CSC6515
Data Mining for Big Data

Stan Matwin, Ph.D., CRC
stan@cs.dal.ca

Faculty of Computer Science
Dalhousie University
Fall 2016

Syllabus

Goals of the course:

- Exposure to the basic components of the Big Data science
- Hands-on experience with some of the data tools used for Big Data
- Enabling self-study of advanced concepts in the Big Data context
- Sensitization and exposure to data privacy issues

Course description

In this course, we will focus on Big Data and the pillars of that emerging discipline: machine learning/data mining, and elements of high-performance computing, data visualization, and data privacy. Significant part of the course will be devoted to selected, efficient methods for building models from data using machine learning techniques. In high-performance computing, we will discuss the cloud from the perspective of the Map-Reduce paradigm, and how to engineer Machine Learning algorithms for Map-reduce. We will discuss the fundamentals of data visualization as a new paradigm for data exploration. We will round up the material with discussion of the basic legal and technical issued related to protection of data privacy in the Big Data context.

Marking scheme (tbc)

- Assignments (3): 60%
- Final exam: 40%

Course plan

Sep. 19	Big Data, intro to ML	
Sep. 26	Linear models	
Oct. 3	Bayesian models	
Oct. 10	Txgvng	
Oct 17	Learning theory	
Oct 24	Text mining; LDA	
Oct 31	Cloud - AWS	
Nov 7	Cloud - AWS	
Nov 14	Amazon Machine Learning	
Nov 21	Clustering ; Project presentation	
Nov 28	Visualization	
Dec 5	Current research; course summary	

Textbooks and material:

We will use **parts of** the following books:

- Flach, P. Machine Learning, Cambridge Univ. Press, 2012
- Jiawei Han, Micheline Kamber, and Jian Pei, Data Mining: Concepts and Techniques, 3rd edition, Morgan Kaufmann, 2011
- Zaki, M., Meira, W., Data Mining and Analysis (avail. online)

We will also use materials available on-line. Thise will be given as the class progresses.

We will use select videos from videolectures.net

Expectations about the students taking this class for credit:

- General familiarity with databases
- Analytical and probability skills at the level of 4th yr CS students
- We will limit our practical exercises and the project to the use of the tool (python, Amazo=n Web Services).
- No particular programming skills are assumed, but familiarity with and practice in at least one programming language may be needed for data preprocessing.

Big Data

- Volume
- Velocity
- Variety
- Veracity
- … and Value


IN 60 SECONDS...
1 NEW DEFINITION IS ADDED ON urban dictionary
1,600+ READS ON Scribd.
13,000+ HOURS MUSIC STREAMING ON PANDORA
12,000+ NEW ADS POSTED ON craigslist
370,000+ MINUTES VOICE CALLS ON skype
98,000+ TWEETS
320+ NEW twitter ACCOUNTS
100+ NEW LinkedIn ACCOUNTS
1 associatedcontent NEW ARTICLE IS PUBLISHED
THE WORLD'S LARGEST COMMUNITY CREATED CONTENT!!
6,600+ NEW PICTURES ARE UPLOADED ON flickr
50+ WordPress DOWNLOADS
695,000+ facebook STATUS UPDATES
125+ PLUGIN DOWNLOADS
79,364 WALL POSTS
510,040 COMMENTS
1,700+ Firefox DOWNLOADS
694,445 SEARCH QUERIES
168 MILLION EMAILS ARE SENT
60+ NEW BLOGS
1,500+ BLOG POSTS
70+ DOMAINS REGISTERED
600+ NEW VIDEOS
25+ HOURS TOTAL DURATION
QUESTIONS ASKED ON THE INTERNET...
100+ Answers.com
40+ YAHOO! ANSWERS
20,000+ NEW POSTS ON tumblr.
13,000+ iPhone APPLICATIONS DOWNLOADED
THE LARGEST SOCIAL READING PUBLISHING COMPANY!!
New Craigslist Ads
Google
Google Search
GO-Globe.com

Volume

1000^4	TB	terabyte
1000^5	PB	petabyte
1000^6	EB	**exabyte**
1000^7	ZB	zettabyte
1000^8	YB	yottabyte

- Library of Congress: 10TB of books, about 3PB of digitized material

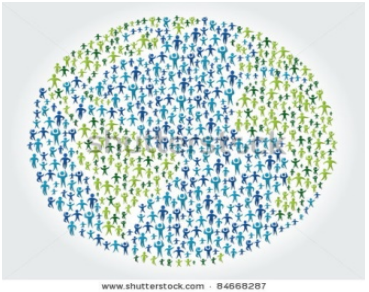
… and some BIG numbers

GB

TB

Petabytes

Exabytes



1 human = 200MB?

Zettabytes

Yottabytes

Velocity

- Sensor data
- Streaming data
- Internet data
- Soc net data
- Etc.

Variety

- Eg medical data
  - Patient data (database, structured)
  - Doctor/nurse notes: text, unstructured
  - Tests: imaging data, graph data
- Challenge: to connect it

