Unsupervised Learning

Christophe Ambroise

2022-2023

Section 1

Introduction

Types of machine learning

Machine learning is usually divided into two main types.

Predictive or supervised learning

- learn a mapping from inputs $\boldsymbol{x}$ to outputs y,
- given a training: set of input-output pairs $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}$.
- output (or response) variable may quantitative or qualitative

Reinforcement learning,

- less commonly used.
- useful for learning how to act or behave when given occasional reward or punishment signals.

Unsupervised learning

- Discover "interesting structure" in the data;
- Sometimes called knowledge discovery, or data analysis.

Exploratory Data analysis (EDA)

Certain methods, mainly geometric, make it possible to highlight the relations which can exist between the various data and to establish a summary (sometimes with statistic characteristic).

In French, the terminology “analyse de données” refers to a subset of what is more commonly called multivariate statistics.

The purpose of the data analysis is exploratory and aims to

- vizualise or
- summarize.

Both aspects are related. This course presents some of the most common methods.

Unsupervised learning

```
When we are learning to see, nobody's telling us
what the right answers are -- we just look
(Geoffrey Hinton)
```

General principle

- Build a model : often related to density estimation (Learn $P(\boldsymbol{x}_i|\boldsymbol{\theta})$)

Unsupervised learning or Data analysis

Data

Data Analysis is a family of methods (sometimes statistical) allowing to explore data represented mostly in the form of

- array where each line describes an object (individual) and each column a variable:
- of similarity (or dissimilarity) pair between objects

Methods

- Clustering (discrete latent variable)
- Factor analysis (continuous latent variable)
- Graphical models (discovering graph structure among variables)

Section 2

A few introductive examples

Image compression

- Data: $(X_i)_i$ the gray levels of an image
- Assumption there are $(Z)_i$ hidden label of the K gray levels classes

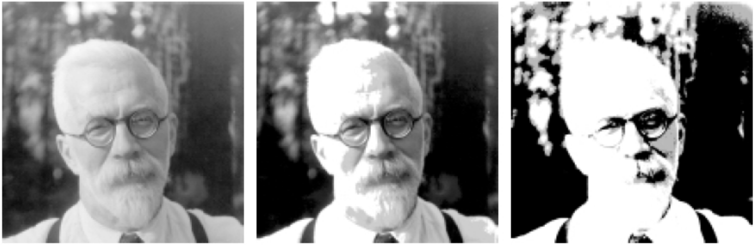


Figure 1: Famous Statistician R. Fisher

Clustering the nodes of a network I

Data

- V a set of given nodes $\in \{1, \dots, n\}$,
- E a set of edges $\in \{1, \dots, n\}^2$,
- $\boldsymbol{X} = (X_{ij})$ the adjacency matrix such that $\{X_{ij} = 1\} = \mathbb{I}\{i \leftrightarrow j\}$.
  - oriented : $X_{ij} \neq X_{ji}$,
  - valued : $X_{ij} \in \mathbb{R}$.

Assumption

hyp.: there exists a hidden structure into Q classes of connectivity,

- $\boldsymbol{Z} = (\boldsymbol{Z}_i)_i$, $Z_{iq} = \mathbb{I}\{i \in q\}$ are indep. hidden variables,
- $\boldsymbol{\alpha} = \{\alpha_q\}$, the *prior* proportions of groups,

French blog

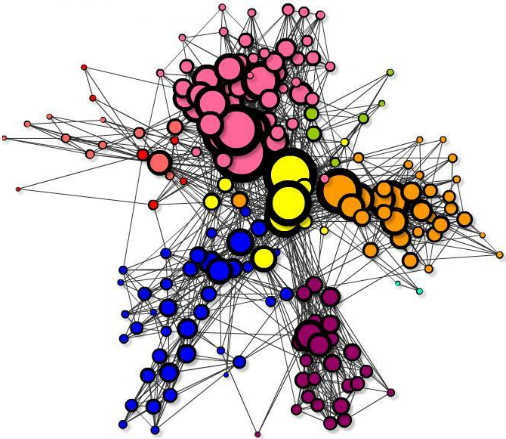


Figure 2: Sample of 250 blogs (nodes) with their links (edges) of the French political Blogosphere

US Blog

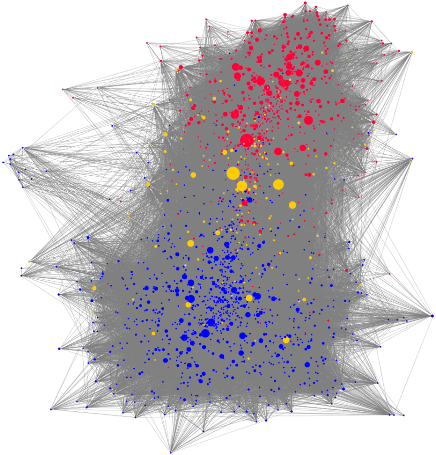


Figure 3: US political blog

Summarizing genetic information

In the Nature article ‘’Gene Mirror Geography Within Europe”, November et al., 2008,

Data

- 3000 individuals
- described by 500,000 SNP (Single Nucleotide Polymorphism)

can be represented by a data table $\boldsymbol{X} = (X_{ij})$ with

- 3000 rows (if 1 meter high)
- 50,0000 columns colonnes (then 166 meters large !!!)

