Audio processing

Mathieu Lagrange

CENTRALE NANTES

February 8, 2019

Course website: https://github.com/mathieulagrange/datasim

Outline

1. Fundamentals in Machine Learning
2. Bias / Variance Tradeoff
3. Dimensionality
4. Timbre
5. Harmony

Outline

Outline

1. Fundamentals in Machine Learning
2. Bias / Variance Tradeoff
3. Dimensionality
4. Timbre
5. Harmony

Outline

Outline

1. Fundamentals in Machine Learning
2. Bias / Variance Tradeoff
3. Dimensionality
4. Timbre
5. Harmony

Problem solved

- does this item belongs to A or B ? (closed set classification)
- does this item belongs to A ? (closed set classification)
- is this item very A or only a bit ? (regression)
- how my data is structured ? (clustering)
- is this item very different from the usual ones ? (anomaly detection)

Types of learning

- **Reinforcement learning:** the world is completely described (explicit reward)
- **Supervised learning:** the relation between the items and the corresponding supervisory signal is known for some items
- **Unsupervised learning:** Discover the structure and regularities of the items by observing them (and potentially living with them)

Types of learning

- **Reinforcement learning:** The machine predicts a scalar reward given once in a while (A few bits for some samples)
- **Supervised learning:** The machine predicts a category or a few numbers for each input (10 to 10,000 bits per sample)
- **Unsupervised learning:** The machine predicts any part of its input for any observed part, eg predicting future frames in videos (Millions of bits per sample)

Yan LeCun Unsupervised learning seminar 2016

Types of learning

Unsupervised Learning is the "Dark Matter" of AI

ML trends: Unsupervised Learning

Unsupervised learning is the only form of learning that can provide enough information

- to train large neural nets with billions of parameters.
- Supervised learning would take too much labeling effort
- Reinforcement learning would take too many trials

Supervised learning

- Let $y \in A$ be the labels assigned to some items $x \in \mathcal{R}^d$
- n couples are available for training: $(x_i, y_i)_{i \leq n}$
- they are assumed to be iid samples $(X_i, Y_i)_{i \leq n}$ from non observed distributions (X, Y)
- from a given x, the system predicts an estimate $\tilde{y}$
- parameters of the system are optimized such that $\tilde{y}_i \approx y_i$

Generalization

- We wish that the precision obtained on the training set is preserved over unseen data
- this is called generalization capabilities

Learning

- the learning system computes $\tilde{y} = \tilde{f}(x)$
- $\tilde{f}$ is chosen among a class $\mathcal{H}$
- assuming that the y paired with x is unique, $y = f(x)$
- The system then compute an approximation $\tilde{f}$ of f.

Empirical risk

In order to qualify $\tilde{f}$

- a measure of the risk $r(\tilde{y}, y)$ shall be defined
- in a regression problem, the risk can be the quadratic one: $r(\tilde{y}, y) = (\tilde{y}, y)^2$
- in a classification problem, the risk can count the number of classification mistakes
- The empirical risk on the data is then

$$\tilde{R}_e(\tilde{f}) = \frac{1}{n} \sum_{i=1}^{n} r(\tilde{f}(x_i), y_i)$$

Generalization risk

Training data being iid samples $(X_i, Y_i)_{i \leq n}$

- The empirical risk is

$$\tilde{R}_e(\tilde{f}) = \frac{1}{n} \sum_{i=1}^{n} r(\tilde{f}(X_i), Y_i)$$

- the generalization risk is thus

$$\tilde{R}(\tilde{f}) = \mathbb{E}[r(\tilde{f}(X), Y)]$$

- we want to minimize the generalization risk, though we have only access to the empirical one.

Risk

The important questions are:

- how do we measure and major the difference between the empirical error $\tilde{R}_e(\tilde{f})$ and the generalization error $R(\tilde{f})$?
- how do we ensure that $R(\tilde{f})$ is low ?

