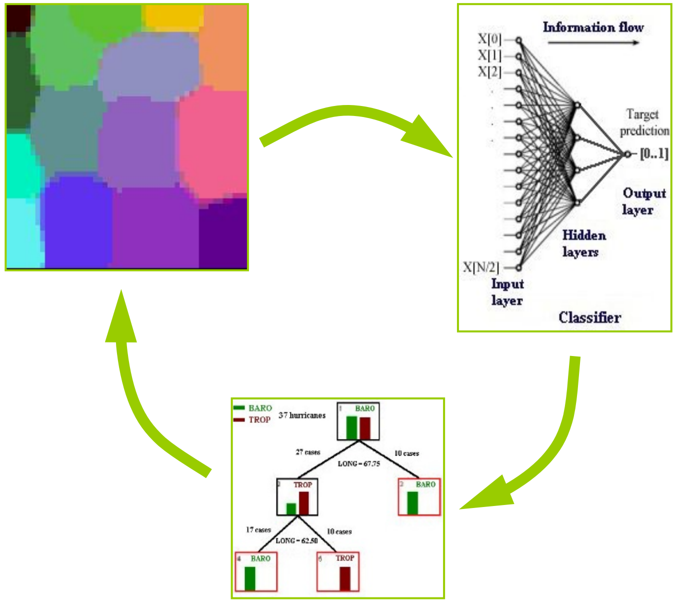
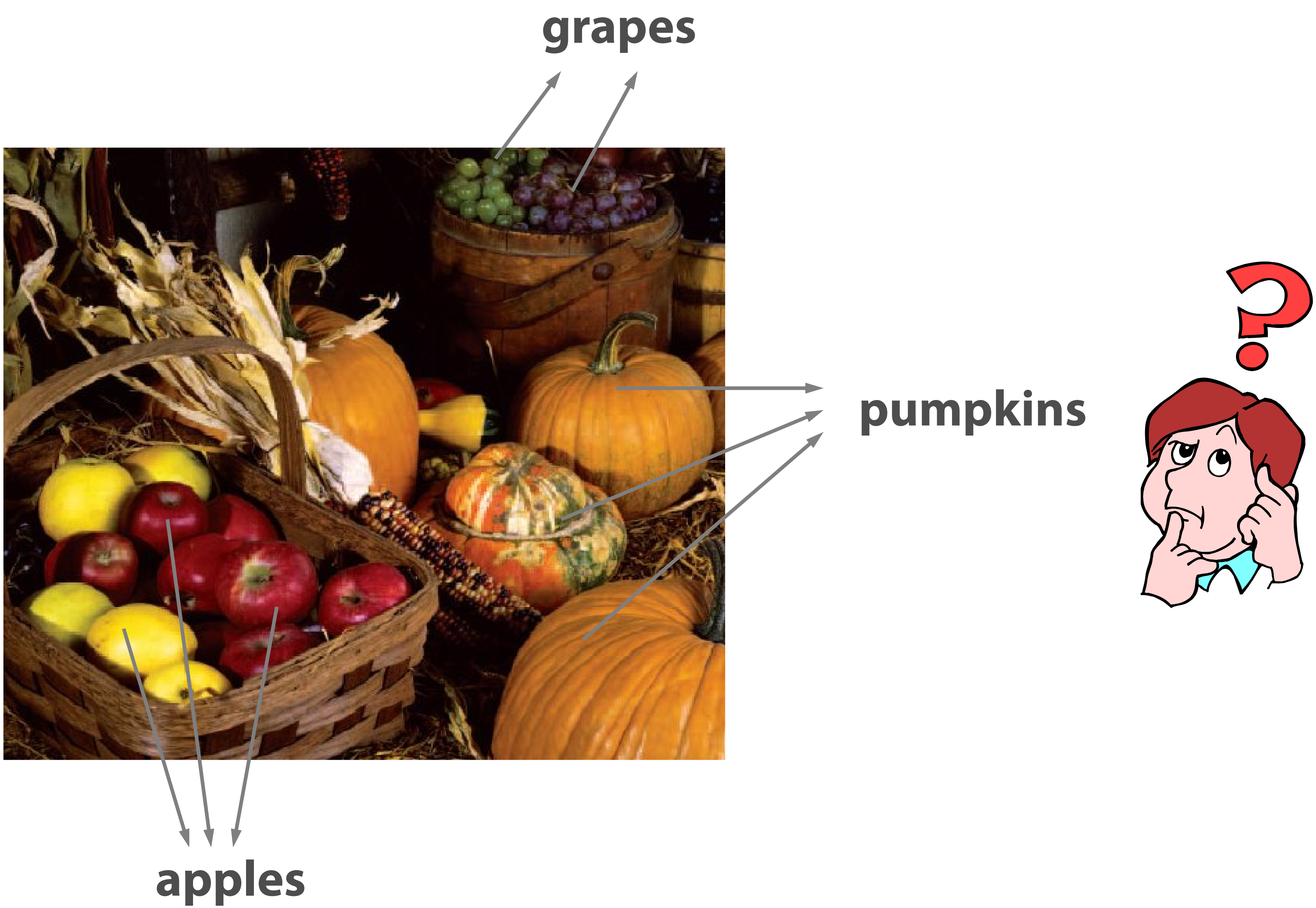
Basic machine learning concepts applied to functional network analysis