SNP

- 90 % of human genetic variation,
- SNP with allelic frequency greater than 1 % are present every 300 base pairs in average (in human genome)

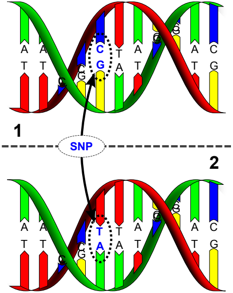


Figure 4: SNP (wikipedia)

Data (continued)

	Father	
Mother	C	T
C	0	1
T	1	2

- How to summarize this table
- Each person is a vector in a 500,000 dimensional space

Solution: Principal component Analysis

Summarize 500,000 columns by 2 columns (linear combination of the original)

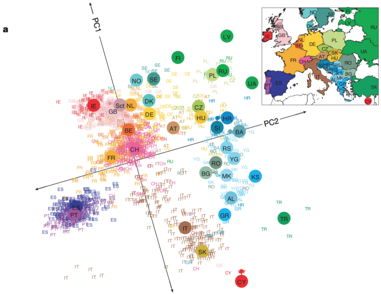


Figure 5: PCA of european population

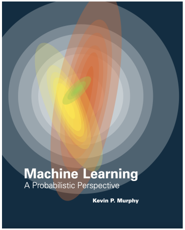
Section 3

Outline and evaluation of the course

Outline

1. Reminder about Bayesian Statistics
2. Discrete Latent Variabless
   - Hidden markov models (HMM)
   - Variational EM (VEM)
   - Stochastic block model (SBM)
3. Continuous Latent Variables
   - Factor Analysis (FA)
   - Independant Component Analysis (ICA)
   - Variational Auto-encoder (VAE)

Practical matters



Reference document

The lecture closely follows and largely borrows material from “Machine Learning: A Probabilistic Perspective” (MLAPP) from Kevin P. Murphy, chapters:

- Chapter 5 : Bayesian Statistics
- Chapter 17: Markov and hidden Markov models
- Chapter 21: Variational inference

Evaluation

1. R project (1/3)
2. Final Exam (2/3)

Section 4

Reminder about Bayesian Statistics

Snooker Frequentist example I

Let consider a snooker table, and a ball. The ball A is launched perpendicularly to a reference edge and stop at a distance l from this billiard edge.

A second ball B is thrown n times and we denote by X the number of times that B stops at a distance l' of the edge such that $l' > l$.

Problem

Estimate the proportion p of the number of times $l' > l$ knowing that $X = x$.

Assumption

$X \sim \mathcal{B}(n, p)$ (binomial distribution).
In this case we have a sample of size 1

Snooker Frequentist example II

Maximum Likelihood Estimation

The likelihood of parameter p writes:

$$\ell(p; x) = C_n^x p^x (1-p)^{n-x},$$

and by canceling the first derivative of the log-likelihood, we obtain

$$\hat{p}_{MV} = arg \max_p \ell(p; x) = \frac{x}{n}$$

Small sample problem

If many launches have been performed, the estimate of p will be satisfactory. On the other hand, if only one throw is observed, we find:

- $x = 0$ gives $\hat{p} = 0$;
- $x = 1$, which gives $\hat{p} = 1$.

In both cases the estimate appears intuitively of very poor quality.

Bayesian Statistics in a nutshell I

The maximum likelihood framework produces point estimates

Reverend T. Bayes (1701-1761)

Two years after the death of the Reverend T. Bayes (1701-1761), a friend of this one, published his *essay in view of solving the doctrine of chances* (Bayes 1763).

Parameters are random variables

In this little booklet, which is the source of the inference modern Bayesian statistic,

- Parameters are no longer treated as deterministic quantities but random as are observations.
- The dual role of the parameters θ and the observations x is described thanks to conditioning by Bayes' theorem:

Bayes Theorem

For a distribution (called *Prior*) π on the parameter θ, and a observation x of density $f(x|\theta)$, the distribution of θ conditionally on x

$$\pi(\theta|x) = \frac{f(x|\theta)\pi(\theta)}{\int f(x|\theta)\pi(\theta)d\theta}.$$

The main innovation of the Bayesian statistical model is the law π on the model parameters.

Prior, Posterior, Cost

Thus, in the context of a Bayesian statistical approach three functions must be specified:

- the law of the observations, the so-called likelihood $f(x|\theta)$;
- the prior distribution on the parameters, $\pi(\theta)$;
- the cost C associated with the decision δ for parameters θ.

Cost is a numerical measure of the quality of a decision.

Bayes estimator

We call the Bayes estimator associated with a prior distribution π and a cost C, any estimator δ^π which, given an observation vector x, minimizes the cost a posteriori

$$\rho(\pi, \delta | x) = E^\pi[C(\theta, \delta)|x] = \int_\theta C(\theta, \delta)\pi(\theta|x)d\theta.$$

Conjugate Prior

When the prior and the posterior have the same form, we say that the prior is a conjugate prior for the corresponding likelihood. Conjugate priors are widely used because they simplify computation, and are easy to interpret

The posterior $p(\theta|x)$ summarizes everything we know about the unknown quantities θ.

Conjugation	Likelihood	Prior	Posterior
Bernoulli	$\mathcal{B}(p)$	$Beta(\alpha, \beta)$	$Beta(\alpha + \sum_{i=1}^n x_i,\ \beta + n - \sum_{i=1}^n x_i)$
Binomial	$\mathcal{B}(p)$	$Beta(\alpha, \beta)$	$Beta(\alpha + \sum_{i=1}^n x_i,\ \beta + \sum_{i=1}^n N_i - \sum_{i=1}^n x_i)$
Multinomial	$\mathcal{M}ult(n, \boldsymbol{p})$	$\mathcal{D}ir(\boldsymbol{\alpha})$	$Dir(\boldsymbol{\alpha} + \sum_{i=1}^n \mathbf{x}_i)$

Posterior Mean, Median, Mode

We can easily compute a point estimate of an unknown quantity by computing the posterior mean, median or mode.

However, the posterior mode, aka the MAP estimate, is the most popular choice:

- it reduces to an optimization problem, for which efficient algorithms often exist.
- MAP estimation can be interpreted in non-Bayesian terms, by thinking of the log prior as a regularizer (see Lasso e.g.)

Example of Bayesian inference I

Let us take the snooker example again and look at the Bayesian approach

- the law on the observations is a binomial, $X \sim \mathcal{B}(n, p)$;
- the ball can equally probably stop ny distance from the edge. Hence $p \sim U_{[0,1]}$;
- let consider a quadratic cost: $C(p, \delta) = (p - \delta)^2$.

