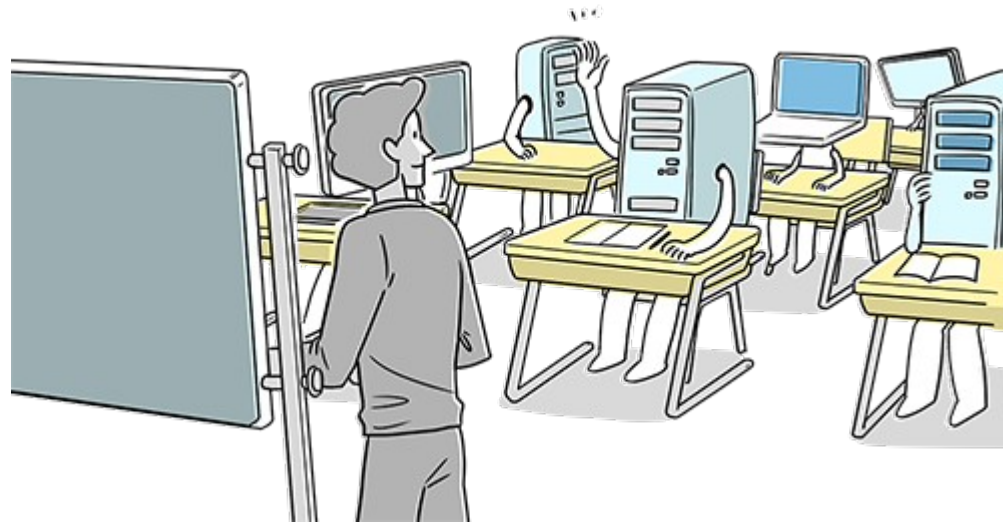


Machine learning

Lecture 1



Marcin Wolter
IFJ PAN

29 January 2019

- Machine learning: what does it mean?
- Software to work with and literature.
- A little bit of mathematics and examples of simple linear classifiers.
- Some examples



Outline of the course

- Introduction: introduction to statistics, what does "Machine Learning" mean? A little bit of mathematics, but also examples of simple linear classifiers.
- Simple non-linear methods like Naive Bayes, k-Nearest Neighbors, Probability Density Estimators and Boosted Decision Trees (BDT).
- Neural Networks and Bayesian Neural Networks.
- Cross-validation and optimization of machine learning algorithms. Introduction to Deep Learning.
- Deep Learning and convolution network. Application of Deep Learning to High Energy Physics problems - Higgs searches.
- Generative deep networks – Generative Adversary Networks (GANs)



Recommended books

- M. Krzyśko, **Systemy uczące się: rozpoznawanie wzorców, analiza skupień i redukcja wymiarowości**. WNT, 2008.
- C. Bishop, **Pattern recognition and machine learning**. Springer, 2009.

and maybe my thesis (unfortunately in Polish):

- M. Wolter, *Metody analizy wielu zmiennych w fizyce wysokich energii*
https://www.epnp.pl/ebook/metody_analizy_wielu_zmiennych_w_fizyce_wysokich_energii

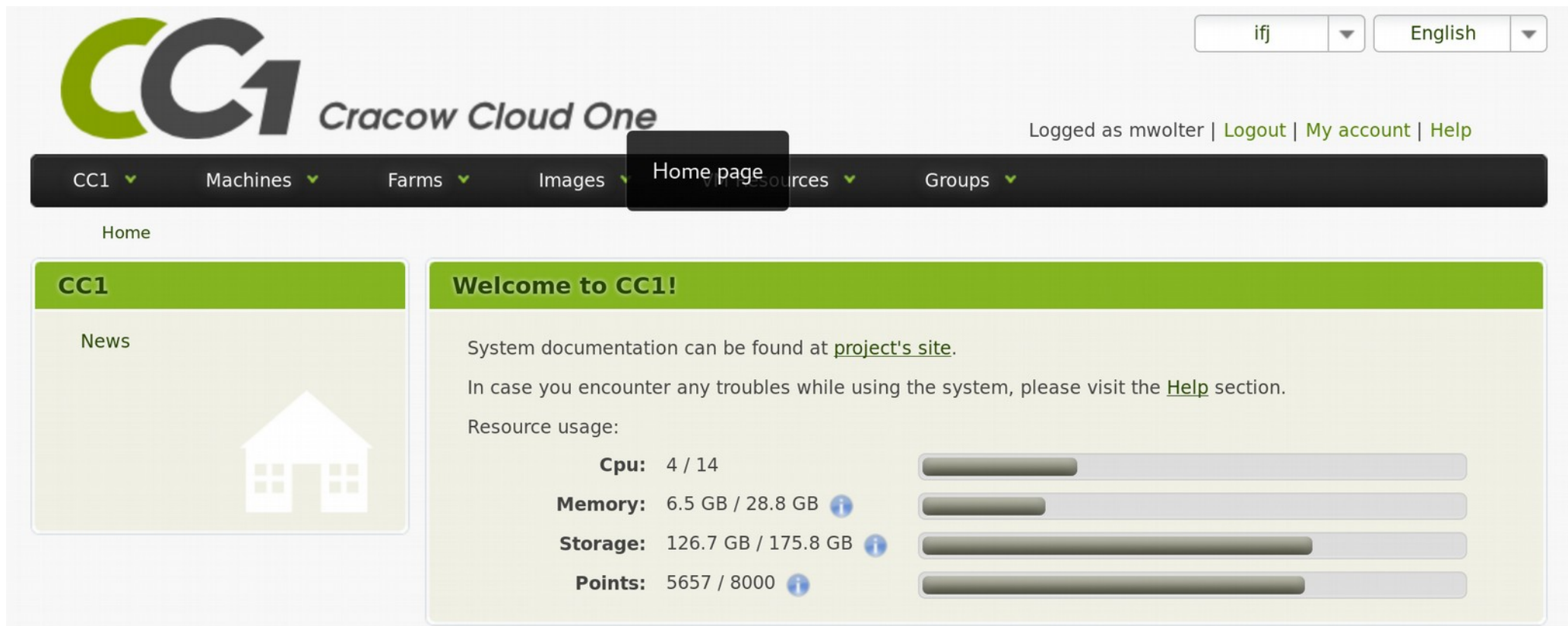


Programs

- **TMVA – integrated with the ROOT package**
<http://tmva.sf.net>
Installs together with root
Very popular at CERN
- <https://keras.io/>
Keras: The Python Deep Learning library
Emulates Deep Neural Network, uses google TensorFlow software
- <http://scikit-learn.org>
scikit-learn - Machine Learning in Python
Simple and efficient tools for data mining and data analysis
Accessible to everybody and reusable in various contexts
Built on NumPy, SciPy, and matplotlib
I have never used scikit for real analysis, but we can learn together!

Computing

- <https://www.cloud.ifj.edu.pl/>
- Register, you can create your virtual linux box and play with it.
- Install root together with TMVA



The screenshot shows the Cracow Cloud One (CC1) website interface. At the top right, there are dropdown menus for 'ifj' and 'English'. The main header features the CC1 logo and the text 'Cracow Cloud One'. Below the header, a navigation bar contains links for 'CC1', 'Machines', 'Farms', 'Images', 'Home page', 'Resources', and 'Groups'. The 'Home page' link is highlighted with a black tooltip. Below the navigation bar, the page content is divided into two main sections. On the left, there is a 'CC1 News' section with a house icon. On the right, there is a 'Welcome to CC1!' section with a green header. This section contains text about system documentation and resource usage. The resource usage section includes four rows of data, each with a label, a value, and a progress bar:

Resource	Usage	Limit
Cpu	4 / 14	
Memory	6.5 GB / 28.8 GB	
Storage	126.7 GB / 175.8 GB	
Points	5657 / 8000	



Caesar cipher

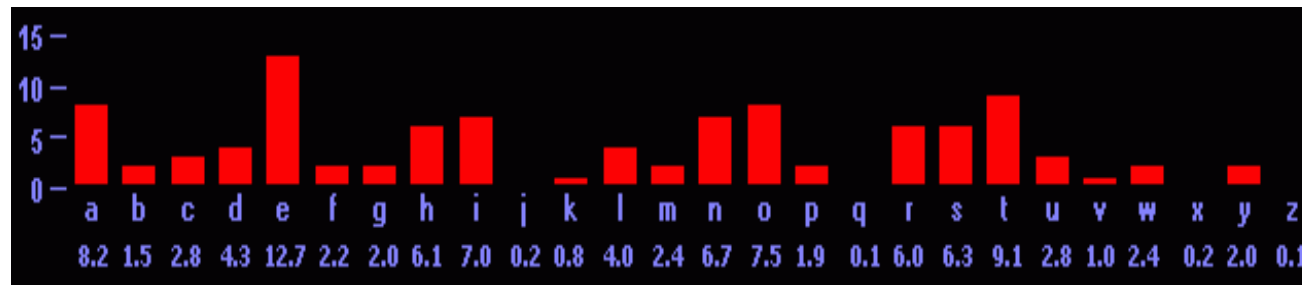
- Each letter of a text is replaced by another, shifted by n letters:

A	B	C	D	E
C	D	E	F	G

- In general Al-Kindi's method can be used to break any replacement cipher (each letter is replaced by another letter, always the same)

A	B	C	D	E
Z	D	P	G	T

- Frequency analysis – the frequency of appearance of different letters is investigated.



- Frequency of different letters in English.

What is an “event”?

- Elementary event is each result of an experiment (like throwing a dice), which result is random. It contains only a single outcome in the sample space.

The numerical value of a probability may sometimes be obtained from its "classical" definition: The probability is equal to the quotient of the number of cases "favouring" a certain event to the total number of "equally possible" cases. (Laplace 1812).

- Let's denote a set of all possible events as Ω . Elements of a set Ω are elementary events ω , so Ω is a set of elementary events. The set of events favouring A is a subset of Ω and:

$$P(A) = \frac{|A|}{|\Omega|}$$

where $|A|$ is a number of elements of a set A , and $|\Omega|$ a number of elements of a set Ω .