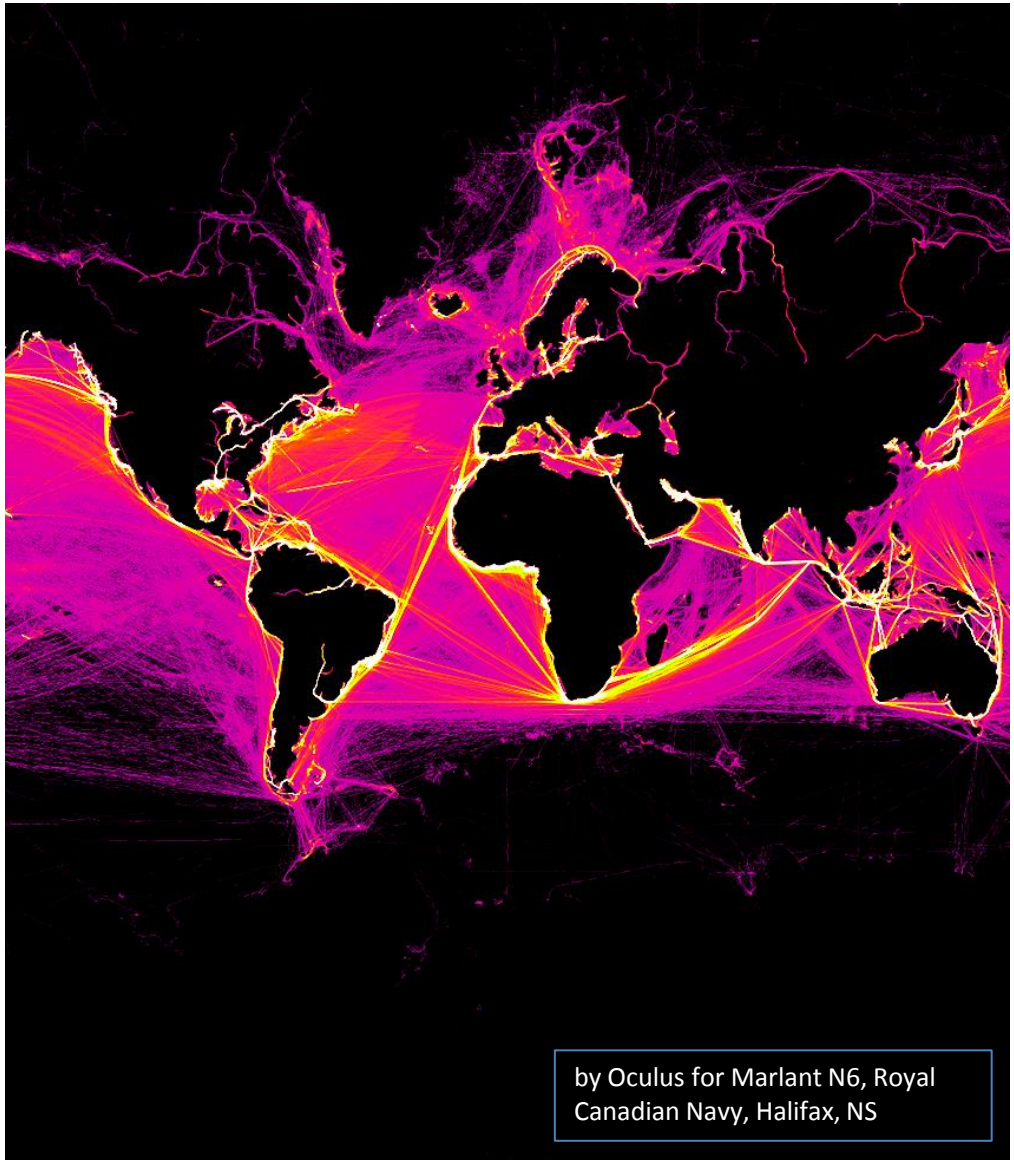
Veracity

- Quality of the data:
  - Noise
  - Missing data
  - Incorrectly entered data
  - …

Another view of Big Data

- Assetization of data
- From data to….
- Actionable knowledge

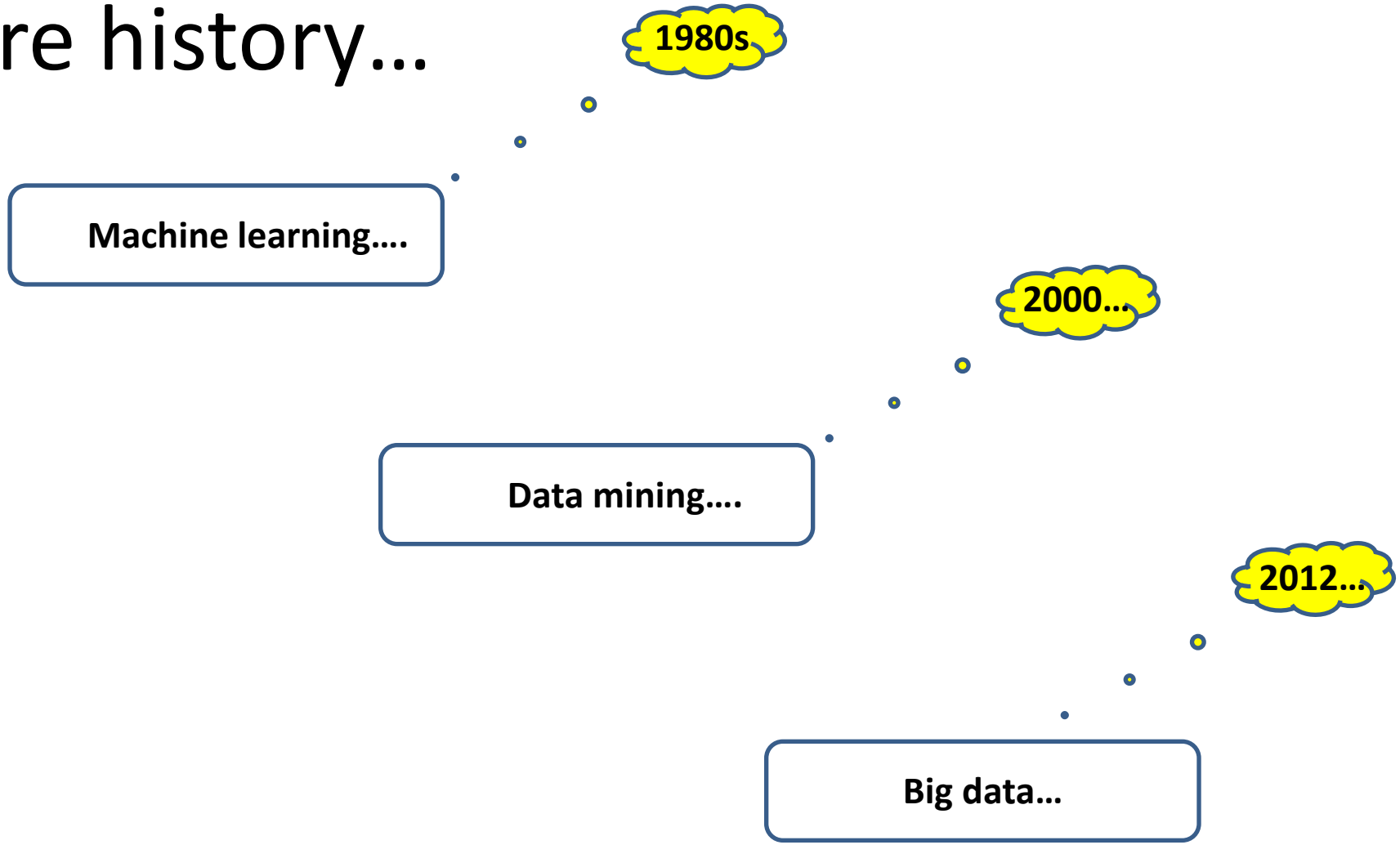
- Collectively represented, shows deep knowledge



Some history

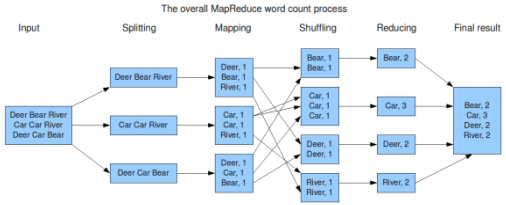
- Technologies behind big data:
  - Data capture/transmission
  - Data bases/storage
  - Data mining
  - HPC (High-Performance Computing)/the Cloud
  - Visualization

