European Network on New Sensing Technologies for Air Pollution Control and Environmental Sustainability - *EuNetAir*

COST Action TD1105

1ST TRAINING SCHOOL

Universitat de Barcelona, Spain, 13 - 15 June 2013

organized by UB, MIND-IN2UB - Dept. of Electronics and CSIC-IDAEA

Action Start date: 01/07/2012 - Action End date: 30/06/2016

Year 1: 2012 - 2013 (*Ongoing Action*)

UNIVERSITAT DE BARCELONA

ISP
Processat de senyal
Intel·ligent

Dr. Antonio Pardo

Function in the Action (External Expert) / apardo@el.ub.edu

ISP - Dept. d'Electrònica, Universitat de Barcelona / Spain

COST is supported by the EU Framework Programme

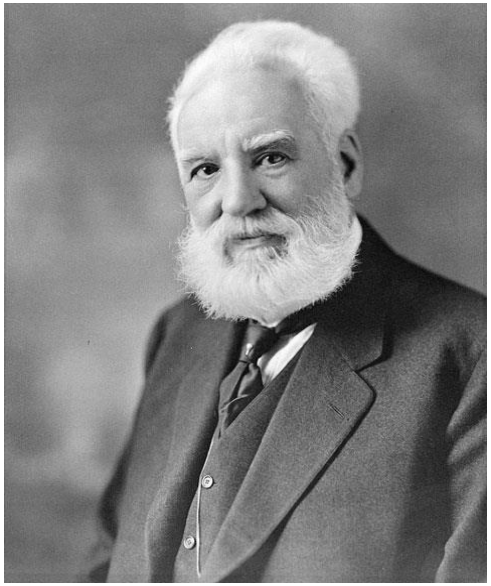
ESF provides the COST Office through a European Commission contract

Outline

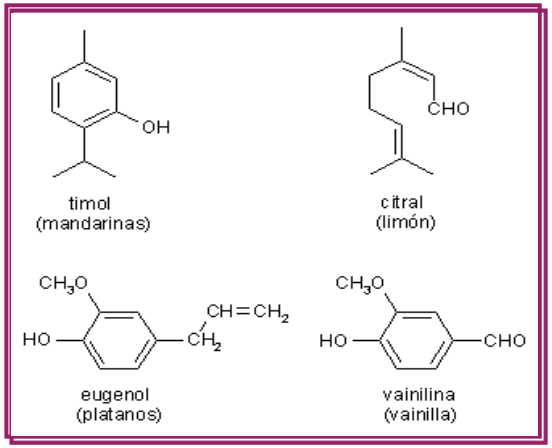
- Introduction to artificial olfaction
  - Fascinating smell
  - Sensors
  - Electronic noses
- The pattern recognition cycle
  - Pre-processing
  - Dimensionality reduction
  - Classification
  - Validation
- Conclusions

Introduction

- Did you ever measure a smell? Can you tell whether one smell is just twice strong as another? Can you measure the difference between one kind of smell and another? It is very obvious that we have very many different kinds of smells, all the way from the odor of violets and roses to asafetida. But until you can measure their likeness and differences, you can have no science of odor.
- If you are ambitious to find a new science, measure a smell.



Alexander Graham Bell (1914)



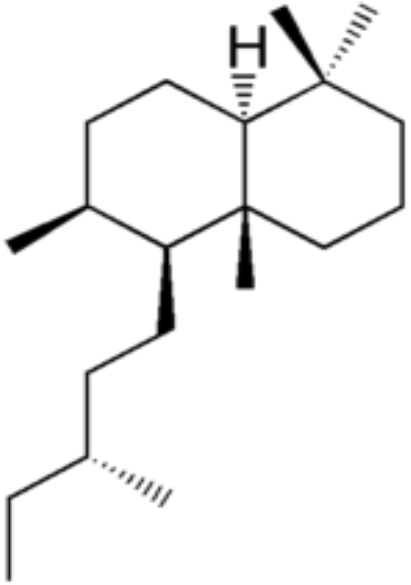
The fascinating sense of smell

- Olfaction is in many occasions considered as a second class sense compared with vision or hearing.
- Olfaction is the most primitive sense and it shows very important similarities across species.
- Olfaction is key for survival, food inspection, mate finding, etc.

- Further facts;
  - Fragrance synthesis industry is a 16 billion dollars industry dominated by few firms:
    - Dragoco (D), Firmenich (CH), Givaudan-Roure(CH), Haarman and Reimer (D), International Flavor and Fragances (US), Quest (UK), Takasago (JP)
  - Gas Sensor industry is a 9 billion dollars industry
    - Dräger(D), Smith Detection(UK), Honeywell Analytics(US), Figaro(JP), FIS(JP)

What makes an odorant?

- It has to be volatile, hydrophobic and a molecular weight less than 300 daltons.
  - The largest known odorant is labdane: 296 mw.
- Volatility falls rapidly with molecular weight, however this is not the only point since some very strong odorants are large molecules (some steroids)
- Humans may differentiate about 10.000 different odors
  - (Some perfumists claim that the capacity is infinite and that there are not two molecules that smell the same)
- The catalog of odorant molecules of major fragrance producers exceed this number.



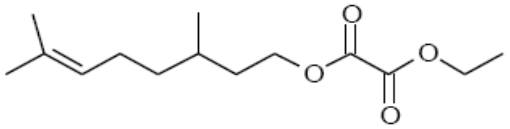
Labdane is the heaviest known odorant (mw. 296)

- **Nobel Prize in Physiology or Medicine for 2004: Richard Axel and Linda B. Buck for their discoveries of "odorant receptors and the organization of the olfactory system".**

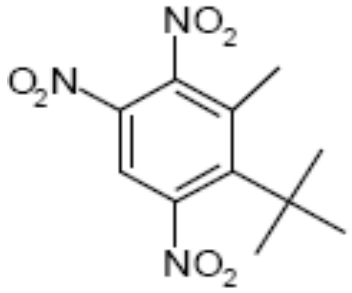
Fascinating regularities vs irregularities

- Some different molecules have pretty similar odors

Ethyl Citronellyl Oxalate

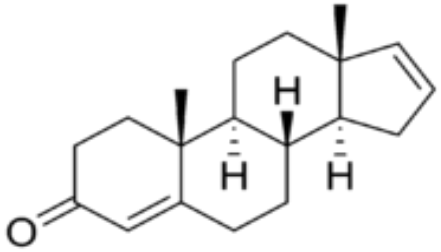


Ambrettolide



Musk

- The same molecule may produce different perceptions to different observers



Androstadienone (mw. 270)

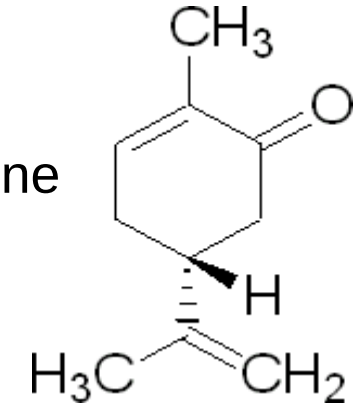
floral, pleasant

Sweat, unpleasant,

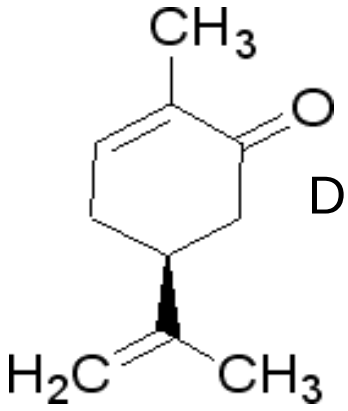
Fascinating regularities vs irregularities

- Odor perception is chiral: enantiomers produce different perceptions

L-carvone



Minty



D-carvone

Caraway

- Of a compilation of 277 enantiomers: only 5% smell identical, 60% similar, and 35% different.

L. Turin, F Yoshii, Structure-odor relations: a modern perspective, Handbook of olfaction and gustation, 2003

Mapping onto perception

- Mapping physical odor attributes onto perception is not well defined in olfaction
  - The smell of a molecule can not be predicted by its physicochemical structure
  - The physicochemical structure of a molecule can not be predicted by its smell
- The link between a perceptual space (made of verbal descriptions) and the stimulus space (made of odorants) is very complex.
  - Perfume industry and olfactory researchers use data sets (Dravnieks' Atlas of Odor Character Profiles) where chemical compounds are matched with a description of their elicited odors. (HexylButyrate → fruity, sweet, pineapple).
  - However, some efforts has been done in order to establish the link
    - A.M. Mamlouka et al, "Quantifying olfactory perception: mapping olfactory perception spaceby using multidimensional scaling and self-organizing maps", Neurocomputing 2003
    - R. M. Khan, "Predicting Odor Pleasantness from Odorant Structure: Pleasantness as a Reflection of the Physical World" The Journal of Neuroscience, 2007

Global vs Analytical Odor assessment

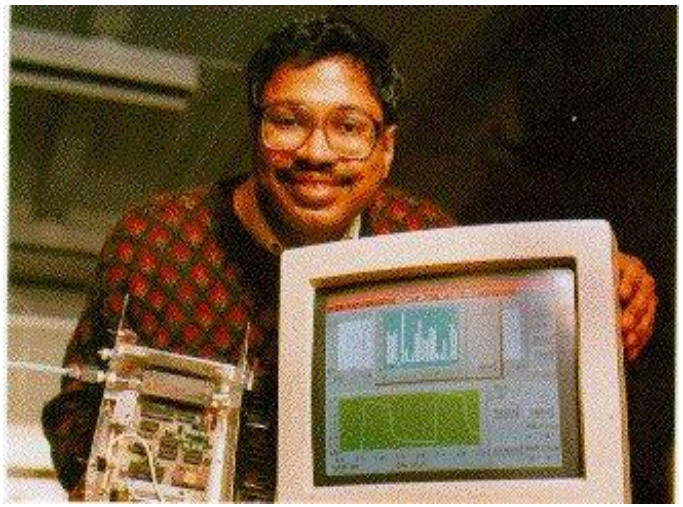
- Analytical Chemistry is dominated by following philosophy:
  - Identify and quantify all components in an odor
  - This requires powerful analytical instrumentation

- Is that always necessary?? Many tasks do not really require that depth
  - Sometimes we just want to differentiate two odors:
    - Is there any difference between A and B? Quality Control
  - Sometimes we need to classify a new odor into a set of reference groups
    - What is our new fragance provider providing us?

- Some others need speed:
  - Alarms,
  - On-line food quality control,…

- Some odors are specially complex (in particular, natural products)
  - Coffee Aroma:
    - Facts:
      - More than 900 chemicals have identified in coffee headspace
      - Good Synthetic Coffee aromas use 20-30 components and even though they are still smell ‘synthetic’
      - Among them not a single molecule smells as ‘coffee aroma’
        - R.A. Buffo, C. Cardelli-Freire, Coffee Flavour: An Overview, Flavour and Fragance Journal (2004)

The artificial olfaction



"… As a test of this hypothesis we have constructed an electronic nose using semiconductor transducers …
… shows that discrimination in an olfactory system could be achieved without the use of highly specific receptors…."

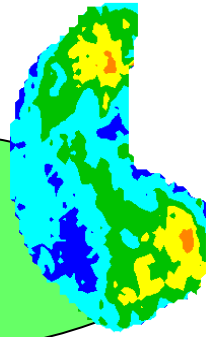
Persaud K, Dodd G. ***Analysis of discrimination mechanisms in the mammalian olfactory system using a model nose*****.**
NATURE, (1982 Sep 23) 299 (5881) 352-5

Multidisciplinarity

Bio-Nano-Micro-Technology

Basic Science

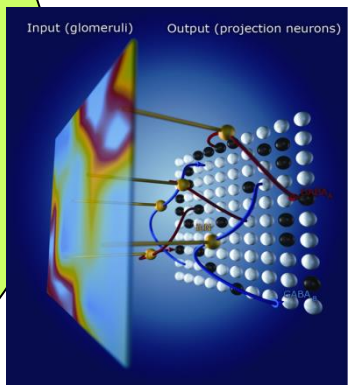
Neuroscience



Sensor Technology

Artificial Olfaction

Computational Neuroscience

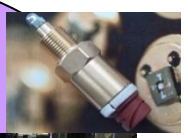


Smart Chemical Instrumentation

Intelligent Signal Processing

$$E(\vec{x}) = \frac{1}{2n}\sum_{\mu=1}^{N}\left[K(\vec{x},\vec{\xi}^{\,\mu})\right]^2 - NK(\vec{x},\vec{x})$$

Applications Medical / Industrial



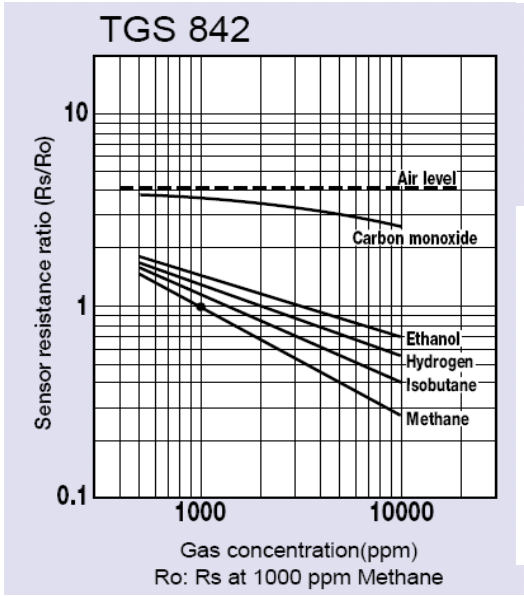
Applied Science / Engineering

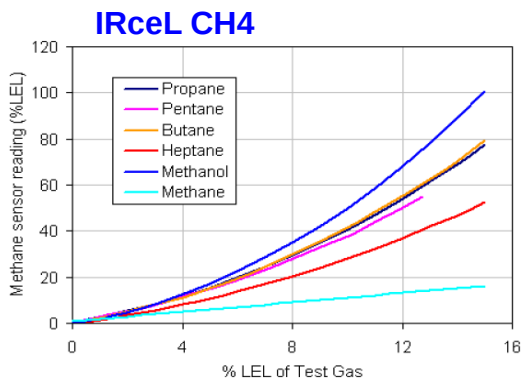
Sensors: Selectivity problem

- There are several gas sensor technologies:
  - MOX, (Metal Oxide)
  - QMB, (Quartz Microbalance)
  - SAW, (Surface Acustic Wave)
  - ElectroChemical
  - Pellistors
  - InfraRed …

- Based on different magnitude variation
  - Resistance / Impedance
  - Current
  - Work function
  - Capacitance
  - Mass
  - Temperature
  - Optical Absorption

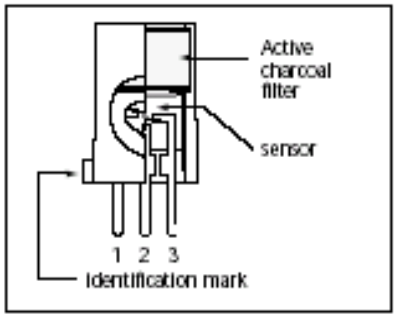
- **There isn't a perfect specific sensor**
  - Detectors suffer false alarms