Example: probability of getting 6 while throwing a dice. Set of elementary events $\Omega = \{1, 2, 3, 4, 5, 6\}$, so the number of elementary events is $|\Omega| = 6$. Set of events favouring $A = \{6\}$, their number is $|A| = 1$. So $P(6) = 1/6$



Pierre-Simon Laplace
(1749–1827)

Frequentist definition of probability

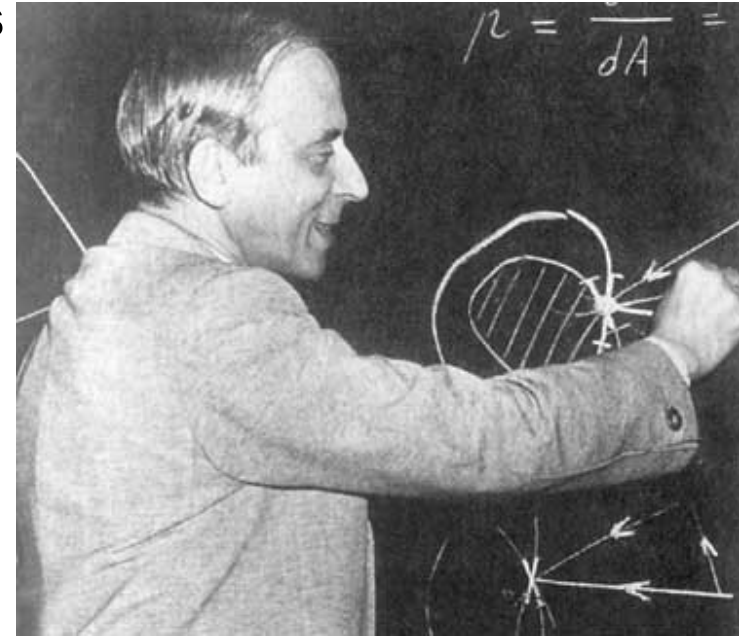
- Probability (frequentist definition) of an event A is a limit (N approaching infinity) of the ratio of n events when the event A occurred to the total number of trials N :

$$P(A) = \lim_{N \rightarrow \infty} \frac{n}{N}$$

What is a probability of getting “6” while throwing a dice? It is the relative ratio of obtaining “6” in an infinite series of throws.

- The definition comes from Richard von Mises (born 19 of April 1883 in Lwów, died 14 July 1953 in Boston) – mathematician, brother of the economist Ludwig von Mises.

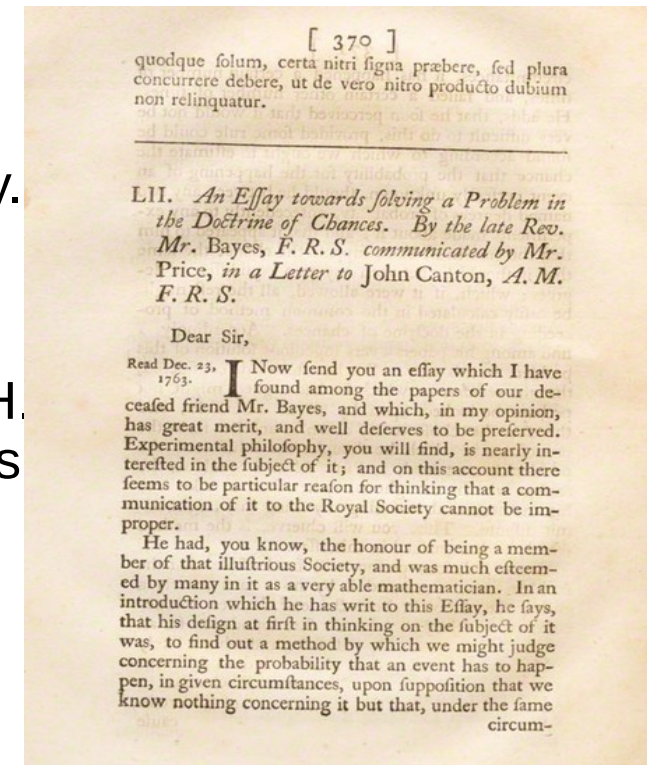
This idea of frequentist probability was used earlier, for example by de’Moivre, Bernoulli, Gauss ...



Bayesian definition



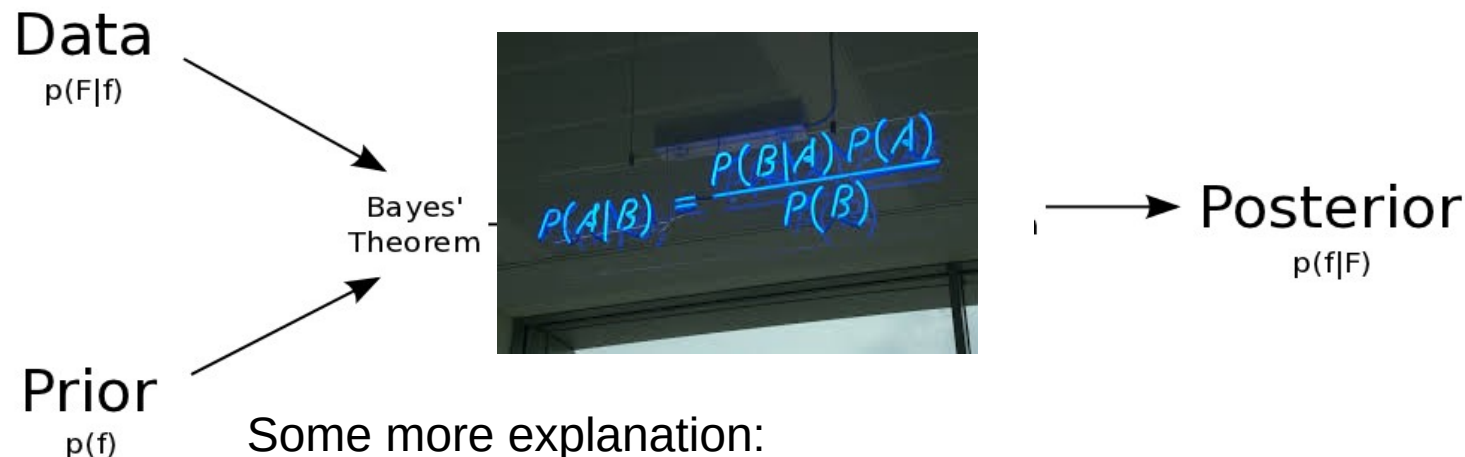
- Probability “a priori”, called unconditional, is a measure of belief, based on rational premises, that a given event will occur.
- In the next step we perform an experiment, called observation, and their results allow to modify the probability. We get the probability “a posteriori”, which is again a measure of belief, but modified by the observation.
- Supporters of the Bayesian approach were P. S. Laplace, H. Poincare and the economist John Keynes, arguing, that this is a method we use analyzing the world around us.



Thomas Bayes (1702 - 1761) was an English statistician, philosopher and Presbyterian minister. The most important work: „Essay Towards Solving a Problem in the Doctrine of Chances”.

Bayesian probability

- Experiment we can't repeat many times: what is a probability to pass an exam?
- Based on our knowledge we estimate: $\frac{1}{2}$ (probability *a priori*).
- But if all people before us didn't pass an exam (set of experiments) and we know our knowledge is not significantly higher we should verify this estimate (probability *a posteriori*).



Some more explanation:

http://el.us.edu.pl/ekonofizyka/index.php/Statystyka_w_uj%C4%99ciu_Bayesowskim

<https://towardsdatascience.com/probability-concepts-explained-bayesian-inference-for-parameter-estimation-90e8930e5348>



Bayes Theorem

- Bayes' theorem relates the conditional (posterior) and marginal (prior) probabilities of events A and B:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- **P(A)** is the prior probability or marginal probability of A. It is a "prior" in the sense that it does not take into account any information about B.
- **P(A|B)** is the conditional probability of A, given B. It is also called the posterior probability because it is derived from or depends upon the specified value of B.
- **Intuitively, Bayes' theorem in this form describes the way in which one's beliefs about observing 'A' are updated by having observed 'B'.**

Bayes Theorem – an example: a cancer test



$$\Pr(A|X) = \frac{\Pr(X|A) \Pr(A)}{\Pr(X)} = \frac{\Pr(X|A) \Pr(A)}{\Pr(X|A) \Pr(A) + \Pr(X|\text{not } A) \Pr(\text{not } A)}$$

- $\Pr(A|X)$ = Chance of having cancer (A) given a positive test (X). This is what we want to know: How likely is it to have cancer with a positive result? .
- $\Pr(X|A)$ = Chance of a positive test (X) given that you had cancer (A). This is the chance of a true positive, let say 80% in our case.
- $\Pr(A)$ = Chance of having cancer (1%).
- $\Pr(\text{not } A)$ = Chance of not having cancer (99%).
- $\Pr(X|\text{not } A)$ = Chance of a positive test (X) given that you didn't have cancer (not A). This is a false positive, 9.6% in our case.
- **In our case $\Pr(A|X)$ is 7.8%**

Bayesian vs. Frequentist approach



- **PROBABILITY: degree of belief** (Bayes, Laplace, Gauss, Jeffreys, de Finetti)
- **PROBABILITY: relative frequency** (Venn, Fisher, Neyman, von Mises).
- **Bayesian approach:** probability is degree of belief. Thus the probability p is our assessment of the probability of success at each trial, based on our current state of knowledge.

If our assessment, initially, is incorrect? As our state of knowledge changes, our assessment of the probability of success changes accordingly.
- **Bayesian inference** is statistical inference in which **evidence or observations are used** to update or to newly infer the probability that a hypothesis may be true.
- This allows for a *cleaner* foundation than the frequentist interpretation.

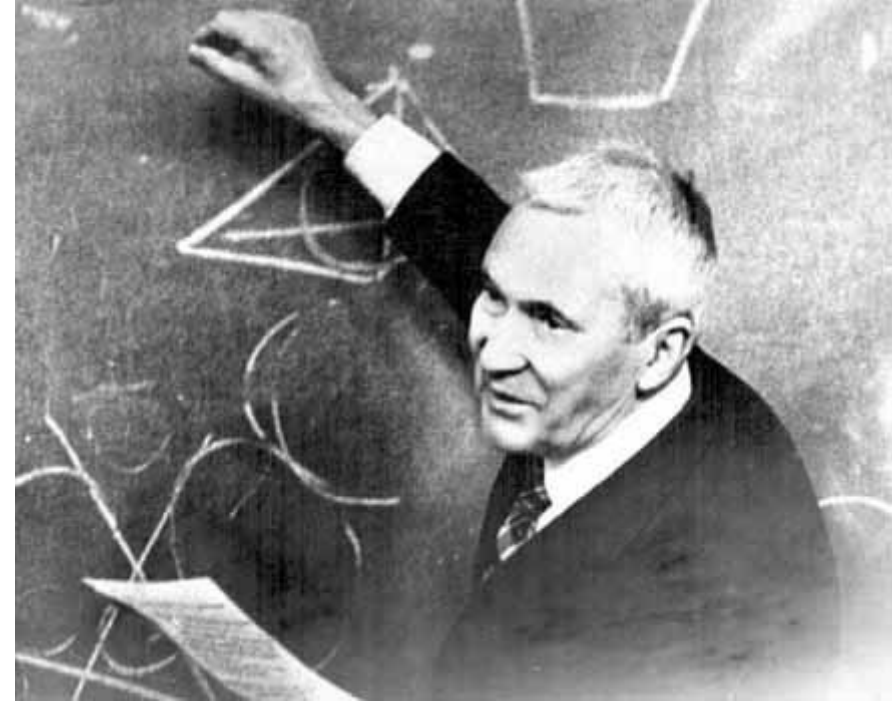
“We don’t know all about the world to start with; our knowledge by experience consists simply of a rather scattered lot of sensations, and we cannot get any further without some a priori postulates. My problem is to get these stated as clearly as possible.”

