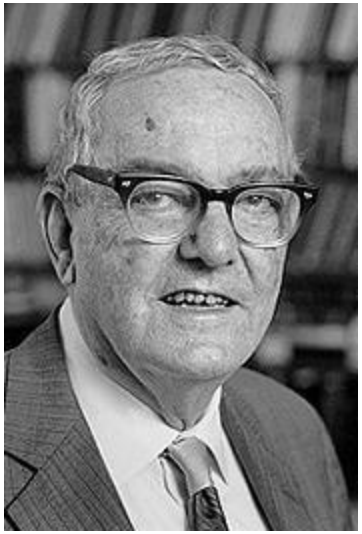
Introduction to Machine Learning

Mohsen Afsharchi

Machine Learning

- **Herbert Alexander Simon**: "Learning is any process by which a system improves performance from experience."
- "Machine Learning is concerned with computer programs that automatically improve their performance through experience. "



Herbert Simon
Turing Award 1975
Nobel Prize in Economics 1978

Why Machine Learning?

- Develop systems that can automatically adapt and customize themselves to individual users.
  - Personalized news or mail filter
- Discover new knowledge from large databases (***data mining***).
  - Market basket analysis (e.g. diapers and beer)
- Ability to mimic human and replace certain monotonous tasks - which require some intelligence.
    - like recognizing handwritten characters
- Develop systems that are too difficult/expensive to construct manually because they require specific detailed skills or knowledge tuned to a specific task (knowledge engineering bottleneck).

Why now?

- Flood of available data (especially with the advent of the Internet)
- Increasing computational power
- Growing progress in available algorithms and theory developed by researchers
- Increasing support from industries

ML Applications

The concept of learning in a ML system

- Learning = Improving with experience at some task
  - Improve over task *T*,
  - With respect to performance measure, *P*
  - Based on experience, *E*.

Motivating Example
Learning to Filter Spam

Example: Spam Filtering

Spam - is all email the user does not want to receive and has not asked to receive

T: Identify Spam Emails

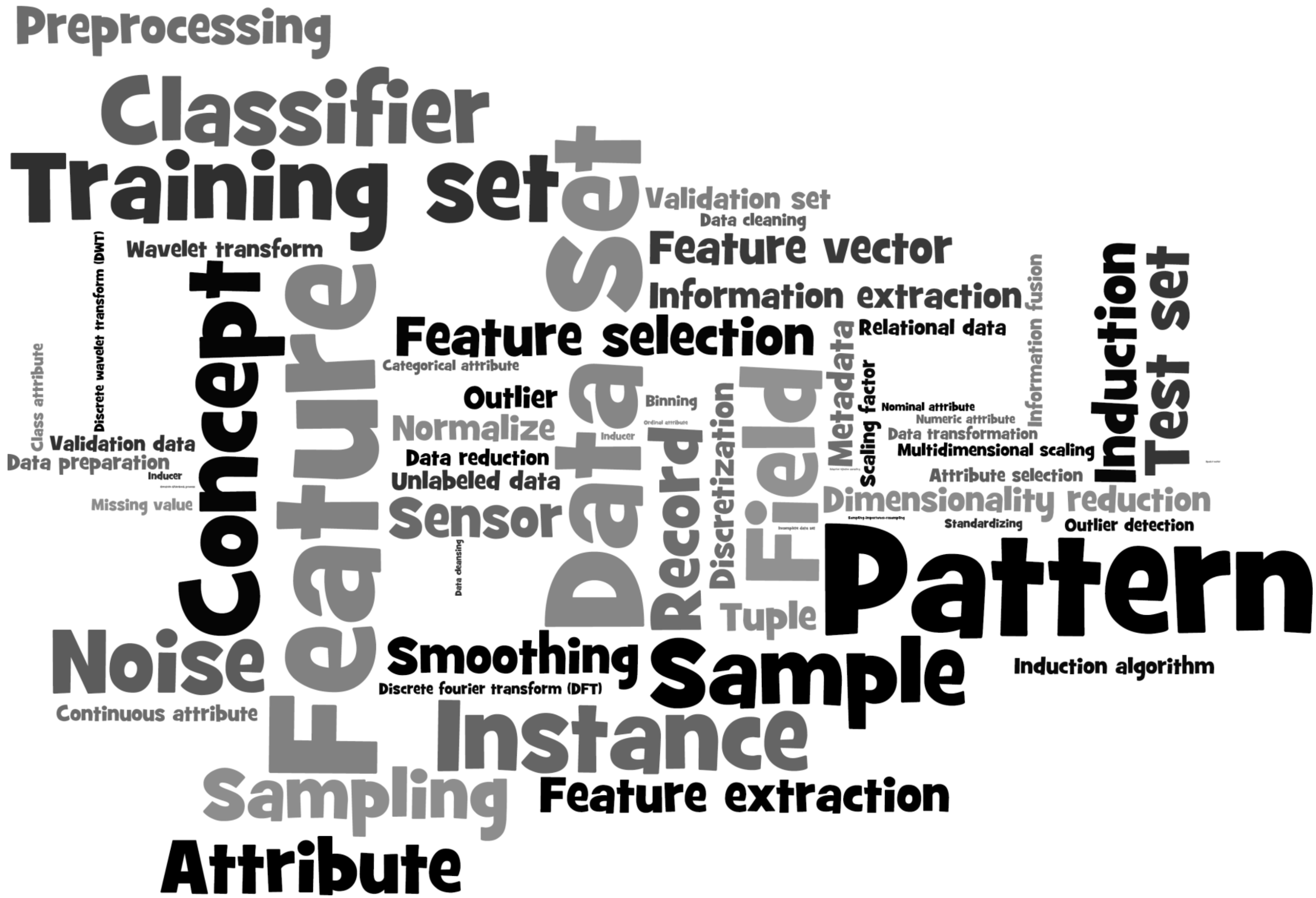
P:

% of spam emails that were filtered

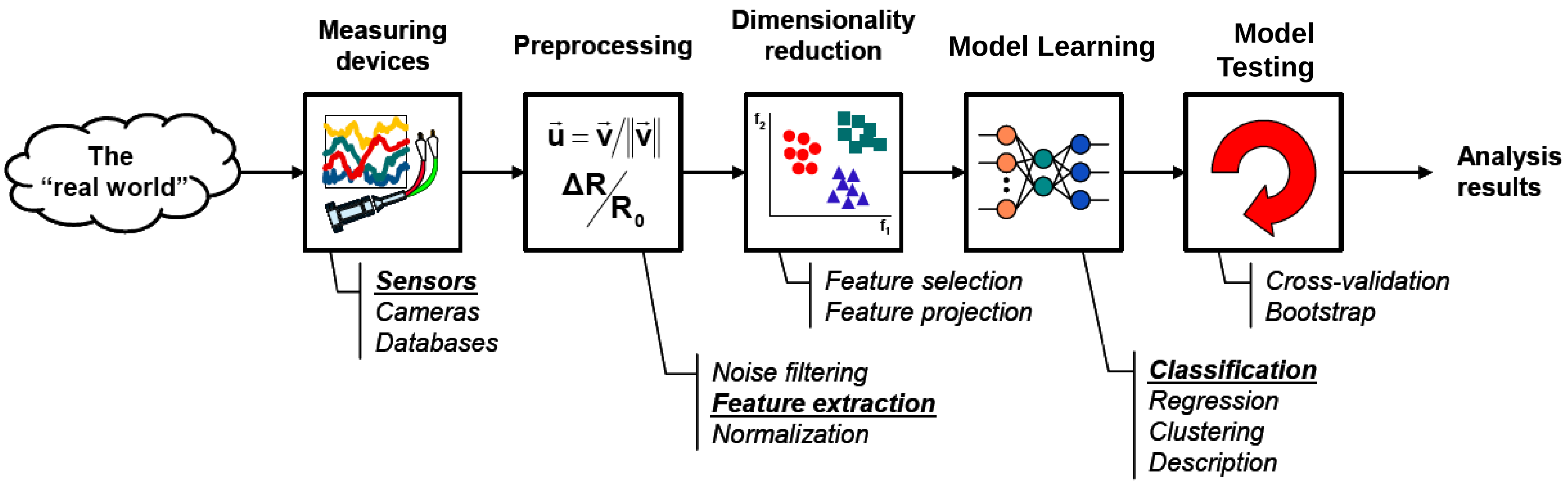
% of ham/ (non-spam) emails that were incorrectly filtered-out

E: a database of emails that were labelled by users

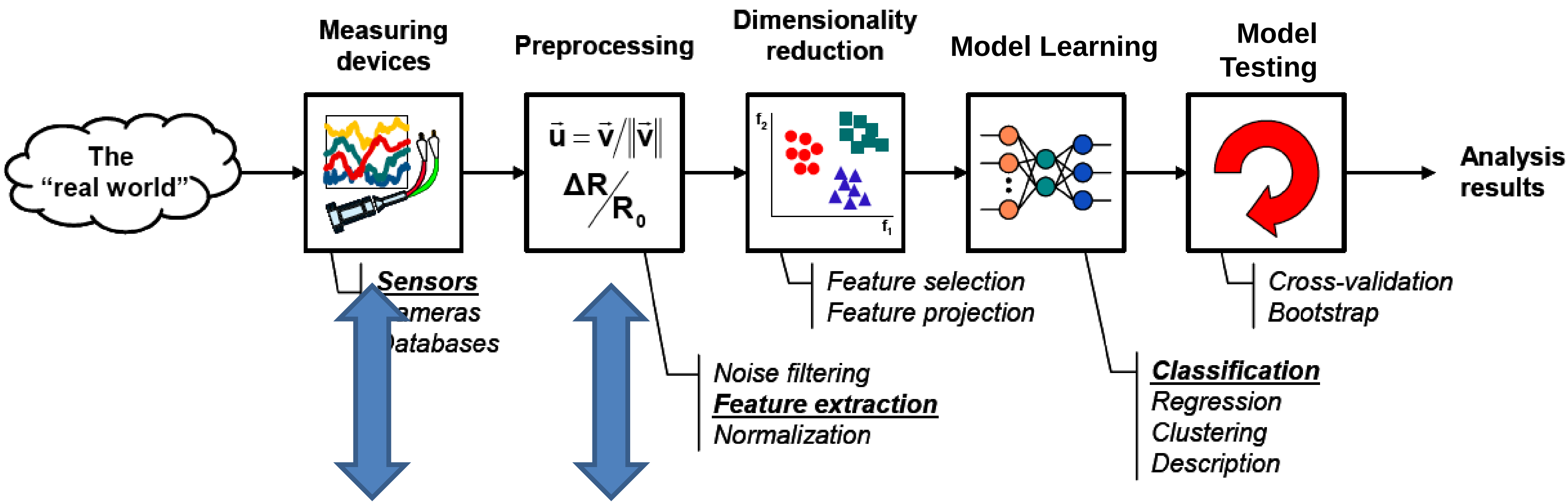
The Learning Process



The Learning Process



The Learning Process in our Example



Email Server

- Number of recipients
- Size of message
- Number of attachments
- Number of "re's" in the subject line

…

Data Set

Input Attributes

Target Attribute

Number of new Recipients	Email Length (K)	Country (IP)	Customer Type	Email Type
0	2	Germany	Gold	Ham
1	4	Germany	Silver	Ham
5	2	Nigeria	Bronze	Spam
2	4	Russia	Bronze	Spam
3	4	Germany	Bronze	Ham
0	1	USA	Silver	Ham
4	2	USA	Silver	Spam

