

Data Mining 2018

Introduction

Ad Feelders

Universiteit Utrecht

The Course

- Literature: Lecture Notes, Articles, Book Chapters, Slides (appear on the course web site).
- Course Form: Lectures (Wednesday, Friday) and computer lab sessions with R (Wednesday).
- Grading: two practical assignments (50%) and a written exam (50%).
- Web Site: <http://www.cs.uu.nl/docs/vakken/mdm/>

Practical Assignments

There are two practical assignments: one assignment with emphasis on programming and one with emphasis on data analysis.

- 1 Write your own classification tree and random forest algorithm in R, and apply the algorithm to a bug prediction problem (30%).
- 2 Text Mining: analyze hotel reviews to distinguish genuine from fake reviews (20%).

Assignments should be completed by teams of 2 students.

What is Data Mining?

- (Knowledge discovery in databases) is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data (Fayyad et al.)
- Analysis of *secondary* data (Hand)
- The induction of understandable models and patterns from databases (Siebes)
- The *data-dependent* process of selecting a statistical model (Leamer, 1978 (!))

What is Data Mining?

Data Mining as a subdiscipline of computer science:

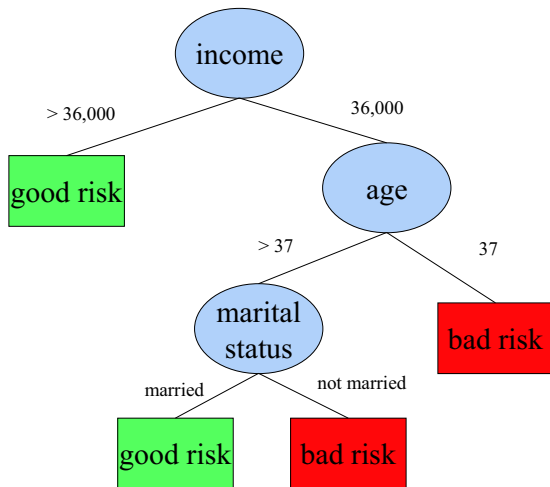
is concerned with the development and analysis of algorithms for the (efficient) extraction of patterns and models from (large, heterogeneous, ...) data bases.

A model is an abstraction of (part of) reality (the application domain).

In our case, models describe relationships among:

- attributes (variables, features),
- tuples (records, cases),
- or both.

Example Model: Classification Tree



Patterns are **local models**, i.e., models that describe only part of the database.

For example, association rules:

$$\textit{Diapers} \rightarrow \textit{Beer}, \textit{support} = 20\%, \textit{confidence} = 85\%$$

Although patterns are clearly different from models, we will use *model* as the generic term.

Diapers → Beer



Reasons to Model

A model

- helps to gain *insight* into the application domain
- can be used to make *predictions*
- can be used for *manipulating/controlling* a system (causality!)

A model that predicts well does not always provide understanding.

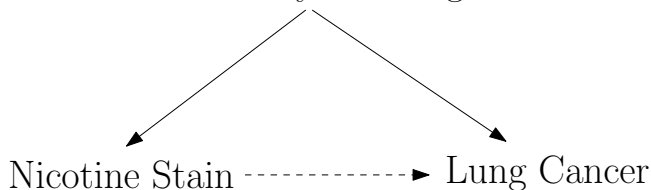
Correlation \neq Causation

Can causal relations be found from data alone?

Causality and Correlation



Heavy Smoking



Washing your hands or cleaning your teeth won't help!

Induction vs Deduction

Deductive reasoning is *truth-preserving*:

- ① All horses are mammals
- ② All mammals have lungs
- ③ Therefore, all horses have lungs

Inductive reasoning *adds information*:

- ① All horses observed so far have lungs
- ② Therefore, all horses have lungs

Induction (Statistical)

- 1 4% of the products we tested are defective
- 2 Therefore, 4% of all products (tested or otherwise) are defective

Inductive vs Deductive: Acceptance Testing Example

100,000 products, sample 1000

Suppose 10 are defective (1% of the sample)

Deductive: $d \in [0.0001, 0.9901]$

Inductive: $d \in [0.004, 0.016]$ with 95% confidence.

$$0.01 \pm \underbrace{\sqrt{\frac{0.01 \times 0.99}{1000}} \times 1.96}_{\approx 0.006}$$

Experimental data

The experimental method:

- Formulate a hypothesis of interest.
- Design an experiment that will yield data to test this hypothesis.
- Accept or reject hypothesis depending on the outcome.