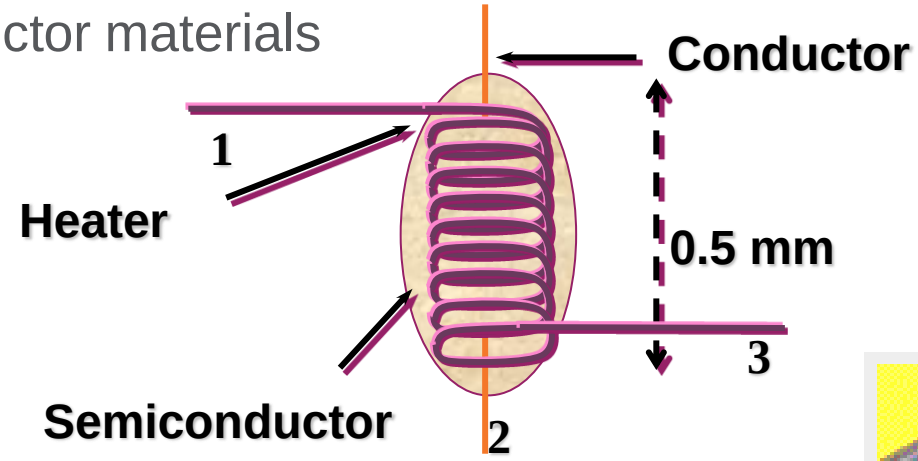


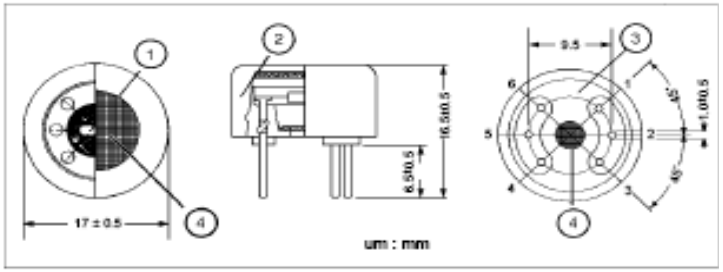


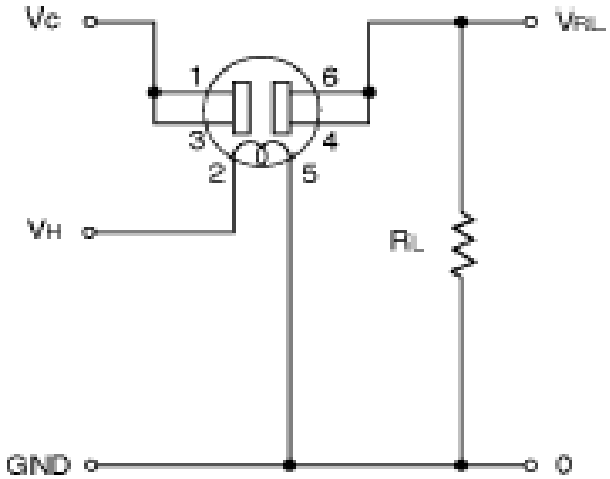
Metal OXide

- Changes in the electrical resistance
  - Based on semiconductor materials

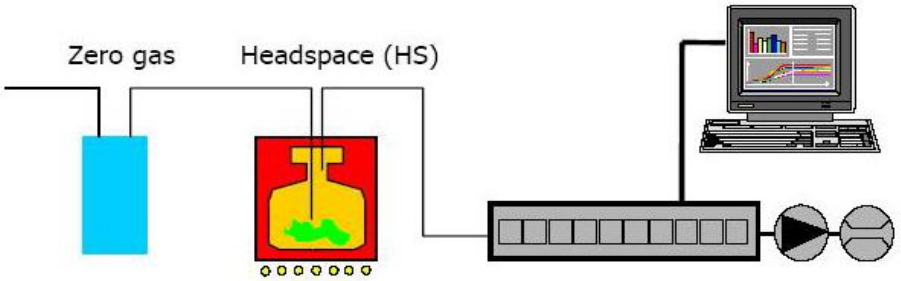


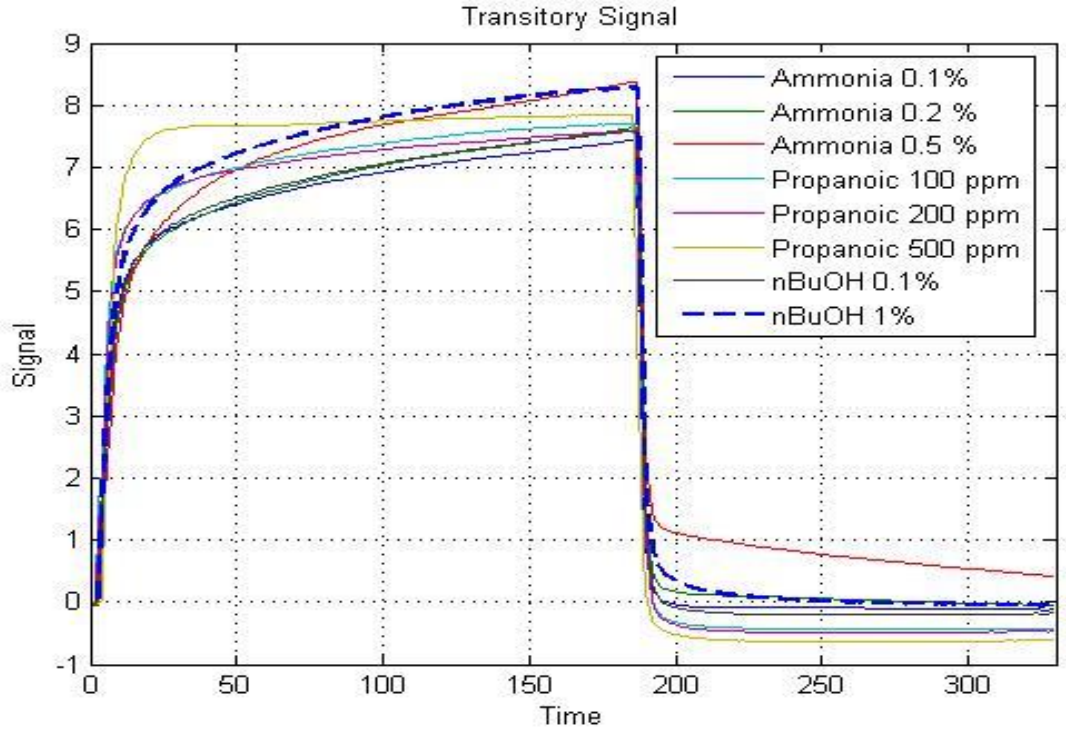






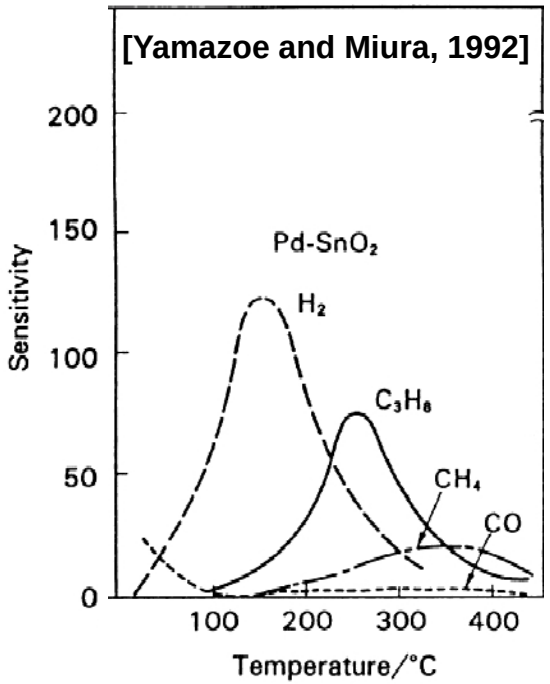
Sensor Signals

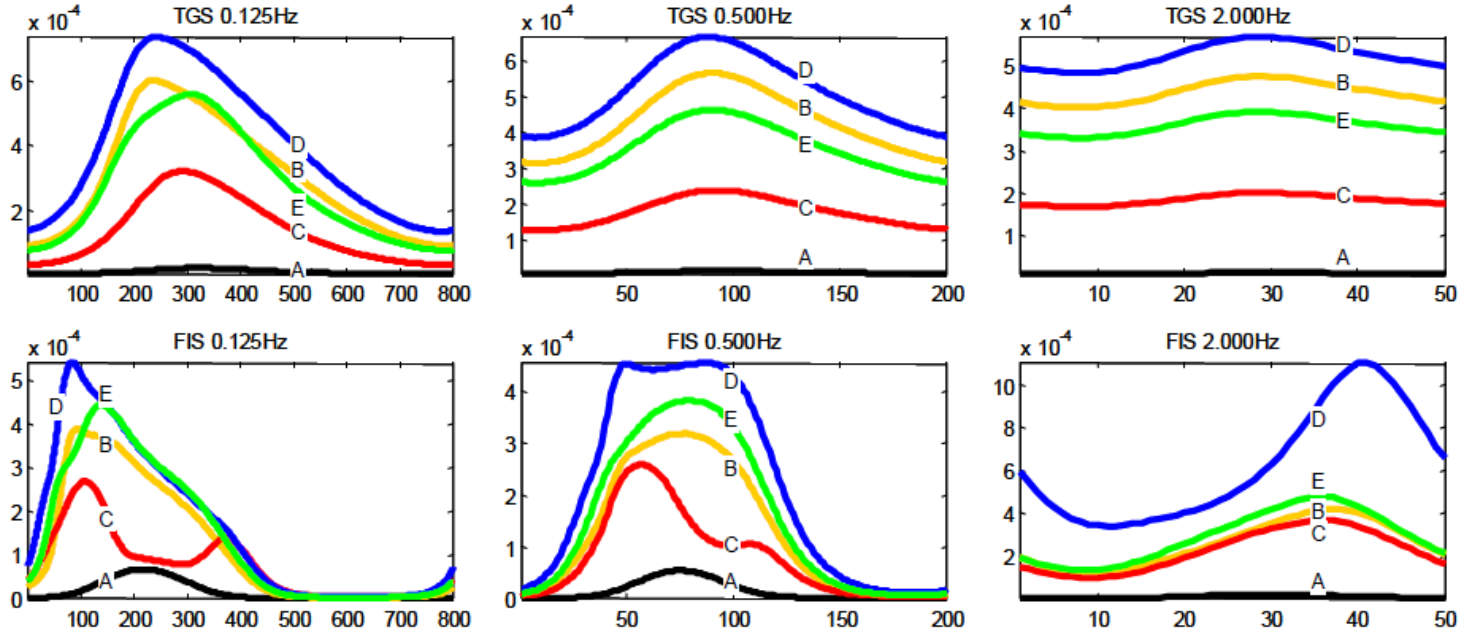




Sensor signals

- Temperature modulation →selectivity of MOS layers depends on the operating temperature
  - Isothermal control: maintain constant temperature
  - Temperature modulation: capture sensor response while changing temperature



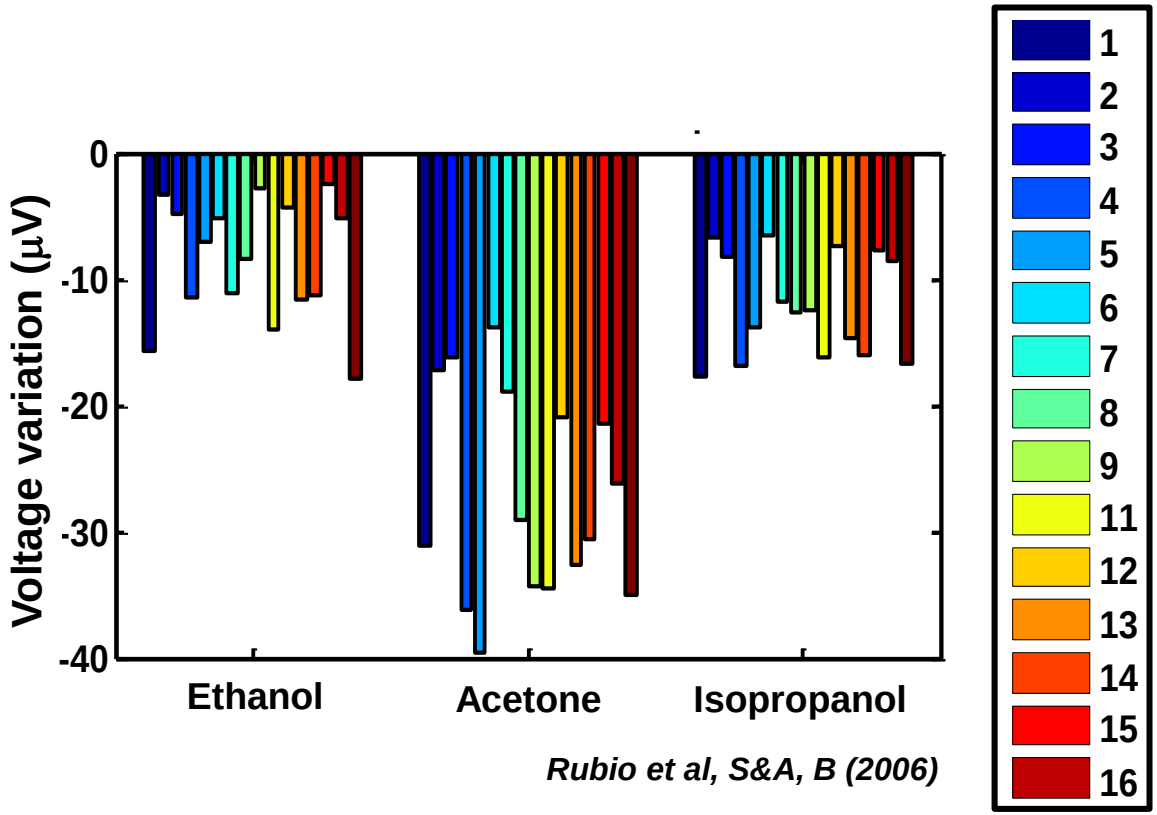


Average conductance of TGS and FIS at various heater frequencies to: air (A), acetone (B), ammonia (C), IPA (D) and vinegar (E)

[Osuna, Proceedings of the 8th Intl. Symp. On Olfaction and Electronic Nose, 2001]

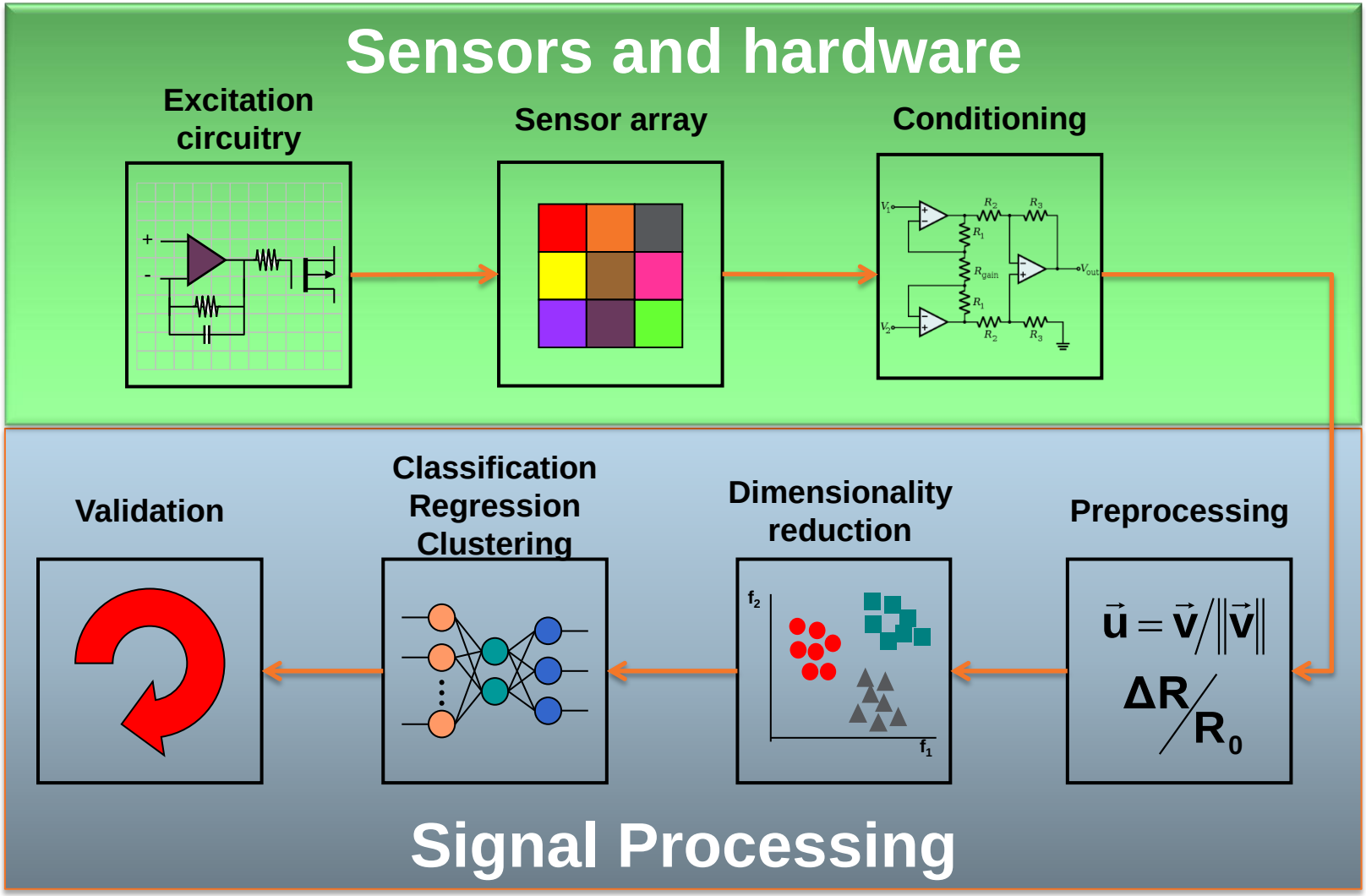
What is an e-nose

- An e-nose is an instrument which combines
  - an **array of chemical sensors** with partial and overlapping specificities
  - a **pattern-recognition system** capable of processing the multivariate response across sensors



Rubio et al, S&A, B (2006)

The pattern recognition cycle



R. Gutierrez-Osuna, IEEE Spectrum, 1998

Signal Conditioning

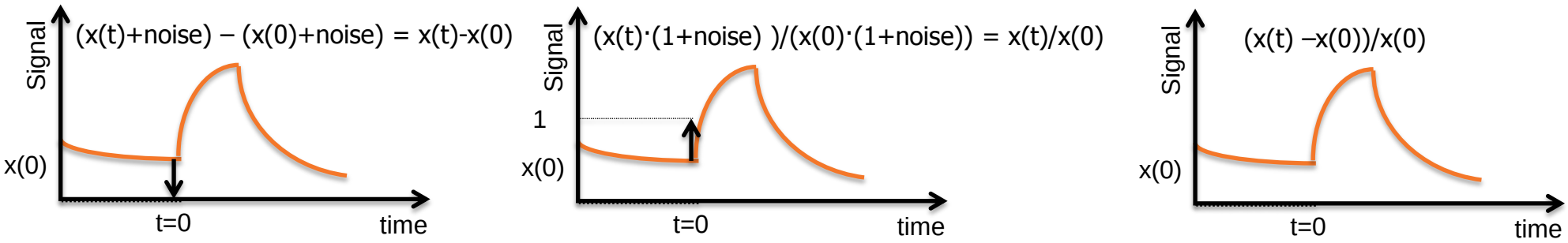
- The electrical signals generated by sensors are often not adequate and must be further processed by a number of analog signal conditioning hardware.
  - Amplification
    - Gain to adapt the signal level
  - Filtering
    - Remove unwanted frequency components (low-pass, high-pass, band pass, band rejects)
  - Compensation
    - Special functions with analog circuits to compensate deficiencies: linearization, logarithmic amplification, temperature compensation,…

Signal Pre-processing

- The goal → prepare the data for multivariate pattern analysis. It is critical and can have a significant impact on the performance of next steps in pattern analysis.
  - Signal pre-processing is somewhat dependent on the underlying sensor technology, but two stages are common:
  - **Baseline correction** → Transform the sensor response relative to its baseline for the purposes of contrast enhancement, scaling and/or drift compensation
  - **Normalizatio**n → there are many sources of systematic variation (for example, concentration of the analyte). Normalization attempts to remove such variation in order to make different measurements comparable

Signal Pre-processing:Baseline correction

- Three baseline manipulation methods are commonly employed:
  - **Difference** → subtracts the baseline and can be used to eliminate additive drift from the sensor response
  - **Relative** → divides by the baseline, removing multiplicative drift, and generating a dimensionless response
  - **Fractional** → subtracts and divides by the baseline, generating dimensionless and normalized responses → this is common for MOX sensors



Handbook of machine olfaction. Wiley-VCH. Editedby T.C. Pearce, S.S.Schiffman, H.T. Nagle and J.W. Gardner

Signal Pre-processing: Normalization

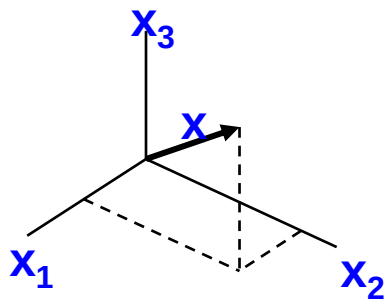
- Two normalization techniques can be distinguish:
  - Local methods → operate across the sensor array for each individual measure in order to compensate for sample-to-sample variations caused by concentration or sensor drift, among others.
    - **vector normalization** → each measure is divided by its norm
  - Global methods → used to ensure that sensor magnitudes are comparable, preventing subsequent pattern-recognition procedures from being overwhelmed by sensors with arbitrarily large values.
    - **sensor autoscaling** → measurement are set to mean = 0 and $\sigma = 1$ → robust to outliers but can not provide tight bounds
    - **sensor normalization** →in which the range of values for each individual feature is set to [0,1] → sensitive to outliers
      - Both can amplify noise→(all the sensors (even those who do not contain information)are weighted equally)