Bias / Variance Tradeoff

Given

- a valid minimizer $\tilde{f} = \underset{h \in \mathcal{H}}{\operatorname{argmin}} R(h)$
- the best approximation $f_a = \underset{h \in \mathcal{H}}{\operatorname{argmin}} \tilde{R}_e(h)$

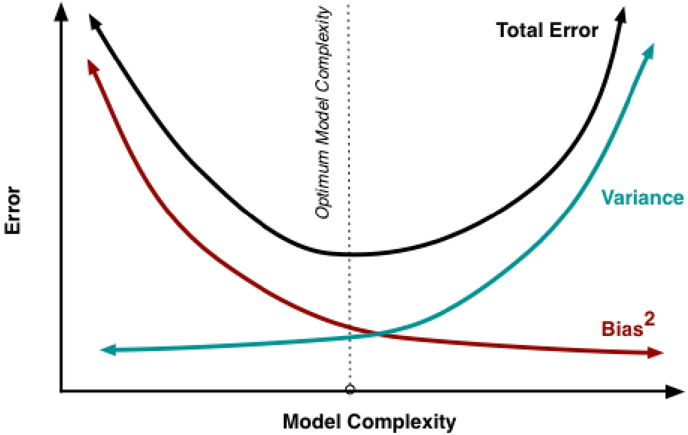
we have

$$R(\tilde{f}_a) \leq R(\tilde{f}) \leq R(\tilde{f}_a) + 2 \max_{h \in \mathcal{H}} |R(h) - \tilde{R}_e(h)|$$

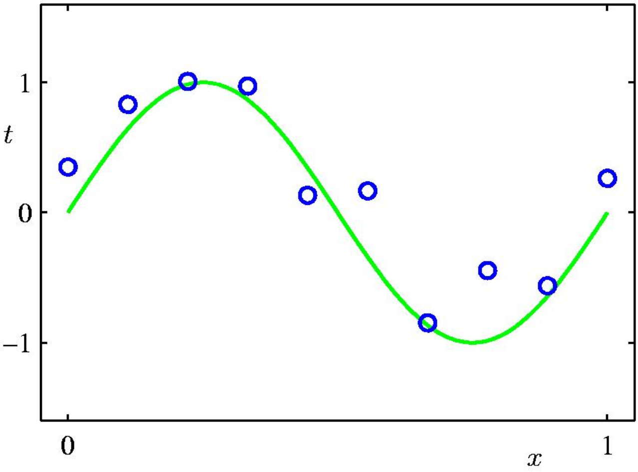
where

- $R(\tilde{f}_a)$ is the minimal generalization error
- $\max_{h \in \mathcal{H}} |R(h) - \tilde{R}_e(h)|$ is the fluctuation error between the empirical risk and average risk over the class of predictors

Bias / Variance Tradeoff



Toy example: polynomial curve fitting



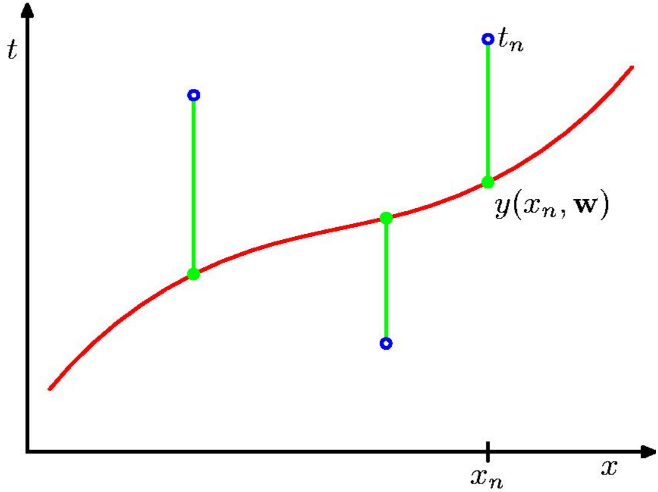
green line: underlying process
blue dots: samples

Toy example: polynomial curve fitting

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \ldots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