Romanian Institute
of Science and Technology

Experimental and Theoretical Neuroscience Laboratory
Str. Ciresilor 29, 400487 Cluj-Napoca, Romania
www.rist.ro

Raul C. Mureşan

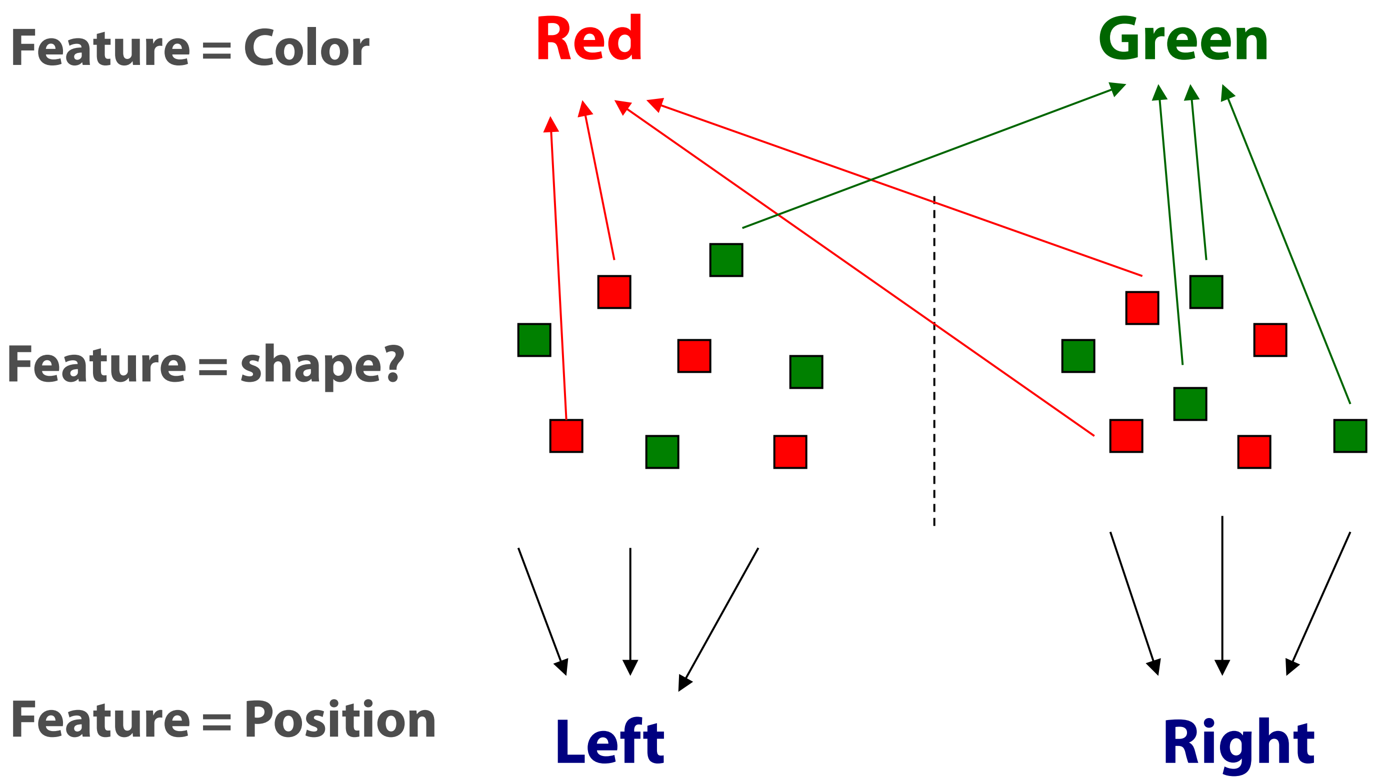

grapes
pumpkins
apples

What is classification?

Classification

(n.) The act of forming into a class or classes; a distribution into groups, as classes, orders, families, etc, according to some common relations or affinities.

