Theory of statistical learning

Introduction to machine learning and pattern recognition

R. Flamary

January 10, 2019

Course overview

What is Pattern Recognition (PR)?

Definitions from the litterature

- The process of assigning a pre-specified category to a physical object or event (*Duda and Hart*).
- Using several examples of complex signals and associated labels (or decisions), PR is a process of automatic decisions for new signals. *Ripley*
- The process of assigning a name y to an observation x. *Schurmann*

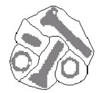
Objective of Pattern Recognition, Machine learning

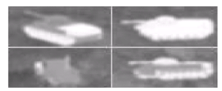
Teach as machine to process automatically a large amount of data (signals, images) in order to solve a given problem.

Exemples of pattern recognition problems

Vision

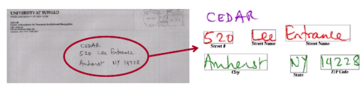
- Product inspection in manufacturing
- Military targets.





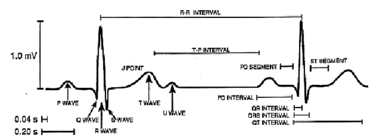
Optical Characters Recognition

- Automatic mail classification.
- Automatic checks amount reading.



Computer Aided Diagnosis

- Medical imagery, EEG, ECG.
- Assist physicians (not replace them).



Types of ML problems

Unsupervised learning

- **Clustering** Organize objects in similar groups (taxonomy of animal species).
- **Probability Density Estimation** Estimate probability distributions from data (distribution of noise).
- **Dimensionality reduction** Represent large dimensional data in a small dimension space for better visualization and interpretation (recommender systems).

Supervised learning

- **Classification** Assign a class to an observation (character recognition, weather presence of rain).
- **Régression** Predict a continuous value from an observation (weather temperature).

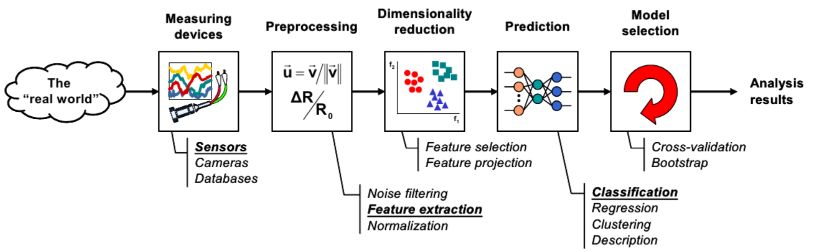
Reinforcement learning

Train a machine to choose actions that maximize a reward (games).

Components of a ML system

A classic system is composed of

- A sensor
- A pre-processing of the data
- A feature extraction step
- A classification step
- A set of examples (training set)



Training datasets

Unsupervised learning

- $\mathbf{x} \in \mathbb{R}^d$ is an observation with d features.
- The training set contains the observations $\{\mathbf{x}_i\}_{i=1}^n$ where n is the number of training points (examples).
- Examples are often stored as a matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ with $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top$ contains the training samples as lines (features are columns).
- d and n define the dimensionality of the learning problem.

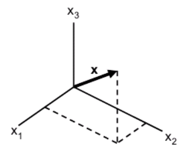
Supervised learning

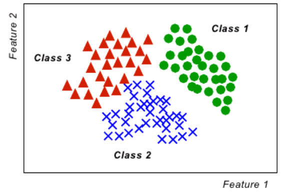
- A label $y_i \in \mathcal{Y}$ is associated to each trainings sample $\mathbf{x}_i$.
- As for the observations the value to predict (label) can be concatenated in a vector $\mathbf{y} \in \mathcal{Y}^n$
- Prediction space $\mathcal{Y}$ can be:
  - $\mathcal{Y} = \{-1, 1\}$ or $\mathcal{Y} = \{1, \ldots, m\}$ for classification problems.
  - $\mathcal{Y} = \mathbb{R}$ for regression problems.
  - Structured for structured prediction (graphs,...).