In that case,

$$
\begin{aligned}
\pi(p|X = x) &= \frac{C_n^x p^x (1-p)^{n-x} \mathbb{1}_{\{p \in [0,1]\}}}{\int_0^1 C_n^x p^x (1-p)^{n-x} dp} \\
&= \frac{p^x (1-p)^{n-x} \mathbb{1}_{\ p \ \ in[0,1]\}}}{\int_0^1 p^x (1-p)^{n-x} dp}.
\end{aligned}
$$

Example of Bayesian inference II

The posterior distribution is therefore a beta distribution, $\mathcal{B}e(x+1, n-x+1)$. It is easy to show that the Bayes estimator associated with a distribution π and a quadratic cost is the posterior mean

$$\delta^{\pi}(x) = E^{\pi}[p|x] = \int p \cdot \pi(p|x) dp.$$

The expectation of a random variable X following a beta distribution, $\mathcal{B}e(\alpha, \beta)$ is given by

$$E[X] = \frac{\alpha}{\alpha + \beta}.$$

The Bayes estimator associated with the quadratic cost is therefore written

$$\delta^{\pi}(x) = \frac{x+1}{n+2}.$$

Example of Bayesian inference III

If many launches have been performed, the estimate of p by this Bayesian procedure will be very close to the estimator of the maximum of likelihood. On the other hand, if only one throw is observed, we find:

- $x = 0$ gives $\hat{p} = \frac{1}{3}$;
- $x = 1$ gives $\hat{p} = \frac{2}{3}$.

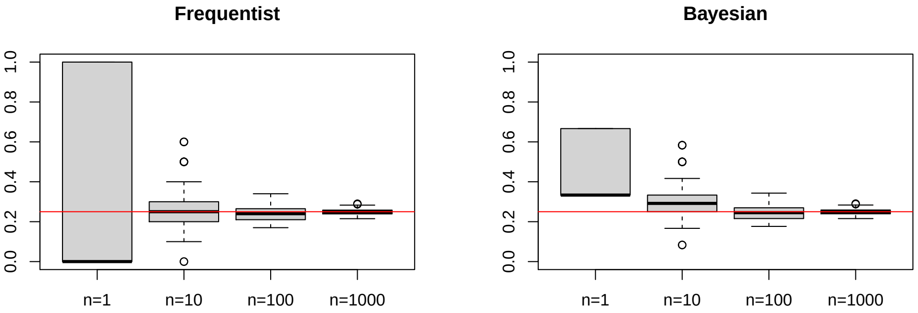
Both of these results seem reasonable.

Note that by taking a cost which is equal to 0 if the decision is correct and 1 otherwise (cost 0-1), the Bayes estimator is, in this case, the same as that obtained by the maximum likelihood method.

Note that the Bayes estimators are justified for a size of finite sample, unlike estimators of the maximum of likelihood which only have asymptotic properties.

R code

Comparing Bayesian and Frequentist estimator for $p = \frac{1}{4}$ and $n \in \{1, 10, 100, 1000\}$.



Bayesian model selection

How to choose the best model ?

A classical strategie is cross-validation. . . .

In a Bayesian framework we can compute the posterior overs models $m \in \mathcal{M}$

$$p(m|\mathcal{D}) = \frac{p(\mathcal{D}|m)p(m)}{\sum_{m \in \mathcal{M}} p(\mathcal{D}|m)p(m)}$$

with

$$p(\mathcal{D}|m) = \int p(\mathcal{D}|\boldsymbol{\theta}, m)p(\boldsymbol{\theta}|m)d\boldsymbol{\theta}$$

We can compute the MAP model $\hat{m} = arg \max_m p(m|\mathcal{D})$

If each model is considered without preference $p(m) \propto 1$, $p(\mathcal{D}|m)$ is the **model evidence** of m.

Bayes factor

The ratio of model evidences

$$\frac{p(\mathcal{D}|m_i)}{p(\mathcal{D}|m_j)}$$

for two models is known as a Bayes factor

Bayesian information criterion

Model evidence can be difficult to compute. A widespread approcahtion is known as the Bayesian information criterion

$$BIC \triangleq \log p(\mathcal{D}|\hat{\theta}) - \frac{df(m)}{2} \log n \approx p(\mathcal{D}|m)$$

where

- $df(m)$ is the number of degrees of freedom in the model m
- $\hat{\theta}$ is the MLE estimator

The approximation is obtained via Gaussian prior and Laplace approximation.

Section 5

Markov and hidden Markov models

Introduction

Models for sequences of observations, $X_1, ..., X_T$, of arbitrary length T.

Such models have applications in - computational biology, - natural language processing, - time series forecasting, etc.

Focus

- Focus on the case where we the observations occur at discrete “time steps”,
- “time” may also refer to locations within a sequence.

Markov models

Markov chain assumes that X_t captures all the relevant information for predicting the future (i.e., we assume it is a sufficient statistic).

If we assume discrete time steps, the joint distribution is a **Markov chain** :

$$p(X_{1:T}) = p(X_1)p(X_2|X_1)p(X_3|X_2)... = p(X1)\prod_{t=2}^{T} p(X_t|X_{t-1})$$

- X_t captures all relevant information for prediction the future
- X_t is a sufficient statistic

Homogeneous discrete Markov Chain

if $p(X_t|X_{t-1})$ is independant of time then the chain is called homogeneous, stationary or time invariant.

We assume in the follwing that the observed variables are discrete, so $X_t \in \{1, ..., K\}$, this is called a finite-state Markov chain.