Brainy Dictionary



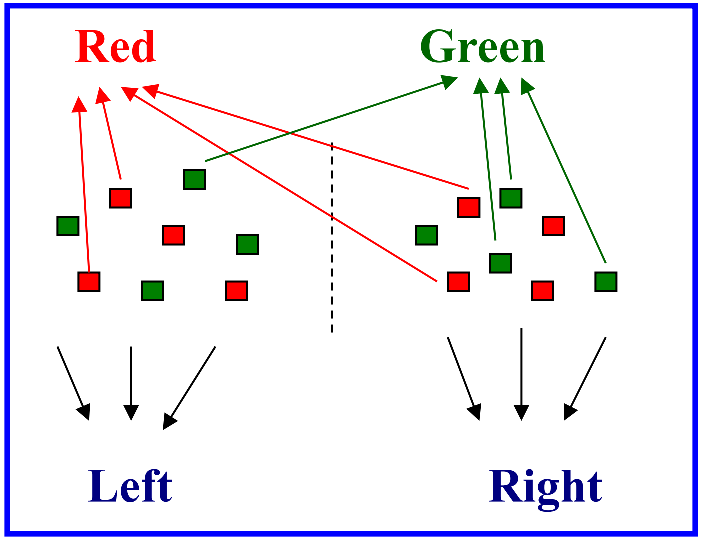
- 2 problems related to classification:

  ⇨ *Finding the <u>relevant features</u>*

  - color
  - position
  - shape (not relevant)

  ⇨ *Finding the <u>classes</u>*

  - for the <u>color</u> feature, we have: <u>Red, Green</u>
  - for the <u>position</u> feature, we have: <u>Left, Right</u>



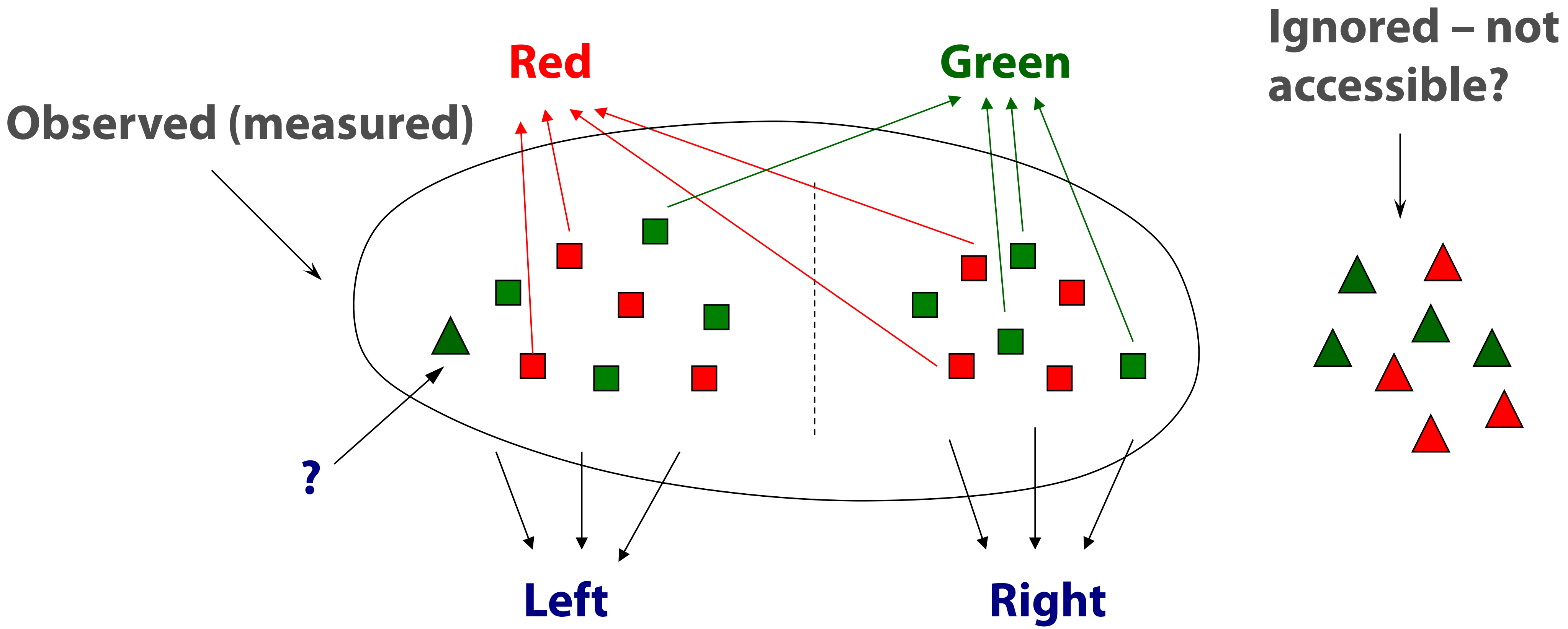
▼

The classification process depends on what the observer is looking for

⇨ NOT looking at relevant features could lead to failure of discovering the classes or classifying items into classes!!!

