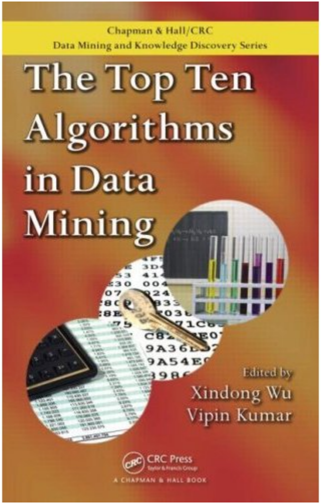
Data Mining and Machine Learning Algorithms

José L. Balcázar

Pascal-2 Bootcamp – Accra, feb 18th, 2011

Overview of the Day

Top Ten Algorithms in Data Mining
IEEE Int. Conf. Data Mining, ICDM'06



Top Ten Algorithms in Data Mining
As voted

1. C4.5 (61 votes)
2. K-Means (60 votes)
3. SVM (58 votes)
4. Apriori (52 votes)
5. EM (48 votes)
6. PageRank (46 votes)
7. AdaBoost (45 votes), kNN (45 votes), Naïve Bayes (45 votes)
8. " (tie)
9. " (tie)
10. CART (34 votes)

http://www.cs.uvm.edu/~icdm/algorithms/index.shtml

Index

Probabilistic Tools

Recap from previous days

1. Probability space, events, random variables;

    mostly discrete spaces;

    always benign measurability situations;

Probabilistic Tools
Recap from previous days

1. Probability space, events, random variables;

   mostly discrete spaces;

   always benign measurability situations;

   ex: petal length 1.234cm or 1.234000001cm?

Probabilistic Tools

Recap from previous days

1. Probability space, events, random variables;

   mostly discrete spaces;

   always benign measurability situations;

   ex: petal length 1.234cm or 1.234000001cm?

   the difference is about one fourth of the Bohr radius,

   the average radius of a hydrogen atom;

Probabilistic Tools

Recap from previous days

1. Probability space, events, random variables;

   mostly discrete spaces;

   always benign measurability situations;

   ex: petal length 1.234cm or 1.234000001cm?

   the difference is about one fourth of the Bohr radius,

   the average radius of a hydrogen atom;
2. Conditional probability;
3. Bayes Theorem;
4. Independence;
5. Expectation;

   $E[\sum_i \alpha_i * X_i] = \sum_i (\alpha_i * E[X_i])$;
6. Empirical frequencies as approximate probabilities.

Partial Implication

Implications with a few exceptions

In propositional logic, $A \Rightarrow B$ does not allow exceptions.

In Data Mining we need to.

And there is no agreement at all about how to do it.

- Support: number of observations in which event X holds: $supp(X)$.
- Normalized support approximates empirically the probability: $\Pr(X) \approx \frac{supp(X)}{n}$.
- Confidence: empirical approximation to the conditional probability:

$$conf(X \to Y) = \frac{supp(XY)}{supp(X)}$$

Confidence
Pros and cons

In favor:

- Quite natural.
- Easy to explain to an educated user.

Confidence
Pros and cons

In favor:

- Quite natural.
- Easy to explain to an educated user.

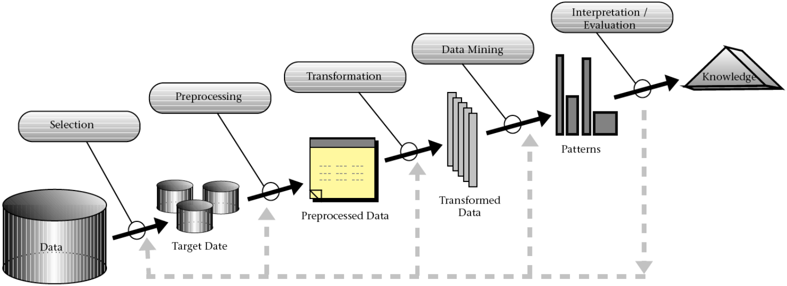
Handle with care:

- High confidence is compatible with negative correlation.
- Normalization solves the problem but introduces another one: symmetry;
- we want confidence to measure an asymmetric notion of partial implication;
- large repertory of alternative measures.

Index

Knowledge Discovery in Data

The process (U Fayyad, 1996)



Knowledge Discovery in Data

The process (U Fayyad, 1996)

1. Selection of data to process,
2. Preprocessing,
3. Transformation,
4. Modeling, with the obtention of models o patterns
5. Validation, interpretation, deploy of the models obtained.

Knowledge Discovery in Data

The process (U Fayyad, 1996)

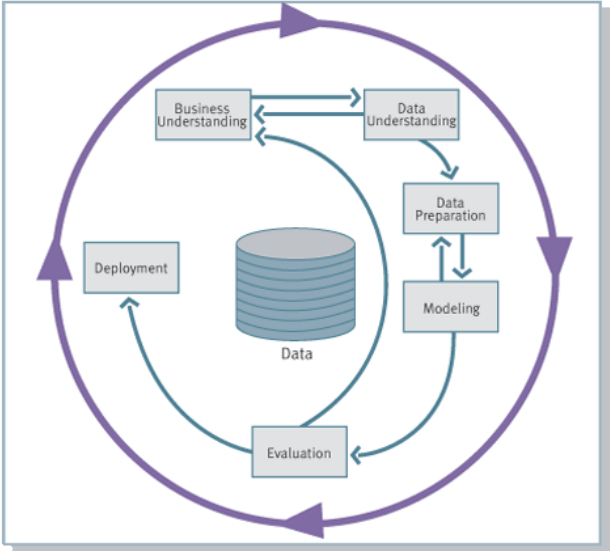
1. Selection of data to process,
2. Preprocessing,
3. Transformation,
4. Modeling, with the obtention of models o patterns
5. Validation, interpretation, deploy of the models obtained.

Modeling (Data Mining for some) allows for ML algorithmics:

- *K-means*,
- *EM*,
- *MAP*,
- *Naïve Bayes*...

CRISP-DM

A more common, industry-designed diagram (1996 as well)



Preprocessing and Transformation

Conceptually trivial... but...

Data formed by *n* observations:

The choice of algorithm (or even of implementation) will dictate a data format... and an encoding!

- Relational: like an SQL table (arff, csv);
- Transactional: set of sets of items (binary attributes);
- Binary Transactional: idem, as *bit-vectors* (arff, csv);
- Relational Transactional: a relational form that contains the same information as the transactional data (csv).

  Inequivalent: some transformations lose information.

  Tiny stones in your shoes:

  - column headers? row identifiers?
  - separator: comma? whitespace? semicolon?
  - a space or comma somewhere amid the data?
  - an end-of-file character amid the data?

Aim

Of a Knowledge Discovery in Data process

Most data has "low sophistication".

Data Mining

activites attempt at finding "more sophisticated views" of the available data, which must be

- correct,
- novel, and
- actionable,

by extracting nontrivial information that is implicit in the data.

All three conditions are rather difficult to make precise.

Connections to Machine Learning, Databases, Statistics...

Remark:

Eclecticism!

- Data is a sample of something? Data is everything there is!
- Does it sort of looks like it works? Get it on board!

Index

Human Interpretation versus Reality

Truth has the bad habit of being utterly complicated

Language

Our great advantage!

