Machine learning for algo trading

An introduction for non-mathematicians

Dr. Aly Kassam

Overview

- High level introduction to machine learning
- A machine learning bestiary
- What has all this got to do with trading?
- Some pitfalls to look out for
- What resources exist?
- What next?

ML – What is it??

*“**Machine learning** is a scientific discipline that deals with the construction and study of algorithms that can learn from data.[1] Such algorithms operate by building a model based on inputs[2]:2 and using that to make predictions or decisions, rather than following only explicitly programmed instructions.”*

Source: Wikipedia

A ML bestiary

Decision trees

Naïve Bayes

Bayesian Nets

Neural Networks

K-means clustering

Genetic algorithms

Logistic regression

K-nearest neighbours

Associated rule learning

Hidden Markov Models

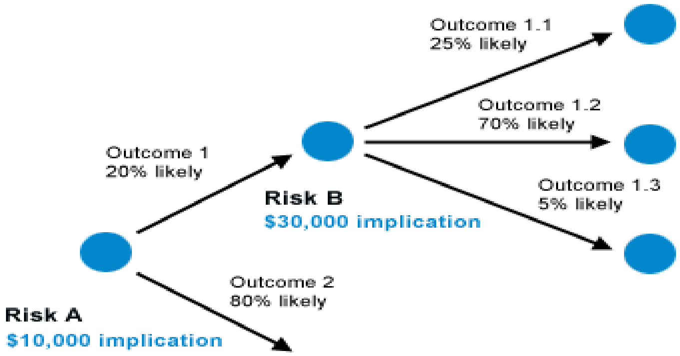
Support Vector Machines

Principal component analysis

A ML bestiary

Supervised learning:

- K-nearest neighbours
- Decision trees
- Logistic regression
- Support Vector Machines
- Bayesian networks
- Neural networks



Unsupervised learning:

- K-means clustering
- Hidden Markov Models
- Principal component analysis
- Associate rule learning

Supervised vs unsupervised

Supervised learning

- **Start with a labelled "training" data set**
- **Used for producing predictive models**
- **Examples are:**
  - Classification
  - Regression

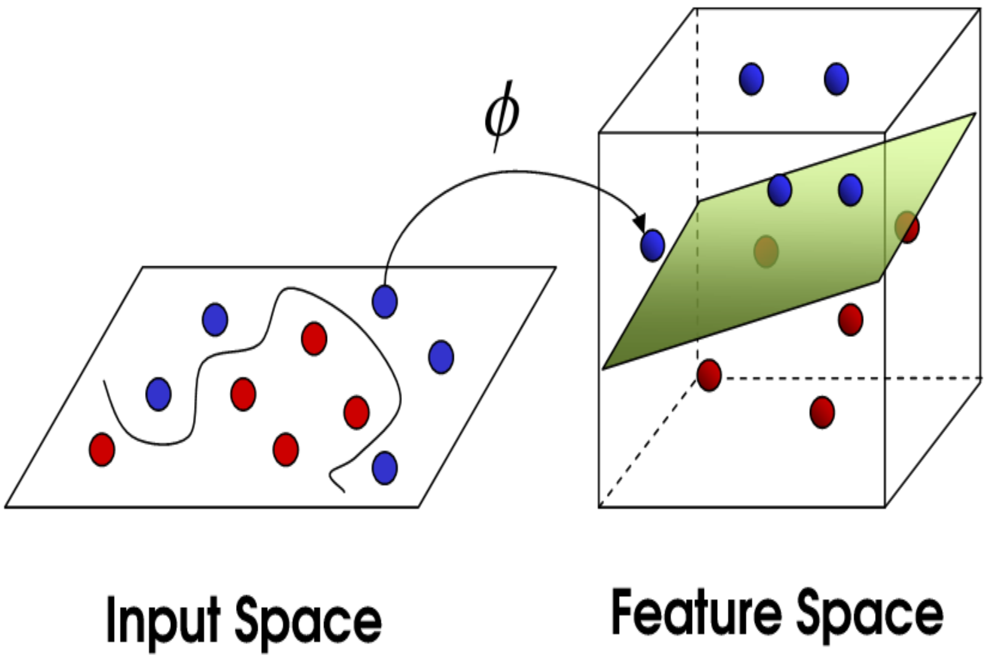
Unsupervised learning

- **No labelling on the data**
- **Used for producing descriptive models**
- **Examples are:**
  - Clustering
  - Association learning