Sir Harold Jeffreys, in a letter to Sir Ronald Fisher dated 1 March, 1934

H.B. Prosper, “Bayesian Analysis”, arXiv:hep-ph/0006356v1 30 Jun 2000

Axiomatic definition

Probability can be defined in many ways...



Kolmogorov axiomatic definition:

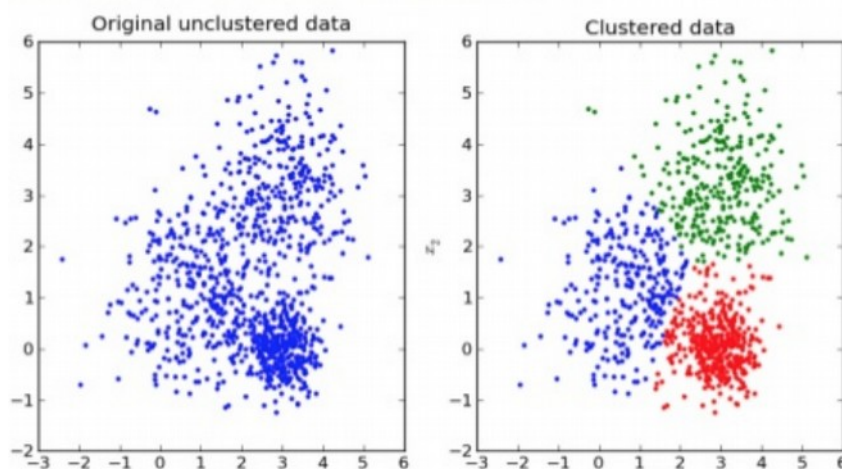
Андрей Николаевич Колмогоров (1903-1987)

Let Q_i denote anything subject to weighting by a normalized linear scheme of weights that sum to unity in a set W . The Kolmogorov axioms state that:

1. For every Q_i in W , there is a real number $Q(Q_i)$ (the Kolmogorov weight of Q_i) such that $0 < Q(Q_i) < 1$.
2. $Q(Q_i) + Q(Q'_i) = 1$, where Q'_i denotes the complement of Q_i in W .
3. For the mutually exclusive subsets Q_1, Q_2, \dots in W ,
 $Q(Q_1 \cup Q_2 \cup Q_3 \cup \dots) = Q(Q_1) + Q(Q_2) + Q(Q_3) + \dots$

What does “machine learning” mean?

- **Machine learning** is a field of computer science that gives computer systems the ability to "learn" (i.e. progressively improve performance on a specific task) with data, without being explicitly programmed.
- Problems:
 - Supervised learning (classification & regression)
 - Clustering (unsupervised learning)
 - Dimensionality reduction
 - Reinforcement learning
 - Many others.....



➤ Unsupervised Learning

- ❑ Technique of trying to find hidden structure in unlabeled data

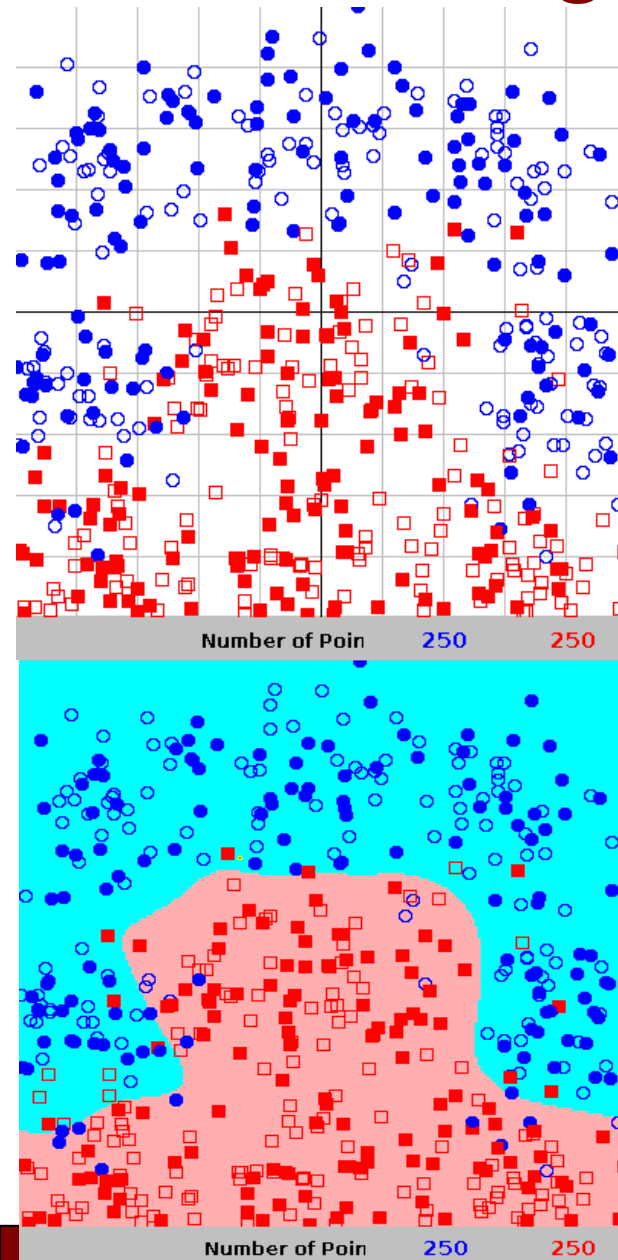
➤ Supervise Learning

- ❑ Technique for creating a function from training data. The training data consist of pairs of input objects (typically vectors), and desired outputs.

How do the (supervised) machine learning algorithms work?

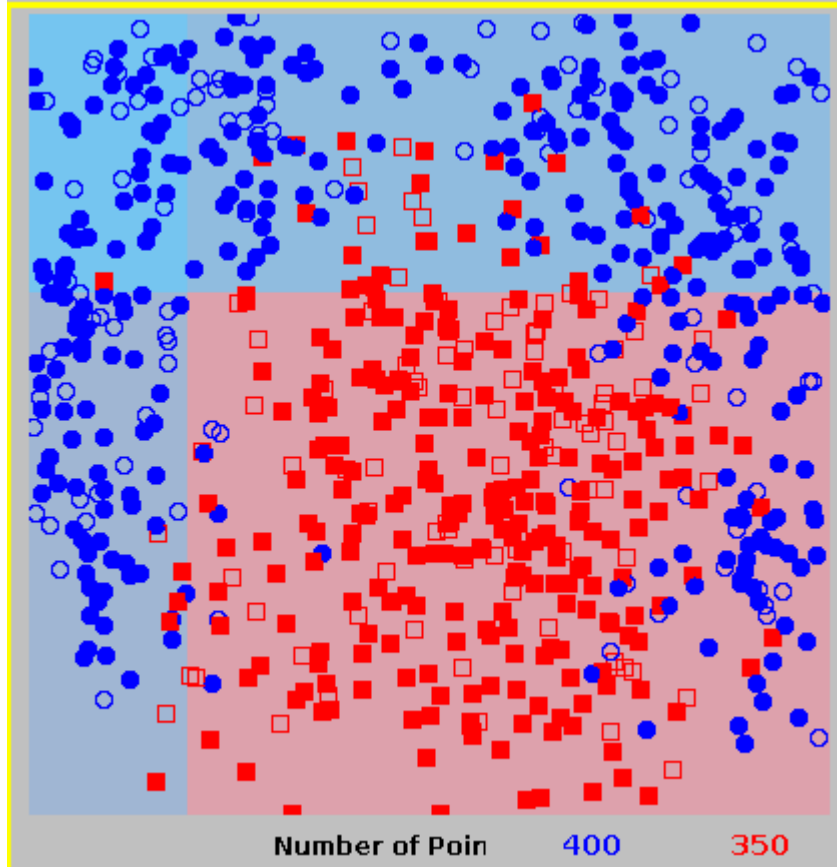
- We need **training data**, for which we know the correct answer, whether it's a signal or background. We divide the data into two samples: training and test.
- We find the best function $f(\mathbf{x})$ which describes the probability, that a given event belongs to the class "signal". This is done by minimizing the loss function (for example χ^2).
- Different algorithms differ by: the class of function used as $f(\mathbf{x})$ (linear, non-linear etc), loss function and the way it's minimized.
- All these algorithms try to approximate the unknown *Bayesian Decisive Function* (BDF) relying on the finite training sample.

BDF -an ideal classification function given by the unknown probability densities of signal and background.

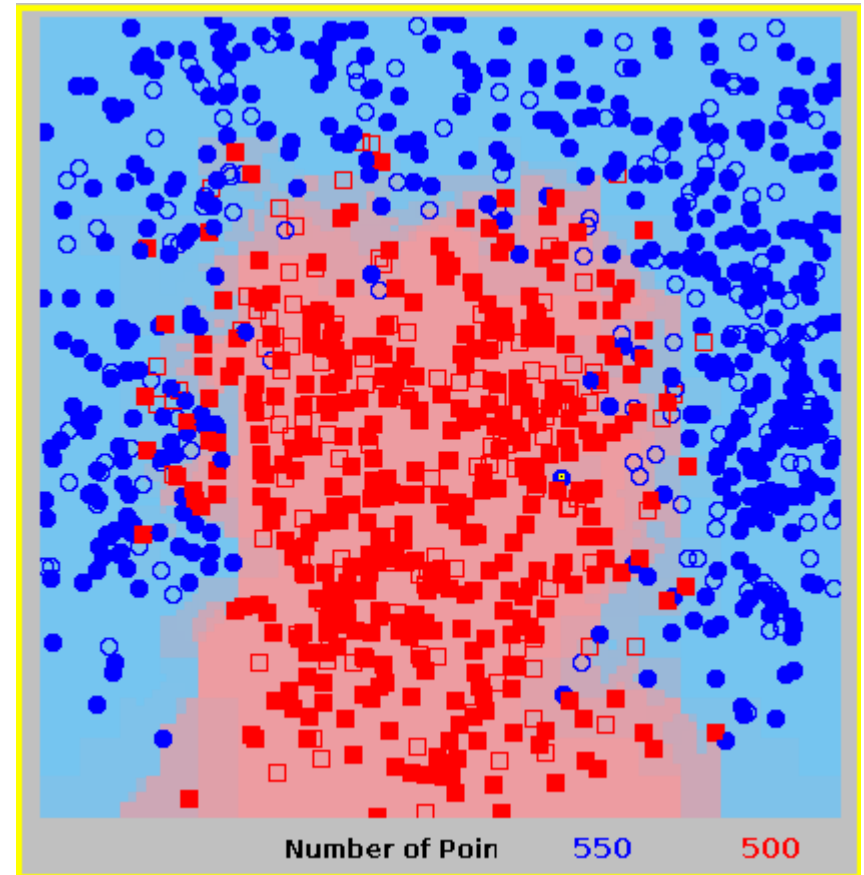


Cuts vs non-linear separation

Cuts



Non-linear separation



Neural Networks, boosted decision trees, and so on....