Features and patterns

- A feature is a distinct trait, or detail of an object. It can be symbolic (ex : a color) or numeric (ex : a size).
- **Definition**
  - A combination of features is represented as a vector $\mathbf{x}$ of dimensionality d.
  - The space of size d is called the representation/feature space.
  - Objects can be represented as points in this space. This representation is called scatter plot

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





- A pattern is a set of traits for an observation. In a classification problem a pattern is composed of a feature vector and a label

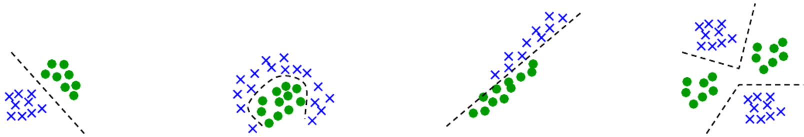
Features

What is a “good” feature ?

The quality of a feature depends on the learning problem.

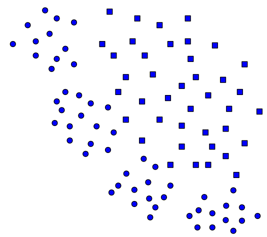
- **Classification** Samples from the same class should have similar values of the feature, examples from different classes should have different values.
- **Regression** The feature should help better predict the value (correlation or at least non-independence with the value to predict).

Other properties



Linear separability *Non-linear separability* *Highly correlated features* *Multi-modal*

Unsupervised learning, data description/exploration



Let $\{\mathbf{x}_i\}_{i=1}^n$ be a training set of n samples of dimension d

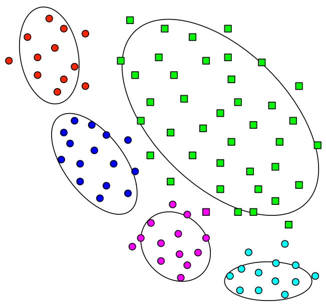
Objectives

- **Clustering** $\{\mathbf{x}_i\}_{i=1}^n \Rightarrow \{\hat{y}_i\}_{i=1}^n$ où $\hat{y}$ is the labels of a group.
- **Probability density estimation** $\{\mathbf{x}_i\}_{i=1}^n \Rightarrow p(\mathbf{x})$.
- **Generative modeling** $\{\mathbf{x}_i\}_{i=1}^n \Rightarrow p(G(\mathbf{z})) = p(\mathbf{x})$ with $\mathbf{z} \sim N(0, \sigma^2)$.
- **Dimensionality reduction** $\{\mathbf{x}_i \in \mathbb{R}^d\}_{i=1}^n \Rightarrow \{\tilde{\mathbf{x}}_i \in \mathbb{R}^p\}_{i=1}^n$ avec $p \ll d$.

Clustering

Objective

- Organize training examples in groups.
- $\{\mathbf{x}_i\}_{i=1}^n \Rightarrow \{\hat{y}_i\}_{i=1}^n$ where $\hat{y} \in \mathcal{Y}$ represents a class ($\{1, \ldots, m\}$)
- Parameters:
  - m number of classes.
  - Similarity measure.



Methods

- k-means.
- Gaussian mixtures.
- Spectral clustering.
- Hierachical clustering.

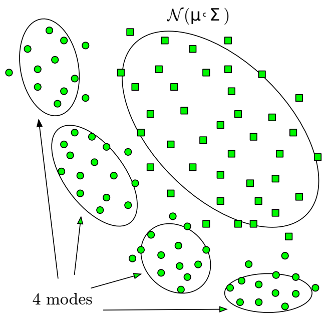
Examples

- Animal Taxonomy.
- Gene clustering.
- Social networks.

Probability density estimation

Objective

- Estimate the probability distribution that generated the data.
- $\{\mathbf{x}_i\}_{i=1}^n \Rightarrow p(\mathbf{x})$ where $p(\mathbf{x})$ is a probability density ($\int p(\mathbf{x}) d\mathbf{x} = 1$)
- Model can be generative.
- Parameters:
  - Type of distribution (Gaussian, ...).
  - Parameters of the law (μ, Σ)



Methods

- Parzen density estimation.
- Histogram (1D/2D).
- Gaussian mixture.

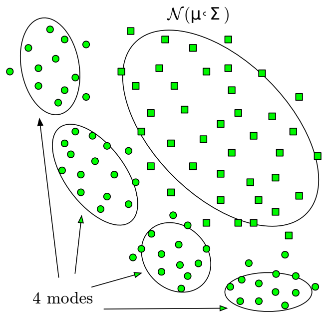
Examples

- Noise estimation.
- Generative data (face,...).
- Novelty detection.