Supervised learning

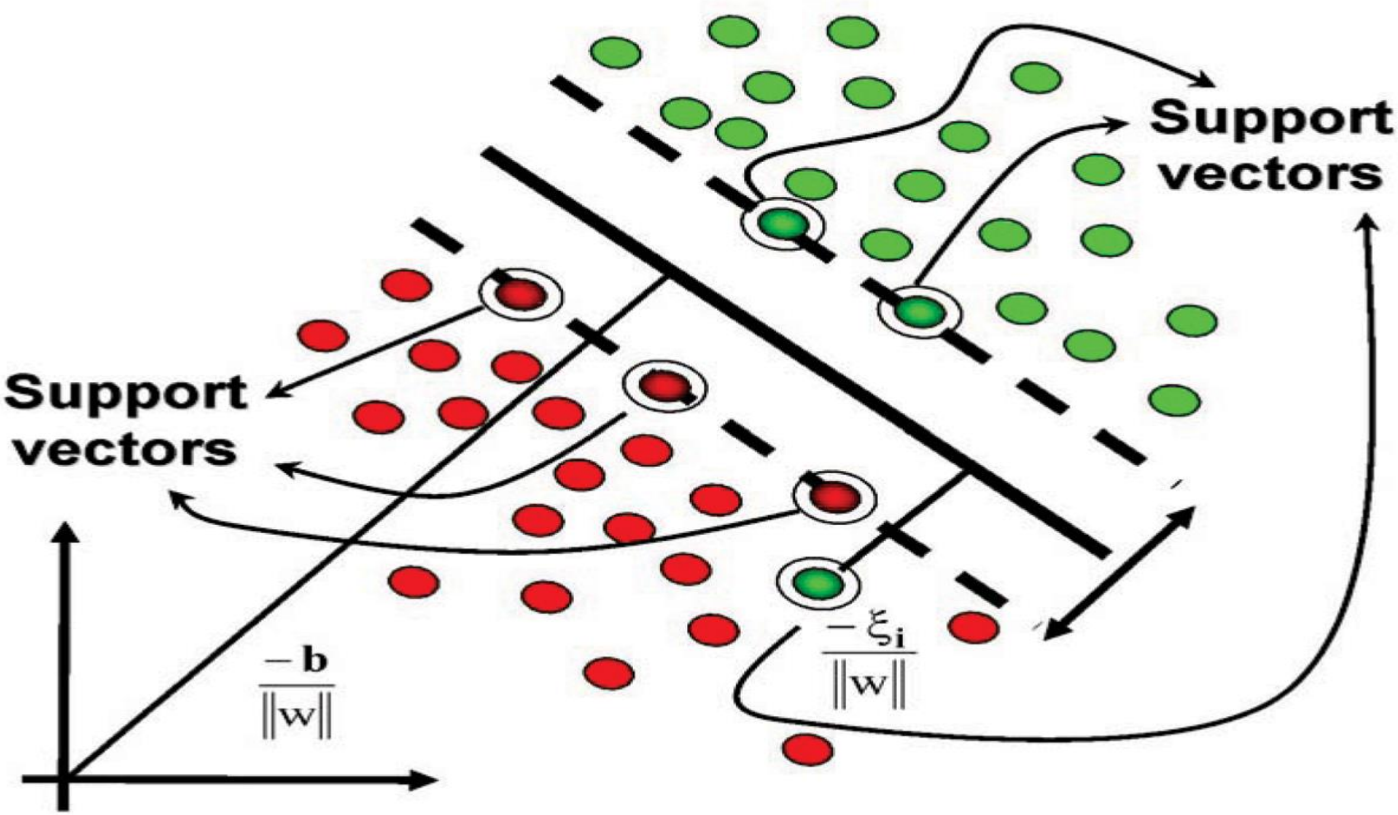
Support Vector Machines

- **Used to separate data into different classes**
- **It's an example of a *linear separator***
- **Works in multiple dimensions**
- **Kernels can be used to add nonlinearity**



Source: www.stackoverflow.com

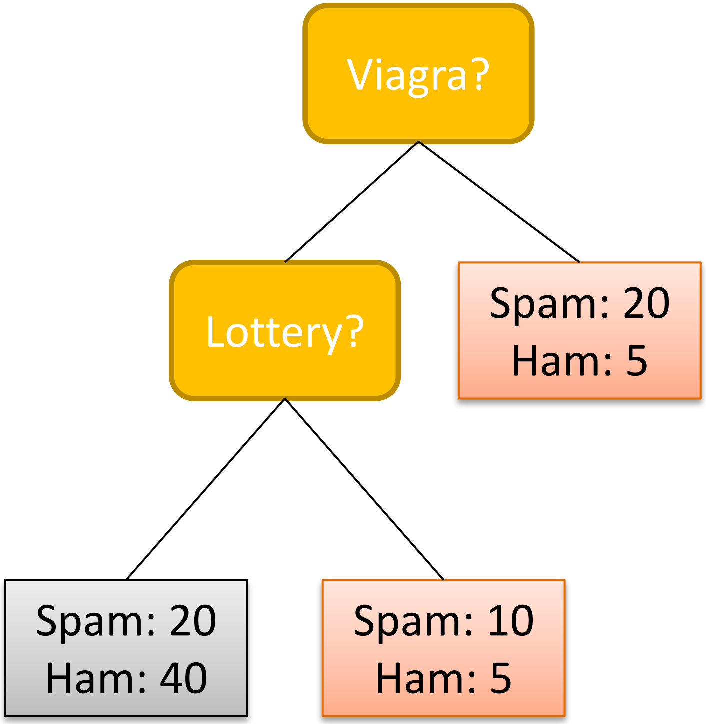
What is the support vector?



