T-61.5140 Project Work, S2016

Teacher: Pekka Marttinen
Project Assistant: Antti Kangasrääsiö

Project Deadline 24.4.2016

1 Introduction

This short introduction is a reminder of various concepts relevant to this project work, and is not intended to act as independent study material as such. Please see the course lectures and book for a more detailed account on the subject in case you are not able to easily follow this introduction.

1.1 Linear Regression

In general a linear regression model for observations y_t and covariates $\boldsymbol{x}_t$ is formulated as

$$y_t = \boldsymbol{\phi}\boldsymbol{x}_t + \varepsilon_t, \tag{1}$$

where ε_t is a random error. If ε_t is drawn from a normal distribution with mean zero and variance σ^2, y_t ends up being distributed according to

$$y_t \sim Normal(\boldsymbol{\phi}\boldsymbol{x}_t, \sigma^2). \tag{2}$$

A conjugate prior for this kind of linear regression is

$$\boldsymbol{\phi} \sim Normal(\boldsymbol{\mu}_\phi, \sigma^2\boldsymbol{\Sigma}_\phi), \tag{3}$$
$$\sigma^2 \sim InverseGamma(\alpha_{\sigma^2}, \beta_{\sigma^2}). \tag{4}$$

If the variables $\boldsymbol{\phi}$ are a priori uncorrelated, $\boldsymbol{\Sigma}_\phi$ is diagonal.

1.2 Mixture Models

Mixture models assume that each observation has been generated from one component model. For each component i we have a weight w_i. Weights from all components sum up to one. In the case of a mixture model with two components that correspond to two different linear models, this gives us the observation model:

$$y_t \sim wNormal(\boldsymbol{\phi}_1\boldsymbol{x}_t, \sigma_1^2) + (1-w)Normal(\boldsymbol{\phi}_2\boldsymbol{x}_t, \sigma_2^2). \tag{5}$$

However, as this sum-form is inconvenient to use in derivations ($\log(a+b)$ is difficult to decompose), we assume additional unobserved binary values z_t that indicate from which component each observation t was generated. In the case of a two-component model only one variable is required with the definition:

$$z_t = 1 \implies y_t \sim Normal(\boldsymbol{\phi}_1 \boldsymbol{x}_t, \sigma_1^2), \tag{6}$$
$$z_t = 0 \implies y_t \sim Normal(\boldsymbol{\phi}_2 \boldsymbol{x}_t, \sigma_2^2). \tag{7}$$

The full observation model can then be written as

$$y_t \sim Normal(\boldsymbol{\phi}_1 \boldsymbol{x}_t, \sigma_1^2)^{z_t} Normal(\boldsymbol{\phi}_2 \boldsymbol{x}_t, \sigma_2^2)^{1-z_t} \tag{8}$$
$$z_t \sim Bernoulli(w). \tag{9}$$

The original mixture distribution can be recovered by integrating (summing) over z_t.

1.3 Estimating MAP parameters using EM

The expectation-maximization (EM) algorithm is an iterative approach for finding the maximum a posteriori (MAP) parameters of a statistical model with missing data. The general approach can be used also for models without missing data, in which case it reduces to coordinate ascent in the parameter space with respect to the model log likelihood.

In general, there are two variants of the EM-algorithm, called *soft EM* and *hard EM*. In soft EM we want to maximize the incomplete data likelihood by treating the missing data as latent variables, whereas in hard EM we are interested in maximizing the complete data likelihood by interpreting the missing data as model parameters. The variant introduced in Lecture 5 was soft EM. For completeness we will discuss both variants here. For more discussion, see e.g. *Pattern Recognition and Machine Learning* (Bishop 2006) chapter 9.

1.3.1 Soft EM

In soft EM we are interested in maximizing the incomplete data log-likelihood

$$\log p(\boldsymbol{D}, \boldsymbol{\theta}) \tag{10}$$

by assuming that there exist some unobserved latent variables $\boldsymbol{Z}$ that, if observed, would better explain the data $\boldsymbol{D}$ and thus allow the posterior of the parameters $\boldsymbol{\theta}$ to be derived more easily. The complete data log-likelihood is denoted by

$$\log p(\boldsymbol{D}, \boldsymbol{Z}, \boldsymbol{\theta}). \tag{11}$$

However, as the variables $\boldsymbol{Z}$ have not been observed, we will estimate the expected values of these variables, which can in turn be used to estimate the model parameters. This leads to an iterative algorithm in which the model parameters $\boldsymbol{\theta}$ will converge to values that (locally) maximize the incomplete data log-likelihood.