- Approximator: polynomial
- Complexity parameter: *M*

Toy example: polynomial curve fitting



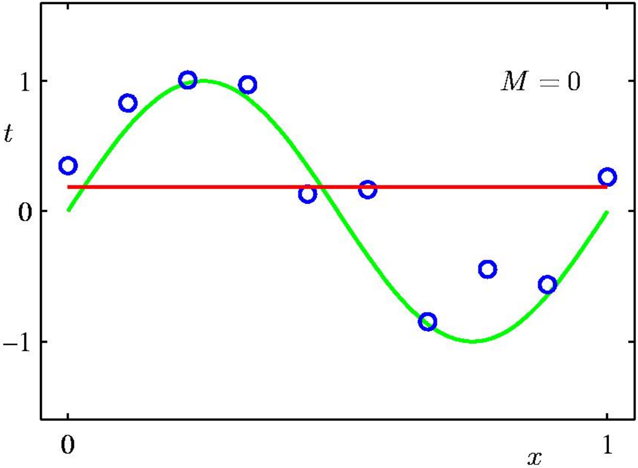
Risk

Toy example: polynomial curve fitting

$$E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w}) - t_n\}^2$$

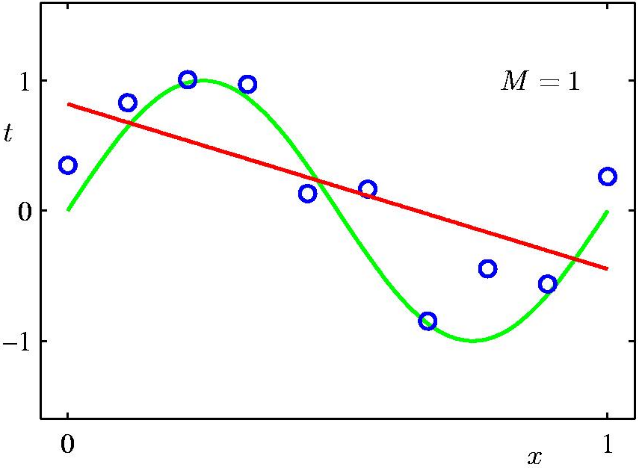
Risk: quadratic loss

Toy example: polynomial curve fitting



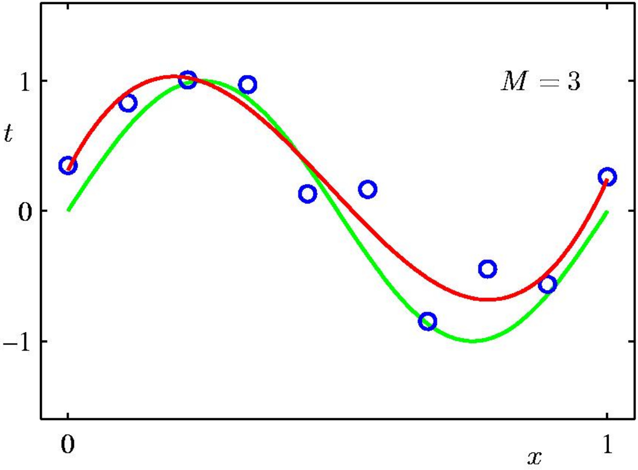
Underfitting

Toy example: polynomial curve fitting



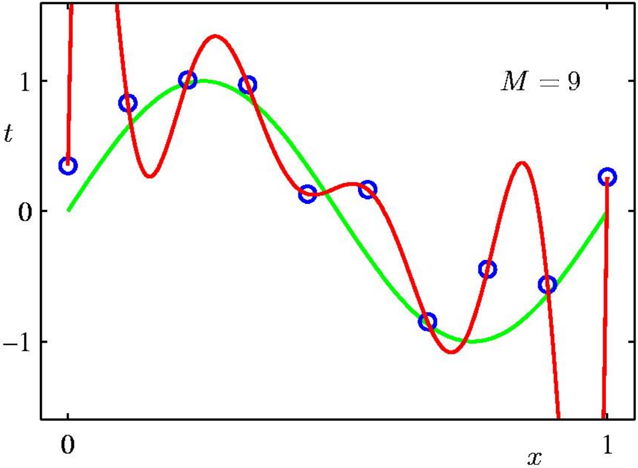
Underfitting

Toy example: polynomial curve fitting



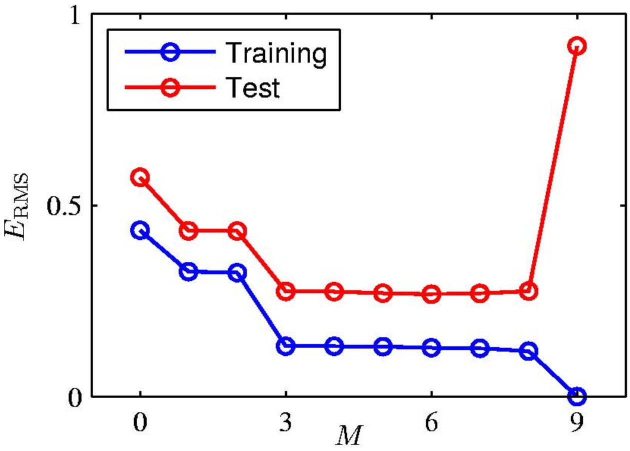
Underfitting

Toy example: polynomial curve fitting



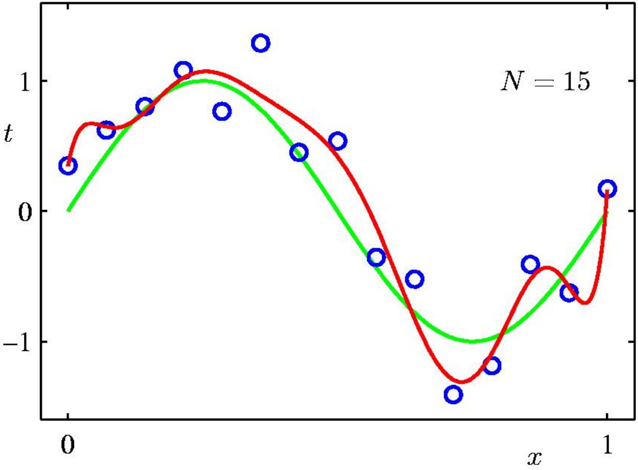
Overfitting

Toy example: polynomial curve fitting



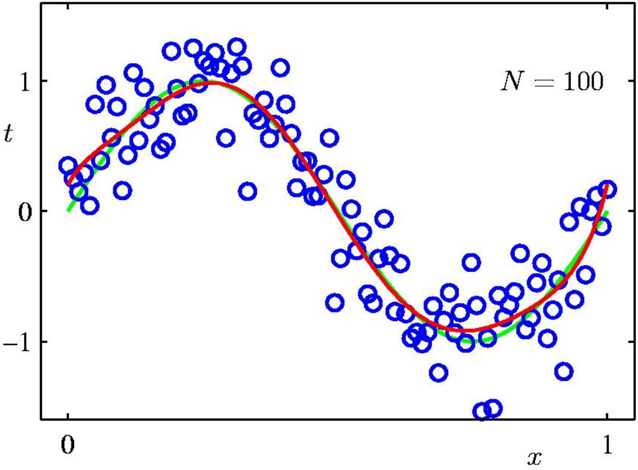
Train / Test

Toy example: polynomial curve fitting



More data