Experimental vs Observational Data

Experimental Scientist:

- Assign level of fertilizer randomly to plot of land.
- Control for other factors that might influence yield: quality of soil, amount of sunlight,...
- Compare mean yield of fertilized and unfertilized plots.

Data Miner:

- Notices that the yield is somewhat higher under trees where birds roost.
- Conclusion: bird droppings increase yield.
- Alternative conclusion: moderate amount of shade increases yield. ("Identification Problem")

Observational Data

- In observational data, many variables may move together in systematic ways.
- In this case, there is no guarantee that the data will be “rich in information”, nor that it will be possible to isolate the relationship or parameter of interest.
- Prediction quality may still be good!

Example: linear regression

$$\widehat{\text{mpg}} = a + b \times \text{cyl} + c \times \text{eng} + d \times \text{hp} + e \times \text{wgt}$$

Estimate a, b, c, d, e from data. Choose values so that sum of squared errors

$$\sum_{i=1}^N (\text{mpg}_i - \widehat{\text{mpg}}_i)^2$$

is minimized.

$$\frac{\partial \widehat{\text{mpg}}}{\partial \text{eng}} = c$$

Expected change in mpg when (all else equal) engine displacement increases by one unit.

Engine displacement is defined as the total volume of air/fuel mixture an engine can draw in during one complete engine cycle.

The Data

```
> cars.dat[1:10,]  
   mpg  cyl eng  hp  wgt  
1   18   8 307 130 3504 "chevrolet chevelle malibu"  
2   15   8 350 165 3693 "buick skylark 320"  
3   18   8 318 150 3436 "plymouth satellite"  
4   16   8 304 150 3433 "amc rebel sst"  
5   17   8 302 140 3449 "ford torino"  
6   15   8 429 198 4341 "ford galaxie 500"  
7   14   8 454 220 4354 "chevrolet impala"  
8   14   8 440 215 4312 "plymouth fury iii"  
9   14   8 455 225 4425 "pontiac catalina"  
10  15   8 390 190 3850 "amc ambassador dpl"
```

Fitted Model

Coefficients:

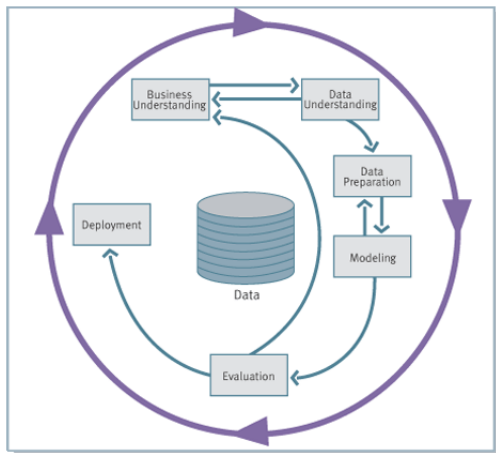
```
              Estimate Pr(>|t|)
(Intercept) 45.7567705 < 2e-16 ***
cyl          -0.3932854 0.337513
eng           0.0001389 0.987709
hp           -0.0428125 0.000963 ***
wgt          -0.0052772 1.08e-12 ***
---
```

Multiple R-Squared: 0.7077

```
> cor(cars.dat)
```

	mpg	cyl	eng	hp	wgt
mpg	1.0000000	-0.7776175	-0.8051269	-0.7784268	-0.8322442
cyl	-0.7776175	1.0000000	0.9508233	0.8429834	0.8975273
eng	-0.8051269	0.9508233	1.0000000	0.8972570	0.9329944
hp	-0.7784268	0.8429834	0.8972570	1.0000000	0.8645377
wgt	-0.8322442	0.8975273	0.9329944	0.8645377	1.0000000

KDD Process: CRISP-DM



This course is mainly concerned with the modeling phase.

Data Preparation

Once we know the data available to solve the data mining problem, we have to prepare the mining database (often one table):

- Select data
- Clean data
- Construct data
- Integrate data

Select Data: Why Reject Data?

- attributes with too many errors
- attributes with too many missing values
- attributes that have no relevance from a domain experts point of view
- a sample may be sufficient

Cleaning data is a complete topic in itself, we mention two problems:

- ① data editing: what to do when records contain *impossible* combinations of values?
- ② incomplete data: what to do with missing values?

