

# Aplicações da Computação para Astronomia

## Mineração de Dados e Simulação

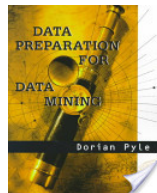
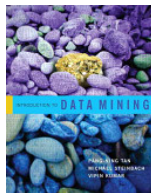
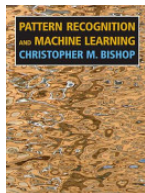
Eduardo Bezerra

Coordenação de Informática & Departamento de Informática & GPCA  
CEFET/RJ

23 de Agosto de 2012

# Credit where credit's due

This seminar was prepared by extracting material from the books bellow and from papers in the Bibliography.



# Overview of the talk

- 1 Introduction
- 2 Data Mining
  - Data preprocessing
  - Taks and Methods
  - Applications
  - Concluding Remarks
- 3 Simulations
- 4 Bibliography

# Overview of the talk

## 1 Introduction

## 2 Data Mining

- Data preprocessing
- Taks and Methods
- Applications
- Concluding Remarks

## 3 Simulations

## 4 Bibliography

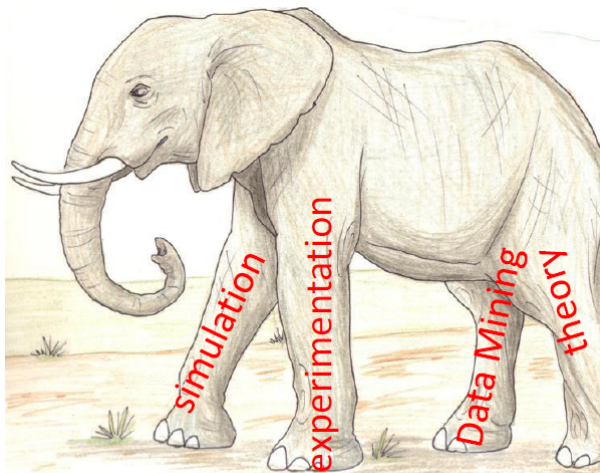
# Fourth Paradigm

“We have entered an era in which most data will never be seen by humans!”

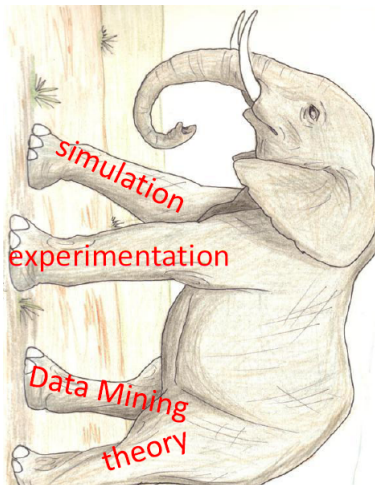
Fourth paradigm: scientific discovery based on **data-intensive** science.



# Fourth Paradigm



# Fourth Paradigm



# Overview of the talk

## 1 Introduction

## 2 Data Mining

- Data preprocessing
- Taks and Methods
- Applications
- Concluding Remarks

## 3 Simulations

## 4 Bibliography

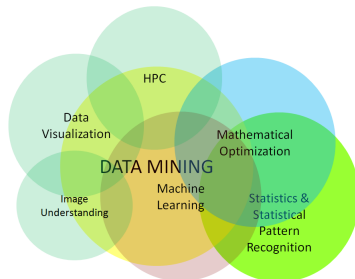


# Data Mining

The purpose of Data Mining is to find patterns in digital data, and translate these patterns into **useful information**.

Data mining problems involve **learning** (i.e., to infer the joint probability distribution that generates the data).

This learning is returned to a *human investigator* (e.g., scientist), which hopefully results in *human learning* (e.g., scientific discovery).



# Data Mining

The realization of a Data Mining task is composed of the following steps:

- 1 Data Preprocessing
- 2 Application of a Data Mining Method
- 3 Evaluation of Results

This is a highly **interactive** and **iterative** process, with feedback loops.