- **The idea behind SVM is to find the hyperplane that results in the greatest margin between the support vectors.**

Decision trees

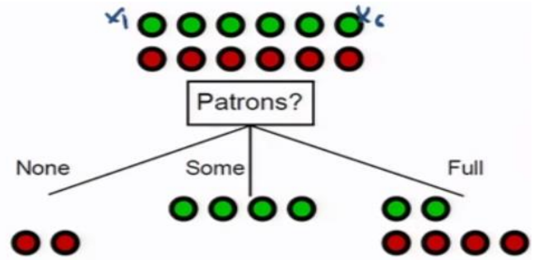
- **Classify data based on a sequence of questions**
- **Classification trees**
  - Separate data into distinct classes
- **Regression trees**
  - Make real number predictions
- **Usually trees are combined into <u>ensemble models</u>**



Source: "Machine Learning", Peter Flach, CUP

Decision tree - Data

From a spreadsheet to a decision node



Examples described by attribute values (Boolean, discrete, continuous, etc.)
E.g., situations where I will/won't wait for a table:

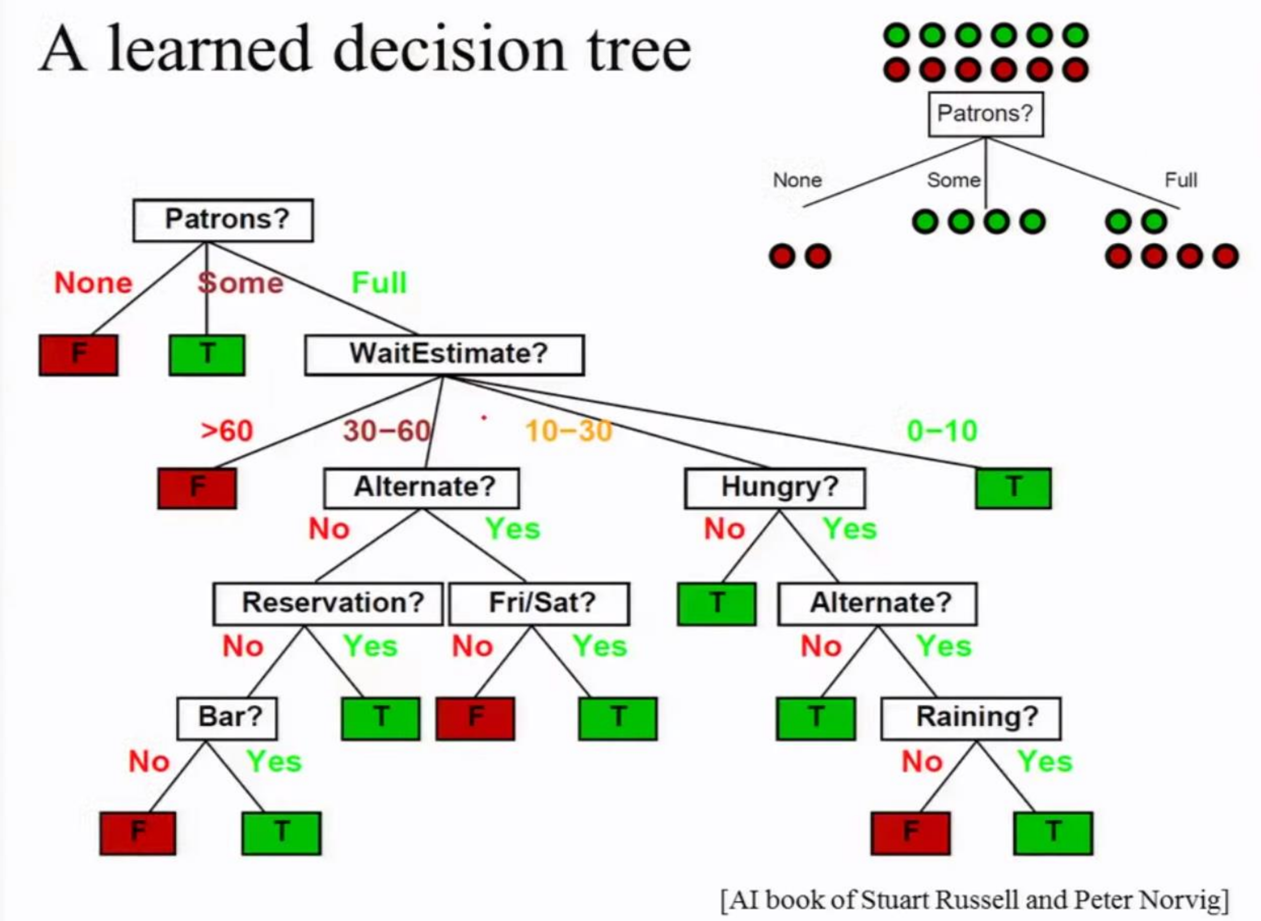
Example	Attributes										Target
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*WillWait*
X_1	T	F	F	T	Some	$$$	F	T	French	0–10	T
X_2	T	F	F	T	Full	$	F	F	Thai	30–60	F
X_3	F	T	F	F	Some	$	F	F	Burger	0–10	T
X_4	T	F	T	T	Full	$	F	F	Thai	10–30	T
X_5	T	F	T	F	Full	$$$	F	T	French	>60	F
X_6	F	T	F	T	Some	$$	T	T	Italian	0–10	T
X_7	F	T	F	F	None	$	T	F	Burger	0–10	F
X_8	F	F	F	T	Some	$$	T	T	Thai	0–10	T
X_9	F	T	T	F	Full	$	T	F	Burger	>60	F
X_{10}	T	T	T	T	Full	$$$	F	T	Italian	10–30	F
X_{11}	F	F	F	F	None	$	F	F	Thai	0–10	F
X_{12}	T	T	T	T	Full	$	F	F	Burger	30–60	T