Instances

Numeric

Nominal

Ordinal

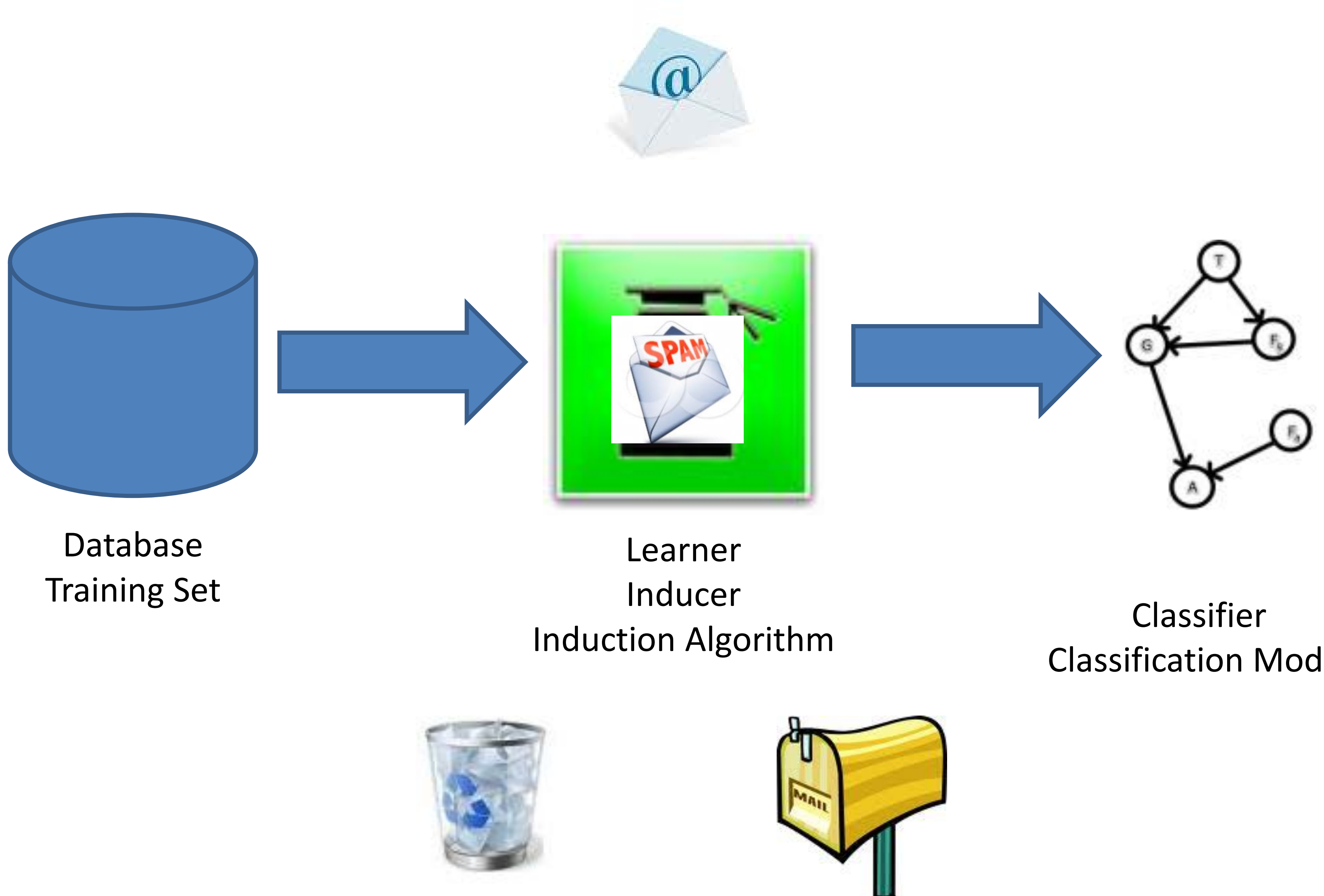
Step 4: Model Learning

Database
Training Set

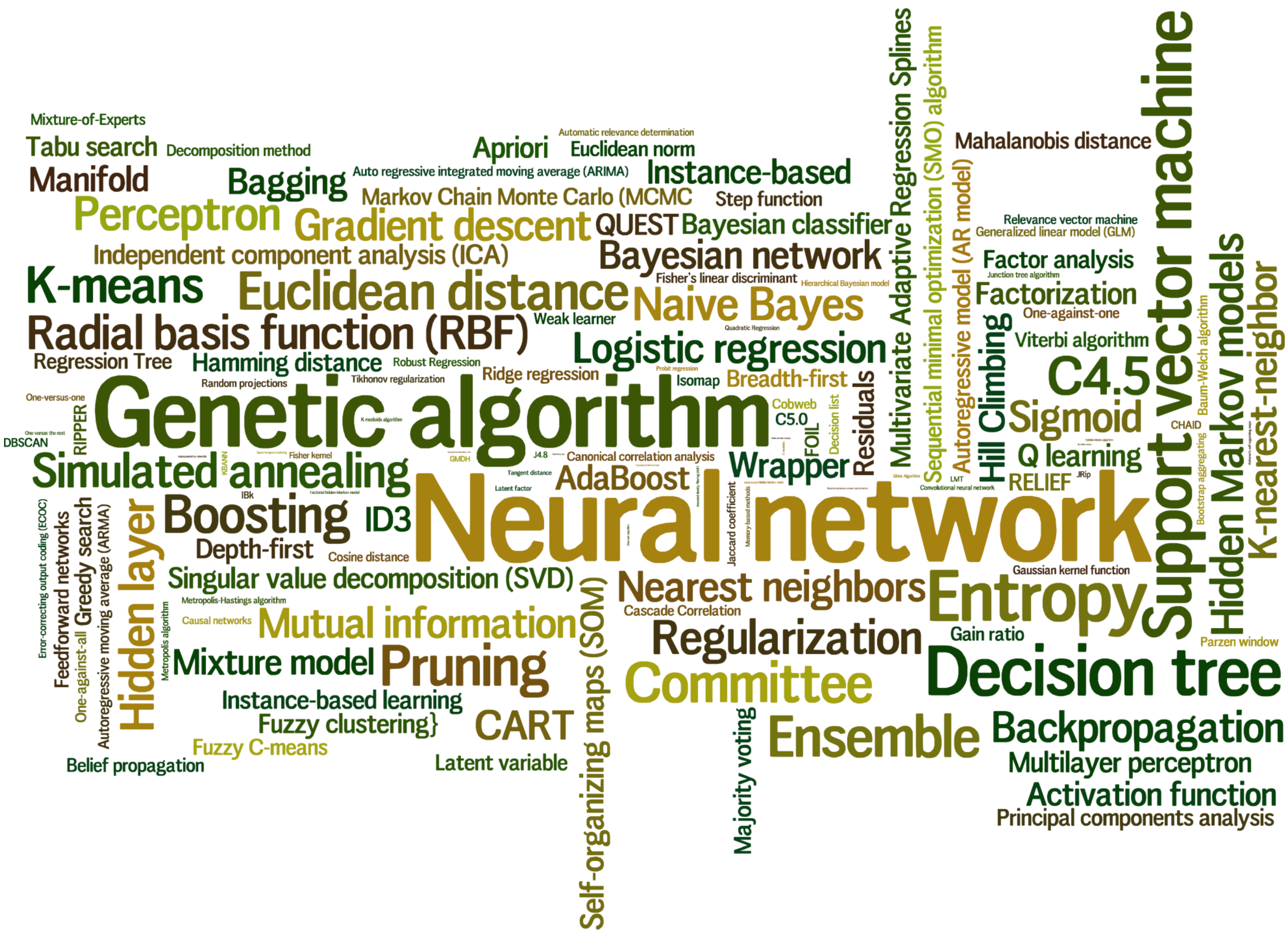
Learner
Inducer
Induction Algorithm

Classifier
Classification Model

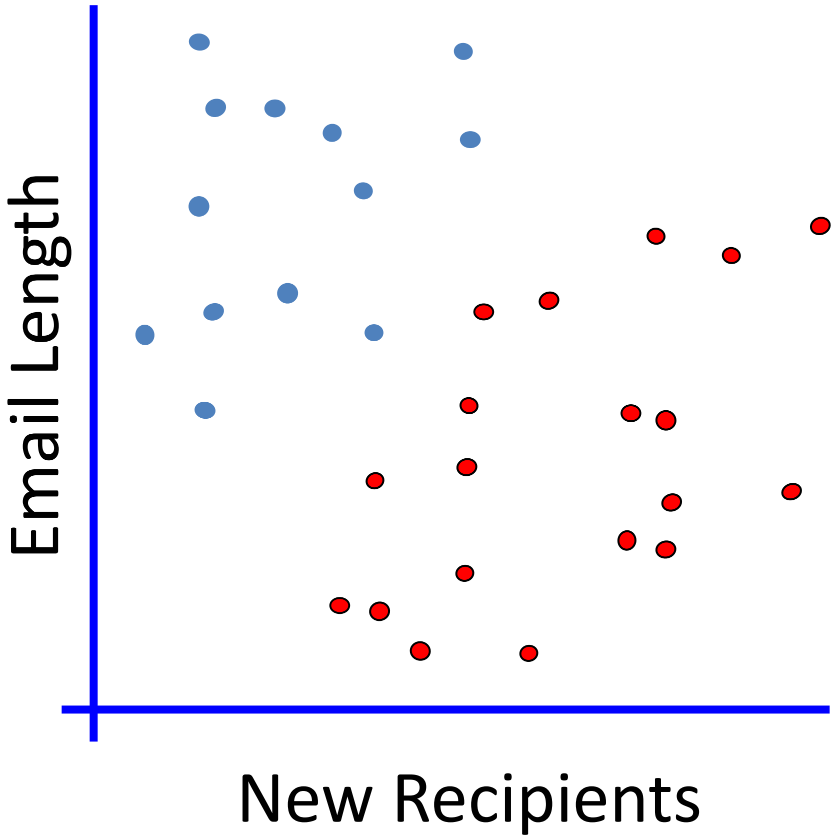
Step 5: Model Testing



Learning Algorithms

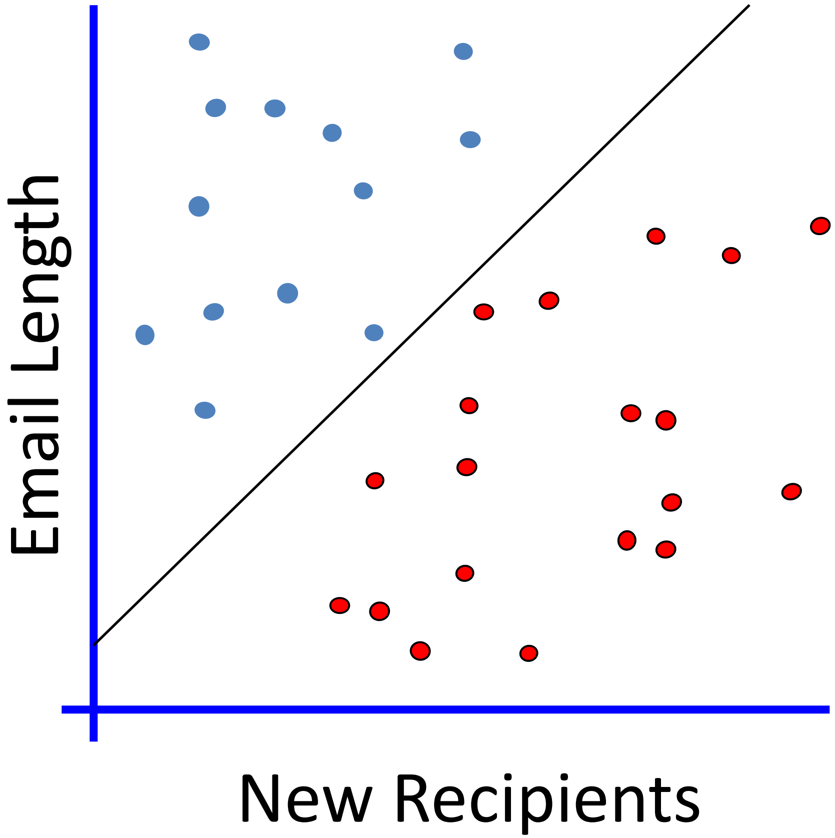


Linear Classifiers



How would you classify this data?

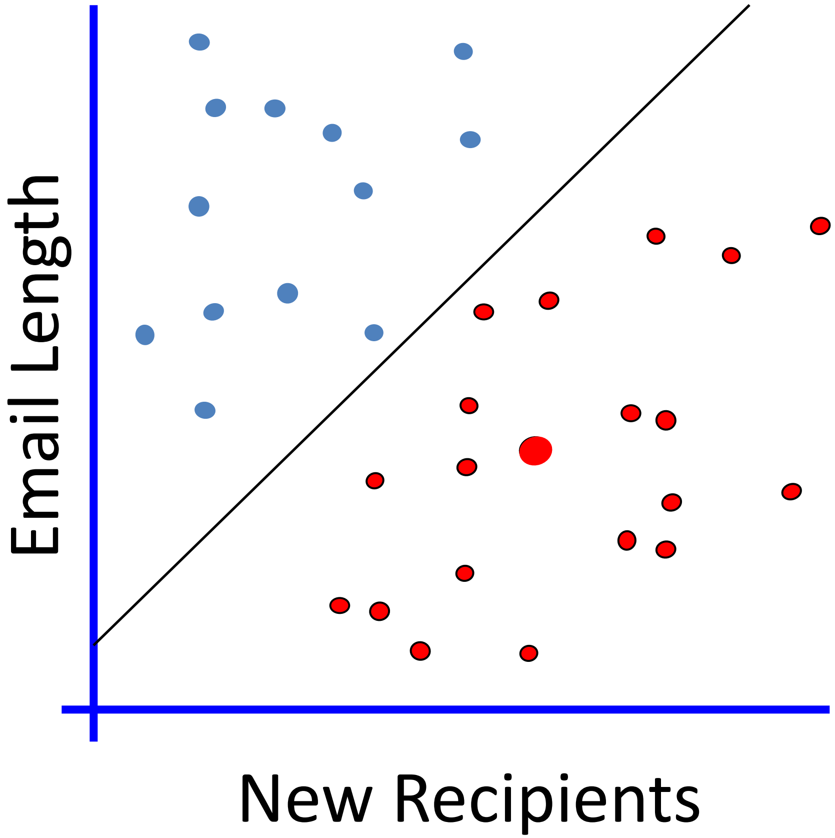
Linear Classifiers



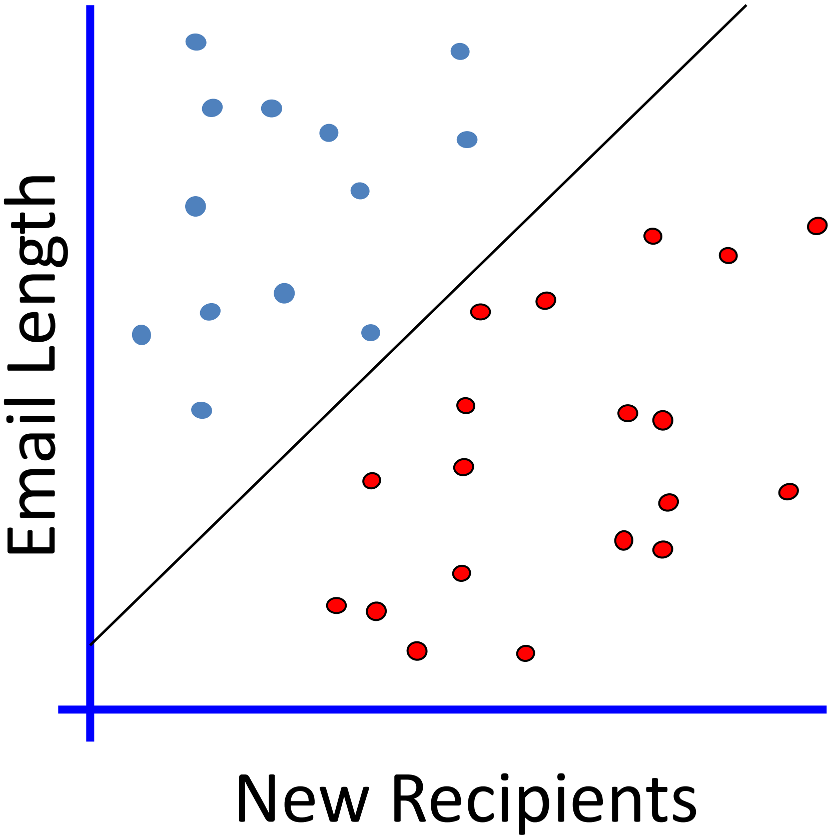
How would you classify this data?

When a new email is sent

1. We first place the new email in the space
2. Classify it according to the subspace in which it resides

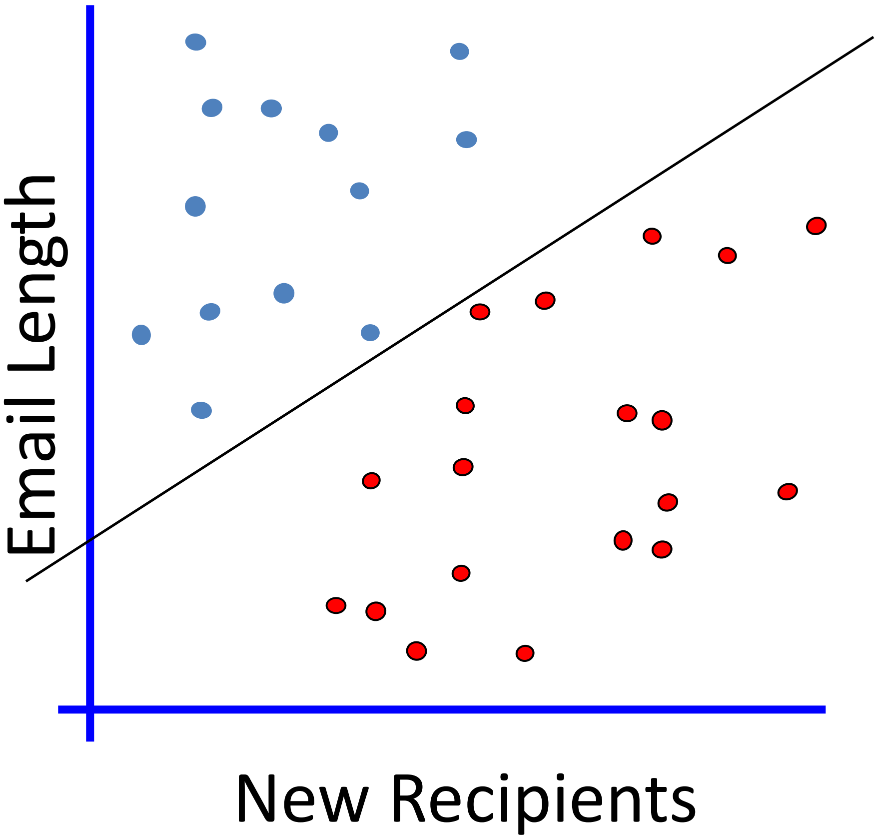


Linear Classifiers



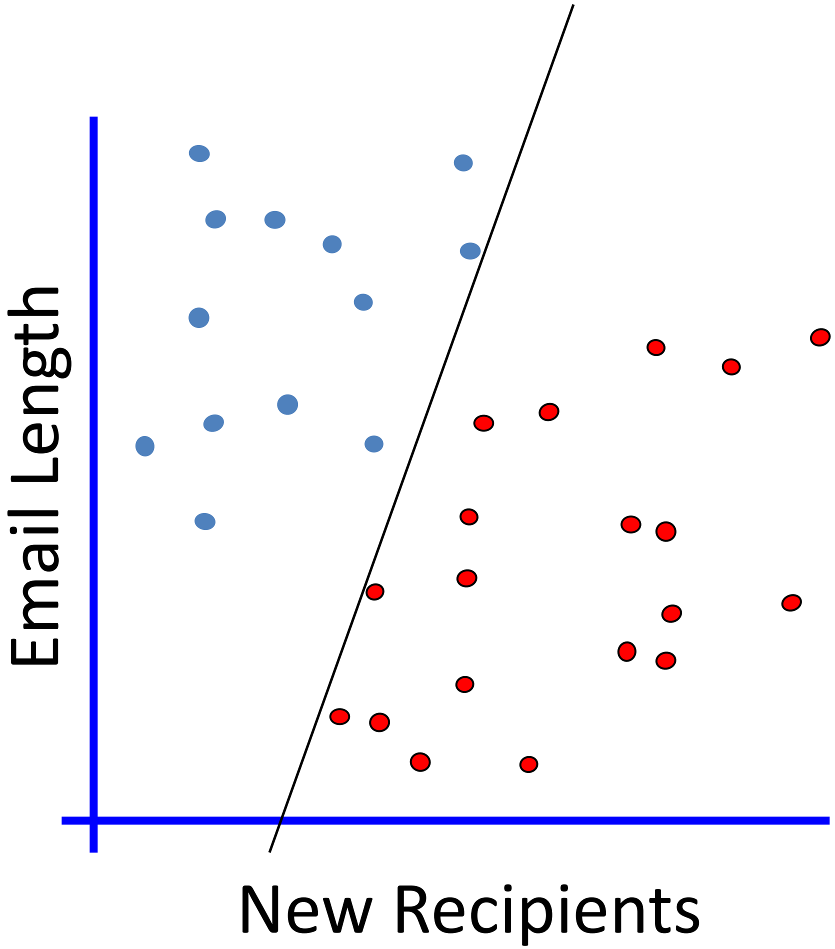
How would you classify this data?

Linear Classifiers



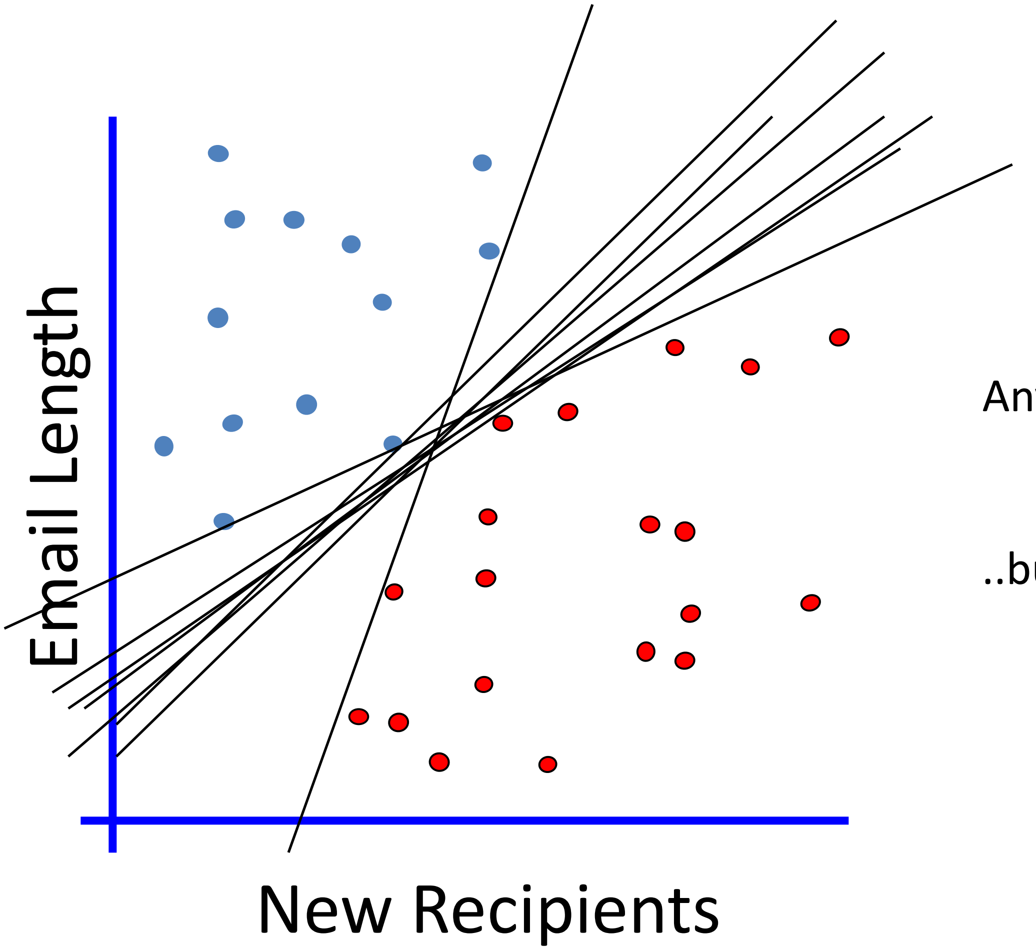
How would you classify this data?

Linear Classifiers



How would you classify this data?

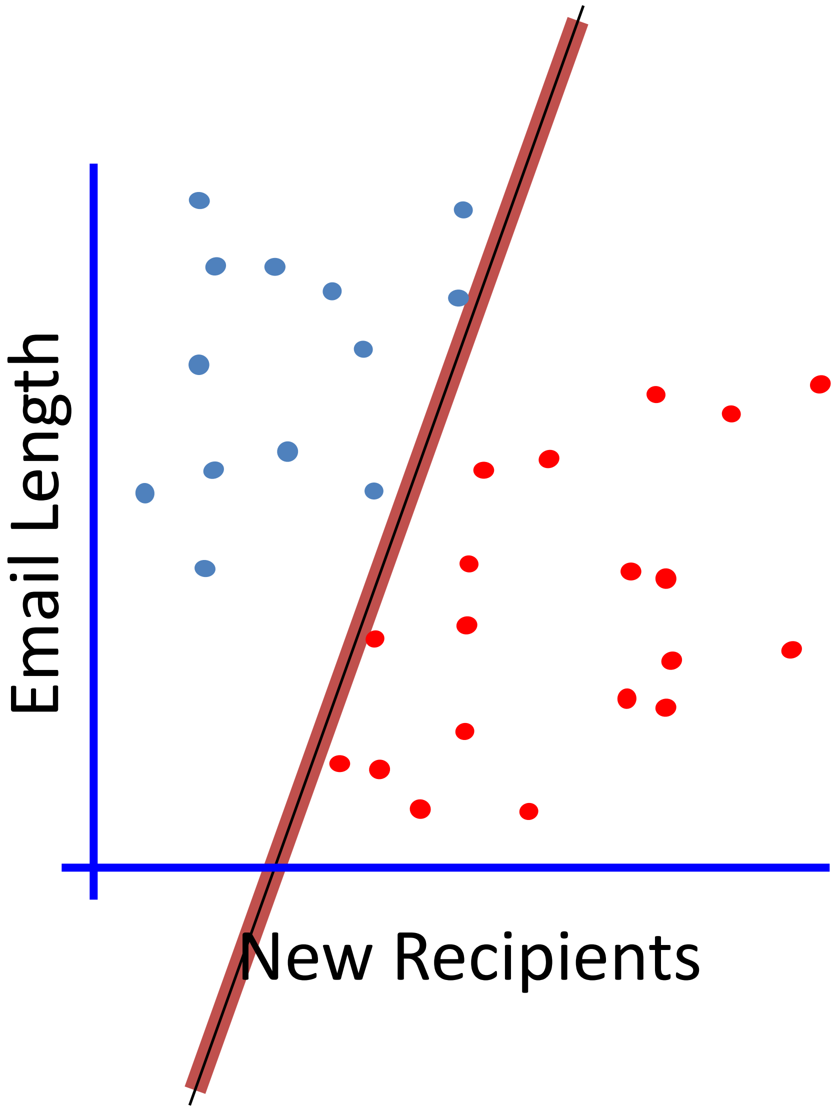
Linear Classifiers



Any of these would be fine..