# Data preprocessing

Data preprocessing can be summarized by the two questions bellow. [HKP11]

- How can the data be preprocessed in order to help improve the quality of the data and, consequently, of the mining results?
- How can the data be preprocessed so as to improve the efficiency and ease of the mining process?

Data preprocessing is often **problem dependent**, and should be carefully applied because the results of many data mining algorithms can be significantly affected by the input data [Pyl99].

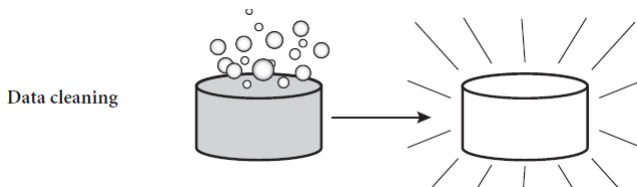
- Data Cleaning
- Data Integration
- Data Transformation
- Data Reduction
- Data Formatting

# Data Cleaning

In general, data will contain one or more types of bad values (e.g., duplicates, incorrect values, noise and outliers).

They may need to be removed either by simply removing the object containing them, ignoring the bad value but using the remaining data, or interpolating a new value using extra information.

Outliers may or may not be excluded, or may be excluded depending on their extremity.



# Data Transformation - Discretization

Some DM algorithms may require the object attributes to be **discrete** instead of **continuous**.

This procedure is needed when creating probability mass functions.

Discretization is a form of **binning**, as in making a histogram.

There are several methods to transform numerical data to categorical/discrete.

# Data Transformation - Handling Missing values

Data may contain **missing values**.

Example in Astronomy: in a cross-matched dataset where an object is not detected in a given waveband.

The reason for a missing value may be simply not known.

Some DM algorithms cannot be given missing values, which will require either the removal of the object or interpolation of the value from the existing data (**missing data imputation**). The advisability of interpolation is problem-dependent.

# Data Transformation - Normalization

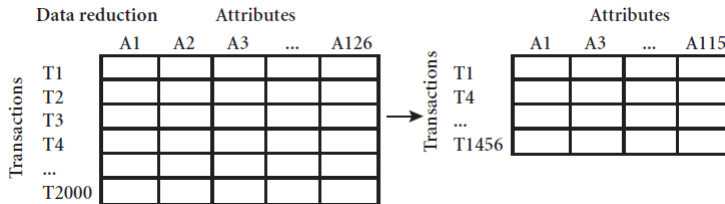
If some axis of the  $n$ -dimensional space created by  $n$  input attributes encompasses a range that, numerically, is much larger than the other axes, it may dominate the results, or create conditions where very large and small numbers interact, causing loss of accuracy.

Techniques of [data normalization](#) can solve this, and examples include

- linear transformations, like scaling by a given amount,
- scaling using the minimum and maximum values so that each attribute is in a given range such as 0-1,
- or scaling each attribute to have a mean of 0 and a standard deviation of 1.

# Data Reduction

Data reduction is divided into **sampling** and **dimension reduction**.





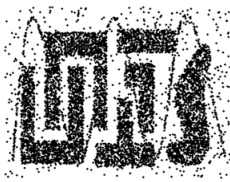
# Data Reduction - Sampling

Data may contain values that are correct but its size might be outside the desired (or possible) range of analysis. Or (in Astronomy) there may simply be a desired range, such as magnitude or position on the sky.

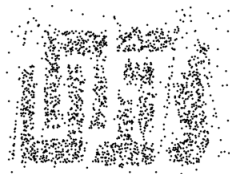
The data may therefore need to be **sampled**.

Sampling is often used for both the preliminary investigation of the data and the final data analysis.

Sample size must be carefully determined!



8000 points



2000 points



500 points

# Data Reduction - Dimension Reduction

In general, a large number of attributes will be available for each object in a dataset, and not all will be required for the data mining task. Indeed, use of all attributes may in many cases *worsen* performance (i.e., quality of results). This is a well-known problem, often called the **curse of dimensionality**.