Generative modeling

Objective

- Estimate a mapping function G that generate similar samples as in $\{\mathbf{x}_i\}_{i=1}^n$.
- $G(\mathbf{z})$ with $\mathbf{z} \sim \mathcal{N}$ approximates the distribution of the data.
- Parameters:
  - Type of distribution for $\mathbf{z}$ (Gaussian, ...).
  - Type of function G.
  - Measure of similarity between $G(z)$ and $\hat{p}(\mathbf{x})$.



Methods

- PCA for Gaussian data.
- Generative Adversarial Networks (GAN)
- Variational Auto-Encoders (VAE)

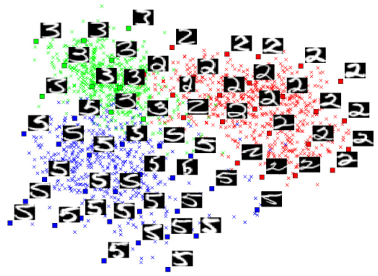
Examples

- Generate realistic images.
- Style adaptation.
- Data modeling.

Dimensionality reduction, visualization

Objective

- Project the data into a low dimensionnal space.
- $\{\mathbf{x}_i \in \mathbb{R}^d\}_{i=1}^n \Rightarrow \{\tilde{\mathbf{x}}_i \in \mathbb{R}^p\}_{i=1}^n$ with $p \ll d$ (often $p = 2$).
- Usage for visualization, pre-processing, denoising.
- Parameters:
  - Type of projection.
  - Similarity measure.



Methods

- Feature selection.
- Principal Component Analysis (PCA).
- Non-linear dimensionality reduction (MDS, tSNE, AutoEncoders)

Examples

- Visualization onto 2D/3D.
- Data interpretation (is features space discriminant?).
- Recommender systems.

Supervised prediction

Let $\{\mathbf{x}_i, y_i\}_{i=1}^n$ be the training set composed of observations $\mathbf{x}_i \in \mathbb{R}^d$ of dimensionality d and the values to predict $y_i \in \mathcal{Y}$.

Objective

- We train a function $f(\cdot) : \mathbb{R}^d \rightarrow \mathcal{Y}$ from a training dataset.
- Types of prediction:
  - **Classification**
    $f(\cdot)$ predicts a class (discrete output) either binary $\mathcal{Y} = \{-1, 1\}$ or multiclass $\mathcal{Y} = \{1, \ldots, m\}$.
  - **Regression**
    $f(\cdot)$ predicts a continuous value ($\mathcal{Y} = \mathbb{R}$) or several ($\mathcal{Y} = \mathbb{R}^p$).

Linear function

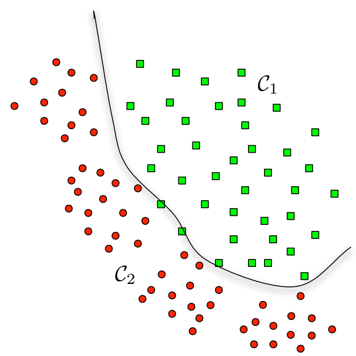
$$f(\mathbf{x}) = \sum_{j=1}^{d} w_j x_j + b = \mathbf{w}^\top \mathbf{x} + b$$

parametrized by $\mathbf{w} \in \mathbb{R}^d$ and $b \in \mathbb{R}$

Binary classification

Objective

- Train a function that predicts -1 or 1.
- $\{\mathbf{x}_i, y_i\}_{i=1}^n \Rightarrow f(\mathbf{x})$.
- Prediction: sign of $f(\cdot)$
- $f(\mathbf{x}) = 0$: decision boundary.
- Parameters:
  - Type of function.
  - Performance measure (what is optimized).



Methods

- Bayesian classifier (from density estimation)
- Linear discrimination
- Support Vector Machines.
- Decision trees, random forests.

Examples

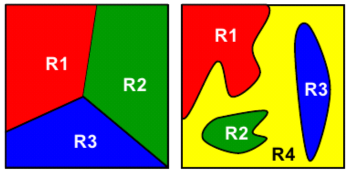
- Optical Character Recognition.
- Computer Aided Diagnosis.
- Computer Vision.
- Weather prediction (rain vs sun).

Multiclass classification

Principle

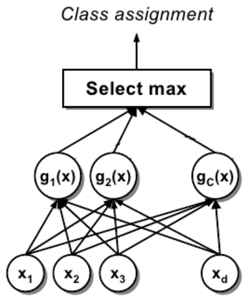
A classifier does a partition of the feature space in several regions associated to different classes.

