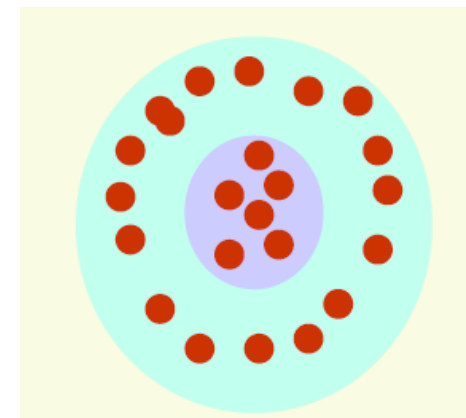
# U.L. – Clustering

## Important parameters?

- Initialization method
- Distance/proximity measure
  - Invariant to scale/rotation?
  - Identify differences
- Objective function
  - May influence clusters shape
- Termination criterion
  - Converged?



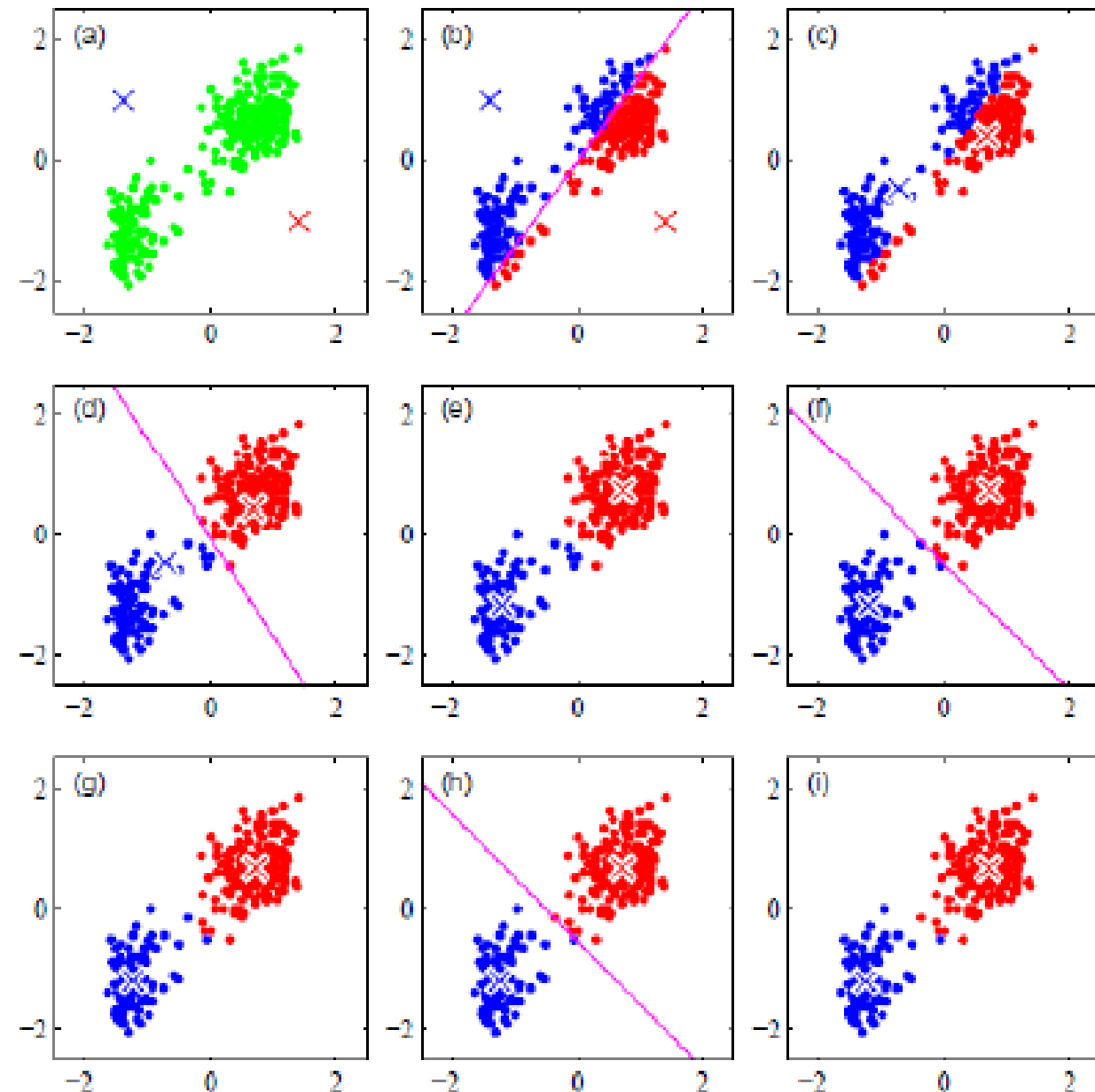
Influence of distance measure



Influence of objective function

# U.L. – K-Means Clustering

```
Initialization();  
While (!satisfied):  
  For (i=1:N)  
    Assign sample  $x_i$  to closest centroid  
  End  
  Move each centroid to average of  
  assigned points  
End
```





# U.L. - Self Organizing Maps (SOM)

## Introduction

- Also known as Kohonen's network or Kohonen's Feature Map
- Unsupervised learning with artificial neural networks
- Based on the brain self-organization
  - Evidence of brain plasticity (adapts to experience)
  - Spatial order and organization in the way the brain behaves
- Proposed by Kohonen in 1982 and has been applied in many domains (text analysis, pattern recognition, image analysis, bioinformatics, cheminformatics)

# U.L. - Self Organizing Maps

## Introduction

- Based on Competitive learning
  - Only activated neurons and their neighbours have their weights updated (local update of the neurons)
- Concept
  - Project high-dimensional data in a lower dimension (typically 2D) while preserving the relationship among the input data.
    - Very good solution to visualize complex data on 2D maps.

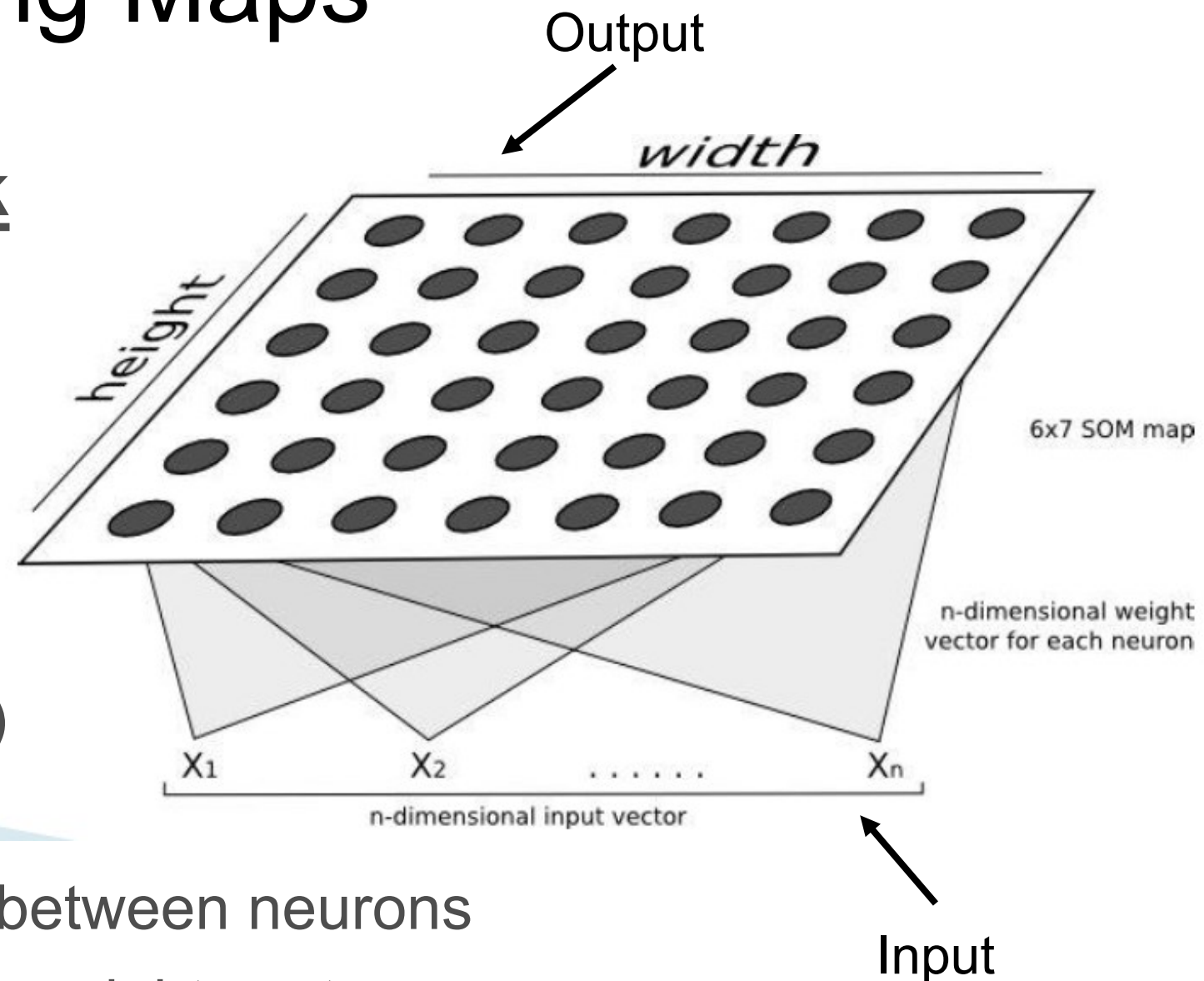
→SOM Objective:

**Neighbour inputs map onto neighbour outputs**

# U.L. - Self Organizing Maps

## Architecture of Network

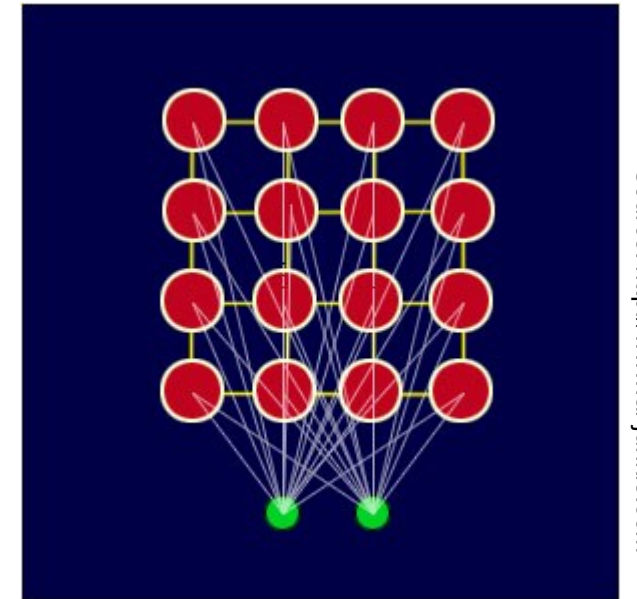
- Consists in two layers
  - Input layer
  - Output layer (competitive)
    - $K$  neurons ( $H \times W$ )
    - Neighbouring relations between neurons
    - Each neuron contains a weight vector
      - Same dimension as input vector!