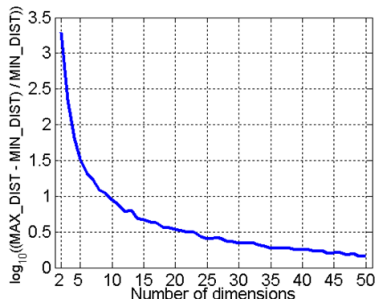


One therefore requires some form of **dimension reduction**, in which one wishes to retain as much of the information as possible, but in fewer attributes.

# Data Reduction - Curse of Dimensionality

Experiment:

- Randomly generate 500 points;
- Compute difference between max and min distance between any pair of points.



# Data Reduction - Curse of Dimensionality

With a finite number of data samples in a high-dimensional feature space with each feature having a number of possible values, an enormous amount of training data is required to ensure that there are several samples with each combination of values.

In general, with a fixed number of training samples, when dimensionality increases [BB10]:

- data becomes increasingly sparse in the space that it occupies;
- definitions of density and distance between points, which is critical for clustering and outlier detection, become less meaningful;
- the power of a predictive model inferred from this data reduces.

Astronomical data is cursed too!

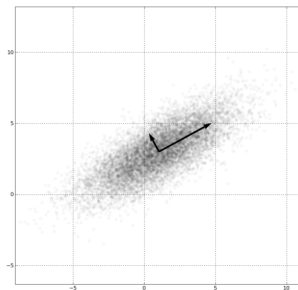
- multiwavelength datasets;
- cross-matching.

# Data Reduction - Dimension Reduction - PCA

The most trivial form of dimension reduction is simply to use one's judgement/experience and select a subset of attributes.

A less subjective and automated approach is **principal component analysis** (PCA).

It picks out the directions which contain the **greatest amount of information**. PCA is limited to *linear* relations.



# Data Reduction - Dimension Reduction - Other Techniques

Some other common dimension reduction techniques:

- Singular Value Decomposition
- Supervised techniques (that use the same or similar algorithms to those used for the actual data mining)
- Optimization methods (e.g., a genetic algorithm can be used in which each individual represents a subset of the training attributes to be used, and the algorithm selects the best subset.)

# Data formatting

Before passing the preprocessed data as input to a data mining method, this data should be put in an appropriate format.

A commonly used format for data mining packages is tabular data as, for example, the [attribute relation file format](#) (ARFF).

Common astronomical data formats include FITS, a binary format, and plain ASCII, while an emerging format is VOTable.

# Taks and Methods

Data mining methods broadly divide into **supervised** (predictive) and **unsupervised** (descriptive) methods. These can be mixed to form a third category, **semi-supervised** methods.

These methods can be used in several [Data Mining Tasks](#):

- Classification
- Clustering
- Time-series analysis
- Association analysis
- Outlier Detection
- ...



# Classification

A classification method relies on a **training set** of objects for which the target property, for example a classification, is known with confidence.

The method is trained on this set of objects, and the resulting mapping is applied to further objects for which the target property is not available.

These additional objects constitute the **testing set**.

# Classification - overview

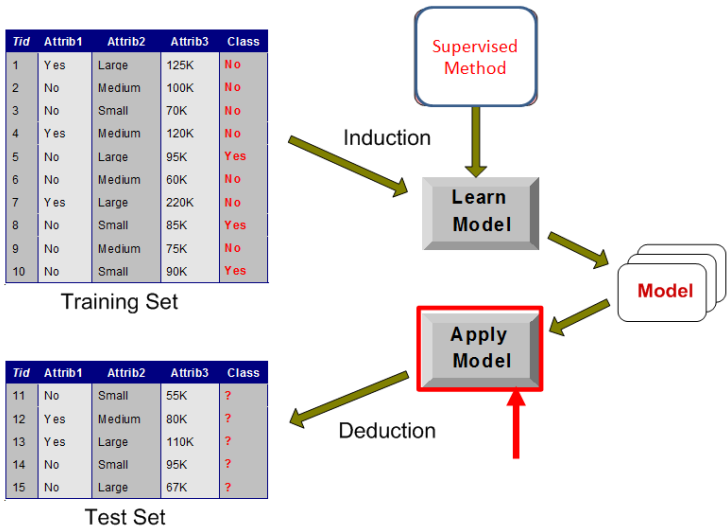


Figure from [TSK05].

