Extra exercises for STK4030

Geir Storvik

Fall 2013

Exercise 1 (Classification and unlabeled data)
We will in this exercise see how estimation can be based on training data with known classes combined with data where the class labels are unknown.

Assume

$$\mathcal{T} = \{\boldsymbol{x}_{k,j}, k = 1, ..., K, j = 1, ..., n_k, \boldsymbol{x}_i, i = 1, ..., N\}$$

where $\boldsymbol{x}_{k,j}$ is an observation from class k while $\boldsymbol{x}_i$ is an observation with unknown class.

We will assume that $\pi_1, ..., \pi_K$ are all equal to $1/K$, while $f_k(\boldsymbol{x}; \boldsymbol{\theta})$ is the probability density for data from class k. We want to estimate $\boldsymbol{\theta}$ based on data $\mathcal{T}$.

(a) Show that

$$f(\boldsymbol{x}_i; \boldsymbol{\theta}) = \sum_{k=1}^{K} \frac{1}{K} f_k(\boldsymbol{x}_i; \boldsymbol{\theta})$$

and use this to derive an expression for the likelihood function $L(\boldsymbol{\theta}; \mathcal{T}) = p(\mathcal{T}|\boldsymbol{\theta})$ as well as the log-likelihood function $l(\boldsymbol{\theta}; \mathcal{T}) = \log[l(\boldsymbol{\theta}; \mathcal{T})]$ for $\mathcal{T}$.

Discuss estimation of $\boldsymbol{\theta}$ only based on $\{\boldsymbol{x}_{k,j}, k = 1, ..., K, j = 1, ..., n_k\}$ related to estimation of $\boldsymbol{\theta}$ based on the whole $\mathcal{T}$.

In order to estimate $\boldsymbol{\theta}$, we will use an iterative procedure called the *EM algorithm* (EM is a abbreviation for Expectation-Maximization). This is a algorithm that can be used when parts of the data is unobserved (or missing or incomplete). In our case we can think of the classes corresponding to $\boldsymbol{x}_i, i = 1, ..., N$ as missing. Let $\boldsymbol{C}_m = (C_1^m, ..., C_N^m)$ be the missing classes.

Denote by $p(\mathcal{T}, \boldsymbol{C}_m|\boldsymbol{\theta})$ the probability density for the "complete" data, that is when *all* classes are known. Treated as a function of $\boldsymbol{\theta}$ this would be the function to maximize if $\boldsymbol{C}_m$ was known. The first step of the EM-algorithm is to **E***stimate* $\log p(\mathcal{T}, \boldsymbol{C}_m|\boldsymbol{\theta})$ by its expectation:

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}') = \mathsf{E}^{\boldsymbol{C}_m}[\log p(\mathcal{T}, \boldsymbol{C}_m|\boldsymbol{\theta})|\mathcal{T}; \boldsymbol{\theta}'].$$

Here the expectation is under the model and our current best guess of $\boldsymbol{\theta}$ being equal to $\boldsymbol{\theta}'$. The next step is the **M***aximization* where a new value for $\boldsymbol{\theta}$ is obtained by maximizing $Q(\boldsymbol{\theta}, \boldsymbol{\theta}')$ wrt $\boldsymbol{\theta}$.

Specifically, the algorithms runs as follows:

Start with a sensible initial value of $\boldsymbol{\theta}^0$ for $\boldsymbol{\theta}$ (e.g. estimates based on only using the data with known classes). For $s = 1, 2, ...$, perform the following steps:

E-step : Find

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{s-1}) = \mathsf{E}^{\boldsymbol{C}_m}[\log p(\mathcal{T}, \boldsymbol{C}_m|\boldsymbol{\theta})|\mathcal{T}; \boldsymbol{\theta}^{s-1}]$$

M-step : Find $\boldsymbol{\theta}^s$ as the value which maximizes $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{s-1})$ wrt $\boldsymbol{\theta}$.