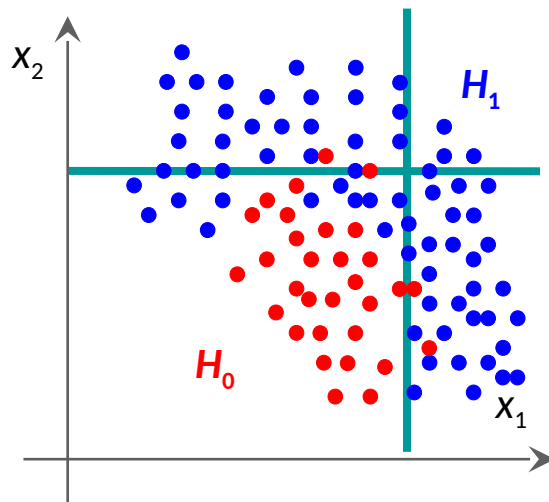
Types of algorithms



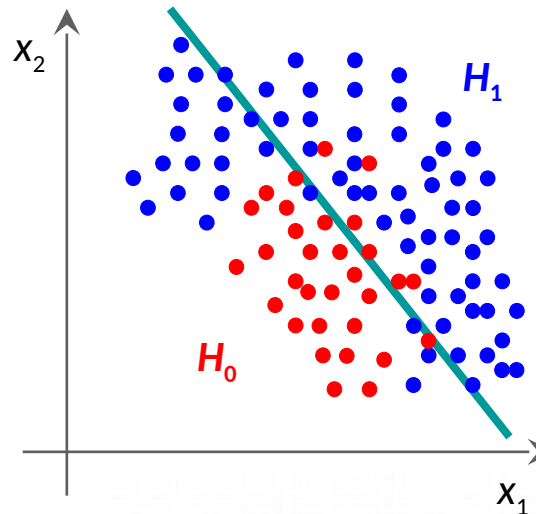
How to use the information available

Classification: find a function $f(x_1, x_2)$ giving the probability, that a given data point belongs to a given class (signal vs background).

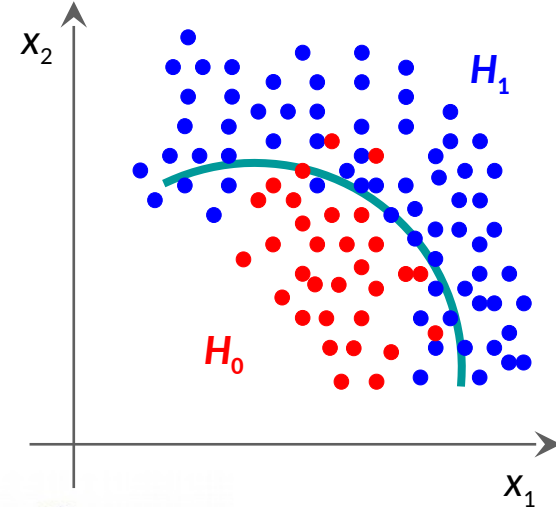
Simple cuts
(easy and intuitive)



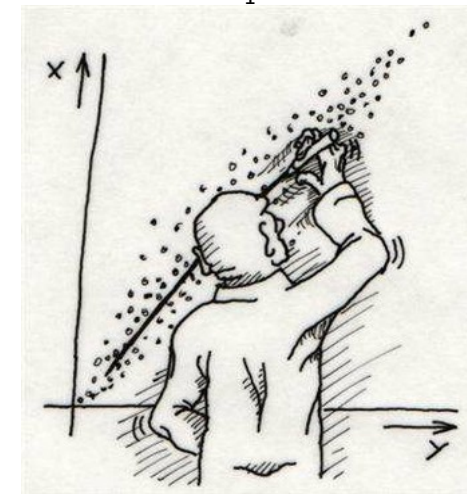
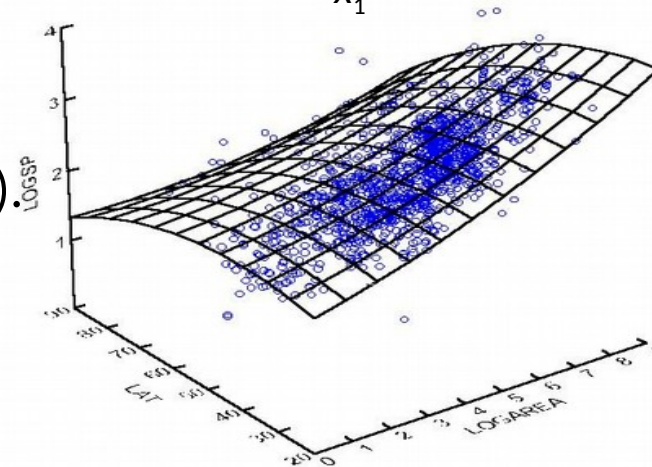
Linear
(fast and stable)



Non-linear
(most effective)



Regression: fit a continuous function
(find particle energy from calo readouts).



Classification

A Bayes classifier:

$$p(S|x) = \frac{p(x|S) p(S)}{p(x|S) p(S) + p(x|B) p(B)}$$

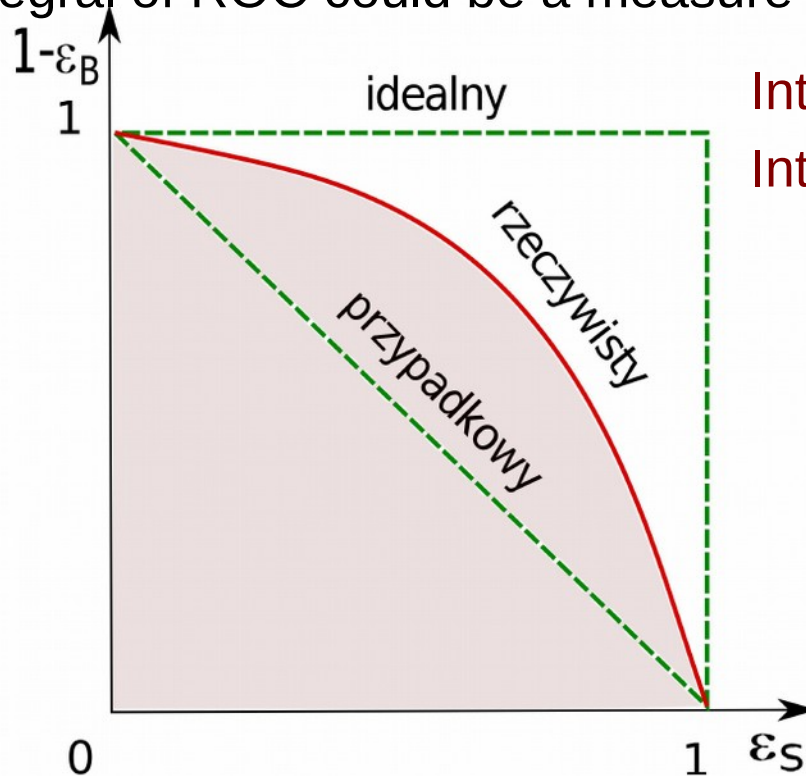
where **S** is associated with $y = \mathbf{1}$ and **B** with $y = \mathbf{0}$. **Bayes classifier** accepts events x if $p(\mathbf{S}|x) > \mathbf{cut}$ as belonging to **S**.

We need to approximate probability distributions $P(x|\mathbf{S})$ and $P(x|\mathbf{B})$.

- If your goal is to **classify objects** with the fewest errors, then the **Bayes classifier** is the **optimal** solution.
- Consequently, if you have a classifier known to be **close** to the **Bayes limit**, then *any* other classifier, *however sophisticated*, can **at best** be only marginally better than the one you have.
 - => If your problem is **linear** you don't gain anything by using sophisticated **Neural Network**
- All classification methods, such as the ones in TMVA, are different numerical approximations of the Bayes classifier.

ROC curve

- ROC (Receiver Operation Characteristic) curve was first used to calibrate radars.
- Shows the background rejection ($1-\varepsilon_B$) vs signal efficiency ε_S . Shows how good the classifier is.
- The integral of ROC could be a measure of the classifier quality:



Integral(ROC) = $\frac{1}{2}$ – random
 Integral(ROC) = 1 - ideal



Practical applications

A Short List of Multivariate Methods

- Cuts
- Linear Discriminants (like Fisher)
- Support Vector Machines
- Naive Bayes (Likelihood Discriminant)
- Kernel Density Estimation
- Decision Trees
- Neural Networks
- Bayesian Neural Networks
- Genetic Algorithms

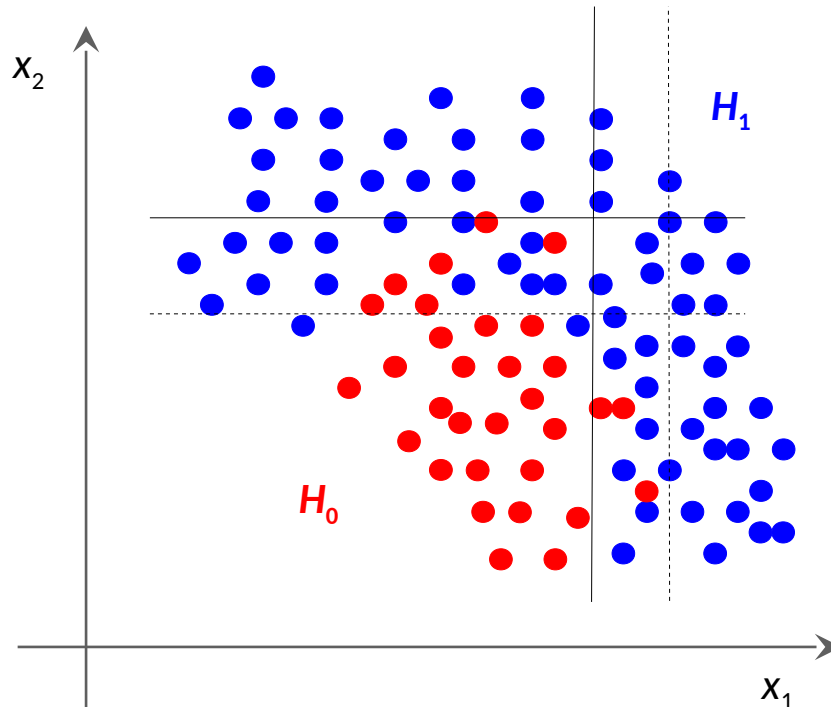
- And many, many others..... I want to present briefly just few of them.