# Clustering

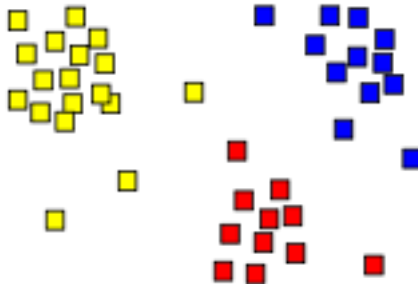
Clustering is the task of assigning a set of objects into groups (called **clusters**) so that the objects in the same cluster are more **similar** to each other than to those in other clusters.

Clustering methods do not require each object in the training set to be labelled with a class.

These methods usually require some kind of initial input for the values of one or more of the adjustable parameters, and the solution obtained depends on this input.

# Clustering

The result of a cluster analysis shown as the coloring of the squares into three clusters. (taken from Wikipedia)



# Clustering

Most clustering (and classification) algorithms rely on the concept of **distance** (**similarity**) between two objects (data points).

$$\text{Manhattan distance: } L_1(d_j, d_k) = \sum_{i=1}^n |d_{ij} - d_{ik}|$$

$$\text{Euclidean distance: } L_2(d_j, d_k) = \sqrt{\sum_{i=1}^n (d_{ij} - d_{ik})^2}$$

$$\text{Cossine distance: } \cos(d_j, d_k) = \frac{\sum_{i=1}^n (d_{ik} \times d_{ij})^2}{\sqrt{\sum_{i=1}^n (d_{ij})^2} \sqrt{\sum_{i=1}^n (d_{ik})^2}}$$

$$\text{Jaccard distance: } J_\delta(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|}$$

# Semi-supervised methods

Classification and clustering are supervised and unsupervised methods, respectively.

Semi-supervised methods make use of both labeled and unlabeled examples when learning a model.

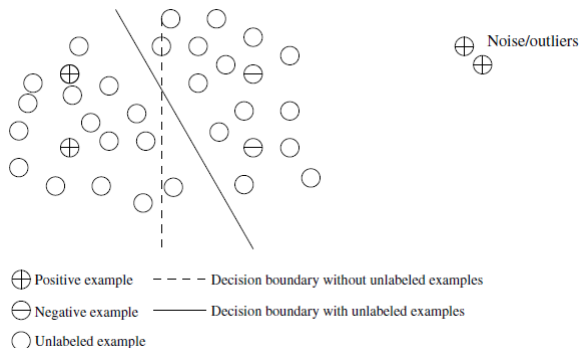
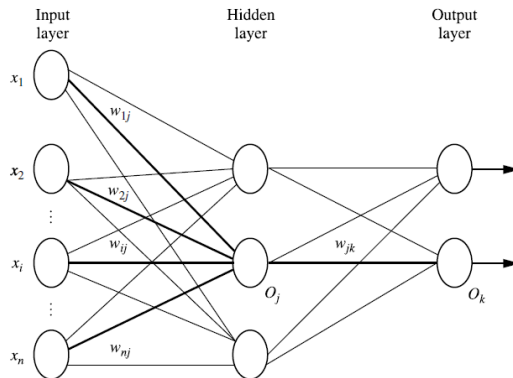


Figure from [HKP11]

# Artificial Neural Networks

A neural network is a set of connected input/output units in which each connection has a weight associated with it.

During the learning phase, the network learns by adjusting the weights.



# Support Vector Machines

SVM uses a nonlinear mapping to transform the original training data into a higher dimension.

Within this new dimension, it searches for the linear optimal separating hyperplane (i.e., a **decision boundary** separating the tuples of one class from another).

With an appropriate nonlinear mapping to a sufficiently high dimension, data from two classes can always be separated by a hyperplane.