# U.L. - Self Organizing Maps

## Architecture of Network: Example

- We have an input vector (in green) of 2 dimensions
  - Input Vector:  $V_1, V_2$
- We have an output layer of 16 nodes
  - Each of the 16 nodes (in red) contains a weight vector  $W$  of 2 dimensions
    - Weight vector:  $W_1, W_2$



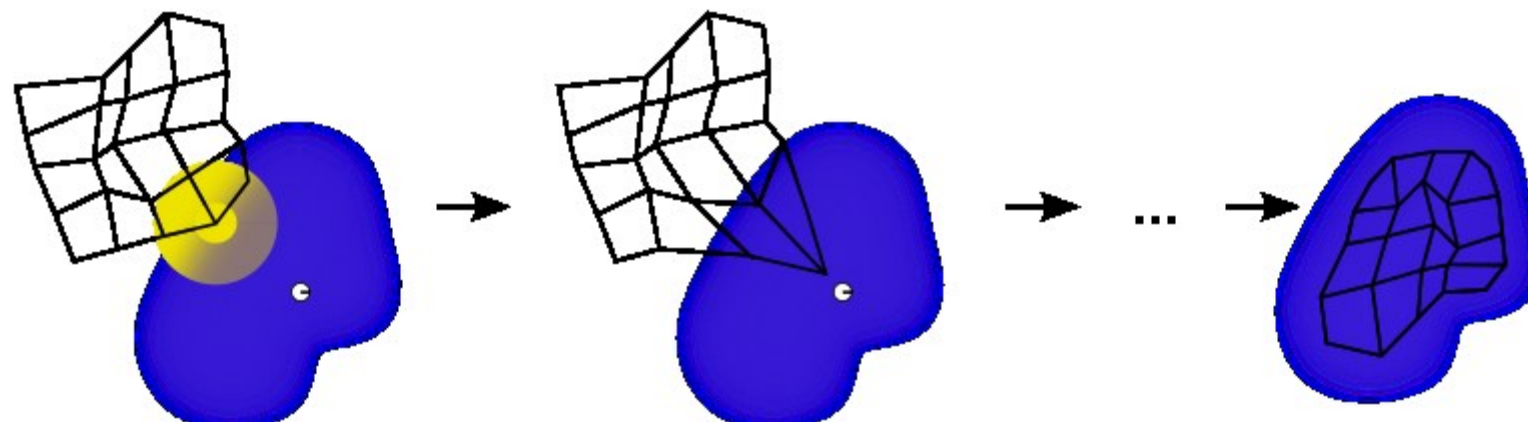
Source: <http://www.ai-junkie.com>



# U.L. - Self Organizing Maps

## Functioning overview

- Goal: Cause different parts of the network to respond similarly to certain input patterns
- Approximate the data distribution with the map



An illustration of the training of a self-organizing map. The blue blob is the distribution of the training data, and the small white disc is the current training datum drawn from that distribution. At first (left) the SOM nodes are arbitrarily positioned in the data space. The node (highlighted in yellow) which is nearest to the training datum is selected. It is moved towards the training datum, as (to a lesser extent) are its neighbors on the grid. After many iterations the grid tends to approximate the data distribution (right). (source: Wikipedia)

# U.L. - Self Organizing Maps

## Concepts overview

- Self-organization emerges due to 3 essential processes:
  1. Competition
  2. Cooperation
  3. Adaptation



# U.L. - Self Organizing Maps

## 1. Competition

When an input pattern is presented to the network, a discriminant function (=distance metric) is computed for all weight vectors (neurons). The neuron with the most similar weight vector to the input pattern is called **Best Matching Unit** (BMU) and is the winner of the competition.

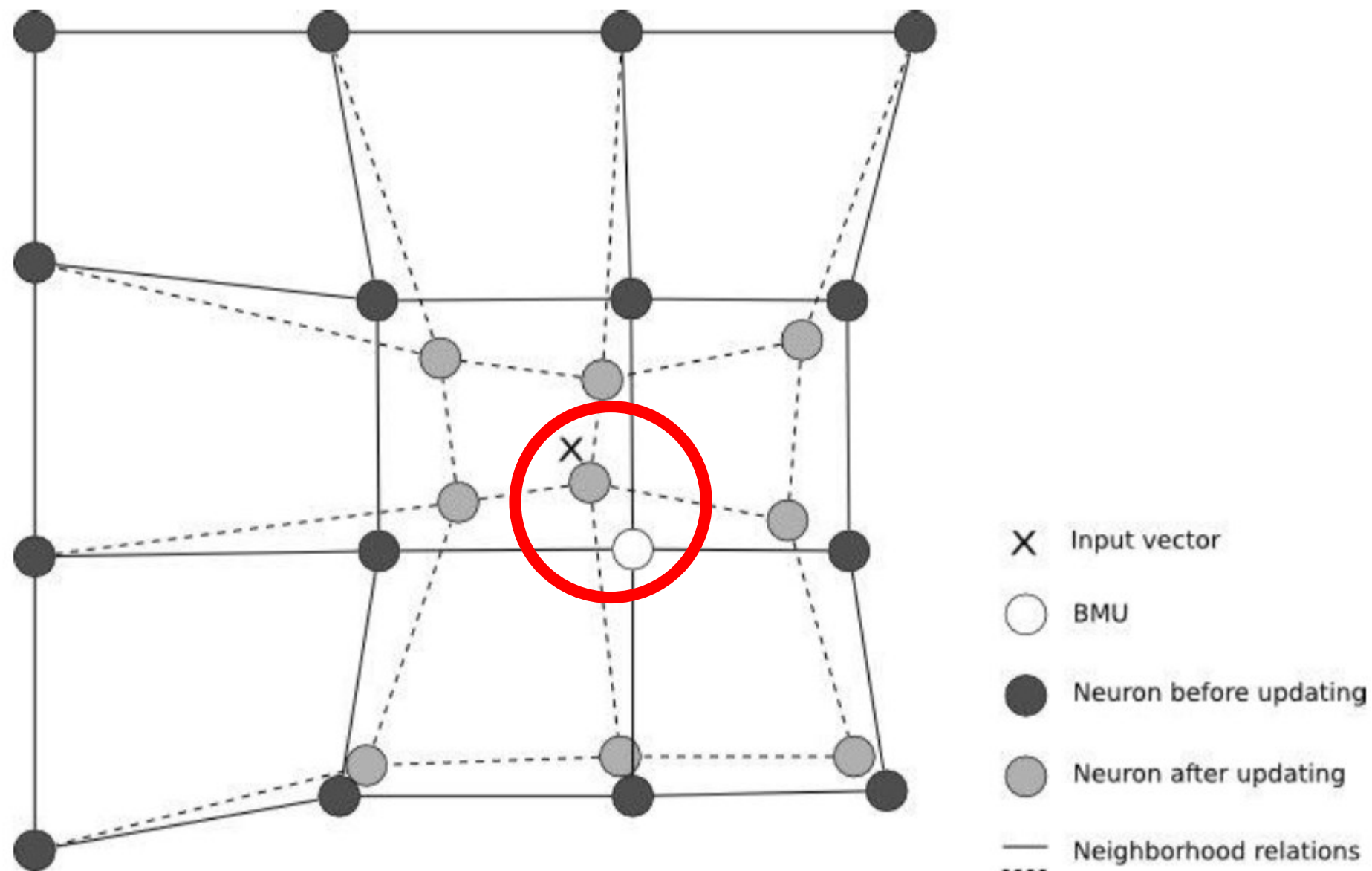
## 2. Cooperation

The BMU determines the **spatial location of a topological neighbourhood of excited neurons**. The neighbouring neurons of the BMU will also be updated.