It can be shown that (see e.g. sec 8.5 in the text book) that at each iteration there will be an increase (or at least not a decrease) in $L(\boldsymbol{\theta}; \mathcal{T}) =$, that is

$$L(\boldsymbol{\theta}^s; \mathcal{T}) \geq L(\boldsymbol{\theta}^{s-1}; \mathcal{T})$$

If the (log-) likelihood function also is bounded (which it often is), the algorithm will converge to a (local) maximum.

(b) Show that for our situation

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{s-1}) = &\sum_{k=1}^{K}\sum_{j=1}^{n_k}[\log p_k(\boldsymbol{x}_{k,j}; \boldsymbol{\theta}) + \log \pi_k] + \\ &\sum_{i=1}^{N}\sum_{k=1}^{K} p(C_i = k|\boldsymbol{x}_i; \boldsymbol{\theta}^{s-1})[\log p_k(\boldsymbol{x}_i; \boldsymbol{\theta}) + \log \pi_k] \end{aligned}$$

where $p(C_i = k|\boldsymbol{x}_i; \boldsymbol{\theta}^{s-1})$ is the conditional probability for $C_i = k$ given that the true parameters are $\boldsymbol{\theta}^{s-1}$.

Hint: Write the log-likelihood for (x_i, C_i) as

$$\sum_{k=1}^{K} I(C_i = k)[\log p_k(x_i) + \log \pi_k]$$

where $I(C_i = k) = 1$ is 1 if $C_i = k$ and 0 otherwise.

(c) Show that maximization of $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{s-1})$ wrt $\pi_k, k = 1, ..., K$ gives

$$\hat{\pi}_k^s = \frac{n_k + \sum_{i=1}^{N} p(C_i = k|\boldsymbol{x}_i; \boldsymbol{\theta}^{s-1})}{n + N}$$

where $n = \sum_k n_k$. Comment on this result!

Hint: Remember that $\sum_k \pi_k = 1$. Use Lagrange's method in order to take this constraint into account.

In the rest of the exercise we will assume $p_k(\boldsymbol{x};\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$. In this case $\boldsymbol{\theta} = \{(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), k = 1, ..., K\}$.

(d) Show that

$$\hat{\boldsymbol{\mu}}_k^s = \frac{\sum_{j=1}^{n_k} \boldsymbol{x}_{k,j} + \sum_{i=1} p(C_i = k|\boldsymbol{x}_i;\boldsymbol{\theta}^{s-1})\boldsymbol{x}_i}{n_k + \sum_{i=1} p(C_i = k|\boldsymbol{x}_i;\boldsymbol{\theta}^{s-1})}.$$

by differentiating $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{s-1})$ wrt $\boldsymbol{\mu}_k$ and but to zero.

One can also show that (you do not need to do this)

$$\widehat{\boldsymbol{\Sigma}}_k^s = \frac{\sum_{j=1}^{n_k}(x_{k,j} - \hat{\boldsymbol{\mu}}_k)(x_{k,j} - \hat{\boldsymbol{\mu}}_k)^T + \sum_{i=1} p(C_i = k|\boldsymbol{x}_i;\boldsymbol{\theta}^{s-1})(x_i - \hat{\boldsymbol{\mu}}_k)(x_i - \hat{\boldsymbol{\mu}}_k)^T}{n_k + \sum_{i=1} p(C_i = k|\boldsymbol{x}_i;\boldsymbol{\theta}^{s-1})}.$$

Comment also on these results!

(e) On the course web page, there is an R-script called `EM_mix.R` which for $K = 2$ first simulate data with $n_k = 10$ and $N = 1000$ from a known set of parameters, followed by estimation only using data with known classes and thereafter the full data set. Try out this script many times and compare the performances of the different estimators. Comment on the results!

Exercise 2 (Analysis of the iris data)
This famous (Fisher's or Anderson's) iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are *Iris setosa*, *versicolor*, and *virginica*.

The data is available in R names `iris`. We will in this exercise consider different methods for constructing a classification method for these data.

- Specify a method for comparing different classification methods (e.g splitting into training/test sets, cross-validation ...)
- Try out different classification methods you have learned on the iris data. Which method performs best?
- For the best method, also make a socalled *confussion matrix*, that is a matrix with rows the true classes and columns the predicted classes.