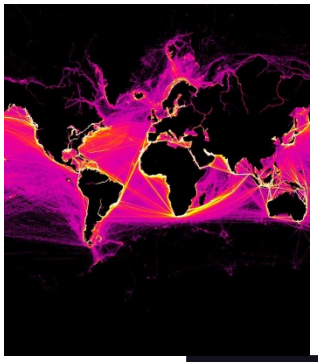
More history…

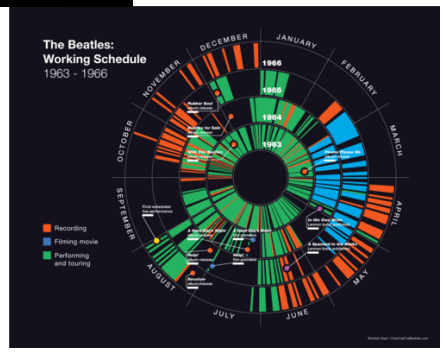


Constituent technologies

- Data analytics
  - Machine Learning
  - Data mining
- HPC
  - Hadoop
- Databases
  - NOSQL
  - Streaming
- Data collection
- Data visualization







Intelligent Maritime Solutions from AIS Satellite Data

- Volume: 400,00 ships; 600M records/month
- Velocity: data constantly generated
- Variety: need to combine other data, eg weather, ocean depth, ocean temp., ship info…
- Veracity: data often entered incorrectly into AIS device

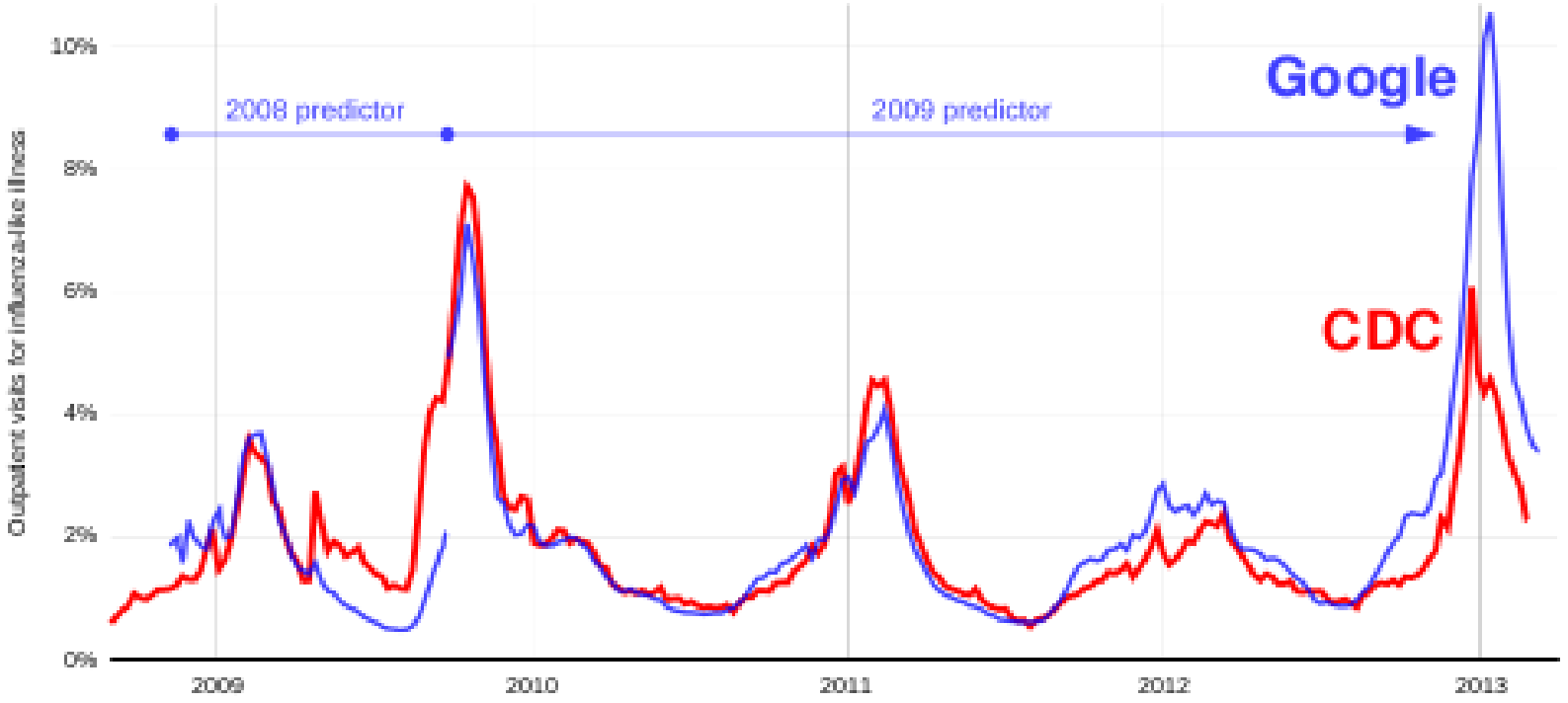
Another example: Google flu….

- Read the paper from Nature

http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en//archive/papers/detecting-influenza-epidemics.pdf

Some Big Data Successes - Google flu

Second divergence in 2012–2013 for U.S.



Machine Learning / Data Mining: basic terminology

- Machine Learning:
  - given a certain task, and a data set that constitutes the task,
  - ML provides algorithms that resolve the task based on the data, and the solution improves with time
- Examples:
  - predicting lottery numbers next Saturday
  - Detecting oil spills on sea surface
  - Assisting Systematic Reviews
  - Profiling
    - Dreams
    - Digital game players
    - Fraudulent CC use

- Data Mining: extracting regularities from a VERY LARGE dataset/database as part of a business/application cycle
- examples:
  - customer churn profiling
  - direct mail targeting/ cross sell
  - security applications: monitoring of
    - Social networks
    - Computer networks
  - prediction of aircraft component failures
  - clustering of genes wrt their beavior

Data Mining Process

- An iterative process which includes the following steps
  - Formulate the problem e.g. Classification/Numeric Prediction ranking
  - Collect the relevant data (No data, no model)
  - Represent the Data in the form of *labeled* examples (a.k.a instances) to be learned from
  - Learn a model/predictor
  - Evaluate the model
  - Fine tune the model as needed

Basic ML tasks

- Supervised learning
  - classification/concept learning – predicting a discrete variable (class attribute)
  - extended to ranking, scoring, and probability prediction (examples)
  - regression: predicting continuous attribute
- Unsupervised learning:
  - clustering: finding groups of “similar” objects
  - associations: in a database, finding that some values of attributes go with some other

Classification: a definition

- Data are given as vectors of attribute values, where the domain of possible values for attribute *j* is denoted as *Aj*, for 1 <= *j* <= *N*. Moreover,a set *C* = {*c*1,., *ck*} of *k* classes is given; this can be seen as a special attribute or label for each record. Often *k* = 2, in which case we are learning a binary classifier.
- Inducing, or learning a classifier, means finding a mapping