We will talk today about:

- Simple ML linear methods:
 - Cuts
 - Fisher linear discriminant
 - Principal Component Analysis, PCA
 - Independent Component Analysis, ICA

Cuts



Optimization of cuts:

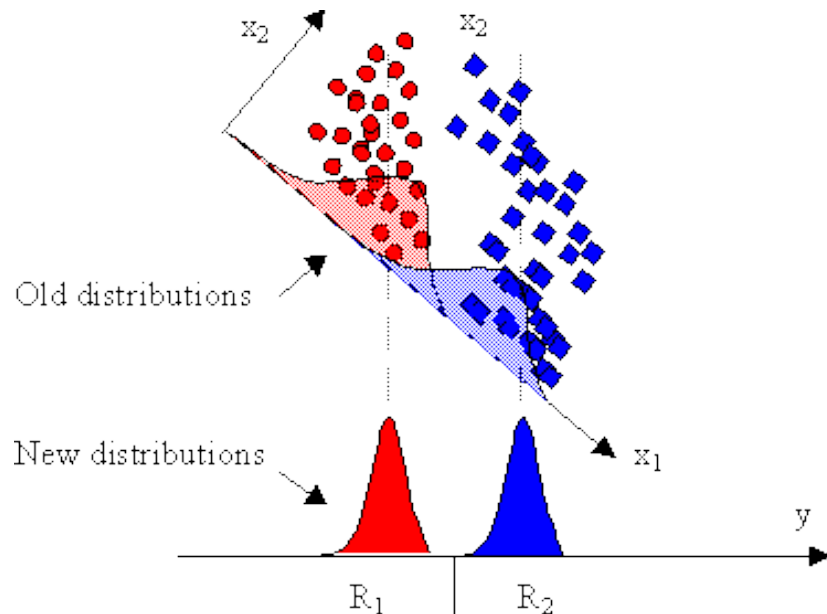
- Move cuts as long as we get the optimal signal vs. background selection. For a given signal efficiency we find the best background rejection → we get the entire ROC curve.
- Optimization methods:
 - Brute force
 - Genetic algorithms
 - Many others...

Fisher discriminants

LDA, Linear Discriminant Analysis

Projection to one dimension, than discrimination

Equivalent to linear separation



We choose a projection vector in such a way, that the separation is maximized.

Method introduced by Fisher in 1936.
Optimal separation for Gaussian distributions.



Fisher's linear discriminant

The terms *Fisher's linear discriminant* and *LDA* are often used interchangeably, although [Fisher's](#) original article *The Use of Multiple Measures in Taxonomic Problems* (1936) actually describes a slightly different discriminant, which does not make some of the assumptions of LDA such as normally distributed classes or equal class covariances.

Suppose two classes of observations have means $\vec{\mu}_{y=0}, \vec{\mu}_{y=1}$ and covariances $\Sigma_{y=0}, \Sigma_{y=1}$. Then the linear combination of features $\vec{w} \cdot \vec{x}$ will have means $\vec{w} \cdot \vec{\mu}_{y=i}$ and variances $\vec{w}^T \Sigma_{y=i} \vec{w}$ for $i = 0, 1$. Fisher defined the separation between these two distributions to be the ratio of the variance between the classes to the variance within the classes:

$$S = \frac{\sigma_{between}^2}{\sigma_{within}^2} = \frac{(\vec{w} \cdot \vec{\mu}_{y=1} - \vec{w} \cdot \vec{\mu}_{y=0})^2}{\vec{w}^T \Sigma_{y=1} \vec{w} + \vec{w}^T \Sigma_{y=0} \vec{w}} = \frac{(\vec{w} \cdot (\vec{\mu}_{y=1} - \vec{\mu}_{y=0}))^2}{\vec{w}^T (\Sigma_{y=0} + \Sigma_{y=1}) \vec{w}}$$

This measure is, in some sense, a measure of the [signal-to-noise ratio](#) for the class labelling. It can be shown that the maximum separation occurs when

$$\vec{w} = (\Sigma_{y=0} + \Sigma_{y=1})^{-1} (\vec{\mu}_{y=1} - \vec{\mu}_{y=0})$$

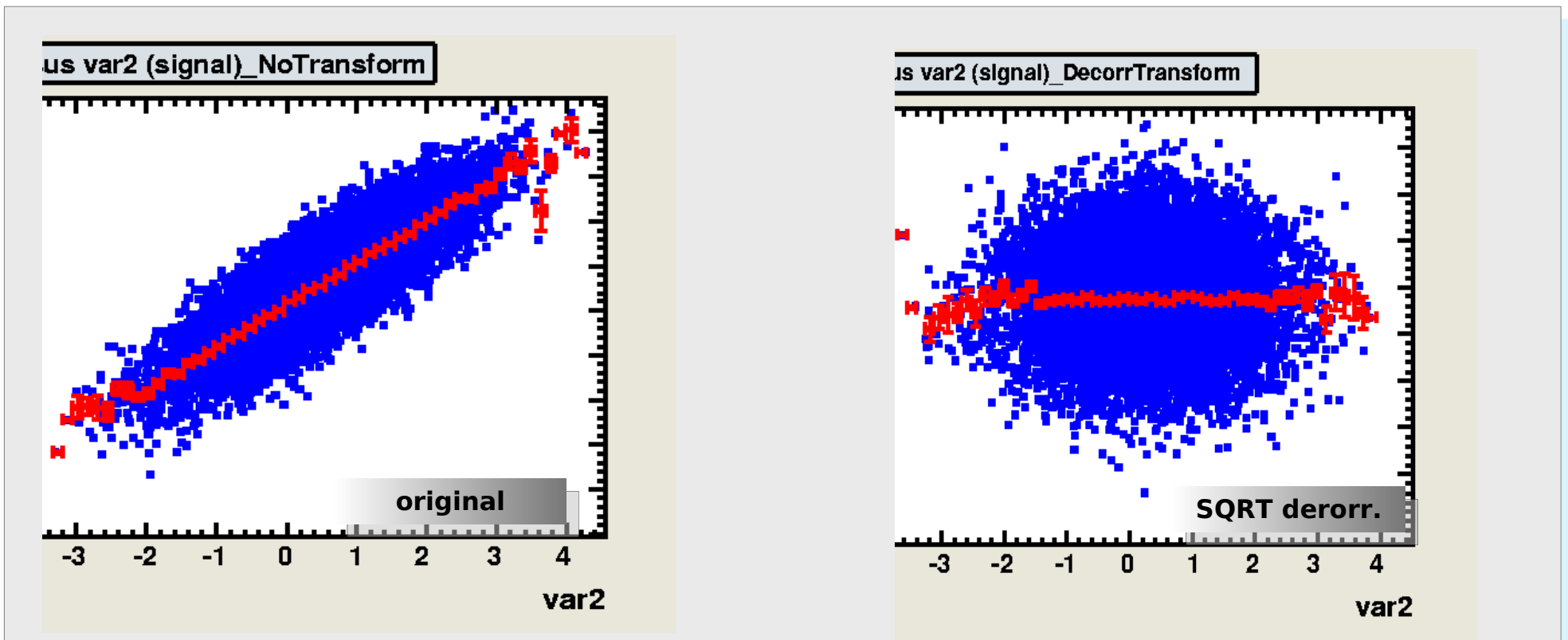
When the assumptions of LDA are satisfied, the above equation is equivalent to LDA.

Be sure to note that the vector \vec{w} is the normal to the discriminant hyperplane. As an example, in a two dimensional problem, the line that best divides the two groups is perpendicular to \vec{w} .

Generally, the data points are projected onto \vec{w} . However, to find the actual plane that best separates the data, one must solve for the bias term b in $w^T \mu_1 + b = -(w^T \mu_2 + b)$.

Decorrelation

- Removes correlation between variables by a rotation in the space of variables

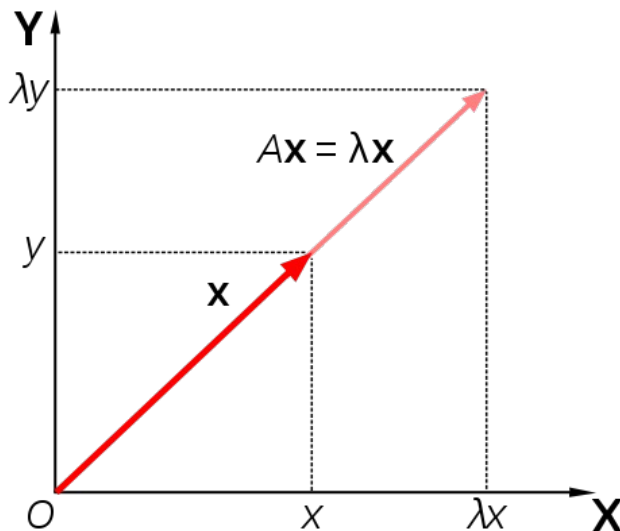


Eigenvalues and eigenvectors

In essence, an eigenvector \mathbf{v} of a linear transformation T is a non-zero vector that, when T is applied to it, does not change direction. Applying T to the eigenvector only scales the eigenvector by the scalar value λ , called an eigenvalue. This condition can be written as the equation

$$T(\mathbf{v}) = \lambda \mathbf{v}$$

referred to as the eigenvalue equation or eigenequation. In general, λ may be any scalar. For example, λ may be negative, in which case the eigenvector reverses direction as part of the scaling, or it may be zero or complex.



