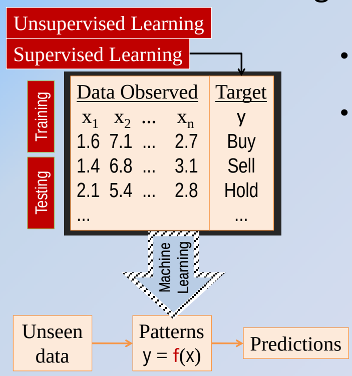
Machine Learning

What is the research agenda?

How to measure success?

How to learn?

Machine Learning Overview



- Supervised learning is function fitting
- Does function f exist?

What y to learn?

How to prepare x?

How should f look like?

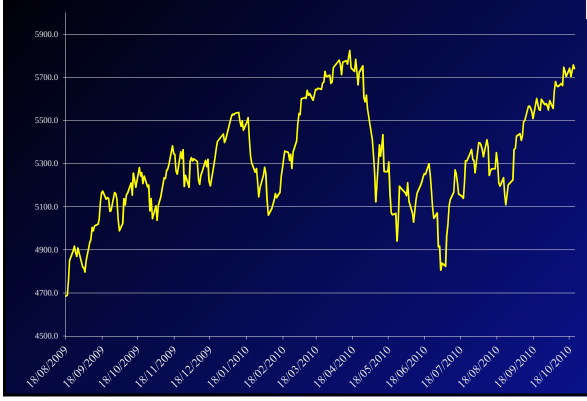
How to evaluate f ?

How to find f ?

What to learn?

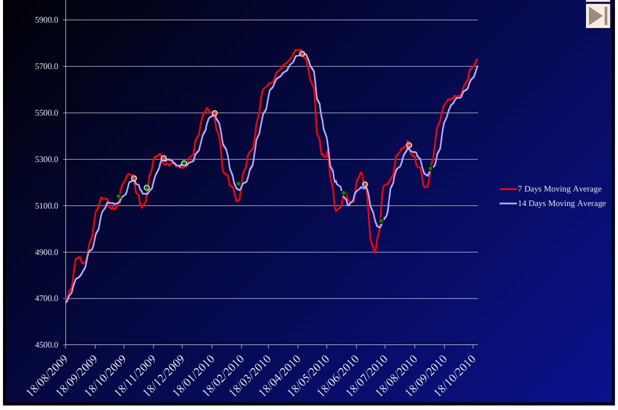
- We could try to predict the price tomorrow
- We cold try to predict whether prices will go up (or down) by a margin
  - E.g. will the price go up by r% within n days?
- Notes:
  - Always ask: can the market be predicted?
  - There is no magic in machine learning
  - Harder task → less chance to succeed

FTSE 2009.08.18-2010.10.22



FTSE 2009.08.18-2010.10.22

FTSE 2009.08.18-2010.10.22



Moving Average Rules to Find

- Let
  - m-MA be the m-days moving average
  - n-MA be the n-days moving average
  - m < n
- Possible rules to find:
  - If the m-MA ≤ n-MA, on day d, but m-MA > n-MA on day d+1, then buy
  - If the m-MA ≥ n-MA, on day d, but m-MA < n-MA on day d+1, then sell

Performance Measures

- Summarising results with Confusion Matrix
  - Basic measures
  - Matthews correlation coefficient
- Receiver Operating Characteristic, ROC
  - Performance
  - Types of classifiers
- Dealing with Scarce Opportunities
  - Unintelligent / Intelligent improvement

Confusion Matrix

Prediction

Reality

	–	+	
–	5	2	7
+	1	2	3
	6	4	10

Reality	Prediction
–	–
+	+
+	–
–	–
–	–
–	–
+	+
–	+
–	+
–	–

Performance Measures

Reality

Ideal Predictions

	–	+	
–	7	0	7
+	0	3	3
	7	3	10

Actual Predictions, Example

	–	+	
–	5	2	7
+	1	2	3
	6	4	10

RC = (5+2) ÷10 = 70%
Precision = 2 ÷ 4 = 50%
Recall = 2 ÷ 3 = 67%

More on Performance Measures

Predictions

Reality

	–	+	
–	TN 5	FP 2	7
+	FN 1	TP 2	3
	6	4	10

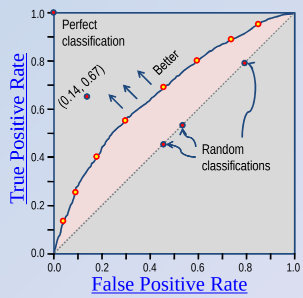
False Positive Rate = FP/(FP+TN) = 2/(2+5) = 14%

True Positive Rate = recall = TP/(TP+FN) = 2/(2+1) = 67%

TN = True Negative
FN = False Negative = Miss = Type II Error
FP = False Positive = False Alarm = Type I Error
TP = True Positive

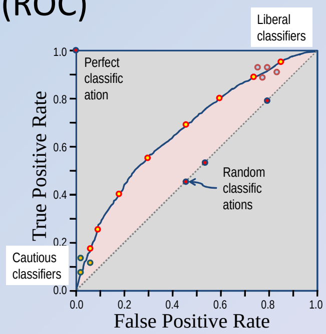
Receiver Operating Characteristic (ROC)