Finding the features

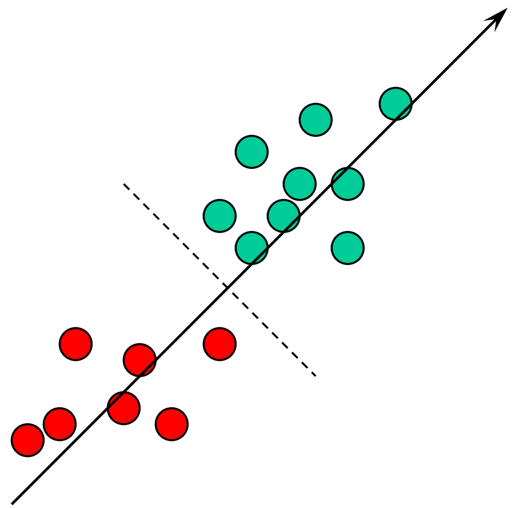
- ⇨ ***Manually*:** easy for simple systems, almost impossible for complex ones
- ⇨ ***Sampling Problem* –** without enough data, some features cannot be observed
- ⇨ ***Significance* –** is a feature due to chance or is it consistent?



- **Dimensionality of feature space**
  - ⇨ **If the items are interrelated => explosion of dimensionality of possible feature spaces (feature space >> signal space)**

    e.g. LFP, 16 channels – 120 pair wise correlations (16 x 15 / 2)
  - ⇨ **Expansion of feature space => Liquid-State Machine, Support Vector Machines**

- **Reduction techniques**
  - ⇨ **Principal Component Analysis (PCA)**
    - **based on variance analysis**
    - **linear**
    - **dangerous, can hide relevant structure**

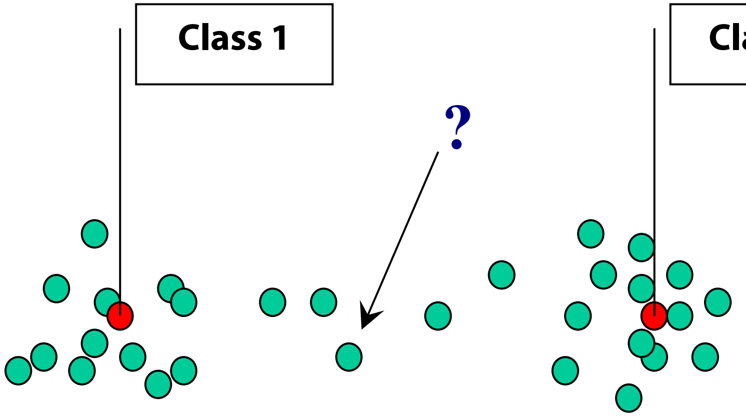


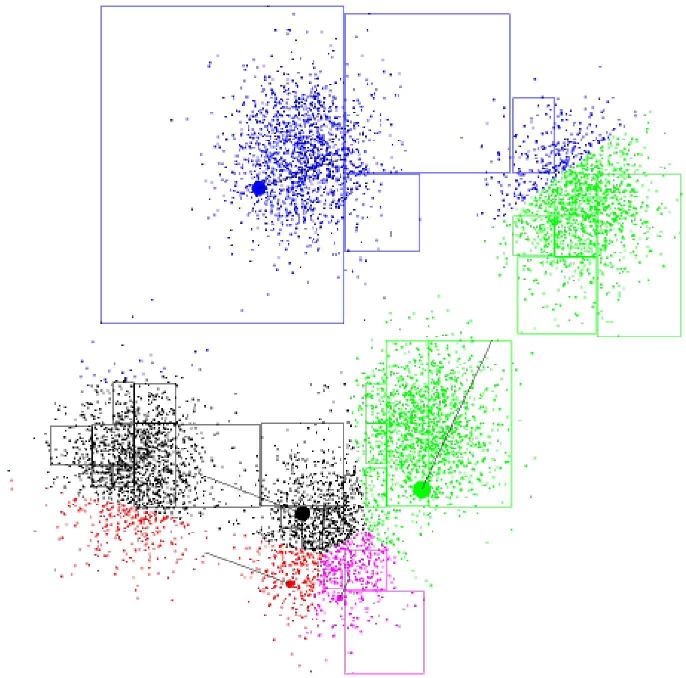
- **For functional network properties (our case)**
  - ⇨ **Features are the statistics of networks that we extract:**
    - ➢ ***node degrees***
    - ➢ ***edge betweenness***
    - ➢ ***occurrence of various motifs (e.g., T1, T2, T3...)***
    - ➢ **...**
  - ⇨ **Global network properties (1 number per network) usually not fit for machine learning ⇨ could simply plot distributions**
    - ➢ ***average path length***
    - ➢ ***global clustering coefficient***
    - ➢ **...**