Classification of examples is positive (T) or negative (F)

[AI book of Stuart Russell and Peter Norvig]

Source: "Machine learning – Decision trees" by Nando de Freitas, on YouTube

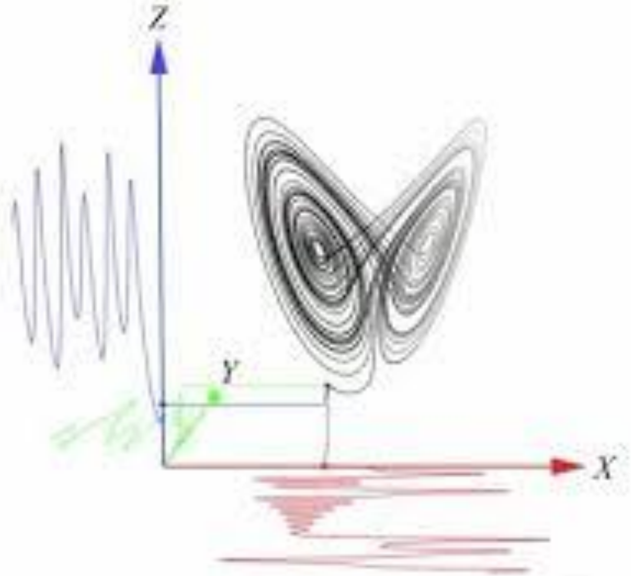
Decision trees – model



Source: "Machine learning – Decision trees" by Nando de Freitas, on YouTube

K-nearest neighbours

- **"Similar" historical points forecast likely future behaviour**

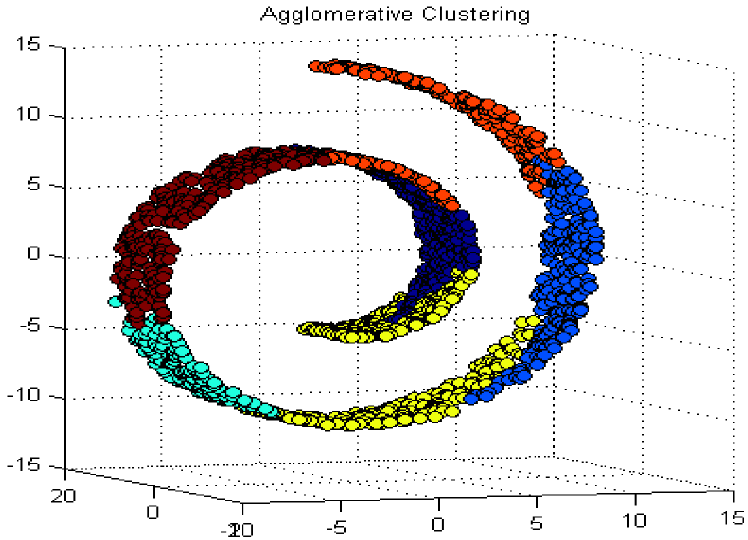


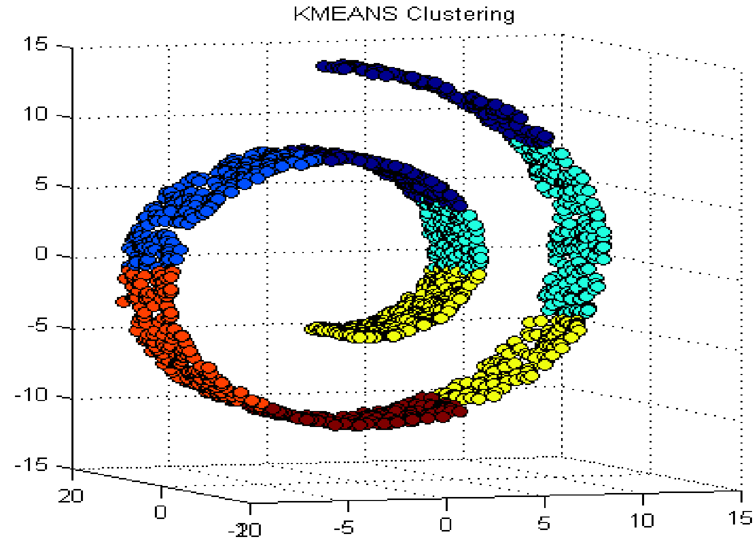
- **Can work on scalar values (find the last k similar values)**