## 3. Adaptation

This last process enables **the excited neurons to increase their resemblance to the input pattern through suitable adjustments to their weight vectors**. Consequently, the response of the winning neuron (and its neighbours) to a subsequent similar input is enhanced

# U.L. - Self Organizing Maps



# U.L. - Self Organizing Maps

## Algorithm

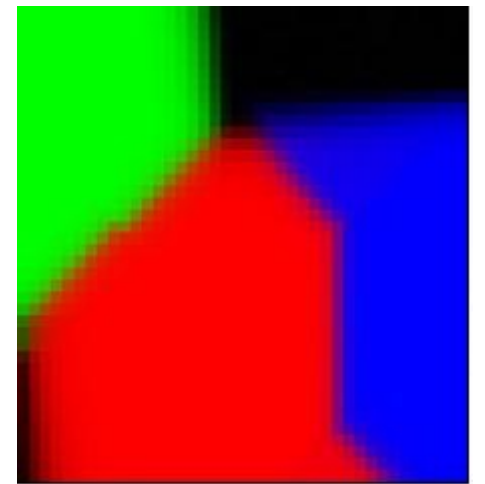
1. Each node weights is initialized
2. A random vector  $V$  is chosen from the training set
3. Every node is examined to determine the BMU.
  - The one with weights being most similar to  $V$  is *the BMU*
4. The neighbours of the BMU are determined.
  - A special function is used for that. (Gaussian, ...)
5. The BMU has its weights updated to resemble the input vector. The neighbours are also updated (less).
6. Go to step 2.

# U.L. - Self Organizing Maps

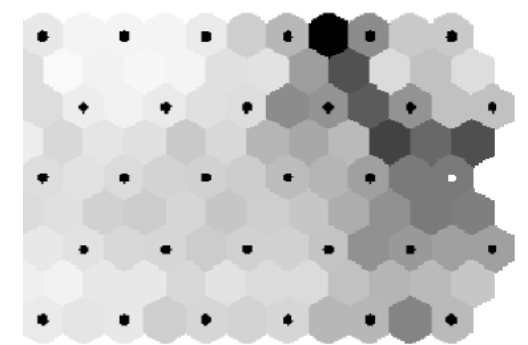
## Visualization

- 2 main visualization techniques
  - **SOM colour map**
    - Visualize the results of the classification on 2D using colour schemes to represent the clusters
  - **SOM similarity map**
    - Also called Unified distance matrix (U-matrix)
    - Represents the distance between adjacent neurons. A dark colouring between neurons represents a large distance.
    - Easy to identify clusters using simple image processing techniques.

SOM colour map



SOM Similarity map

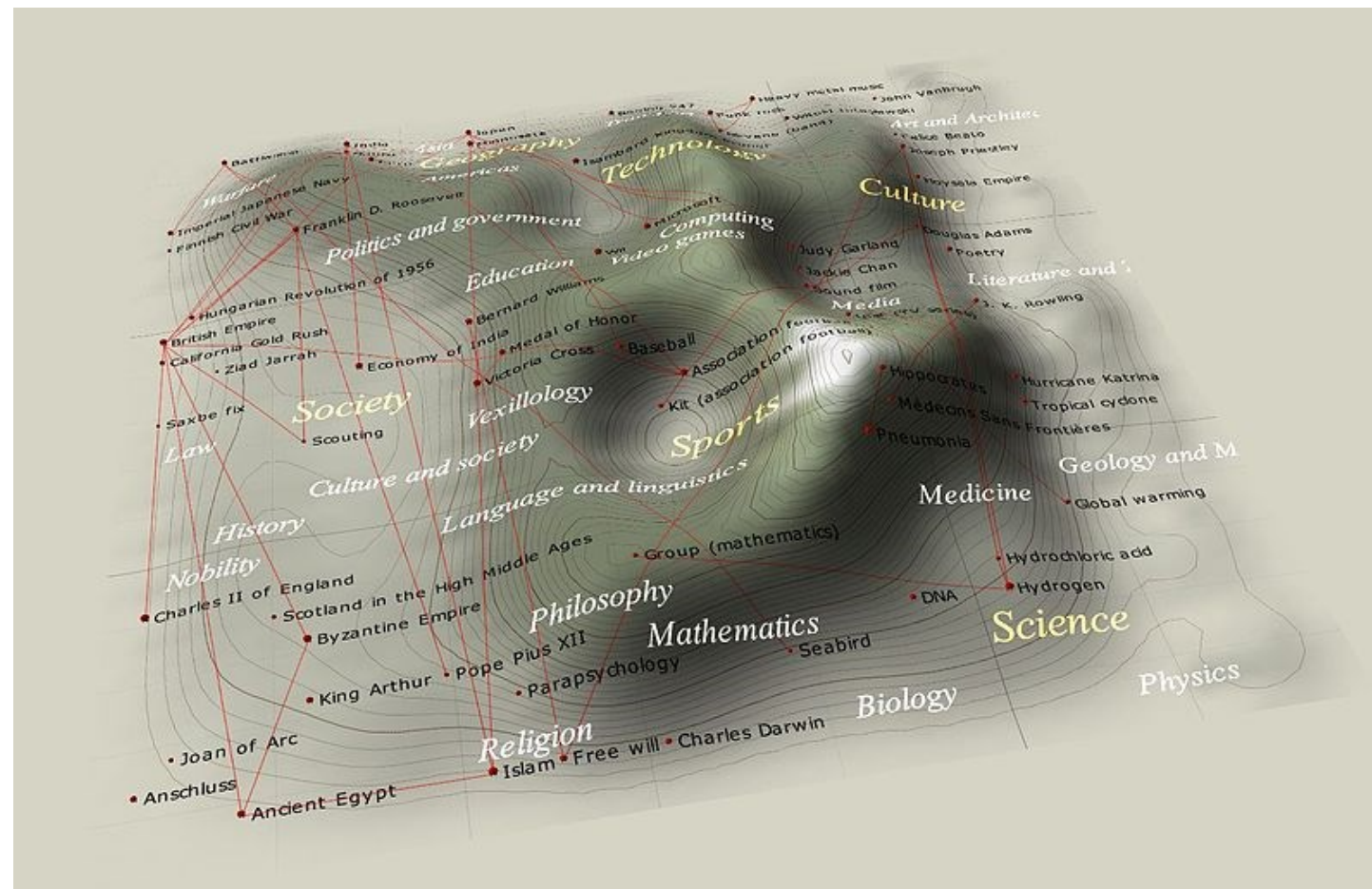




# U.L. - Self Organizing Maps

## Visualization

- Advanced technique



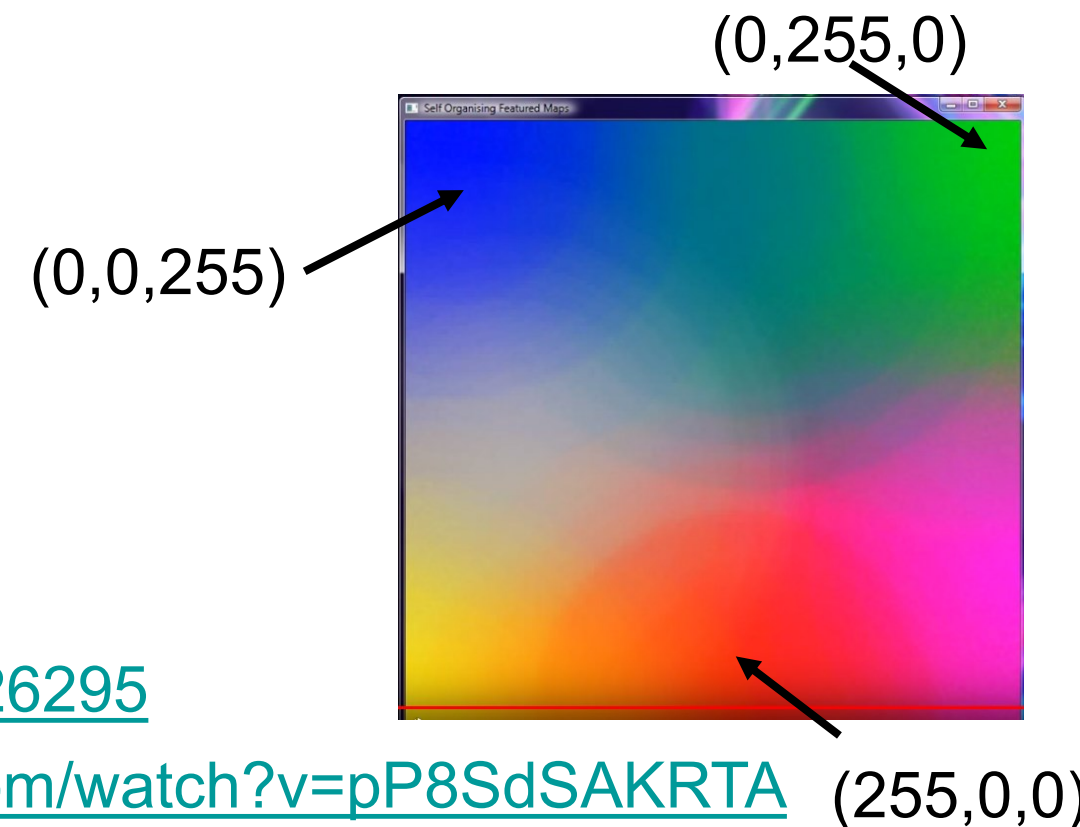
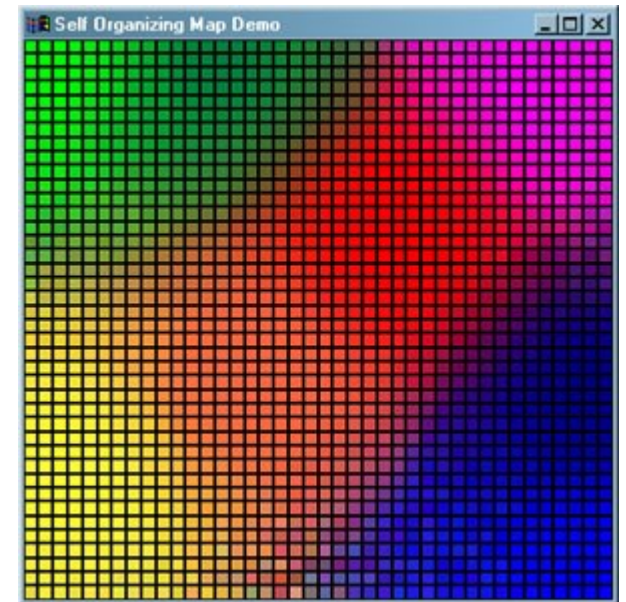
Source: <http://en.wikipedia.org>

Cartographical representation of a self-organizing map (u-matrix) based on Wikipedia featured article data (word frequency). Distance is inversely proportional to similarity. The "mountains" are edges between clusters.