- Comunication among different persons (or the same person in different moments),
- Memorization,
- Creation (and, in particular, colective creation),
- Decision making towards a goal...

Our linguistic ability gives us an abstraction capability with important advantages as of the efficiency of our interaction with reality.

Humans are almost constantly constructing models.

Models
Everywhere!

- Current reality, potential future reality:

  This building, an adder (electronic circuit), a symphony, a software system. . .

Models
Everywhere!

- Current reality, potential future reality:

  This building, an adder (electronic circuit), a symphony, a software system...
- Models:

  A building's blueprints, a circuit diagram, a music score, a set of UML specifications...

Modeling language:

verbal, written, or graphic (or...) convention;

goal-dependent!

Why Models?
Tools to help human comprehension

Model:

Expression of a

- simplified but
- expectedly useful

description of actual or potential facts.

Ingredients:

- Conceptualization of reality.
- Invention allowed.
- Observations, in varying degree of faithfulness.

The Data Mining process aims at modeling on the basis of observations (data) about an existing and complicated reality.

Why Models in Data Mining?

It became a business!

Goal:

A (most often) monetary or (sometimes) human advantage.

- Attain it through successful predictions, at least partially.
- Predicting at random does not bring any advantage: anyone can do it. We want to do better!
  - Predict on the basis of "something".
  - Do we happen to have some data available?
  - Do we happen to have all the data available?
  - Does it suffice to have data?

Why Models in Data Mining?

It became a business!

Goal:

A (most often) monetary or (sometimes) human advantage.

- Attain it through successful predictions, at least partially.
- Predicting at random does not bring any advantage: anyone can do it. We want to do better!
  - Predict on the basis of "something".
  - Do we happen to have some data available?
  - Do we happen to have all the data available?
  - Does it suffice to have data?
- Uncertainty is a crucial ingredient!

  Many endeavors invented to handle incertitude; here we follow more or less classical probability theory and statistics.

Why Models in Data Mining?

It became a business!

Goal:

A (most often) monetary or (sometimes) human advantage.

- Attain it through successful predictions, at least partially.
- Predicting at random does not bring any advantage: anyone can do it. We want to do better!
  - Predict on the basis of "something".
  - Do we happen to have some data available?
  - Do we happen to have all the data available?
  - Does it suffice to have data?
- Uncertainty is a crucial ingredient!

  Many endeavors invented to handle incertitude; here we follow more or less classical probability theory and statistics.
- Inductive biases are crucial as well.

  Data has all rights to mean nothing. Our assumption that they do influences the process.

Risks

Let's be careful about

Main mistake in Data Mining:

Not enough data!

- Analyzing the data or torturing it?
- A misconception that sometimes arises:

  If we have less data, we will find less information.

Risks

Let's be careful about

Main mistake in Data Mining:

Not enough data!

- Analyzing the data or torturing it?
- A misconception that sometimes arises:

  If we have less data, we will find less information.
  Wrong!

Risks

Let's be careful about

Main mistake in Data Mining:

Not enough data!

- Analyzing the data or torturing it?
- A misconception that sometimes arises:

  If we have less data, we will find less information.
  Wrong!
  If we have less data, we will find more information!

Risks

Let's be careful about

Main mistake in Data Mining:

Not enough data!

- Analyzing the data or torturing it?
- A misconception that sometimes arises:

  If we have less data, we will find less information.
  Wrong!
  If we have less data, we will find more information!
  Just that it will be less true!

Risks

Let's be careful about

Main mistake in Data Mining:

Not enough data!

- Analyzing the data or torturing it?
- A misconception that sometimes arises:

  If we have less data, we will find less information.
  Wrong!
  If we have less data, we will find more information!
  Just that it will be less true!
- "Let's just find the algorithm that works for our data."
  - Be careful with "mental overfitting".
  - Find out and learn about Wolpert's "No free lunch theorem".
  - Trying to work on the data with no explicit biases simply hides from us our biases.

Nowadays we start having sometimes "decent" dataset sizes...
...and the problem becomes to process them.

Taxonomy of Modeling Tools in Data Mining

Careful: not universal

- Descriptive Models:
  - Clustering,
  - Association.
- Predictive Models:
  - Classification (Discrimination): non-numeric, unstructured prediction space
  - Categorization and Multiclassification: non-numeric, structured prediction space
  - Ranking: non-numeric prediction on a total ordering
  - Regression (Interpolation): numeric prediction space
    - Linear,
    - Polynomial,
    - ...

Geometry
An important scale

Machine Learning and Data Mining models employ "geometry" to very varying degrees.

- No geometry at all:
  - PAC Learning, most of Query Learning,
  - Propositional Logic and variants (association rules),
  - Basic probabilistic models.
- Some geometry (somewhat algebraic):
  - Parametric views versus parameter-free views;
  - Shows up when we indulge in "continuity assumptions": "close" observations should be "treated similarly".
  - Brings the great power of linear algebra in: can perform wonders, such as working in infinite-dimensional spaces where any single vector would never fit the memory of a finite computer;
  - but: must come up with a sensible notion of distance that makes the continuity assumption sensible.

Index

Clustering
Computer-achieved abstraction

Group observations:

Make up your mind about how to "see" the observations in your dataset grouped together.

- Treat similar cases similarly
  (e. g. marketing campaigns);
- Identify "approximately common" characteristics of population segments;
- Get a more succinct explanation of what is in your data
  such as representing each "cluster" by a single point.

Besides K-Means and EM, there are many more:

K-medoids, PAM, CLARANS, CobWeb, BIRCH, Chameleon, DBSCAN, OPTICS ...

- Spectral Clustering,
- Biclustering and Conceptual Clustering,
- Hierarchical Clustering
  - agglomerative
  - divisive

Besides K-Means and EM, there are many more:

K-medoids, PAM, CLARANS, CobWeb, BIRCH, Chameleon, DBSCAN, OPTICS . . .

- Spectral Clustering,
- Biclustering and Conceptual Clustering,
- Hierarchical Clustering
  - agglomerative
  - divisive

But, ¿what is this “clustering” really? Why so many different algorithms?

Clustering Intuitions

To keep in mind and keep re-interpreting

Optimize

some sort of objetive function in such a way that we get

- "short distances within" each cluster

  (Main condition that observations within the same cluster "look alike"),

- "long distances between" clusters

  (Secondary condition that observations lying in different clusters "do not look alike").

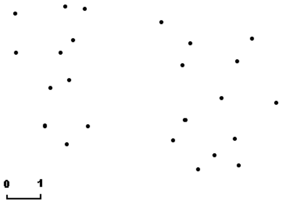
A Formal Approach

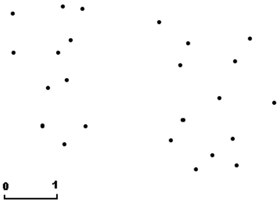
Trying to define "clustering"

Kleinberg axioms:

A very interesting proposal.

- Scale invariance: Where each observation lies matters, but not the unit length.
- Richness: No clustering is externally forbidden "a priori".
- Consistency: Reducing intra-cluster distances and/or enlarging inter-cluster distances does not change the clustering.





A Formal Approach

Trying to define “clustering”

Kleinberg axioms:

A very interesting proposal.