Toy example: polynomial curve fitting



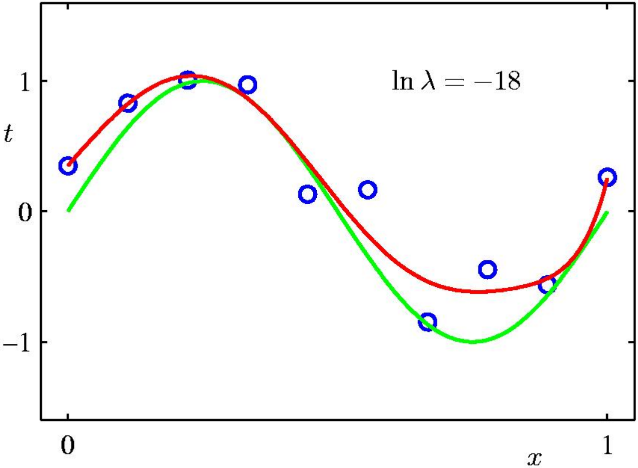
Much more data

Toy example: polynomial curve fitting

$$\widetilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n,\mathbf{w}) - t_n\}^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2$$

Penalization

Toy example: polynomial curve fitting



Penalization

learning

- can be viewed as an interpolation problem
- around known values
- the more regular the manifold, the easier the task
- regularity is linked to differentiability (Fréchet, Gâteaux)

Defining regularity

- we consider the *Lipschitz* regularity
- we say that $f : \Omega \to \mathbb{R}$ is locally Lipschitz
- if there exist $C_x > 0$ so that

$$\forall x' \in \Omega, \left| f(x) - f(x') \right| \leq C_x \left\| x - x' \right\|$$

- f is uniformly lipschitz on Ω if for all $x \in \Omega$ there is $C > 0$ so that $C_x < C$.