Exercise 3 (GAM on bone dataset)
The `bone` dataset, available from
http://www.uio.no/studier/emner/matnat/math/STK4030/h13/data/
are relative spinal bone mineral density measurements on 261 North American adolescents. Each value is the difference in `spnbmd` taken on two consecutive visits, divided by the average. The age is the average age over the two visits.

Variables:

`idnum` : identifies the child, and hence the repeat measurements

`age` : average age of child when measurements were taken

`gender` : male or female

`spnbmd` : Relative Spinal bone mineral density measurement

Consider the following R-commands:

```
library(mgcv)
library(mgcv)
bone = read.table("../data/bone.data",header=T)
bone.gam = gam(spnbmd~s(age)+gender,data=bone)
plot(bone.gam)
summary(bone.gam)
```

(*a*) Explain the model behind the `gam` call.

(*b*) Copy the commands from the lecture in order to find the optimal degrees of smoothness using both the AIC and the BIC criterions.

Hint: Specify the `tuning.scale` vector in the range 0.1 to 6.

(*c*) Also fit a model where `age` is linear. Use some criterion to compare the different models.

(*d*) Discuss the results obtained.

Exercise 4 (Trees and CPUs)
In this exercise we will use regression trees for fitting a model to the relative performance measure and characteristics of 209 CPUs. The dataset can be made available in R by the command

```
library(MASS)
```

More information about the dataset can be obtained by

```
names(cpus)
help(cpus)
```

Note that the first variable is the name of Manufacturer and model and is not a variable to be used.

We further need to make the functions for trees available through the command

```
library(rpart)
```

(*a*) In order to fit a tree to data, the command `rpart` can be used. Try out the commands

```
cpus.rp <- rpart(log10(perf)~syct+mmin+mmax+cach+chmin+chmax,
                 cp=1e-3,cpus)
print(cpus.rp,cp=0.01)
plot(cpus.rp,uniform=T);text(cpus.rp,digits=3)
```

Try to understand what you get out of this.

Also try to remove 10 randomly chosen points from the data-set and fit a new tree. Discuss differences in the trees.

(*b*) An important part in fitting an approperiate tree is pruning. The amount of pruning depends on the complexity parameter (CP). Try out the commands

```
printcp(cpus.rp)
plotcp(cpus.rp)
```

and describe the different columns in the output from the first command. Using the 1-SE rule (choosing the smallest model within one standard deviation of the mimimum), which CP value would you choose?

(*c*) You can obtained a pruned tree with a given CPvalue with

```
cpus.rp1 <- prune(cpus.rp,cp=??)
```

where `??` is to be replaced by a specific value. Try this out and make a plot of the pruned three. Comment on differences from the full tree.

(*d*) Calculate the in-sample error based on the pruned tree.

Hint: Use the generic `predict` command.

(*e*) Try to write down a model and a likelihood function which justifies the $Q_m(T)$ measure as defined in (9.15) in the text-book.

Exercise 5 (Measures for classification trees)
Consider a classificaton tree where the full space $\mathcal{R}^p$ is divided into $|T|$ regions $R_1, ..., R_{|T|}$. Further assume that

$$\Pr(Y = k|x \in R_m) = p_{mk}.$$

For data $\{(\boldsymbol{x}_i, y_i), i = 1, ..., n\}$, consider a Likelihood function of the form

$$L = \prod_{i=1}^{n} \prod_{k=1}^{K} p_{m(i)k}^{I(y_i=k)}$$

where $m(i)$ is the region that $\boldsymbol{x}_i$ belongs to.

(*a*) Discuss what kind of assumptions the likelihood above is based on.

(*b*) Define

$$N_m = \sum_{i=1}^{n} I(\boldsymbol{x}_i \in R_m),$$
$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{i:\boldsymbol{x}_i \in R_m} I(y_i = k).$$

Show that

$$l = \log(L) = \sum_{m=1}^{|T|} \sum_{k=1}^{K} N_m \hat{p}_{mk} \log p_{mk}.$$

(*c*) Assume now that we insert $\hat{p}_{mk}$ as an estimate for p_{mk} into the loglikelihood. Show that

$$-l = \sum_{m=1}^{|T|} N_m Q_m(T)$$

for a suitable choise of $Q_m(T)$. Which of the measures defined in equation (9.17) in the textbook does this measure correspond to?

