A statistical R package for cluster weighted modelling

Simona Minotti[†] Giorgio A. Spedicato[‡]

[†]Universitá La Bicocca
Milan, Italy

[‡] Universitá la Sapienza
Rome, Italy

March 25, 2013

Table of contents

Outline

The CWMEM package

The CWREM package has been developed to estimate Cluster Weighted Modeling (CWM) models as described in ([Murphy]) and ([Minotti2009]).
R code has been inspired by ([Murphy]) original Matlab code, which has been converted in R and improved. Until now, it is the unique R package to estimate CWM models.
The package provides the estimation of local CWM and performs the classification tasks by means of posterior group probabilities.
The models' parameters estimation is performed by EM algorithm ([Dempster1977]).

Review of CWM

CMW framework is expressed by the general formula:

$$p(\mathbf{x}, y) = \sum_{g=1}^{G} p(y|\mathbf{x}, \Omega_g)\, p(\mathbf{x}|\Omega_g)\, \pi_g$$

where $\pi_g = p(\Omega_g)$ is the mixing weight of group Ω_g, $p(\mathbf{x}|\Omega_g)$ is the probability density of $\mathbf{x}$ given Ω_g and $p(y|\mathbf{x}, \Omega_g)$ is the conditional density of the response variable Y given the predictor vector $\mathbf{x}$ and the group Ω_g, $g = 1, \ldots, G$

Review of CWM

The $p(\mathbf{x}|\Omega_g)$, $g = 1, \ldots, G$ are usually assumed to be multivariate Gaussian, that is $p(\mathbf{x}|\Omega_g) = \phi_d(\mathbf{x}; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)$. Moreover, the $p(y|\mathbf{x}, \Omega_g)$, $g = 1, \ldots, G$, can be modeled again by a Gaussian distribution with variance $\sigma^2_{\varepsilon,g}$ around some deterministic function of $\mathbf{x}$, say $\gamma_g(\mathbf{x})$; here, we refer to local models given by a linear mapping $\gamma_g(\mathbf{x}) = \mathbf{b}'_g\mathbf{x} + b_{g0}$, with $\mathbf{b}_g \in \Re^d$, $b_{g0} \in \Re$ and then:

$$p(\mathbf{x}, y; \boldsymbol{\theta}) = \sum_{g=1}^{G} \phi(y; \mathbf{b}'_g\mathbf{x} + b_{g0}, \sigma^2_{\varepsilon,g})\, \phi_d(\mathbf{x}; \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g)\, \pi_g$$

Review of CWM

It has been demonstrated ([MVI2009]) that CWM generalizes Finite Mixtures of Regressions (FMR),

$$p(y|\mathbf{x};\psi) = \sum_{g=1}^{G} p(y|\mathbf{x},\Omega_g)\pi_g = \sum_{g=1}^{G} \phi(y;\mathbf{b}'_g\mathbf{x} + b_{g0}, \sigma^2_{\varepsilon,g})\,\pi_g$$

and Finite Mixtures of Regressions with Concomitant variables (FMRC),

$$p(y|\mathbf{x};\psi^*) = \sum_{g=1}^{G} \phi(y;\mathbf{b}'_g\mathbf{x} + b_{g0}, \sigma^2_{\varepsilon,g})\,p(\Omega_g|\mathbf{x},\gamma)$$

The expectation step

The maximum-likelihood estimation of parameters $\psi_g = (\pi_g, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g, b_{g0}, \mathbf{b}_g, \sigma^2_{\epsilon,g})'$ $(g = 1, \ldots, G)$ can be performed by means of the EM algorithm ([Dempster1977]), here re-written for the CWM following the usual statistical approach introduced for Mixtures Models.