Transition matrix

When X_t is discrete $X_t \in \{1, ..., K\}$, the conditional distribution $p(X_t|X_{t-1})$ can be written as a $K \times K$ matrix, known as the transition matrix A

where $A_{ij} = p(X_t = j|X_{t-1} = i)$ is the probability of going from state i to state j. $\forall j,\ \sum_i A_{ij} = 1$

Example 2 states Markov Chain

$$A = \begin{pmatrix} 1-\alpha & \alpha \\ \beta & 1-\beta \end{pmatrix}$$

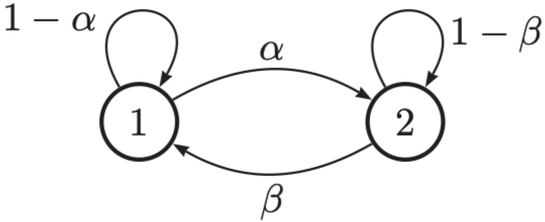


Figure 6: 2 states Markov chain

Example of DNA mutation Markov Chain

$$A = \left[\begin{array}{ccccc} A & C & G & T & \\ & & \vdots & & A \\ & & P(X_t=G \mid X_{t-1}=C) & \cdots & C \\ & & & & G \\ & & & & T \end{array}\right]$$

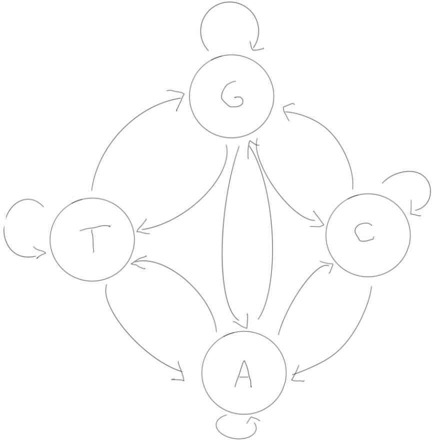


Figure 7: 4 states Markov chain

Exercice

Simulate a DNA sequence of length 100

The n-step transition matrix $A(n)$

$$A_{ij} \triangleq p(X_{t+n} = j | X_t = i)$$

which is the probability of getting from i to j in exactly n steps.

$A(1) = A$

Chapman-Kolmogorov equations

$$A_{ij}(m+n) = \sum_k A_{ik}(m) A_{kj}(n)$$

We can write the above as a matrix multiplication

$$A(m+n) = A(m)A(n)$$

And thus $A(n) = A^n$

Language modeling

One important application of Markov models is to make statistical language models, which are probability distributions over sequences of words.

We define the state space to be all the words of some language.

n-gram models

- The marginal probabilities $p(X_t = k)$ are called unigram statistics.
- using first-order Markov model $p(X_t = k | X_{t-1} = j)$ is called a bigram model.
- using second-order Markov model $p(X_t = k | X_{t-1} = j, X_{t-2} = i)$ is called a trigram model.
- ...

Exercice

- Estimate unigram and bigram models for the letters {a,...,z,-} from a set of Leonard Cohen songs

Language models

can be used for several things, such as the following:

- **Sentence completion** A language model can predict the next word given the previous words in a sentence. This can be used to reduce the amount of typing required, which is particularly important for disabled users or uses of mobile devices.
- **Data compression** Any density model can be used to define an encoding scheme, by assigning short codewords to more probable strings.
- **Text classification** Any density model can be used as a class-conditional density and hence turned into a (generative) classifier.
- **Automatic essay writing** One can sample from $p(x_{1:t})$ to generate artificial text.

MLE for Markov language models

The probability of any particular sequence of length T is given by

$$\begin{aligned} p(x_{1:T}|\theta) = &\quad \pi(x_1)A(x_1,x_2)...A(x_{T-1},x_T) \\ = &\quad \prod_j \pi_j^{\mathbb{I}_{x_1=j}} \prod_{t=2}^{T} \prod_{jk} (A_{jk})^{\mathbb{I}}_{(x_t=k,x_{t-1}=j)} \end{aligned}$$

Hence the log-likelihood of a set of N sequences $\mathcal{D} = (x_1, ..., x_N)$, where $xi = (x_{i,1}, ..., x_{i,Ti})$ is a sequence of length Ti, is given by

$$logp(\mathcal{D}|\theta) = \log p(x_i|\theta) = N_j^1 \log \pi_j + N_{jk} \log A_{jk}$$

MLE estimates are normalized Counts

$N_j^1 \triangleq \sum_{i=1}^{N} \mathbb{I}_{(x_{i1}=j)}$, $N_{jk} \triangleq \sum_{i=1^N} \sum_{t=1}^{Ti-1} \mathbb{I}_{(x_t=k,x_{t-1}=j)}$

$\hat{\pi}_j = \frac{N_j^1}{\sum_j N_j^1}$, $\hat{A}_{jk} = \frac{N_{jk}}{\sum_k N_{jk}}$

The problem of zero-counts

very acute whenever

- the number of states K,
- and/or the order of the chain, n, is large.

A simple solution

use add-one smoothing, where we simply add one to all the empirical counts before normalizing (Bayesian interpretation. . .)

- However add-one smoothing assumes all n-grams are equally likely, which is not very realistic.

Empirical Bayes version of deleted interpolation

Deleted interpolation (Chen and Goodman 1996)

Defines the transition matrix as a convex combination of the bigram requencies $f_{jk} = \frac{N_{jk}}{N_j}$ and the unigram frequencies $f_k = \frac{N_k}{N}$:

$$A_{jk} = (1 - \lambda) f_{jk} + \lambda f_k$$

The term λ is usually set by cross validation.

An equivalent simple hierarchical Bayesian model

Prior $A_j \sim Dir(\alpha_0 m_1, ..., \alpha_0 m_K) = Dir(\alpha_0 m) = Dir(\alpha)$

- A_j is row j of the transition matrix,
- m is the prior mean (satisfying $\sum_k m_k = 1$) and
- α_0 is the prior strength.

Posterior $A_j \sim Dir(\alpha + N_j)$ where $N_j = (N_{j1}, ..., NjK)$ is the vector that records the number of times we have transitioned out of state j to each of the other states.