From signal to features and patterns

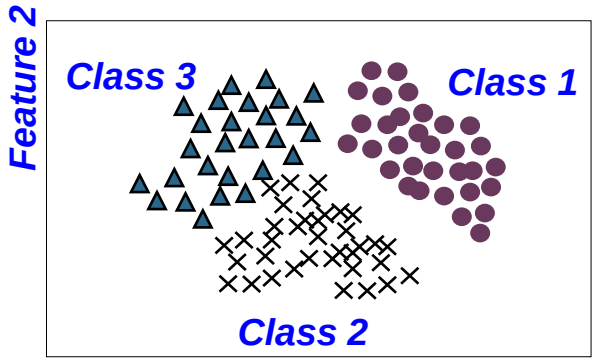
- Feature → is any distinctive aspect, quality or characteristic
  - Features may be symbolic (i.e., color) or numeric (i.e., height)
- The combination of d features is represented as a d-dimensional vector called a *feature vector*
- The d-dimensional space defined by the feature vector is called *feature space*
- Objects are represented as points in feature space. This representation is called a *scatter plot*

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \\ x_d \end{bmatrix}$$

Feature vector



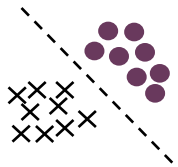
Feature space (3D)



Scatter plot (2D)

Features, patterns and classifiers

- Pattern
  - Pattern is a composite of traits or features characteristic of an individual
  - In classification, a pattern is a pair of variables $\{x, \omega\}$ where
    - ***x*** is a collection of observations or features (feature vector)
    - ω is the concept behind the observation (label)
- What makes a "good" feature vector?
  - The quality of a feature vector is related to its ability to discriminate examples from different classes
    - Examples from the same class should have similar feature values
    - Examples from different classes have different feature values

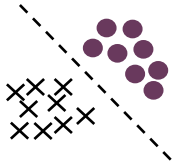


"Good" features

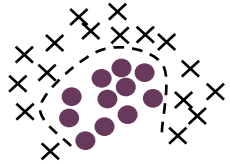
"Bad" features

Features, patterns and classifiers

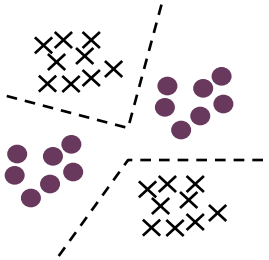
- More feature properties



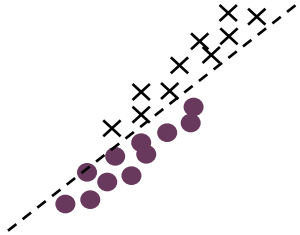
Linear separability



Non-linear separability

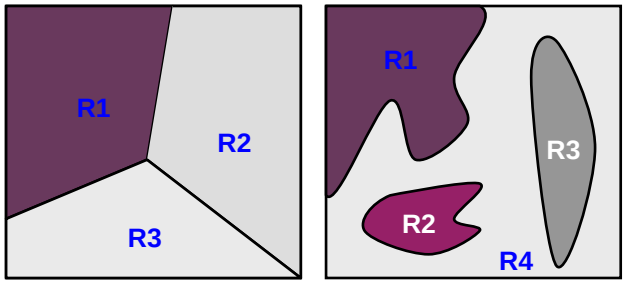


Multi-modal



Highly correlated features

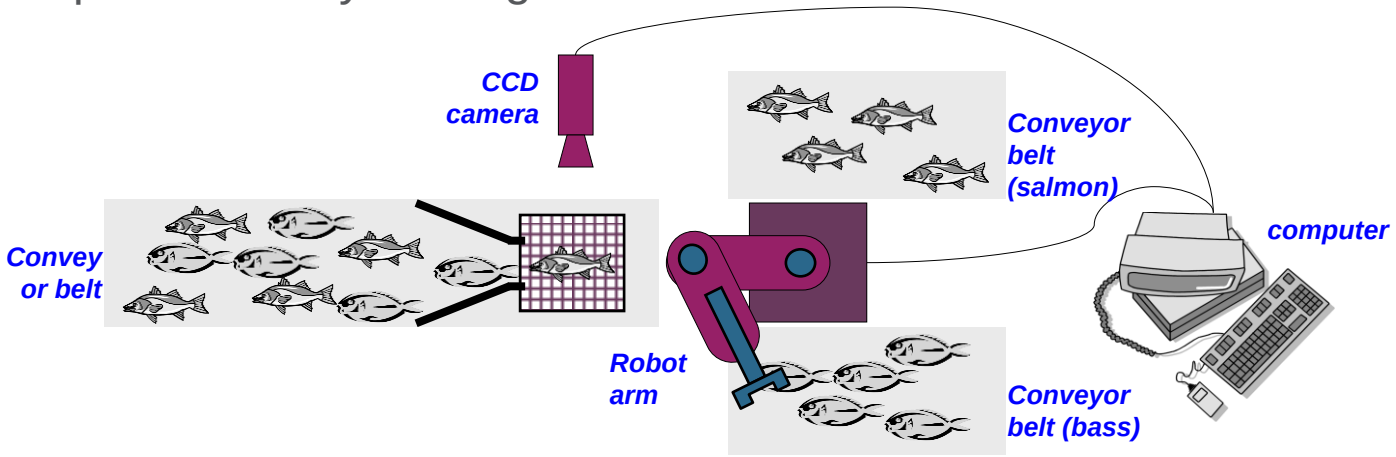
- Classifiers
  - The goal of a classifier is to partition feature space into class-labeled **decision regions**
  - Borders between decision regions are called **decision boundaries**



Ricardo Gutierrez-Osuna
Texas A&M University

An example

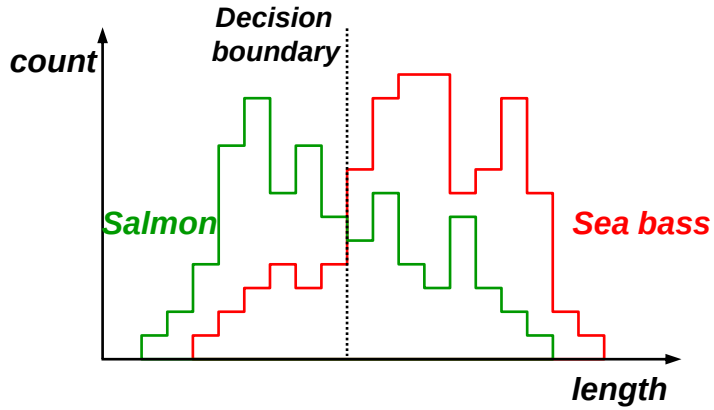
- Consider the following scenario*
  - A fish processing plan wants to automate the process of sorting incoming fish according to species (salmon or sea bass)
  - The automation system consists of
    - a conveyor belt for incoming products
    - two conveyor belts for sorted products
    - a pick-and-place robotic arm
    - a vision system with an overhead CCD camera
    - a computer to analyze images and control the robot arm



*Adapted from Duda, Hart and Stork, Pattern Classification, 2nd Ed.

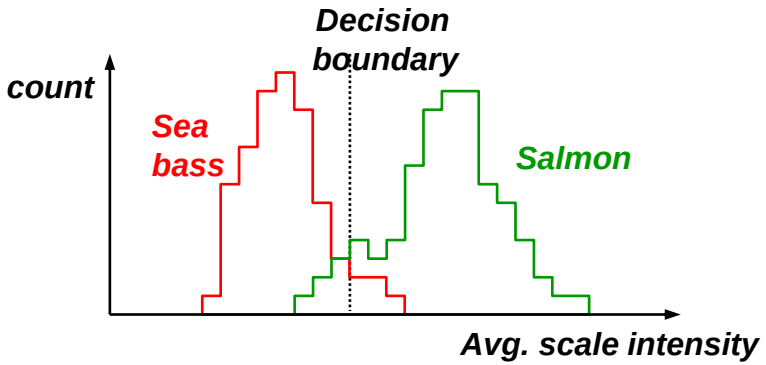
An example

- Sensor
  - The camera captures an image as a new fish enters the sorting area
- Preprocessing
  - Adjustments for average intensity levels
  - Segmentation to separate fish from background
- Feature Extraction
  - Suppose we know that, on the average, sea bass is larger than salmon
- Classification
  - Collect a set of examples from both species
    - Plot a distribution of lengths for both classes
  - Determine a decision boundary (threshold) that minimizes the classification error
    - We estimate the system's probability of error and obtain a discouraging result of 40%
  - **What is next?**

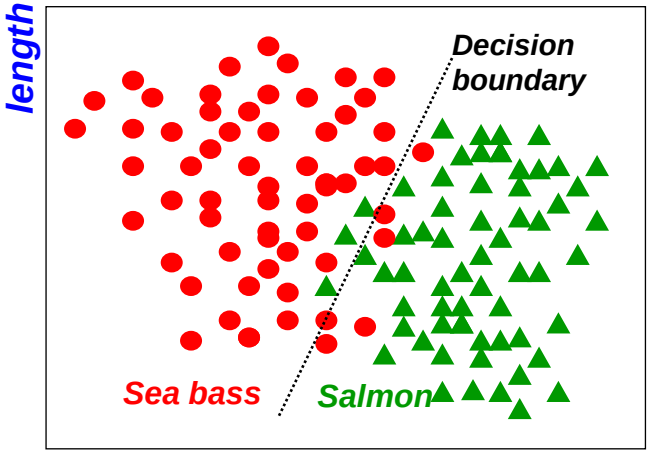


An example

- Improving the performance of our PR system
  - Committed to achieve a recognition rate of 95%, we try a number of features
    - Width, Area, Position of the eyes w.r.t. mouth...
    - only to find out that these features contain no discriminatory information
  - Finally we find a “good” feature: average intensity of the scales

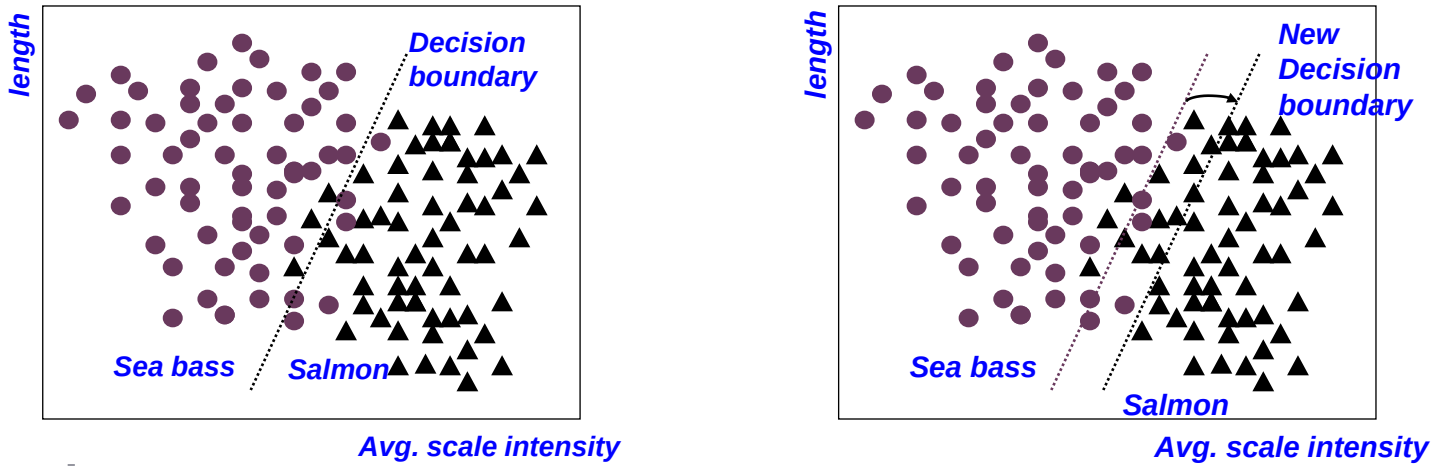


- We combine “length” and “average intensity of the scales” to improve class separability
- We compute a linear discriminant function to separate the two classes, and obtain a classification rate of 95.7%



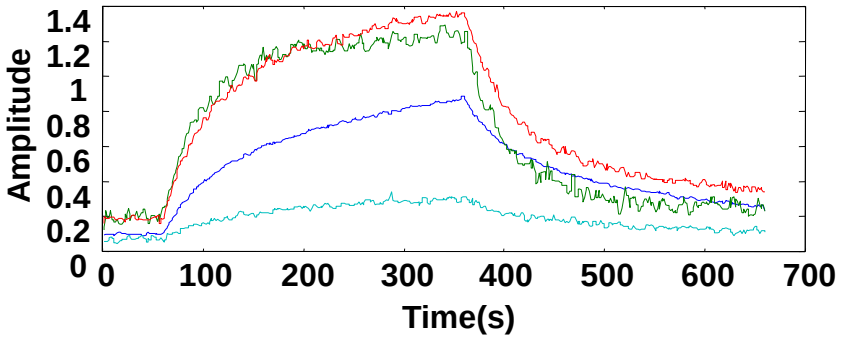
An example

- Cost Versus Classification rate
  - Is classification rate the best objective function for this problem?
    - The **cost** of misclassifying salmon as sea bass is that the end customer will occasionally find a tasty piece of salmon when he purchases sea bass
    - The **cost** of misclassifying sea bass as salmon is a customer upset when he finds a piece of sea bass purchased at the price of salmon
  - We could intuitively shift the decision boundary to minimize an alternative cost function



The curse of dimensionality

- Initial Feature Space Definition:
  - N: number of Sensors
  - T_s: Sampling Time
  - T_{tot}: Measurement time

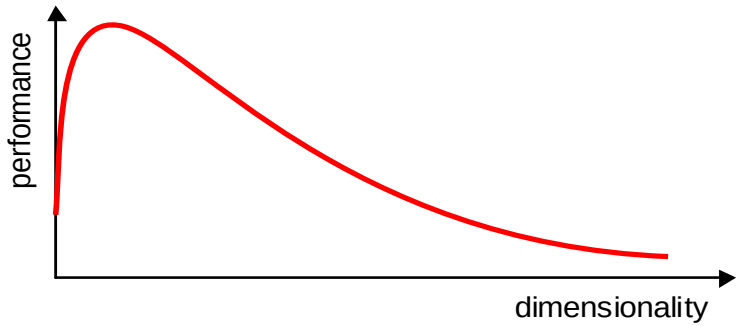


- The Initial Feature Space is given by the concatenation of the sampled signals to form a vector of dimension: $N*T_{tot}/T_s$.
- Some typical numbers:
  - T_s: 0.01s, T_{tot}: 1200s, N=8
  - Dimension= $9.6 \cdot 10^5$

Dimensionality reduction

- The “curse of dimensionality” [Bellman, 1961]
  - Refers to the problems associated with multivariate data analysis as the dimensionality increases
  - The performance of most classifiers degrades with the addition of irrelevant features.

- In practice, the curse of dimensionality means that
  - For a given sample size, there is a maximum number of features above which the performance of our classifier will degrade rather than improve
    - In most cases, the information that was lost by discarding some features is compensated by a more accurate mapping in lower-dimensional space

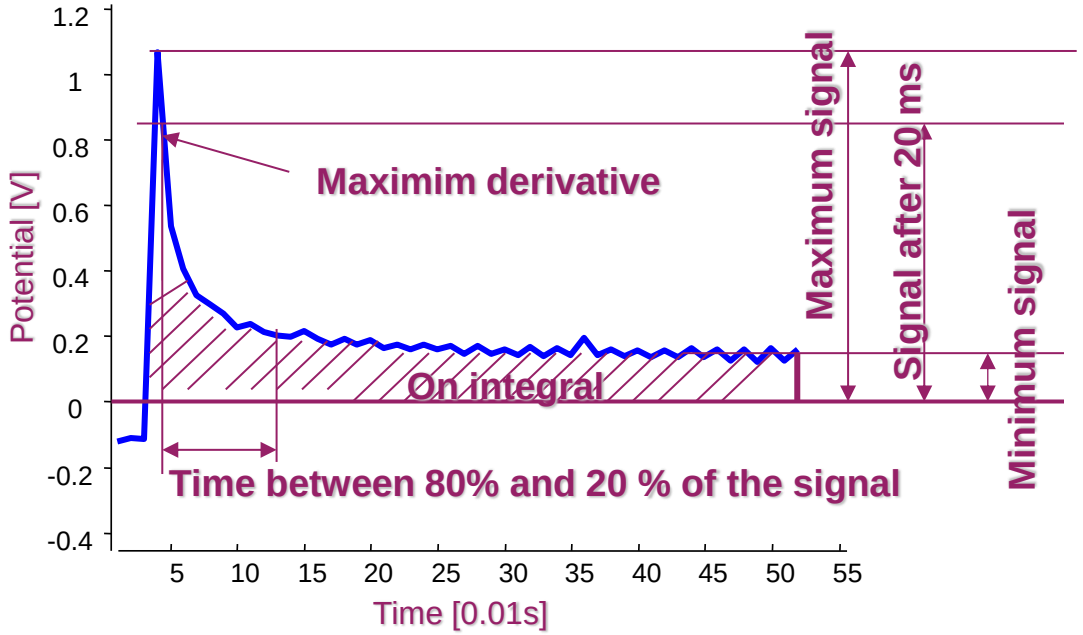


- **<u>A dimensionality reduction step is usually needed</u>**

Dimensionality reduction

- One option →Reduce dimensionality by heuristic methods
  - by extracting a single parameter (e.g. steady-state, final or maximum response) from each sensor, disregarding the initial transient response.
  - However transient analysis can significantly improve the performance of gas sensor arrays

- By this way we will reduce the dimensionality and we will build a feature vector made from magnitudes extracted or selected from the signal using the expertise of an engineer with previous knowledge
- Lets explore other non-heuristic alternatives



Dimensionality reduction

- Two approaches to perform dim. reduction $\Re^N \rightarrow \Re^M$ (M<N)
  - **Feature selection**: choosing a subset of all the features

$$[x_1\ x_2 \ldots x_N] \xrightarrow{\text{feature selection}} [x_{i_1}\ x_{i_2} \ldots x_{i_M}]$$

