Monte-Carlo and Empirical Methods for Statistical Inference, FMS091/MAST11

L12 — EM-algorithm

Hypotheses testing

The basis of a hypothesis test consists of

- A null hypothesis that we wish to test.
- A test statistic \( t(y) \), i.e. a function of the data.
- A rejection or critical region \( R \).

If the test statistic falls in the rejection region, then we reject the null hypothesis. We divide hypotheses into

- Simple  Specifies a single distribution for the data, e.g. \( y \sim N(\theta, \sigma^2) \) with \( H_0 : \theta = 0 \) and \( \sigma^2 \) known.
- Composite  Specifies more than one distribution for the data, e.g. \( y \sim N(\theta, \sigma^2) \) with \( H_0 : \theta = 0 \) and \( \sigma^2 \) unknown; or \( y \sim N(\theta, \sigma^2) \) with \( H_0 : \theta \leq 0 \) and \( \sigma^2 \) known.

Monte Carlo test of a simple hypothesis

Algorithm

1. Draw \( N \) samples, \( y^{(1)}, \ldots, y^{(N)} \), from the distribution specified by \( H_0 \).
2. Calculate the test statistic, \( t^{(i)} = t(y^{(i)}) \) for each sample.
3. Estimate the p-value using Monte Carlo integration as

\[
\hat{p}(y) = \frac{1}{N} \sum_{i=1}^{N} 1 \left( t^{(i)} \geq t(y) \right).
\]

4. If \( \hat{p}(y) \leq \alpha \) reject \( H_0 \).
Testing composite hypothesis

For a composite hypothesis $H_0$ defines a set of distributions. Thus, we can’t simply simulate from $H_0$. Three alternatives exist:

- **Pivotal tests**: Find a **pivotal** test statistic such that $t(Y)$ has the same distribution for all distributions in $H_0$. (Ex: Rank based tests.)
- **Conditional tests**: Convert the composite hypothesis to a simple by conditioning on a **sufficient statistic**. (Ex: Permutation based test.)
- **Bootstrap tests**: Replacing the composite hypothesis by a simple based on an estimate from data.

Bootstrap tests

- Remember the relation between hypothesis testing and confidence intervals.
- Knowing how to calculate Bootstrap confidence intervals, we directly have a tool for doing tests on the form $H_0 : \theta = \theta_0$ or $H_0 : \theta > \theta_0$.
- In general, the idea in a Bootstrap test is to estimate $P_0$ from data with $\hat{P}$.
- Given $\hat{P}$ we revert to a simple Monte Carlo test, simulating from $\hat{P}$.
- Note that this is similar to ordinary Bootstrap with the exception that we require $\hat{P}$ to fulfill the null-hypothesis.
- As an example we compare:
  - $y \sim N(\theta, \sigma^2)$ with $H_0 : \theta = 0$ and $\sigma^2$ unknown.

The EM-algorithm

- In some cases we might have incomplete observations.
  - Censored data
  - Missing data
  - Latent variable models
- Difficult to handle using ordinary estimation techniques.
- Bayesian statistics, with the model estimated using MCMC provides one alternative.
- Another alternative is the EM-algorithm (Dempster, Laird & Rubin; 1977).
- “proposed many times in special circumstances”.

Basic setup:

- We have observed some data $y$.
- Additional data $z$ is “missing”.
- The estimation problem would be “easy” if $z$ was known.

In principle we could write out the likelihood for the observed data as

$$L(\theta; y) = \int f(y, z|\theta) dz.$$ 

However the integral over the unknown data is often hard to compute. Instead we want to construct an approximation of the log-likelihood based on the observed data.
The EM-algorithm

1. Write out the log-likelihood assuming that all the data is known, \( \log L(\theta; y, z) \).
2. Compute the expected value of the log-likelihood over the unknown observations \( \mathbb{E}_z(\log L(\theta; y, z) | y, \theta_{\text{guess}}) \).
3. Take \( Q(\theta, \theta_{\text{guess}}) = \mathbb{E}_z(\log L(\theta; y, z) | y, \theta_{\text{guess}}) \). \( Q \) can now be seen as the average possible value of the log-likelihood given known observations and guessed parameters.
4. Update our guess of the parameters by maximising \( Q(\theta, \theta_{\text{guess}}) \).
5. Repeat from 2.