  $F: A_1 \times A_2 \times A_N \rightarrow C,$

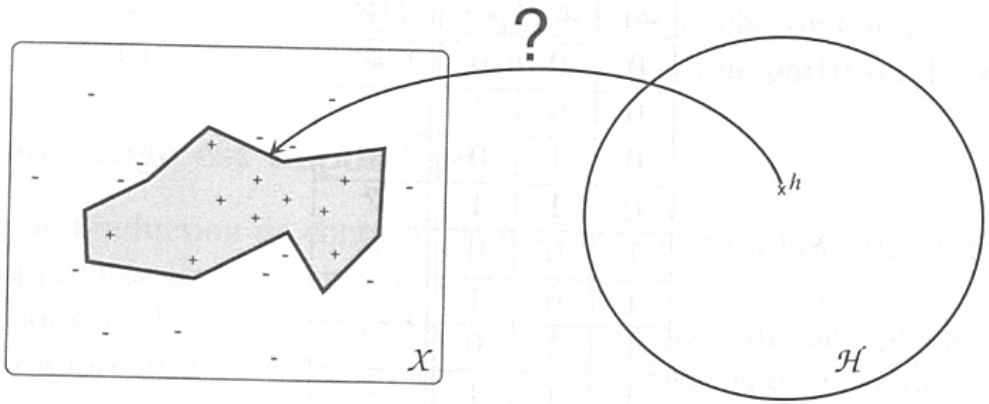
  given a finite training set $X1 = \{<x_{ij}, c_i>, 1 <= j <= N, c_i \in C, 1 <= i <= M\}$ of M labeled examples [comment on noise]

- We assume that data is represented as fixed size vectors of attributes (AVL representation): eg all patients are represented by the same 38 attributes, perhaps in conceptual groupings into personal, social, medical
- F belongs to a fixed language, e.g. F can be
  - a set of n -1 dimensional hyperplanes partitioning an n-dimensional space into k subspaces, or
  - a decision tree with leaves belonging to C, or
  - a set of rules with consequents in C.
- We also want F to perform well, in terms of its predictive power on (future) data not belonging to X1 [predictive power]

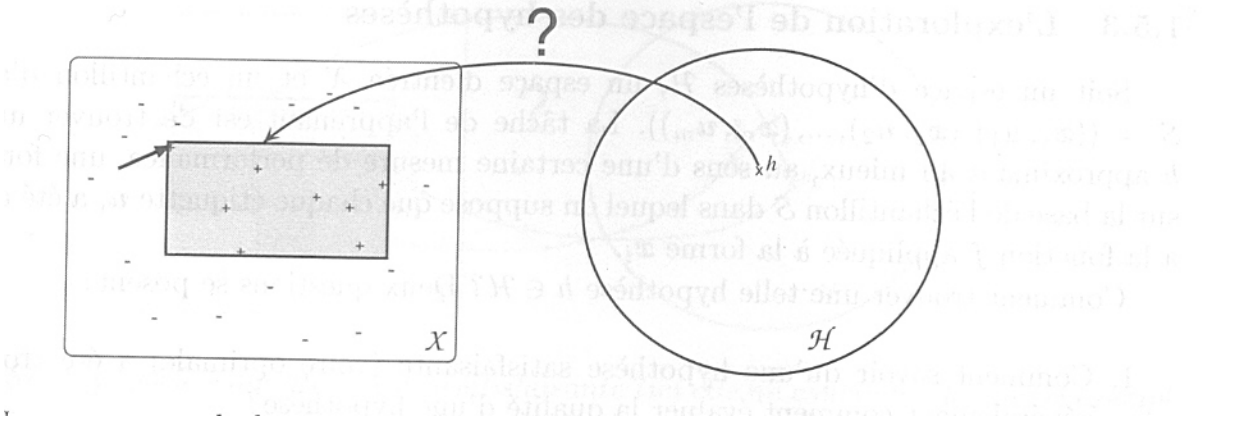
- In data base terminology, we “model” one relation
- There are methods that deal with multi-relational representations (multiple tables), - multi-relational learning AKA Inductive Logic Programming

The language of the hypotheses must be adequate for the concept we learn:

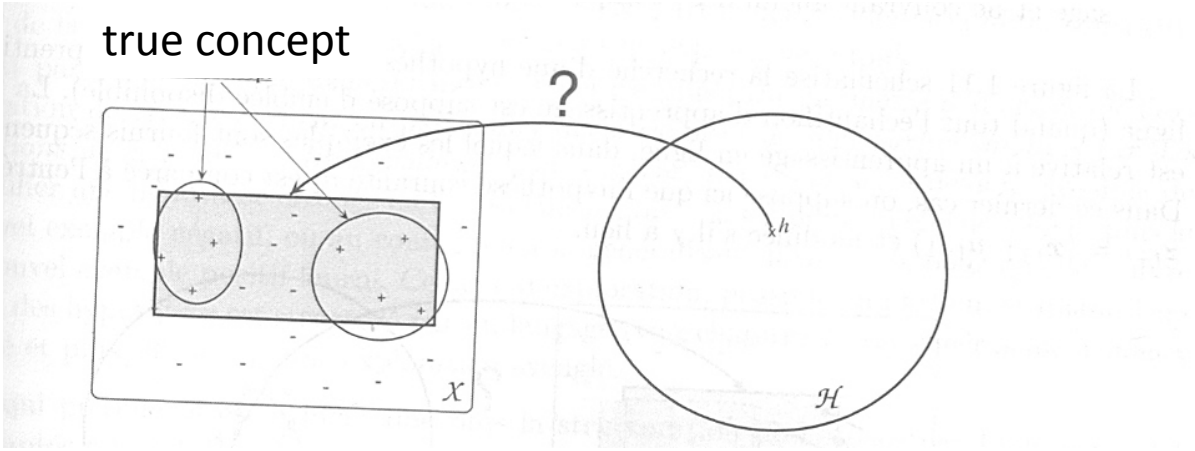
- A concept (a particular type of model = binary classifier) is a partition of the space of all instances (consider the number of concepts for, e.g. 800 examples)



- The shape (i.e. language) of the concept determines how much we generalize:

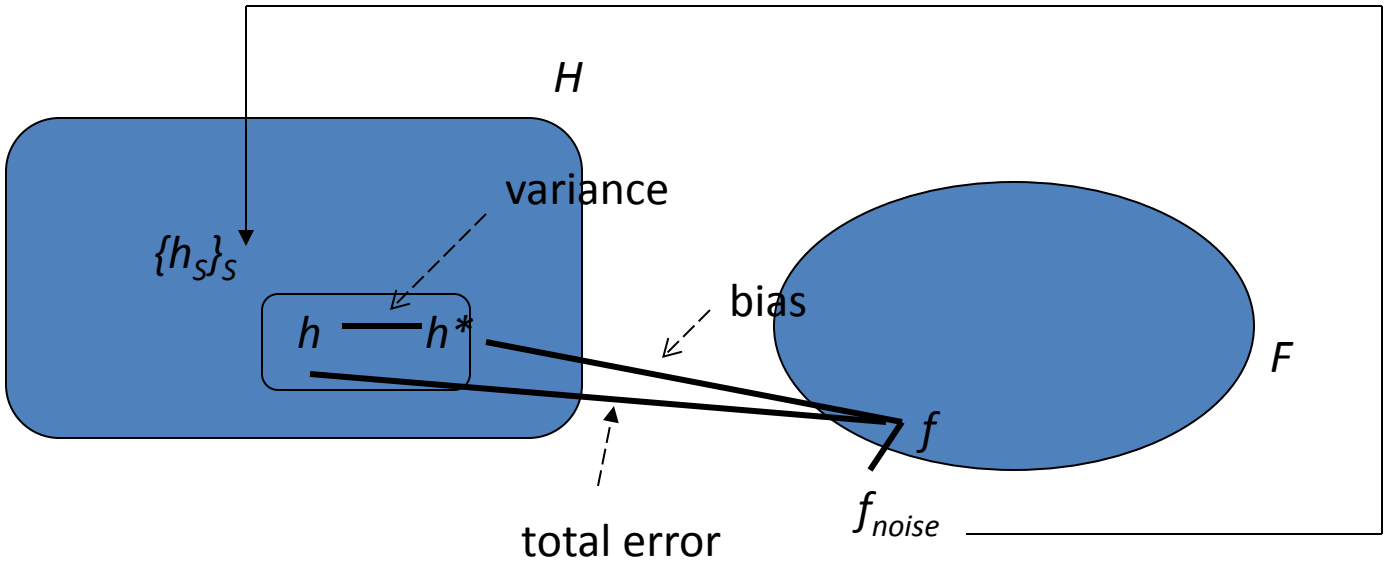


- If the language is not adequate, *l* will be a very poor approximation of *h*:



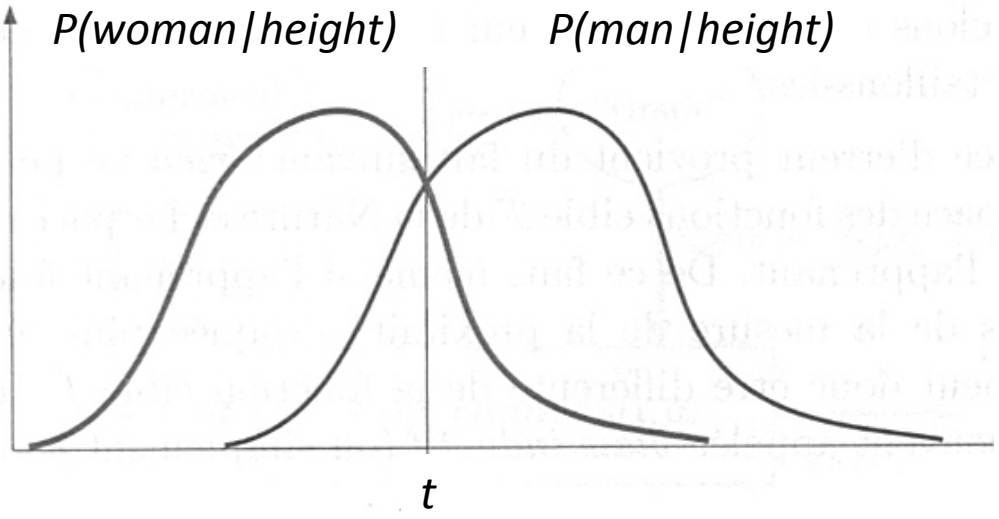
Bias-variance compromise

- Bias: the difference between *h* and *f* due to the language of *H* and *F*
- Variance (estimation error): due to inability of finding *h*,* the *best h* in *H*



Example of bias and variance

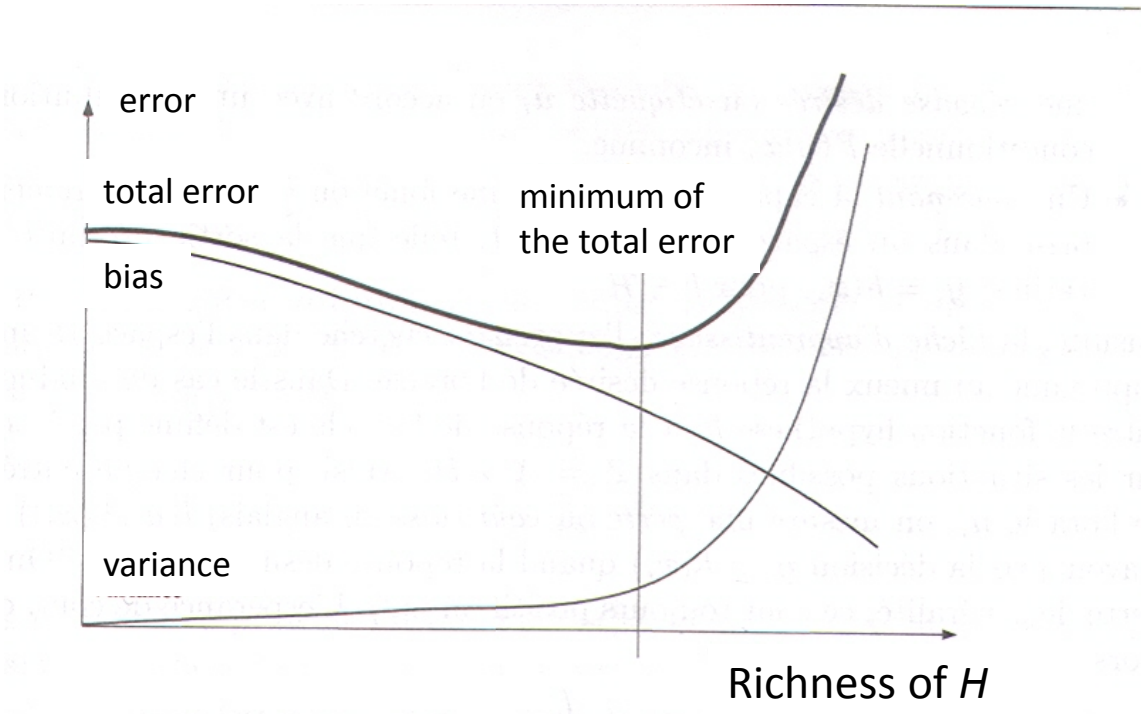
- Learning the sex of a person
- Particularly simple language bias: a "hyperplane". For a single attribute, this is just one number (point on a line)