- Scale invariance: Where each observation lies matters, but not the unit length.
- Richness: No clustering is externally forbidden “a priori”.
- Consistency: Reducing intra-cluster distance and/or enlarging inter-cluster distances does not change the clustering.

Theorem

No clustering algorithm at all can achieve all three properties.

A Formal Approach
Trying to define "clustering"

Kleinberg axioms:

A very interesting proposal.

- Scale invariance: Where each observation lies matters, but not the unit length.
- Richness: No clustering is externally forbidden "a priori".
- Consistency: Reducing intra-cluster distance and/or enlarging inter-cluster distances does not change the clustering.

Theorem

No clustering algorithm at all can achieve all three properties.

Choose your favorite target for disbelieving; it is easy now, that is "afterwards"...

¿What are the reasonable axioms then?

(Ben-David and others' work).

K-Means: Goal

Minimize the squared error

Geometry (working hypothesis):

Euclidean distance on the reals (parametric in disguise!).

- Data: n real vectors x_i, positive integer k;
- want: to split them into k clusters C_j;
- we will pick a real vector c_j representing each cluster C_j (its centroid);
- we want to minimize the average squared error:

$$\frac{1}{n}\sum_j \sum_{x_i \in C_j} d(x_i, c_j)^2$$

Note:

We do not require the c_j to be among the x_i.

K-Means: Goal

Minimize the squared error

Geometry (working hypothesis):

Euclidean distance on the reals (parametric in disguise!).

- Data: n real vectors x_i, positive integer k;
- want: to split them into k clusters C_j;
- we will pick a real vector c_j representing each cluster C_j (its centroid);
- we want to minimize the average squared error:

$$\frac{1}{n}\sum_{j}\sum_{x_i \in C_j} d(x_i, c_j)^2$$

Note:

We do not require the c_j to be among the x_i.

Bad news: Utterly infeasible

Complexity theorists say: *NP-hard*.

K-Means: Partial Approach

Let's think a bit more about it

If heavens would give us the centroids:

Then, constructing the clusters is easy: each point to its closest centroid, as otherwise the error increases.

K-Means: Partial Approach

Let's think a bit more about it

If heavens would give us the centroids:

Then, constructing the clusters is easy: each point to its closest centroid, as otherwise the error increases.

If heavens would give us the clusters:

Then, finding the centroids is easy: minimize $\sum_{x_i \in C} d(x_i, c)^2$, by forcing the derivative to zero; each centroid is set at the mass center of its cluster, as otherwise the error increases.

K-Means: HowTo

Stage-wise approximation

We alternate

among the two things we know how to do, starting from k initial centroid candidates:

- recompute the clusters,
- recompute the centroids,
- repeat.

Initial candidates:

- Random?
- One random, then further data points each as far as possible from the previous ones?
- Often advisable: try several runs!

We will be revisiting K-Means in the afternoon.

Index

Probabilistic Prediction

Probability-based predictive models

Context:

Classification.

- Relational data:
  - structured in tuples of attribute/value pairs.
- To predict: the value of one chosen "class" attribute.
- Probabilistic prediction in a merely frequentist sense: counting;
- when is the prediction to be issued?

Context:
Classification.

- Relational data:
  - structured in tuples of attribute/value pairs.
- To predict: the value of one chosen "class" attribute.
- Probabilistic prediction in a merely frequentist sense: counting;
- when is the prediction to be issued?
  - before seeing anything? "a priori" predictor: the most common value for the class (*ZeroR* predictor);

Probabilistic Prediction

Probability-based predictive models

Context:

Classification.

- Relational data:
  - structured in tuples of attribute/value pairs.
- To predict: the value of one chosen "class" attribute.
- Probabilistic prediction in a merely frequentist sense: counting;
- when is the prediction to be issued?
  - before seeing anything? "a priori" predictor: the most common value for the class (*ZeroR* predictor);
  - after seeing all values for all non-class attributes? "a posteriori" predictor: the most common value for the class, conditioned to the values seen (*MAP* predictor, for "maximum a posteriori").

$$\arg\max_C\{Pr(C|A_1 \dots A_n)\}$$

MAP Prediction
Unfortunately infeasible

A small case:
Task of binary classification:

- Assume ten attributes with four values each;
- Then we need to store 2^{20} conditional probabilities;
- and we need to estimate 2^{20} conditional probabilities.

Rule of thumb:
Ten or more observations per parameter to estimate might be still far from sufficient, but are necessary anyway; with less, don't even dream.

MAP Prediction
Unfortunately infeasible

A small case:
Task of binary classification:

- Assume ten attributes with four values each;
- Then we need to store 2^{20} conditional probabilities;
- and we need to estimate 2^{20} conditional probabilities.

Rule of thumb:
Ten or more observations per parameter to estimate might be still far from sufficient, but are necessary anyway; with less, don't even dream.

However,
we will see in the lab a successful application of MAP.

MAP Prediction

Unfortunately infeasible

A small case:

Task of binary classification:

- Assume ten attributes with four values each;
- Then we need to store 2^{20} conditional probabilities;
- and we need to estimate 2^{20} conditional probabilities.

Rule of thumb:

Ten or more observations per parameter to estimate might be still far from sufficient, but are necessary anyway; with less, don't even dream.

However,

we will see in the lab a successful application of MAP.
(But I will be cheating!)

Conditional Independence Assumption

One way out

Bayes rule

Applied to $\arg\max_C \{Pr(C|A_1 \dots A_n)\}$:

$$Pr(C|A_1 \dots A_n) =$$
$$Pr(A_1 \dots A_n|C) * Pr(C)/Pr(A_1 \dots A_n)$$

We can forget about the divisor, as it is the same for all values of C and does not modify the max.

Now we assume independence conditioned to the class value:

$$Pr(A_1 \dots A_n|C) * Pr(C) =$$
$$Pr(A_1|C) * \dots * Pr(A_n|C) * Pr(C)$$

Naïve Bayes

Rather good for such a simple approach

Precompute $Pr(A_i|C)$ for each value of each attribute conditioned to the class value; do it through the empirical frequency.

Instead of predicting

$$\arg\max_C\{Pr(C|A_1 \ldots A_n)\},$$

we predict

$$\arg\max_C\{Pr(A_1|C) * \ldots * Pr(A_n|C) * Pr(C)\}$$

Variant: the “Laplace correction” makes up for cases that might be potentially missing; some tools (like Weka) apply it (without warning).

Predictor Evaluation

Simplest case first: binary accuracy

Confusion matrix

(also known as Contingency matrix):

- True positives (positive prediction, hit)
- False positives (positive prediction, fail: false alarm)
- True negatives (negative prediction, hit)
- False negatives (negative prediction, fail)

Warning:

Note that our reference to the true label is only indirect.

Simple generalization to an $n \times n$ confusion matrix if the problem at hand consists of n class values.

Accuracy, hit ratio:

Number of hits divided by total number of predictions.

We see the accuracy of Naïve Bayes on some examples.

Further Predictor Evaluation

Sometimes accuracy is insufficient

Alternative quantities:

- Confidence of "positive label" ⇒ "positive prediction":

  Sensitivity (recall in *IR*): ratio of true positives to all positively labeled cases;
- Confidence of "positive prediction" ⇒ "positive label":

  Precision: ratio of true positives to all positively predicted cases;
- Confidence of "negative label" ⇒ "negative prediction":

  Specificity: ratio of true negatives to all negatively labeled cases.