Empirical Bayes version of deleted interpolation

the posterior predictive density is $p(X_{t+1} = k | X_t = j, \mathcal{D}) = \bar{A}_{jk} = \frac{N_{jk} + \alpha_0 m_k}{\sum_k Njk + \alpha_0} = (1 - \lambda_j) f_{jk} + \lambda_j m_k$ where $\bar{A}_{jk} = E[A_{jk} | \mathcal{D}, \alpha]$ and $\lambda_j = \frac{\alpha_j}{N_j + \alpha_0}$

Remark

We have to choose α_0 and m...

Stationary distribution of a Markov chain

We are often interested in the long term distribution over states, which is known as the stationary distribution of the chain

Distribution over states

let $\pi_t(j) = p(X_t = j)$

- Assume that π_t is a row vector.
- If we have an initial distribution over states of π_0, then at time 1 we have $\pi_1(j) = \sum_i \pi_0(i) A_{ij}$, which can be written as $\pi_1 = \pi_0 A$

Stationary distribution

If we ever reach a stage where

$$\pi = \pi A$$

then we say we have reached the stationary distribution (also called the invariant distribution or equilibrium distribution).

Existence of a Stationnary distribution

Let $(X_n)_n$ be a homogeneous Markov chain with finite state space and transition matrix A verifying: there exists an integer $k \in \mathbb{N}^*$ such that all the coefficients of A^k are strictly positive. Then, the law of X_n converges to the unique probability law π invariant by A at an exponential rate. More precisely, there exists $h > 0$ and $\rho \in [0, 1[$ such that for all $n \in \mathbb{N}^*$ and for all states x and y,

$$|A^n(y, x) - \pi(x)| \leq h\rho^n.$$

Computing the stationary distribution

Eigenvector associated with unit eigenvalue

To find the stationary distribution, we can just solve the eigenvector equation $A^T v = v$, and then to set $\pi = v^T$, where v is an eigenvector with eigenvalue 1.
Eigenvalues of A and A^T are the same.

A more general approach

We have K constraints from $\pi(I - A) = 0_{Kx1}$ and 1 constraint from $\pi\mathbb{1}_{Kx1} = 1$.
Since we only have K unknowns, this is overconstrained.
Let us replace any column (e.g., the last) of $I - A$ with 1, to get a new matrix, call it M. Next we define $r = [0, 0, ..., 1]$, then solve

$$\pi M = r$$

.

Exercise

Let

$$A^T = \begin{pmatrix} 0 & 1/2 & 1 \\ 1 & 0 & 0 \\ 0 & 1/2 & 0 \end{pmatrix}$$

be the transpose of a transition matrix. Compute the stationary distribution.

Exercise

Solution 1

```r
A <- matrix(c(0, 1/2,1,1,0,0,0,1/2,0),3,3)
eigen.res<-eigen(t(A))
pi<-eigen.res$vectors[,1]
pi<-t(pi)%*%A
pi/sum(pi)
```

```
##         [,1]   [,2]   [,3]
## [1,] 0.4+0i 0.4+0i 0.2+0i
```

Solution 2

```r
M<-diag(rep(1,3))-A
M[,3]<-rep(1,3)
solve(t(M),b=c(0,0,1))
```

```
## [1] 0.4 0.4 0.2
```

When does a stationary distribution exist?

Irreducible chain

A necessary condition to have a unique stationary distribution is that the state transition diagram be a singly connected component, i.e., we can get from any state to any other state. Such chains are called irreducible.

Aperiodic chain

Define the period of state i: $d(i) = gcd\{t : A_{ii}(t) > 0\}$
We say a state i is aperiodic if $d(i) = 1$. (A sufficient condition to ensure this is if state i has a self-loop, but this is not a necessary condition.) A chain is aperiodic if all its states are aperiodic.

Irreducible, aperiodic finite state Markov chain

Every irreducible (singly connected), aperiodic finite state Markov chain has a limiting distribution, which is equal to π, its unique stationary distribution. Every regular finite state chain has a unique stationary distribution, where a regular chain is one whose transition matrix satisfies $A^n_{ij} > 0$ for n and all i,j. Consequently, after n steps, the chain could be in any state, no matter

Application: Google's PageRank algorithm for web page ranking

The web is a giant directed graph, where nodes represent web pages (documents) and edges re We can get a refined search by storing the location of each word in each document.

Basic Ranking

For each word, we store a list of the documents where this word occurs.

Taking the web structure into account

But the link structure of the web provides an additional source of information.

Each incoming link is weighted by the source's authority. Thus we get the following recursive definition for the authoritativeness of page j, also called its **PageRank**:

$$\pi_j = \sum_i \pi_i A_{ij}$$

where A_{ij} is the probability of following a link from i to j. We recognize a stationary distribution of a Markov chain.

Mixture models and the EM algorithm

Latent variable models

Assume that the observed variables are correlated because they arise from a hidden common “cause”. Model with hidden variables are also known as latent variable models or LVMs.

Latent variables

In general there are L latent variables, $z_{i1}, ..., z_{iL}$, and D visible variables, $x_{i1}, ..., x_{iD}$, where usually $D >> L$. If we have $L > 1$, there are many latent factors contributing to each observation, so we have a many-to-many mapping. If $L = 1$, we we only have a single latent variable; in this case, z_i is usually discrete, and we have a one-to-many mapping.

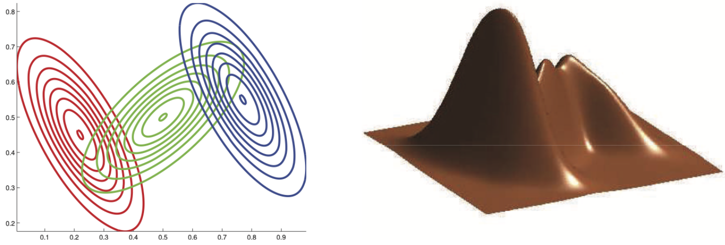
Mixture models

The simplest form of LVM is when $z_i \in \{1, ..., K\}$, representing a discrete latent state:

- $p(z_i) = Cat(\pi)$ (proportions)
- $p(x_i|z_i = k) = p_k(x_i)$ (class densities, components),

$$p(x_i|\theta) = \sum_k \pi_k p_k(x_i)$$

π_k satisfy $0 \leq \pi_k \leq 1$ and $\sum_k \pi_k = 1$.