Example of bias and variance

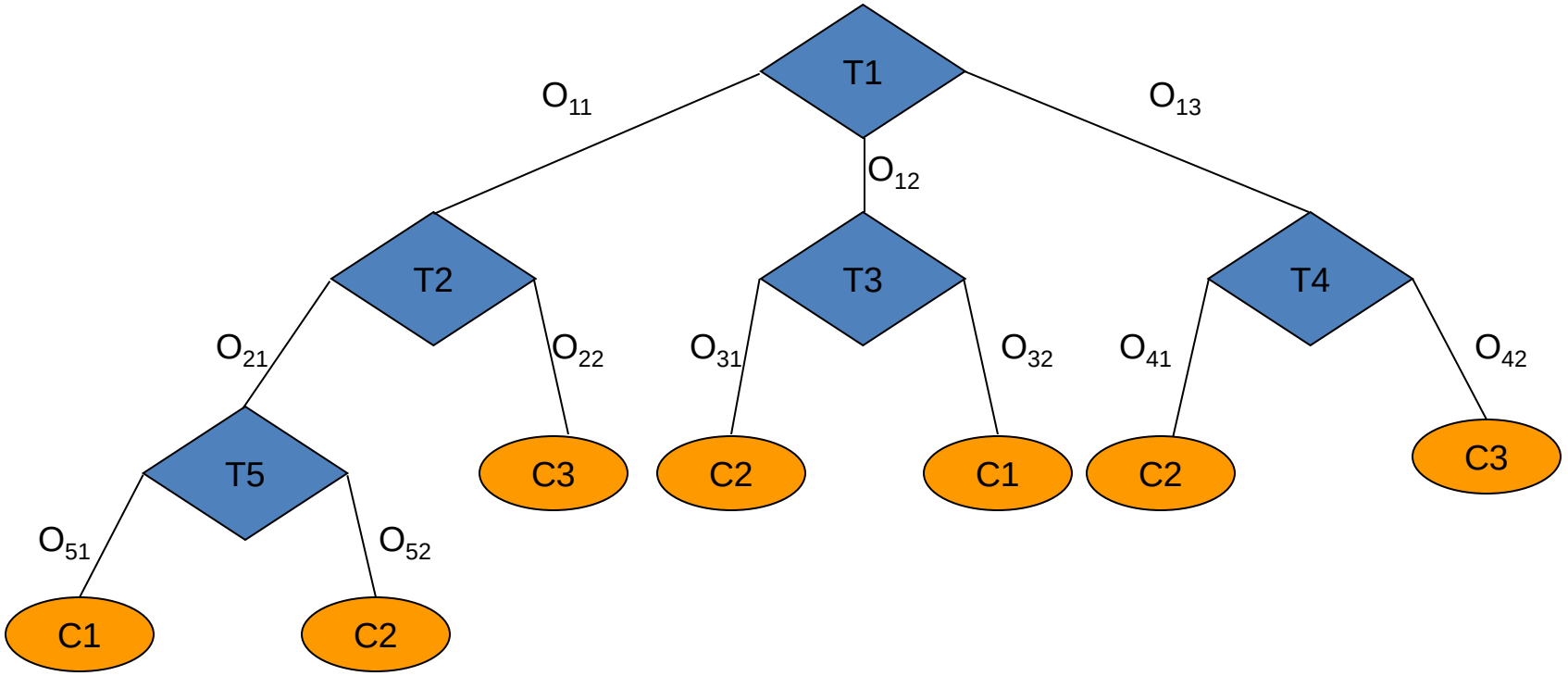
- Bias is very bad: relatively poor choice of H leads to poor discrimination between the two classes
- Variance is good: for different samples, i.i.d. (independently and identically distributed) samples will result is same shape gaussians
- Imagine instead that we have 50 physical attributes of people. Bias is low: there may exist a perfectly discriminant function in this highly dimensional space. But variance is bad: for a limited sample we may not find the best hyperplane

Bias-variance compromise



Decision Trees

(read ch. 1.1-1.5 from the “The Top Ten Algorithms in Data Mining”)



Tests: T1, …, T5

Test Outcomes: O_{11}, …, O_{52}

Predictions: C1, …, C3

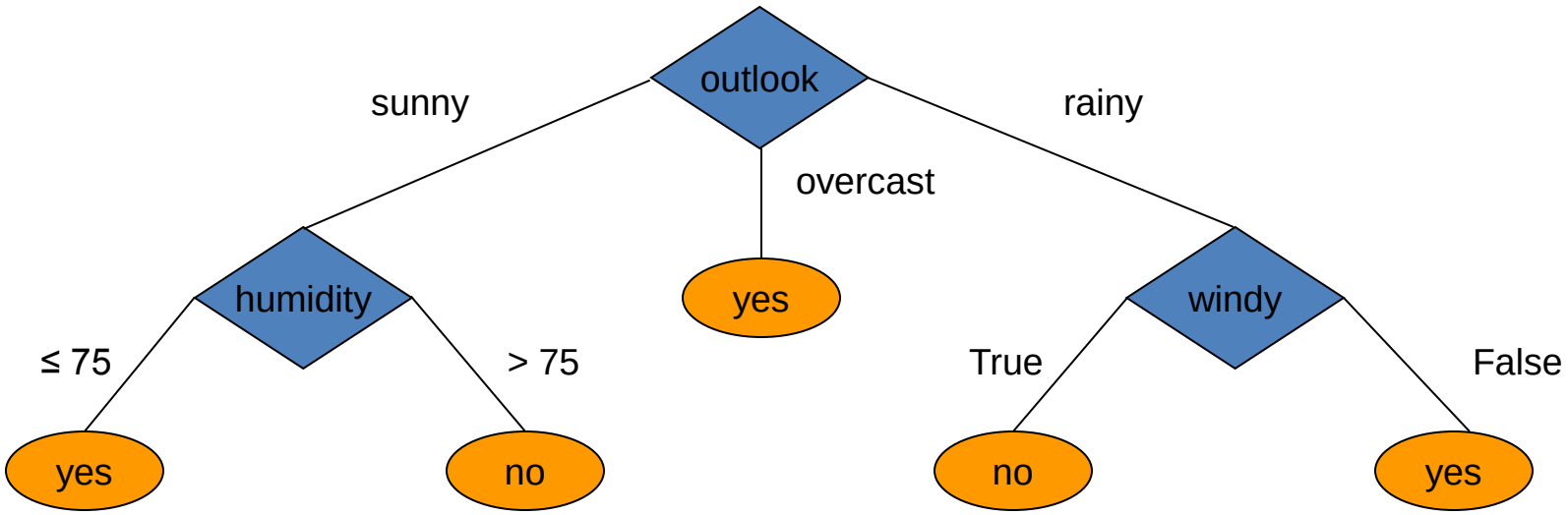
Example 1: Are We Going to Play Outdoors

- Predict whether there will be an outdoor game depending on the existing weather conditions
- Classification: yes (play), no (don’t play)
- Data:A collection of past observation about the days that there was or was not an out door game. The data is already collected and labeled.