Matrix A acts by stretching the vector \mathbf{x} , not changing its direction, so \mathbf{x} is an eigenvector of A .

$$\begin{matrix} \mathbf{A} & & \mathbf{Q} & & \mathbf{\Lambda} & & \mathbf{Q}^{-1} \\ \left[\begin{array}{|c|} \hline \text{grid} \\ \hline \end{array} \right] & = & \left[\begin{array}{|c|} \hline \mathbf{v}_1 \\ \hline \mathbf{v}_2 \\ \hline \mathbf{v}_3 \\ \hline \end{array} \right] \left[\begin{array}{|c|} \hline \lambda_1 & 0 & 0 \\ \hline 0 & \lambda_2 & 0 \\ \hline 0 & 0 & \lambda_3 \\ \hline \end{array} \right] \left[\begin{array}{|c|} \hline \mathbf{v}_1 \\ \hline \mathbf{v}_2 \\ \hline \mathbf{v}_3 \\ \hline \end{array} \right]^{-1} \\ \text{Eigen vectors} & & \text{Eigen values} & & \text{Eigen vectors} \\ \text{of} & & \text{of} & & \text{of} \\ \mathbf{A} & & \mathbf{A} & & \mathbf{A} \end{matrix}$$

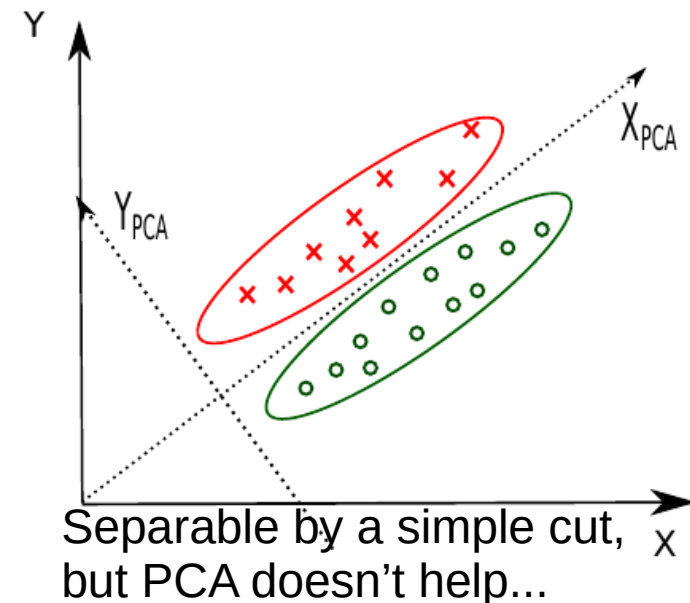
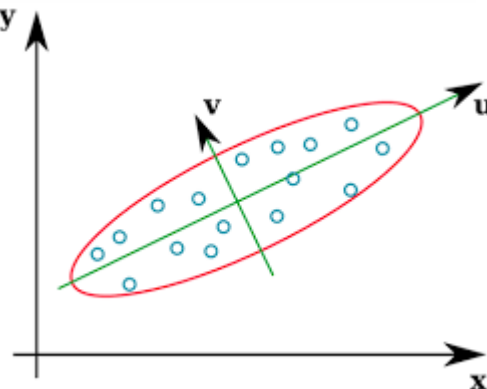
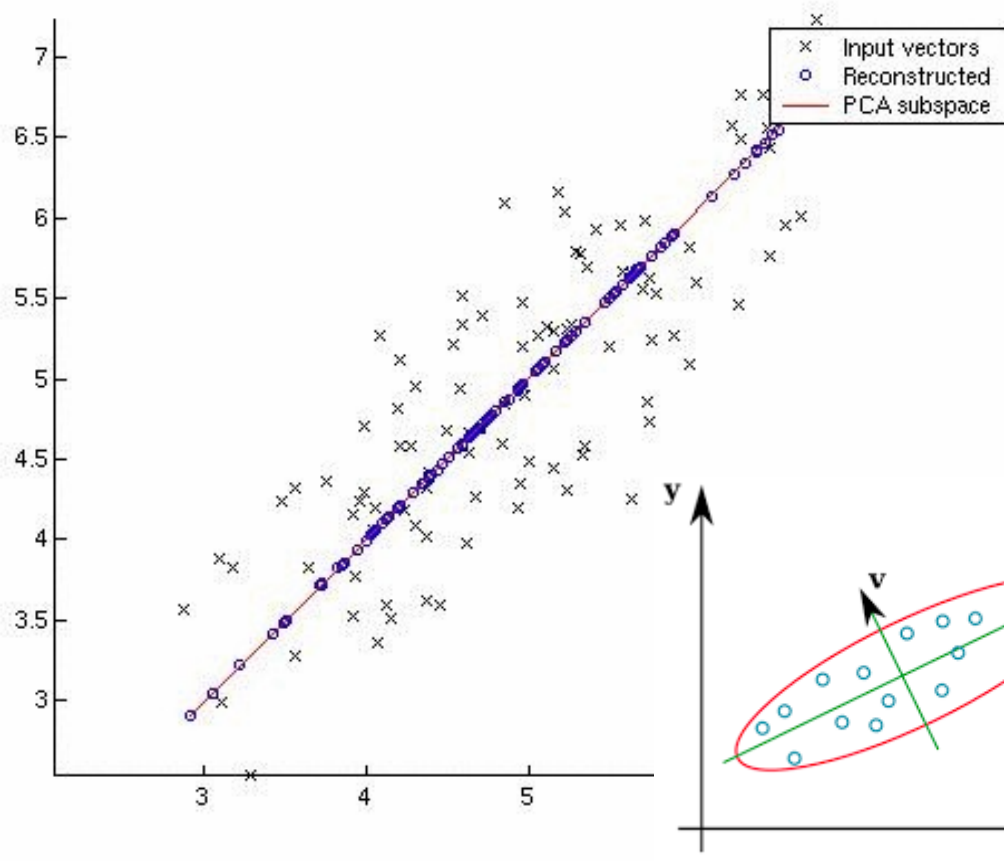
Eigendecomposition of a matrix

Principal Component Analysis - PCA

- Task: reduce the number of dimensions minimizing the loss of information
- Finds the orthogonal base of the covariance matrix, the eigenvectors with the smallest eigenvalues might be skipped

Procedure:

- Find the covariance matrix $\text{Cov}(X)$
- Find eigenvalues λ_i and eigenvectors v_i
- Skip smallest λ_i
- Unsupervised learning & dimensionality reduction



Independent Component Analysis ICA

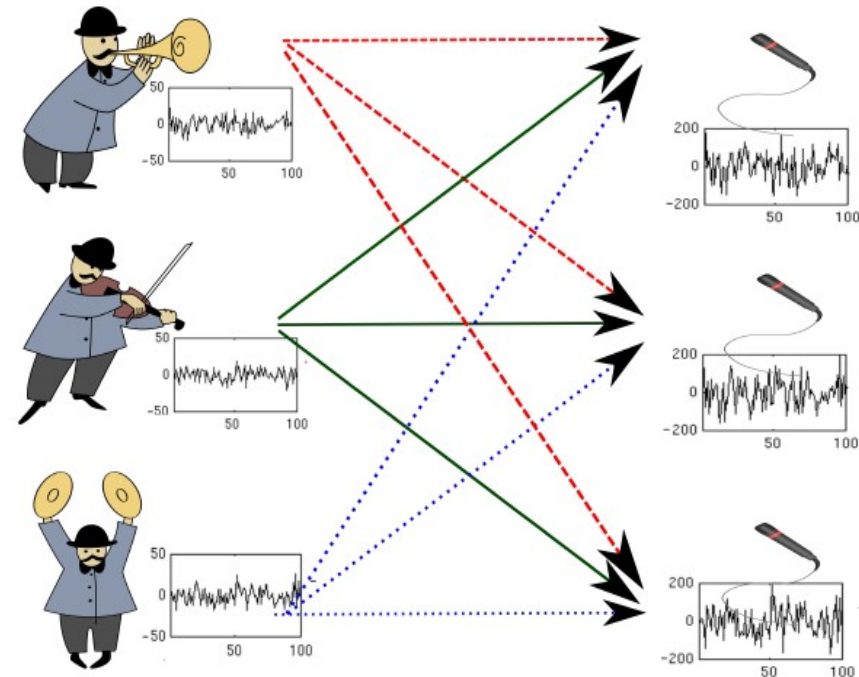
Developed at Helsinki University of Technology <http://www.cis.hut.fi/projects/ica/>

● Problem:

- Assume, that signal \mathbf{X} is a linear combination $\mathbf{X} = \mathbf{AS}$ of independent sources \mathbf{S} . The mixing matrix \mathbf{A} and vector of sources \mathbf{S} are unknown.
- **Task:** find a matrix \mathbf{T} (inverted \mathbf{A}), such that elements of vector $\mathbf{U} = \mathbf{TX}$ are statistically independent. \mathbf{T} is the matrix returning the original signals.

● Applications:

- Filtering of one source out of many others,
- Separation of signals in telecommunication,
- Separation of signals from different regions of brain,
- Signal separation in astrophysics,
- Decomposition of signals in accelerator beam analysis in FERMILAB.





How does ICA work?

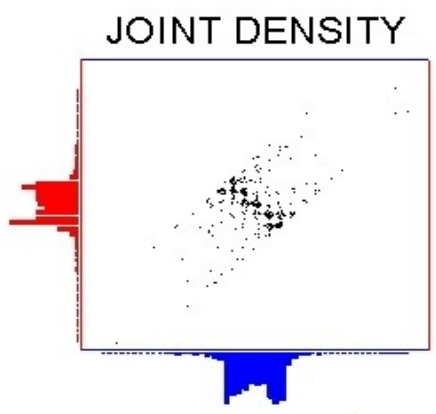
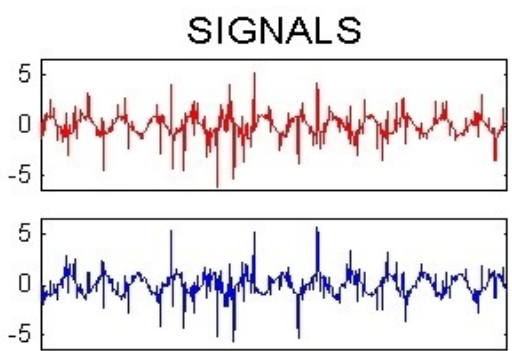
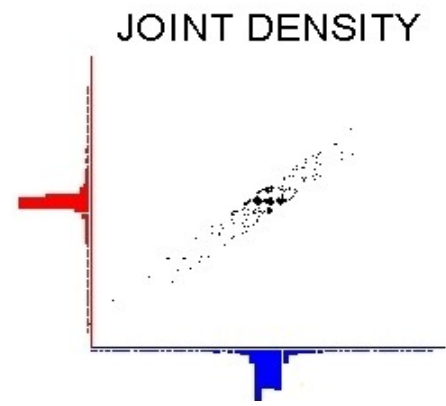
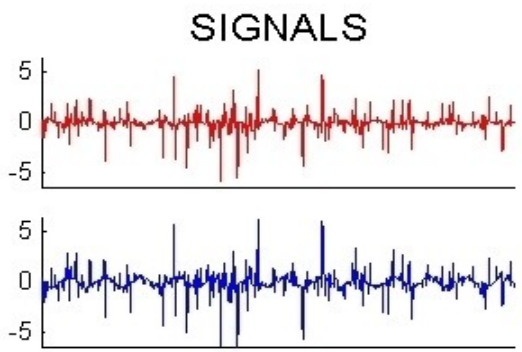
- We have two measured signals and we want to separate them into two independent sources.
- Preparing data - decorrelation (correlation coefficients equal zero, $\sigma=1$).

Superposition of many independent distributions gives Gaussian in the limit.