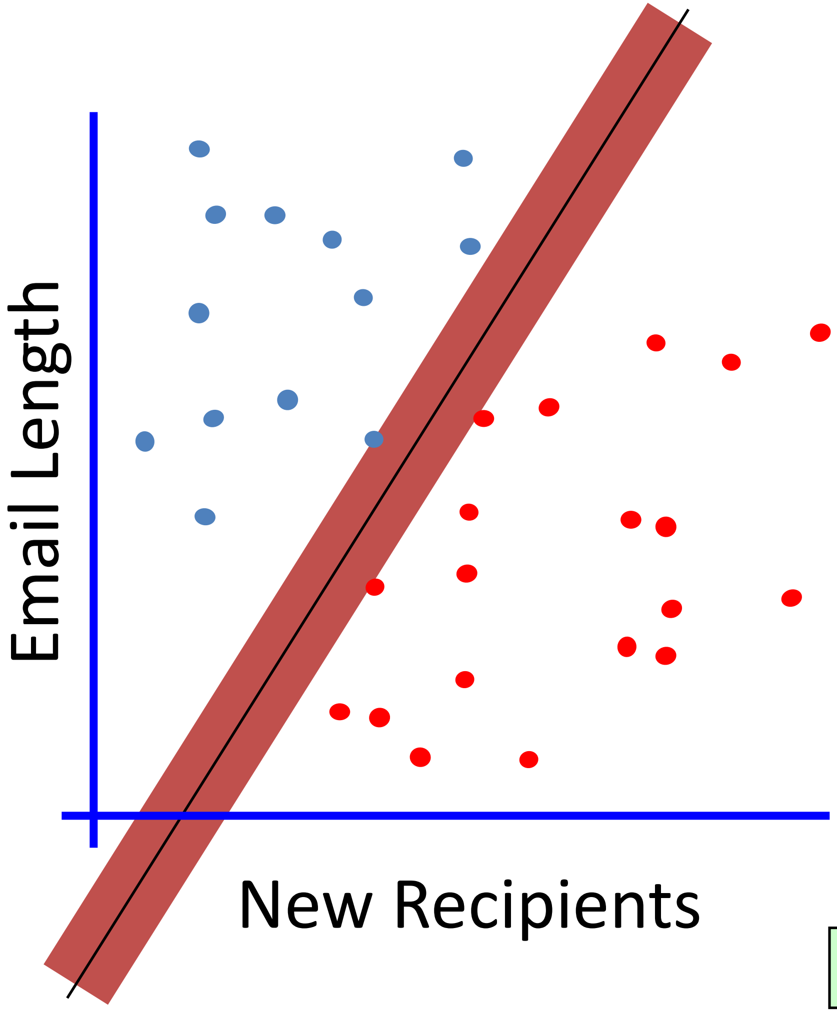
..but which is best?

Classifier Margin



Define the margin of a linear classifier as the width that the boundary could be increased by before hitting a datapoint.

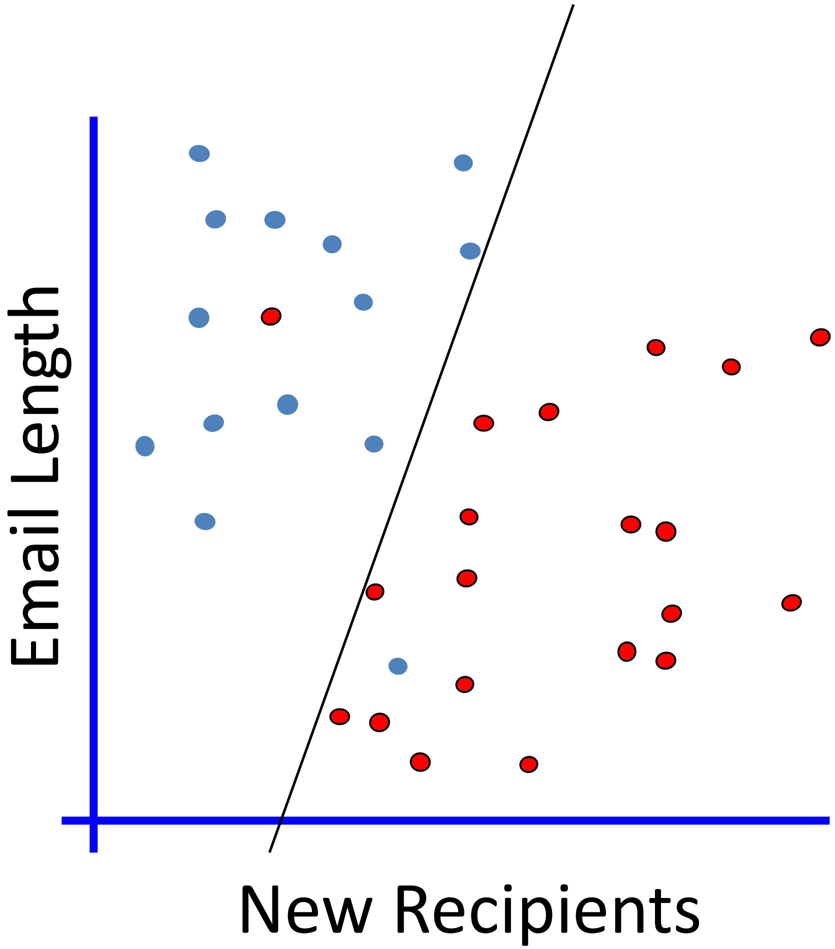
Maximum Margin



The maximum margin linear classifier is the linear classifier with the, maximum margin.
This is the simplest kind of SVM (Called an LSVM)

Linear SVM

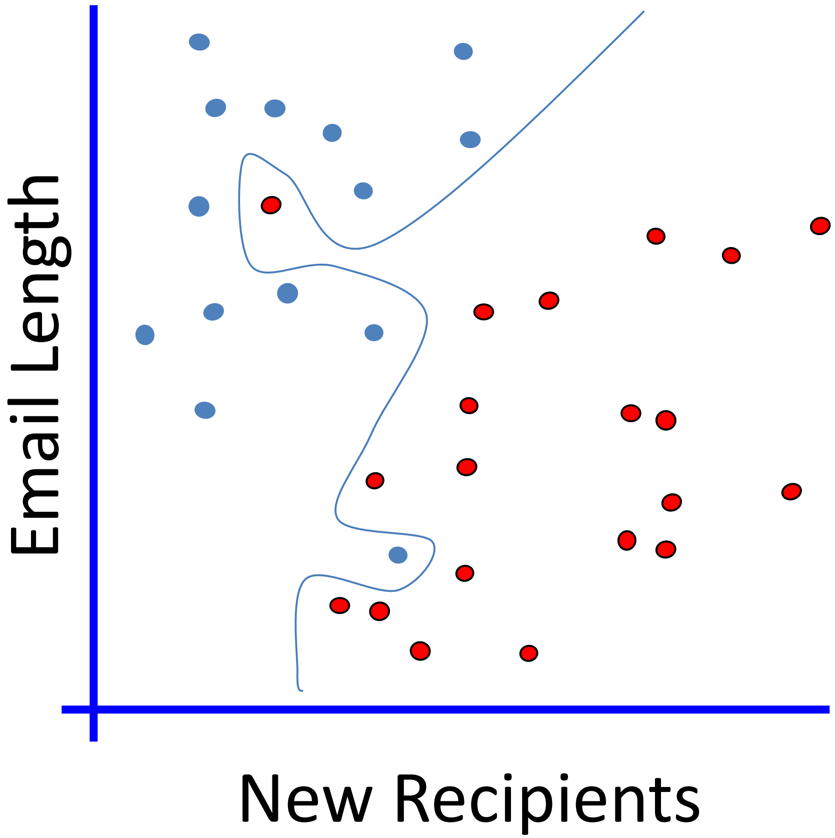
No Linear Classifier can cover all instances



How would you classify this data?

- Ideally, the best decision boundary should be the one which provides an optimal performance such as in the following figure

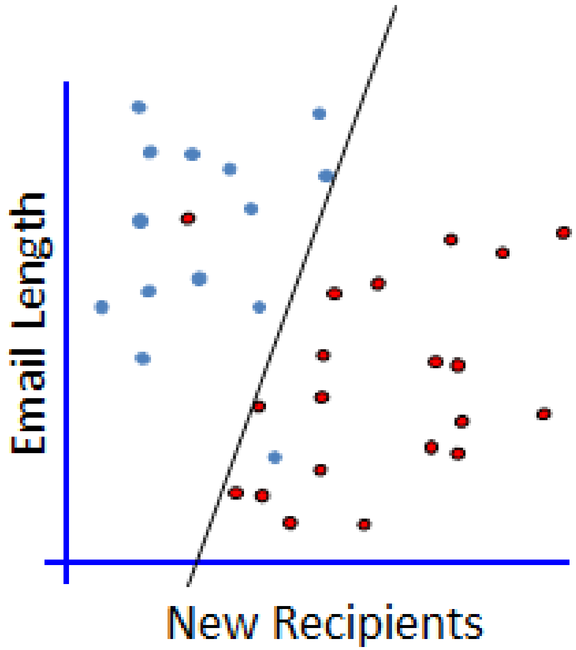
No Linear Classifier can cover all instances



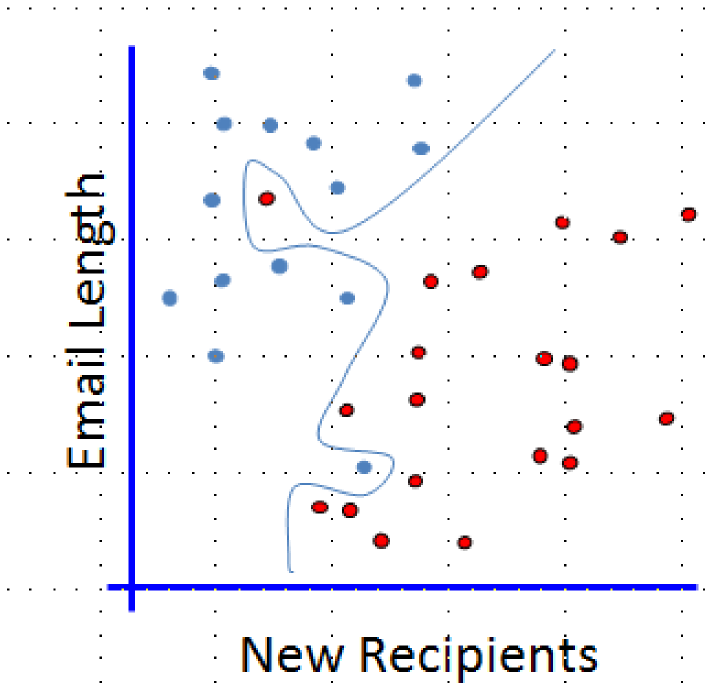
- However, our satisfaction is premature because the central aim of designing a classifier is to correctly classify novel input

Issue of generalization!

Which one?



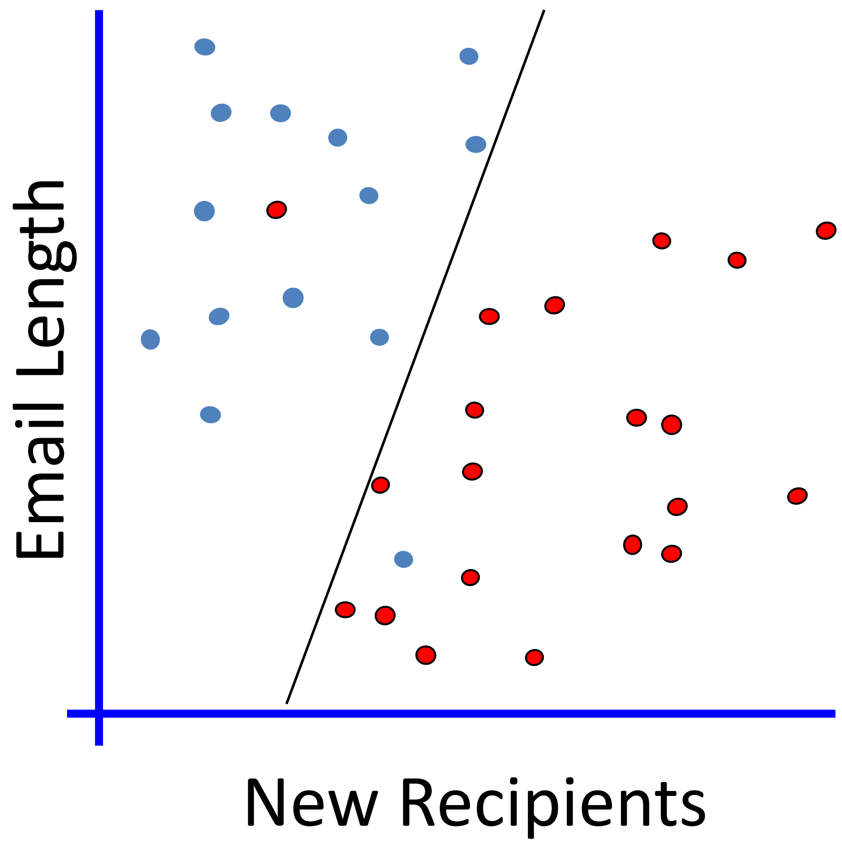
2 Errors
Simple model



0 Errors
Complicated model

Evaluating What's Been Learned

1. We randomly select a portion of the data to be used for training (the training set)
2. Train the model on the training set.
3. Once the model is trained, we run the model on the remaining instances (the test set) to see how it performs