# U.L. - Self Organizing Maps

## Example: Mapping RGB colors

- *Inputs*
  - 3 dimensional:  $R, G, B$
  - Training set: randomly selected RGB colours
- *Outputs*
  - 2500 units (50x50 grid)
  - Each unit has a 3 weights vector
  - Connected in a 2D matrix



Video (3d): <http://vimeo.com/14026295>

Video (2d): <http://www.youtube.com/watch?v=pP8SdSAKRTA>

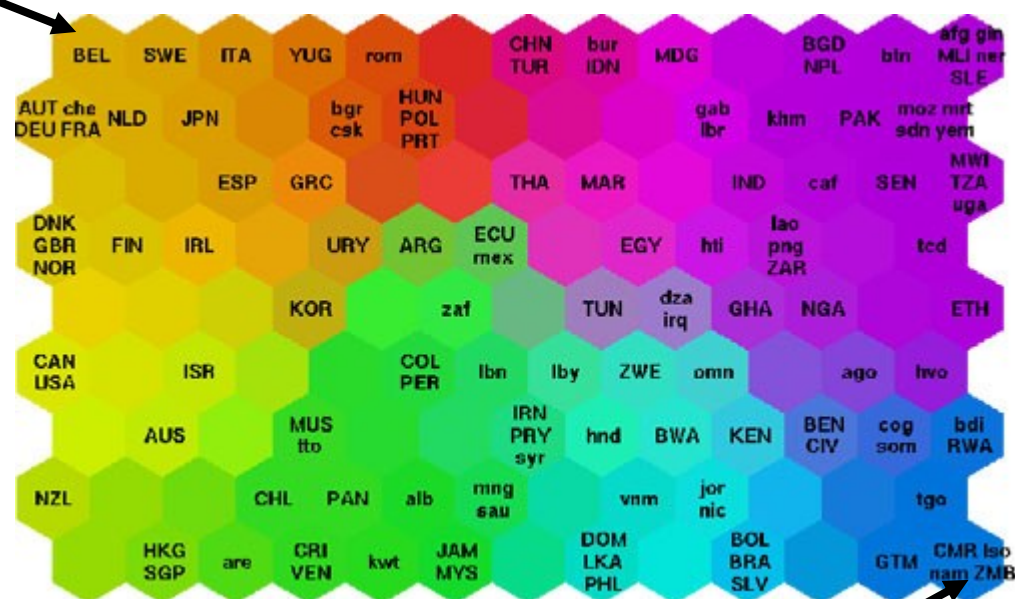
Applet: <http://davis.wpi.edu/~matt/courses/soms/applet.html>



# U.L. - Self Organizing Maps

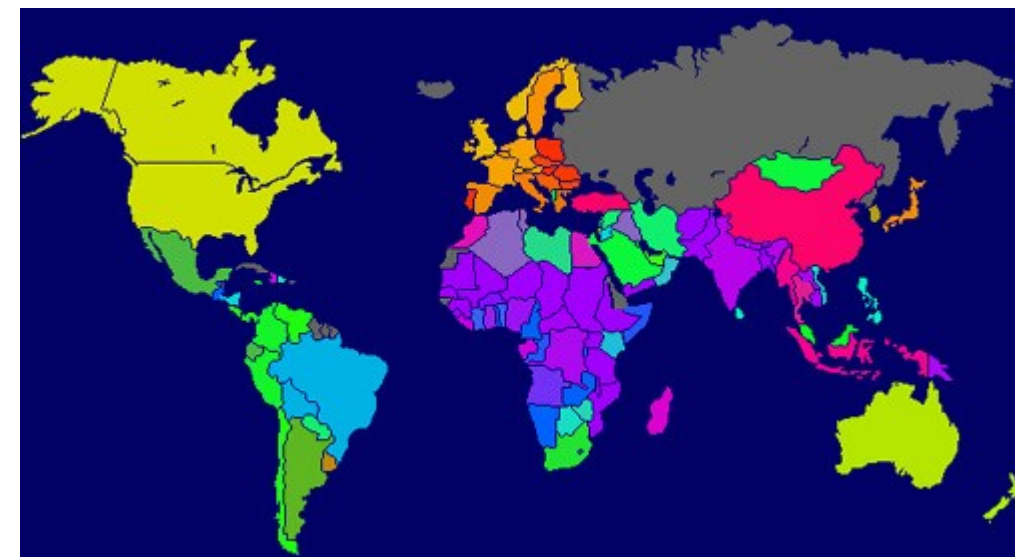
## Example: Poverty Map

“best”



Colour map

“worst”



Colour information plotted on the world map

Source: <http://www.ai-junkie.com>

A SOM has been used as visualization aids for the quality-of-life in the world. The SOM provides a clear visualization of the <mysterious> statistics. 39 indicators (state of health, nutrition, educational services, etc.) have been used to cluster countries with similar quality-of-life factors.

# U.L. - Self Organizing Maps

- Advantages
  - Projects **high-dimensional data onto a 2-dimensional map**
  - The projection preserves the topology of the data so that **similar data items will be mapped to nearby locations** on the map.
- Disadvantages
  - Large quantity of <good quality> representative training data is required
  - No generally accepted measure of ‘quality’ of a SOM
    - How good is the classification/representation is left to user appreciation?
  - Requires human interpretation of the results

# U.L. – Conclusion

- Unsupervised learning is good for finding inherent structure in data
  - Clustering
  - Quantization (=compression of data)
  - Similarities
- Very easy to design algorithms but very hard to evaluate the quality of the result
  - May not make sense for human
- General purpose unsupervised learning does not exist, for best results, it should be tuned to each application at hand



# Semi-Supervised learning

# Semi-supervised Learning - Introduction

- What is semi-supervised learning ?
  - A mix between supervised and unsupervised
  - Methods to **learn from both labeled and unlabeled data**

## Concept:

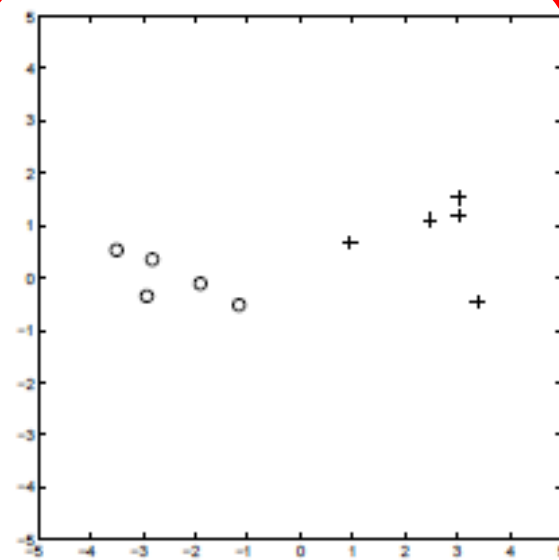
Use the labeled data to infer labels for the unlabeled data to improve the accuracy of the classifier

- When should we use it?
  - Small quantity of labeled data & large quantity of unlabeled data

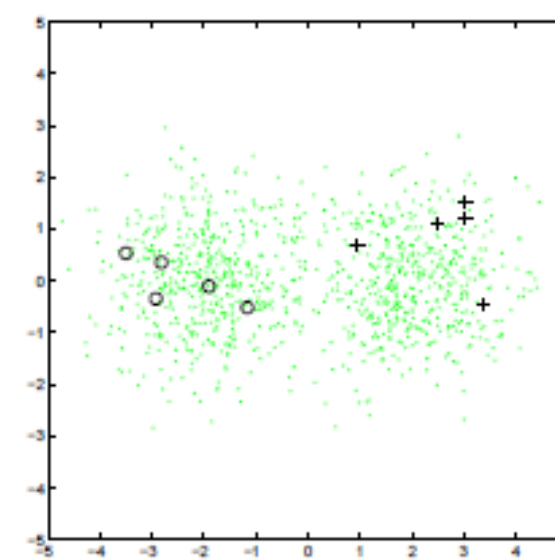


# SSL - Introduction

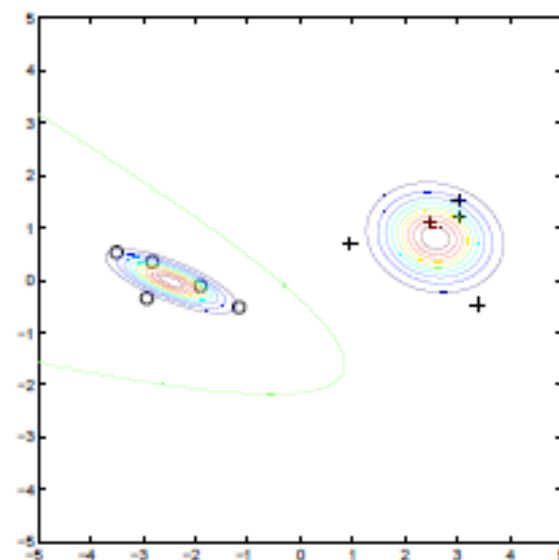
Generate a model for the data!