If binary coding is used for the latent variables, as is often done in the case of mixture models, we call the probability that a latent variable has value 1 (instead of 0) as the the *responsibility score* of that latent variable. Mathematically, the responsibility score for the

latent variable z_{tk}, which indicates whether observation t was generated by component k, is defined as

$$\gamma(z_{tk}) = E[p(z_{tk} = 1)|\boldsymbol{\theta}, \boldsymbol{d}_t]. \tag{12}$$

Notice that this step is based on the current estimate of the model parameters $\boldsymbol{\theta}$. The calculation of these responsibility scores (Expectancies) is called the E-step of the EM-algorithm.

Now to estimate the most likely model parameters, we want to find the maximum of the expectation of the complete data log-likelihood, which equals to

$$\boldsymbol{\theta} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}[\sum_{\boldsymbol{Z}} \Big(\gamma(\boldsymbol{Z}) \log p(\boldsymbol{D}, \boldsymbol{Z}|\boldsymbol{\theta})\Big) + \log p(\boldsymbol{\theta})], \tag{13}$$

where $p(\boldsymbol{\theta})$ is the prior for the model parameters. The calculation of the optimal parameters (Maximization) is called the M-step of the EM-algorithm.

In soft EM we have the guarantee that the E-step followed by the M-step will not decrease the incomplete data log-likelihood.

1.3.2 Hard EM

In hard EM we are interested in optimizing the complete data likelihood by interpreting the missing data as model parameters. By optimizing each of these in turn, we are essentially performing coordinate ascent in the joint space of the actual model parameters and missing data values.

The algorithm is composed of two alternating steps. In the E-step, we calculate the mode of the missing data $\boldsymbol{Z}$ given the current estimate of the model parameters and observed data:

$$\hat{\boldsymbol{z}}_t = E_{mode}[\boldsymbol{z}_t|\boldsymbol{\theta}, \boldsymbol{d}_t]. \tag{14}$$

The main difference from soft EM is that we calculate the mode instead of the mean. The reason for this is that the mode of the distribution will always be a valid value for the missing data[1]. In effect this causes us to make 'hard' allocations for the latent variables in hard EM, whereas in soft EM we used 'soft' probability values.

In the M-step we find the parameters that maximize the complete data likelihood of the model, given the current estimated values for the latent variables and observed data:

$$\boldsymbol{\theta} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}}[\log p(\boldsymbol{D}, \hat{\boldsymbol{Z}}|\boldsymbol{\theta}) + \log p(\boldsymbol{\theta})]. \tag{15}$$

In hard EM we have the guarantee that neither the E-step or the M-step does not decrease the complete data log-likelihood. This algorithm will cause both the missing data $\boldsymbol{Z}$ and model parameters $\boldsymbol{\theta}$ to converge to values that (locally) maximize the complete data log likelihood.

[1]For example the mode of a binary variable is always either 0 or 1 (when ties are broken in some fashion), but the mean is in [0, 1]. This guarantee allow us to calculate the complete data likelihood while maintaining the formulation's equivalence to the incomplete data likelihood. For example, Eq. 8-9 equals Eq. 5 only when z_t are binary.

1.3.3 Notes

The benefits of hard EM are that it is generally faster to converge and that it provides us with MAP hard allocations of the latent variables (if these are of interest). The drawbacks are that it doesn't find the MAP model parameters regarding the incomplete data model and that it is generally more likely to get stuck in local optima. Conversely, the benefits of soft EM are that it is generally less likely to get stuck in local optima and that it gives us the MAP model parameters regarding the incomplete data model. The drawbacks are that it is generally slower to converge and that it doesn't give us the MAP hard allocation of the latent variables.

It should be noted that the EM algorithm results will be optimal only locally. For multimodal distributions, it is often a good practice to run the algorithm separately from multiple independent starting points, and select the end result with best likelihood as the final result.

In this exercise, you can use whichever formulation of EM you feel more comfortable with, as long as you use it consistently and state which one you selected.

2 Project Work Description

In this project work your task is the following:

1. Write down the mathematical description for a mixture model with two linear components:
   - Full likelihood function (2p)
   - Full log-likelihood function (2p)
2. Derive the EM update equations for the parameters of this model:
   - Update equations for $\boldsymbol{\phi}_1$ and $\boldsymbol{\phi}_2$ (2p)
   - Update equations for σ_1^2 and σ_2^2 (2p)
   - Update equation for w (2p)
   - Update equation for z_t (2p)
3. Implement this model (attach for example selected parts of your code):
   - Model initialization (2p)
   - Log-likelihood function (4p)
   - Update equations (8p)
4. Test this model:
   - Generate training and validation data from the mixture model. Analyze how well the model (trained with the training data) can explain the validation data with different data dimensionality and different amounts of generated data when you only do the fitting once. How do the results change, and why, when you start the EM from multiple locations and choose the best fit? (6p)
5. Compare the two models (simple linear model and mixture with two linear components). Do the analyses with both low (eg. 2) and high (eg. 10) data dimensionality as well as with small (eg. 10) and large (eg. 100) amount of samples. Use separate validation set as before.
   - Draw data from the simple linear model, analyze how well each of the candidate models is able to explain the data. (6p)
   - Draw data from the mixture model, analyze how well each of the candidate models is able to explain the data. (6p)
   - Draw data from the mixture model, analyze which candidate model is able to explain the data better as a function of the similarity of the two linear components in the true model (e.g. cosine similarity). Explain your findings. (6p)