  - **Feature extraction**: creating new features by combining existing ones

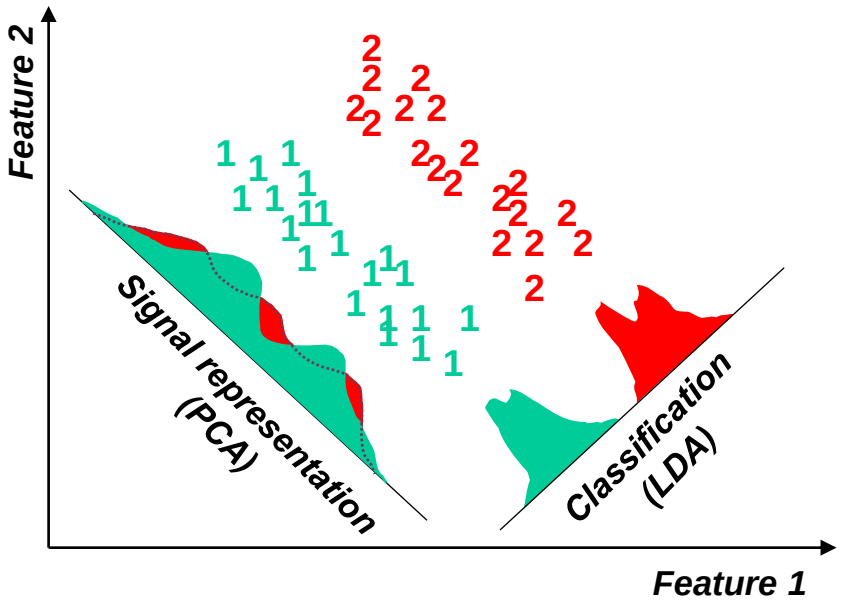
$$[x_1\ x_2 \ldots x_N] \xrightarrow{\text{feature extraction}} [y_1\ y_2 \ldots y_M] = f([x_{i_1}\ x_{i_2} \ldots x_{i_M}])$$

    - In either case, the goal is to find a low-dimensional representation of the data that preserves (most of) the information or structure in the data
- Linear feature extraction
  - The "optimal" mapping y=f(x) is, in general, a non-linear function whose form is problem-dependent
    - Hence, feature extraction is commonly limited to linear projections **y=Wx**

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \xrightarrow{\text{linear feature extraction}} \begin{bmatrix} y_1 \\ y_2 \\ \\ y_M \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1N} \\ w_{21} & w_{22} & \cdots & w_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ w_{M1} & w_{M2} & & w_{MN} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \\ x_N \end{bmatrix}$$

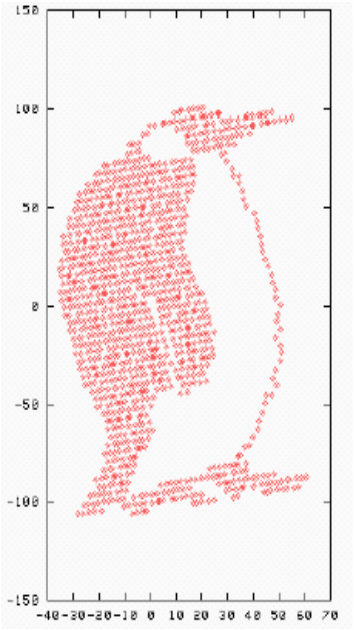
Signal representation vs classification

- Two criteria can be used to find the “optimal” feature extraction mapping y=f(x)
  - **Signal representation**: The goal of feature extraction is to represent the samples accurately in a lower-dimensional space
  - **Classification**: The goal of feature extraction is to enhance the class-discriminatory information in the lower-dimensional space
- Within the realm of linear feature extraction, two techniques are commonly used
  - Principal Components (PCA)
    - Based on signal representation
  - Fisher’s Linear Discriminant (LDA)
    - Based on classification



Principal Components Analysis

- What is PCA → A method of analyzing multivariate data in order to express their variation in a minimum number of new uncorrelated variables (principal components).
  - PCA simply performs a coordinate rotation that aligns the transformed axes with the directions of maximum variance



The concept of variance is very important. It is a fundamental assumption that the directions of maximum variance are directly related with the hidden phenomena we want to discover.

BUT, there is no guarantee that the directions of maximum variance will contain good features for discrimination

This example shows a projection of a three-dimensional data set into two dimensions

- Initially, except for the elongation of the cloud, there is no apparent structure in the set of points
- Choosing an appropriate rotation allows us to unveil the underlying structure.

What is 'loadings' and 'scores

$$y = W^T x \rightarrow \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \xrightarrow{\text{linear feature extraction}} \begin{bmatrix} y_1 \\ y_2 \\ \\ y_M \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1N} \\ w_{21} & w_{22} & \cdots & w_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ w_{M1} & w_{M2} & & w_{MN} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$$

X	X1: Feature #1	X2: Feature #2	…	Xp: Feature #p
object #1	$x_{1,1}$	$x_{1,2}$	…	$x_{1,p}$
…	…	**THE DATA SET**	…	…
object #n	$x_{n,1}$	$x_{n,2}$	…	$x_{n,p}$

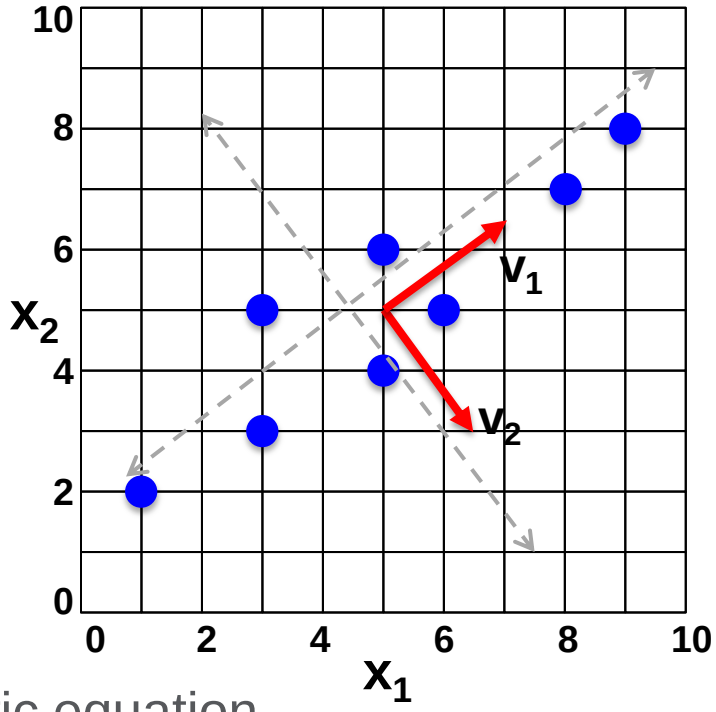
W	PC1: **w1**	PC2: **w2**	…	PCp: **wp**
	$w_{1,1}$	$w_{1,2}$	…	$w_{1,p}$
	…	**THE ROTATION**	…	…
	$w_{p,1}$	$w_{p,2}$	…	$w_{p,p}$

- Loadings →Relation between X and Principal Components (PC)
- Scores →Coordinates of X in de PC space
- Loading plot →Map of variables: how the variables relate to each other
- Scores plot →Map of the Samples: how the samples relate to each other

In order to avoid mathematics lets only say that the calculation of the principal components involve the calculation of eigenvectors and eigenvalues of the covariance matrix of X

PCA sinthetic example

- **Compute the principal components for the following 2-dimensional dataset**
  - X = (x_1,x_2) = {(1,2),(3,3),(3,5),(5,4),(5,6),(6,5),(8,7),(9,8)}
  - Look at the plot to get an idea of the solution
- **Solution (by hand)**
  - The covariance estimate of the data is:



$$\Sigma_x = \begin{bmatrix} 6{,}25 & 4{,}25 \\ 4{,}25 & 3{,}50 \end{bmatrix}$$

The eigenvalues are the zeros of the characteristic equation

$$\Sigma_x V = \lambda V \rightarrow |\Sigma_x - \lambda| = 0 \rightarrow \begin{vmatrix} 6{,}25 - \lambda & 4.25 \\ 4.25 & 3.5 - \lambda \end{vmatrix} = 0 \rightarrow \lambda_1 = 9.34;\ \ \lambda_2 = 0.41$$

The eigenvectors are the solutions of the system

$$\begin{bmatrix} 6.25 & 4{,}25 \\ 4{,}25 & 3{,}5 \end{bmatrix} \begin{bmatrix} v_{11} \\ v_{12} \end{bmatrix} = \begin{bmatrix} \lambda_1 v_{11} \\ \lambda_1 v_{12} \end{bmatrix} \rightarrow \begin{bmatrix} v_{11} \\ v_{12} \end{bmatrix} = \begin{bmatrix} 0{,}81 \\ 0{,}59 \end{bmatrix}$$

$$\begin{bmatrix} 6.25 & 4{,}25 \\ 4{,}25 & 3{,}5 \end{bmatrix} \begin{bmatrix} v_{21} \\ v_{22} \end{bmatrix} = \begin{bmatrix} \lambda_2 v_{11} \\ \lambda_2 v_{12} \end{bmatrix} \rightarrow \begin{bmatrix} v_{21} \\ v_{22} \end{bmatrix} = \begin{bmatrix} -0{,}59 \\ 0{,}81 \end{bmatrix}$$

Ricardo Gutierrez-Osuna
Texas A&M University

PCA example (adapted from de PLS toolbox: Eigenvector reserach inc)

- The following data was published in Time Magazine in January 1996
  - The data show the beer, wine and liquor consumption (liters per year), life expectancy (years) and heart disease rate (cases per 100,000 per year) for 10 countries

5 variables →

10 samples variables ↓

	Liquor	Wine	Beer	Life expectancy	Heart disease rate
	Ltr/year	Ltr/year	Ltr/year	years	Cases/10e5/year
C1	2,50	63,50	40,10	78	61,10
C2	0,90	58,00	25,10	78	94,10
C3	1,70	46,00	65,00	78	106,40
C4	1,20	15,70	102.1	78	173,00
C5	1,50	12,20	100.0	77	199,70
C6	2,00	8,90	87,80	76	176,00
C7	3,80	2,70	17,10	69	373,60
C8	1,00	1,70	140,00	73	283,70
C9	2,10	1,00	55,00	79	34,70
C10	0,80	0,20	50,40	73	36,40

Variables are in different units, so an autoscaling process is needed

C7 presents a significative difference: the highest liquor consumption, lowest beer consumption, lowest life expectancy, and highest heart disease rate

Are there more trends?

PCA example

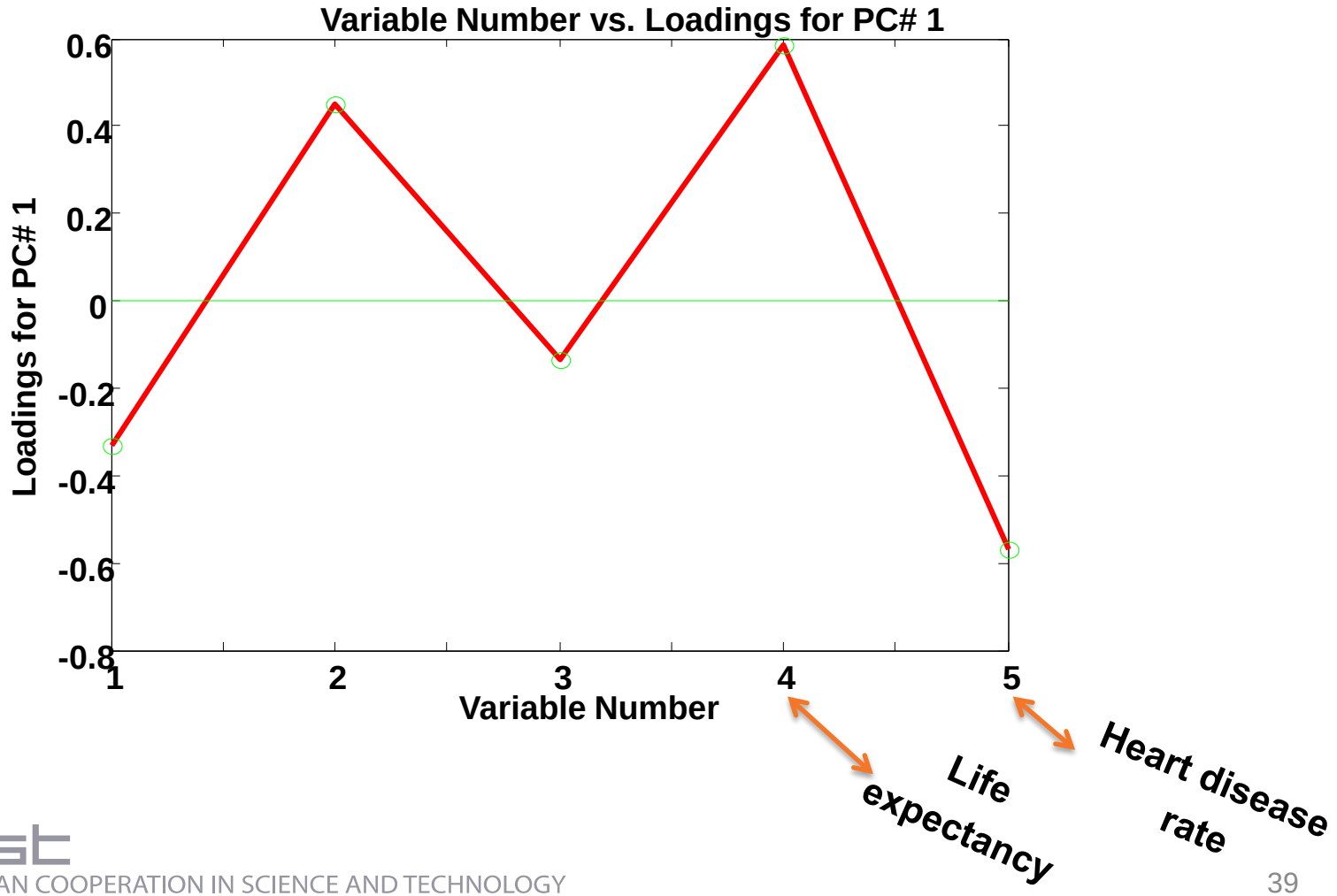
- With normalized data, 2 principal components capture more or less 75% of variance

Percent Variance Captured by PCA Model

Principal Component Number	Eigenvalue of Cov(X)	% Variance Captured This PC	% Variance Captured Total
1	2.33e+000	46.52	46.52
2	1.40e+000	28.05	74.57
3	5.87e-001	11.73	86.30
4	4.16e-001	8.32	94.62
5	2.69e-001	5.38	100.00

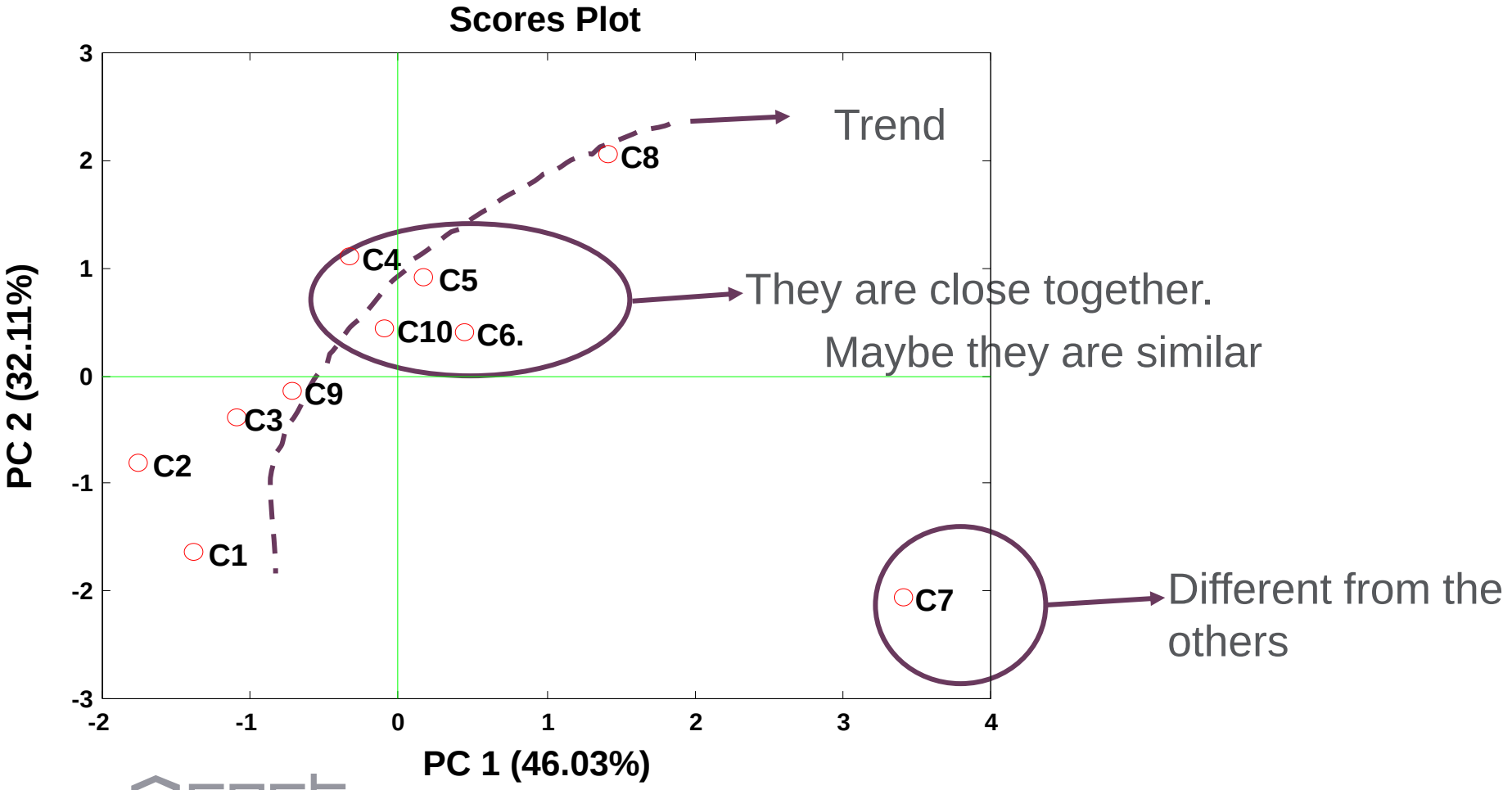
PCA example → Loadings

- Relation between X and Principal Components (PC)