- Each predication measure occupies one point
  - Points on diagonal represent random predictions
- A curve can be fitted on multiple measures
  - Note: points may not cover widely
- Area under the curve measures the classifier's performance



Receiver Operating Characteristic (ROC)

- A classifier may occupy only one area in the ROC space
- Other classifiers may occupy multiple points
- Some classifiers may be tuned to occupy different parts of the curve



Scarce opportunities 1

Reality

Ideal Predictions

	–	+	
–	9,900	0	99%
+	0	100	1%
	99%	1%	

Ideal prediction
Accuracy = Precision = Recall = 100%

Easy Predictions

	–	+	
–	9,900	0	99%
+	100	0	1%
	100%	0%	

Easy score on accuracy
Accuracy = 99%, Precision = ?
Recall = 0%

- Easy to achieve high accuracy
- Worthless predications

Scarce opportunities 2

Reality

Easy Predictions

	–	+	
–	9,900	0	99%
+	100	0	1%
	100%	0%	

Easy score on accuracy
Accuracy = 99%, Precision = ?
Recall = 0%

Random Predictions

	–	+	
–	9,801	99	99%
+	99	1	1%
	99%	1%	

Random move from – to +
Accuracy = 98.02%
Precision = Recall = 1%

- Unintelligent improvement of recall
  - Random +'ve predictions

Scarce opportunities 3

Reality

Random Predictions

	–	+	
–	9,801	99	99%
+	99	1	1%
	99%	1%	

Random move from – to +
Accuracy = 98.02%
Precision = Recall = 1%

Predictions

	–	+	
–	9,810	90	99%
+	90	10	1%
	99%	1%	

Better moves from – to +
Accuracy = 98.2%
Precision = Recall = 10%

- A more useful predication
- Sacrifice accuracy (from 99% in easy prediction)

Machine Learning Techniques

- Early work: Quinlan's ID3 / C4.5/ C5
  - Building decision trees
- Neural networks
  - A solution is represented by a network
  - Learning through feedbacks
- Evolutionary computation
  - Keep a population of solutions
  - Learning through survival of the fittest

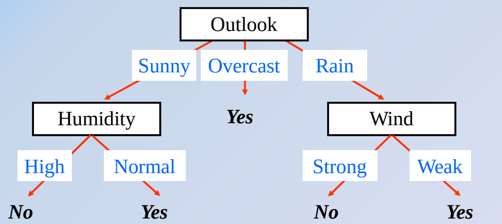
ID3 for machine learning

- ID3 is an algorithm invented by Quinlan
- ID3 performs supervised learning
- It builds decision trees
- Perfect fitting with training data
- Like other machine learning techniques:
  - No guarantee that it fits testing data
  - Danger of "over-fitting"

Example Classification Problem: Play Tennis?

Day	Outlook	Temper.	Humid.	Wind	Play?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

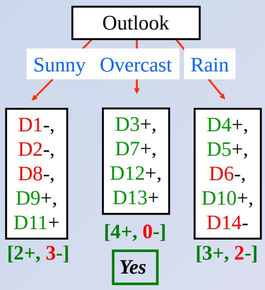
Example Decision Tree



- Decision to make: *play tennis?*

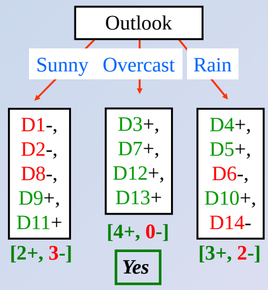
Example: ID3 Picks an attribute

- Pick an attribute *A*
- Compute Gain(S, *A*):
  - *Outlook*: 0.246
  - *Humidity*: 0.151
  - *Wind*: 0.048
  - *Temperature*: 0.029
- *Outlook* is picked



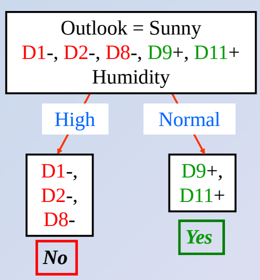
Simplified ID3 in Action (1)

1) Not all examples agree on conclusion
2) Pick one attribute
   - "*Outlook*" is picked
3) Divide examples according to values in *Outlook*
4) Build each branch recursively
   - "*Yes*" for "*Overcast*"



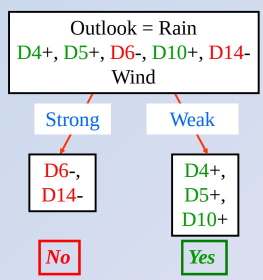
Simplified ID3 in Action (2)

1) Expand *Outlook=Sunny*
2) Not all examples agree
3) Pick one attribute
   - "*Humidity*" is picked
4) Divide examples
5) Build two branches
   - "*No*" for "*High*"
   - "*Yes*" for "*Normal*"