Finding the classes

- **Known classes**: e.g. face classification
- **Unknown classes**: it becomes a problem of "clustering"
  - ⇨ Sampling problem – observing enough items
  - ⇨ Significance – especially in noisy data

Clustering = finding representative "points" (classes)





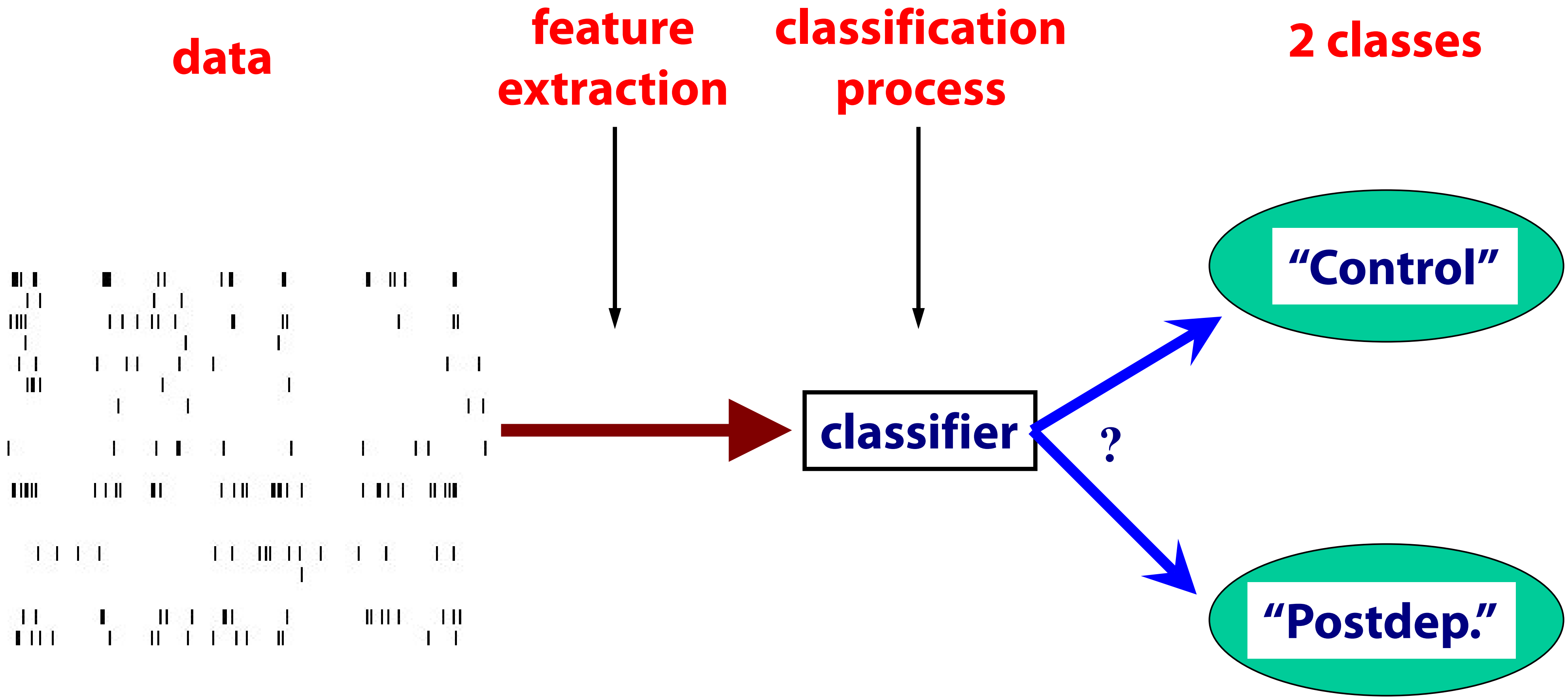
- **Classes in our case (SyBil-AA)**
  - ⇨ **We are lucky!**
  - ⇨ **We know the conditions that we want to disentangle**
    - ➢ ***Control (class 1)***
    - ➢ ***Post-dependent (class 2)***