- **Can also work with vectors**
- **Defining a pattern as a vector, forms the basis of pattern recognition**
- **See:**
  - "Machine Learning and Pattern Recognition for Algorithmic Forex and Stock Trading" (all 19 videos!) on YouTube for an example of this…

Unsupervised learning

Unsupervised learning

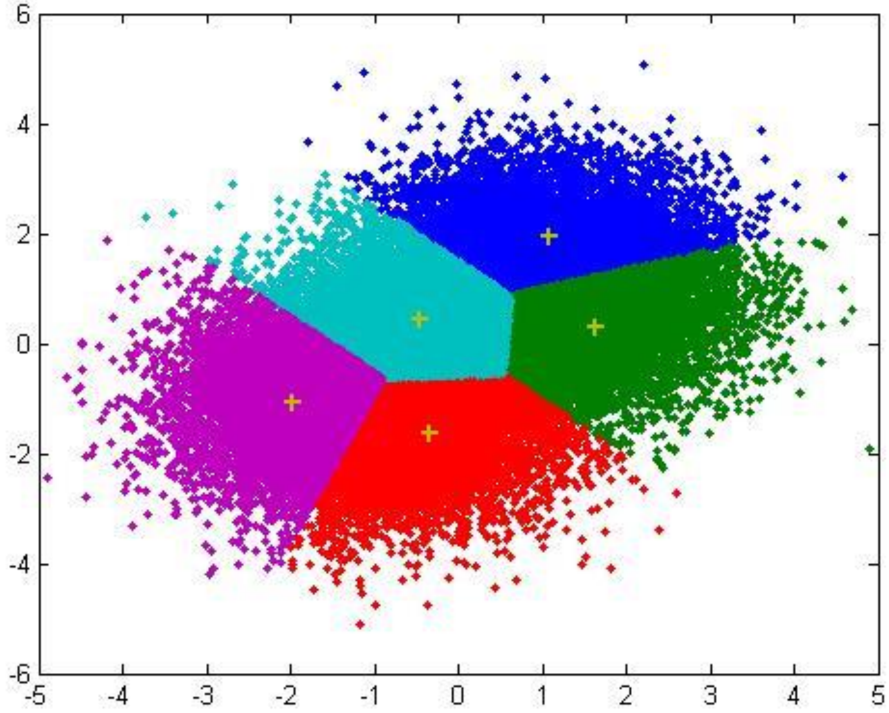
- **As mentioned before, unsupervised learning is more concerned with <u>descriptive models</u>, e.g.**
  - Clustering (hierarchical or k-means)
  - Association rule learning ("if this, then that")
  - Dimensionality reduction (e.g. PCA)