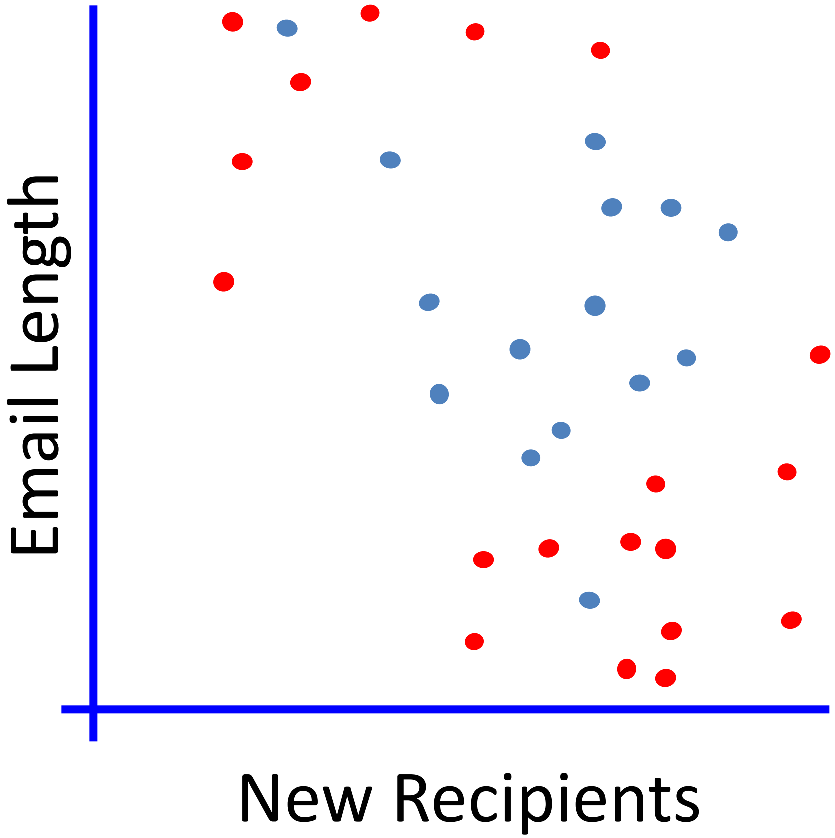


Confusion Matrix

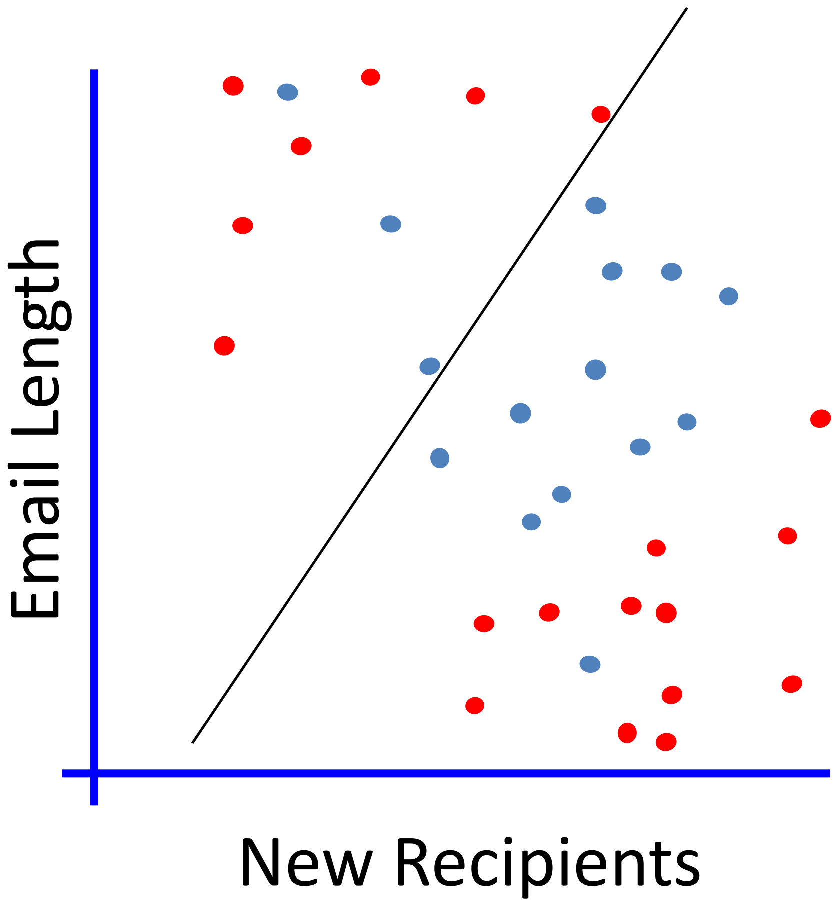
Classified As

Actual	Blue	Red
Blue	7	1
Red	0	5

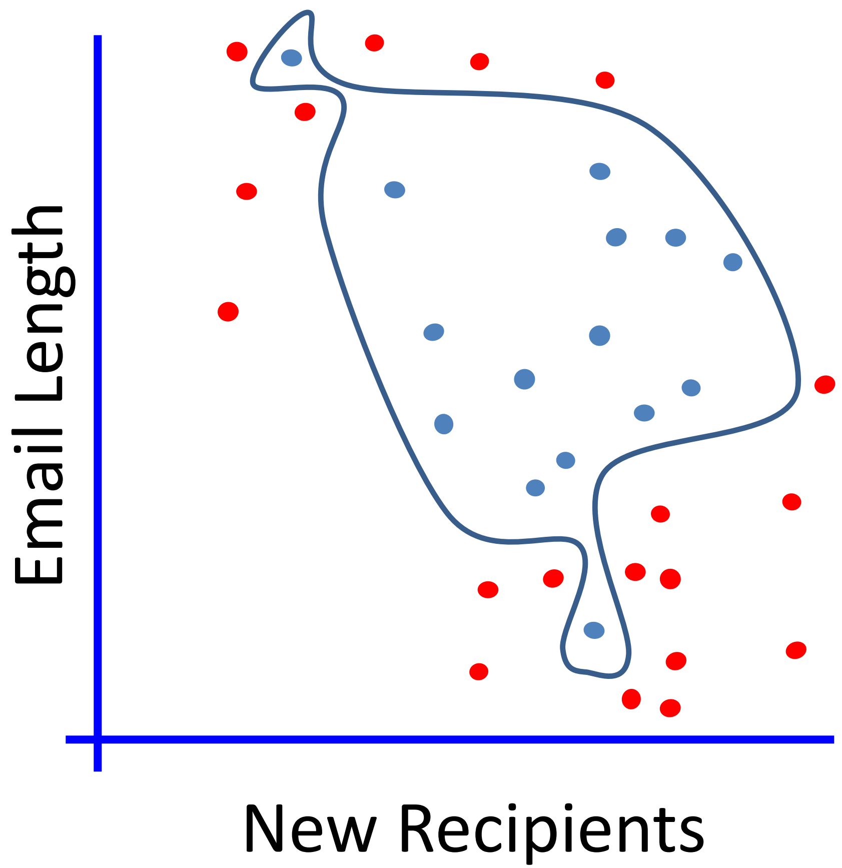
The Non-linearly separable case



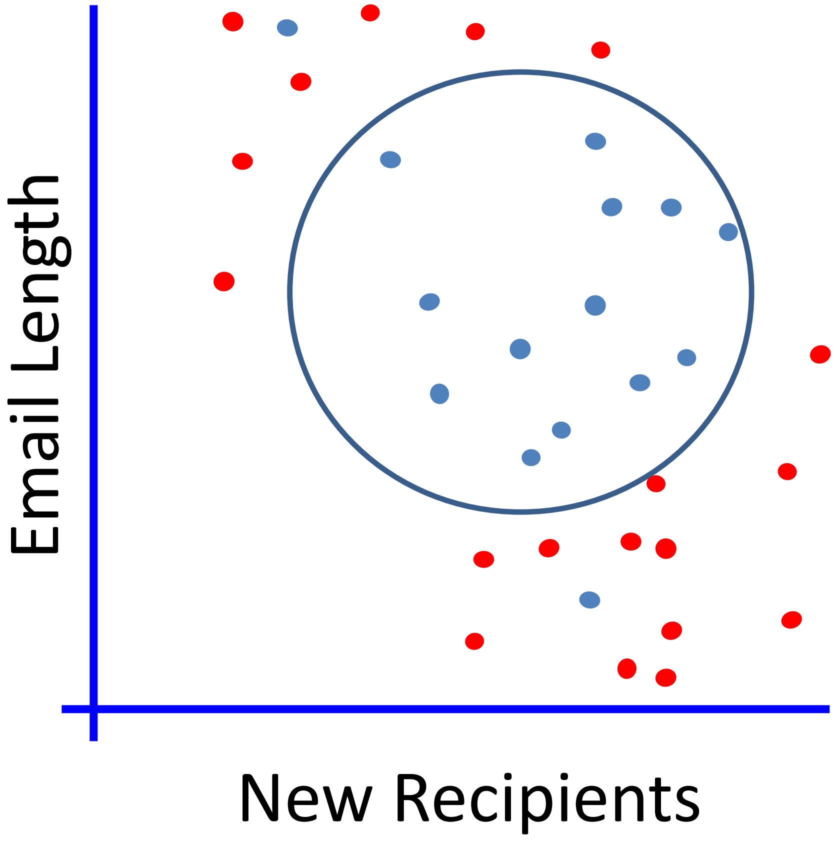
The Non-linearly separable case



The Non-linearly separable case

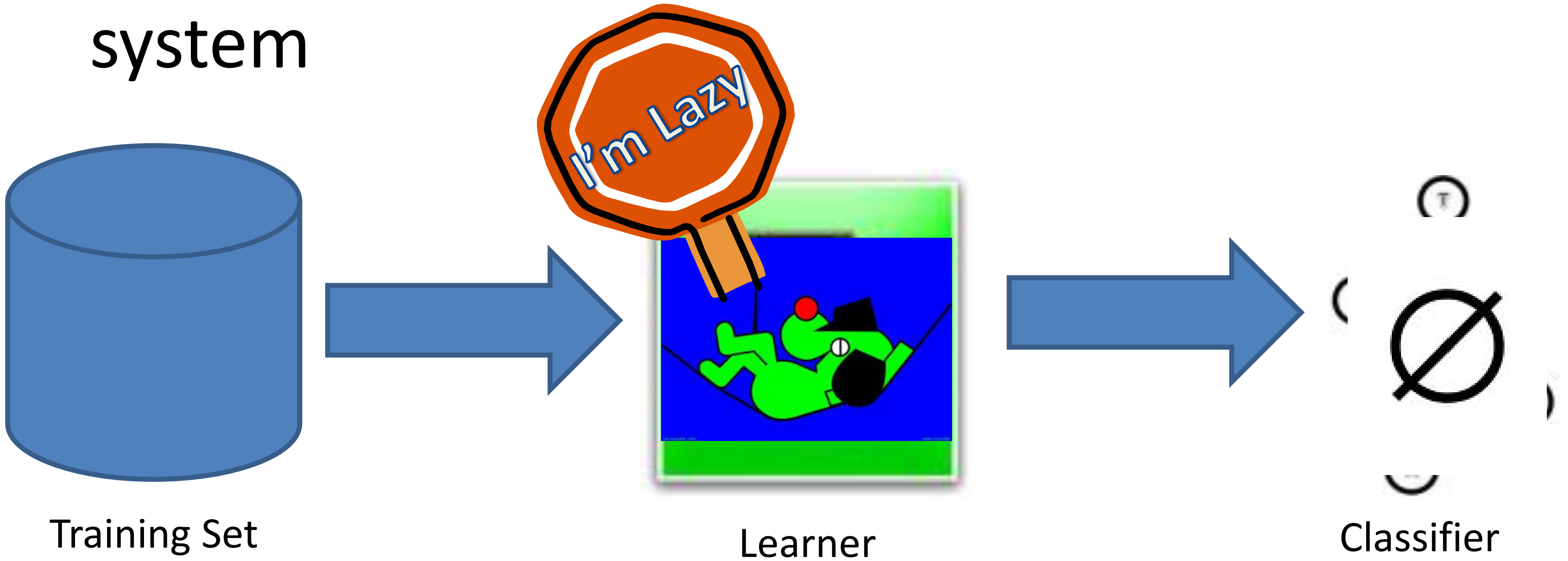


The Non-linearly separable case



Lazy Learners

- Generalization beyond the training data is delayed until a new instance is provided to the system

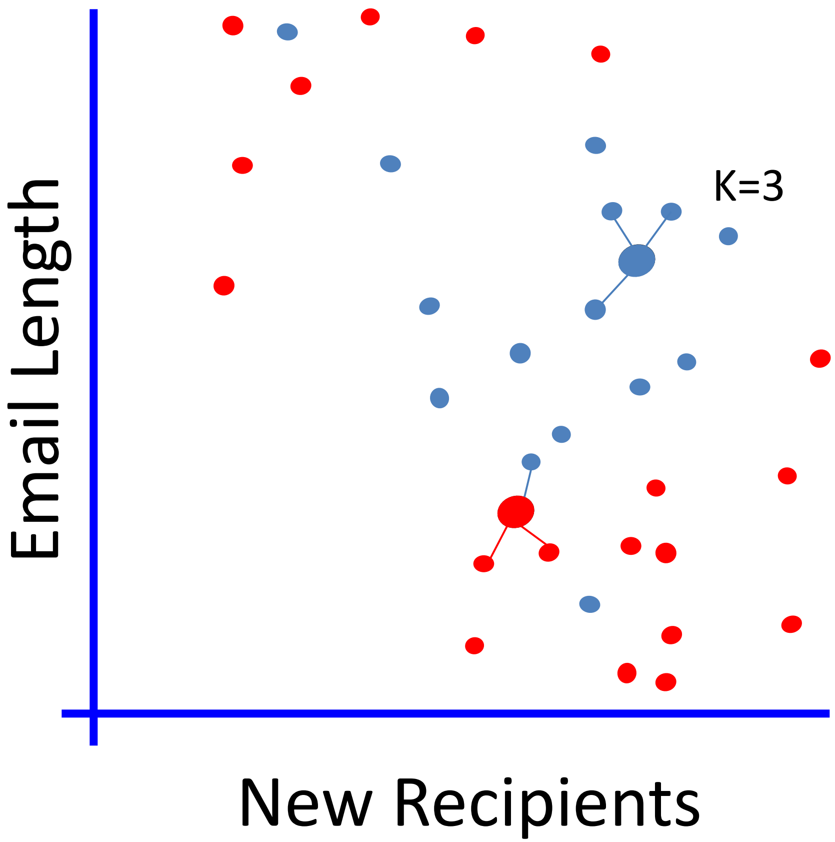


Lazy Learners

Instance-based learning

Lazy Learner: k-Nearest Neighbors

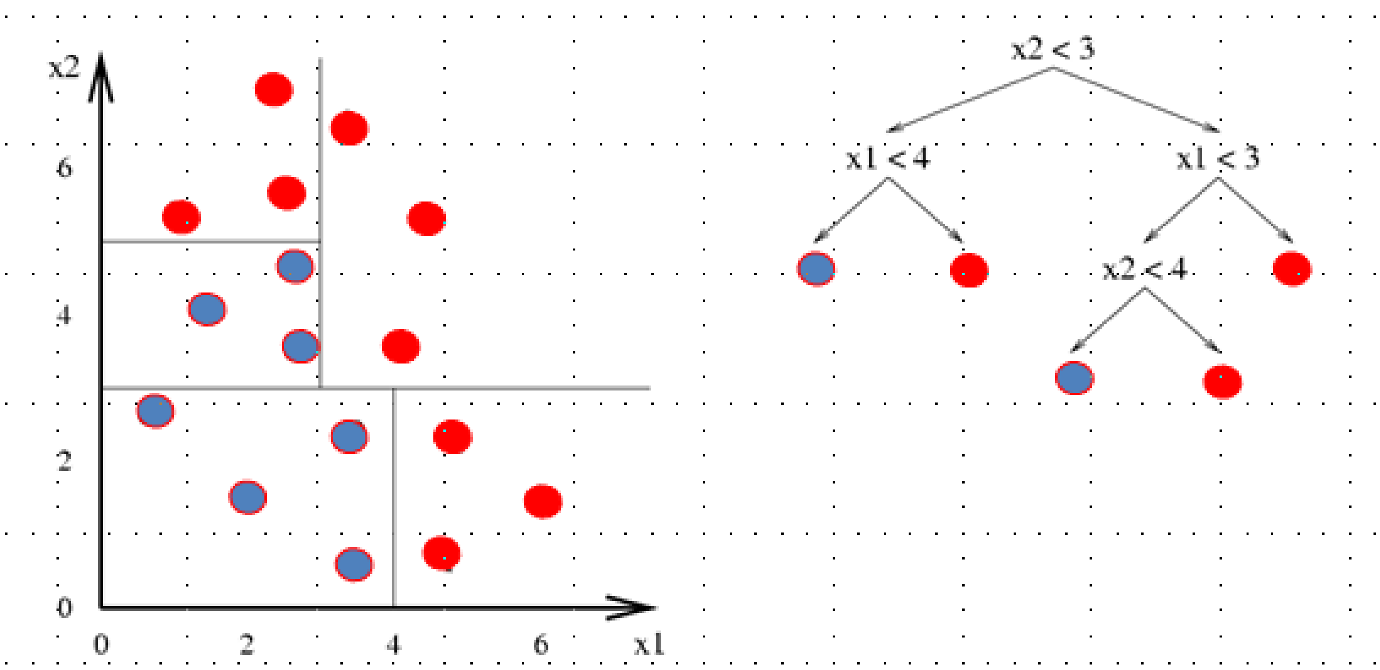
- What should be k?
- Which distance measure should be used?



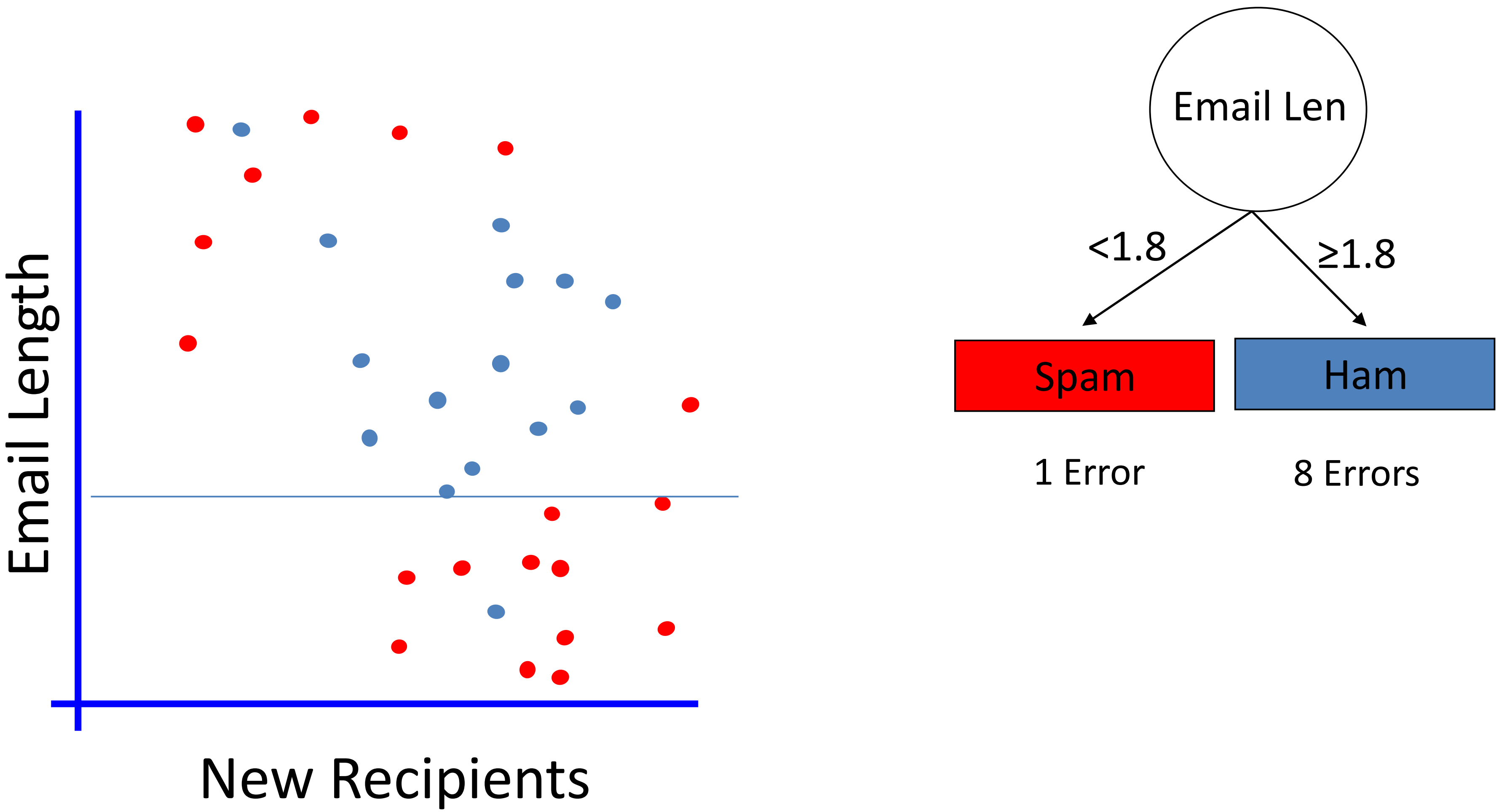
Decision tree

- A flow-chart-like tree structure
- Internal node denotes a test on an attribute
- Branch represents an outcome of the test
- Leaf nodes represent class labels or class distribution

Decision trees divide the feature space into axis-parallel rectangles, and label each rectangle with one of the K classes.

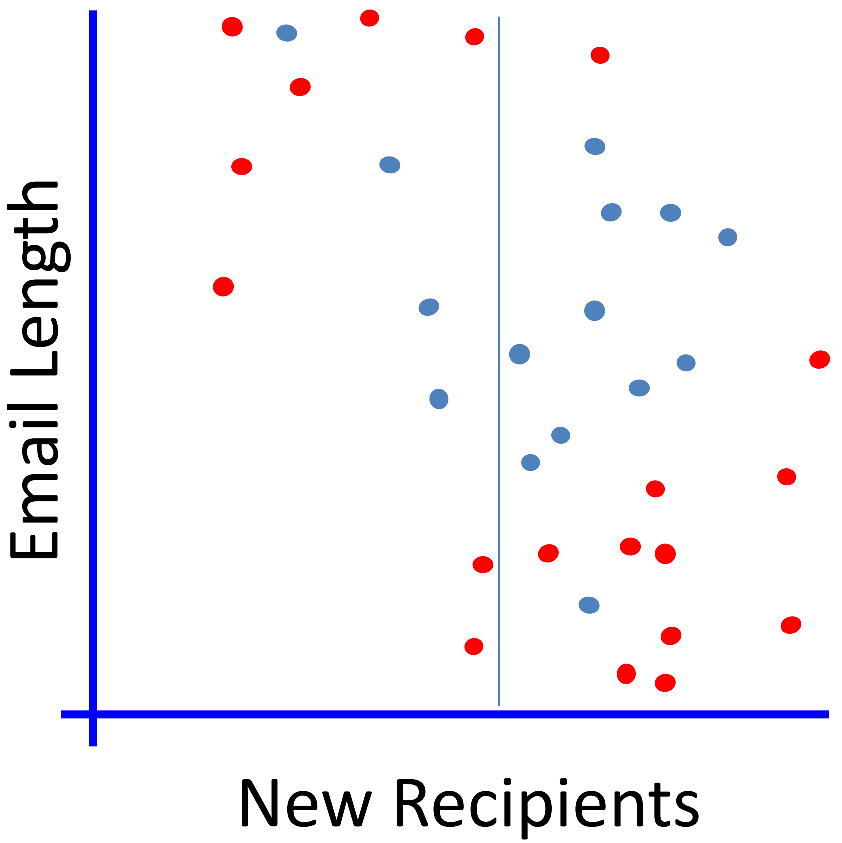


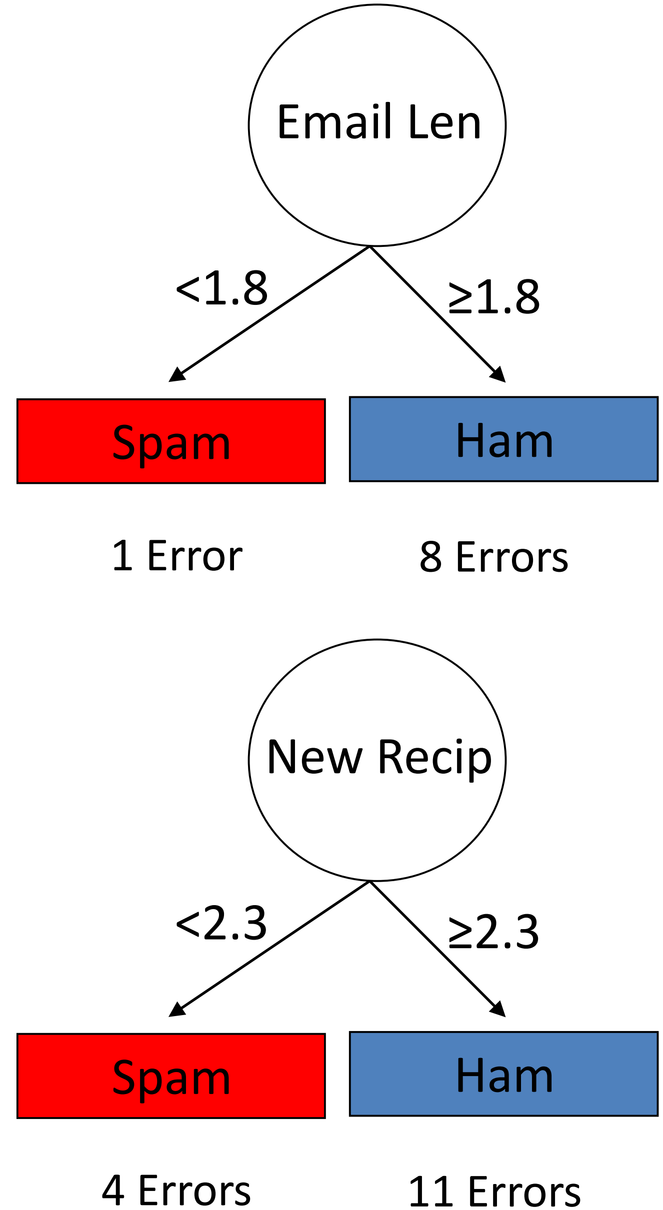
Top Down Induction of Decision Trees



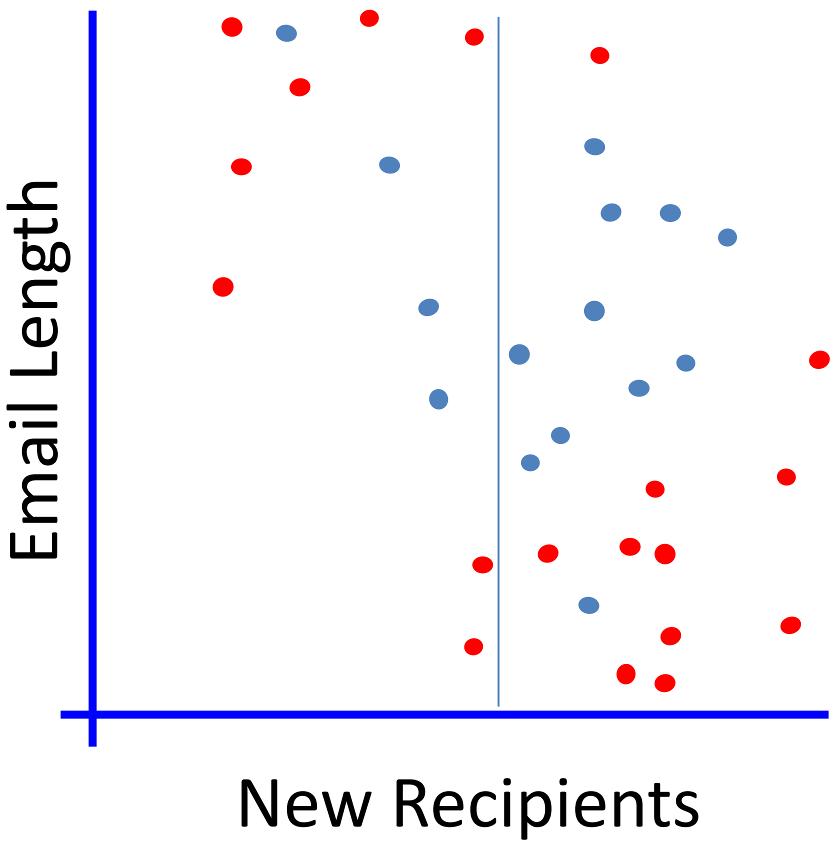
A single level decision tree is also known as Decision Stump