Representation for Outdoor Game Prediction

- Each example records the following information (a.k.a *features* or *attributes*)
- outlook {sunny, overcast, rainy}
- temperature real
- humidity real
- windy {TRUE, FALSE}
- play {yes, no} -> the label

Outdoor Game Play Decision Tree



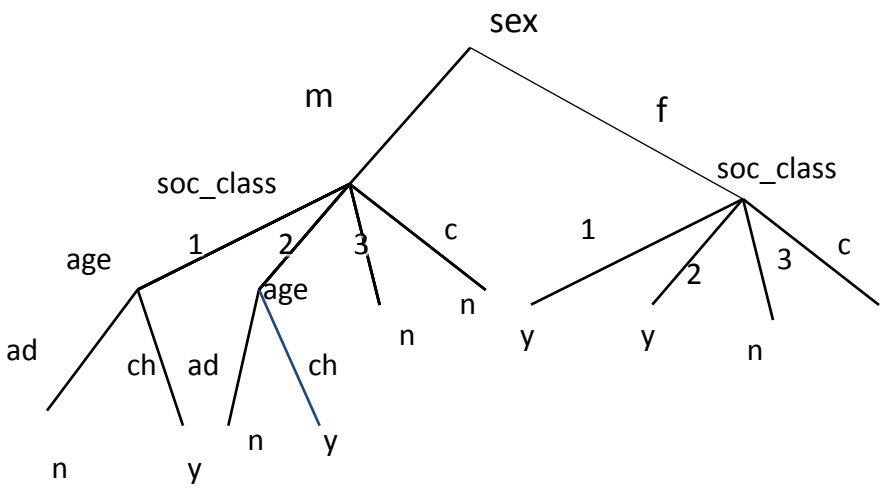
Example 2: Who would survive Titanic's sinking

- Predict whether a person on board would have survived the tragic sinking
- Classification: yes (survives), no (does not survive)
- Data:The data is already collected and labeled for all 2201 people on board the Titanic.

Example 2: Representation for the Titanic Survivor Prediction

- Each example records the following *attributes*
- social class {first class,second class, third class, crew member}
- age {adult, child}
- sex {male, female}
- survived {yes, no}

Titanic Survivor model



y

Induction of decision trees: an algorithm building a DT from data…

building a *univariate (single attribute is tested)* decision tree from a set T of training cases for a concept C with classes C_1,…C_k

Consider three possibilities:

- **T contains 1 or more cases all belonging to the same class C_j. The decision tree for T is a leaf identifying class C_j**
- **T contains no cases. The tree is a leaf, but the label is assigned heuristically, e.g. the majority class in the parent of this node**

- T contains cases from different classes. T is divided into subsets that seem to lead towards collections of cases. A test t based on a single attribute is chosen, and it partitions T into subsets $\{T_1,\ldots,T_n\}$. The decision tree consists of a decision node identifying the tested attribute, and one branch for ea. outcome of the test. Then, the same process is applied recursively to ea.T_i

Choosing the test

- why not explore all possible trees and choose the simplest (Occam's razor)? But this is an NP complete problem. E.g. in the 'Titanic' example there are millions of trees consistent with the data

- idea: to choose an attribute that best separates the examples according to their class label
- This means to maximize the difference between the info needed to identify a class of an example in T, and the same info after T has been partitioned in accordance with a test X
- Entropy is a measure from information theory [Shannon] that measures the quantity of information

- information measure (in bits) of a message is - $\log_2$ of the probability of that message
- notation: S: set of the training examples; freq(C_i, S) = number of examples in S that belong to C_i;

selecting 1 case and announcing its class has info measure -
$\log_2(\text{freq}(C_i, S)/|S|)$ bits

to find information pertaining to class membership in all classes:

$$\text{info}(S) = -\sum_1(\text{freq}(C_i, S)/|S|)*\log_2(\text{freq}(C_i, S)/|S|)$$

after partitioning according to outcome of test X:

$$\text{info}_X(T) = \sum|T_i|/|T|*\text{info}(T_i)$$

gain(X) = info(T) - $\text{info}_X(T)$ measures the gain from partitioning T according to X

We select X to maximize this gain

Data for learning the weather (play/don't play) concept (Witten p. 10)

Day	Outlook	Temp	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Ovcst	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Col	Normal	Weak	yes
6	Rain	Cool	Normal	Strong	No
7	Ovcst	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	Ovcst	Mild	High	Strong	Yes
13	Ovcst	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

$\text{Info}(S) = -(9/14)\log_2(9/14)-(5/14)\log_2(5/14) = 0.940$

Selecting the attribute

- Gain(S, Outlook) = 0.247:

Info(Outlook) = = 5/14*0.970+4/14*0+5/140.970 = 0.693

[(3/5)$\log_2$(3/5)-(2/5)$\log_2$ (2/5) = 0.970]

- Gain(S, Humidity) = 0.151
- Gain(S, Wind) = 0.048
- Gain(S, Temp) = 0.029

- Choose Outlook as the top test

Gain ratio

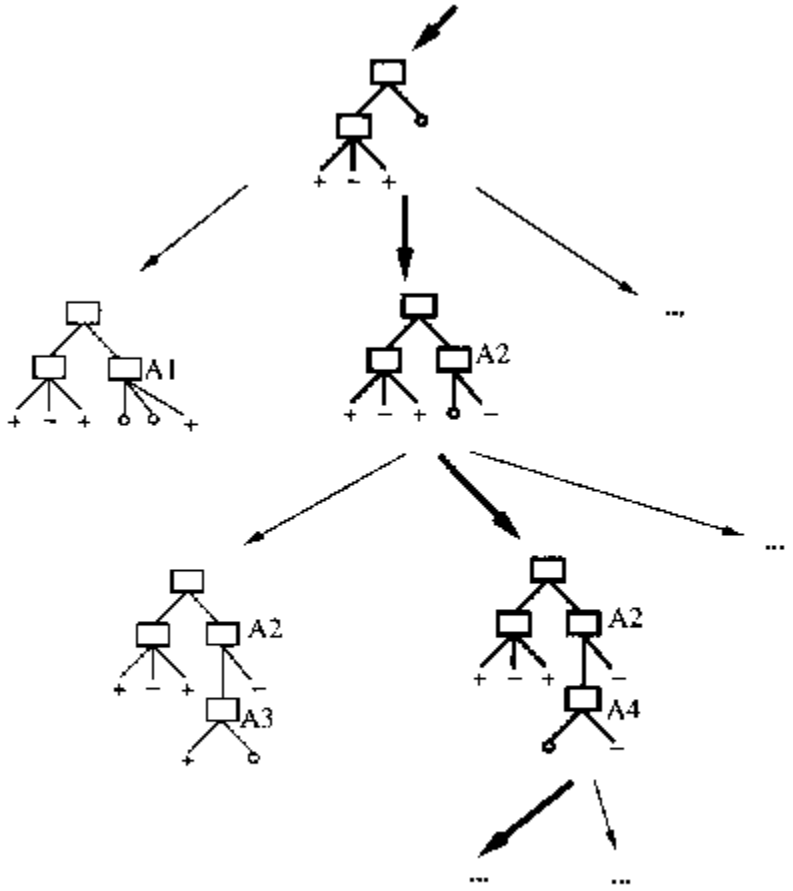
- info gain favours tests with many outcomes (patient id example)
- consider split info(X) = $\sum |T_i|/|T| * \log(|T_i|/|T|)$

measures potential info. generated by dividing T into n classes (without considering the class info)

gain ratio(X) = gain(X)/split info(X)

shows the proportion of info generated by the split that is useful for classification: in the example maximize gain ratio