The approximator

- Let us consider a nearest neighbor classifier to approximate the manifold of interest:

$$\tilde{f}(x) = f(x_i) \text{ pour } i = \arg\min_{i' \leq n} \|x - x_{i'}\|$$

- This algorithm does not allow the control of the fluctuation error (high variance), but is efficient to reduce the approximation error as it compute piecewise constant approximations around training examples.

The training data

- Assuming the ideal case where the training data points are uniformly spread over $\Omega = [0, 1]^d$
- we can show that to achieve a given prediction error $C\epsilon$
- the number of traiing samples shall be

$$n \geq \frac{\epsilon^{-d} d^{d/2}}{(2\pi e)^{d/2}}$$

- which for $n > 5$ is totally impractical.
- This is the *curse of dimensionality*.

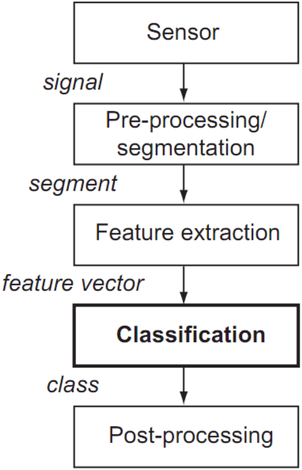
And

- one second of sound is $\in \mathbb{R}^{44100}$
- one image is $\in \mathbb{R}^{10^8}$
- one hour of video is ...
- ??

Dimensionality reduction

- The main assumption in ML is that there exist
- a lower dimensionality manifold over which the functions we want to approximate are
- angle 1: characterize this manifold
- angle 2: identify invariance properties of this manifold

Data processing pipeline



Definition

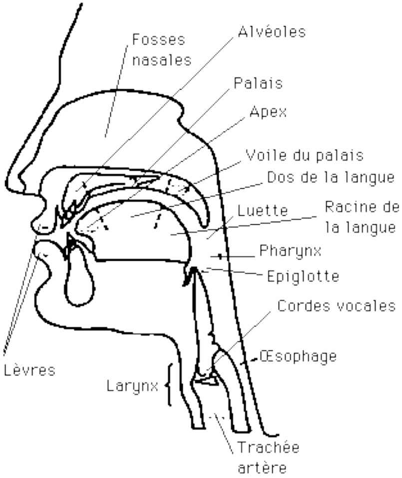
Timbre is

- the character or quality of a musical sound or voice
- as distinct from its pitch
- and intensity.

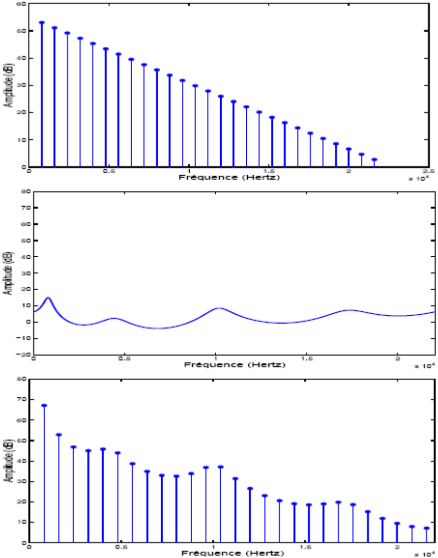
Applications

- Speaker recognition
- Speech recognition
- Instrument recognition
- Musical Genre recognition

Speech production



Source filter model



Expressing invariance for timbre

We can seek local / global invariance or stability to feature change

- time shift
- amplitude change
- pitch shift

Time shift

Invariance to local time shift can be achieved

- by considering the magnitude spectrogram
- as the phase is discarded
- the representation is invariant to time shift smaller than the size of the analysis window

Amplitude change

We seek stability here by considering the logarithmic compression of magnitude values