Mixtures of Gaussians

In this model, each base distribution in the mixture is a multivariate Gaussian with mean $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$:

$$p(x_i|\theta) = \sum_k \pi_k \mathcal{N}(x_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Mixture of multinoullis

class- conditional density is a product of Bernoullis:

$$p(x_i|z_i = k, \theta) = \prod_j Ber(x_{ij}|\mu_{jk}).$$

EM algorithm

Data

- Observed data **:** $x_{1:n}$
- Missing (or hidden) data : **:** $z_{1:n}$

Principle

- Starting from θ^0
- At step q
  - E(xpectation) step: $Q(\theta, \theta^q) = E_{Z_{1:n}|\boldsymbol{x}_{1:n}}[\log P(\boldsymbol{x}_{1:n}, \boldsymbol{z}_{1:n}, \theta)]$
  - M(aximisation) step: $\theta^{q+1} = argmax_\theta Q(\theta, \theta^q)$

EM algorithm

At each iteration the log-likelihood of the parameters increase

$$
\begin{aligned}
Q(\theta^{q+1}, \theta^q) &\geq Q(\theta^q, \theta^q) \\
0 &\leq Q(\theta^{q+1}, \theta^q) - Q(\theta^q, \theta^q) \\
0 &\leq E_{Z_{1:n}|\boldsymbol{x}_{1:n}}[\log \frac{P(\boldsymbol{x}_{1:n}, \boldsymbol{z}_{1:n}, \theta^{q+1})}{P(\boldsymbol{x}_{1:n}, \boldsymbol{z}_{1:n}, \theta^q)}] \\
E_{Z_{1:n}|\boldsymbol{x}_{1:n}}[\log \frac{P(\boldsymbol{x}_{1:n}, \boldsymbol{z}_{1:n}, \theta^{q+1})}{P(\boldsymbol{x}_{1:n}, \boldsymbol{z}_{1:n}, \theta^q)}] &\underset{Jensen}{\leq} \log E_{Z_{1:n}|\boldsymbol{x}_{1:n}}[\frac{P(\boldsymbol{x}_{1:n}, \boldsymbol{z}_{1:n}, \theta^{q+1})}{P(\boldsymbol{x}_{1:n}, \boldsymbol{z}_{1:n}, \theta^q)}] \\
0 &\leq \log \int \frac{P(\boldsymbol{x}_{1:n}, \boldsymbol{z}_{1:n}, \theta^{q+1})}{P(\boldsymbol{x}_{1:n}, \boldsymbol{z}_{1:n}, \theta^q)} P(\boldsymbol{z}_{1:n}|\boldsymbol{x}_{1:n}, \\
0 &\leq \log \frac{P(\boldsymbol{x}_{1:n}, \theta^{q+1})}{P(\boldsymbol{x}_{1:n}, \theta^q)}
\end{aligned}
$$

Exercice

Programmer un algorithme EM pour les mélange de Poisson univariés

Hidden Markov models

Model

1. a discrete-time, discrete-state Markov chain, with hidden states $z_t \in \{1, ..., K\}$
2. plus an observation model (emission) $p(\boldsymbol{x}_t|z_t)$

Joint distribution

$$p(z_{1:T}, x_{1:T}) = p(z_{1:T})p(x_{1:T}|z_{1:T}) = p(z_1)\prod_{t=2}^{T} p(z_t|z_{t-1})\prod_{t_1}^{T} p(x_t|z_t)$$

discrete or continuous observations in HMM

Discrete

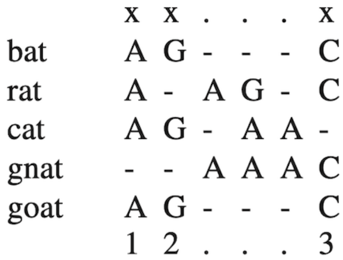
If observations are discrete, it is common for the observation model to be an observation matrix:
$p(x_t = l|z_t = k, \theta) = B(k, l)$

Continuous

If the observations are continuous, it is common for the observation model to be a conditional Gaussian:
$p(\boldsymbol{x}_t|z_t = k, \theta) = \mathcal{N}(\boldsymbol{x}_t|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$

Applications

- Automatic speech recognition
- Activity recognition
- Gene finding
- Protein sequence alignment. x_t represents an amino acid, and z_t represents whether this matches the latent consensus sequence at this location. This model is called a profile HMM. The HMM has 3 states, called match, insert and delete.



	x	x	.	.	.	x
bat	A	G	-	-	-	C
rat	A	-	A	G	-	C
cat	A	G	-	A	A	-
gnat	-	-	A	A	A	C
goat	A	G	-	-	-	C
	1	2	.	.	.	3

Figure 9: DNA sequence alignment

Types of Inference in HMMs

Filtering

compute the belief state $p(z_t|x_{1:t})$ online, or recursively, as the data streams in.

Prediction

compute $p(z_{t+h}|x_{1:t})$, where $h > 0$ is called the prediction horizon.

Smoothing

compute $p(z_t|x_{1:T})$ offline, given all the evidence.