Data Editing: Example

We have the following edits (impossible combinations):

$$E_1 = \{Driver's Licence=yes, Age < 18\}$$

$$E_2 = \{Married=yes, Age < 18\}$$

Make the record:

$$Driver's Licence=yes, Married=yes, Age=15$$

consistent by changing attribute values. What change(s) would you make?

Of course it's better to *prevent* such inconsistencies in the data!

Seminal Paper: I.P. Fellegi, D. Holt: A systematic approach to automatic edit and imputation, Journal of the American Statistical Association 71(353), 1976, pp. 17-35.

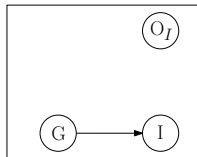
What to do with missing values?

- One can remove a tuple if one or more attribute values are missing. Danger: how representative and random is the remaining sample? Also, you may have to throw away a large part of the data!
- One can remove attributes for which values are missing. Danger: this attribute may play an important role in the model you want to induce.
- You do *imputation*, i.e., you fill in a value. Danger: the values you guess may have a large influence on the resulting model.

Missing Data Mechanisms: MCAR

Suppose we have data on gender and income.
Gender is fully observed, income is sometimes missing.

MCAR: Income is missing completely at random.



For example: $\Pr(\text{income} = ?) = 0.1$

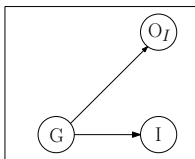
- There will be no bias if we remove tuples with missing income.
- Imputation:
 - If person is male, pick a random male with income observed and fill in his value.
 - If person is female, pick a random female with income observed and fill in her value.

Missing Data Mechanisms: MAR

Probability that income is missing depends on gender. For example:

$$\Pr(\text{income} = ? | \text{gender} = \text{male}) = 0.2$$

$$\Pr(\text{income} = ? | \text{gender} = \text{female}) = 0.05$$



- This time there *will* be bias if we remove tuples with missing income. What kind of bias?
- Imputation: same as before, still works.

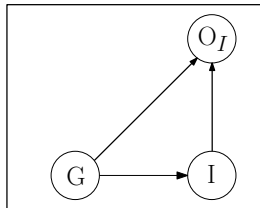
Missing Data Mechanisms: MNAR

Probability that income is missing depends on value of income itself.

$$\Pr(\text{income} = ? | \text{income} > 8000) = 0.4$$

$$\Pr(\text{income} = ? | 2000 < \text{income} < 8000) = 0.01$$

$$\Pr(\text{income} = ? | \text{income} < 2000) = 0.2$$



- We can't “repair” this unless we have knowledge about the missing data mechanism.

Missing Data

- Unfortunately, we cannot infer *from the observed data alone* whether or not the missing data mechanism is MAR or MNAR.
- We might have knowledge about the nature of the missing data mechanism however ...
- Practice: if you don't know, assume MAR and hope for the best.

Construct Data

Quite often, the raw data is not in the proper format for analysis, for example:

- You have dates of birth and you suspect that age plays a role.
- You have data on income and fixed expenses and you think disposable income is important.
- You have to analyze text data, for example the body of e-mail messages to develop a spam filter.

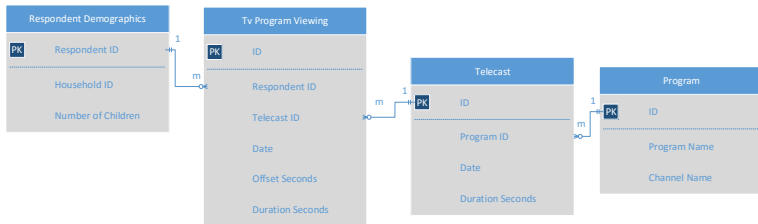
You could represent the text as a *bag-of-words*.

Integrate Data: The Mining Table

- Integration of data from different databases (data fusion), for example:
 - the 'same' attribute has two different names.
 - two attributes with the same name have a different meaning.
 - the 'same' attribute has two different domains.
- One-one relationships between tables are easy, but what to do with 1-n relationships?

One to Many Relationships: TV Viewing

Predict household composition from TV viewing behavior.