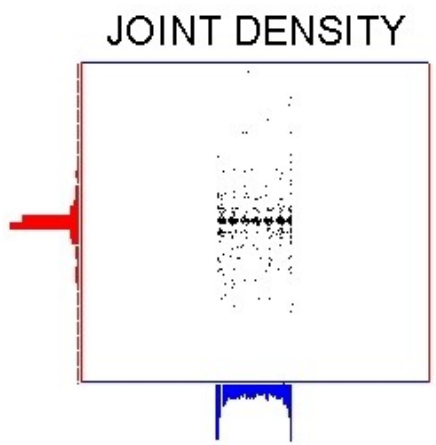
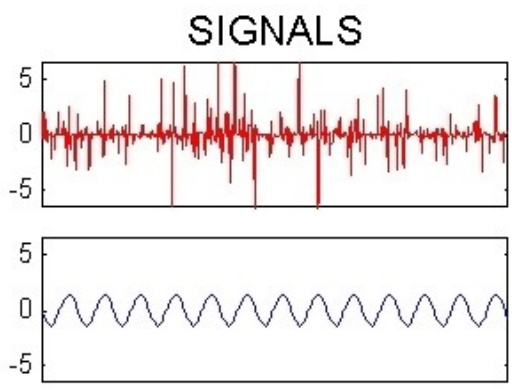
- ICA – rotation, signals should be maximally non-Gaussian (measure of non-Gaussianity might be kurtosis).

● *Kurtosis:*
$$\text{Kurt} = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \mu)^4}{\sigma^4} - 3$$

where μ is the mean of the distribution and σ is a standard deviation.

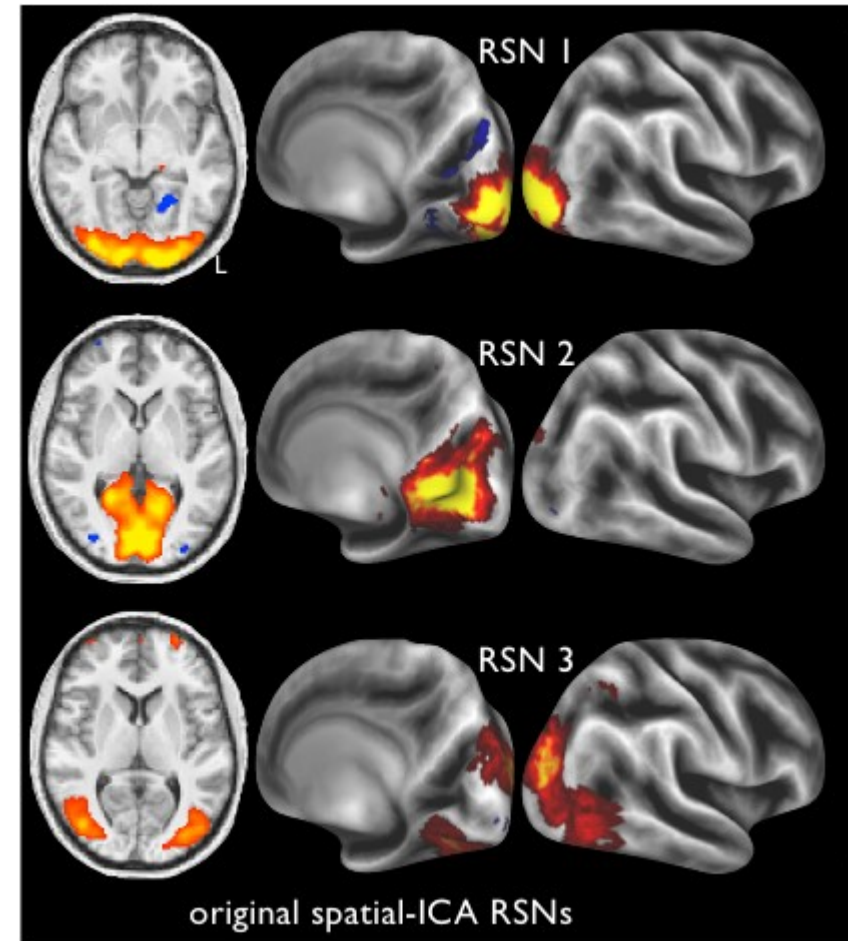
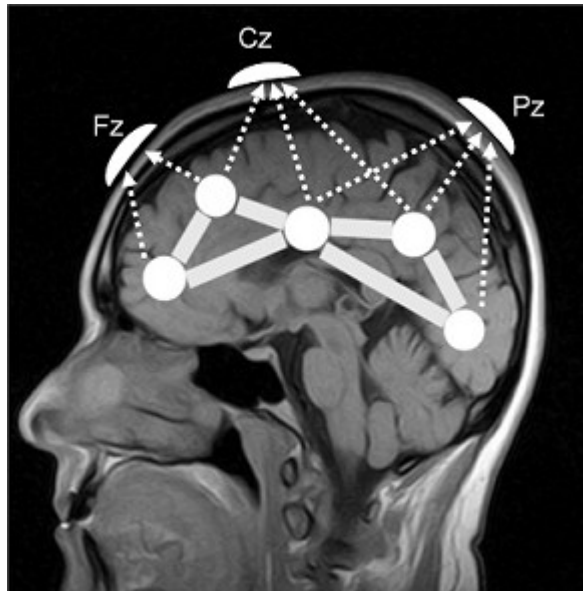