    or

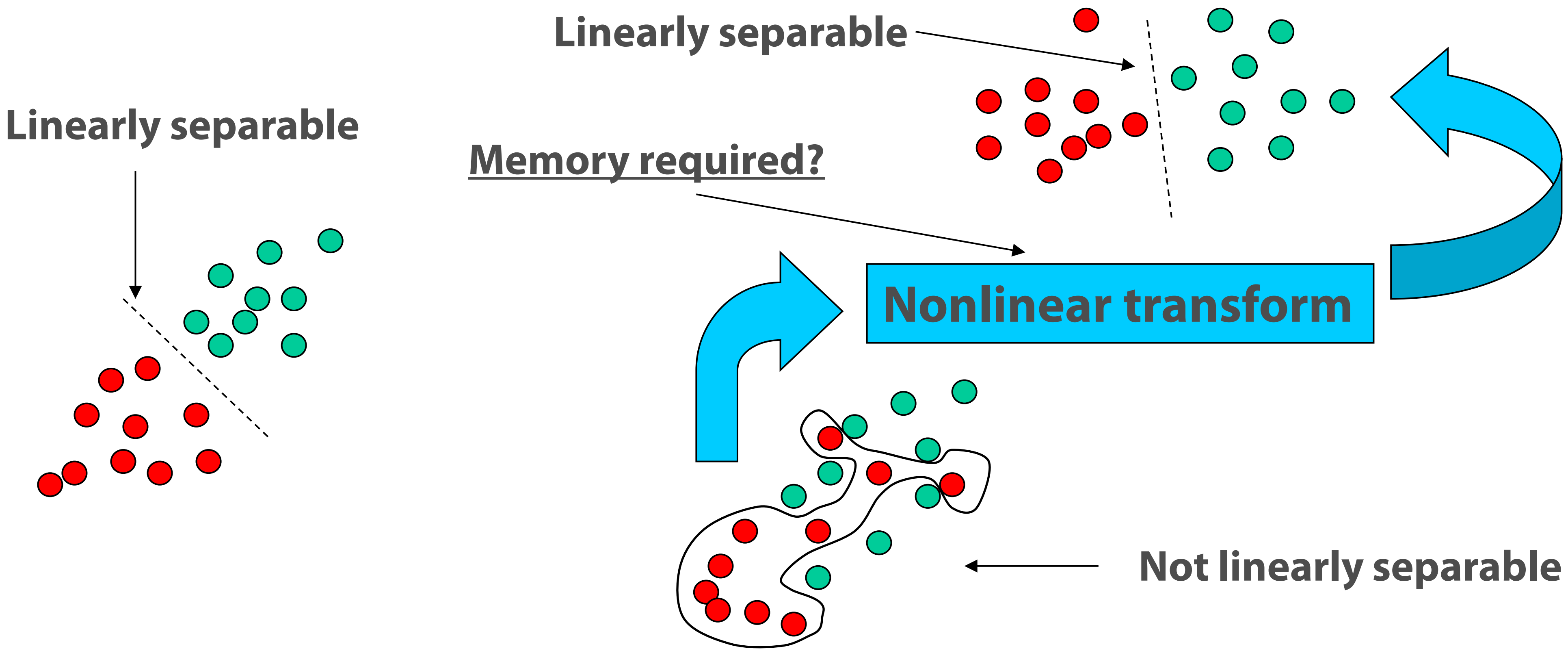
    - ➢ ***Control (class 1)***
    - ➢ ***EtOH (class 2)***
    - ➢ ***Abstinence (class 3)***
    - ➢ ***Naltrexone (class 4)***

Classifying

- *= Finding the class **a new item** belongs to ("mapping, labeling")*
- It is a **decision problem**
- Need to **train** a classifier first (machine learning)