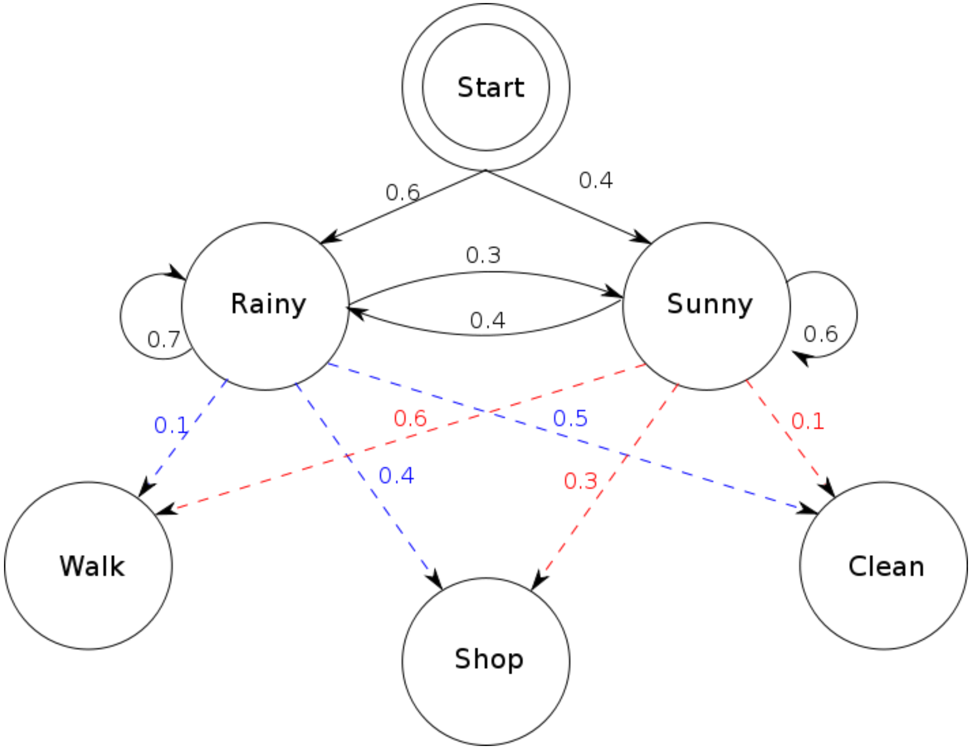
K-means clustering

- **Divide a universe of data into a collection of <u>similar sets</u>**
- **Find relationships in the data and group similar items together**



Hidden Markov models

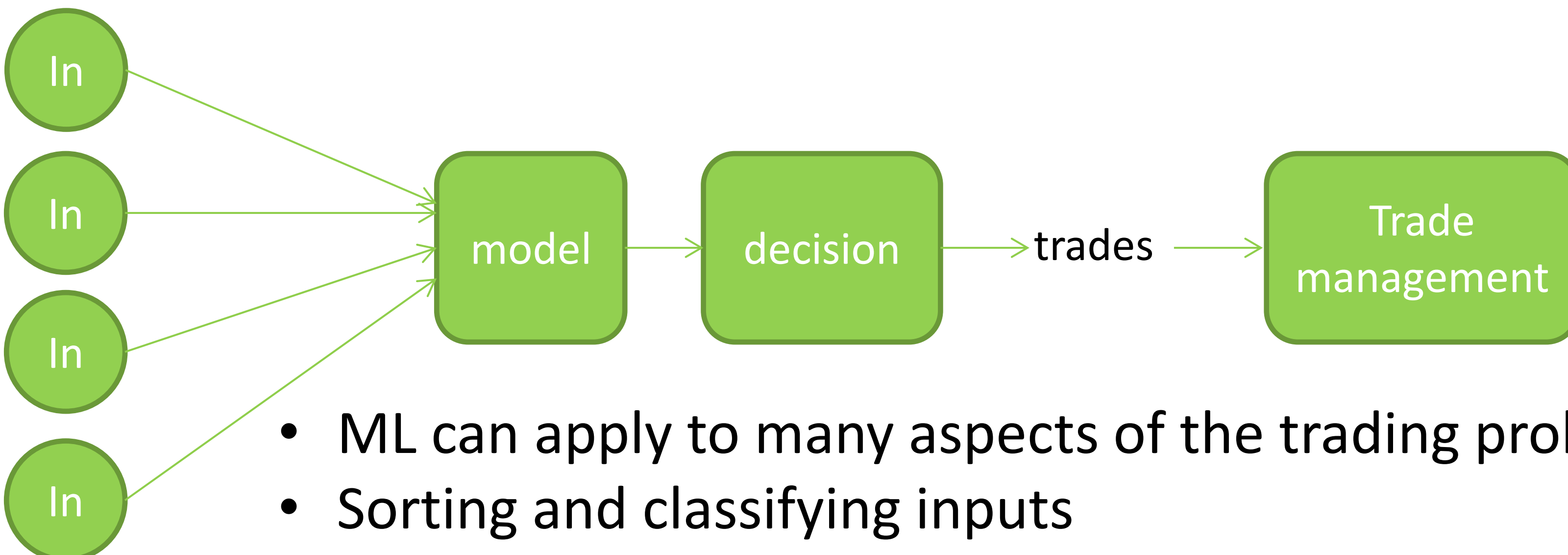
- A system can exist in a number of states
- Each state can produce a number of measurable outcomes based on a probability distribution
- Transitions between states can occur with probabilities defined in a transition matrix
- Transitions to new states depend only on the current state (hence a Markov process)



Source: Wikipedia HMM

What has this to do with trading?