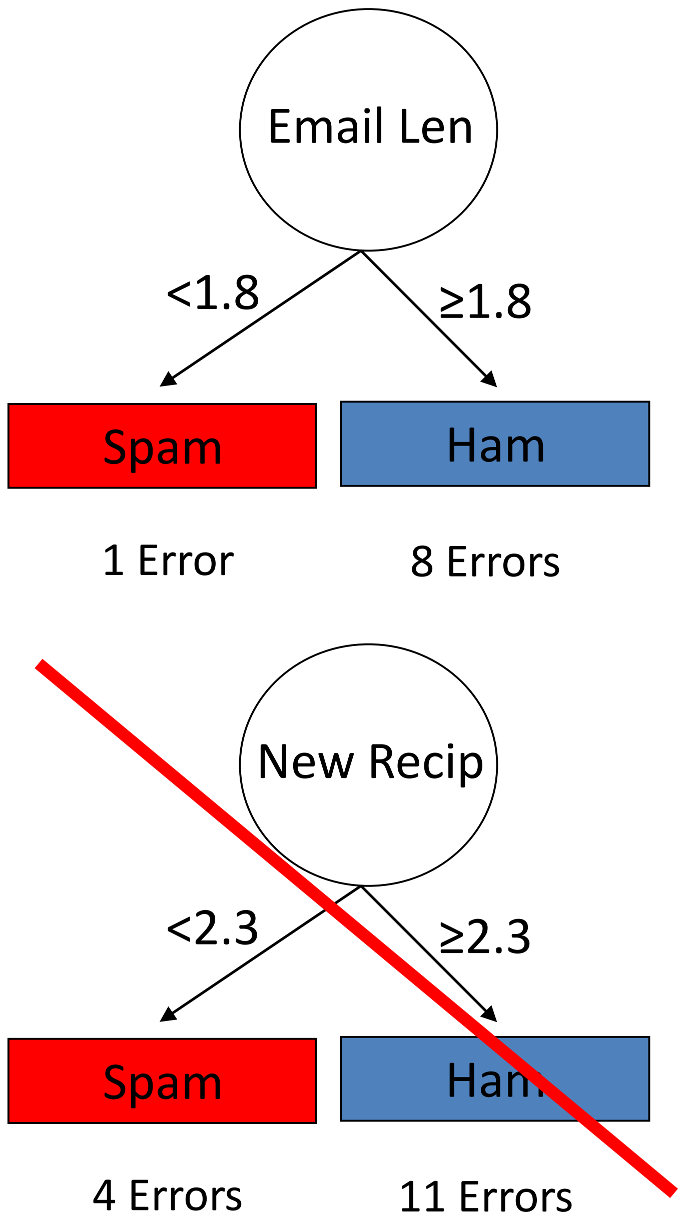
Top Down Induction of Decision Trees



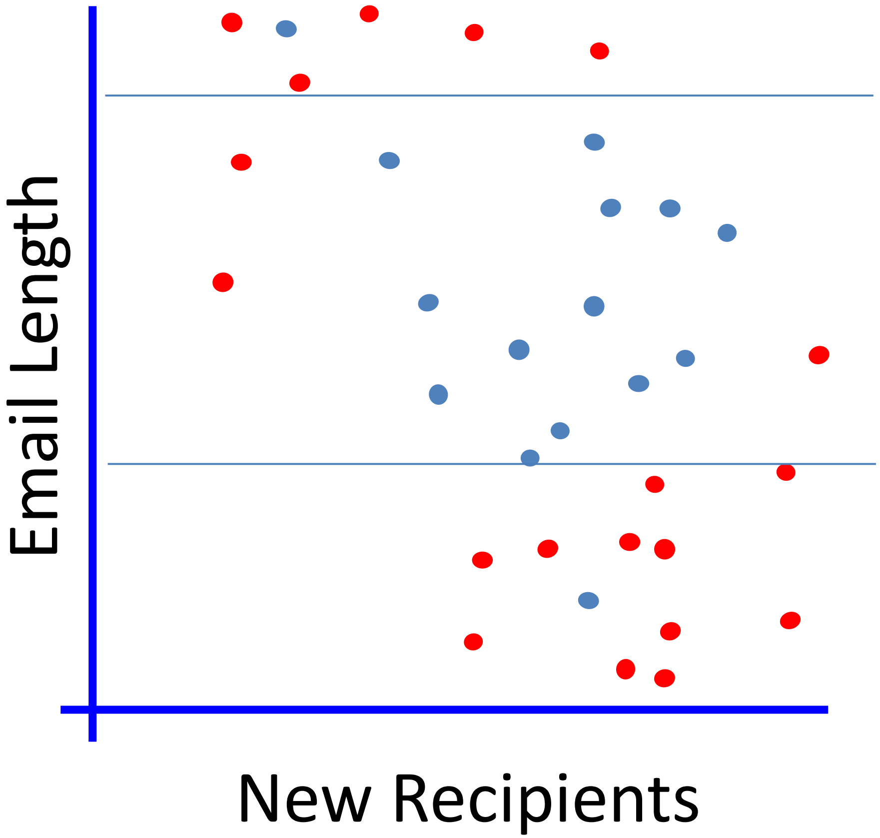


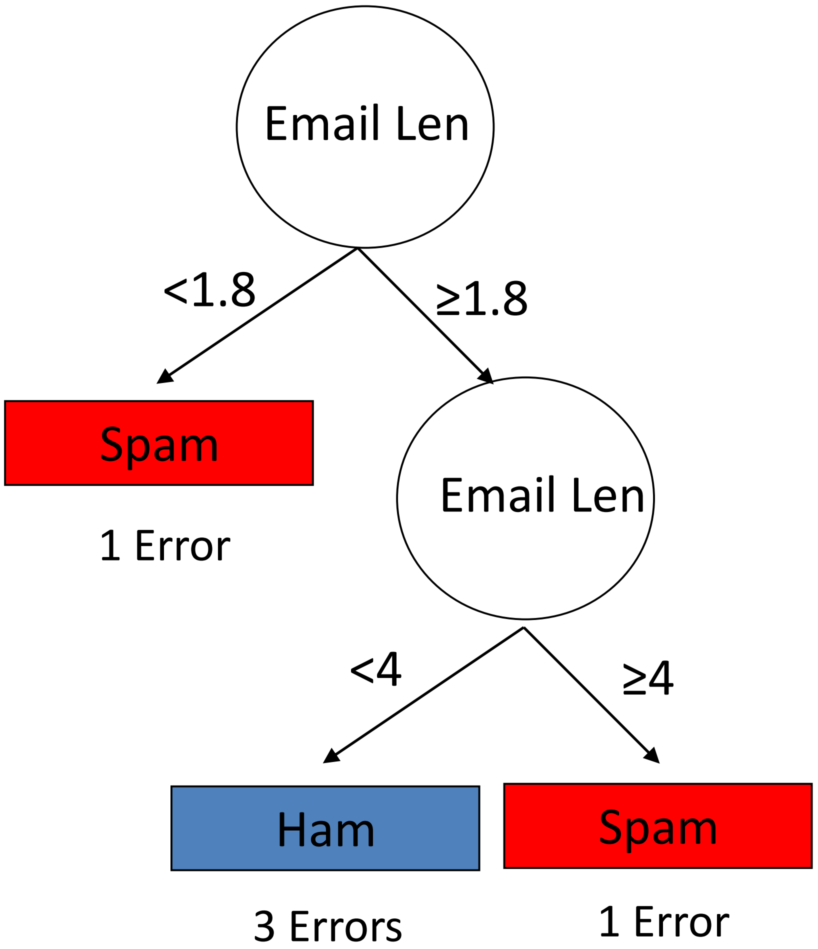
Top Down Induction of Decision Trees



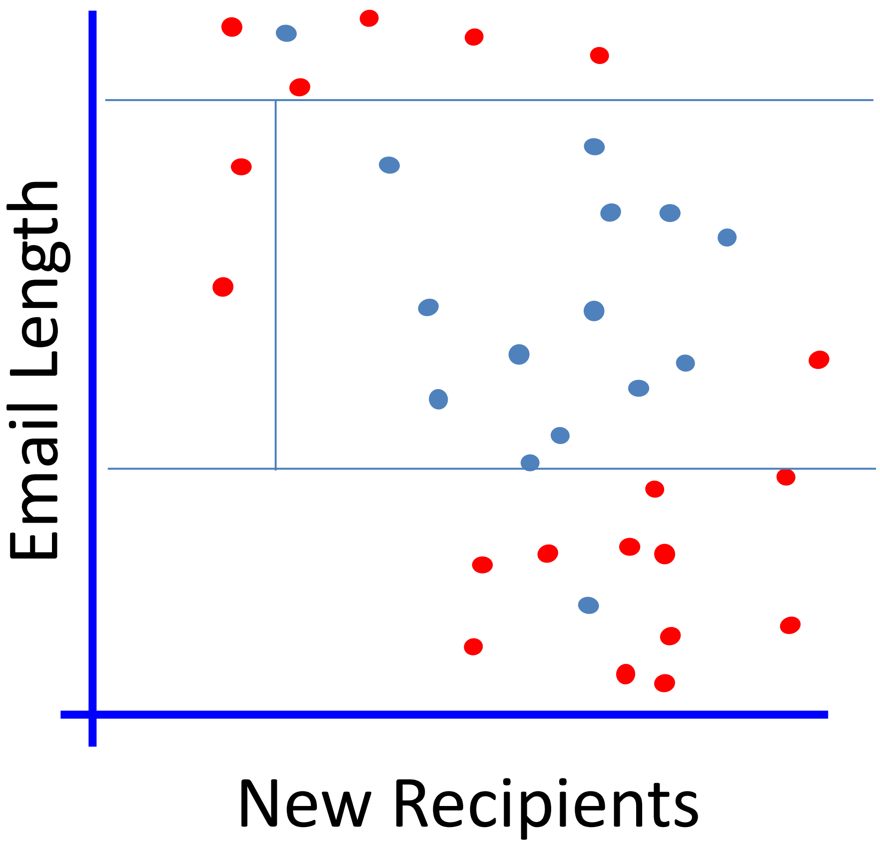


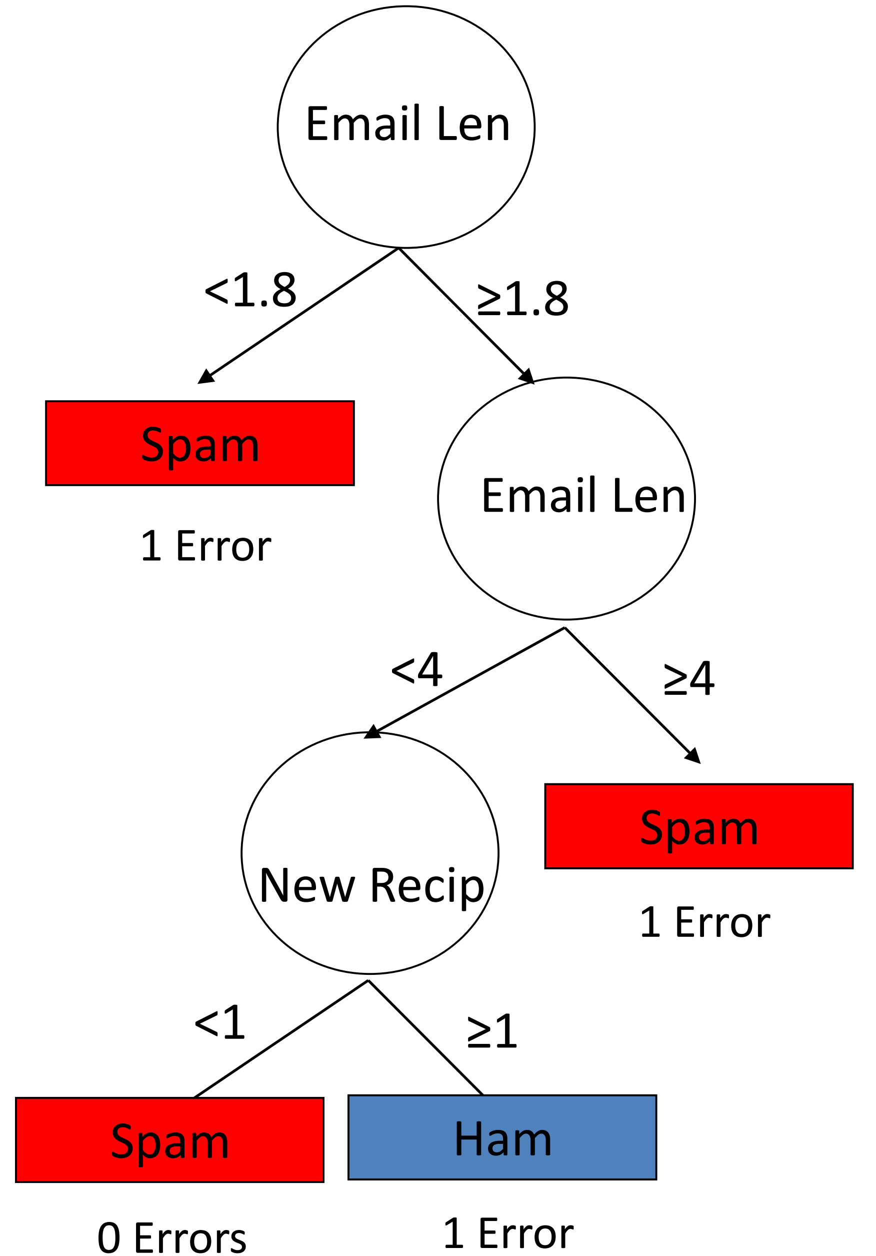
Top Down Induction of Decision Trees



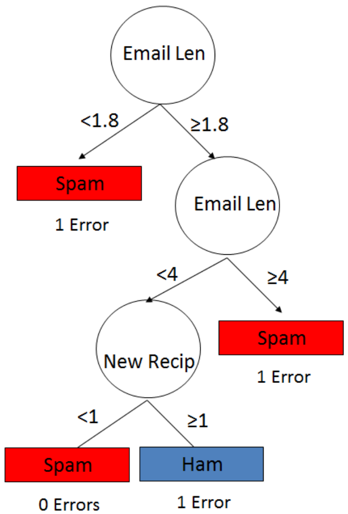


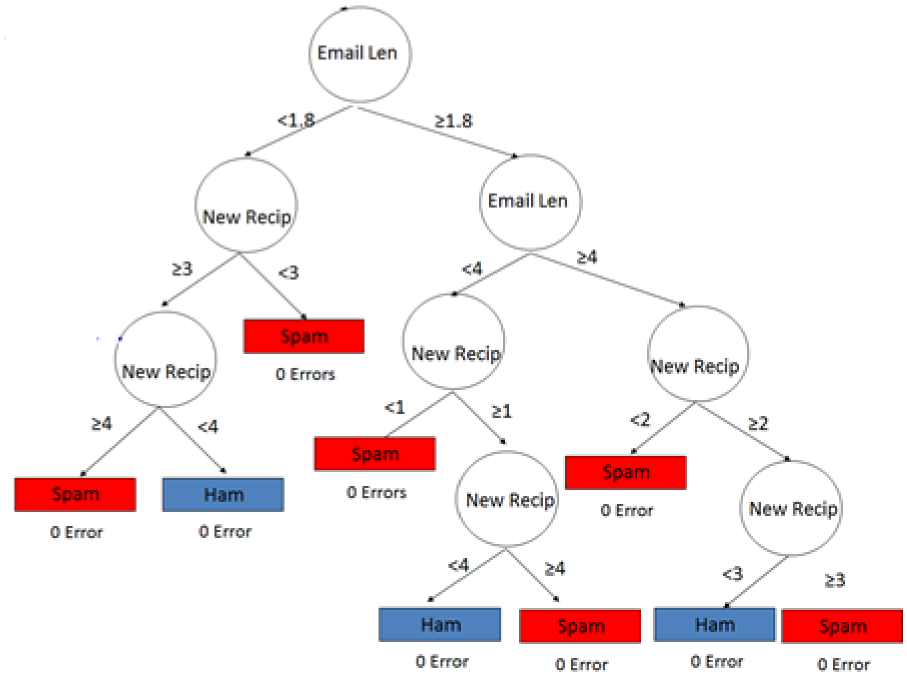
Top Down Induction of Decision Trees

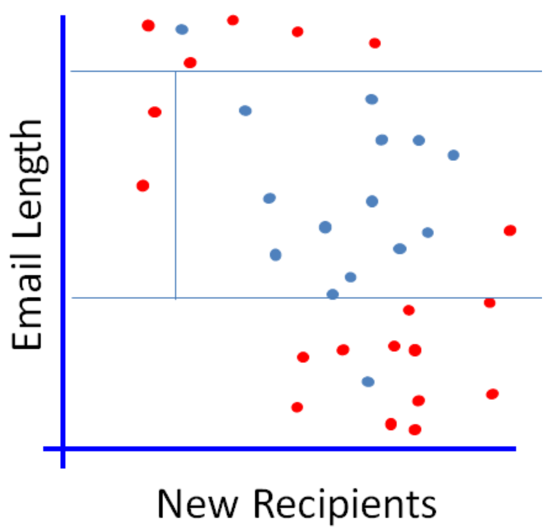


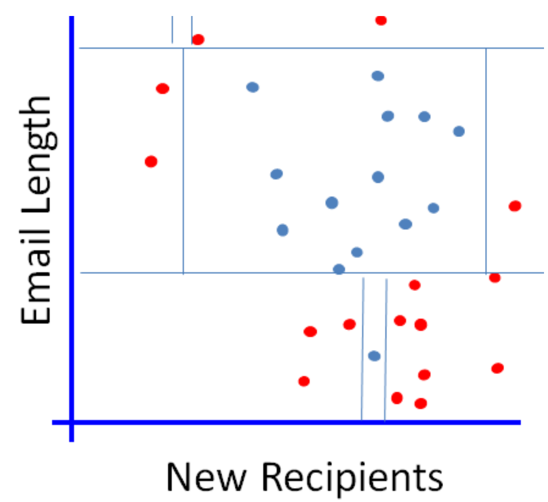


Which One?