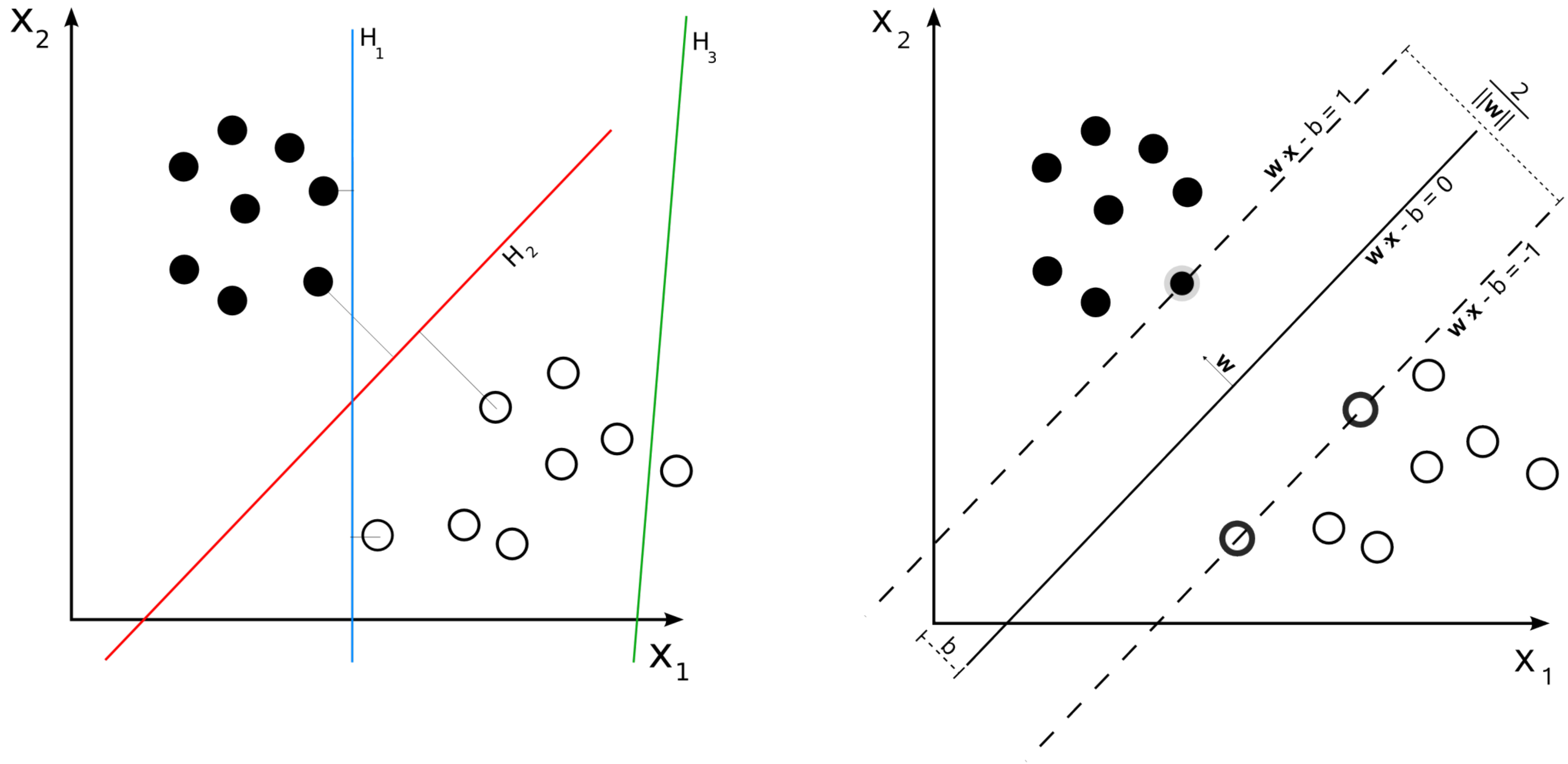


- Learning can be easier or harder depending on the separatrix
- Linear separability – nice ⇨ powerful linear algebra
- Nonlinear systems – unfortunately most real world systems (brain)
  - ⇨ use nonlinear classifiers (e.g., MLP), or
  - ⇨ use an intermediate, nonlinear, feature space transform (SVM, LSM)



Types of classifiers (I): the Support Vector Machine (SVM)

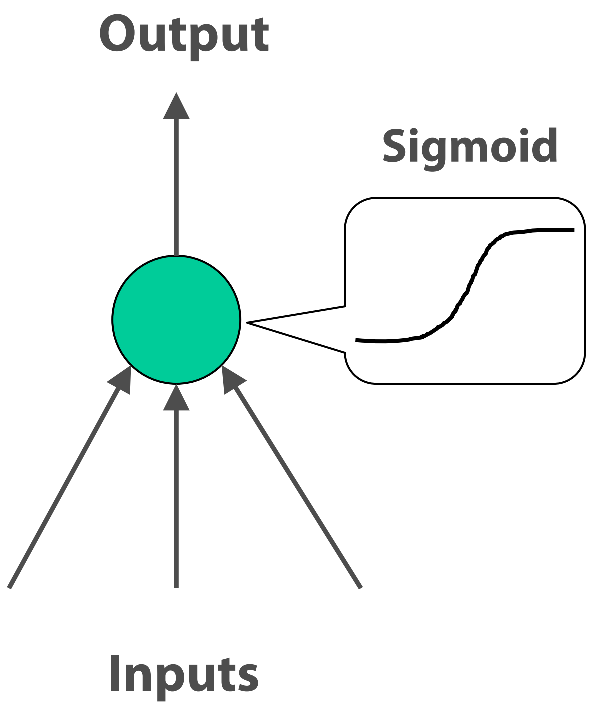
- **Optimal boundary decision**: maximum margin method