- Boundaries between the regions are called decision boundaries
- Classifying a new example $\mathbf{x}$ consists in finding its region and assign the corresponding label.



One-Against-All strategy

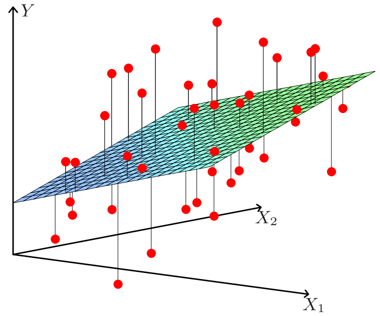
- In a One-Against-All strategy classifier is represented by an ensemble of discriminant functions $g_i(\mathbf{x})$: the predicted class for sample $\mathbf{x}$ is class j such that $g_j(x) > g_i(x)$ for all $i \neq j$.
- The output score can be used to estimate probabilities for each class using the `softmax` function instead of `max`.



Regression

Objective

- Train a function predicting a continuous value.
- $\{\mathbf{x}_i, y_i\}_{i=1}^{n} \Rightarrow f(\mathbf{x})$.
- Parameters:
  - Type of function.
  - Performance measure.
  - Prediction error.



Methods

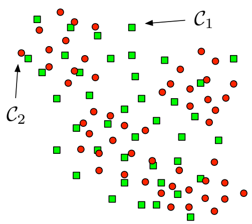
- Least Square (LS).
- Ridge regression.
- Lasso.
- Kernel regression.

Examples

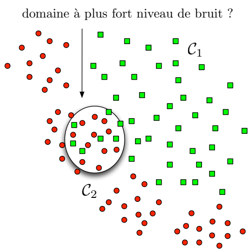
- Movement prediction.
- Inverse problems.
- Weather prediction (temperature).

Real data (1)

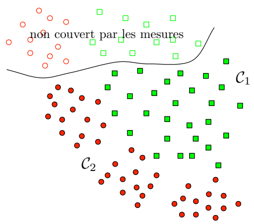
- Unrelated features



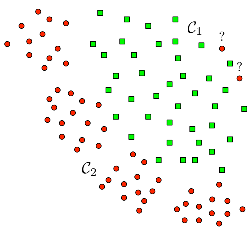
- Noise



- Non-representative



- Outliers

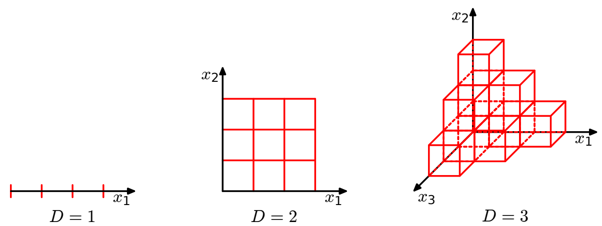


Real data(2)

Dataset dimensionality

We always have a finite number n of training samples of dimensionality d.

Curse of dimensionality



The curse of dimensionality illustrate the fact that when the dimensionality of the data increase the number of samples necessary for sampling the domain increases exponentialy with the dimension.

Model selection

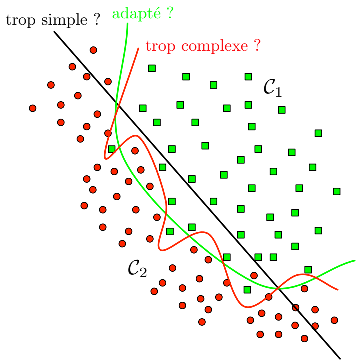
How to select ?

Model	Training	Prediction
Too simple	- -	- -
Adapted	+	+
Too complex	++	- -

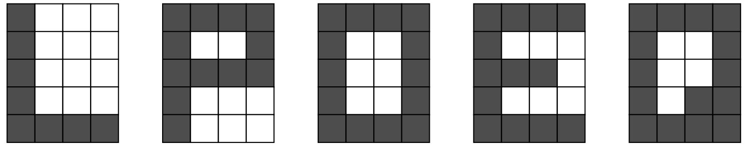
- Over-fitting occurs when the model is too complex. (remember only the training samples)
- We want to predict well on new data!

Validation

- Split the data in learning/validation sets.
- Maximize performance on validation data.
- Validation needs a good performance measure.



Simple classification problem



- Develop an algorithm able to discriminate between the 5 classes L,P,O,E,Q
  - Find discriminant features (pixels)
  - Propose a binary tree classifier using only pixel values.