The result is the Expectation-Maximization algorithm.

The EM-algorithm

Algorithm:
Choose a starting value \( \theta^{(0)} \) and repeat for \( i = 1, 2, \ldots \) until convergence.

E-step
Compute the expectation of the log-likelihood with respect to the unknown data, conditional on the parameter guess and known data.

\[
Q(\theta, \theta^{(i-1)}) = \mathbb{E}_z \left( \log L(\theta; y, z) | y, \theta^{(i-1)} \right).
\]

M-step
Compute the maximum with respect to the unknown parameter

\[
\theta^{(i)} = \arg\max_\theta Q(\theta, \theta^{(i-1)}).
\]

Examples — Censored data

- The lifetime for a type of light bulbs is assumed to follow an exponential distribution with expectation \( \theta \).
- To estimate \( \theta \), \( n \) independent light bulbs are observed and their lifetimes \( y_i \) are recorded.
- The experiment is run for a pre-determined duration, \( h \).
- At the end of the experiment \( m \) light bulbs are still working.
- This gives \( n - m \) observations of lifetimes, \( y_i \), and \( m \) unknown lifetimes \( z_i \) that are larger than \( h \).

Derive an EM-algorithm to estimate \( \theta \).
Censored data (cont.)

1. Write down the log-likelihood as if all observations where known.
2. Separate the log-likelihood into parts with known, \( y \), and unknown, \( z \), data.
3. Take the expectation of the unknown parts, with respect to the unknown data \( z \), conditional on a guess, \( \theta_{\text{old}} \), of the parameters, and the known data \( y \).
4. Maximise the resulting expression with respect to \( \theta \) to obtain a new guess \( \theta_{\text{new}} \).

Censored v.s. Truncated data

Censored data Data above/below a certain value has not been observed; the number of exceedances have been registered.

Truncated data Data above/below a certain value has not been observed; the number of exceedances have not been registered.

- Both can be handled using the EM-algorithm but truncated data is harder.
- For censored data we had a term
  \[ E_z (\sum_{i=1}^{m} z_i | y, \theta_{\text{guess}}). \]
- For truncated data the corresponding term is
  \[ E_{z,M} (\sum_{i=1}^{M} z_i | y, \theta_{\text{guess}}). \]

(Gaussian) Mixture Models

- Assume that we have observations from one of several (Normal) distributions called classes.
- The probability of data coming from class \( k \) is \( \pi_k \).
- The distribution of each class is \( [y|\text{from class } k] \in \mathcal{N}(\mu_k, \Sigma_k) \).
- This generates a Gaussian mixture model with density
  \[ p(y|\pi, \mu, \Sigma) = \sum_{k=1}^{K} \pi_k p(y|\text{from class } k, \mu_k, \Sigma_k). \]
- Possible usages
  - Modeling heavy tailed distributions.
  - Classification/clustering of data.

(Gaussian) Mixture Models (cont.)

- If we knew the class belonging, \( z_i \), of each observation \( y_i \) the problem would be trivial.
- Thus the problem consists of two parts:
  1. Determine the class belongings \( z \).
  2. Estimating the parameters \( \Psi = \{\pi, \mu, \Sigma\} \).
- Augmenting the model with the unknown class belongings \( z_i \) the log-likelihood becomes
  \[
  \log L(\Psi; y, z) = \log \prod_{i=1}^{n} \pi_{z_i} p(y_i|z_i, \mu_k, \Sigma_k) \\
  = \sum_{i=1}^{n} \sum_{k=1}^{K} 1(z_i = k) \log \left( \pi_k p(y_i|z_i = k, \mu_k, \Sigma_k) \right). 
  \]
(Gaussian) Mixture Models — E-step