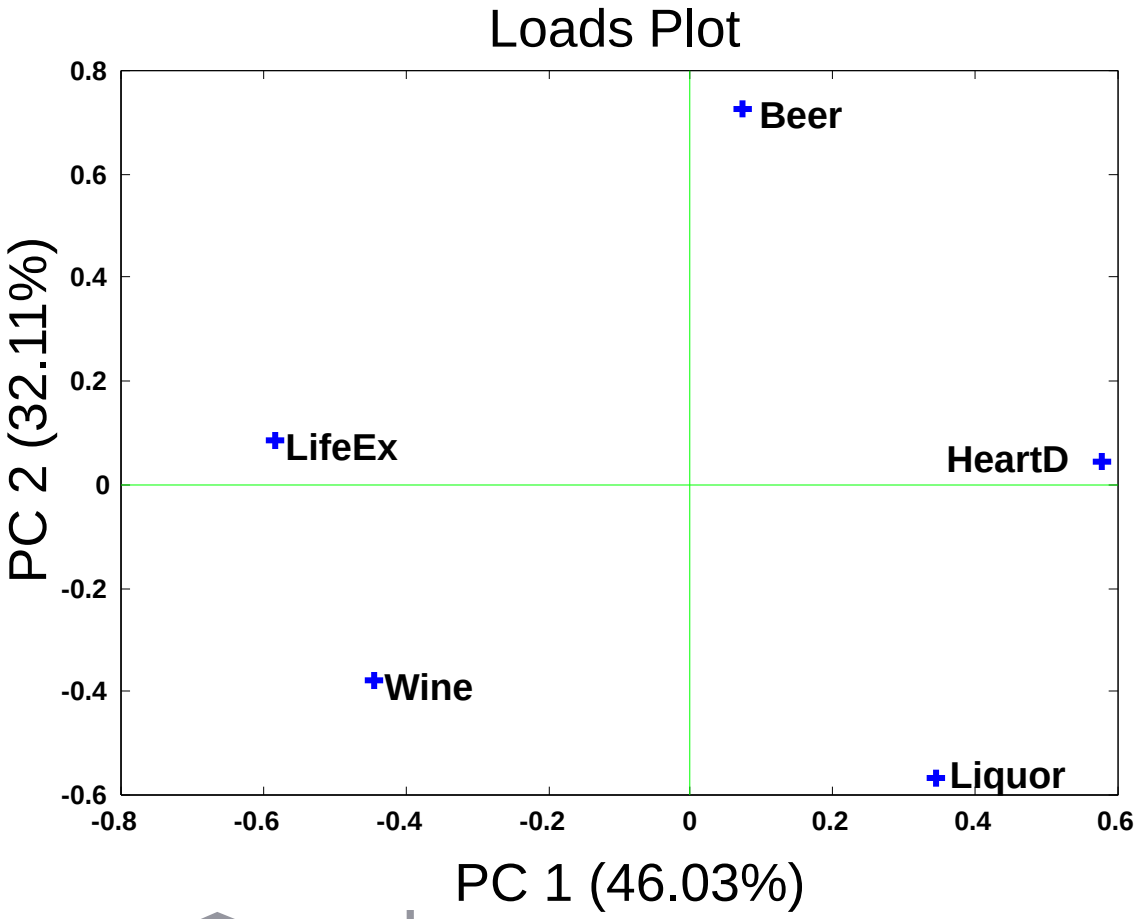
PCA example → Scores plot

- We can get a quick idea of the relationships between samples looking to scores plots



PCA example → Loadings plot

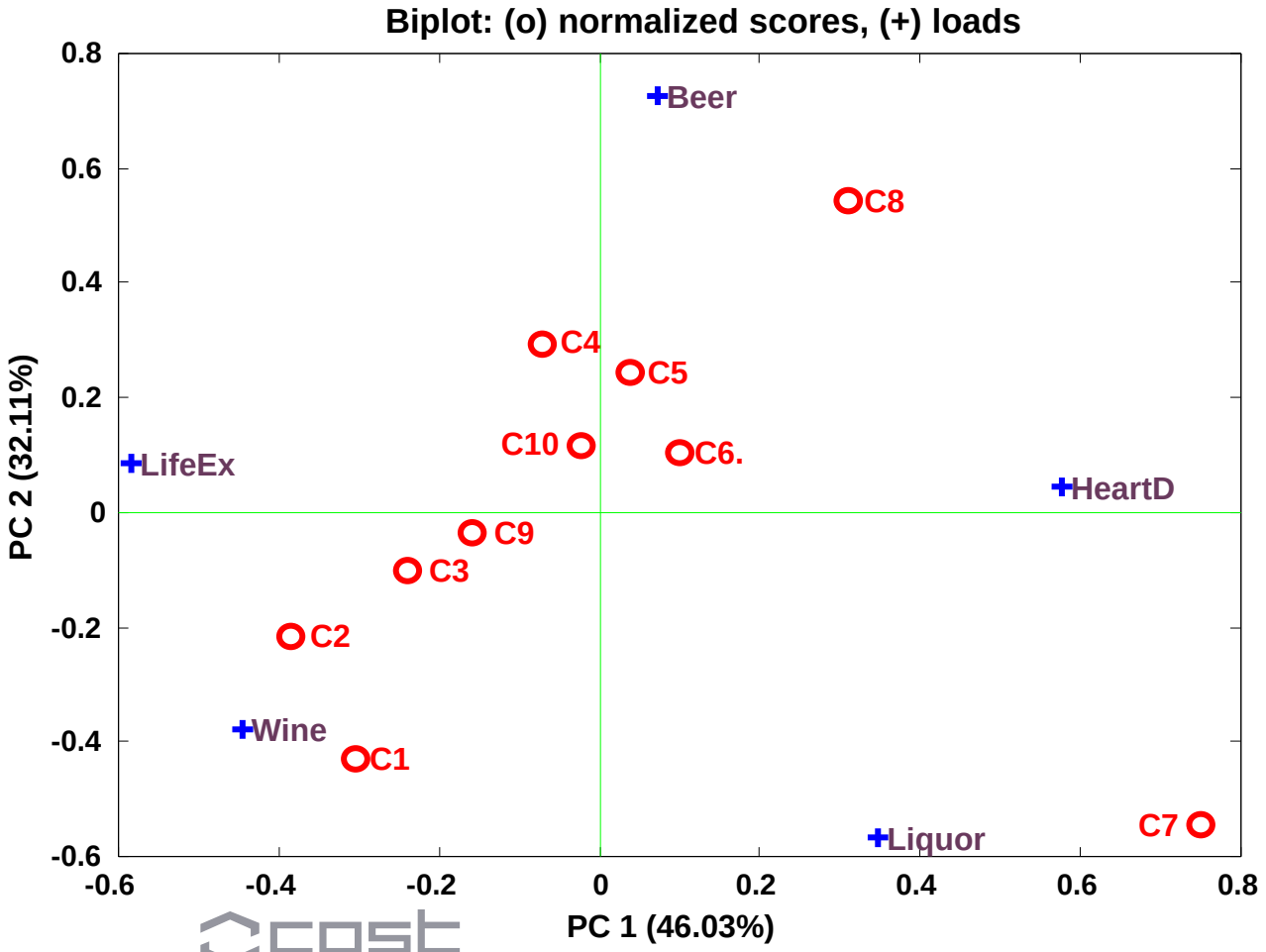
- Relationship between variables: loading plots



- None variables are very similar
- LifeEx and HeartD are the most significant variables, and one is opposite to the other.
  - So they are anticorrelated
    - One can expect than LiveEx is anticorreleted with HeartD

PCA example → biplot

- Which variables are responsible of differences between countries?
  - Consider scores and loadings in the same graph: biplot (some normalization must be done)

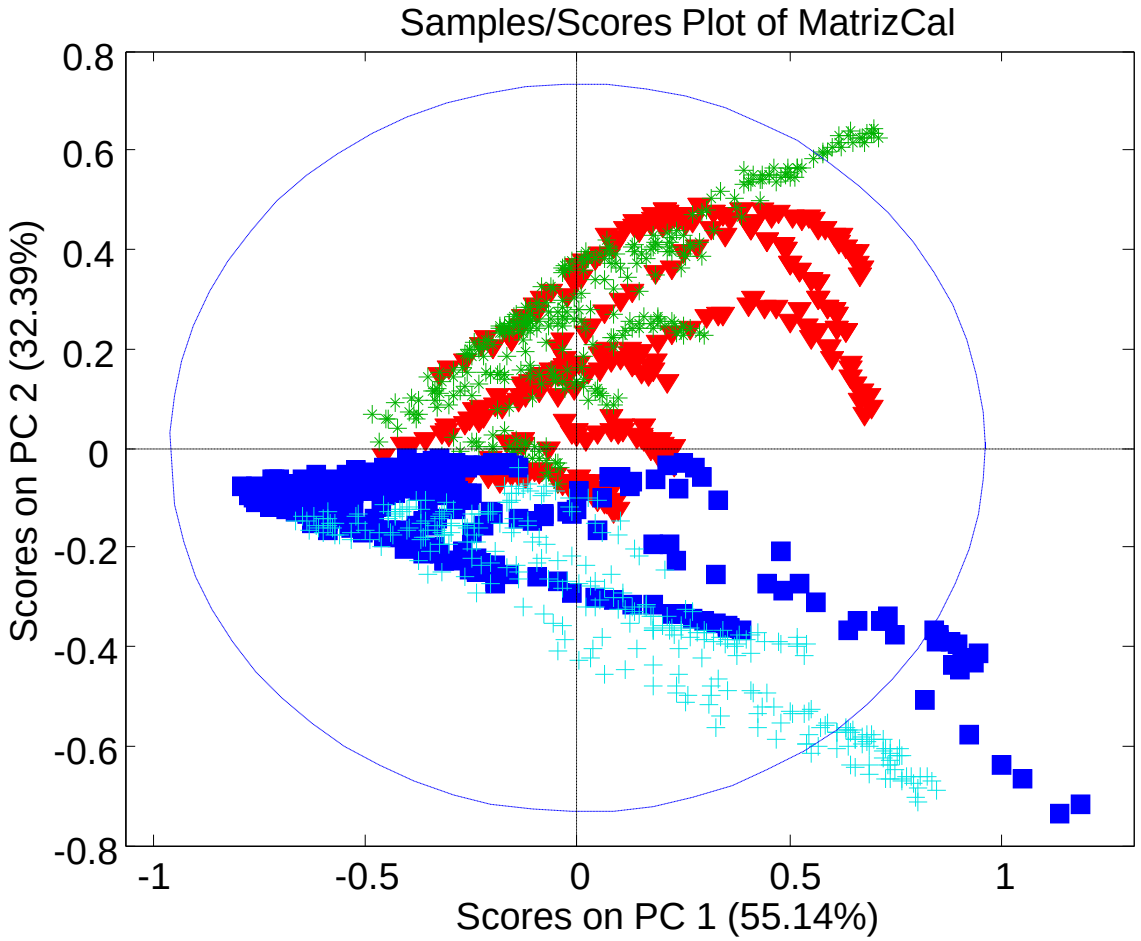


- C7 tends to have:
  - high liquor consumption,
  - low beer consumption,
  - high heart disease
  - and low life expectance

- That points can be confirmed looking to the initial table

Typical pattern recognition results with PCA

- PCA results for the exploratory analysis from the four class of wine samples



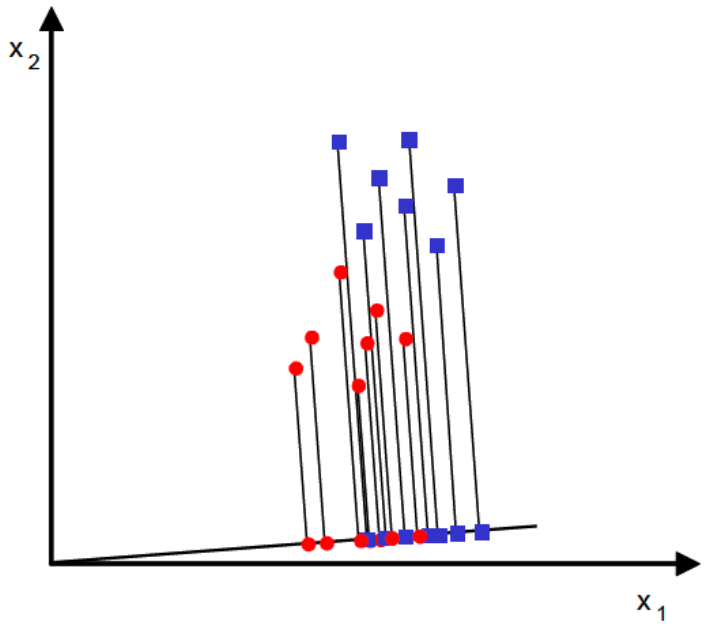
Montilla-Moriles wines
Jerez wines
Huelva wines
Valdepeñas wines

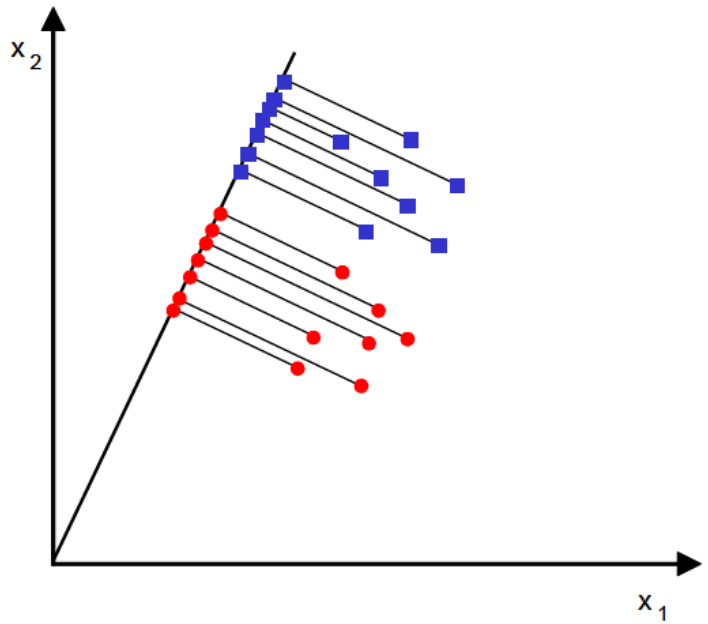
- **Problem**: It can be seen that wines are separeted by its alcohol contens

R. Garrido-Delgado et al, Direct coupling of a gas–liquid separator to an ion mobility spectrometer for the classification of different white wines using chemometrics tools, Talanta 2011

Linear Discriminant Analysis

- The objective of LDA is to perform dimensionality reduction while preserving as much of the class discriminatory information as possible
  - Assume a set of D-dimensional samples $\{x_1, x_2, \ldots, x_N\}$, N_1 of which belong to class ω_1, and N_2 to class ω_2
  - We seek to obtain a scalar y by projecting the samples x onto a line $y=w^Tx$
  - Of all possible lines we want the one that maximizes the separability of the scalars



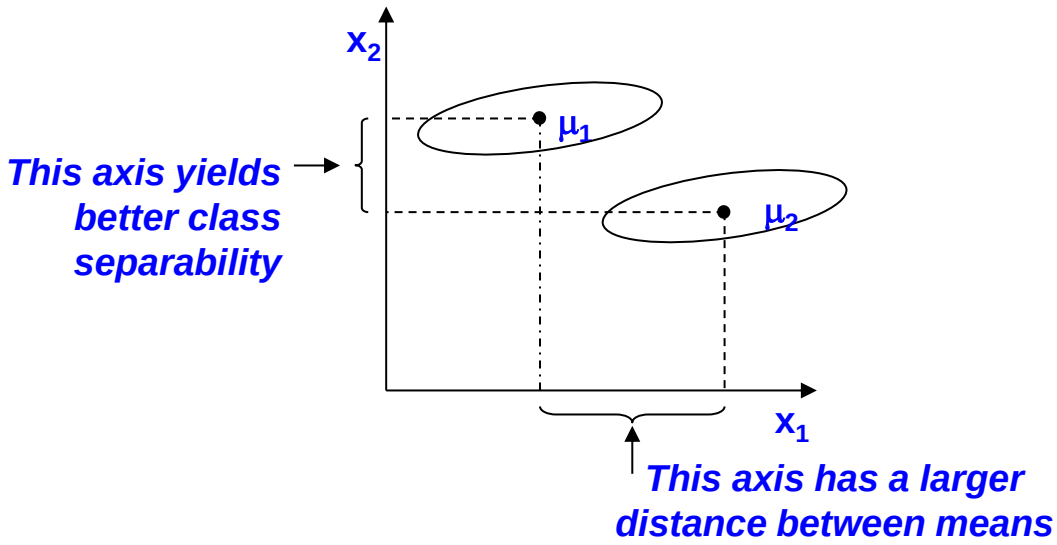


Linear Discriminant Analysis

- In order to find a good projection vector, we need to define a measure of separation between the projections
  - We could choose the distance between the projected means

$$J(w) = \left| \tilde{\mu}_1 - \tilde{\mu}_2 \right| = \left| w^T (\mu_1 - \mu_2) \right|$$

  - However, the distance between projected means is not a very good measure since it does not take into account the standard deviation within the classes



Ricardo Gutierrez-Osuna
Texas A&M University

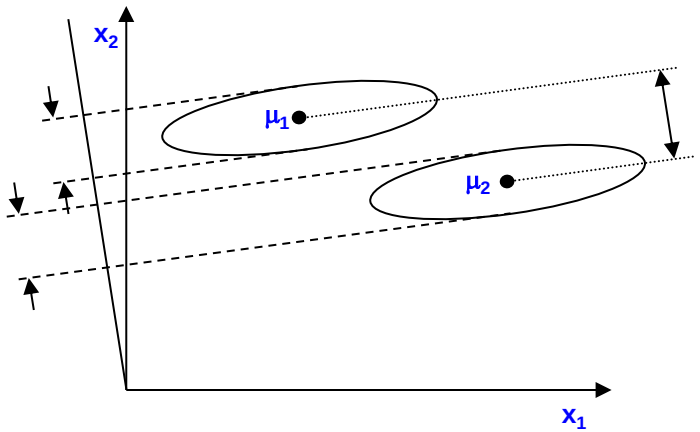
Linear Discriminant Analysis

- The solution proposed by Fisher is to normalize the difference between the means by a measure of the within-class variance
- For each class we define the scatter, an equivalent of the variance, as $\tilde{s}_i^2 = \sum_{y \in \omega_i} (y - \tilde{\mu}_i)^2$
- And the quantity $(\tilde{s}_1^2 + \tilde{s}_2^2)$ is called the within-class scatter of the projected examples

- The Fisher linear discriminant is defined as the linear function wTx that maximizes the criterion function

$$J(w) = \frac{|\tilde{\mu}_1 - \tilde{\mu}_2|^2}{\tilde{s}_1^2 + \tilde{s}_2^2}$$

- In a nutshell: we look for a projection where examples from the same class are projected very close to each other and, at the same time, the projected means are as farther apart as possible