Exercise:

Express accuracy as a linear combination of sensitivity and specificity, and interpret the weights.

ROC Space

Predictors lead to points in ROC space

Consider the unit square:

Top left will mean performing quite well.

- The x coordinate is the false positive rate: the ratio of false positives to negative labels (1 minus the specificity).

  (Specificity backwards.)

- The y coordinate is the true positive rate: ratio of true positives to positive labels (the sensitivity).

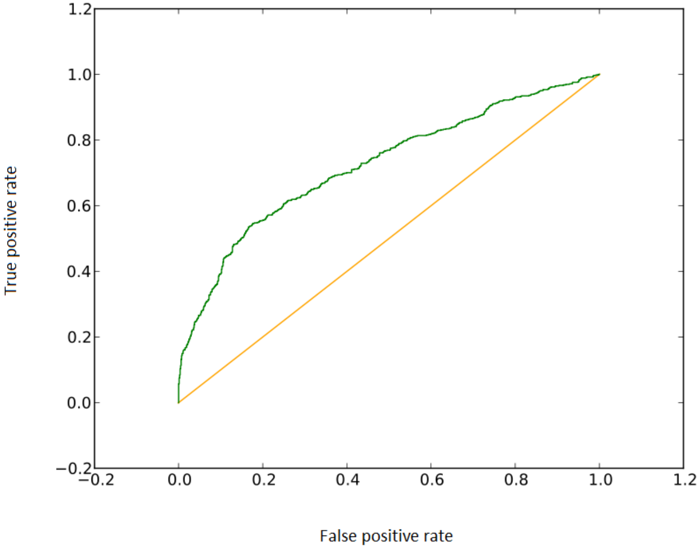
Exercise:

Find the intuitive meaning of various regions of ROC space:

- Half-square below the main diagonal,
- around the center,
- near the corners...

ROC space and ROC curves

The curve is formalized in a minute



Towards ROC Curves

Some predictors provide further information

Ranked predictions:

Predictors that may "bet" on pairs of observations, effectively sorting them.

- For instance, MAP and Naïve Bayes have several options:
  - Higher probability for the "positive" class value;
  - Larger difference of probabilities with respect to other class values;
- Regression-based predictors inherit the real line ordering;
- *Information Retrieval* algorithms are often able to order observations according to the expected relevance.

The ROC Curve

For predictors that are able to rank their observations

Tweak the predictor (usually by thresholding or by sorting all the n observations), so as to classify as negative exactly k points.

ROC curve:

(Receiver/Relative Operating Characteristics).

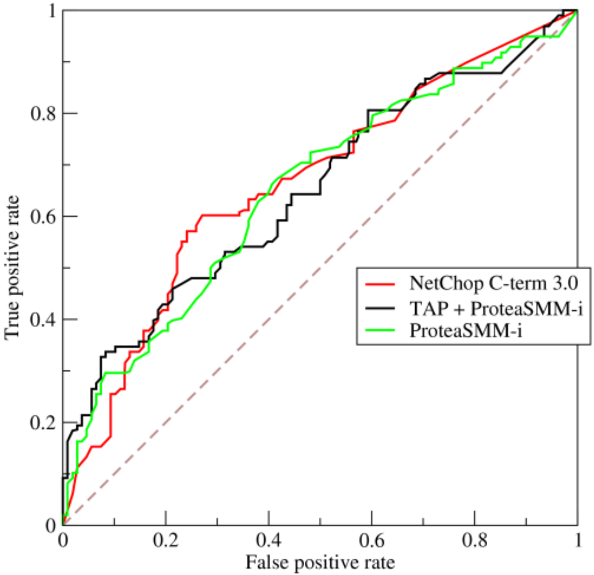
for each k from 0 to n,

plot the ROC space point corresponding to predicting negatively to the k lowest-ranked observations.

We get a curve from (0,0), where we reject everything and there are no false positives, all the way to (1,1) where we accept everything and there are no false negatives.

Examples of ROC curves

Source: Wikipedia, 2009



The Area Under the ROC Curve, AUC

Fashionable but dangerous

Motivation:

ROC Curves often do not lead to a clear winner among several choices of a classifier.

- AUC reduces each classifier's performance on a dataset to a single number.
- Thus allowing us to compare classifiers.
- However, it corresponds to weighting differently the false positive errors than the false negative errors,

The Area Under the ROC Curve, AUC

Fashionable but dangerous

Motivation:

ROC Curves often do not lead to a clear winner among several choices of a classifier.

- AUC reduces each classifier's performance on a dataset to a single number.
- Thus allowing us to compare classifiers.
- However, it corresponds to weighting differently the false positive errors than the false negative errors,
- and the weights depend on the classifier.

The Area Under the ROC Curve, AUC

Fashionable but dangerous

Motivation:

ROC Curves often do not lead to a clear winner among several choices of a classifier.

- AUC reduces each classifier's performance on a dataset to a single number.
- Thus allowing us to compare classifiers.
- However, it corresponds to weighting differently the false positive errors than the false negative errors,
- and the weights depend on the classifier.
- Thus, we should avoid that usage.
- See Hand (Machine Learning Journal, 2009) for further explanations and alternatives.

Index

(Definite) Horn Formulas

Definiteness issues glossed over

Propositional world

Boolean-valued variables.

- Models (binary strings): a Boolean value per variable; equivalently: the set of variables true in it.
- (Definite) Horn Clause: one single positive disjunct, like $\neg a \vee \neg b \vee c$.
- Equivalent form as implication, like $a \wedge b \Rightarrow c$.
- Horn Formula: conjunction of Horn Clauses.

(Definite) Horn Formulas

Definiteness issues glossed over

Propositional world

Boolean-valued variables.

- Models (binary strings): a Boolean value per variable; equivalently: the set of variables true in it.
- (Definite) Horn Clause: one single positive disjunct, like $\neg a \vee \neg b \vee c$.
- Equivalent form as implication, like $a \wedge b \Rightarrow c$.
- Horn Formula: conjunction of Horn Clauses.
- Implications: $(a \wedge b \Rightarrow c) \wedge (a \wedge b \Rightarrow c) \equiv (a \wedge b \Rightarrow c \wedge d)$.

(Definite) Horn Formulas

Definiteness issues glossed over

Propositional world

Boolean-valued variables.

- Models (binary strings): a Boolean value per variable; equivalently: the set of variables true in it.
- (Definite) Horn Clause: one single positive disjunct, like $\neg a \vee \neg b \vee c$.
- Equivalent form as implication, like $a \wedge b \Rightarrow c$.
- Horn Formula: conjunction of Horn Clauses.
- Implications: $(a \wedge b \Rightarrow c) \wedge (a \wedge b \Rightarrow c) \equiv (a \wedge b \Rightarrow c \wedge d)$.

Main Property

A set of models can be axiomatized by a Horn Formula if and only if it is closed under intersection.

Implications, I

A real-life example

Logs from a virtual learning platform

Transactions on propositional variables:

one for each “area” of the course.

`announcements`, `assessments`, `assignments`, `contents`, `forum`, `organizer`, ...

Implications, I
A real-life example

Logs from a virtual learning platform

Transactions on propositional variables:

one for each "area" of the course.