# Support Vector Machines

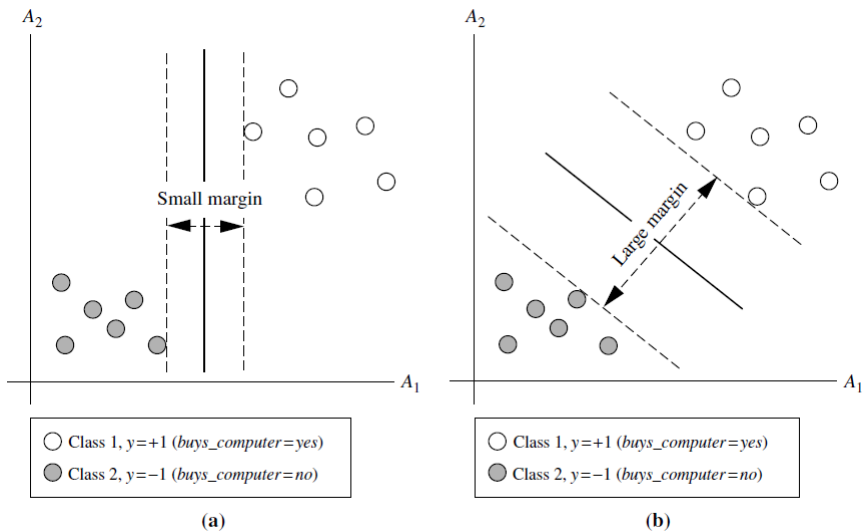


Figure from [HKP11]

# Decision Trees

Decision tree learning uses a **decision tree** as a predictive model which maps observations about an item to conclusions about the item's **target value**.

In these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels.

# Decision Trees

<i>RID</i>	<i>age</i>	<i>income</i>	<i>student</i>	<i>credit_rating</i>	<i>Class: buys_computer</i>
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

Figure from [HKP11]

# Decision Trees

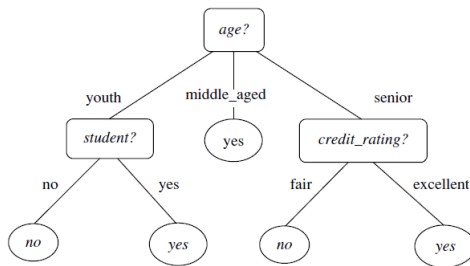


Figure from [HKP11]

# Decision Trees

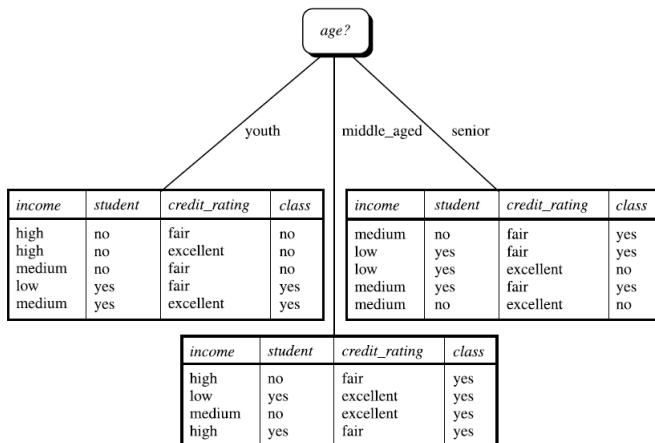


Figure from [HKP11]

# Application: Star-Galaxy Separation

Almost all stars are unresolved in photometric datasets. Galaxies, however, despite being further away, generally subtend a larger angle, and thus appear as extended sources.

However, other astrophysical objects such as quasars and supernovae, also appear as point sources.

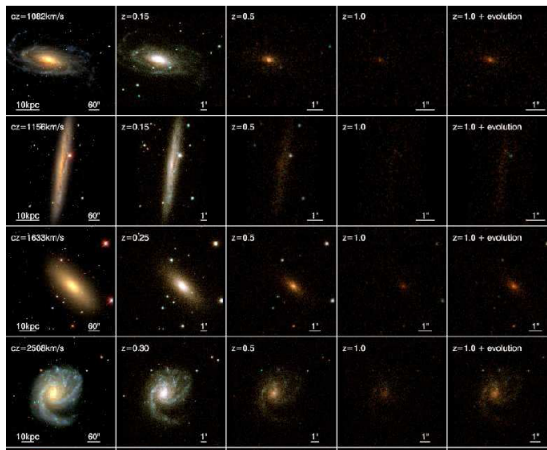
Thus, the separation of photometric catalogs into stars and galaxies is an important problem.