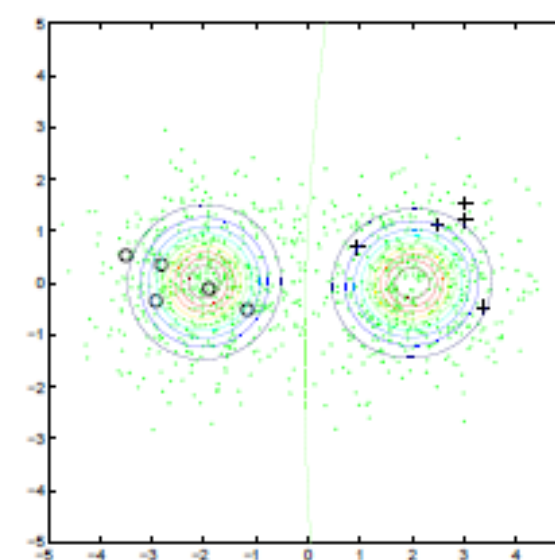
(a) labeled data



(b) labeled and unlabeled data (small dots)



(c) model learned from labeled data



(d) model learned from labeled and unlabeled data

Resulting  
model does  
**not** correctly  
model the data

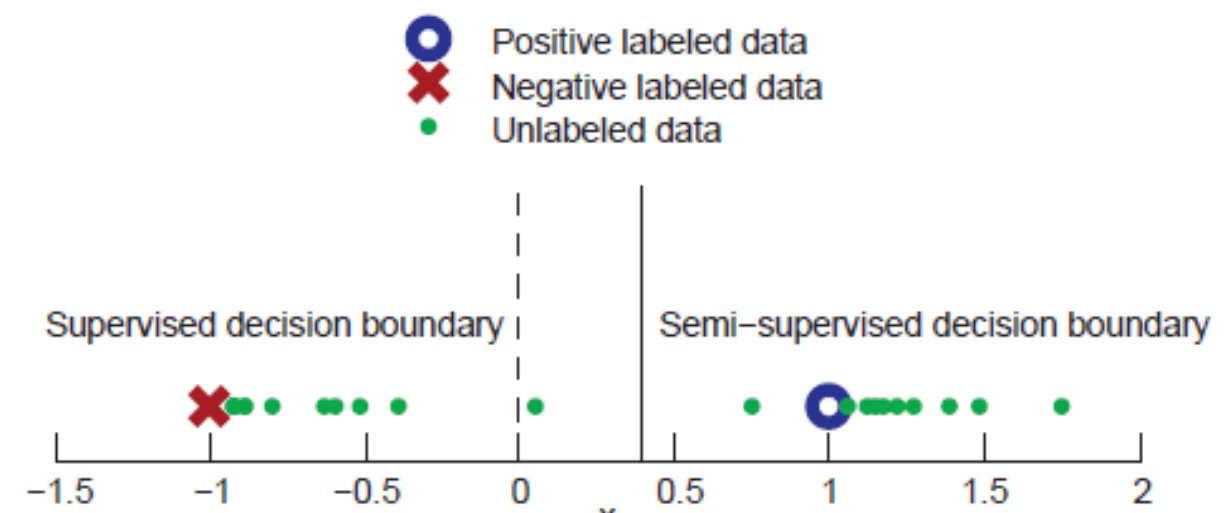
Resulting  
model does  
correctly  
model the data



# SSL - Introduction

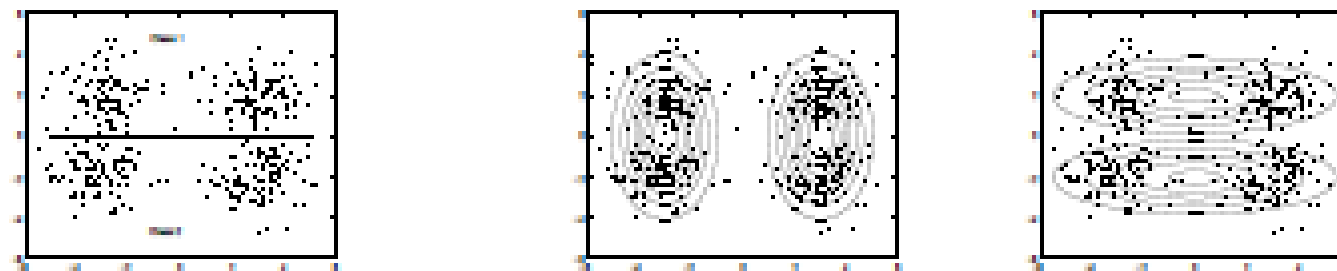
Example: binary classification task

- Supervised learning:
  - Input: 2 labeled samples
  - Output: decision boundary = 0
- Semi supervised learning :
  - Input: 2 labeled samples and many unlabeled samples
  - Assumption: classes form coherent clusters (Gaussian distribution around a central mean)
  - Result: Decision boundary = 0.4



# SSL - Introduction

## Importance of the model!



(a) Horizontal class separation (b) High probability (c) Low probability

Figure 3: If the model is wrong, higher likelihood may lead to lower classification accuracy. For example, (a) is clearly not generated from two Gaussians. If we insist that each class is a single Gaussian, (b) will have higher probability than (c). But (b) has around 50% accuracy, while (c)'s is much better.

## Assumption

Clusters are well separated and can be modelled



# SSL – Some Methods

- **Self Training**
- **Co-Training**
- **Active Learning**
- Graph Based Methods
- Low Density Separation
- Etc.

# SSL – Self Training

## Concept

Also known as self-learning or bootstrapping

L = labeled data and U = unlabeled data

- Train a supervised classifier  $\langle C \rangle$  with the training set L
- Use  $\langle C \rangle$  to classify all samples of U
- Remove the most confident classified samples of U and add them to the training set.
- Retrain the classifier and repeat

# SSL – Self Training

## 1-Nearest-Neighbor: Algorithm

**Algorithm 2.7. Propagating 1-Nearest-Neighbor.**

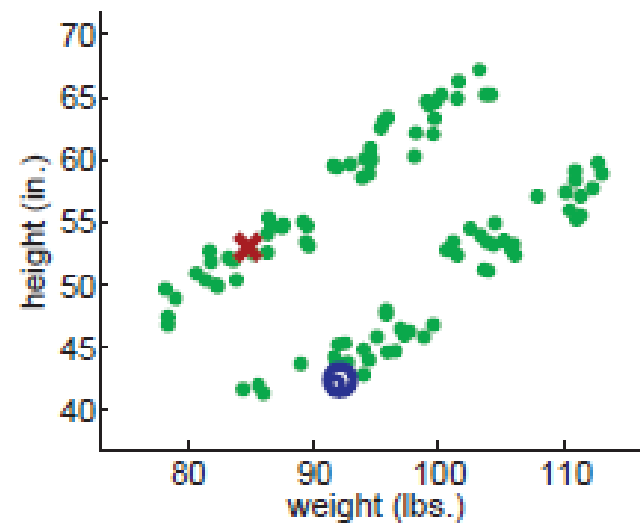
*Input: labeled data  $\{(\mathbf{x}_i, y_i)\}_{i=1}^l$ , unlabeled data  $\{\mathbf{x}_j\}_{j=l+1}^{l+u}$ , distance function  $d()$ .*

- 1. Initially, let  $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^l$  and  $U = \{\mathbf{x}_j\}_{j=l+1}^{l+u}$ .*
- 2. Repeat until  $U$  is empty:*
- 3.   Select  $\mathbf{x} = \operatorname{argmin}_{\mathbf{x} \in U} \min_{\mathbf{x}' \in L} d(\mathbf{x}, \mathbf{x}')$ .*
- 4.   Set  $f(\mathbf{x})$  to the label of  $\mathbf{x}$ 's nearest instance in  $L$ . Break ties randomly.*
- 5.   Remove  $\mathbf{x}$  from  $U$ ; add  $(\mathbf{x}, f(\mathbf{x}))$  to  $L$ .*