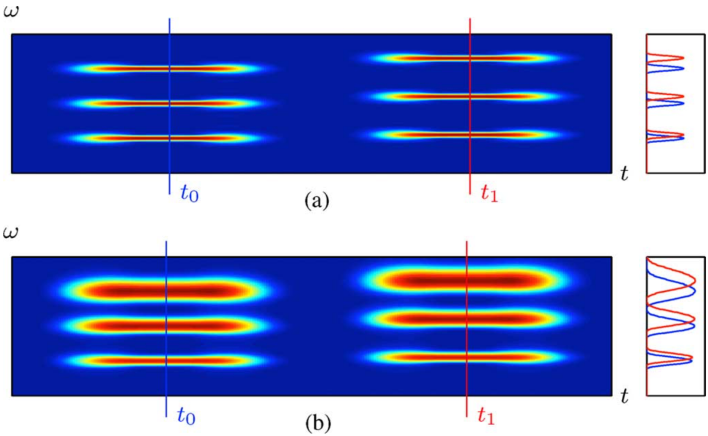
- compresses the dynamic range of values
- rectify the amplitude across frequencies
- makes frequency estimates less sensitive to slight variations in input (power variation due to speaker mouth moving closer to mike)
- Ecology: Human response to signal level is logarithmic

Stability to small pitch shift

We first seek stability to small pitch shift

- by considering a logaritmic scale of the frequencies
- Ecology: Human hear frequency scale is logarithmic
- Mathematical explanation through properties of the scalogram

Scalogram

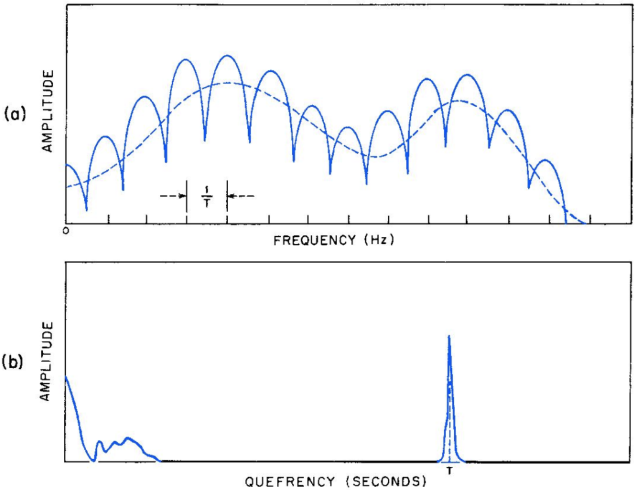


Invariance to Pitch ripples

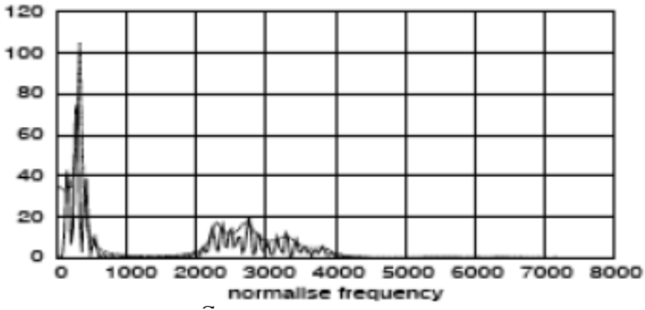
When we focus on timbre, we want to be invariant to pitch change

- In a source / filter model
- the periodic source induces many peaks in the spectrum
- we have to get rid of them