LDA: Some algebra (I'm sorry)

- To find the optimum projection for a 2 classes problem, we must express $J(w)$ as a function of w
  - First, we define a measure of the scatter in feature space x

$$S_i = \sum_{x \in \omega_i} (x - \mu_i)(x - \mu_i)^T$$
$$S_1 + S_2 = S_W$$

    - where S_W is called the within-class scatter matrix
- The scatter of the projection y can then be expressed as a function of the scatter matrix in feature space x

$$\tilde{s}_i^2 = \sum_{y \in \omega_i} (y - \tilde{\mu}_i)^2 = \sum_{x \in \omega_i} (w^T x - w^T \mu_i)^2 =$$
$$= \sum_{x \in \omega_i} w^T (x - \mu_i)(x - \mu_i)^T w = w^T S_i w$$

$$\tilde{s}_1^2 + \tilde{s}_2^2 = w^T S_W w$$

Additional Algebra: I apologize again

- Similarly, the difference between the projected means can be expressed in terms of the means in the original feature space

$$(\tilde{\mu}_1 - \tilde{\mu}_2)^2 = (w^T\mu_1 - w^T\mu_2)^2 = w^T \underbrace{(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T}_{S_B} w = w^T S_B w$$

  - The matrix S_B is called the between-class scatter.
- We can express J(w) in terms of x and w as → $J(w) = \dfrac{w^T S_B w}{w^T S_W w}$
- To find the maximum of J(w) → derive and equate to zero →finally, the projection vector w* which maximizes J(w) is

$$w^* = \underset{w}{\operatorname{argmax}}\left\{\frac{w^T S_B w}{w^T S_W w}\right\} = S_W^{-1}(\mu_1 - \mu_2)$$

- Using a similar development, LDA can be generalized to problems with more than 2 classes.

$$W^* = \left[w_1^* \mid w_2^* \mid \cdots \mid w_{C-1}^*\right] = \operatorname{argmax}\left\{\frac{W^T S_B W}{W^T S_W W}\right\} \Rightarrow (S_B - \lambda_i S_W) w_i^* = 0$$

Example

- LDA projection for the following 2D dataset

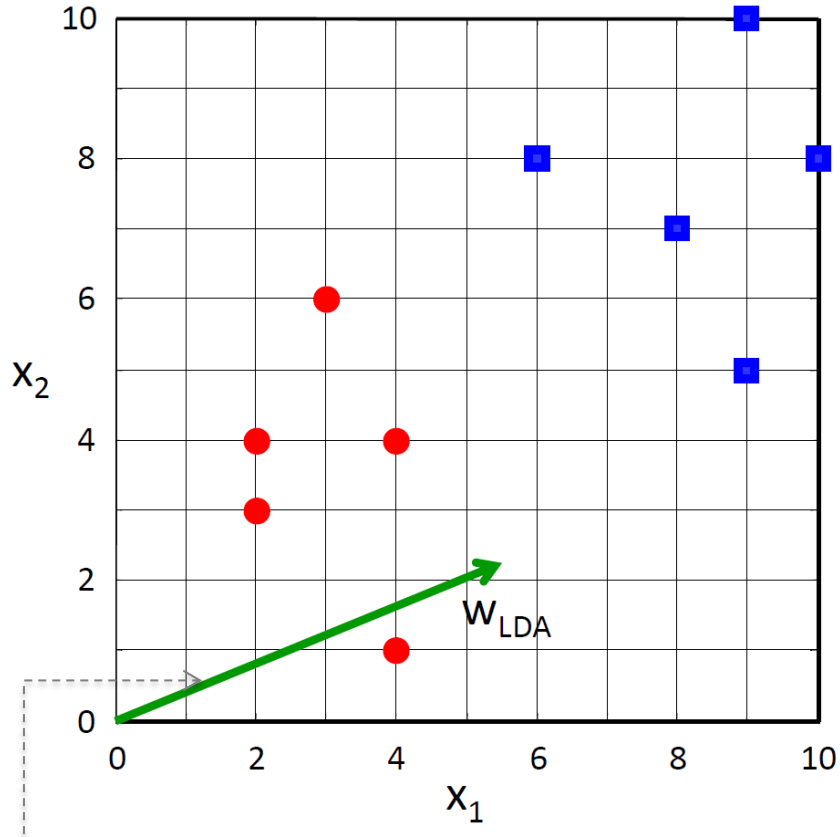
$$X1 = \{(4,1), (2,4), (2,3), (3,6), (4,4)\}$$
$$X2 = \{(9,10), (6,8), (9,5), (8,7), (10,8)\}$$

- Solution (by hand)
  - The class statistics are

$$S_1 = \begin{bmatrix} .8 & -.4 \\ -.4 & 2.64 \end{bmatrix} S_2 = \begin{bmatrix} 1.84 & -.4 \\ -.4 & 2.64 \end{bmatrix}$$

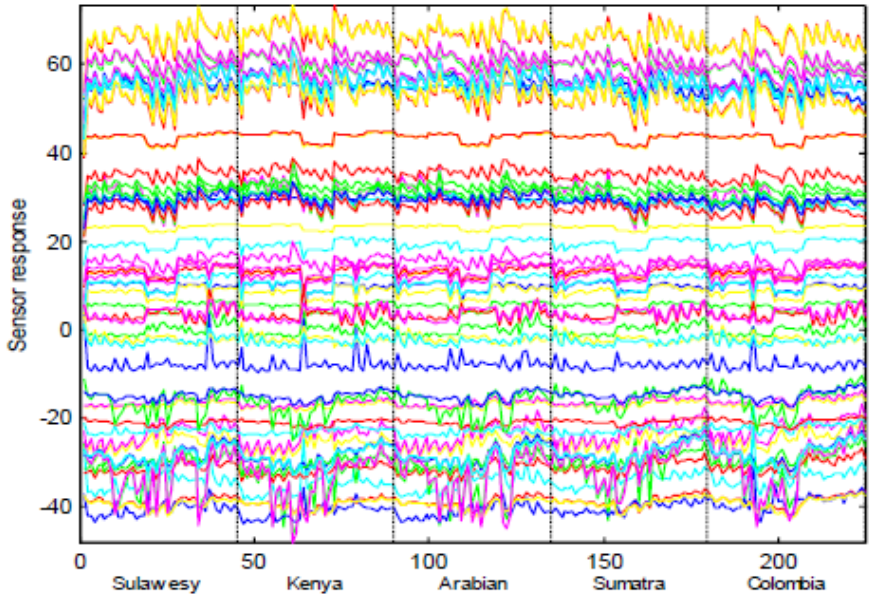
$$\mu_1 = [3.0\ 3.6]^T;\ \ \mu_2 = [8.4\ 7.6]^T$$

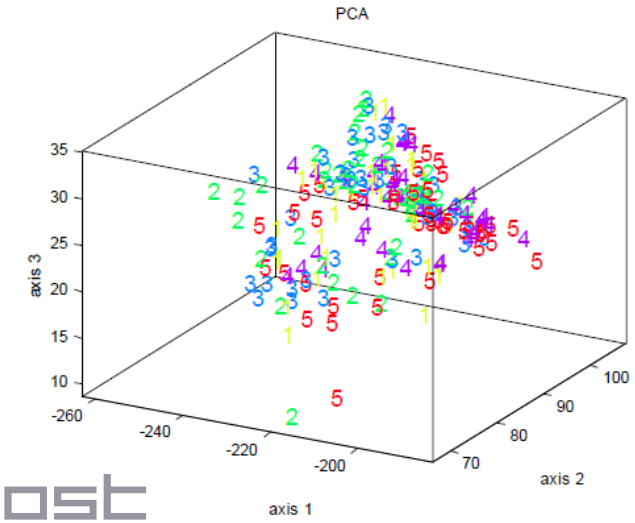
$$w^* = S_W^{-1}(\mu_1 - \mu_2) = [-.91\ -.39]^T$$

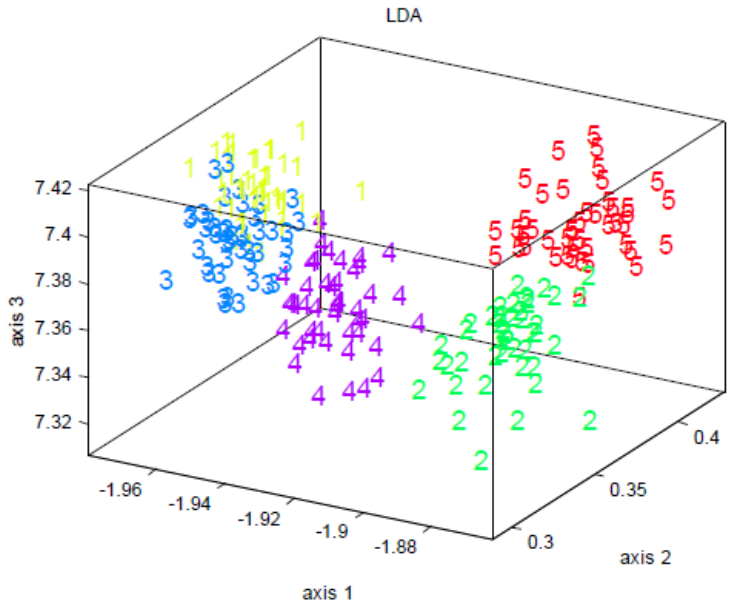


LDA vs PCA

- **This example illustrates the performance of PCA and LDA on an odor recognition problem**
  - Five types of coffee beans were presented to an array of gas sensors
  - For each coffee type, 45 “sniffs” were performed and the response of the gas sensor array was processed in order to obtain a 60-dimensional feature vector
- **Results**
  - From the 3D scatter plots it is clear that LDA outperforms PCA in terms of class discrimination
  - This is one example where the discriminatory information is not aligned with the direction of maximum variance

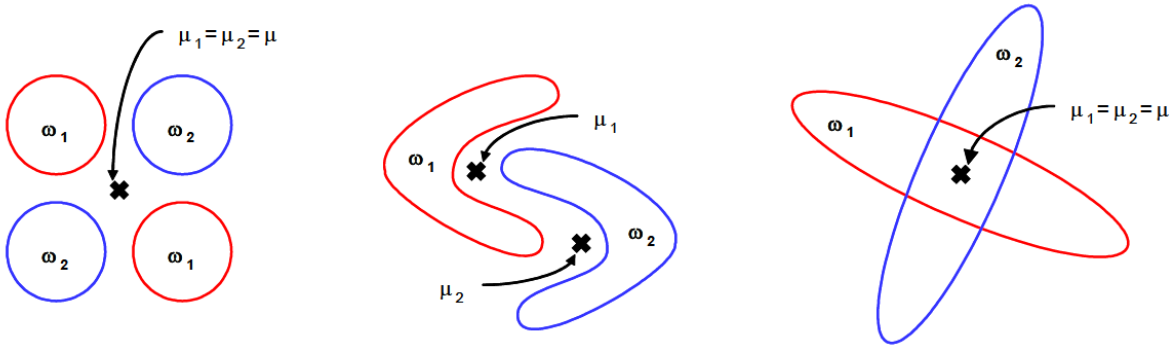




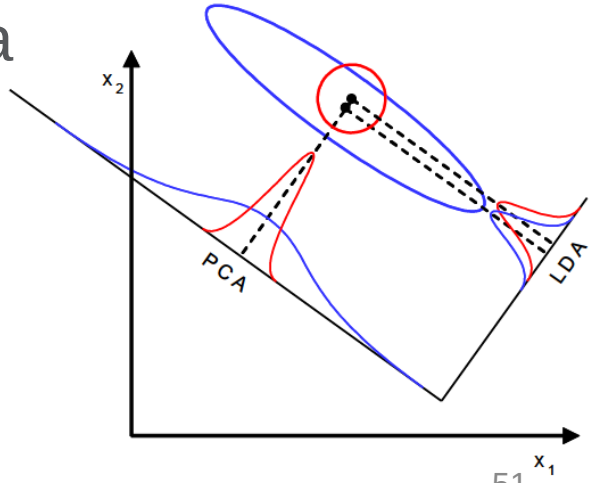


Limitations of LDA

- LDA assumes unimodal Gaussian likelihoods
  - If the densities are significantly non-Gaussian, LDA may not preserve any complex structure of the data needed for classification



- LDA will fail when the discriminatory information is not in the mean but rather in the variance of the data

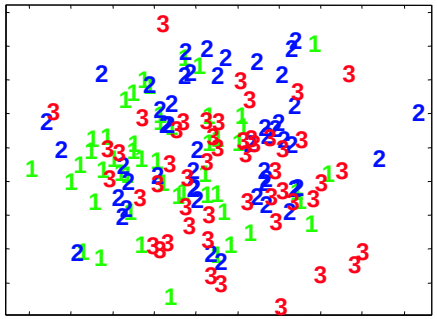


Ricardo Gutierrez-Osuna
Texas A&M University

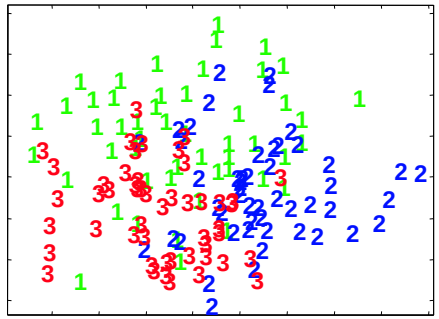
Limitations of LDA

- LDA has a tendency to overfit training data
  - To illustrate this problem, we generate an artificial dataset
    - Three classes, 50 examples per class, with the exact same likelihood: a multivariate Gaussian with zero mean and identity covariance
    - As we arbitrarily increase the number of dimensions, classes appear to separate better, even though they come from the same distribution