- **Questions:**
  - What category of problem is trading?
  - What are good inputs? What are the outputs?



inputs

- ML can apply to many aspects of the trading problem
- Sorting and classifying inputs
- Making predictions based on decisions
- Estimating probabilities of movements or outcomes

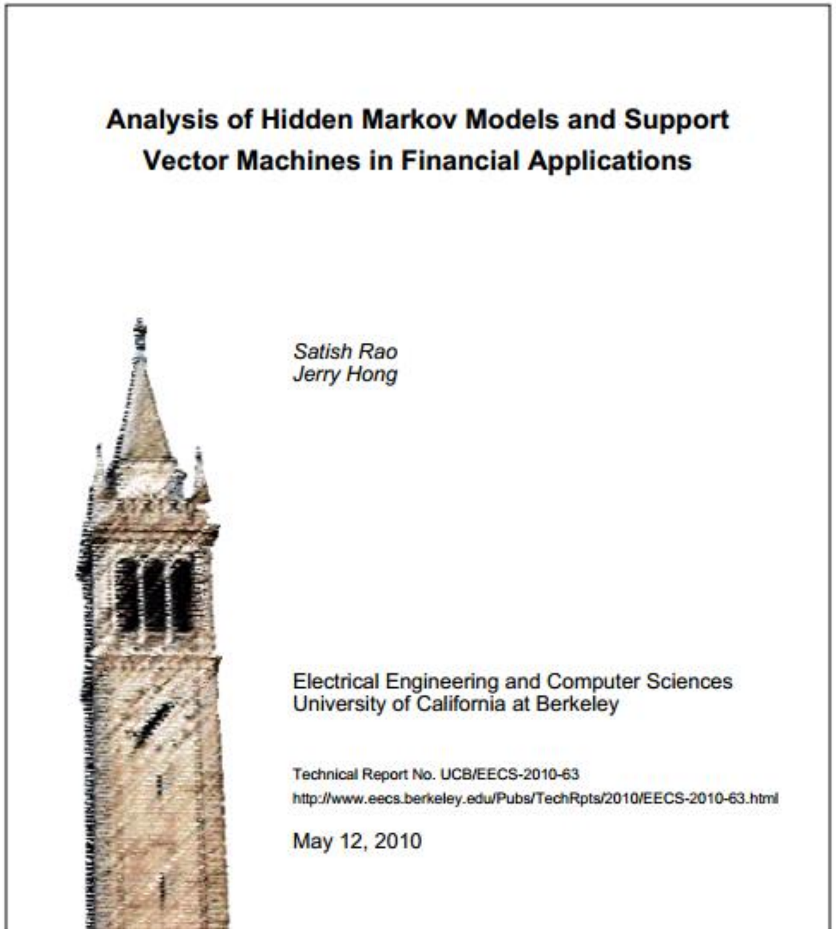
Feature selection

- **Features are important both for**
  - Input data
  - “Predicted” response (outputs)
- **Too many features → overfitting… Be careful!**

- **Input features**
  - Technical indicators
  - Changes in Prices, Volumes, Ratios
  - External series
  - News feeds
  - Time of day
  - …?

- **Output features**
  - Discrete moves (up/down/flat etc.)
  - True/false
  - Probabilities
  - …?

Example 1 – Rao and Hong 2010



Analysis of Hidden Markov Models and Support Vector Machines in Financial Applications

Satish Rao
Jerry Hong

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2010-63
http://www.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-63.html

May 12, 2010

- **Rao and Hong try to predict future prices of 10 stocks and 1 index**
- **They used:**
  - K-means clustering
  - Hidden Markov model
  - Support vector machine
- **Inputs were:**
  - EMA7, EMA50, EMA200
  - MACD, RSI, ADX,
  - High, Low, Close, Close>EMA200, lagged profits

Example 1 – Rao and Hong 2010

- **Methodology 1 – unsupervised (HMM)**
  - (1) use the K-means to identify 5 hidden states (clusters)

    [big price up, small price up, no change, small price down, big price down]
  - (2) Use HMM and daily lag profits to determine:

    “What is the probability of seeing a big price drop tomorrow given today’s state and observations.”
- **Methodology 2 – supervised (SVM)**
  - Classify each training day as a buy/sell signal, and use the 10 inputs described above to train the SVM
  - Use a RBF Kernel to produce nonlinear decision boundary
  - Experimented with adding a new input – # of news stories

Example 1 – Rao and Hong 2010

Results – HMM

ticker	accuracy prediction
pot	0.5172
aapl	0.5172
gs	0.5517
mos	0.5517
ibm	0.5172
msft	0.7586
gg	0.5862
bac	0.5862
goog	0.5517
c	0.5172
sp_500	0.5172

Table 1: Results from HMM using K-means learner

(Trained from 01/2004-09/2009 / Tested from 09/2009 – 11/2009)

Results – SVM

ticker	No News	News
pot	0.5667	0.7
aapl	0.5333	0.6667
gs	0.6	0.5333
mos	0.6	0.6
ibm	0.5667	0.5333
msft	0.8	0.6333
gg	0.5333	0.6333
bac	0.5667	0.6333
goog	0.5333	0.5667
c	0.5	0.5667
sp_500	0.5333	0.7