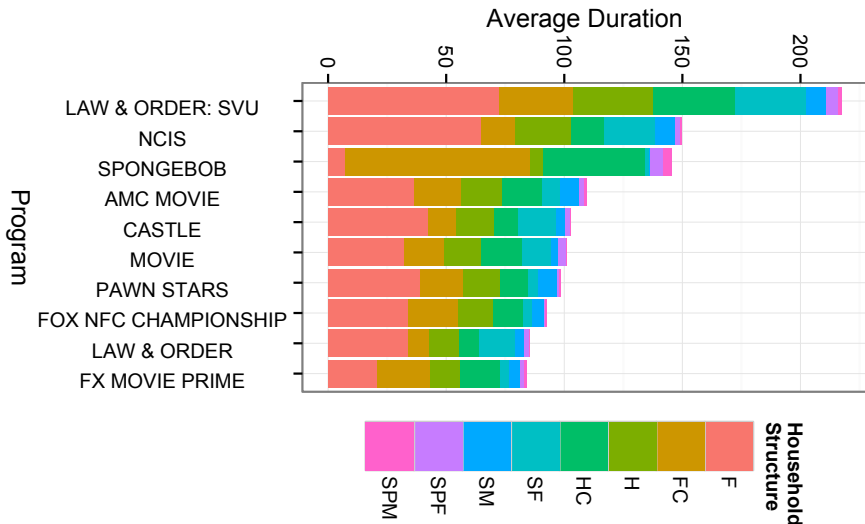
Aggregating the data

Viewing behaviour has to be aggregated, for example:

- Weekly viewing frequency of different programs.
- Weekly viewing duration of different programs.
- Weekly viewing frequency of different program categories.
- etc.

Potentially results in a huge number of attributes.

Some descriptive statistics

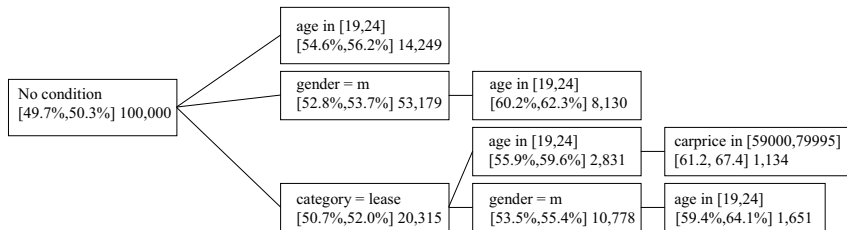


Modeling: Data Mining Tasks

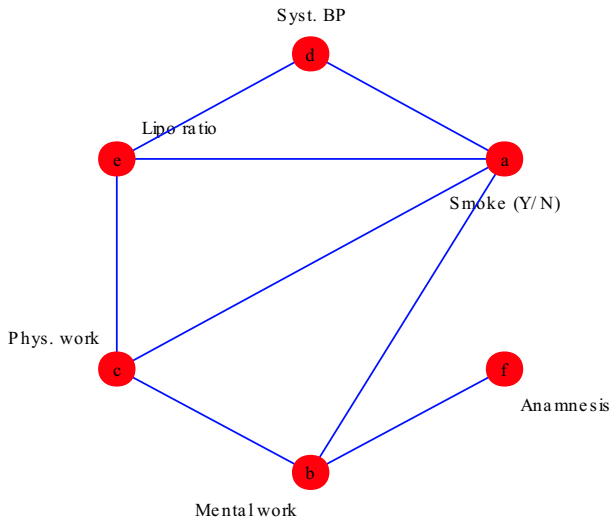
- Classification / Regression
- Dependency Modeling (Graphical Models; Bayesian Networks)
- Frequent Pattern Mining (Association Rules)
- Subgroup Discovery (Rule Induction; *Bump-hunting*)
- Clustering
- Ranking

Subgroup Discovery

Find groups of objects (persons, households, transactions, ...) that score relatively high (low) on a particular *target* attribute.

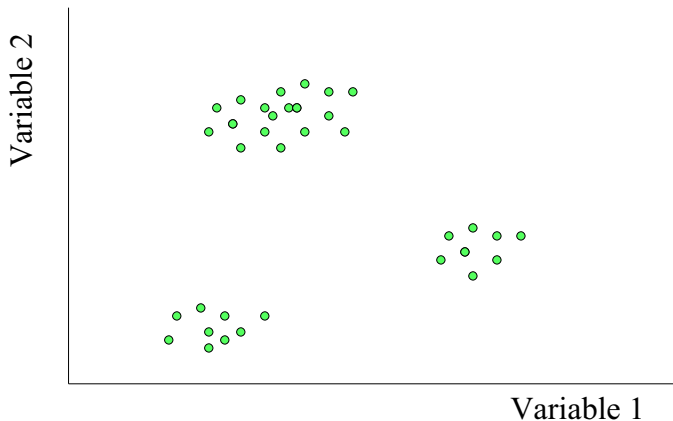


Dependency Modeling: Coronary Heart Disease



Clustering

Put objects (persons, households, transactions, ...) into a number of groups in such a way that the objects within the same group are similar, but the groups are dissimilar.