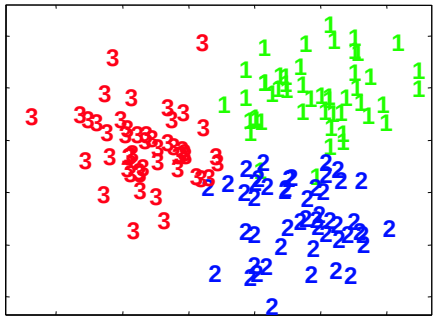
10 dimensions



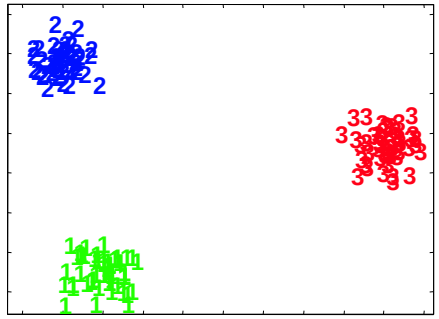
50 dimensions



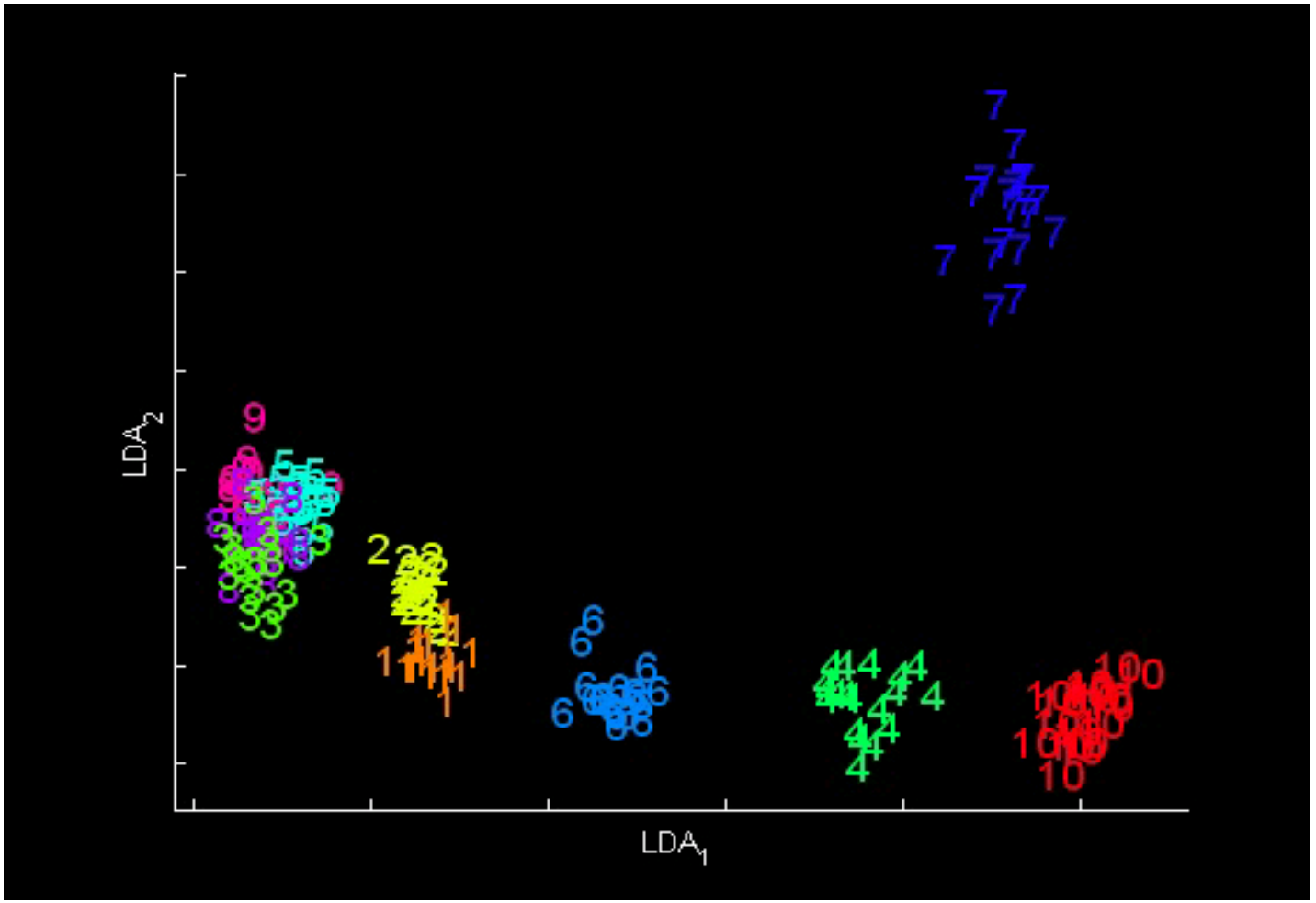
100 dimensions



150 dimensions



Typical pattern recognition results with LDA

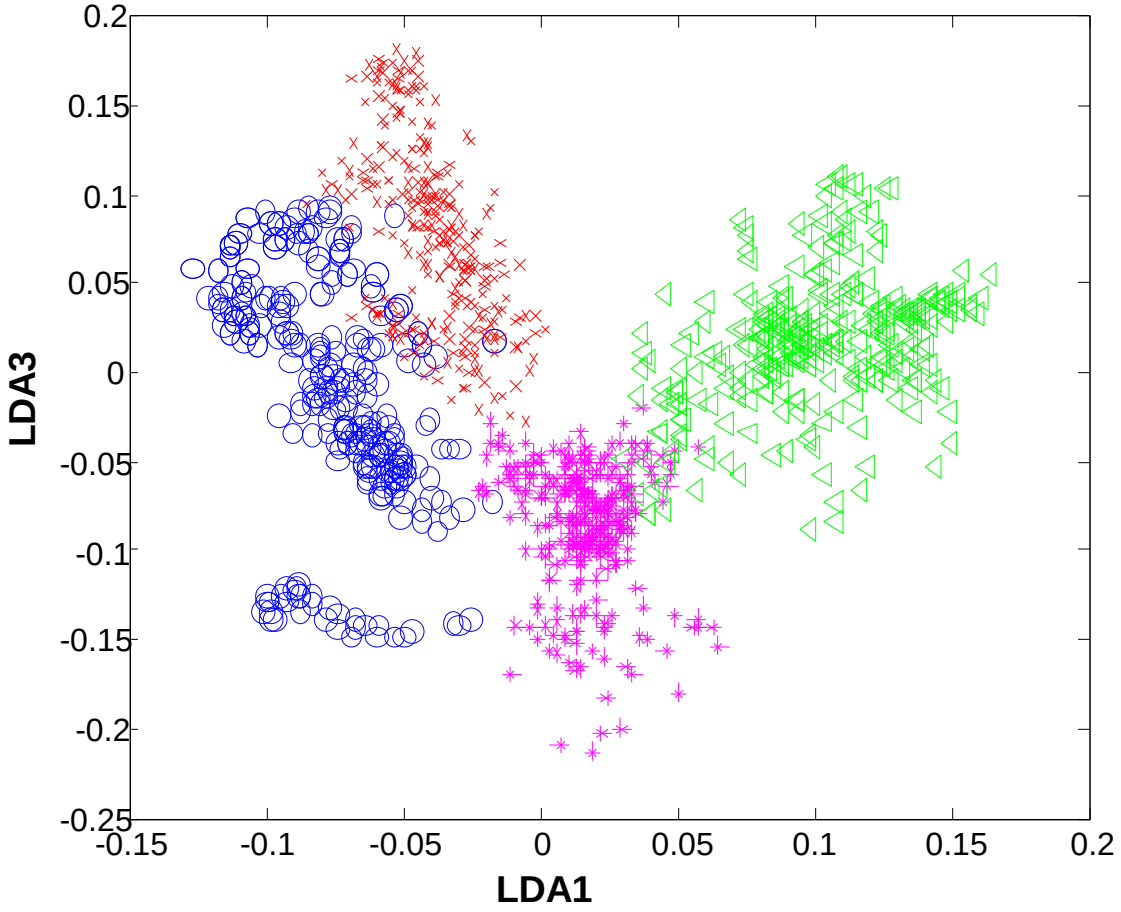


Odor labels

1	*Coke*
2	*Diet-Coke*
3	*Pepsi*
4	*Dr.-Pepper*
5	*Cherry-Coke*
6	*Cherry-Pepsi*
7	*Cheerwine*
8	*RC-Cola*
9	*Eckerd-Cola*
10	*Eckerd-Dr.-Riffic*

Gutierrez-Osuna, A Method for Evaluating Data-Preprocessing Techniques for Odor Classification with an Array of Gas Sensors IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS 1999

Typical pattern recognition results with PCA-LDA



Montilla-Moriles wines
Jerez wines
Huelva wines
Valdepeñas wines

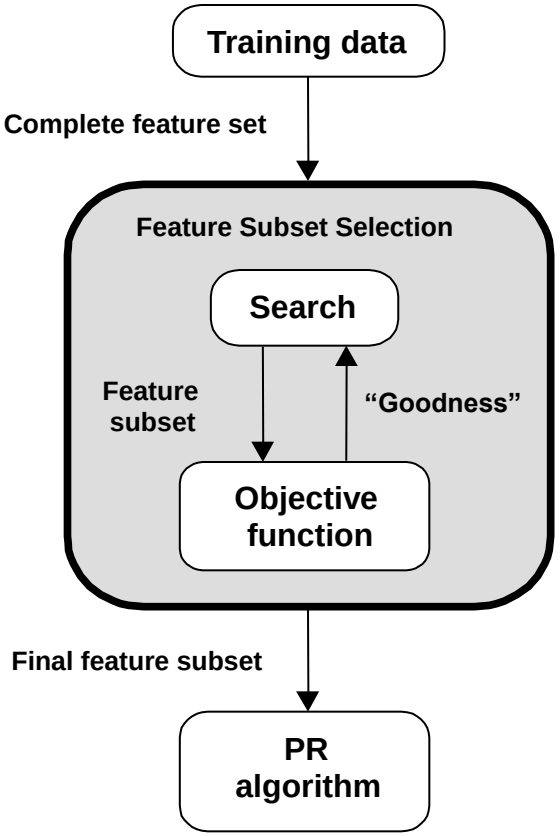
R. Garrido-Delgado et al, Direct coupling of a gas–liquid separator to an ion mobility spectrometer for the classification of different white wines using chemometrics tools, Talanta 2011

Feature Subset Selection

- Why Feature Subset Selection?
  - Feature Subset Selection is necessary in a number of situations
    - Features may be expensive to obtain
      - You evaluate a large number of features (sensors) in the test bed and select only a few for the final implementation
    - You may want to extract meaningful rules from your classifier
      - When you transform or project, the measurement units (length, weight, etc.) of your features are lost
    - Features may not be numeric
      - A typical situation in the machine learning domain
  - In addition, fewer features means fewer parameters for pattern recognition
    - Improved the generalization capabilities
    - Reduced complexity and run-time
  - Although FSS can be thought of as a special case of feature extraction (think of a sparse projection matrix with a few ones), in practice it is a quite different problem
    - FSS looks at dimensionality reduction from a different perspective
    - FSS has a unique set of methodologies

Search strategy and objective function

- Feature Subset Selection requires
  - A search strategy to select candidate subsets
  - An objective function to evaluate candidates
- Search Strategy
  - Exhaustive evaluation involves $\binom{N}{M}$ feature subsets for a fixed value of M, and 2^N subsets if M must be optimized as well
    - This number of combinations is unfeasible, even for moderate values of M and N
      - For example, exhaustive evaluation of 10 out of 20 features involves 184,756 feature subsets; exhaustive evaluation of 10 out of 20 involves more than 10^{13} feature subsets
  - A search strategy is needed to explore the space of all possible feature combinations
- Objective Function
  - The objective function evaluates candidate subsets and returns a measure of their “goodness”, a feedback that is used by the search strategy to select new candidates

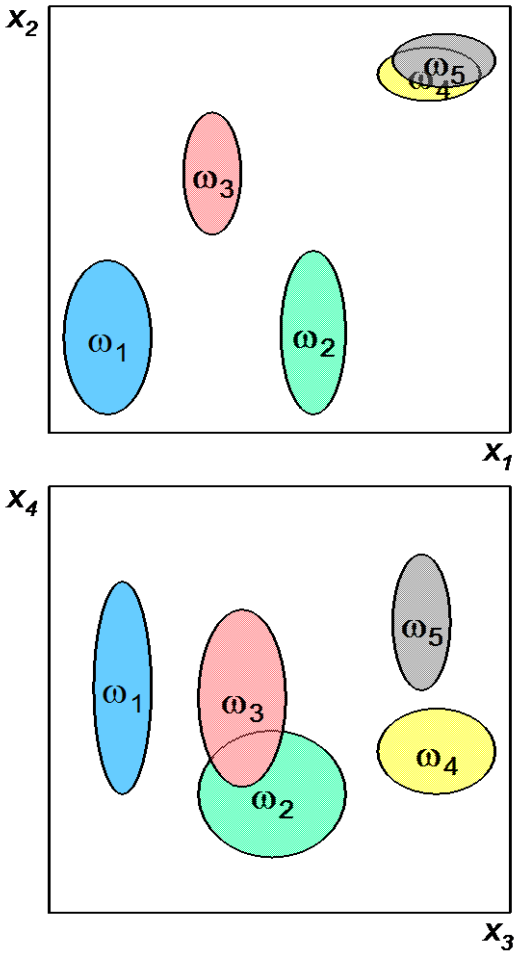


Search strategies

- FSS search strategies can be grouped in three categories
  - Sequential
    - These algorithms add or remove features sequentially, but have a tendency to become trapped in local minima
      - Sequential Forward Selection
      - Sequential Backward Selection
      - Sequential Floating Selection
  - Exponential
    - These algorithms evaluate a number of subsets that grows exponentially with the dimensionality of the search space
      - Exhaustive Search (already discussed)
      - Branch and Bound
      - Beam Search
  - Randomized
    - These algorithms incorporating randomness into their search procedure to escape local minima
      - Simulated Annealing
      - Genetic Algorithms

Naïve sequential feature selection

- One may be tempted to evaluate each individual feature separately and select those M features with the highest scores
  - Unfortunately, this strategy will very rarely work since it does not account for feature dependence
- An example will help illustrate the poor performance of this naïve approach
  - The scatter plots show a 4-dimensional pattern recognition problem with 5 classes
    - The objective is to select the best subset of 2 features using the naïve sequential FSS procedure
  - A reasonable objective function will generate the following feature ranking: $J(x_1)>J(x_2)\approx J(x_3)>J(x_4)$
    - If we were to choose features according to the individual scores $J(x_k)$, we would choose x_1 and either x_2 or x_3, leaving classes ω_4 and ω_5 non separable
    - The optimal feature subset turns out to be $\{x_1, x_4\}$, because x_4 provides the only information that x_1 needs: discrimination between classes ω_4 and ω_5

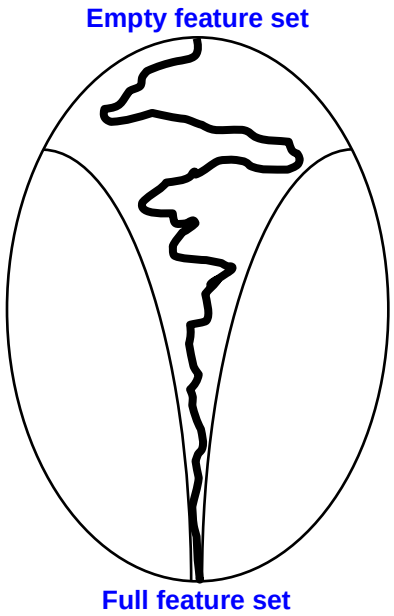


Sequential Forward Selection (SFS)

- Sequential Forward Selection is a simple greedy search

1. Start with the empty set Y={∅}
2. Select the next best feature $x^+ = \underset{x \in X - Y_k}{\operatorname{argmax}}\left[J(Y_k + x)\right]$
3. Update $Y_{k+1}=Y_k+x$; k=k+1
4. Go to 2

- Notes
  - SFS performs best when the optimal subset has a small number of features
    - When the search is near the empty set, a large number of states can be potentially evaluated
    - Towards the full set, the region examined by SFS is narrower since most of the features have already been selected
    - The main disadvantage of SFS is that it is unable to remove features that become obsolete with the addition of new features



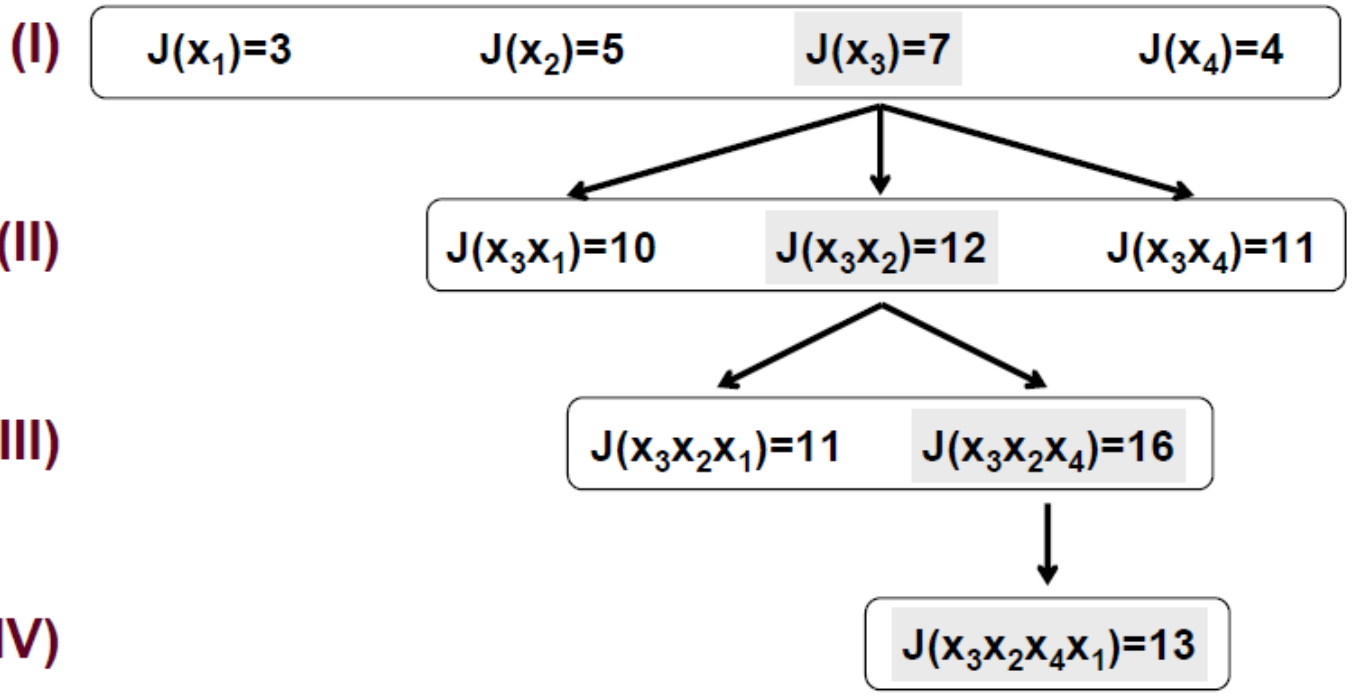
SFS example

- Assuming the objective function J(X) below, perform a Sequential Forward Selection to completion

$$J(X) = -2x_1x_2 + 3x_1 + 5x_2 - 2x_1x_2x_3 + 7x_3 + 4x_4 - 2x_1x_2x_3x_4$$

  - where x_k are indicator variables that determine if the k-th feature has been selected (x_k=1) or not (x_k=0)

- Solution

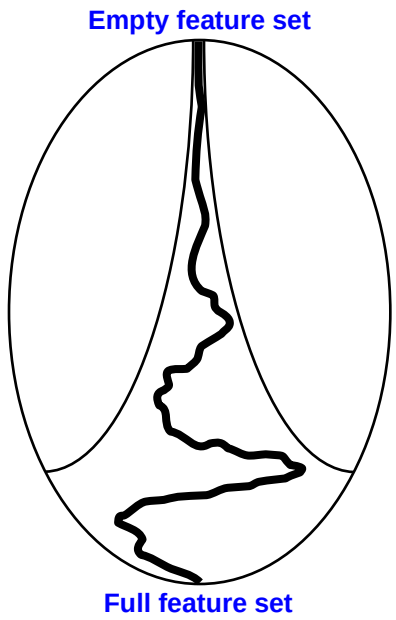


Sequential Backward Selection (SBS)

- Sequential Backward Selection works in the opposite manner as SFS

1. Start with the full set Y=X
2. Remove the worst feature $x^- = \underset{x \in Y_k}{\operatorname{argmax}}[J(Y_k - x)]$
3. Update $Y_{k+1}=Y_k-x$; k=k+1
4. Go to 2

- Notes
  - SBS works best when the optimal feature subset has a large number of features, since SBS spends most of its time visiting large subsets
  - The main limitation of SBS is its inability to reevaluate the usefulness of a feature after it has been discarded



Seq. Floating Selection (SFFS and SFBS)

- There are two floating methods
  - Sequential Floating Forward Selection (SFFS) starts from the empty set
    - After each forward step, SFFS performs backward steps as long as the objective function increases
  - Sequential Floating Backward Selection (SFBS) starts from the full set
    - After each backward step, SFBS performs forward steps as long as the objective function increases
- SFFS Algorithm (SFBS is analogous)

1. Start with the empty set $Y=\{\varnothing\}$
2. Select the best feature

$$x^+ = \underset{x \notin Y_k}{\operatorname{argmax}}\left[J(Y_k + x)\right]$$

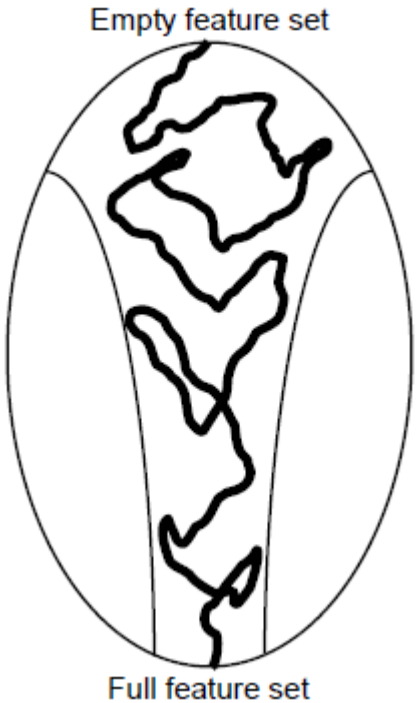
$$Y_k = Y_k + x^+; \quad k = k+1$$

3. Select the worst feature*

$$x^- = \underset{x \in Y_k}{\operatorname{argmax}}\left[J(Y_k - x)\right]$$

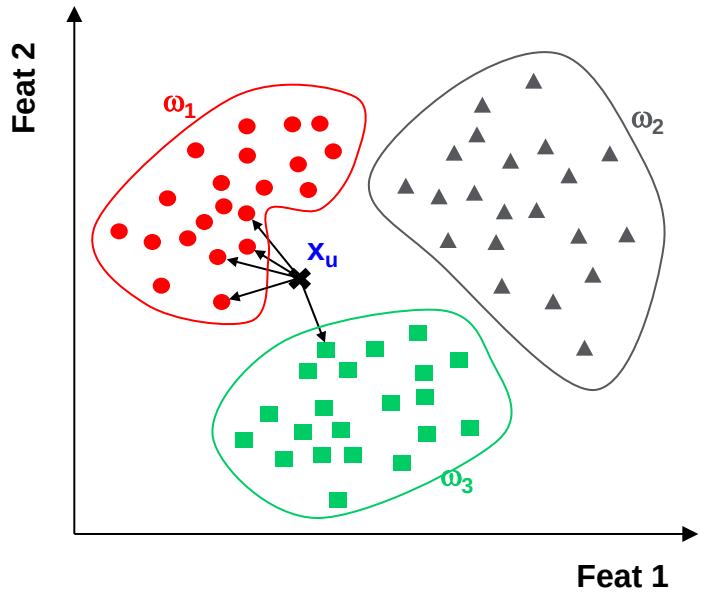
4. If $J(Y_k - x^-) > J(Y_k)$ then
   $Y_{k+1} = Y_k - x$; $k = k+1$
   go to Step 3
   else
   go to Step 2