Whitened signals and density

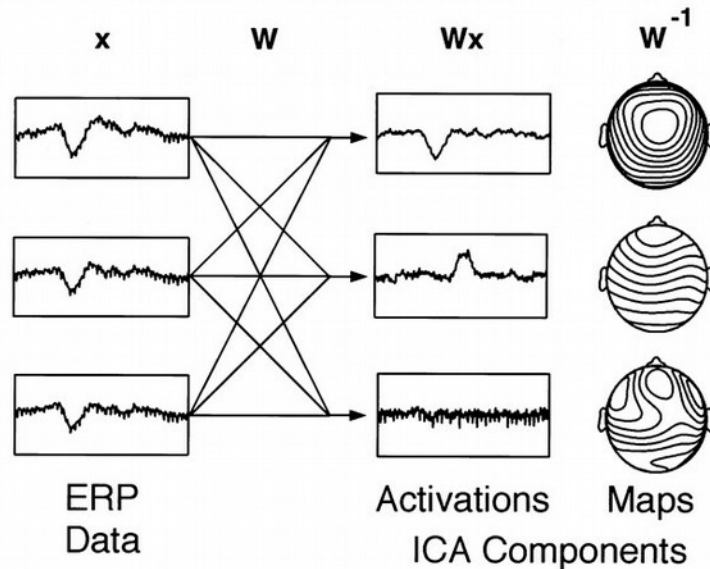


Separated signals after 5 steps of FastICA

ICA – brain research, signal separation



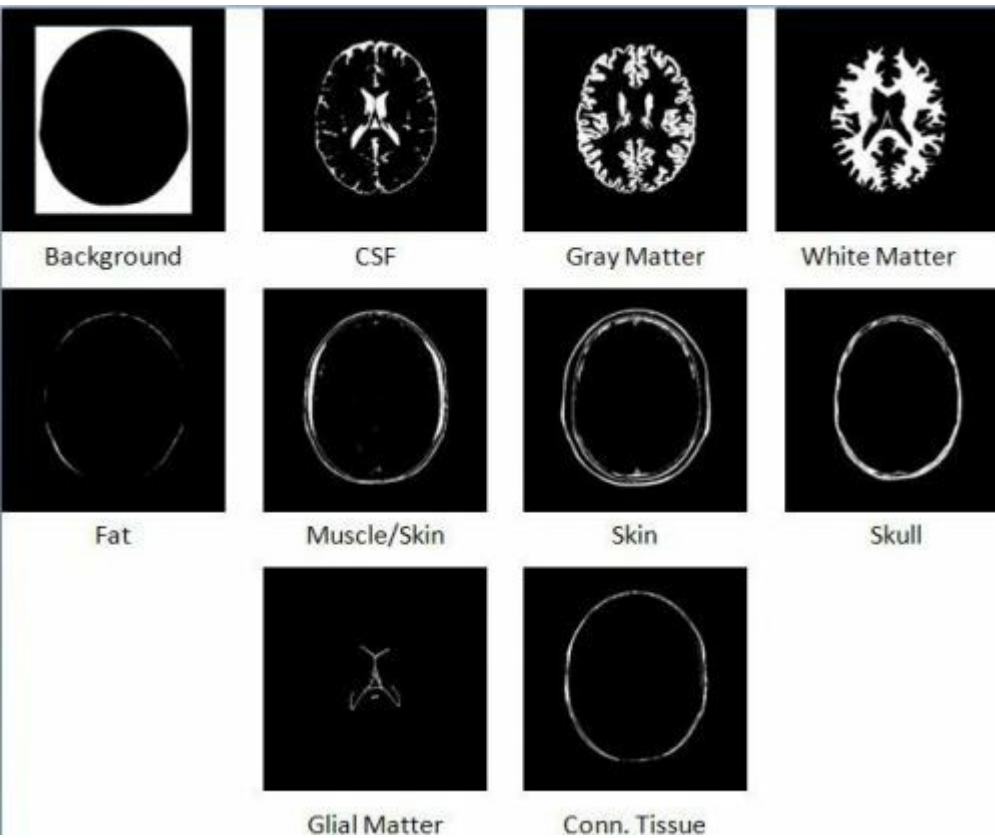
ICA Decomposition



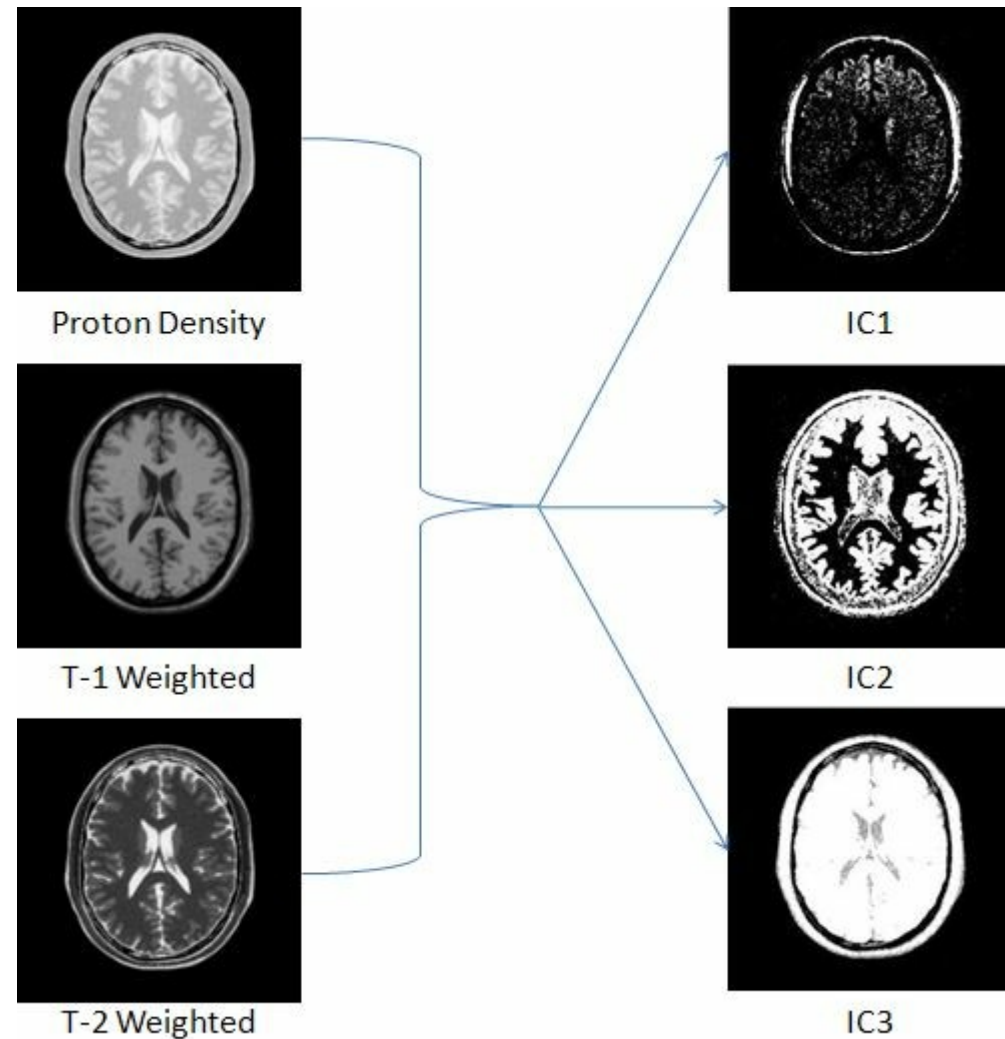
3 components from 21-dimensional decomposition using the "spatial-ICA" algorithm.

PNAS February 21, 2012 vol. 109 no. 8 3131-3136

ICA and magnetic resonance



Sources of signals

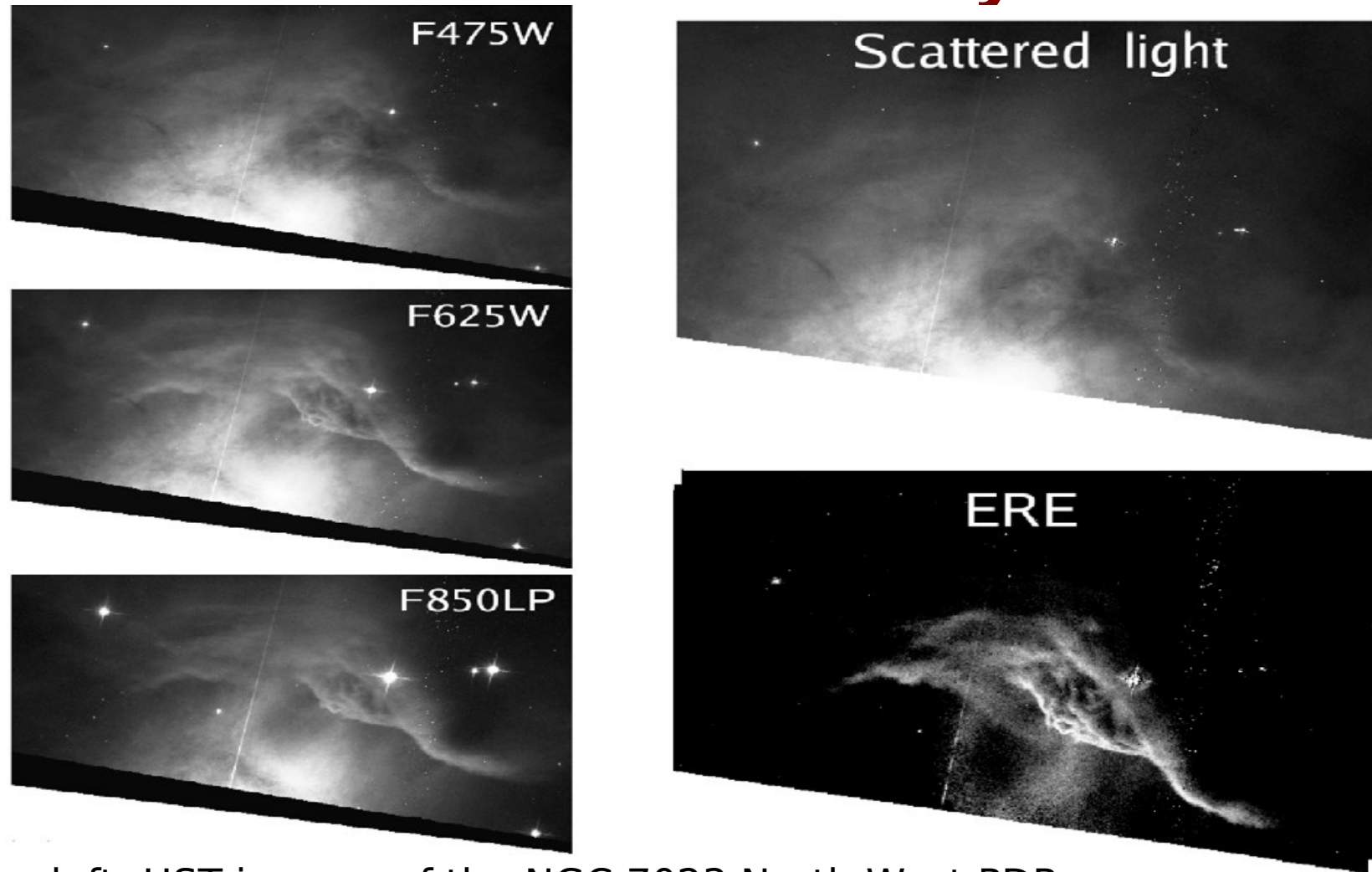


Measured

Separated components

*Blind Source Separation in Magnetic Resonance Images
January 30, 2010 by Shubhendu Trivedi*

ICA – astronomy



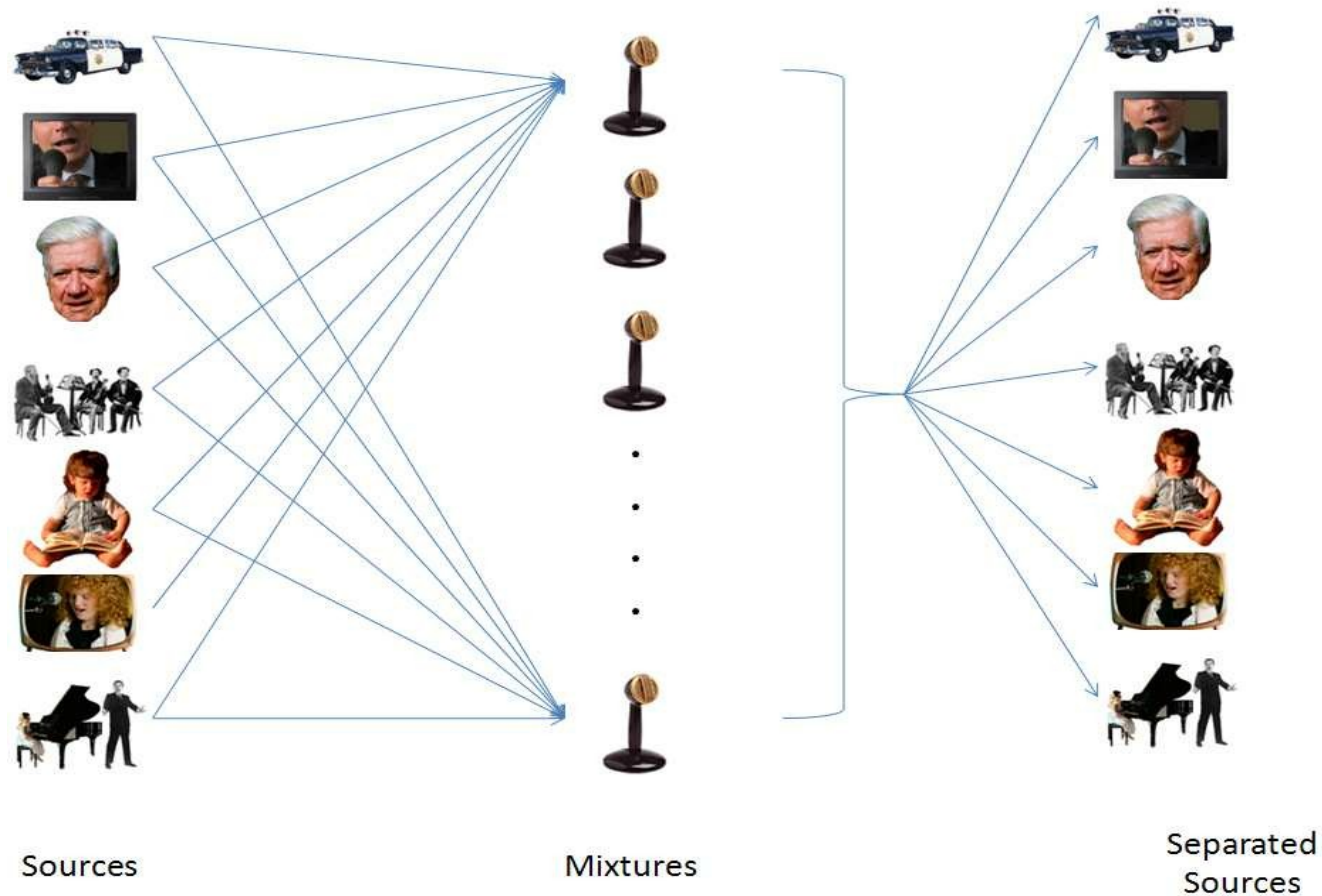
On the left: HST images of the NGC 7023 North-West PDR in three SDSS wide-band filters. On the right: scattered light and ERE (Extended Red Emission) images extracted with FastICA from the observations.

A&A 479, L41-L44 (2008)
DOI: 10.1051/0004-6361:20079158

Desert

http://cni.salk.edu/~tewon/Blind/blind_audio.html

- Cocktail Party Demo - applet showing how the the ICA algorithm works – blind separation of sound sources.



Homework for this week

- Create a **UBUNTU box** on **CLOUD** or use your linux laptop
- Install any new **ROOT version 6** (not 5!!!) from the page <https://root.cern.ch/downloading-root>
- Run any example script:
 - **C++** <https://root.cern.ch/root/html/tutorials/>
 - **Python** https://root.cern.ch/doc/v614/group__tutorial__pyroot.html
- In case of problems please contact me!
- We try to make a very short exercise after each lecture.