The huge number of galaxies and stars in typical surveys requires that such separation be *automated*.

This is an instance of a [classification task](#).

# Application: Galaxy Morphology Classification (1)

Studies exist that train some **classification algorithm** to assign **T types** to images for which measured parameters are available. Such parameters can be purely morphological, or include other information such (e.g., color).



## Application: Galaxy Morphology Classification (2)

Manually assigning class labels to objects to form a training set for classification is very time-consuming and error-prone.

Recently, Galaxy Zoo project employed [crowdsourcing](#): an application was made available online in which members of the general public were able to view images from the SDSS and assign classifications according to an outlined scheme.

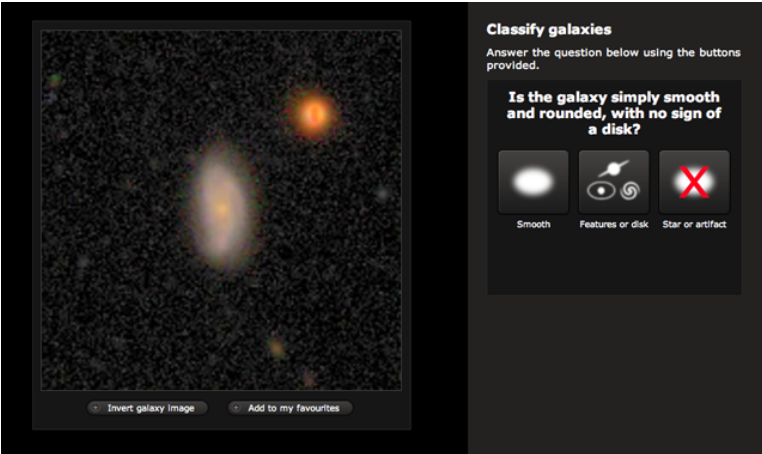
In a period of six months over 100,000 people provided over 40 million classifications for a sample of 893,212 galaxies.

This approach is interesting and represents a *complementary* one to automated algorithms.

[Idea to be tested: gamification!](#) (use of game design techniques, game thinking and game mechanics to enhance non-game contexts. –wikipedia)



# Application: Galaxy Morphology Classification (2)



**Classify galaxies**

Answer the question below using the buttons provided.

**Is the galaxy simply smooth and rounded, with no sign of a disk?**

Smooth      Features or disk      Star or artifact

Image taken from <http://arstechnica.com/science/2010/10/galaxy-zoo-shows-how-well-crowdsourced-citizen-science-works/>.

# Application: Stellar Clusters Detection

([Ongoing collaboration work!](#)) Problem: In a given field, segregate the field and cluster stars.

Currently playing with an [extension](#) of DBSCAN (a clustering algorithm) and a spectral clustering algorithm to analyse photometric data.

# Concluding Remarks

Time flies like a comet!

Timestamped datasets are expected to become increasingly important with the advent of LSST. For example, LSST is expected to

- photograph the entire available sky every few nights. (– LSST FAQ)
- produce a list of 1000 new supernovae each night for 10 years [Bor09].

There are several challenges, though:

- handling multiple observations of the same object,
- handling heteroskedasticity (i.e., variability itself can change),
- robust classification of large streams of data in **real time**,
- the volume and storage of time domain information.

Time series analysis techniques can be applied here.

# Concluding Remarks

There is no single best DM algorithm.

- aka **No Free Lunch Theorem** [DHS01, Bis06].
- There is no simple method to select the optimal algorithm to use.
- The most appropriate algorithm can depend not only on the dataset, but also the application for which it will be employed.
- Likewise, many DM tools and frameworks exist, but it is unlikely that one of them will be able to perform all steps necessary from raw catalog to desired science result, particularly for large datasets.