`announcements`, `assessments`, `assignments`, `contents`, `forum`, `organizer`, ...

- Student's sessions are logged;
- for each session, we know whether each "area" was visited in that session;

Implications, I
A real-life example

Logs from a virtual learning platform

Transactions on propositional variables:

one for each "area" of the course.

`announcements`, `assessments`, `assignments`, `contents`, `forum`, `organizer`, ...

- Student's sessions are logged;
- for each session, we know whether each "area" was visited in that session;
- therefore each session is a transaction,
- or, equivalently, a propositional model.

Implications, I

A real-life example

Logs from a virtual learning platform

Transactions on propositional variables:

one for each "area" of the course.

`announcements`, `assessments`, `assignments`, `contents`, `forum`, `organizer`, ...

- Student's sessions are logged;
- for each session, we know whether each "area" was visited in that session;
- therefore each session is a transaction,
- or, equivalently, a propositional model.

Example of an implication:

$\texttt{announcements} \wedge \texttt{assignments} \Rightarrow \texttt{assessments} \wedge \texttt{organizer}$

It is again the conjunction of two Horn clauses.

Implications, II

As a data analysis tool

Implications are a "classic" in the Data Mining field.

Examples from a "Machine Learning abstracts" dataset

descent $\Rightarrow$ gradient
hilbert $\Rightarrow$ space

Implications, II

As a data analysis tool

Implications are a "classic" in the Data Mining field.

Examples from a "Machine Learning abstracts" dataset

descent $\Rightarrow$ gradient

hilbert $\Rightarrow$ space

carlo $\Rightarrow$ monte

monte $\Rightarrow$ carlo

Implications, II

As a data analysis tool

Implications are a “classic” in the Data Mining field.

Examples from a “Machine Learning abstracts” dataset

descent ⇒ gradient
hilbert ⇒ space
carlo ⇒ monte
monte ⇒ carlo

Example from a “census” dataset

Exec-managerial Husband ⇒ Married-civ-spouse

Implications, II

As a data analysis tool

Implications are a “classic” in the Data Mining field.

Examples from a “Machine Learning abstracts” dataset

descent $\Rightarrow$ gradient
hilbert $\Rightarrow$ space
carlo $\Rightarrow$ monte
monte $\Rightarrow$ carlo

Example from a “census” dataset

Exec-managerial Husband $\Rightarrow$ Married-civ-spouse

Algorithms

extract frequent (or frequent closed) sets by either breadth-first search (apriori), or depth-first search (eclat, charm), or other schemes to find association rules in them.

Implications obey the Armstrong inference schemes, originally from functional dependency analysis in Databases:

- Reflexivity: if $Y \subseteq X$, infer $X \Rightarrow Y$;
- Augmentation: from $X \Rightarrow X'$ and $Y \Rightarrow Y'$, infer $XY \Rightarrow X'Y'$;
- Transitivity: from $X \Rightarrow Y$ and $Y \Rightarrow Z$, infer $X \Rightarrow Z$.

Soundness and completeness

Using these schemes, one can infer from a set of implications exactly those implications that become logically entailed by them: any dataset in which the premises are satisfied must satisfy as well the conclusions.

The Logic of Implications, II
Optimal Axiomatizations

Given all the implications that hold for a set of models,

- some of them may be redundant (logically entailed);
- taking these out would give an irredundant basis;
- but there may be various ways to reach irredundant bases,
- and they may be of very different sizes.

The Logic of Implications, II
Optimal Axiomatizations

Given all the implications that hold for a set of models,

- some of them may be redundant (logically entailed);
- taking these out would give an irredundant basis;
- but there may be various ways to reach irredundant bases,
- and they may be of very different sizes.

Minimum-size axiomatizations: the Guigues-Duquenne basis

- a canonical, minimum-size basis for implications;
- equivalent notion in functional dependencies;
- the Horn Query Learning algorithm AFP constructs it.

Towards Standard Association Rules, I
A relaxed notion of "correctness"

There are reasons to be satisfied with an implication even in the presence of counterexamples.

- Transmission or keying errors;
- mistakes in filling up forms;
- mixed populations;
- ...

Partial "approximate" implications that allow for exceptions.

Towards Standard Association Rules, II

Confidence-based framework

"Census" dataset:

Some facts found:

- Husband $\Rightarrow$ Male

Towards Standard Association Rules, II

Confidence-based framework

"Census" dataset:

Some facts found:

- Husband $\Rightarrow$ Male... does not hold! (see tuple 7110)

Towards Standard Association Rules, II

Confidence-based framework

"Census" dataset:

Some facts found:

- Husband $\Rightarrow$ Male... does not hold! (see tuple 7110)
- Similarly, Wife $\Rightarrow$ Female does not hold either: there are two tuples declaring Male and Wife.
- Consequence: over sixty full-confidence rules of the form Husband, SomethingElse $\Rightarrow$ Male.
- Confidence (and support) thresholds seem insufficient!

The Danger Of Absolute Confidence Thresholds

But, how to convince everyone else?

Dataset CMC (Contraceptive Method Choice)

A rule of over 10% support and 90% confidence:

`near-low-wife-education   no-contraception-method`

$\longrightarrow$

`good-media-exposure`

The Danger Of Absolute Confidence Thresholds

But, how to convince everyone else?

Dataset CMC (Contraceptive Method Choice)

A rule of over 10% support and 90% confidence:

`near-low-wife-education` `no-contraception-method`

$\longrightarrow$

`good-media-exposure`

But the support of "`good-media-exposure`" is over 92%.

The Danger Of Absolute Confidence Thresholds

But, how to convince everyone else?

Dataset CMC (Contraceptive Method Choice)

A rule of over 10% support and 90% confidence:

`near-low-wife-education` `no-contraception-method`

$\longrightarrow$

`good-media-exposure`

But the support of "`good-media-exposure`" is over 92%.

- The most natural normalization to avoid this problem (deviation from independence, also called lift) is symmetric.
- Many alternative definitions of $X \rightarrow Y$, almost all on the basis of the supports of X, Y, XY, and $X \cap Y$.
- Rich and complex landscape, leading to an "axiomatic" study of all these alternatives.

Redundancy in Association Rules, I

A Logic-based view

Standard Association Mining Process

User provides dataset and thresholds for support and confidence, and gets all rules that hold in the dataset at those levels or higher.

Huge set of rules, growing further for lower thresholds. How to offer the user a smallish set of output rules?

- Our (rather obvious) proposal of "plain" redundancy: $X \to Y$ is redundant with respect to $X' \to Y'$ if $\mathrm{conf}(X \to Y) \geq \mathrm{conf}(X' \to Y')$ in every dataset.
- A natural variant, closure-based redundancy, reads the same, but under a condition to share the same closure space.
- That variant offers a way to treat separately implications from partial rules; implications "sneak in" anyway, and they allow better summarization through the GD basis.

Redundancy in Association Rules, II

Minimum-Size Bases

Basic antecedent X of Y (with $X \subseteq Y$):

- work only among closures: both X and Y must be closed;
- "representative rules" variant: X not necessarily closed;
- confidence of $X \to Y$ must be at least γ;
- but falls below γ if either we enlarge Y, or we reduce X.