We have provided you with derivations for the simple linear model (below) and example code where the simple linear model has been implemented (Python code in a zip file). The model definition for the two component mixture model is also given below. Hyperprior parameters are: $\alpha_w = 3$, $\beta_w = 3$, $\lambda_\phi = 1$, $\mu_\phi = 0$, $\alpha_{\sigma^2} = 5$, $\beta_{\sigma^2} = 1$.

3 Reporting Instructions

Return your project work as a **single file pdf** document to **antti.kangasraasio[at]aalto.fi**. You can do the project work by yourself, or with a pair from the course.

Requirements:

- **Email title** starts with text: `[project work]`.
- Document has a title page with your name(s), student number(s), course name, date.
- Document format should be similar to this document (e.g. in latex use `\documentclass[12pt]{article}`).
- Explanation of your work (enough detail to evaluate that you have done all of the required work items listed in project work description; your grade will be based on the document alone).
- Returned before due date **24.4.2016 at 23:59**; late submissions are not graded.
- Document should be no longer than 10 pages including title page; pages after that are not graded.

4 Grading

- 40-50 p / 2 points
- 30-39 p / 1.5 points
- 20-29 p / 1 point
- $<$ 20 p / no pass

5 Clarification etc.

In case you need clarification to the project work instructions, find yourself utterly lost, or believe that you have found a mistake in some part of this document, you can contact the assistant by email: antti.kangasraasio[at]aalto.fi. Also in this case, start the email title with the text `[project work]`. Please expect more delay to the responses near the project deadline (week 16).

5.1 Estimating posterior of linear model using EM

5.1.1 Model definition

$$p(y_t|\boldsymbol{\phi}, \sigma^2, \boldsymbol{x}_t) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(\boldsymbol{x}_t\boldsymbol{\phi} - y_t)^2}{2\sigma^2}), \tag{16}$$

$$p(\boldsymbol{\phi}|\sigma^2, \boldsymbol{\Sigma}_\phi, \boldsymbol{\mu}_\phi) = (2\pi\sigma^2)^{-0.5P}|\boldsymbol{\Sigma}_\phi|^{-0.5} \exp(-\frac{1}{2\sigma^2}(\boldsymbol{\phi} - \boldsymbol{\mu}_\phi)^T\boldsymbol{\Sigma}_\phi^{-1}(\boldsymbol{\phi} - \boldsymbol{\mu}_\phi)), \tag{17}$$

$$p(\sigma^2|\alpha_{\sigma^2}, \beta_{\sigma^2}) = \frac{\beta_{\sigma^2}^{\alpha_{\sigma^2}}}{\Gamma(\alpha_{\sigma^2})}(\sigma^2)^{-\alpha_{\sigma^2}-1} \exp(-\frac{\beta_{\sigma^2}}{\sigma^2}). \tag{18}$$

Observations $(y_t, \boldsymbol{x}_t)$, $t \in \{1, ..., T\}$. Using $\boldsymbol{\mu}_\phi = \mathbf{1}\mu_\phi$ and $\boldsymbol{\Sigma}_\phi = diag(\lambda_\phi)$.

5.1.2 Log-likelihood

$$\log p(y, \boldsymbol{\phi}, \sigma^2|\cdots) = \sum_{t=1}^{T}\Big(- 0.5\log(2\pi) - 0.5\log(\sigma^2) - \frac{(\boldsymbol{x}_t\boldsymbol{\phi} - y_t)^2}{2\sigma^2}\Big) \tag{19}$$

$$- 0.5P\log(2\pi\sigma^2\lambda_\phi) - \frac{1}{2\sigma^2\lambda_\phi}(\boldsymbol{\phi} - \boldsymbol{\mu}_\phi)^T(\boldsymbol{\phi} - \boldsymbol{\mu}_\phi) \tag{20}$$

$$+ \alpha_{\sigma^2}\log(\beta_{\sigma^2}) - \log(\Gamma(\alpha_{\sigma^2})) - (\alpha_{\sigma^2} + 1)\log(\sigma^2) - \frac{\beta_{\sigma^2}}{\sigma^2}, \tag{21}$$

where P is the dimensionality of the data.