Ranking

For example:

- Rank web pages with respect to their relevance to a query.
- Rank job applicants with respect to their suitability for the job.
- Rank loan applicants with respect to default risk.
- ...

Has similarities with regression and classification, but in ranking we are often only interested in the *order* of objects.

Components of DM algorithms

Data Mining Algorithms can often be regarded as consisting of the following components:

- ① A representation language: what models are we looking for?
- ② A quality function: when do we consider a model to be good?
- ③ A search algorithm: how do we find some or all good or best models?

Representation Languages

Representation languages define the set of all possible models, e.g.,

- linear models: $y = b_0 + b_1x_1 + \dots + b_nx_n$
- association rules: $X \rightarrow Y$
- subgroups: $X_1 \in V_1 \wedge \dots \wedge X_n \in V_n$
- classification trees
- Bayesian networks (DAGs)

Quality Functions

The quality score of a model often contains two elements:

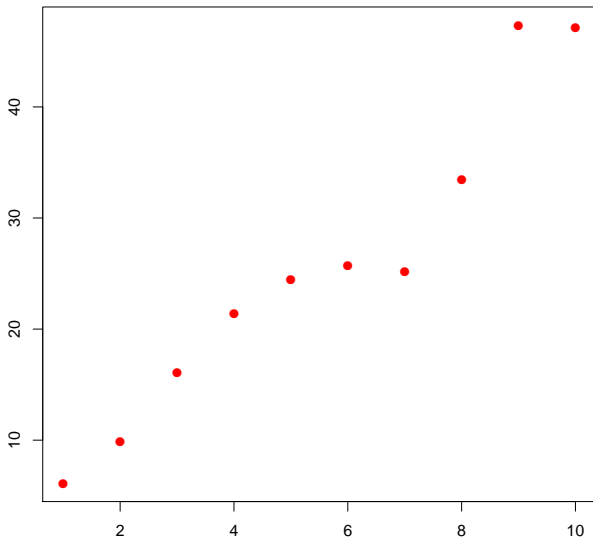
- How well does the model fit the data?
- How complex is the model?

For example (regression)

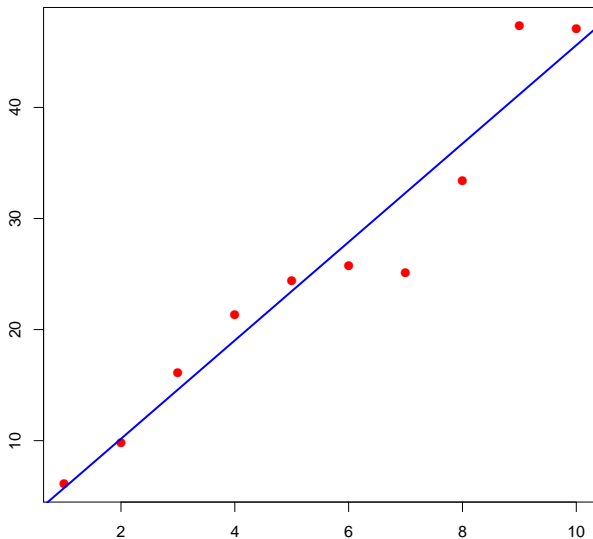
$$\text{score} = \sum_{i=1}^N (y_i - \hat{y}_i)^2 + 2 \times \# \text{ parameters}$$

If independent test data is used, the quality score usually only considers the fit on the test data.