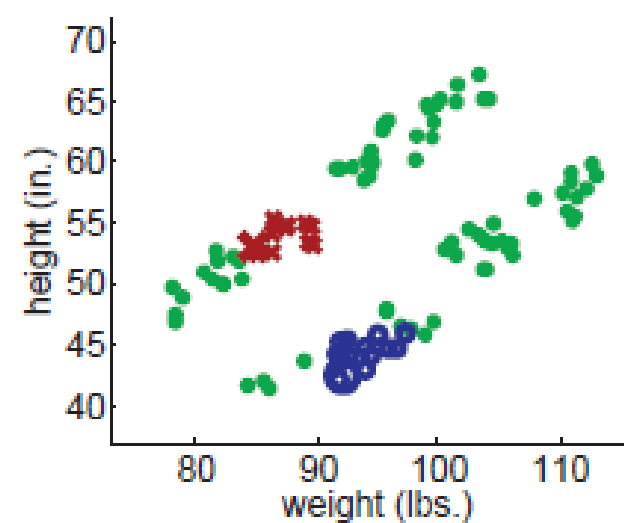


# SSL – Self Training

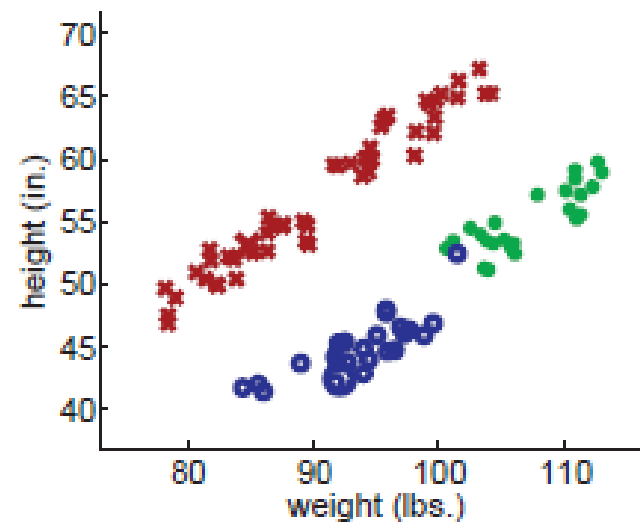
## 1-Nearest-Neighbor: Example



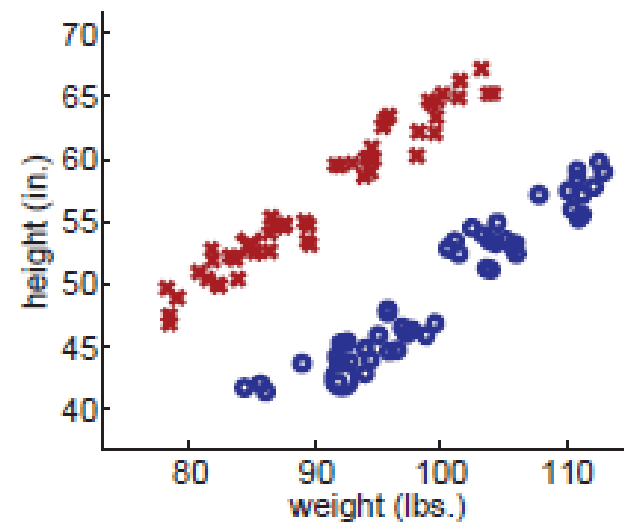
(a) Iteration 1



(b) Iteration 25



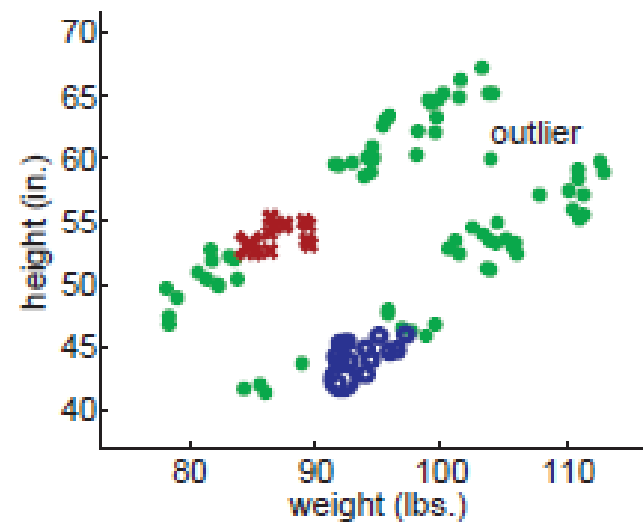
(c) Iteration 74



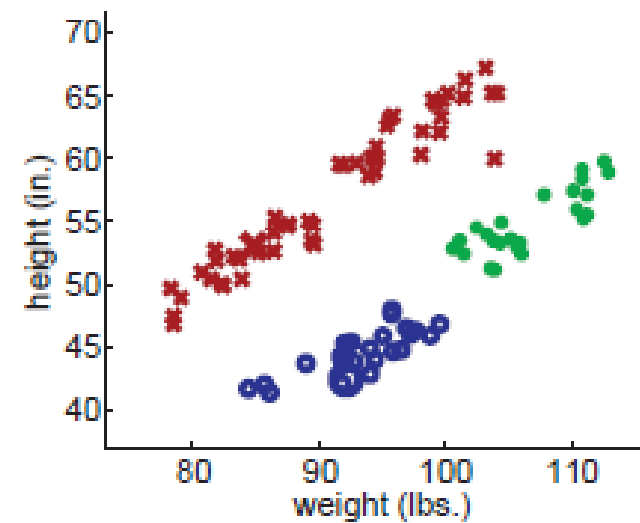
(d) Final labeling of all instances

# SSL – Self Training

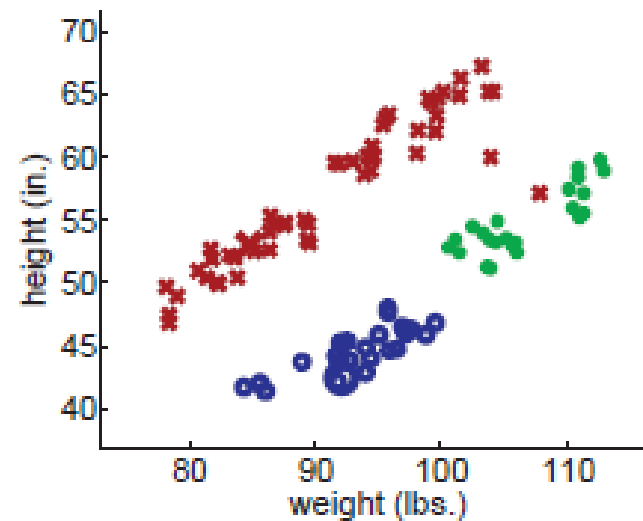
## 1-Nearest-Neighbor: Outlier potential problem



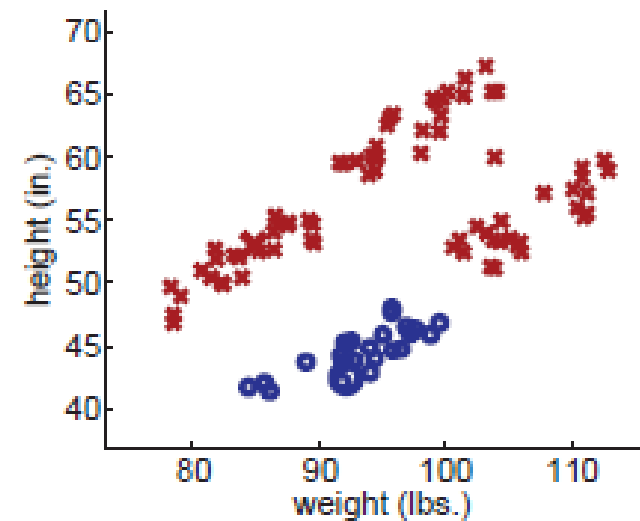
(a)



(b)



(c)



(d)

# SSL – Co-Training

## Introduction

- Concept:
  - Self training : one classifier to label unlabeled examples
  - Co-training: use two distinct classifiers based on distinct features.
- Same as asking the advice of two independent persons instead of only one....
- Advantages
  - Less chance to make mistakes during classifications
  - Features may complement each others

# SSL – Co-Training

## Theory

The data can be described by two different sets of disjoint features (called views)

## Assumptions

- *Sufficiency assumption*: each view should be sufficient to predict the class
- *Independence assumption*: the two views are conditionally independent (the two views are independent given the class)

# SSL – Co-Training

## Concept

### Algorithm 4.1. Co-Training.

*Input: labeled data  $\{(\mathbf{x}_i, y_i)\}_{i=1}^l$ , unlabeled data  $\{\mathbf{x}_j\}_{j=l+1}^{l+u}$ , a learning speed  $k$ .*

*Each instance has two views  $\mathbf{x}_i = [\mathbf{x}_i^{(1)}, \mathbf{x}_i^{(2)}]$ .*

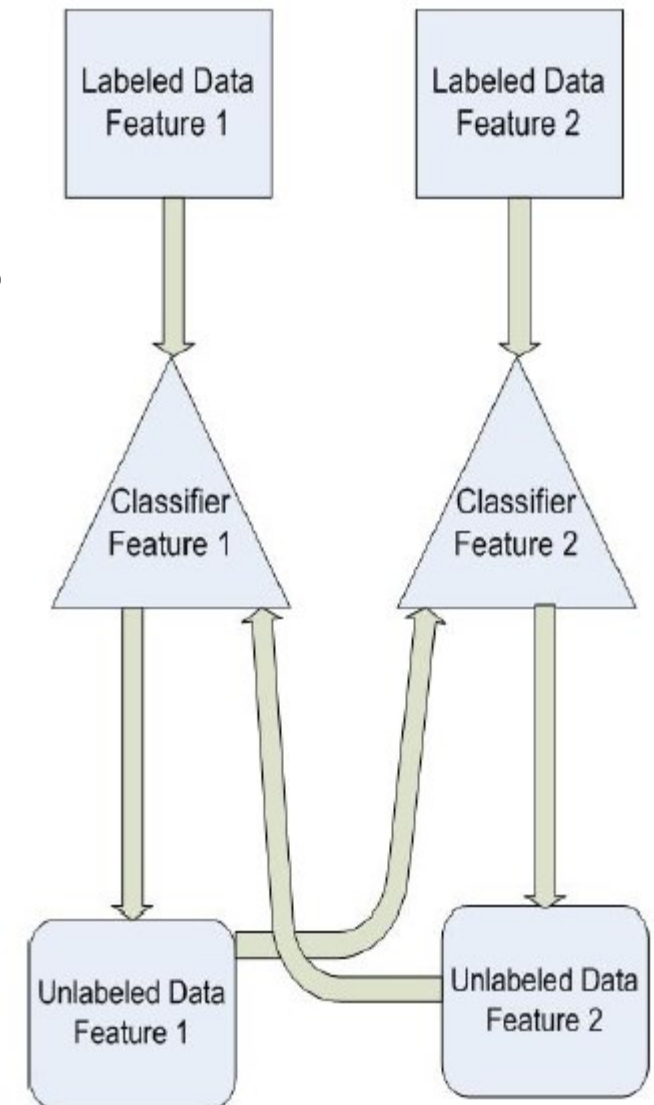
- 1. Initially let the training sample be  $L_1 = L_2 = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_l, y_l)\}$ .*
- 2. Repeat until unlabeled data is used up:*
- 3. Train a view-1 classifier  $f^{(1)}$  from  $L_1$ , and a view-2 classifier  $f^{(2)}$  from  $L_2$ .*
- 4. Classify the remaining unlabeled data with  $f^{(1)}$  and  $f^{(2)}$  separately.*
- 5. Add  $f^{(1)}$ 's top  $k$  most-confident predictions  $(\mathbf{x}, f^{(1)}(\mathbf{x}))$  to  $L_2$ .  
Add  $f^{(2)}$ 's top  $k$  most-confident predictions  $(\mathbf{x}, f^{(2)}(\mathbf{x}))$  to  $L_1$ .  
Remove these from the unlabeled data.*