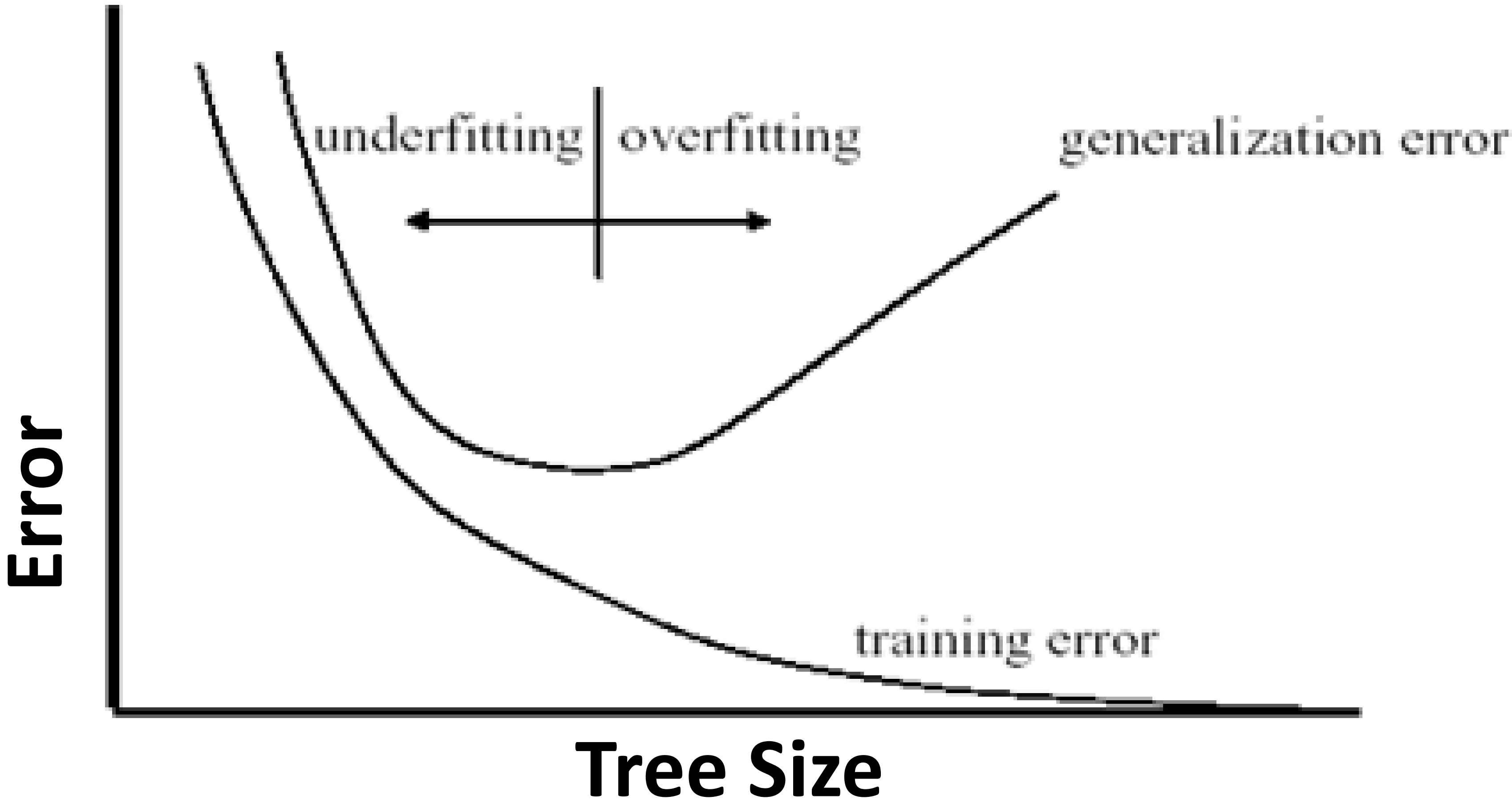

Email Len
<1.8
≥1.8
Spam
1 Error
Email Len
<4
≥4
New Recip
Spam
1 Error
<1
≥1
Spam
Ham
0 Errors
1 Error


Email Len
<1.8
≥1.8
New Recip
Email Len
≥3
<3
<4
≥4
New Recip
Spam
0 Errors
New Recip
New Recip
≥4
<4
<1
≥1
<2
≥2
Spam
Ham
Spam
Spam
0 Error
0 Error
0 Errors
0 Error
New Recip
New Recip
<4
≥4
<3
≥3
Ham
Spam
Ham
Spam
0 Error
0 Error
0 Error
0 Error


Email Length
New Recipients


Email Length
New Recipients

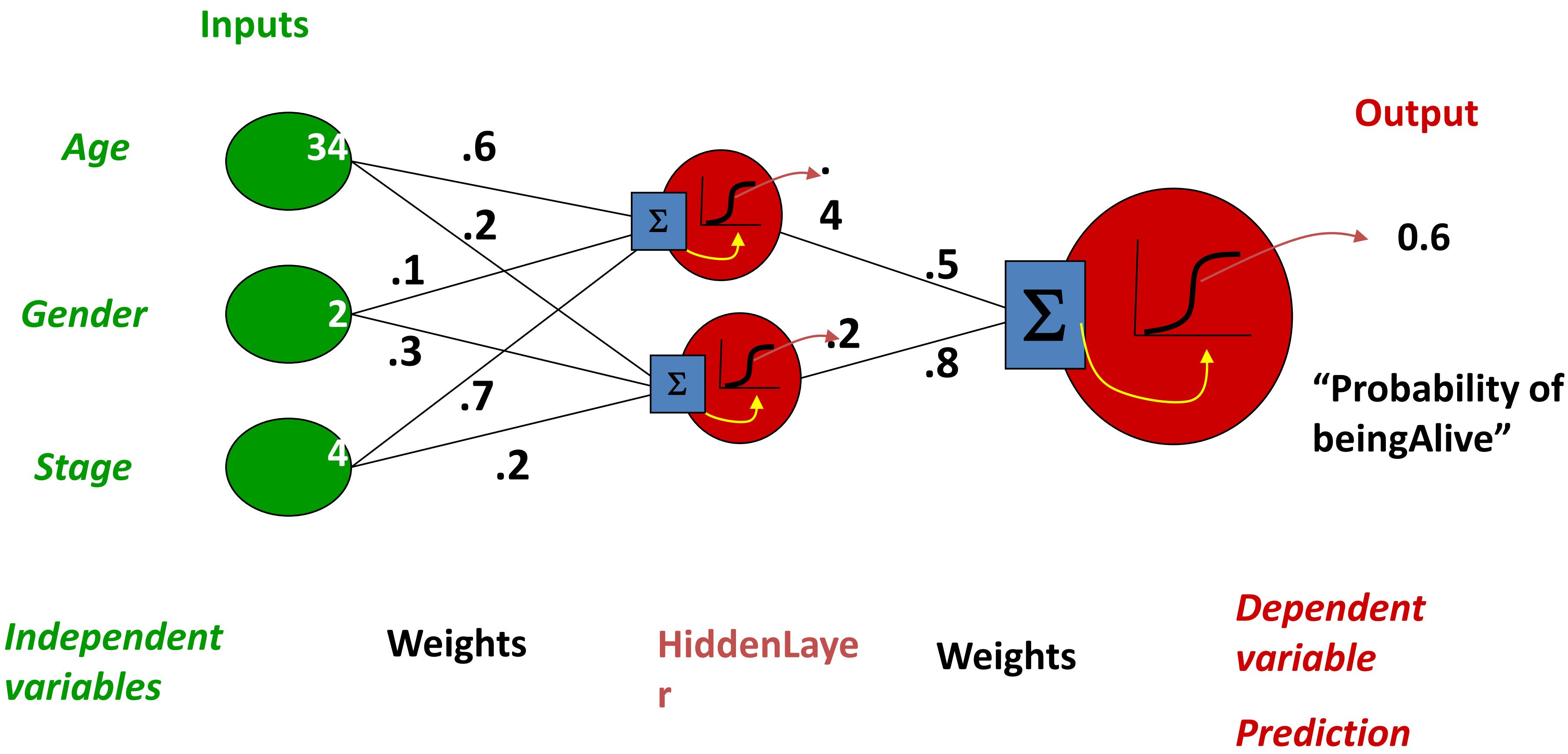
Overfitting and underfitting



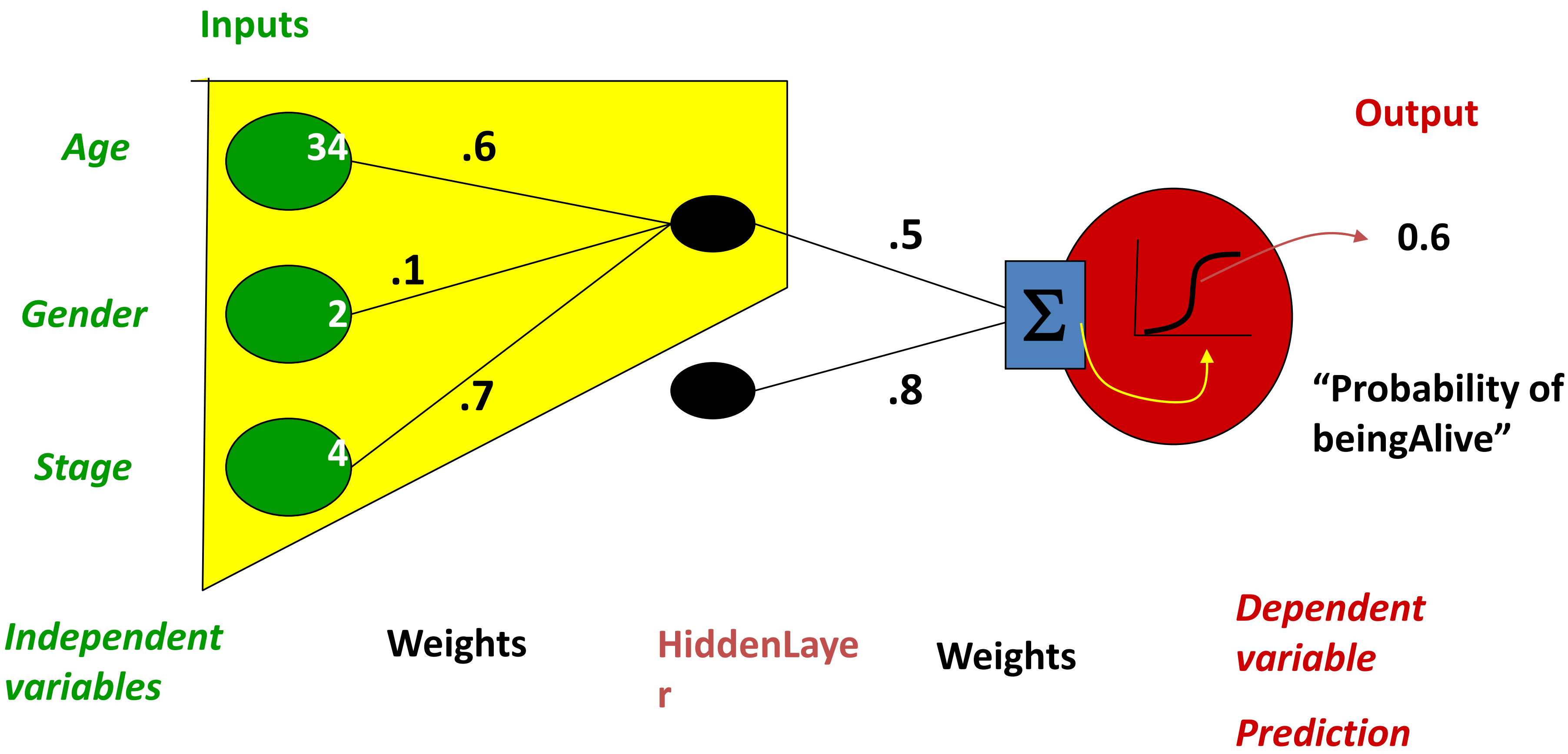
Overtraining: means that it learns the training set too well – it overfits to the training set such that it performs poorly on the test set.

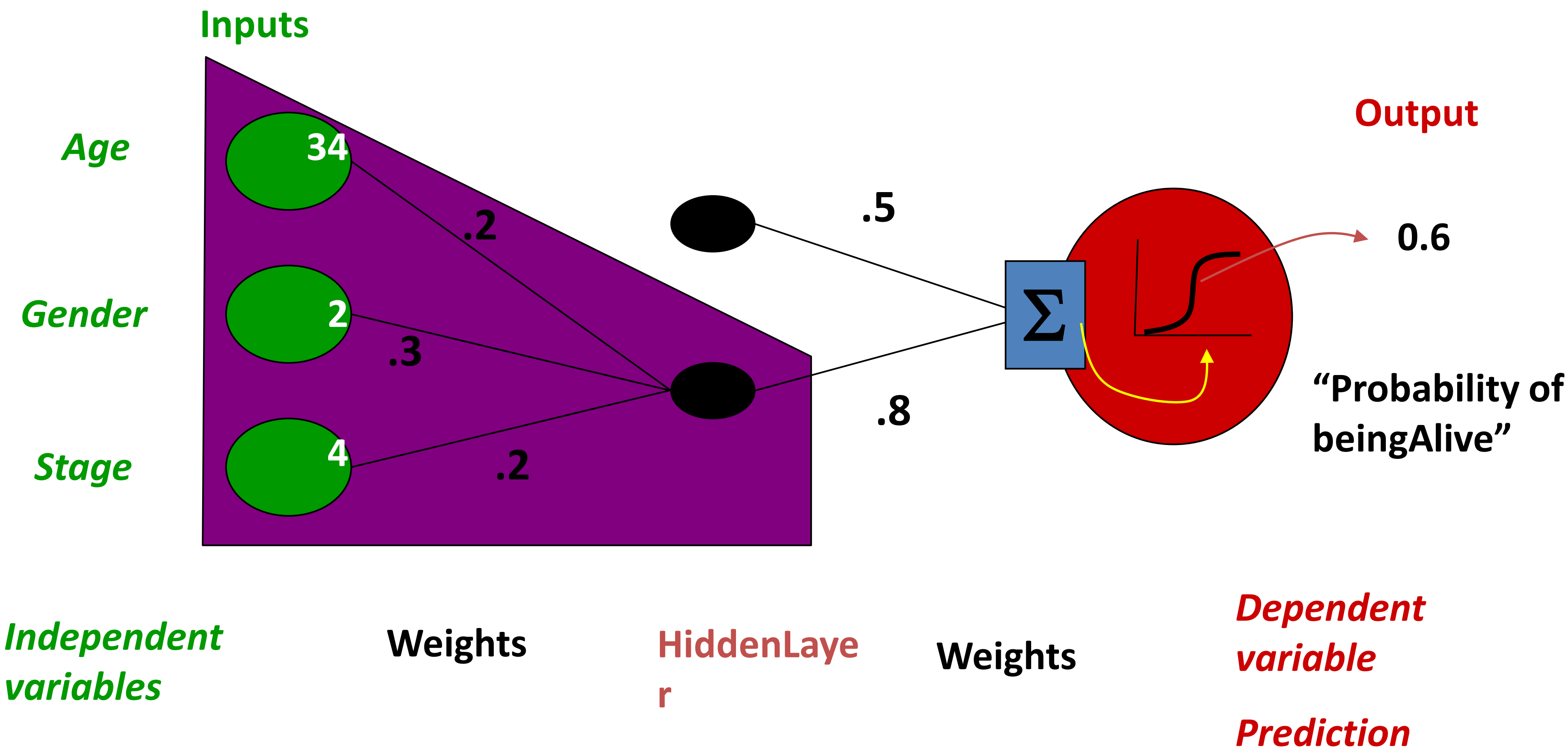
Underfitting: when model is too simple, both training and test errors are large