5.1.3 EM-equations

EM reduces to coordinate ascent in the parameter space as all variables are observed.

$$\frac{\partial}{\partial\sigma^2}\log p(\cdot) = 0 \Leftrightarrow \qquad \sum_{t=1}^{T}\Big(- \frac{1}{2\sigma^2} + \frac{(\boldsymbol{x}_t\boldsymbol{\phi} - y_t)^2}{2\sigma^4}\Big) \tag{22}$$

$$-0.5P\frac{1}{\sigma^2} + \frac{1}{2\sigma^4\lambda_\phi}(\boldsymbol{\phi} - \boldsymbol{\mu}_\phi)^T(\boldsymbol{\phi} - \boldsymbol{\mu}_\phi) \tag{23}$$

$$-(\alpha_{\sigma^2} + 1)\frac{1}{\sigma^2} + \frac{\beta_{\sigma^2}}{\sigma^4} = 0 \tag{24}$$

$$\Leftrightarrow \qquad \sigma^2 = \frac{\beta_{\sigma^2} + \frac{1}{2}\sum_{t=1}^{T}(\boldsymbol{x}_t\boldsymbol{\phi} - y_t)^2 + \frac{1}{2\lambda_\phi}(\boldsymbol{\phi} - \boldsymbol{\mu}_\phi)^T(\boldsymbol{\phi} - \boldsymbol{\mu}_\phi)}{\alpha_{\sigma^2} + 1 + \frac{1}{2}(T + P)} \tag{25}$$

$$\frac{\partial}{\partial\boldsymbol{\phi}}\log p(\cdot) = 0 \Leftrightarrow \qquad -\sum_{t=1}^{T}\frac{\boldsymbol{x}_t^T\boldsymbol{x}_t\boldsymbol{\phi} - \boldsymbol{x}_t^T y_t}{\sigma^2} - \frac{\boldsymbol{\Sigma}_\phi^{-1}(\boldsymbol{\phi} - \boldsymbol{\mu}_\phi)}{\sigma^2} = 0 \tag{26}$$

$$\Leftrightarrow \qquad \boldsymbol{\phi} = \Big(\boldsymbol{\Sigma}_\phi^{-1} + \sum_{t=1}^{T}\boldsymbol{x}_t^T\boldsymbol{x}_t\Big)^{-1}\Big(\boldsymbol{\Sigma}_\phi^{-1}\boldsymbol{\mu}_\phi + \sum_{t=1}^{T}\boldsymbol{x}_t^T y_t\Big) \tag{27}$$

5.2 Linear mixture model with two components

5.2.1 Complete data model definition

$$p(y_t|\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \sigma_1^2, \sigma_2^2, \boldsymbol{x}_t, z_t) = \Big(\frac{1}{\sqrt{2\pi\sigma_1^2}} \exp(-\frac{(\boldsymbol{x}_t\boldsymbol{\phi}_1 - y_t)^2}{2\sigma_1^2})\Big)^{z_t} \Big(\frac{1}{\sqrt{2\pi\sigma_2^2}} \exp(-\frac{(\boldsymbol{x}_t\boldsymbol{\phi}_2 - y_t)^2}{2\sigma_2^2})\Big)^{1-z_t}, \tag{28}$$

$$p(z_t|w) = w^{z_t}(1-w)^{1-z_t}, \tag{29}$$

$$p(w|\alpha_w, \beta_w) = \frac{\Gamma(\alpha_w + \beta_w)}{\Gamma(\alpha_w)\Gamma(\beta_w)} w^{\alpha_w - 1}(1-w)^{\beta_w - 1}, \tag{30}$$

$$p(\boldsymbol{\phi}_j|\sigma_j^2, \boldsymbol{\Sigma}_\phi, \boldsymbol{\mu}_\phi) = (2\pi\sigma_j^2)^{-0.5P} |\boldsymbol{\Sigma}_\phi|^{-0.5} \exp(-\frac{1}{2\sigma_j^2}(\boldsymbol{\phi}_j - \boldsymbol{\mu}_\phi)^T \boldsymbol{\Sigma}_\phi^{-1}(\boldsymbol{\phi}_j - \boldsymbol{\mu}_\phi)), \tag{31}$$

$$p(\sigma_j^2|\alpha_{\sigma^2}, \beta_{\sigma^2}) = \frac{\beta_{\sigma^2}^{\alpha_{\sigma^2}}}{\Gamma(\alpha_{\sigma^2})} (\sigma_j^2)^{-\alpha_{\sigma^2}-1} \exp(-\frac{\beta_{\sigma^2}}{\sigma_j^2}). \tag{32}$$

Observations $(y_t, \boldsymbol{x}_t)$, $t \in \{1, ..., T\}$, binary class indicator $z_t \in \{0, 1\}$, mixture component weights $w \in [0, 1]$, model index $j \in \{1, 2\}$. Using $\boldsymbol{\mu}_\phi = \mathbf{1}\mu_\phi$ and $\boldsymbol{\Sigma}_\phi = diag(\lambda_\phi)$.