# SSL – Co-Training

## Practice

- Train a pair of classifiers using small sets of labeled examples
- Unlabelled examples which are confidently labeled by one of the classifiers are added, with labels, to the training set of the other classifier.
- Retrain classifiers



# SSL – Co-Training

## Example: Identify Cars on a highway

- Two-views:
  - Grey level of image
  - Background-subtracted image
- Data:
  - 50 labeled examples
  - 22000 unlabeled images
- Classifier
  - Adaboost (=SVM)



Input Image



Detection result

“Unsupervised Improvement of Visual Detectors using Co-Training” - ICCV '03

# SSL – Co-Training

## Example: Identify Cars on a highway

### Results

- Tested on 90 labeled images

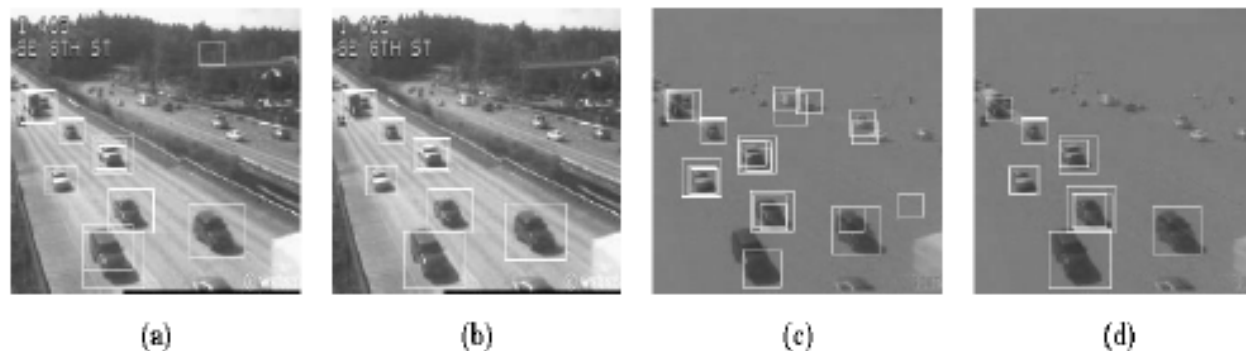


Figure 5: Detection results. (a)- The gray level classifier before co-training. (b)- The gray level classifier after co-training. (c)- The background subtracted classifier before co-training. (d)- The background subtracted classifier after co-training.

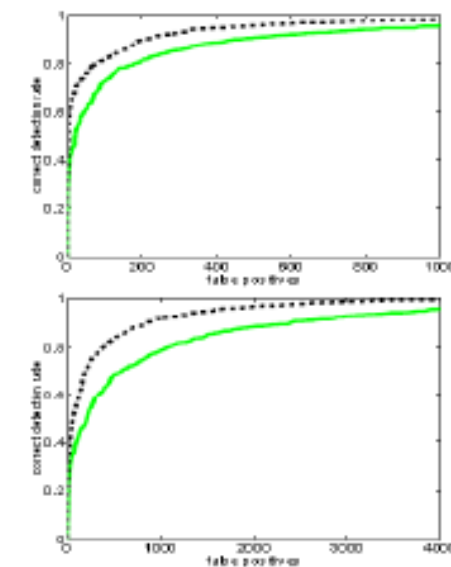


Figure 4: ROC curves. Green/Grey line: the original classifier. Black dashed: the co-trained classifier. TOP the GREY classifier. BOTTOM: the BackSub classifier.

“Unsupervised Improvement of Visual Detectors using Co-Training” - ICCV '03

# SSL – Active Learning

## Concept

Semi-supervised learning where the user or an external trustworthy source of information can be queried to obtain a label for the current sample when there is too much uncertainty.

Very similar to human behaviour when classifying object. If the person classifying an object is too unsure, he then asks another person (an expert if possible)

# SSL – Active Learning

## Application: Spam filter

- When you receive a new mail,
  - sometimes the interface prompts you for information:

Is this mail a spam ?

- You classify it as spam/not spam

→ You refined the algorithm classification and its next results should be better as you have manually classified an uncertain sample.



# SSL – Conclusion

## Applications

- Applicable to any problem where Supervised Learning can be applied.

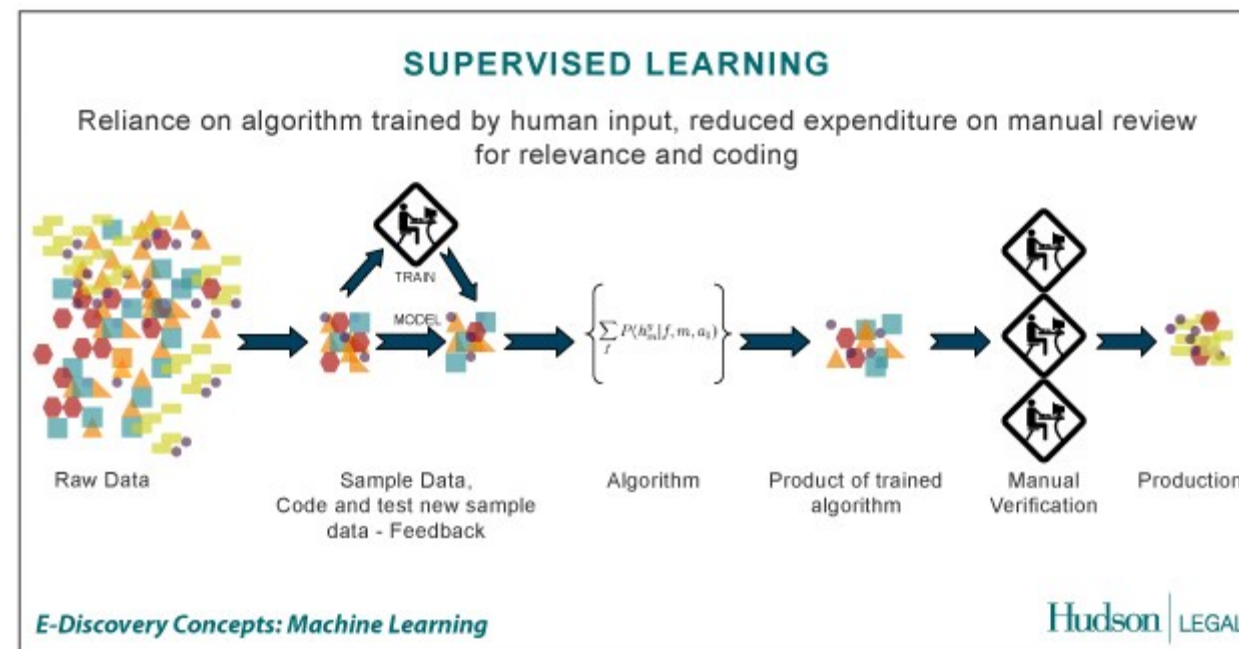
## Limitations

- A **bad** choice of **model** may lead to **bad results**
- Assumption that the **data is well separated in clusters**
  - Otherwise it may lead to bad results
- Gain over supervised learning is **only worth when the number of labeled examples (L) is small**
  - If L is large, then the gain is small