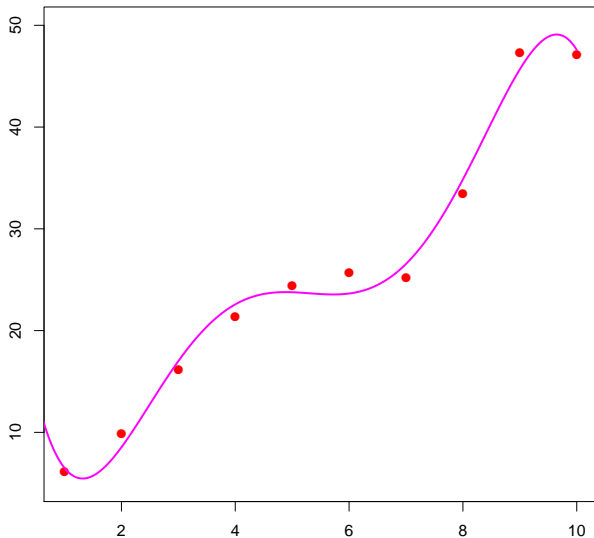
The training data



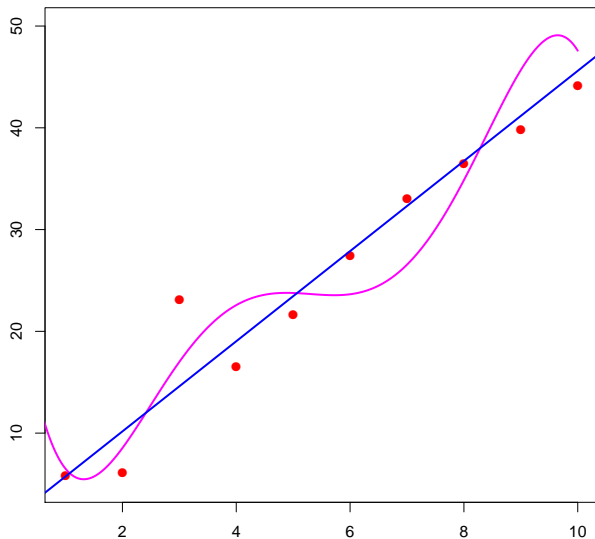
Fitting a linear model to the training data



A degree 5 polynomial fits the training data better!



Over-fitting: linear model generalizes better to new data



Search for the good models

- Sometimes we can check all possible models, because there are rules with which to prune large parts of the search space; for example, the “a priori principle” in frequent pattern mining.
- Usually we have to employ *heuristics*
 - A general search strategy, such as a hill-climber or a genetic (evolutionary) algorithm.
 - Search operators that implement the search strategy on the representation language. Such as, a neighbour operator for hill climbing and cross-over and mutation operators for genetic search.

Search: example

In linear regression, we want to predict a numeric variable y from a set of predictors x_1, \dots, x_n . We might include any subset of predictors, so the search space contains 2^n models. E.g., if $n = 30$, we have $2^{30} = 1,073,741,824$, i.e. about one billion models in the search space.

It is common to use a hill-climbing approach called stepwise search:

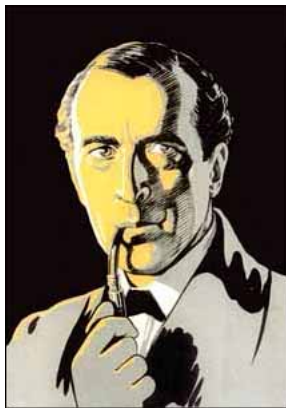
- 1 Start with some initial model, e.g. $y = b_0$, and compute its quality.
- 2 Neighbours: add or remove a predictor.
- 3 If all neighbours have lower quality, then stop and return the current model; otherwise move to the neighbour with highest quality and return to 2.

Classical Text Book Approach (Theory Driven)

- Specify hypothesis (model) of interest.
The model is determined up to a fixed number of unknown parameters.
- Collect relevant data.
- Estimate the unknown parameters from the data.
- Perform test, typically whether a certain parameter is zero, using the same data!

It is allowed to use the same data for fitting the model and testing the model, because we did not use the data to determine the model specification.

Data Mining: Sherlock Holmes Inference



“No data yet...It is a capital mistake to theorize before you have all the evidence. It biases the judgements.”

A study in scarlet

Data Mining (Data Driven)

A simple analysis scenario could look like this:

- Formulate question of interest.
- Select potentially relevant data.
- Divide the data into a training and test set.
- Use the training set to fit (many) different models.
- Use the test set to compare how well these models generalize.
- Select the model with the best generalization performance.

In this scenario, we cannot use the training data both to fit models and to test models!

K-fold cross-validation

- Divide the database in k parts
- For each of the k parts do
 - Use the remaining $k - 1$ parts to train the model.
 - Predict on the part that was not used for training.
 - Compute accuracy of the predictions.
- The average of the k accuracies is an estimate of the accuracy of the model induced from the complete database.