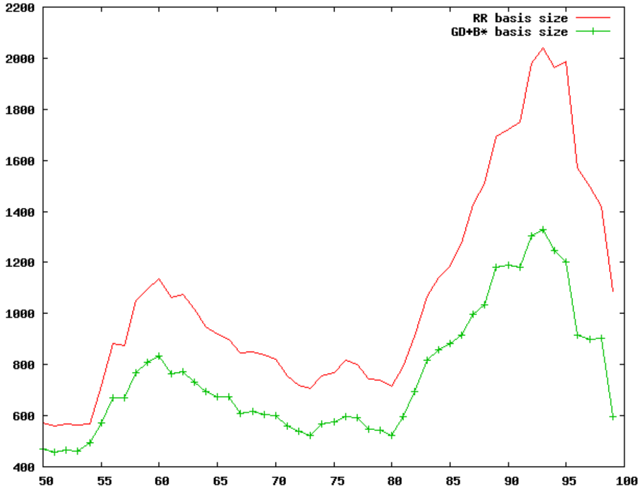
Basis $\mathcal{B}^*$: $X \to Y - X$ for all closed Y and all basic antecedents X of Y, provided $Y - X \neq \emptyset$.

Facts

1. These rules hold with confidence γ,
2. All the rules that hold with confidence γ can be inferred from these rules plus the implications, and
3. Any alternative set of rules with the same properties has at least as many rules as this one.

Irredundant Rules for Dataset FIMI pumsb-star

In a couple of alternative formulations



Quantifying absolute novelty

Confidence Width: one of four related notions

For a given support and confidence thresholds

Assume we have run an "association miner":

- Discard redundant rules: we are left just with the basis.
- Discarded rules are entailed by the basis.
- Each rule left, say R, is not entailed by the others.
- This means that the other rules would not suggest that R passes the confidence threshold, say γ.

Quantifying absolute novelty

Confidence Width: one of four related notions

For a given support and confidence thresholds

Assume we have run an "association miner":

- Discard redundant rules: we are left just with the basis.
- Discarded rules are entailed by the basis.
- Each rule left, say R, is not entailed by the others.
- This means that the other rules would not suggest that R passes the confidence threshold, say γ.
- But maybe R becomes redundant at a lower confidence!

For a given support and confidence thresholds

Assume we have run an "association miner":

- Discard redundant rules: we are left just with the basis.
- Discarded rules are entailed by the basis.
- Each rule left, say R, is not entailed by the others.
- This means that the other rules would not suggest that R passes the confidence threshold, say γ.
- But maybe R becomes redundant at a lower confidence!
- Let γ' be the tightest confidence at which R is redundant, and let's consider the quotient γ/γ'.

Low Novelty

Novel, but barely

Suppose:

- The confidence of R is γ.
- Other rules of confidence γ do not entail it.
- Thus, it is irredundant with respect to the rest of the rules found at confidence γ.
- But, if we had run the process at a confidence slightly lower, say $\gamma' < \gamma$, maybe some R' would have been found that entails R.

R only belongs to the basis during the short interval $(\gamma', \gamma]$ of values for the confidence threshold; γ/γ' is low.

At its own confidence, it is novel, but really not too much.

Low Novelty

Novel, but barely

Suppose:

- The confidence of R is γ.
- Other rules of confidence γ do not entail it.
- Thus, it is irredundant with respect to the rest of the rules found at confidence γ.
- But, if we had run the process at a confidence slightly lower, say $\gamma' < \gamma$, maybe some R' would have been found that entails R.

R only belongs to the basis during the short interval $(\gamma', \gamma]$ of values for the confidence threshold; γ/γ' is low.

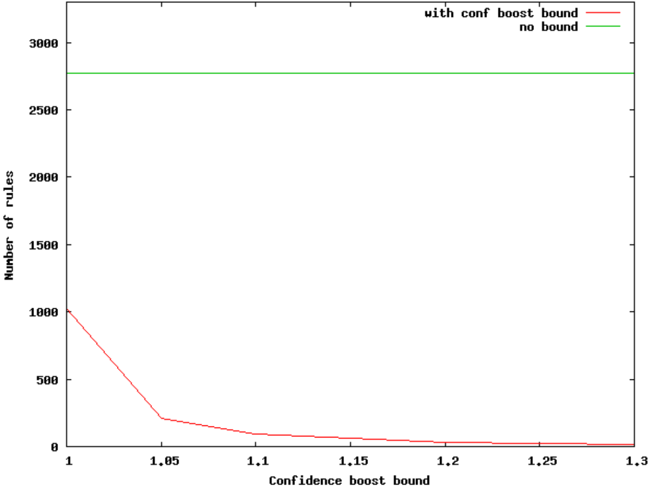
At its own confidence, it is novel, but really not too much.

Conversely, if any such γ' is considerably lower, R states novel information: we only can make it redundant with rules of much lower confidence, and γ/γ' is high.

Confidence Boost

A somewhat more sophisticate variant of confidence width

Counts of association rules from the Adult dataset again, mined at 2.5% support and 75% confidence, with or without a confidence boost bound.



Index

Bias and Variance: Intuition

Two sources of prediction error

Variance:

Risk arising from the data.

- Data is seen as a sample;
- different samples may lead to different predictions;
- one cannot rule out the risk that the sample is a particularly bad one, just due to sheer bad luck;
- it is modeled by variance in the good old statistics sense.

Bias and Variance: Intuition

Two sources of prediction error

Variance:

Risk arising from the data.

- Data is seen as a sample;
- different samples may lead to different predictions;
- one cannot rule out the risk that the sample is a particularly bad one, just due to sheer bad luck;
- it is modeled by variance in the good old statistics sense.

Bias:

Risk arising from your family of hypotheses.

- In a poor family of hypothesis,

  even the best one might not be very good.

Bias and Variance: Formalization

Just a tiny bit

Context:

A prediction task, say regression.

- Value we want to predict, y;
- Sample s which reveals some information about y;
- Estimator $e(s)$ that tries to pinpoint y after seeing s.

Bias and Variance: Formalization

Just a tiny bit

Context:

A prediction task, say regression.

- Value we want to predict, y;
- Sample s which reveals some information about y;
- Estimator $e(s)$ that tries to pinpoint y after seeing s.

Variance:

Quadratic average error of $e(s)$ used as estimator of its own average $E[e(s)]$: $E[(e(s) - E[e(s)])^2]$.

Bias and Variance: Formalization

Just a tiny bit

Context:

A prediction task, say regression.

- Value we want to predict, y;
- Sample s which reveals some information about y;
- Estimator $e(s)$ that tries to pinpoint y after seeing s.

Variance:

Quadratic average error of $e(s)$ used as estimator of its own average $E[e(s)]$: $E[(e(s) - E[e(s)])^2]$.

Bias:

Absolute expected error of e with respect to the true target: $|E[e(s)] - y|$. (It is independent of s.)

Note the different "scale":
we will square the bias to compensate for this.