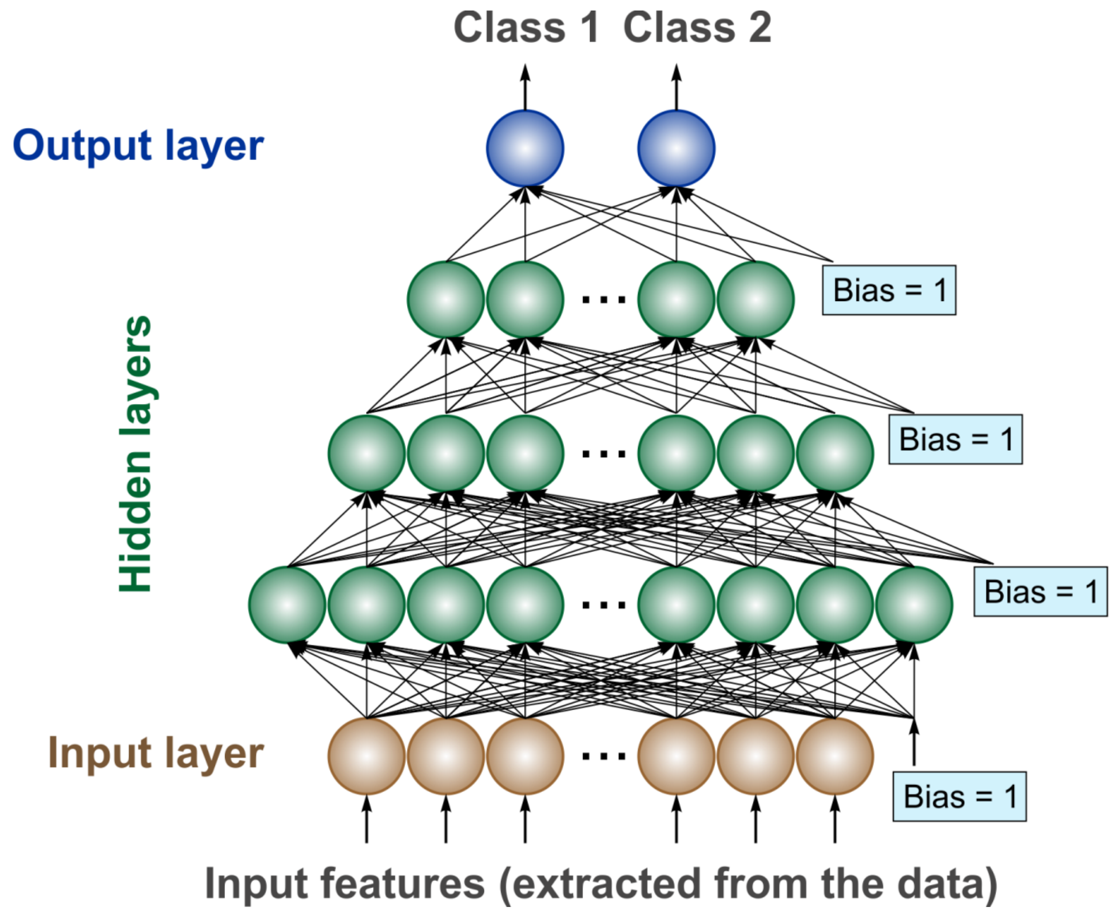


- **Standard SVM is linear… (not good enough for brain data)**
- **Non-linear SVM: is a kernel method - the kernel trick**

Types of classifiers (II): the MultiLayer Perceptron (MLP)

- Also called **Neural Networks**
- Huge success recently ⇨ **Deep learning**
- Non-linear activation functions ⇨ inherently **non-linear**





Important facts about classification

- *Classification could be a **highly non-linear** mapping*
  - ☞ Check **effect size**!
  - ☞ Draw **careful** conclusions…
- *Go back to data and identify the **relevant features***
  - ☞ Because classifiers could **jump onto anything** that is informative…
  - ☞ Classifying time series could be especially **misleading**
- *For **non-linear** classifiers, difficult to conclude on the original data*
  - ☞ The **nonlinear mapping can confuse you** (small effects in the original space could be large in the transformed space)
  - ☞ You need to **destroy features of the original data** to understand what the classifier extracted

Solutions

- *Use simple classifiers, when possible*
  - ☞ Average multidimensional vectors for example
  - ☞ They can do reasonably well on the right features
  - ☞ Not clear if they work on functional network data ☹
- *Always check for effect size*
  - ☞ To understand how robust the discrimination is
- *Go back to the original data and understand what the classifier extracted*
  - ☞ Only then you could conclude about relevant brain processes