In fact, learning DTs with the gain ratio heuristic is a search:



- Hill-climbing search
- Info gain is the search heuristic
- Covering the examples is the search criterion
- Inductive bias: sorter trees are preferred

Probability Estimation Trees (PETs)

- Error rate does not consider the probability of the prediction, so in PET
- Instead of predicting a class, the leaves give a probability
- Very useful when we do not want just the class, but examples most likely to belong to a class (e.g. direct marketing)
- No additional effort in learning PET compared to DTs
- Requires different evaluation methods

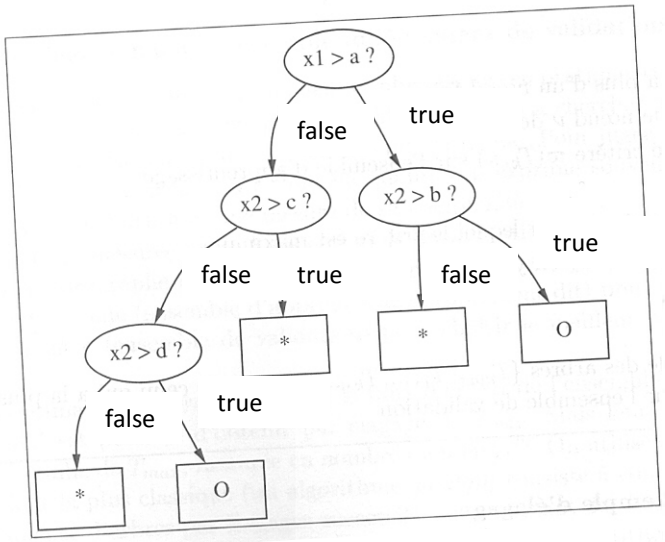
Regression trees

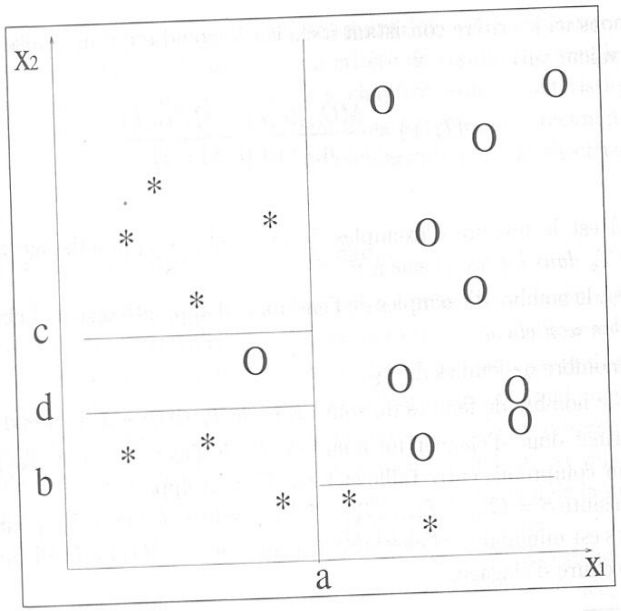
- A slightly different kind of decision trees, where in the leaves the classification is accomplished with a **linear regression model** trained on the contents of the leaf
- Often used on numerical data
- See [Torgo] sec. 2.6.2

Continuous attributes

- a simple trick: sort examples on the values of the attribute considered; choose the midpoint between ea two consecutive values. For m values, there are m-1 possible splits, but they can be examined linearly
- It's a kind of discretization (see later in class)
- cost?

Geometric interpretation of decision trees: axis-parallel area





data: two numerical attributes x1 and x2

Pruning

- Overfitting: getting away from training data
- Predictive performance (on data **not seen** during training)
- Pruning: discarding 1 or more subtrees and replacing them with leaves
- pruning causes the tree to misclassify some training cases. But it may improve performance on validation data

Error-based pruning: suppose error rate can be predicted. Then moving bottom-up consider replacing each subtree with a leaf, or its most frequently used branch. Do it if the replacement *decreases* the error rate

One practical way to measure the post-pruning error rate is to measure the error on a hold-out set (eg 10% of data). This hold-out set would be used for pruning purposes only

Evaluating models

- Accuracy
- MSE
- ROC/AUC
- See http://www.slideshare.net/pierluca.lanzi/machine-learning-and-data-mining-14-evaluation-and-credibility for a good presentation of the evaluation material

Empirical evaluation of accuracy in classification tasks

- The confusion matrix
- Accuracy

- The basic idea in evaluating classifier performance is to count how many times the classifier is correct and incorrect when applied on the testing set.
- This is nicely represented in a *confusion matrix*

label	assigned=T	assigned=F
true=T	TP	FN
true=F	FP	TN

- The most common measure of classifier performance is accuracy ACC= $\frac{TP+TN}{N}$ or its complement error rate = 1−ACC= $1 - \frac{TP+TN}{N} = \frac{FN+FP}{N}$

Computing accuracy: in practice

- partition the set E of all *labeled* examples (examples with their classification labels) into a *training set* X1 and a *testing (validation) set X2.* Normally, X1 and X2 are disjoint
- use the training set for learning, obtain a hypothesis H, set acc := 0
- for ea. element t of the testing set,

  apply H on t; if H(t) = label(t) then acc := acc+1
- acc := acc/|testing set|

Testing - cont'd

- Given a dataset, how do we split it between the training set and the test set?
- cross-validation (n-fold)
  - partition E into n groups
  - choose n-1 groups from n, perform learning on their union
  - repeat the choice n times
  - average the n results
  - usually, n = 3, 5, 10

Comparing models

- Assume you have two models whose performance you want to compare empirically
- One approach
  - Do a k-fold cross-evaluation of each on the same "total" dataset
  - Compute average accuracy across k folds
  - Assess the statistical significance of the difference (eg Student t-test)

MSE

- Mean Square Error – a measure appropriate for
  - Binary setting (two classes)
  - Numerical predictions (regression)