Exercise 6 (Leave-one-out cross-validation)
The leave-one out cross-validation for estimate of prediction error is defined through

$$CV = \frac{1}{N} \sum_{i=1}^{N} L(y_i, \hat{f}^{-i}(x_i))$$

where $\hat{f}^{-i}(x_i)$ is the prediction of $f(x_i)$ based on all observations *except* the ith observation.

Consider a linear smoothing model where in-sample predictions based on all data are given by $\hat{f}(x_i) = \sum_{l=1}^{N} S_{il} y_l$ while the leave-one out cross-validation estimate is given by $\hat{f}^{-i}(x_i) = \sum_{l \neq i} S_{il}^{-i} y_l$.

(*a*) Show that if $S_{il}^{-i} = S_{il}/(1 - S_{ii})$, then

$$y_i - \hat{f}^{-i}(x_i) = \frac{y_i - \hat{f}(x_i)}{1 - S_{ii}}. \qquad (*)$$

Discuss what practical consequences this result have with respect to calculation of CV.

(*b*) Consider now linear regression $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ with the least squares estimates $\hat{\boldsymbol{\beta}} = [\boldsymbol{X}^T\boldsymbol{X}]^{-1}\boldsymbol{X}^T\boldsymbol{Y}$. Denote by $\boldsymbol{X}_{-i}$ the design matrix excluding row i. Show that

$$\hat{f}^{-i}(\boldsymbol{x}_i) = \boldsymbol{x}_i^T [\boldsymbol{X}_{-i}^T \boldsymbol{X}_{-i}]^{-1} [\boldsymbol{X}^T\boldsymbol{Y} - \boldsymbol{x}_i y_i]$$

(*c*) Show that $\boldsymbol{X}_{-i}^T\boldsymbol{X}_{-i} = \boldsymbol{X}^T\boldsymbol{X} - \boldsymbol{x}_i\boldsymbol{x}_i^T$ and use this to show that

$$[\boldsymbol{X}_{-i}^T\boldsymbol{X}_{-i}]^{-1} = [\boldsymbol{X}^T\boldsymbol{X}]^{-1} + \frac{[\boldsymbol{X}^T\boldsymbol{X}]^{-1}\boldsymbol{x}_i\boldsymbol{x}_i^T[\boldsymbol{X}^T\boldsymbol{X}]^{-1}}{1 - \boldsymbol{x}_i^T[\boldsymbol{X}^T\boldsymbol{X}]^{-1}\boldsymbol{x}_i}.$$

(*d*) Show that $S_{ii} = \boldsymbol{x}_i^T[\boldsymbol{X}^T\boldsymbol{X}]^{-1}\boldsymbol{x}_i$.

(*e*) Use these results to show that

$$\hat{f}^{-i}(\boldsymbol{x}_i) = \frac{1}{1 - S_{ii}}\hat{f}(x_i) - \frac{S_{ii}}{1 - S_{ii}}y_i$$

and finally show that for linear regression with ordinary least squares estimation, equation (*) is satisfied.

Exercise 7

The `zip` dataset contains normalized handwritten digits, automatically scanned from envelopes by the U.S. Postal Service. The original scanned digits are binary and of different sizes and orientations; the images here have been deslanted and size normalized, resulting in 16 x 16 grayscale images (Le Cun et al., 1990).

The data are in two files, and each line consists of the digit id (0-9) followed by the 256 grayscale values. There are 7291 training observations and 2007 test observations.

At the course homepage a file `zip_nn_exer.R` contain commands for performing classification based on neural network.

(*a*) Go through the commands and try to understand what is done. In particular comment on the part doing the normalization.

(*b*) Run the commands and comment on the results.

(*c*) Compare this with some of the other methods you have learned through the course.