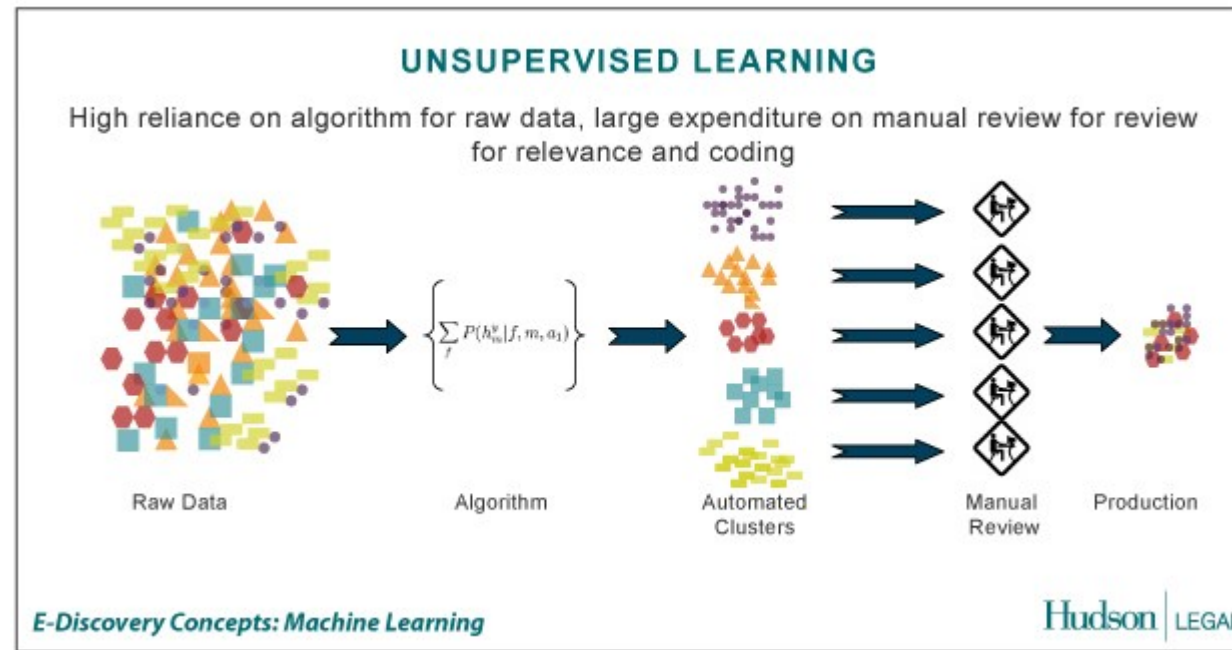
# Overview

# Overview - SUPERVISED



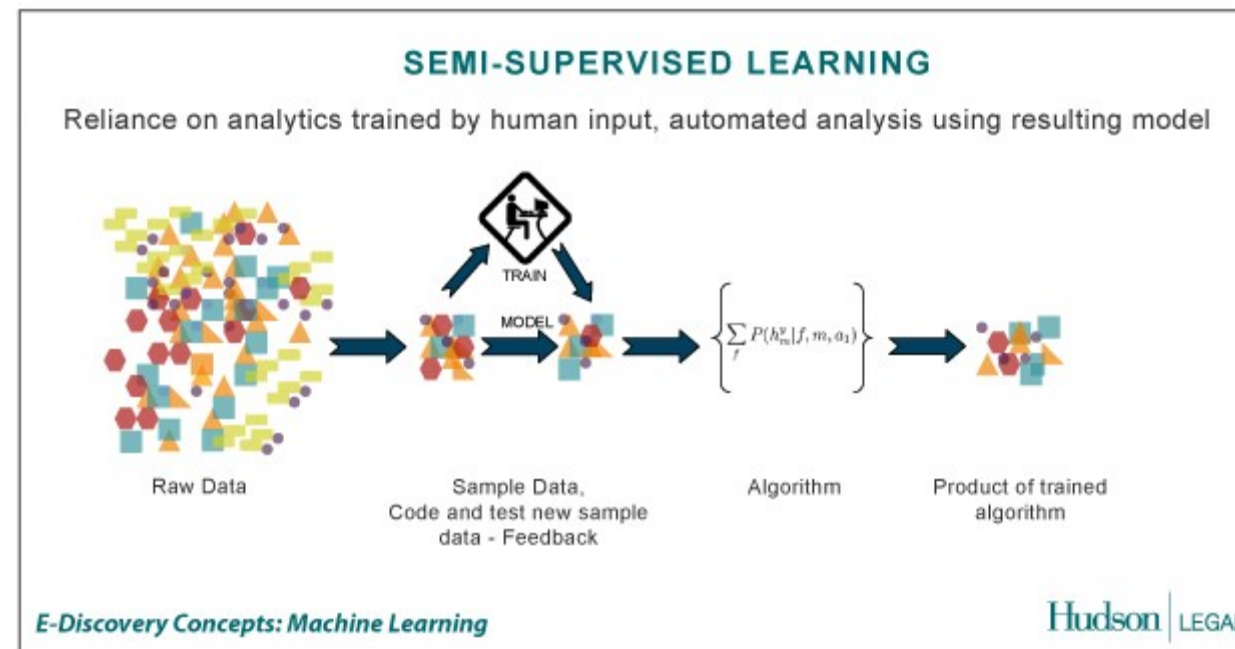
- Require a large amount of labeled samples
- Requires human intervention to label many samples
  - Time consuming & boring task
- Requires small human intervention to evaluate the results

# Overview - UNSUPERVISED



- Does not require labeled samples
- Can explore data by itself
  - Interesting for <Big data>
- Importance of proximity measure
  - Not always easy to determine an efficient one
- Requires large human intervention to evaluate the results
  - Results may not make sense for human – hard to evaluate

# Overview – SEMI-SUPERVISED

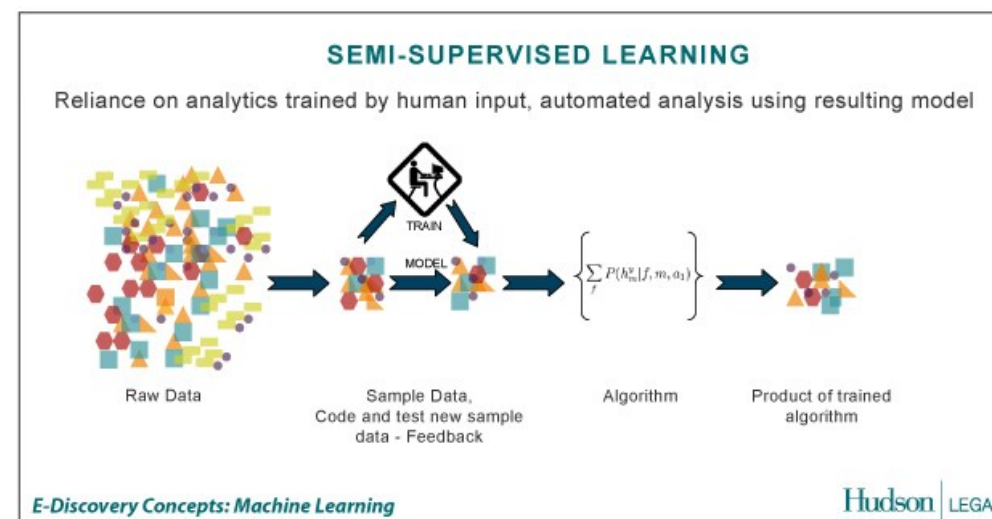
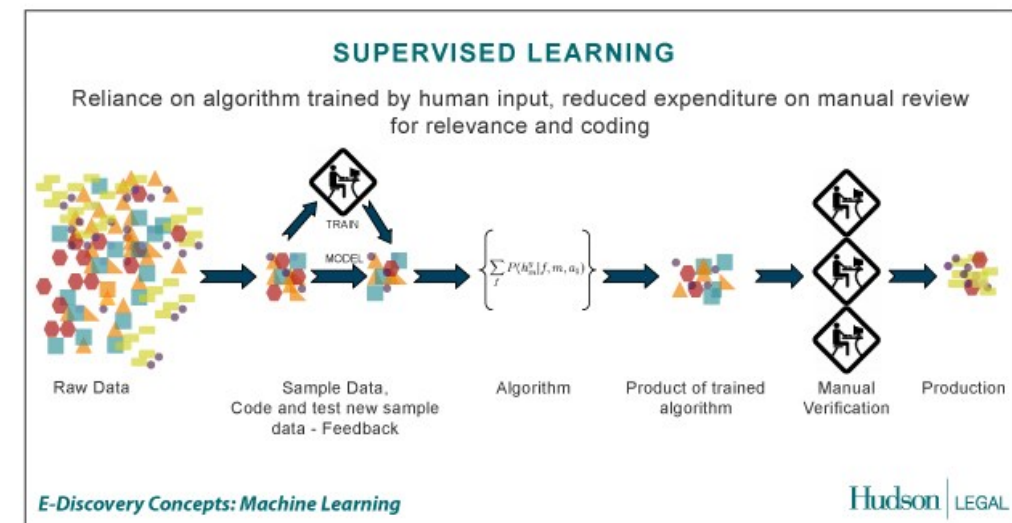
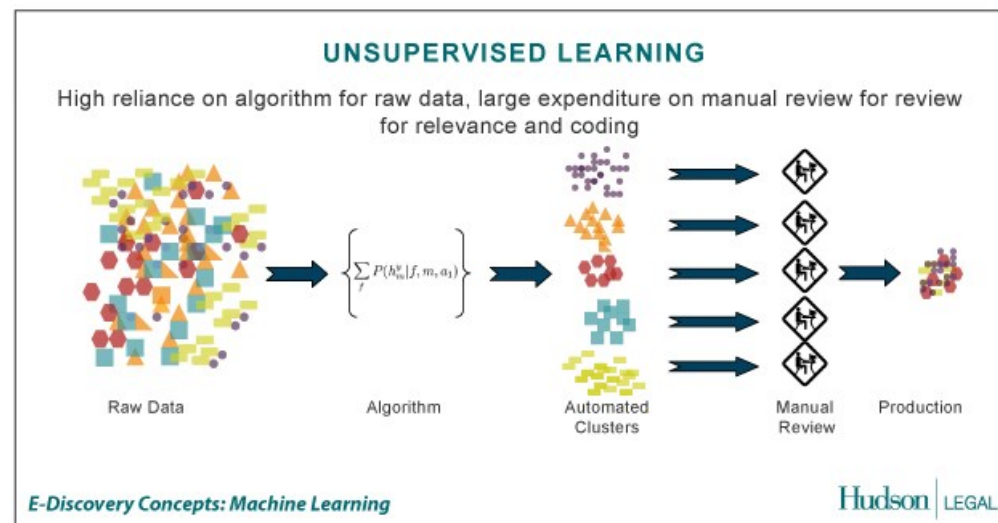


- Requires a few labeled samples
  - Requires human intervention to label a few samples
- Importance of the model!
  - Bad model may lead to erroneous classification
- Cannot be applied to any data
  - Classes must be well separated



# Conclusion

**Computer do not classify by themselves – you always have to teach them something!!**



Computer do not classify by themselves – **you always have to tell/teach them something!!**

**UNSUPERVISED LEARNING**  
 High reliance on algorithm for raw data, large expenditure on manual review for review for relevance and coding

Features?


Proximity measure?

Objective function?

Clusters

Review

*E-Discovery Concepts: Machine Learning*



**SUPERVISED LEARNING**  
 Reliance on algorithm trained by human input, reduced expenditure on manual review for relevance and coding

Features?


Labels?

Algorithm?

Parameters?

ion

*E-Discovery Concepts: Machine Learning*



**SEMI-SUPERVISED LEARNING**  
 Reliance on analytics trained by human input, automated analysis using resulting model


Features?

Model?

Algorithm?

Confidence?

*E-Discovery Concepts: Machine Learning*



# What You Should Know

- Unsupervised Learning
  - Clustering concepts and goals
  - Self-Organizing Maps
    - The 3 essentials concepts
    - Short description of the algorithm
    - Output visualization techniques
- Semi-Supervised learning
  - Describe the concepts
  - Explain the 3 examples of SSL (self, co & active learning )
- Overview
  - Main differences between methods