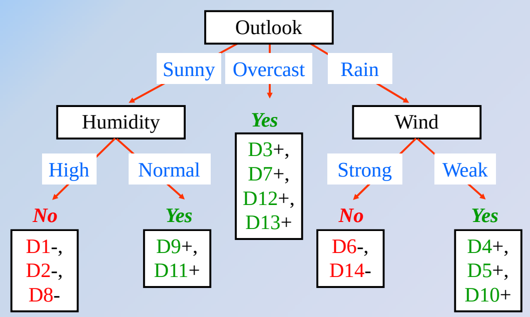


Simplified ID3 in Action (3)

1) Expand *Outlook=Rain*
2) Not all examples agree
3) Pick one attribute
   - "*Wind*" is picked
4) Divide examples
5) Build two branches
   - "*No*" for "*Strong*"
   - "*Yes*" for "*Weak*"



ID3 in Action – Tree Built



Remarks on ID3

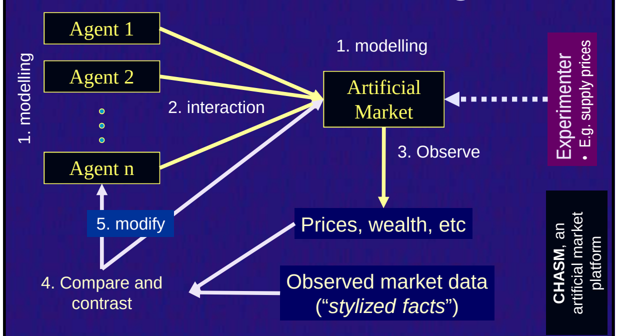
- Decision trees are easy to understand
- Decision trees are easy to use
- But what if data has noise?
  - I.e. under the same situation, contradictory results were observed
- Besides, what if some values are missing from the decision tree?
  - E.g. "Humidity = Low"
- These are handled by C4.5 and See5 (beyond our syllabus)

Exercise Classification Problem - Admit?

Student	Maths	English	Physics	IT	Exam
1	A	A	C	B	Admit
2	A	A	A	C	Admit
3	A	B	C	A	Admit
4	A	A	B	A	Admit
5	B	A	A	C	Admit
6	B	B	C	A	Admit
7	A	B	B	B	Admit
8	B	A	B	C	Admit
9	B	B	A	B	Admit
10	C	A	B	A	Admit
11	C	A	C	A	Reject
12	A	C	C	B	Reject
13	C	A	B	C	Reject
14	B	C	A	C	Reject

Suppose the rule is: "Only accept students with at least three (A or Bs), including at least one A". Could ID3 find it?

Unsupervised Learning in Trading and Markets Mechanism Design

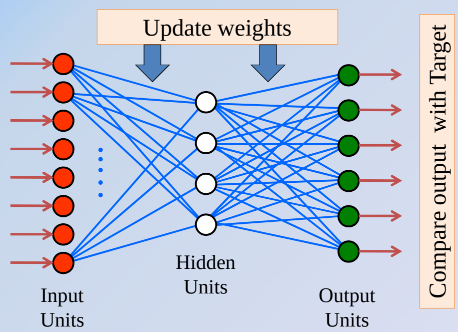


I may not know how a solution looks like, but can tell what is good / bad

Machine Learning Summary

- Supervised learning is function fitting
- First question: what function to find?
- Do such functions exist?
  - If not, all efforts are futile
- How to measure success?
- Finally, find such functions (ID3, NN, EA)
- Caution: avoid too much faith
- Unsupervised learning is generate and test
  - One know what is good/bad when one sees it

Artificial Neural Network



Terminology in GA

String with Building blocks with values: | 1 | 0 | 0 | 0 | 1 | 1 | 0 | — *Chromosome* with *Genes* with *alleles*

evaluation ↓ *fitness*

- Example of a candidate solution
- in binary representation (vs ***real coding***)
- A ***population*** is maintained

Example Problem For GA

- maximize f(x) = 100 + 28x – x^2
  - optimal solution: x = 14 (f(x) = 296)
- Use 5 bits representation
  - e.g. binary 01101 = decimal 13
  - f(x) = 295
- ➢ Note: representation issue can be tricky
  - ➢ Choice of representation is crucial

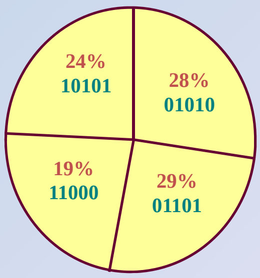
Example Initial Population

To maximize f(x) = 100 + 28x – x2

No.	String	Decim.	f(x)	weight	Accum
1	01010	10	280	28	28
2	01101	13	295	29	57
3	11000	24	196	19	76
4	10101	21	247	24	100
		Accum	1,018		
		averg	254.5		

Selection in Evolutionary Computation

- Roulette wheel method
  - The fitter a string, the more chance it has to be selected



Crossover

			F(x)
1 0 1 0 1 →	1 0 0 1 0	18	280
0 1 0 1 0 →	0 1 1 0 1	13	295
0 1 1 0 1 →	0 1 1 1 0	14	296
0 1 0 1 0 →	0 1 0 0 1	9	271

Sum of fitness: 1,142

Average fitness: 285.5

Evolutionary Computation: Model-based Generate & Test