Table 10: Results from SVM using both no news and with news with (RBF kernel 2)

Test data: Last 30 days. Without retraining

Example 2 – Random forests

- **Two examples:**
  - Lauretto, Silva, Andrade 2013, **“Evaluation of a Supervised Learning Approach for Stock Market Operations”**
  - Theofilatos, Likothanassis and Karathanasopoulos 2012, **“Modeling and Trading the EUR/USD Exchange Rate Using Machine Learning Techniques”**

- **Both teams use Random Forests (classification trees) to build classifiers**

Example 2 – Random forests

- **Lauretto et al. – methodology**
  - Daily equities data (OHLCV)
  - Inputs are SMAs, EMAs, ROC, Stoch, MACD, RSI
  - Classes are: {Buy-Sell, Sell-Buy, Do-nothing}

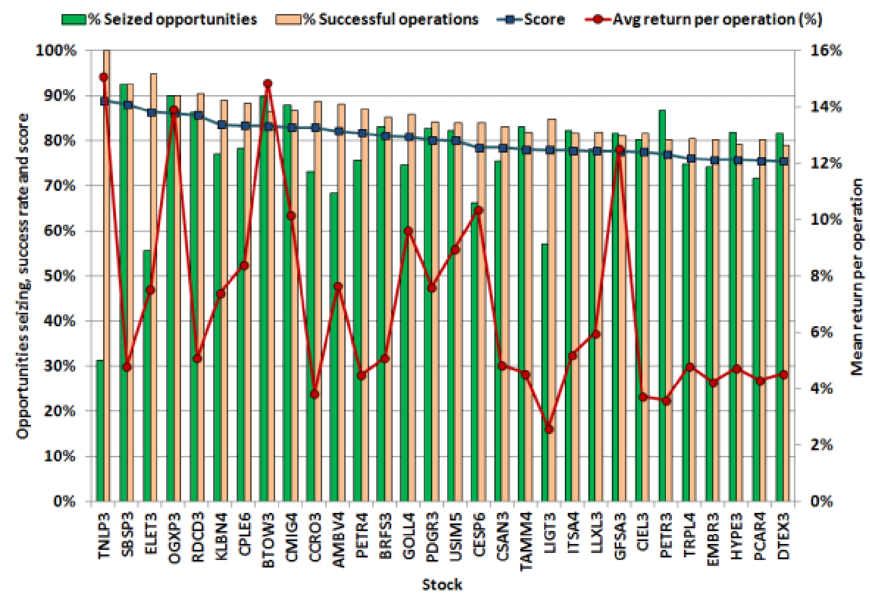


Figure 2: Performance indicators *SeizOport*, *SuccOper*, *AvgRetOper* and *Score* for the 30 stocks with maximum score values.

- **Lauretto et al. – results*:**
  - 80% "successful devised operations"
  - 70% "seized opportunities"
  - Average return per operation: 4%

* WTF???

Example 2 – Random forests

- **Theofilatos et al. – methodology**
  - Predicting one day ahead EUR/USD
  - Only use autoregressive inputs, i.e. up to 10 days lagged data used as inputs
  - Compared a bunch of ML algos, including SVM (with RBF kernel), RFs, NN, Naïve Bayes

Theofilatos et al. – results:

	Artificial Neural Networks (MLP)	**SVM**	**Random Forests**
Information Ratio (excluded costs)	0.22	0.43	0.72
Annualized Volatility (excluded costs)	11.46%	11.46%	11.45%
Annualized Return (excluding costs)	2.51%	4.90%	8.29%
Maximum Drawdown (excluding costs)	-18.89%	-14.74%	-9.94%
Correct Directional Prediction	50.12%	52.65%	53.50%
Transaction costs	0.92	0.92%	1.01%
Annualized Return (including costs)	1.59%	3.98%	7.28

Problems and gotchas

- **Are YOU smarter than a machine? Don't forget everything you already know…**
- **If you don't believe it, it's probably not real!**

- **How many datasets should you use?**
  - 3! Training, validation, out of sample testing
- **Input data needs pre-processing and scaling**
- **Over-fitting – regularisation, out of sample**
- **Computation speed, (online) (re-)training**
- **No peeping!**
- **GI/GO –> SNAFU**

Resources

- **Software**
  - Quantopian
  - Lucena
  - Azure
  - MATLAB
  - Python, R
  - WEKA
  - RapidMiner
  - JavaML
  - LibSVM

- **Online learning**
  - Coursera
  - Udacity
  - Nando de Freitas YouTube channel
  - Quantopian
  - Lucena
  - (wikipedia)

What next?

- **DO NOT DESTROY MANKIND!**

Questions?

Dr. Aly Kassam

aly@quantsupport.com