Error Descomposition

Error is made of bias and variance

Let's add up variance and bias squared:

$$
\begin{aligned}
&E[(e(s) - E[e(s)])^2] + \\
&\qquad (E[e(s)] - y)^2 = \\
&E\left[e(s)^2 - 2E[e(s)]e(s) + E[e(s)]^2\right] + \\
&\qquad E[e(s)]^2 - 2yE[e(s)] + y^2 = \\
&E[e(s)^2] - 2E[e(s)]E[e(s)] + E[e(s)]^2 + \\
&\qquad E[e(s)]^2 - 2yE[e(s)] + y^2 = \\
&E[e(s)^2] - 2yE[e(s)] + y^2 = \\
&E[e(s)^2] - E[2y\, e(s)] + E[y^2] = E[e(s)^2 - 2y\, e(s) + y^2] = \\
&E[(e(s) - y)^2]
\end{aligned}
$$

Error Descomposition

Error is made of bias and variance

Let's add up variance and bias squared:

$$
\begin{aligned}
&E[(e(s) - E[e(s)])^2] + \\
&\qquad\qquad (E[e(s)] - y)^2 = \\
&E\left[e(s)^2 - 2E[e(s)]e(s) + E[e(s)]^2\right] + \\
&\qquad\qquad E[e(s)]^2 - 2yE[e(s)] + y^2 = \\
&E[e(s)^2] - 2E[e(s)]E[e(s)] + E[e(s)]^2 + \\
&\qquad\qquad E[e(s)]^2 - 2yE[e(s)] + y^2 = \\
&E[e(s)^2] - 2yE[e(s)] + y^2 = \\
&E[e(s)^2] - E[2y\, e(s)] + E[y^2] = E[e(s)^2 - 2y\, e(s) + y^2] = \\
&E[(e(s) - y)^2]
\end{aligned}
$$

They add up to the average quadratic error!

The consequence is that it becomes difficult to "tune manually" the flexibility of our inductive bias (the hypothesis class).

Index

Uncertainty Sources

What is really the relationship between data and prediction?

Prediction:

On the basis of the given data, but:

- the data might truly determine the value to be predicted, just that we don't know in what way;

Uncertainty Sources

What is really the relationship between data and prediction?

Prediction:

On the basis of the given data, but:

- the data might truly determine the value to be predicted, just that we don't know in what way;
- the data might not determine the value to be predicted: case of extra, inaccessible latent variables;

Uncertainty Sources

What is really the relationship between data and prediction?

Prediction:

On the basis of the given data, but:

- the data might truly determine the value to be predicted, just that we don't know in what way;
- the data might not determine the value to be predicted: case of extra, inaccessible latent variables;
- the data might not repeat exactly (e.g. reals).

Option: explicit or implicit continuity assumption.

For instance: Naïve Bayes on floats?
Replace the discrete conditional distribution by "something else".
(Usually a Gaussian — lab later).

Probability and Likelihood

Two sides of the same coin

Leads to a “heavily parametric” point of view.

Write a function thus:

$$(m, d, \vec{y}) \mapsto \prod_i \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}(\frac{y_i - m}{d})^2}$$

Probability and Likelihood

Two sides of the same coin

Leads to a “heavily parametric” point of view.

Write a function thus:

$$(m, d, \vec{y}) \mapsto \prod_i \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y_i - m}{d}\right)^2}$$

1. If you know the parameters, what is the probability?

$$\Pr_{N(m,d)}(\vec{y}) = \prod_i \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y_i - m}{d}\right)^2}$$

Probability and Likelihood
Two sides of the same coin

Leads to a "heavily parametric" point of view.

Write a function thus:

$$(m, d, \vec{y}) \mapsto \prod_i \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y_i - m}{d}\right)^2}$$

1. If you know the parameters, what is the probability?

$$\mathrm{Pr}_{N(m,d)}(\vec{y}) = \prod_i \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y_i - m}{d}\right)^2}$$

2. If you know instead the data, what can you say about the likelihood of the parameters? (Does it look like $N(m, d)$ for "these" m and d?):

$$\mathcal{L}(m, d|\vec{y}) = \prod_i \frac{1}{d\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y_i - m}{d}\right)^2}$$

Nearest Neighbors

The data is the model

Assumption:

Similar observations lead to similar responses.

- Keep all the data in an appropriate data structure;
- Predict the most common response among the k nearest neighbors of a new observation to predict on.

Nearest Neighbors
The data is the model

Assumption:

Similar observations lead to similar responses.

- Keep all the data in an appropriate data structure;
- Predict the most common response among the k nearest neighbors of a new observation to predict on.
- Often, the continuity assumption is correct.
- Often, it is not.
- In high dimensions, finding out the k nearest neighbors is computationally nontrivial.

Decision Trees

Can you imagine explaining your NB predictor to your boss?

Can we make do by checking a single attribute?

If not. . .

Decision Trees

Can you imagine explaining your NB predictor to your boss?

Can we make do by checking a single attribute?

If not. . . recurse!

Can we make do by checking a single attribute?

If not. . . recurse!

- Measure somehow the "heterogeneity" of the observations, and
- Pick one "test" of the value of an attribute so that the split reduces the "joint heterogeneity".

Decision Trees

Can you imagine explaining your NB predictor to your boss?

Can we make do by checking a single attribute?

If not. . . recurse!

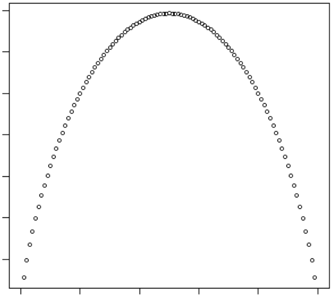
- Measure somehow the "heterogeneity" of the observations, and
- Pick one "test" of the value of an attribute so that the split reduces the "joint heterogeneity".

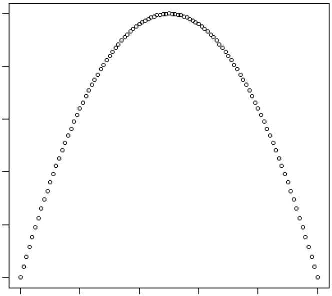
Several variants of this idea (ID3, C4.5, C5.0, CART):

- the prediction follows a decomposition of the input space in "axis-parallel cuboids", but
- "tests" can be made in different ways, and
- there are several possible notions of "heterogeneity".

Heterogeneity

Shannon information versus Gini index (2-valued case)





Classifier Border Repertory

Class-separaton shapes

- Decision Stumps:
  - axis-parallel hyperplanes,
- Decision Trees:
  - unions thereof,
- kNN, NB:
  - complex shapes...

Classifier Border Repertory

Class-separaton shapes

- Decision Stumps:
  - axis-parallel hyperplanes,
- Decision Trees:
  - unions thereof,
- kNN, NB:
  - complex shapes...
- Linear predictors:
  - Separating hyperplanes

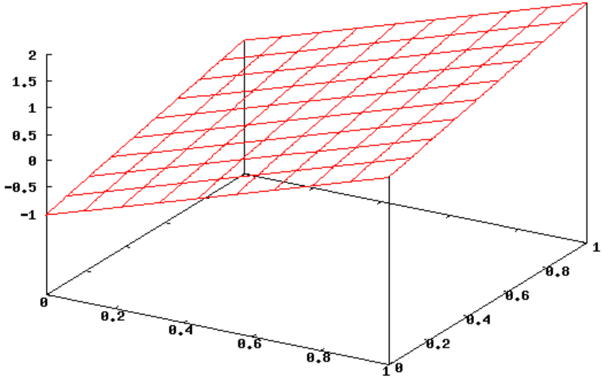
Classifier Border Repertory

Class-separaton shapes

- Decision Stumps:
  - axis-parallel hyperplanes,
- Decision Trees:
  - unions thereof,
- kNN, NB:
  - complex shapes...
- Linear predictors:
  - Separating hyperplanes (not necessarily in the same space!)
    - Hard threshold,
    - Soft threshold

A linear separator

In R^3: $2x + y - 1$



Slogan: maximal margin; don't get closer to any of the classes more than absolutely necessary.