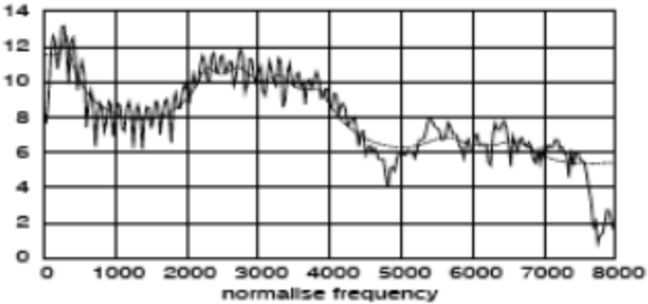
Simplified source: filter spectrum



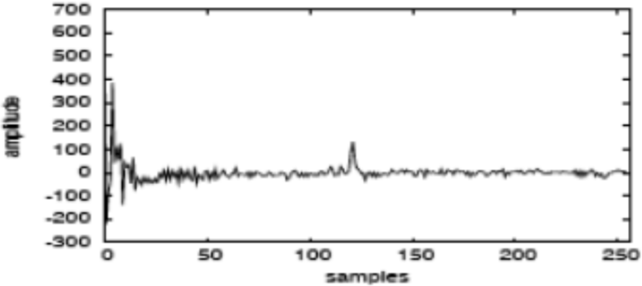
The cepstrum



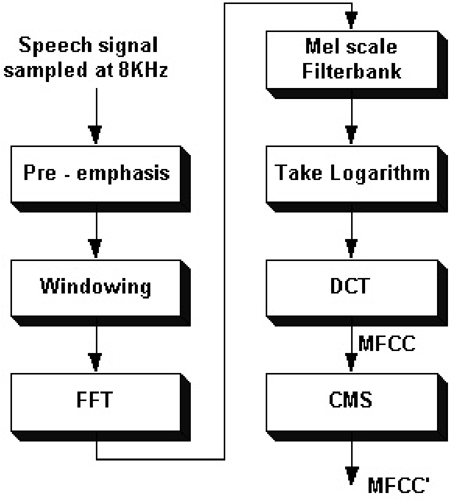
The cepstrum



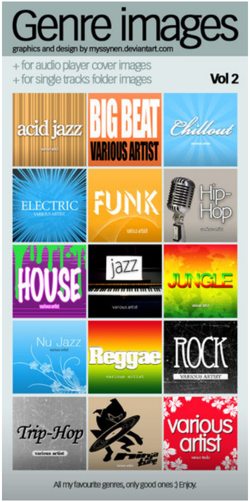
The cepstrum



Mfcc


Speech signal
sampled at 8KHz
Pre - emphasis
Windowing
FFT
Mel scale
Filterbank
Take Logarithm
DCT
MFCC
CMS
MFCC'

Genre Classification



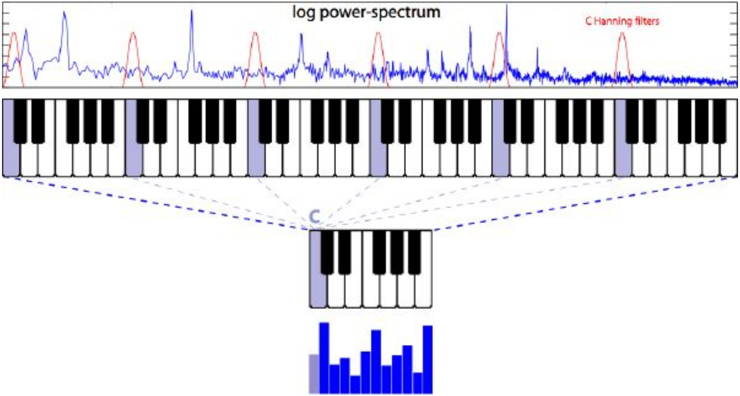
1. Fundamentals in Machine Learning
2. Bias / Variance Tradeoff
3. Dimensionality
4. Timbre
5. Harmony

Focus on Harmony

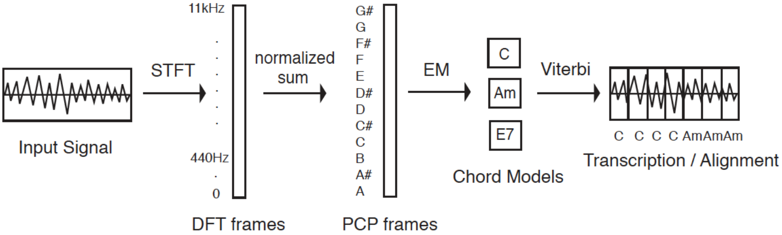
For some applications we want to focus on the harmonic content