- The expectations of the log-likelihood is

\[ E_z \left( \log L(\psi; y, z) | y, \psi_{\text{guess}} \right) \]

\[ = \sum_{i=1}^{n} \sum_{k=1}^{K} w_{ik} \log(\pi_k p(y_i | z_i = k, \mu_k, \Sigma_k)) \]

- We note that

\[ E_z(1(z_i = k) | y, \psi_{\text{guess}}) = P(z_i = k | y, \psi_{\text{guess}}). \]

Gaussian Mixture Models — M-step

Having calculated the expectation of the log-likelihood we have that

\[ Q(\psi, \psi_{\text{guess}}) = E_z \left( \log L(\psi; y, z) | y, \psi_{\text{guess}} \right) \]

\[ = \sum_{i=1}^{n} \sum_{k=1}^{K} w_{ik} \left( \log(\pi_k) + \log \left( p(y_i | z_i = k, \mu_k, \Sigma_k) \right) \right) \]

\[ = \sum_{i=1}^{n} \sum_{k=1}^{K} w_{ik} \left( \log(\pi_k) - \frac{1}{2} \log |\Sigma_k| - \frac{d}{2} \log(2\pi) - \frac{(y_i - \mu_k)^\top \Sigma_k^{-1} (y_i - \mu_k)}{2} \right). \]

(Gaussian) Mixture Models — E-step

The probability of belonging to class \( k \) is

\[ P(z_i = k | y, \psi_{\text{guess}}) = \frac{p(z_i = k, y | \psi_{\text{guess}})}{p(y | \psi_{\text{guess}})} \]

\[ = \frac{p(y | z_i = k, \psi_{\text{guess}}) p(z_i = k | \psi_{\text{guess}})}{\sum_k p(z_i = k, y | \psi_{\text{guess}})} \]

\[ = \frac{p(y | z_i = k, \psi_{\text{guess}}) \pi_k}{\sum_k p(y | z_i = k, \psi_{\text{guess}}) \pi_k} = w_{ik} \]

Gaussian Mixture Models — M-step

The new estimates of \( \{\pi, \mu, \Sigma\} \) are obtained by maximizing the \( Q \)-function.

\[ \pi_k = \frac{1}{n} \sum_{i=1}^{n} w_{ik} \]

For Gaussian mixture densities, \( y_i | z_i = k \in N(\mu_k, \Sigma_k) \), the new estimates of \( \mu_k \) and \( \Sigma_k \) are given by

\[ \mu_k = \frac{1}{n\pi_k} \sum_{i=1}^{n} w_{ik} y_i \]

and

\[ \Sigma_k = \frac{1}{n\pi_k} \sum_{i=1}^{n} w_{ik} (y_i - \mu_k)^\top (y_i - \mu_k). \]
Gaussian Mixture Models — Old Faithful

Hidden Markov models (HMM) can be considered a generalization of mixture models where the hidden variables are related through a Markov process rather than independent of each other.

Are used in many different applied subjects, we look at two examples in signal processing and genetics.

The model is estimated using the EM algorithm.

The E-step can not be calculated analytically so we use Gibbs sampling to estimate it.

The resulting algorithm is sometimes called the MCEM algorithm.

We use Bootstrap tests to test some model properties.

This project thus combines Bootstrap, MC-integration, and the EM-algorithm and is perhaps the first example of a truly computationally intensive problem so far in the course.

Home Assignment 3 — Oral exam

The oral exam will be in groups of 2 to 4.

Should take around 30 minutes.

General discussion about the course in general and projects, what you did, how and why.

People from the same project groups in different oral exam groups.

Grading is based primarily on the projects.

Possible times are Friday in the exam week, week 1, and week 2 of the next study period.