MAP estimation

computing $\arg\max_{z_{1:T}} p(z_{1:T}|x_{1:T})$, which is a most probable state sequence (**Viterbi decoding**)

EM algorithm for HMMs I

Complete log-likelihood:

$$\begin{aligned}\log P(x_{1:T}, z_{1:T}; \theta) &= \sum_k \mathbb{I}_{(z_1=k)} \log \pi_k \\ &+ \sum_{t=2}^{T} \sum_{j,k} \mathbb{I}_{(z_{t-1}=j; z_t=k)} \log A_{jk} \\ &+ \sum_{t=1}^{T} \sum_{k} \mathbb{I}_{(z_t=k)} \log \Psi_t(k)\end{aligned}$$

where $\Psi_t(k) = p(x_t | z_t = k)$

EM algorithm for HMMs II

Expectation

$$\begin{aligned} Q(\theta, \theta q) &= \sum_k P_{\theta_q}(z_1 = k|x_{1:T}) \log \pi_k \\ &+ \sum_{t=2}^{T} \sum_{j,k} P_{\theta_q}(z_{t-1} = j; z_t = k|x_{1:T}) \log A_{jk} \\ &+ \sum_{t=1}^{T} \sum_{k} P_{\theta_q}(z_t = k|x_{1:T}) \log \Psi_t(k) \end{aligned}$$

Principle

- **E-step**. Compute $P_{\theta_q}(z_t = k|x_{1:T})$ and $P_{\theta_q}(z_{t-1} = j; z_t = k|x_{1:T})$,
  - forward
  - backward equations
- **M-step**. $\theta^{q+1} = arg \max_\theta Q(\theta, \theta^q)$

The forwards-backwards algorithm

Computing $P_{\theta_q}(z_t = k|x_{1:T})$ can be achieved via the forward-backward algorithm.

The key decomposition relies on the fact that we can break the chain into two parts, the past and the future, by conditioning on z_t:

$$p(z_t = j|x_{1:T}) \propto p(z_t = j, x_{t+1:T}|x_{1:t}) \propto p(z_t = j|x_{1:t})p(x_{t+1:T}|z_t = j, x_{1:t})$$

The forward algorithm

Compute the filtered marginals, $p(z_t|x_{1:t})$ in an HMM

$$\begin{aligned}\alpha_t(j) &\triangleq p(z_t = j|x_{1:t}) = p(z_t = j|x_t, x_{1:t-1}) \\ &= \frac{p(x_t|z_t = j, x_{1:t-1})p(z_t = j|x_{1:t-1})}{p(x_t|x_{1:t-1})}\end{aligned}$$

where

$$p(z_t = j|x_{1:t-1}) = \sum_l p(z_t = j, z_{t-1} = l|x_{1:t-1}) = \sum_l p(z_t = j|z_{t-1} = l)p(z_{t-}$$

$$p(x_t|x_{1:t-1}) = \sum_k p(x_t, z_t = k|x_{1:t-1}) = \sum_k p(z_t = k|x_{1:t-1})p(x_t|z_t = k)$$

The distribution $p(z_t|x_{1:t})$ is called the (filtered) belief state at time t, and is a vector of K numbers $\boldsymbol{\alpha}_t$ where

$$\boldsymbol{\alpha}_t \propto \psi_t \odot (A^T \boldsymbol{\alpha}_{t-1})$$

The backward algorithm

Compute the conditional likelihood of future evidence given that the hidden state at time t is j.

$$\beta_t(j) \triangleq p(x_{t+1:T}|z_t = j)$$

$$\begin{array}{rcl}
\beta_{t-1}(i) & = & p(x_{t:T}|z_{t-1} = i) \\
& = & \sum_j p(z_t = j, x_t, x_{t+1:T}|z_{t-1} = i) \\
& = & \sum_j p(x_{t+1:T}|z_t = j, x_t, z_{t-1} = i)p(z_t = j, x_t|z_{t-1} = i) \\
& = & \sum_j p(x_{t+1:T}|z_t = j, x_t, z_{t-1} = i)p(z_t = j|z_{t-1} = i)P(x_t|z_{t-1} =
\end{array}$$

$$\boldsymbol{\beta}_{t-1} = \boldsymbol{A}(\psi_t \odot \boldsymbol{\beta}_t)$$

The smoothed posterior marginal

$$P(z_t = j | x_{1:T}) \propto p(z_t = j | x_{1:t}) p(x_{t+1:T} | z_t = j)$$

$$\gamma_t(j) \propto \alpha_t(j) \beta_t(j)$$

Forward and Backward initialisation

Forward

$$\boldsymbol{\alpha}_1 \propto \boldsymbol{\pi} \odot \psi_1$$

with $\sum_k \alpha_1(k) = 1$

Backward

$$\beta_T(i) = p(x_{T+1:T}|z_T = i) = p(\emptyset|z_T = i) = 1$$

which is the probability of a non-event

Two-slice marginal

$$\begin{aligned}\zeta_{t,t+1}(i,j) &\triangleq p(z_t = i, z_{t+1} = j | x_{1:T}) \\ &\propto \alpha_t(i)\psi_{t+1}(j)\beta_{t+1}(j)A_{ij}\end{aligned}$$

MAP versus MPM

The (jointly) most probable sequence of states is not necessarily the same as the sequence of (marginally) most probable states.

MPM (Maximizer of the Posterior Marginals)

$$\tilde{z} = (\arg\max_{z_1} p(z_1|x_{1:T}), ..., \arg\max_{Z_T} p(z_T|x_{1:T}))$$

MAP (Maximum A Posteriori)

$$z^\star = \arg\max_{z_{1:T}} p(z_{1:T}|x_{1:T})$$

MAP computation via Viterbi Algorithm I

The most probable hidden path given the data can be computed by a forward- backward recursion.