- get rid of the instrumentation
- be invariant to transposition

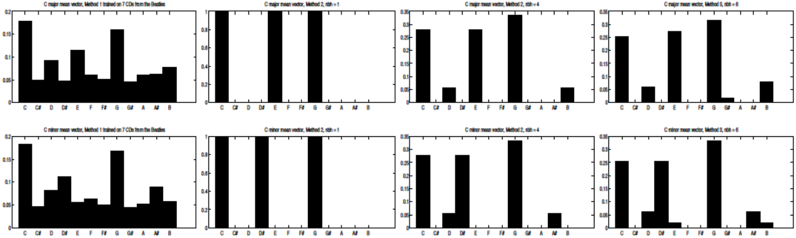
Chroma



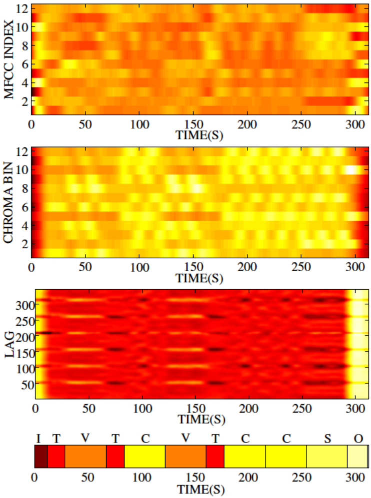
Chord detection pipeline



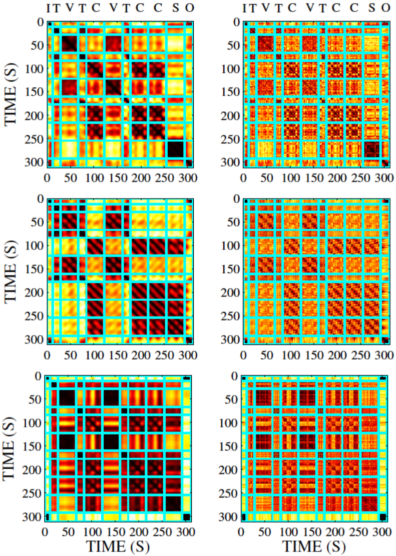
Chord examples



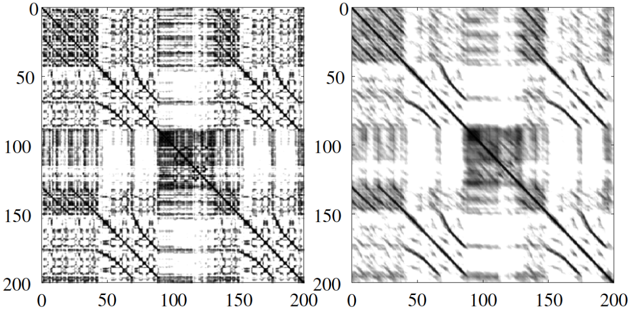
Musical Structure detection



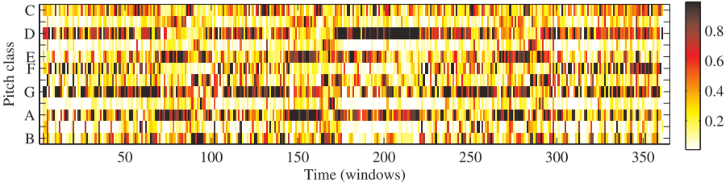
Musical Structure detection



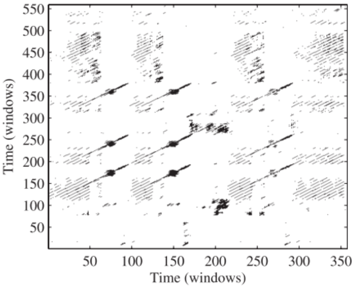
Musical Structure detection



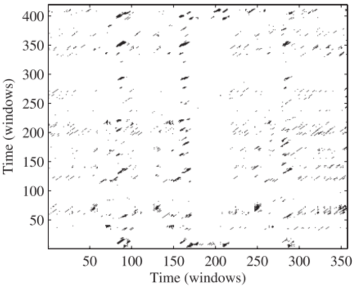
Cover


C
D
E
F
G
A
B
Pitch class
50
100
150
200
250
300
350
Time (windows)
0.8
0.6
0.4
0.2

Cover



Time (windows)
Time (windows)
550
500
450
400
350
300
250
200
150
100
50
50 100 150 200 250 300 350



Time (windows)
Time (windows)
400
350
300
250
200
150
100
50
50 100 150 200 250 300 350