# Concluding Remarks

Garbage in, garbage out.

- The result can only be as good as the data.
- If the data are not sufficient for the task, or are poorly collected or incorrectly treated, the result will not be useful.
- Data preparation is more than half of every data mining process.

# Concluding Remarks

Collaboration is necessary.

- DM algorithms will not necessarily establish which patterns or relationships are important scientifically.
- The optimal configuration of parameters in a DM method and the adequate use of preprocessing techniques is not obvious.
- Likewise, computer scientists normally are not used to terms like optical transients, photometry, spectrometry, isochrones, and the like.

Therefore, a sucessfull project will probabilly require close collaboration between the [data miner](#) and the [domain scientist](#).

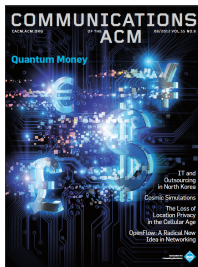
# Overview of the talk

- 1 Introduction
- 2 Data Mining
  - Data preprocessing
  - Taks and Methods
  - Applications
  - Concluding Remarks
- 3 Simulations**
- 4 Bibliography

# Cosmic Simulations

Cover story in the ACM Magazine of August 2012:

*With the help of supercomputers, scientists are now able to create models of large-scale astronomical events.*





# Cosmic Simulations

A snapshot visualization from the Bolshoi simulation depicting the evolution of gas density.

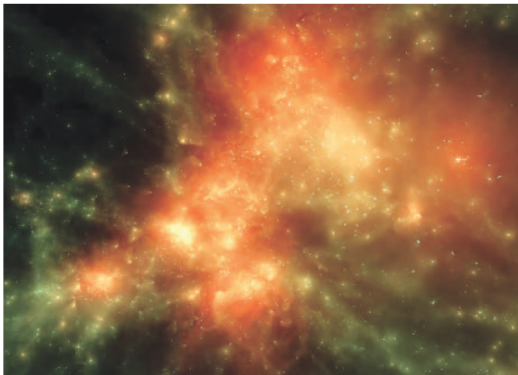


Figure from ACM Magazine, August 2012.

# Glimpse into birth of the Milky Way

Researchers from the University of California and from the University of Zurich produced a realistic simulation of the birth of the Milky Way.

Simulation included 1.4 million processor-hours on NASA's state-of-the-art Pleiades supercomputer.

Movie resulting from the computer simulation: [http://www.youtube.com/watch?v=nccfTciNQWY&feature=player\\_embedded](http://www.youtube.com/watch?v=nccfTciNQWY&feature=player_embedded)

The simulation took 8 months; it would take 570 years in a personal computer.

# Overview of the talk

- 1 Introduction
- 2 Data Mining
  - Data preprocessing
  - Taks and Methods
  - Applications
  - Concluding Remarks
- 3 Simulations
- 4 Bibliography

# Bibliography I



N. M. Ball and R. J. Brunner.

Data Mining and Machine Learning in Astronomy.

*International Journal of Modern Physics D*, 19:1049–1106, 2010.



C.M. Bishop.

*Pattern Recognition and Machine Learning*.

Information Science and Statistics. Springer, 2006.



Kirk D. Borne.

Scientific data mining in astronomy.

*CoRR*, abs/0911.0505, 2009.



R.O. Duda, P.E. Hart, and D.G. Stork.

*Pattern classification*.

Pattern Classification and Scene Analysis: Pattern Classification. Wiley, 2001.



Jiawei Han, Micheline Kamber, and Jian Pei.

*Data Mining: Concepts and Techniques*.

Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2011.

# Bibliography II



Dorian Pyle.

*Data preparation for data mining.*

Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.



Pang-Ning Tan, Michael Steinbach, and Vipin Kumar.

*Introduction to Data Mining.*

Addison Wesley, 1 edition, May 2005.

*It has been said that astronomers have been doing data mining for centuries: “the data are mine, and you cannot have them!” [Bor09]*

**Thank you!**