Neural Network Model

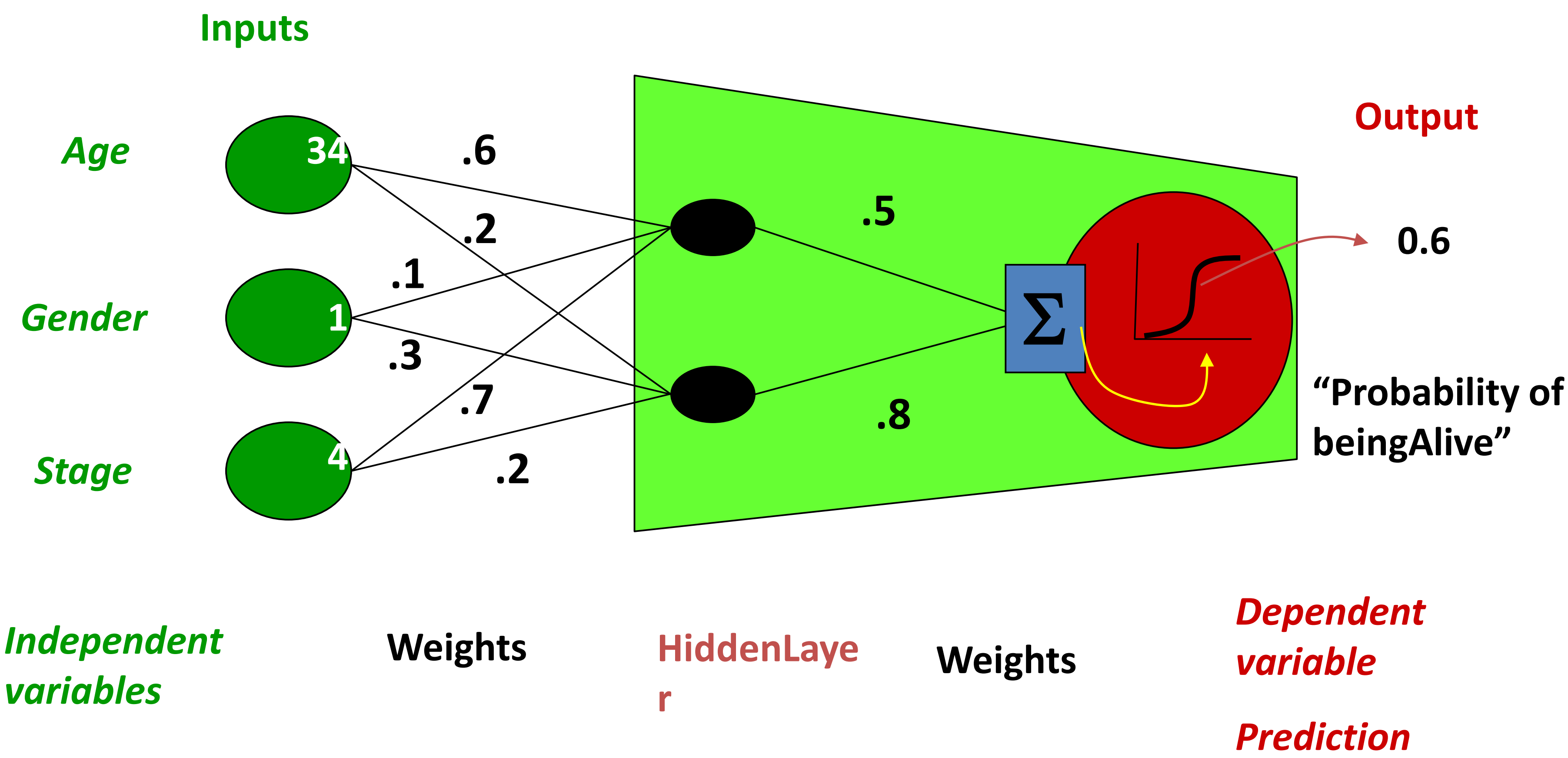


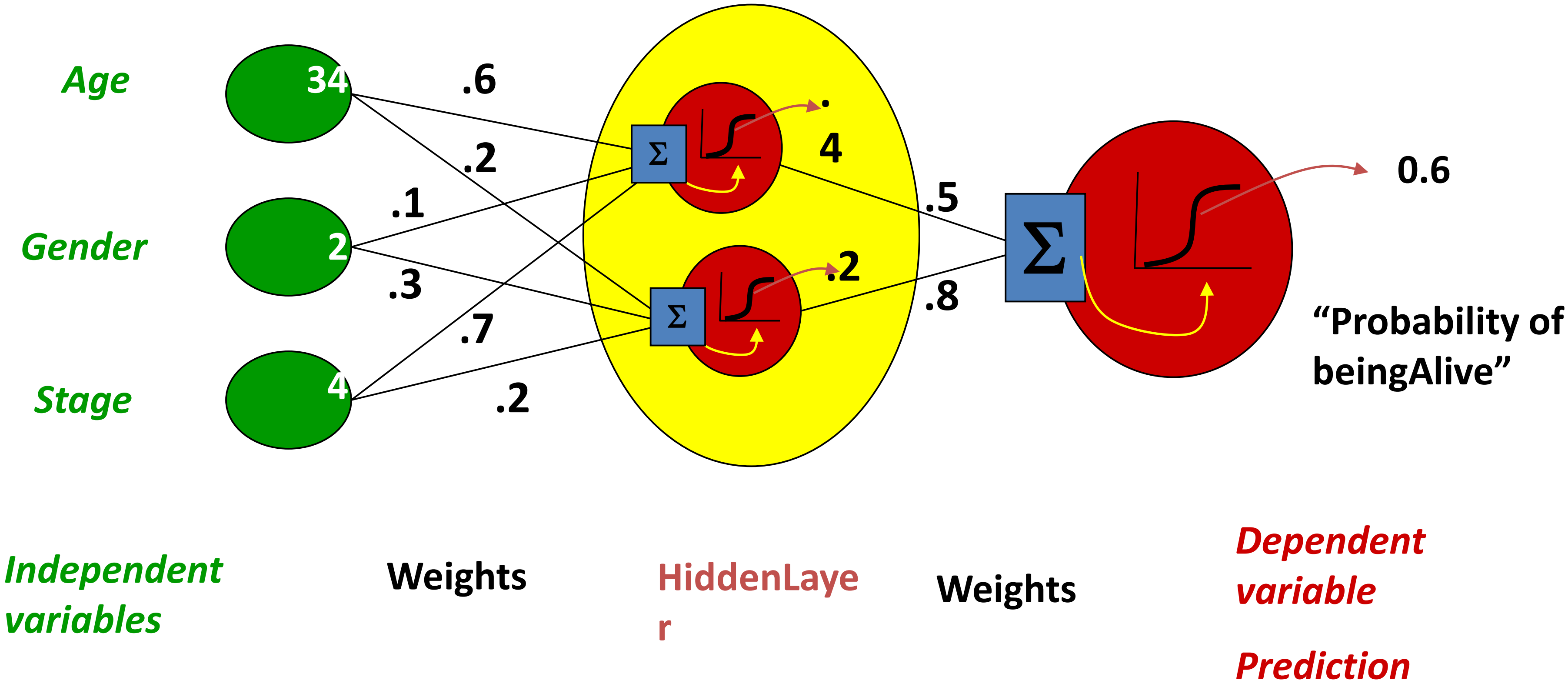
“Combined logistic models”





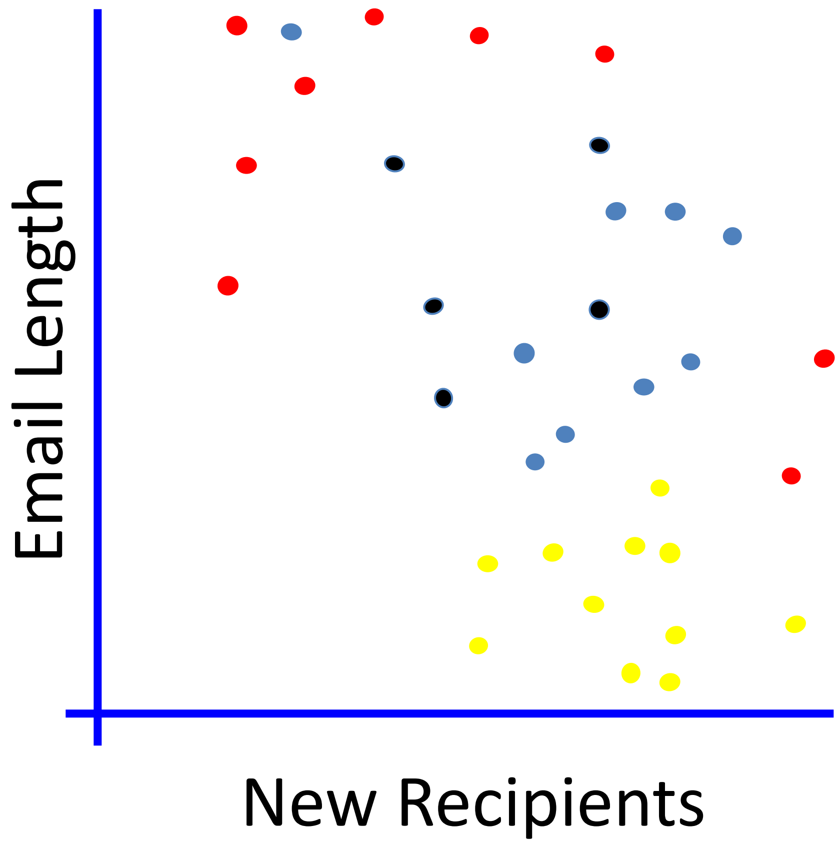
Inputs
Age
34
Gender
2
Stage
4
.2
.3
.2
.5
.8
Σ
Output
0.6
"Probability of beingAlive"
Independent variables
Weights
HiddenLayer
Weights
Dependent variable
Prediction


Inputs
Age
Gender
Stage
34
1
4
.6
.2
.1
.3
.7
.2
.5
.8
Σ
Output
0.6
“Probability of beingAlive”
Independent variables
Weights
HiddenLayer
Weights
Dependent variable
Prediction


Age
34
Gender
2
Stage
4
.6
.2
.1
.3
.7
.2
.
4
.2
.5
.8
0.6
"Probability of beingAlive"
Independent variables
Weights
HiddenLayer
Weights
Dependent variable
Prediction

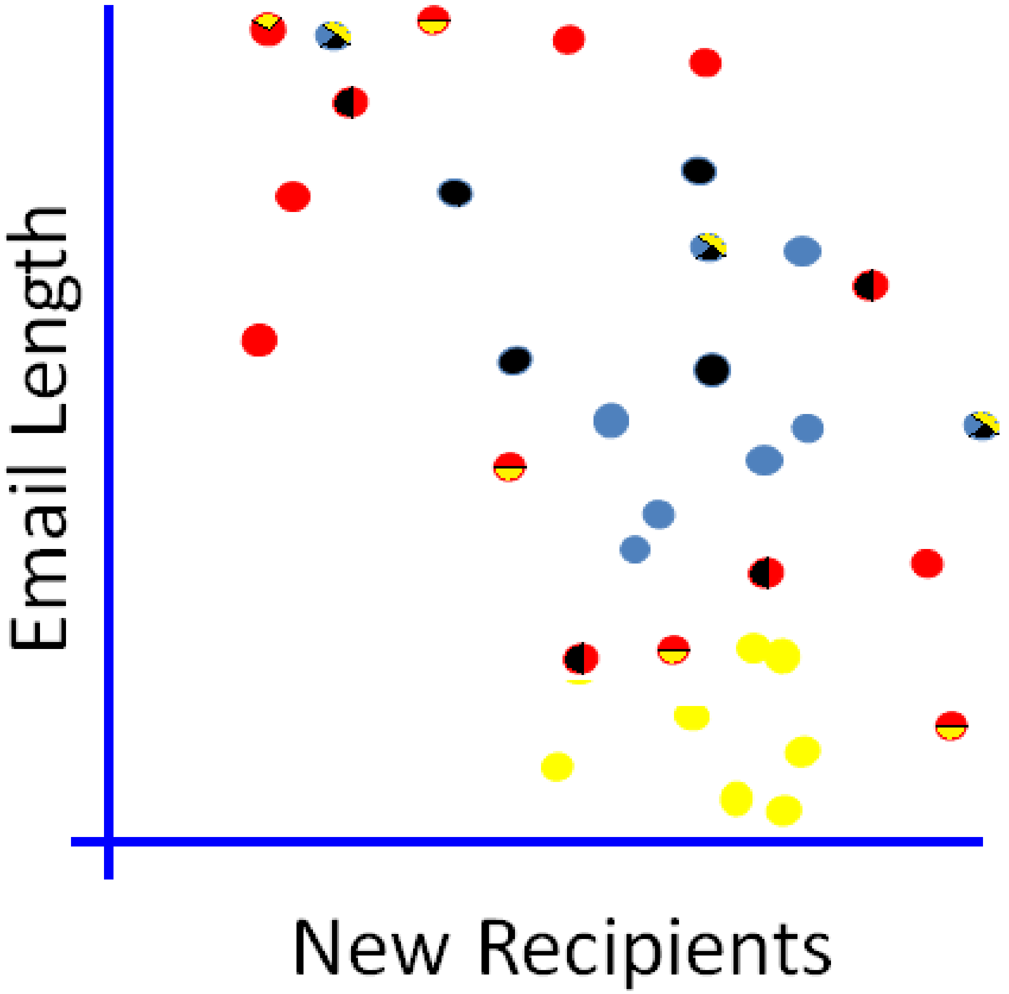
Learning Tasks

Supervised Learning - Multi Class



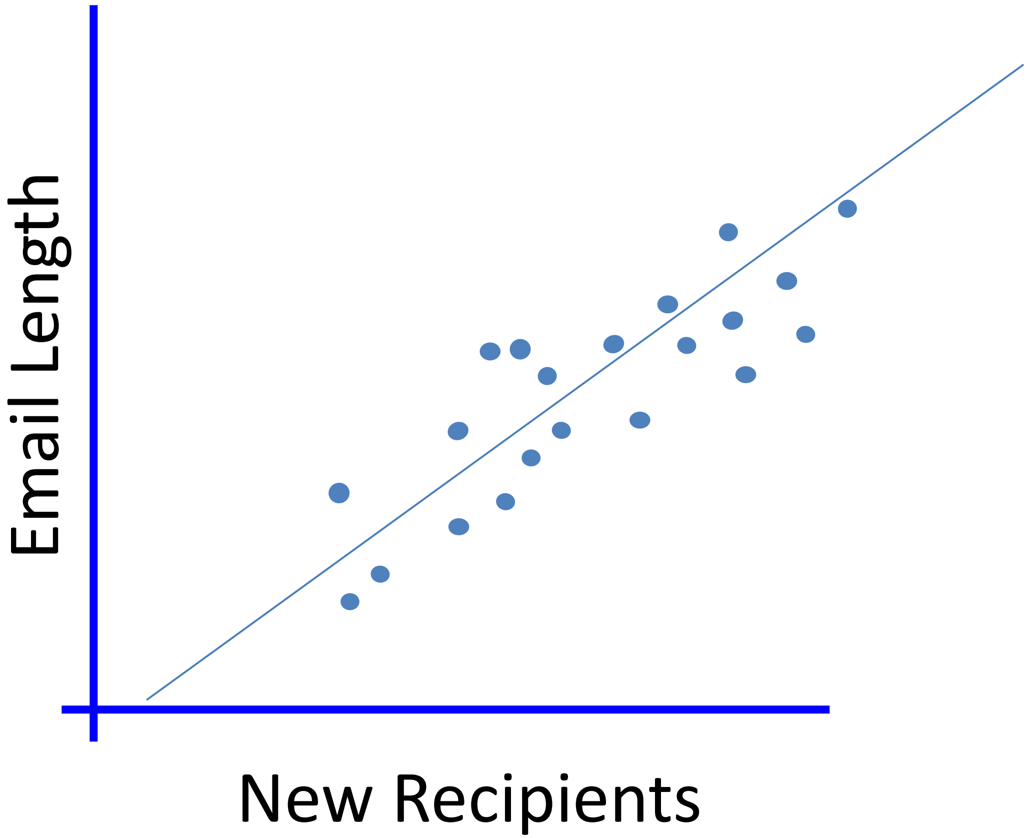
Supervised Learning - Multi Label

Multi-label learning refers to the classification problem where each example can be assigned to multiple class labels simultaneously



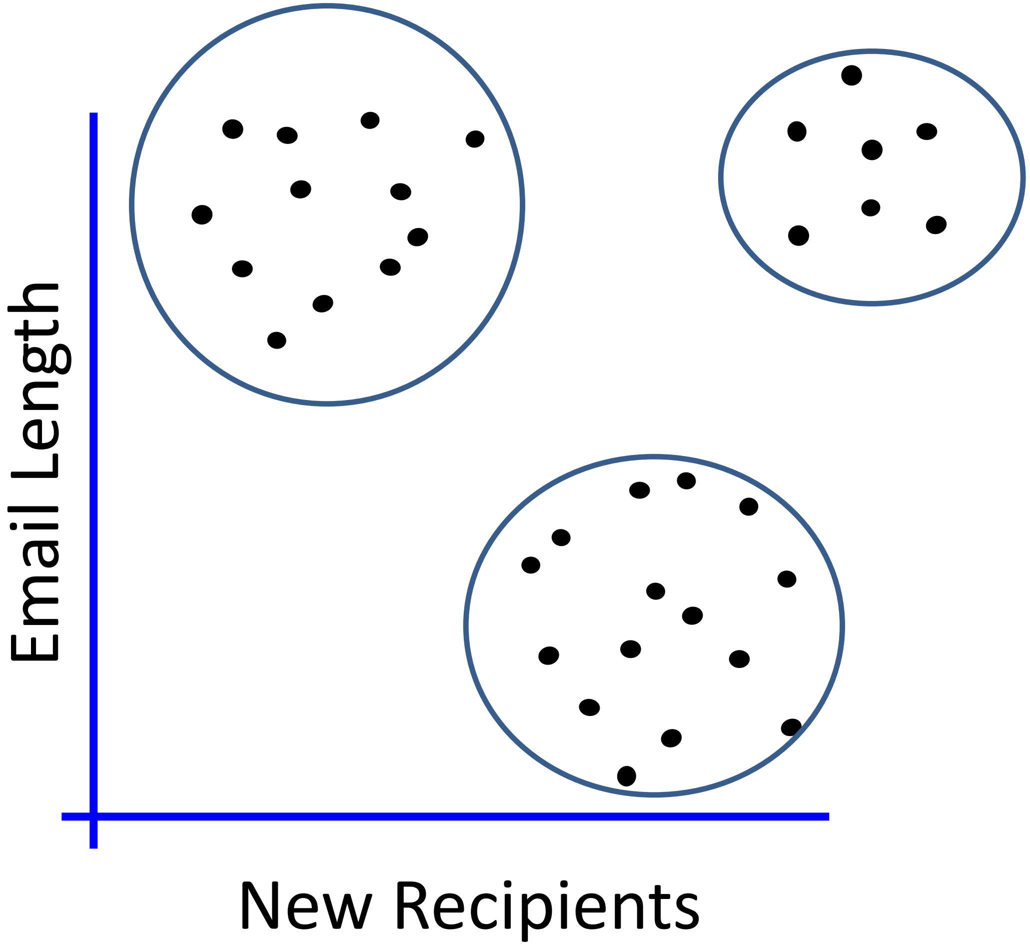
Supervised Learning - Regression

*Find a relationship between a **numeric** dependent variable and one or more independent variables*



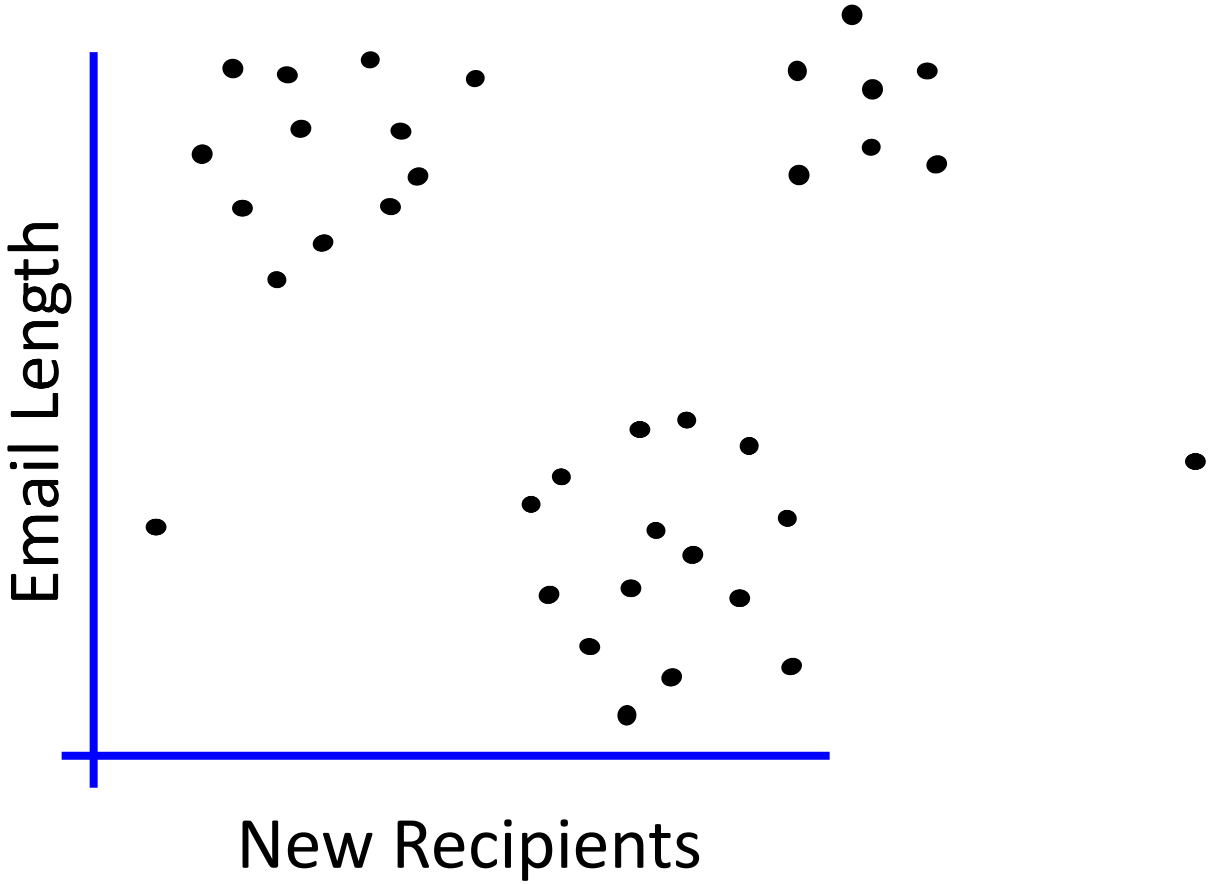
Unsupervised Learning - Clustering

Clustering is the assignment of a set of observations into subsets (called *clusters*) so that observations in the same cluster are similar in some sense