Remark

We consider the maximisation of the joint likelihood since

$$\arg\max_{z_{1:T}} p(z_{1:T}|x_{1:T}) = \arg\max_{z_{1:T}} p(z_{1:T}, x_{1:T})$$

Principle

Dynamic programming (Bellma 1953):

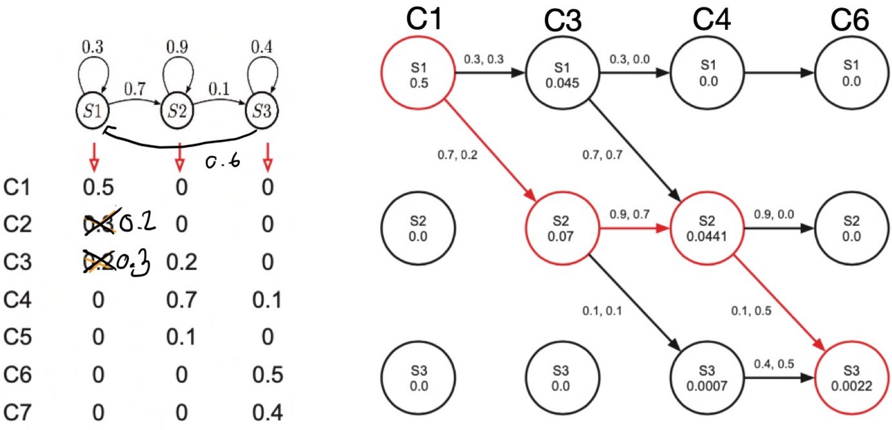
- Problem with optimal substructure
- Each subproblem is used to find the optimal solution of the main problem
- Example: shortest path, sequence alignment ...

MAP computation via Viterbi Algorithm II

3 steps

- splitting into subproblems
- solving subproblems
- reconstitution of the optimal solution

Viterbi Algorithm example



C1	0.5	0	0
C2	~~0.3~~ 0.2	0	0
C3	~~0.2~~ 0.3	0.2	0
C4	0	0.7	0.1
C5	0	0.1	0
C6	0	0	0.5
C7	0	0	0.4

Figure 10: Viterbi

Viterbi Algorithm

Forward

$$V_{1k} = \pi_k \psi_1(k), \forall k$$

For $t \geq 2$ (optimal choices for the hidden states)

$$\left\{ \begin{array}{cl} V_{tl} & = \max_k V_{t-1,k} A_{kl} \psi_t(l) \\ S_{t-1}(l) & = \arg\max_k V_{t-1,k} A_{kl} \psi_t(l) \end{array} \right.$$

Backward

$$z_T^\star = \arg\max_k V_{Tk}$$

For $t \leq T$ (backtracking)

$$z_t^\star = \arg\max_k S_t(z_{t+1}^*)$$

Section 6

Variational inference illustrated with Stochastic Block Model

Variational inference

- more general class of deterministic approximate inference algorithms
- based on variational inference (Jordan et al. 1998; Jaakkola and Jordan 2000; Jaakkola 2001; Wainwright and Jordan 2008a).

The basic idea

- pick an approximation $q(\boldsymbol{x})$ to the distribution from some tractable family,
- try to make this approximation as close as possible to the true posterior, $p^*(\boldsymbol{x}) \triangleq p(\boldsymbol{x}|D)$.
- reduces inference to an optimization problem.
- trade accuracy for speed

Stochastic block model example

A generative model

$$(z_i) \sim \mathcal{M}(1, \boldsymbol{\pi} = (\pi_1, ..., \pi_K))$$

$$x_{ij} | z_i = k, z_j = \ell \sim \mathcal{B}(\gamma_{k\ell})$$

Example : affiliation model

$$\gamma_{kl} = \begin{cases} \alpha, & \text{if } k = l \\ \beta, & \text{otherwise} \end{cases}$$

SBM

$$p(x_{ij} = 1 | z_i = k, z_j = \ell) = \gamma_{k\ell}^{x_{ij}} (1 - \gamma_{k\ell})^{(1 - x_{ij})}$$

$$\log p(\boldsymbol{x}, \boldsymbol{z}) = \log p(\boldsymbol{z})$$

Section 7

Appendix

Dirichlet distribution $Dir(\boldsymbol{\alpha})$ I

- Peter Gustav Lejeune Dirichlet (13 février 1805 , Düren – · 5 mai 1859 , Göttingen)
- continuous multivariate probability distribution parameterized by a vector $\boldsymbol{\alpha}$ of positive reals.
- used as prior distributions in Bayesian statistics,
- conjugate prior of the categorical distribution and multinomial distribution.

The Dirichlet distribution of order $K \geq 2$ with parameters $\alpha_1, ..., \alpha_K > 0$ has a probability density function

$$f\left(x_1, \ldots, x_K; \alpha_1, \ldots, \alpha_K\right) = \frac{1}{\mathrm{B}(\boldsymbol{\alpha})} \prod_{i=1}^{K} x_i^{\alpha_i - 1}$$

where the normalizing constant is the multivariate beta function.

Dirichlet distribution $Dir(\alpha)$ II

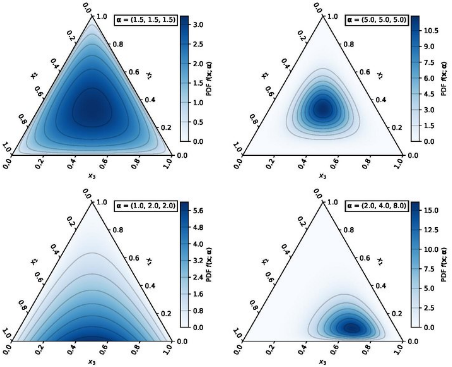


Figure 11: Dirichlet distributions

Multivariate beta function

$$\mathcal{B}\rceil(\boldsymbol{\alpha}) = \frac{\prod_i \Gamma(\alpha_i)}{\Gamma(\sum_i \alpha_i)}$$

Moments

Denoting $\alpha_0 = \sum_{i=1}^{K} \alpha_i$, we have

$$E[X_i] = \frac{\alpha_i}{\alpha_0},$$

$$Var[X_i] = \frac{\alpha_i(\alpha_0 - \alpha_i)}{\alpha_0^2(\alpha_0 + 1)}.$$

$$Cov[X_i, X_j] = \frac{-\alpha_i \alpha_j}{\alpha_0^2(\alpha_0 + 1)}.$$