Optimization rendering:

Maximize m, under the constraints: $y_i \frac{(w^T x + b)}{||w||} \geq m$.

(Plus a funny trick on the scaling!)

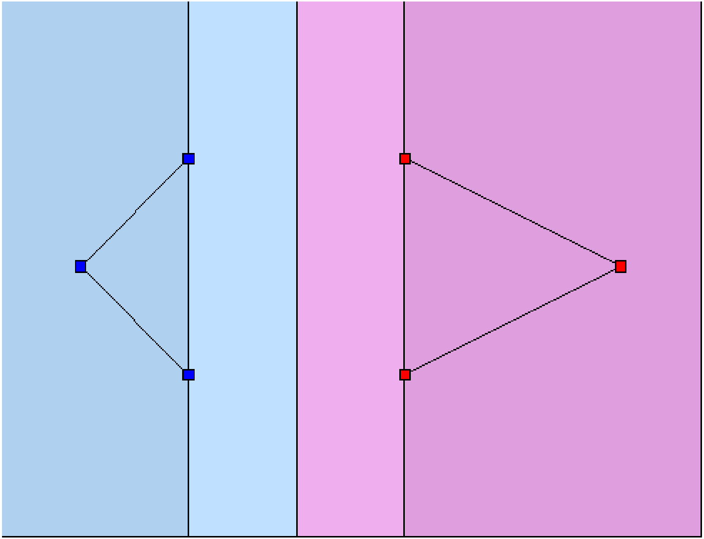
- Hard margin,
- Soft margin.

Karush/Kuhn/Tucker conditions

from optimization theory tell us valuable info about our predictors.

Maximal-Margin Hyperplane

Intuition of convex hulls fully correct and useful



Reproducing Kernel Hilbert Spaces:

Can be obtained through scalar products.

A two-dimensional conic:

$$w_1 x_1^2 + w_2 x_2^2 + w_s x_1 x_2 + w_4 x_1 + w_5 x_2 + w_6$$

is the scalar product of the weights $(w_1, w_2, w_3, w_4, w_5, w_6)$ with a "transformed" input point (x_1, x_2) a R^6:

$$f(x_1, x_2) = (x_1^2, x_2^2, x_1 x_2, x_1, x_2, 1)$$

(Please compute $((x_1, x_2)(y_1, y_2) + 1)^2$.)

AdaBoost, I

Intent on Improving

One of the two eponyms of ensemble methods

(The other one being *bagging*).
Both predictors and observations are weighted.

- Construct a weak, simple predictor,
- assign a weight to it,
- update the weights of the observations by increasing it on the mistakes,
- repeat while the process actually improves.

AdaBoost, II
Some details

For a new predictor h:

- find its weighted error ϵ

  (adding up the weights of the "mistake" observations)
- if $\epsilon \geq \frac{1}{2}$, discard h and stop;
- reweigh the observations:
  - $d = \frac{1-\epsilon}{\epsilon}$;
  - multiply by d the weight of correctly predicted observations,
  - divide by d the weight of incorrectly predicted observations,
- assign to h weight $\log d$.

Error bound:

$$e^{-2\sum(\frac{1}{2}-\epsilon_t)^2}$$

Index

Conclusions

A field full of beautiful and useful ideas

The design and understanding of methods to extract descriptive or predictive models from data is a difficult but fascinating task.

Conclusions

A field full of beautiful and useful ideas

The design and understanding of methods to extract descriptive or predictive models from data is a difficult but fascinating task.

Main message:

If you join in, try to understand beyond where ideas work.

- Make as many assumptions as you need to do useful work,

  as no data analysis process will conceivably work without an inductive bias,

Conclusions

A field full of beautiful and useful ideas

The design and understanding of methods to extract descriptive or predictive models from data is a difficult but fascinating task.

Main message:

If you join in, try to understand beyond where ideas work.

- Make as many assumptions as you need to do useful work,

  as no data analysis process will conceivably work without an inductive bias,

- but try to be aware of which ones are actually at work,

  as sometimes data analysis processes are assuming further than explicitly assumed,

Conclusions

A field full of beautiful and useful ideas

The design and understanding of methods to extract descriptive or predictive models from data is a difficult but fascinating task.

Main message:

If you join in, try to understand beyond where ideas work.

- Make as many assumptions as you need to do useful work,

  as no data analysis process will conceivably work without an inductive bias,
- but try to be aware of which ones are actually at work,

  as sometimes data analysis processes are assuming further than explicitly assumed,
- and be permanently ready to challenge each and every one of them.

Index

Lab Session
Let's stop talking and start doing

Activities:

Some of them wide open...

- Online demos of algorithms, simplified:
  - K-Means 2D,
  - K-Means "3D",
  - Various predictors 2D.
- Dataset formats.
- Interactive decision trees on Orange; workflows.
- Naïve Bayes alone; Naïve Bayes versus MAP: ROC curves.
  - (One example of something not to do!)
- Brief demo of workflow-based Data Mining open source tools.
- Association rules: state of the art and an experimental new system.

Click Places, then Downloads

Two files

Unzip the two zipped files in different folders.
(Feel free to copy the zipped files to your pendrive.)
Knime can be called by double-clicking on the icon of the binary.
The other folder contains

- a Windows version of Knime,
- example datasets to play with,
- simple python implementations of ROC curves from NB and MAP,
- the *yacaree* association rule miner,
- a book on statistical learning,
- these slides,
- and some further materials that could be useful.

Online Demos of Algorithms
Simplified

(Copy and paste from file onlineDemoLinks.txt)

Links to some demo webpages

(You do not need to follow all of them.)

- K-means 2D (two of them, there are quite a few more):
  - http://home.dei.polimi.it/matteucc/ ...
  - http://www.paused21.net/off/kmeans/bin/
- K-means 3D in RGB space:
  - http://www.leet.it/home/lale/ ...
- Quadratic kernel idea:
  - http://www.youtube.com/watch?v=3liCbRZPrZA (search for svm on youtube)
- LibSVM (including toy):
  - http://www.csie.ntu.edu.tw/~cjlin/libsvm/
- Locboost demo of 2D predictors on top of Weka:
  - http://www.cs.technion.ac.il/~rani/LocBoost/

Dataset Formats

Relational or Transactional

1. Compare files weather.*
2. Compare files weather.nominal.*
3. Compare files eprints.*
4. Compare files haireyescolor.*
5. Compare files basket*.txt

Open Source Tools
And decision tree demo on Orange

See file toolList.

We demonstrate the Interactive Decision Tree to illustrate the notion of data mining workflow, pioneered by the commercial tool Clementine (now IBM's SPSS).

NB versus MAP

A negative howto

We open program roc.py and adjust the predictor, the dataset file, and the value of the class attribute.

Re-do on yourself variants of the exercises we made so far,
and/or
pick one or more example datasets and explore them with Knime at your leisure.
You can also try yacaree on transactional datasets.
Keep calling me when in doubt.