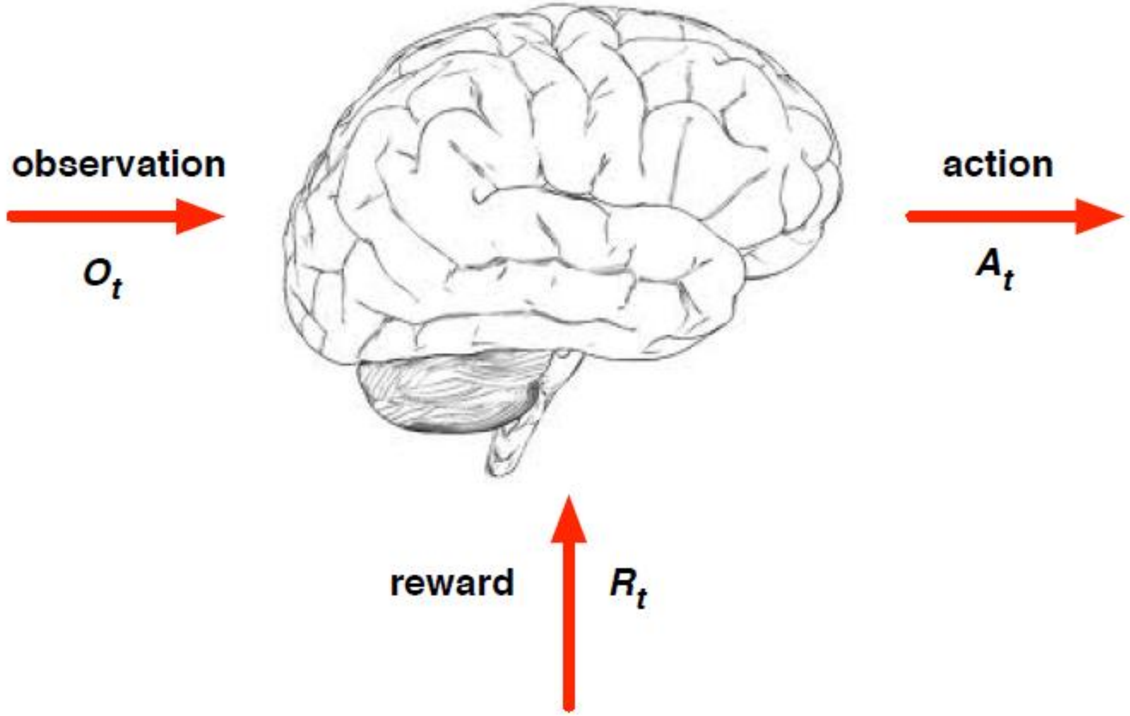


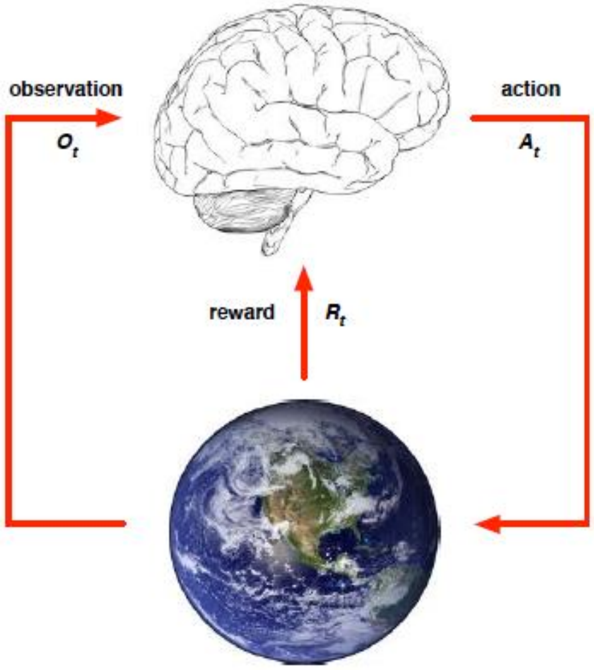
Unsupervised Learning–Anomaly Detection

Detecting patterns in a given data set that do not conform to an established normal behavior.



Reinforcement Learning



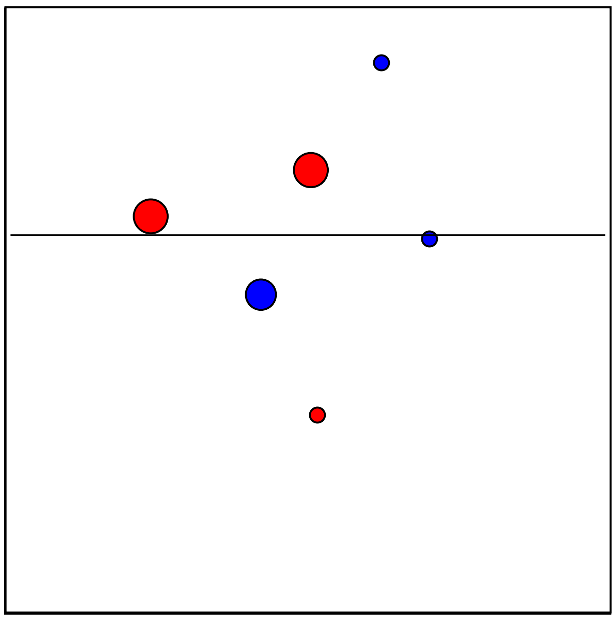


- At each step t the agent:
  - Executes action A_t
  - Receives observation O_t
  - Receives scalar reward R_t
- The environment:
  - Receives action A_t
  - Emits observation O_{t+1}
  - Emits scalar reward R_{t+1}
- t increments at env. step

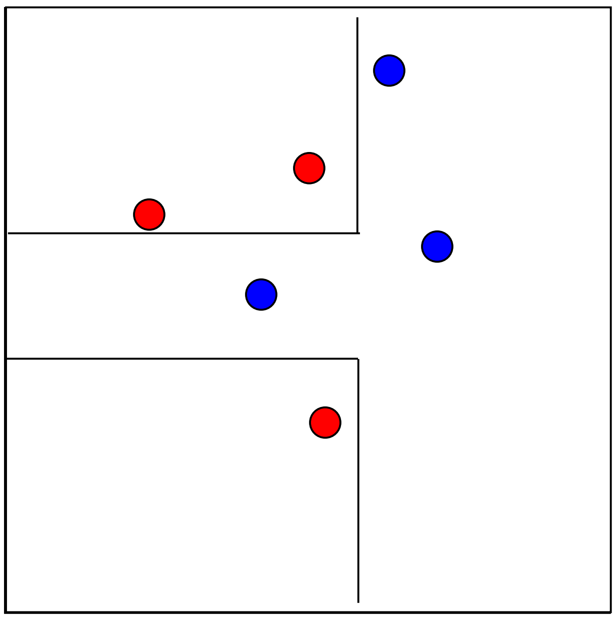
Ensemble Learning

- The idea is to use multiple models to obtain better predictive performance than could be obtained from any of the constituent models.
- Boosting involves incrementally building an ensemble by training each new model instance to emphasize the training instances that previous models misclassified.

Example of Ensemble of Weak Classifiers



Training



Combined classifier

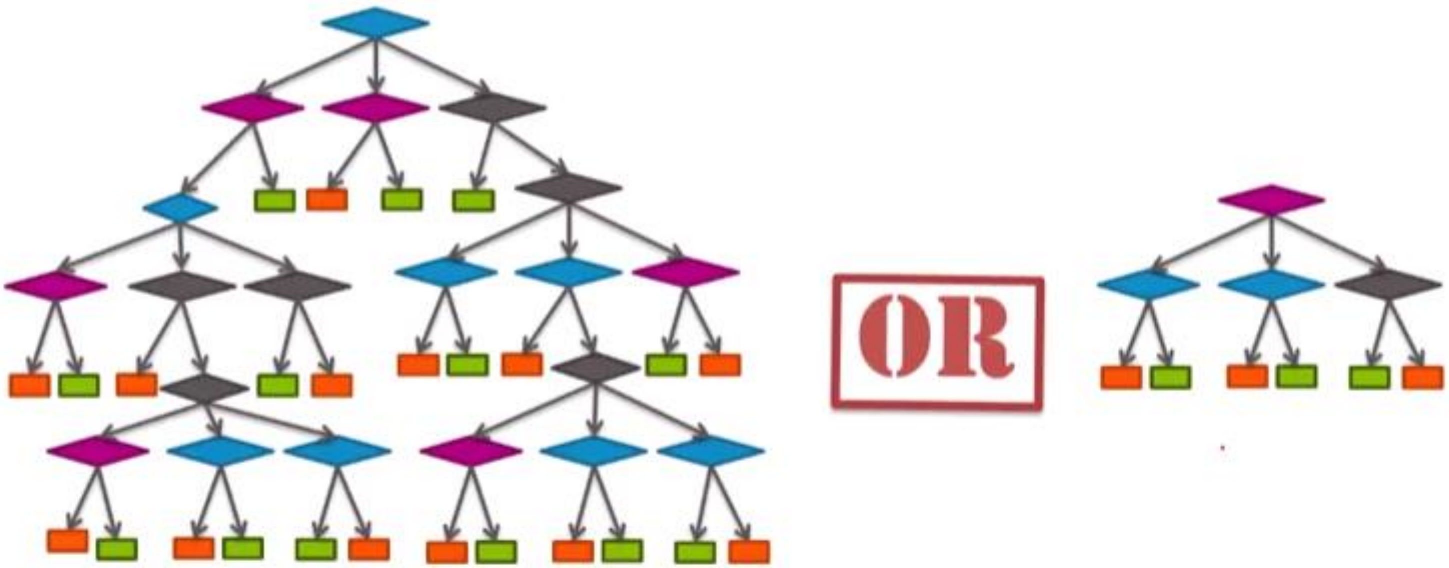
Main Principles

Occam's razor (14th-century)

- Among competing hypotheses the one with fewest assumptions should be selected.
- **The Occam Dilemma:** Unfortunately, in ML, accuracy and simplicity interpretability) are in conflict.

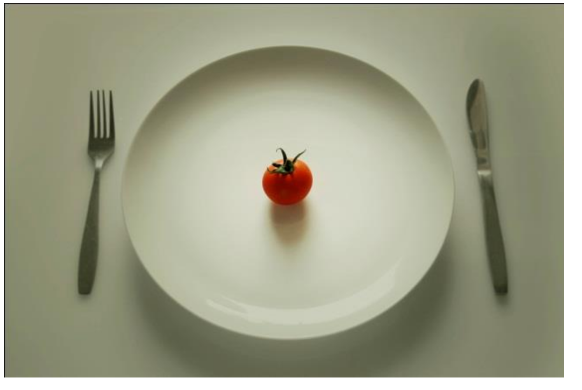
Complexity	Train error	Validation error
Simple	0.23	0.24
Moderate	0.12	0.15
Complex	0.07	0.15
Super complex	0	0.18

Simple or Complex



No Free Lunch Theorem in Machine Learning (Wolpert, 2001)

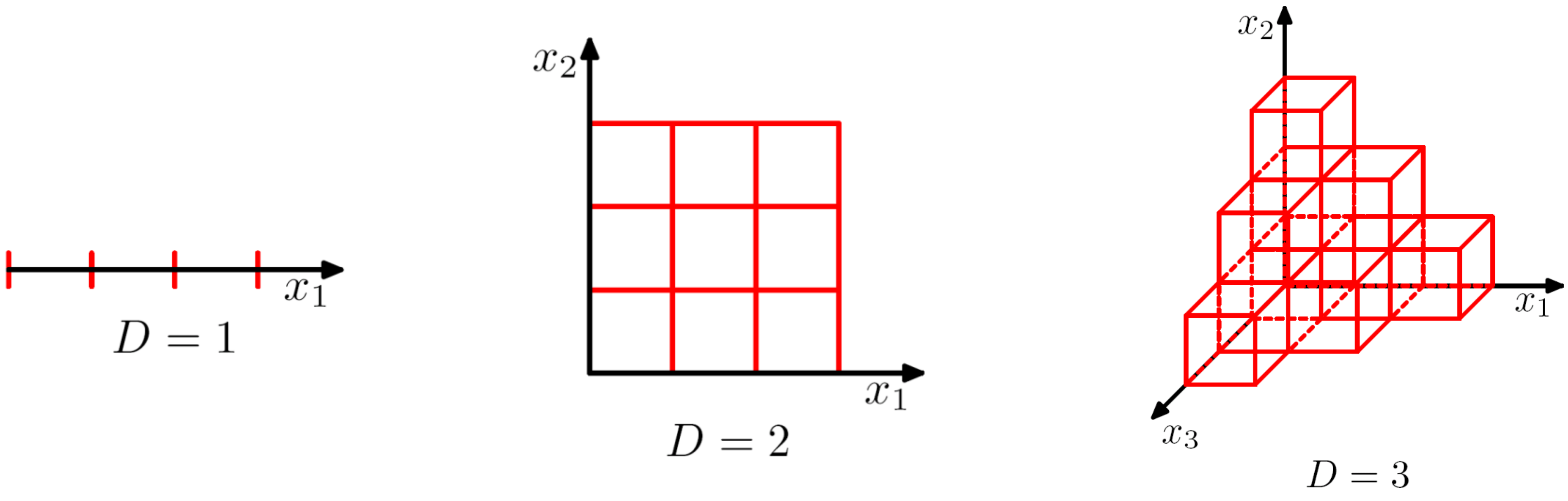
- *"For any two learning algorithms, there are just as many situations (appropriately weighted) in which algorithm one is superior to algorithm two as vice versa, according to any of the measures of "superiority"*



So why developing new algorithms?

- Practitioner are mostly concerned with choosing the most appropriate algorithm for the **problem at hand**
- This requires some a priori knowledge – data distribution, prior probabilities, complexity of the problem, the physics of the underlying phenomenon, etc.
- The *No Free Lunch* theorem tells us that – unless we have some a priori knowledge – simple classifiers (or complex ones for that matter) are not necessarily better than others. However, given some a priori information, certain classifiers may better **MATCH** the characteristics of certain type of problems.
- The main challenge of the practitioner is then, to identify the correct match between the problem and the classifier! …which is yet another reason to arm yourself with a diverse set of learner arsenal !

Less is More?
The Curse of Dimensionality
(Bellman, 1961)



Less is More?
The Curse of Dimensionality

- Learning from a high-dimensional feature space requires an enormous amount of training to ensure that there are several samples with each combination of values.
- With a fixed number of training instances, the predictive power reduces as the dimensionality increases.
- As a counter-measure, many dimensionality reduction techniques have been proposed, and it has been shown that when done properly, the properties or structures of the objects can be well preserved even in the lower dimensions.
- Nevertheless, naively applying dimensionality reduction can lead to pathological results.

While **dimensionality reduction** is an important tool in machine learning/data mining, we must always be aware that it can distort the data in misleading ways.

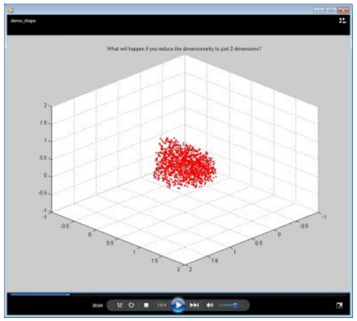
Above is a two dimensional projection of an intrinsically three dimensional world….

Original photographer unknown

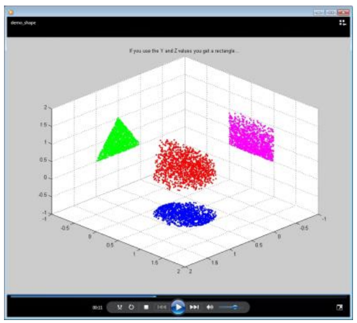
See also www.cs.gmu.edu/~jessica/DimReducDanger.htm (c) eamonn keogh

Screen dumps of a short video from www.cs.gmu.edu/~jessica/DimReducDanger.htm
I recommend you imbed the original video instead

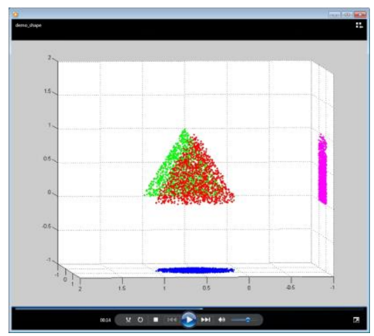
A cloud of points in 3D



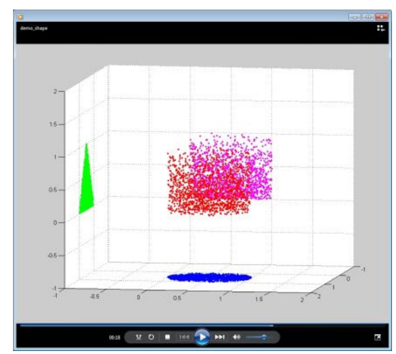
Can be projected into 2D
XY or XZ or YZ



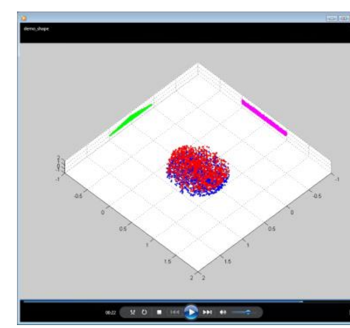
In 2D XZ we see
a triangle



In 2D YZ we see
a square



In 2D XY we see
a circle



Less is More?

- In the past the published advice was that high dimensionality is dangerous.
- But, Reducing dimensionality reduces the amount of information available for prediction.
- Today: try going in the opposite direction: Instead of reducing dimensionality, increase it by adding many functions of the predictor variables.
- The higher the dimensionality of the set of features, the more likely it is that separation occurs.

Source of Training Data

- Provided random examples outside of the learner’s control.
  - Passive Learning
  - Negative examples available or only positive? Semi-Supervised Learning
  - Imbalanced
- Good training examples selected by a “benevolent teacher.”
  - “Near miss” examples
- Learner can query an oracle about class of an unlabeled example in the environment.
  - Active Learning
- Learner can construct an arbitrary example and query an oracle for its label.
- Learner can run directly in the environment without any human guidance and obtain feedback.
  - Reinforcement Learning
- There is no existing class concept
  - A form of discovery
  - Unsupervised Learning
    - Clustering
    - Association Rules

Other Learning Tasks

- **Other Supervised Learning Settings**
  - Multi-Class Classification
  - Multi-Label Classification
  - Semi-supervised classification – make use of labeled and unlabeled data
  - One Class Classification – only instances from one label are given
- **Ranking and Preference Learning**
- **Sequence labeling**
- **Cost-sensitive Learning**
- **Online learning and Incremental Learning- Learns one instance at a time.**
- **Concept Drift**
- **Multi-Task and Transfer Learning**
- **Collective classification – When instances are dependent!**