Notice that you'll need to do some book-keeping to avoid infinite loops



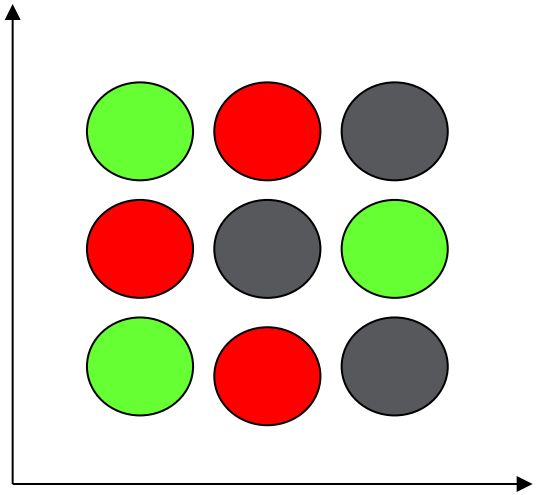
K Nearest Neighbor classifier

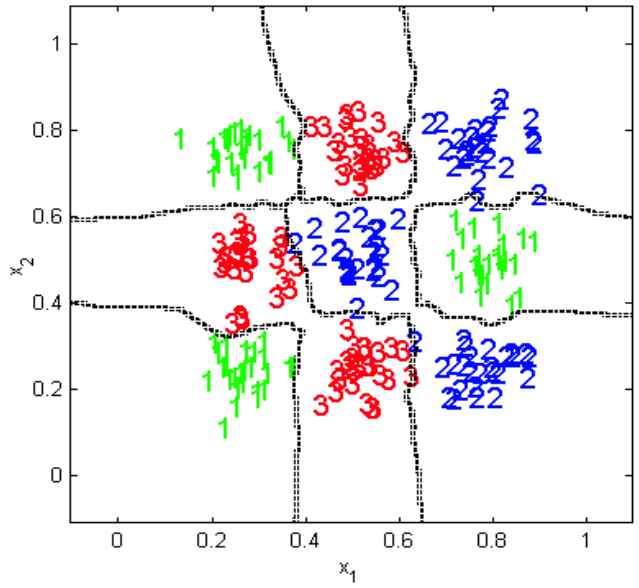
- The kNN classifier is a very intuitive method
  - Examples are classified based on their similarity with training data
    - For a given unlabeled example $x_u \in \Re^D$, find the k "closest" labeled examples in the training data set and assign x_u to the class that appears most frequently within the k-subset
- The kNN only requires
  - An integer k
  - A set of labeled examples
  - A metric to measure "closeness"

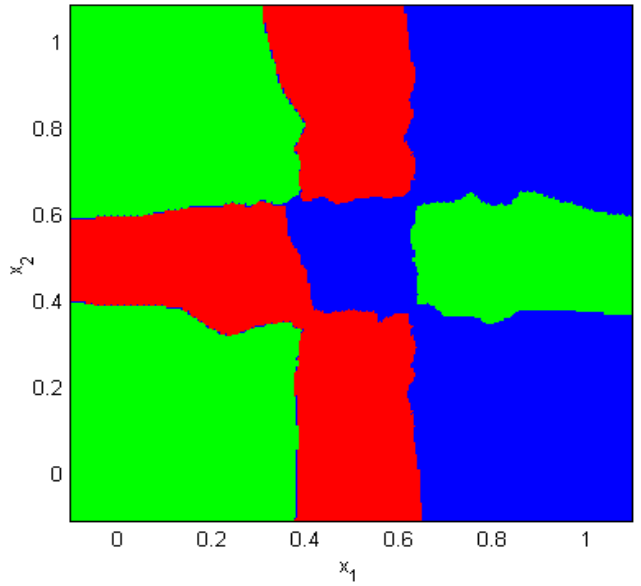


kNN in action: example 1

- We generate data for a 2-dimensional 3-class problem, where the class-conditional densities are multi-modal, and non-linearly separable
- We used kNN with
  - k = five
  - Metric = Euclidean distance

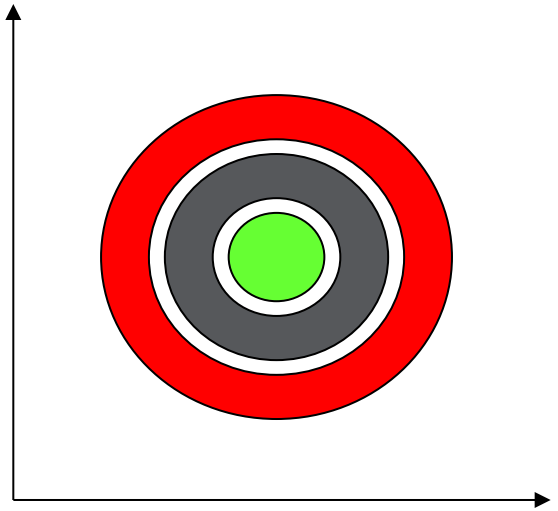


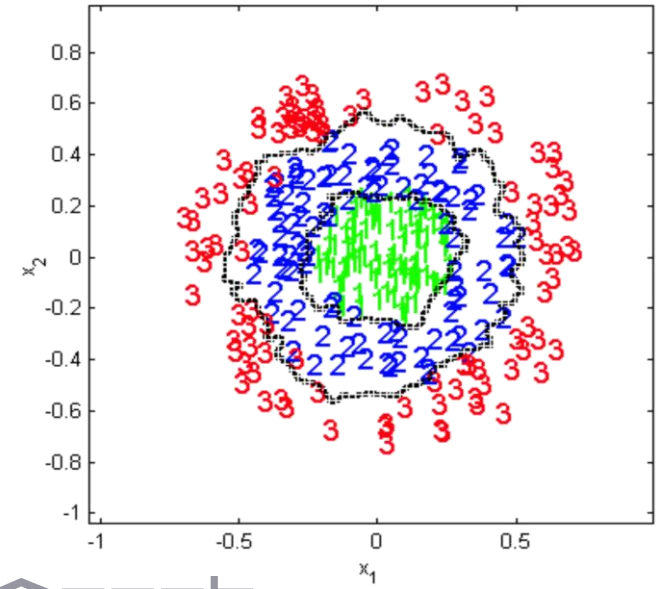


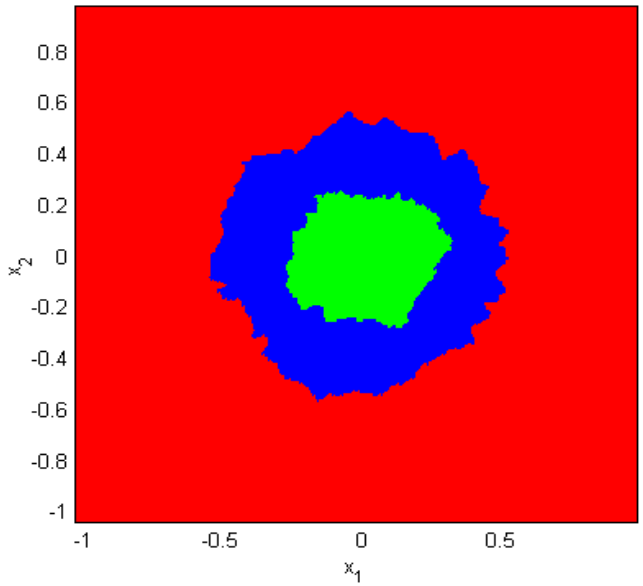


kNN in action: example 2

- We generate data for a 2-dim 3-class problem, where the likelihoods are unimodal, and are distributed in rings around a common mean
  - These classes are also non-linearly separable
- We used kNN with
  - k = five
  - Metric = Euclidean distance

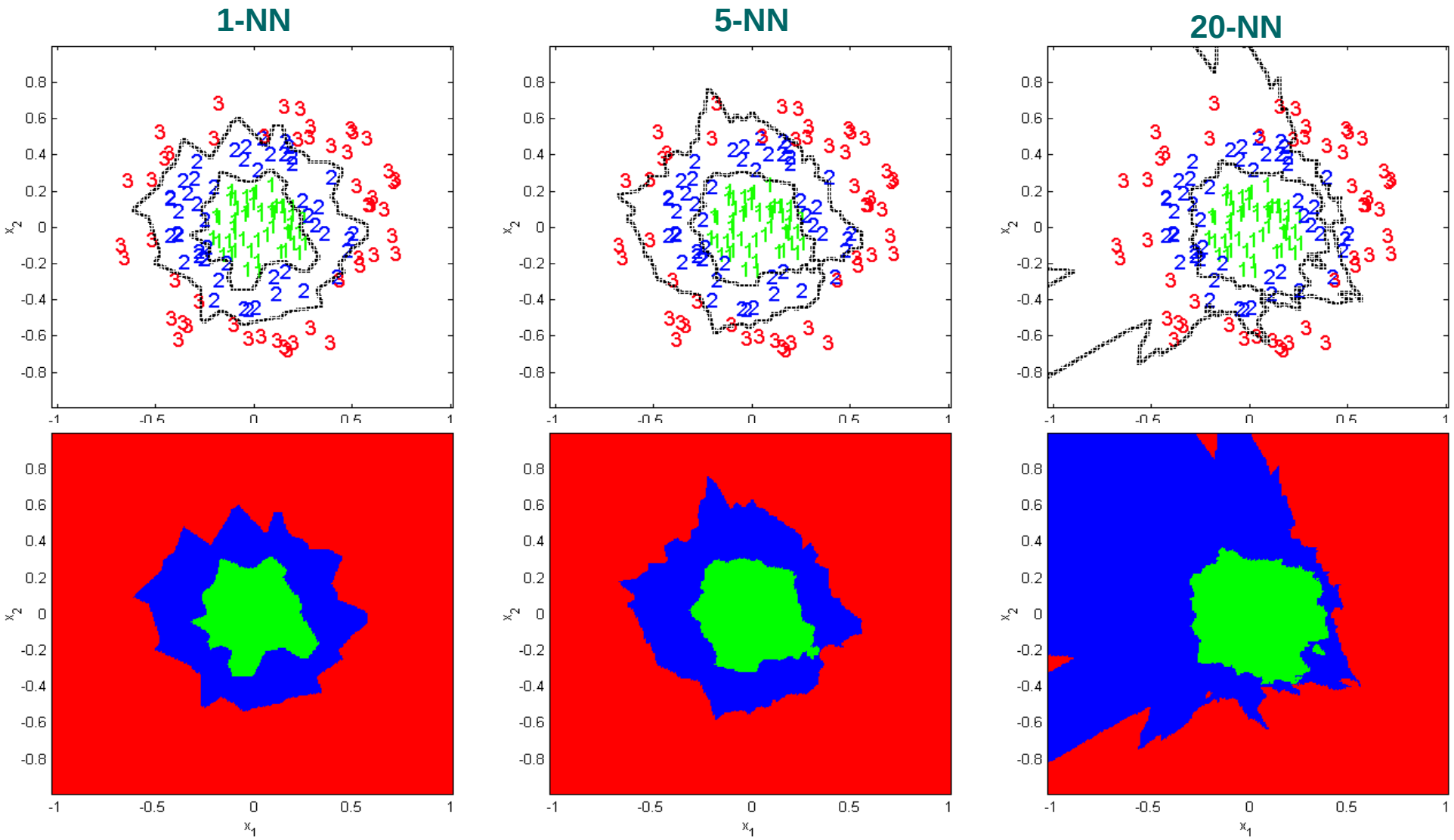






Ricardo Gutierrez-Osuna
Texas A&M University

kNN versus 1NN

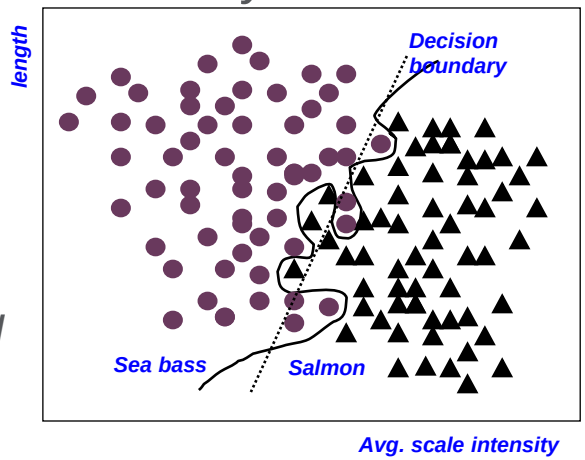


Characteristics of the kNN classifier

- Advantages
  - Analytically tractable, simple implementation
  - Uses local information, which can yield highly adaptive behavior
  - Lends itself very easily to parallel implementations
- Disadvantages
  - Large storage requirements
  - Computationally intensive recall
  - Highly susceptible to the curse of dimensionality
- 1NN versus kNN
  - The use of large values of k has two main advantages
    - Yields smoother decision regions
    - Provides probabilistic information: The ratio of examples for each class gives information about the ambiguity of the decision
  - However, too large values of k are detrimental
    - It destroys the locality of the estimation
    - In addition, it increases the computational burden

The fish factory example again

- The issue of generalization
  - The recognition rate of our linear classifier (95.7%) met the design specs, but we still think we can improve the performance of the system
    - *We then design ultra-sophysticated nonlinear pattern recognition system based on artificial neural networks and, using all the database, an impressive classification rate of 99.9975% is obtained, with the following decision boundary*



  - Satisfied with our classifier, we integrate the system and deploy it to the fish processing plant
    - A few days later the plant manager calls to complain that the system is misclassifying an average of 25% of the fish
    - **What went wrong?**

Validation

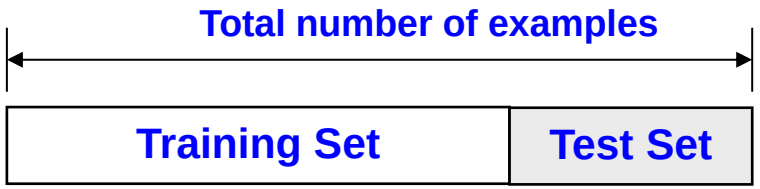
- Validation techniques are motivated by two fundamental problems in pattern recognition: model selection and performance estimation
- Model selection
  - Almost invariably, all pattern recognition techniques have one or more free parameters
    - The number of neighbors in a kNN classification rule
    - The network size, learning parameters and weights in MLPs,…
  - How do we select the “optimal” parameter(s) or model for a given classification problem?
- Performance estimation
  - Once we have chosen a model, how do we estimate its performance?
    - Performance is typically measured by the TRUE ERROR RATE, the classifier’s error rate on the population

Motivation

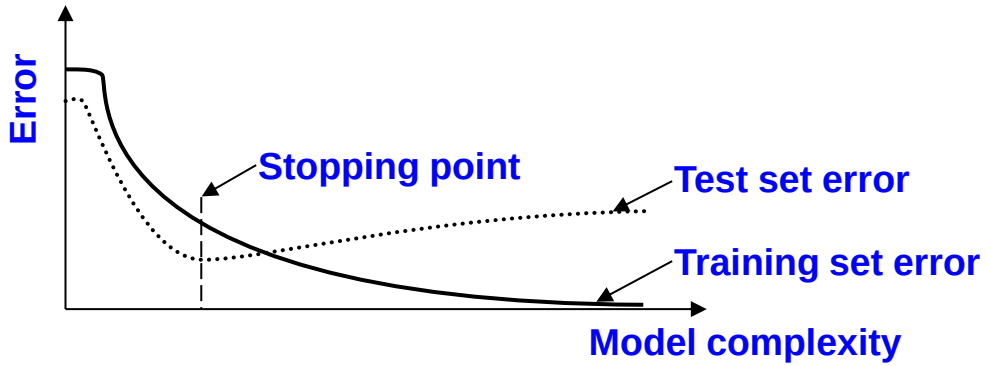
- If we had access to an unlimited number of examples these questions have a straightforward answer
  - Choose the model that provides the lowest error rate on the entire population and, of course, that error rate is the true error rate
- In real applications we only have access to a finite set of examples, usually smaller than we wanted
  - One approach is to use the entire training data to select our classifier and estimate the error rate
    - This naïve approach has two fundamental problems
      - The final model will normally overfit the training data
        - This problem is more pronounced with models that have a large number of parameters
      - The error rate estimate will be overly optimistic (lower than the true error rate)
        - In fact, it is not uncommon to have 100% correct classification on training data
  - A much better approach is to split the training data into disjoint subsets: the holdout method

The holdout method

- Split dataset into two groups
  - Training set: used to train the classifier
  - Test set: used to estimate the error rate of the trained classifier



- A typical application the holdout method is determining a stopping point in the determination of the model complexity. Do not add complexity to the model beyond this stopping point, it is useless.



The holdout method

- The holdout method has two basic drawbacks
  - In problems where we have a sparse dataset we may not be able to afford the "luxury" of setting aside a portion of the dataset for testing
  - Since it is a single train-and-test experiment, the holdout estimate of error rate will be misleading if we happen to get an "unfortunate" split
- The limitations of the holdout can be overcome with a family of resampling methods at the expense of more computations
  - Cross Validation
    - Random Subsampling
    - K-Fold Cross-Validation
    - Leave-one-out Cross-Validation
  - Bootstrap

Conclusions

- Signal processing is a key for extracting useful information from datasets
- Pattern recognition needs to cover specific stages:
  - preprocessing, dimensionality reduction, classification and validation
- Dimensionality reduction
  - PCA may not find the most discriminatory axis
  - LDA may overfit the data
- Classifiers
  - kNN is very versatile but is computationally inefficient
- Validation
  - A fundamental subject, oftentimes overlooked
  - **Acknowledgments**: We wish to thank Dr. Ricardo Gutierrez-Osuna for his kind permission to use his training material.

Thank you for your attention