E-step: Given the current estimates $\hat{\psi}_g^{(k)}$ for the g-th group at the k-th iteration, replace the missing value z_{ng} (where $z_{ng} = 1$ if the n-th observation comes from the g-th group and $z_{ng} = 0$ elsewhere) by the estimate of the posterior probability $\tau_{ng}^{(k)}$ $(n = 1, \ldots, N;\ g = 1, \ldots, G)$ of group membership conditional on $(\mathbf{x}_n, y_n)$:

$$\hat{\tau}_{ng}^{(k)} = \frac{\pi_g^{(k)} \phi(y_n; \mathbf{b}_g'^{(k)} \mathbf{x}_n + b_{g0}^{(k)}, \sigma^{2(k)}_{\epsilon,g}) \phi_d(\mathbf{x}_n; \boldsymbol{\mu}_g^{(k)}, \boldsymbol{\Sigma}_g^{(k)})}{\sum_{j=1}^{G} \pi_j^{(k)} \phi(y_n; \mathbf{b}_j'^{(k)} \mathbf{x}_n + b_{j0}^{(k)}, \sigma^{2(k)}_{\epsilon,j}) \phi_d(\mathbf{x}_n; \boldsymbol{\mu}_j^{(k)}, \boldsymbol{\Sigma}_j^{(k)})}.$$

The maximization step

M-step: Given the estimates $\hat{\tau}_{ng}^{(k)}$, it can be proved that the updates of the parameter estimates $\hat{\psi}_g^{(k+1)}$ $(g = 1, \ldots, G)$ at the $(k+1)$-th iteration are:

$$
\begin{aligned}
\pi_g^{(k+1)} &= \frac{1}{N} \sum_{n=1}^{N} \tau_{ng}^{(k)} \\
\boldsymbol{\mu}_g^{(k+1)} &= \frac{\sum_{n=1}^{N} \tau_{ng}^{(k)} \mathbf{x}_n}{\sum_{n=1}^{N} \tau_{ng}^{(k)}} \\
\boldsymbol{\Sigma}_g^{(k+1)} &= \frac{\sum_{n=1}^{N} \tau_{ng}^{(k)} (\mathbf{x}_n - \boldsymbol{\mu}_g^{(k+1)})(\mathbf{x}_n - \boldsymbol{\mu}_g^{(k+1)})'}{\sum_{n=1}^{N} \tau_{ng}^{(k)}} \\
\mathbf{b}_g^{(k+1)} &= \frac{\frac{\sum_{n=1}^{N} \tau_{ng}^{(k)} y_n \mathbf{x}_n}{\sum_{n=1}^{N} \tau_{ng}^{(k)}} - \frac{\sum_{n=1}^{N} \tau_{ng}^{(k)} y_n}{\sum_{n=1}^{N} \tau_{ng}^{(k)}} \frac{\sum_{n=1}^{N} \tau_{ng}^{(k)} \mathbf{x}_n}{\sum_{n=1}^{N} \tau_{ng}^{(k)}}}{\frac{\sum_{n=1}^{N} \tau_{ng}^{(k)} \mathbf{x}_n' \mathbf{x}_n}{\sum_{n=1}^{N} \tau_{ng}^{(k)}} - (\frac{\sum_{n=1}^{N} \tau_{ng}^{(k)} \mathbf{x}_n}{\sum_{n=1}^{N} \tau_{ng}^{(k)}})^2} \\
b_{g0}^{(k+1)} &= \frac{\sum_{n=1}^{N} \tau_{ng}^{(k)} y_n}{\sum_{n=1}^{N} \tau_{ng}^{(k)}} - \mathbf{b}_g^{'(k+1)} \frac{\sum_{n=1}^{N} \tau_{ng}^{(k)} \mathbf{x}_n}{\sum_{n=1}^{N} \tau_{ng}^{(k)}} \\
\sigma_{\epsilon,g}^{2(k+1)} &= \frac{\sum_{n=1}^{N} \tau_{ng}^{(k)} [y_n - (\mathbf{b}_g^{'(k+1)} \mathbf{x}_n + b_{g0}^{'(k+1)})]^2}{\sum_{n=1}^{N} \tau_{ng}^{(k)}}.
\end{aligned}
$$

Outline

General outline

cwmEm package has been compiled under 2.9.1 R version.
Current release is 0.0.1.
S3 programming style has been used, but an upgrade to the more flexible S4 programming style is scheduled in next software releases. See ([Chambers2008]) for further details on programming with R language.
The core function is **cwrEm** function that creates *cwr*objects. A cwrObj object contains cwr local models parameters estimates, prior and posteriors & groups membership informations, goodness of fit indexes.

General outline

Summary and **print** methods are applicable to *cwr* objects to obtain and summarize parameters' estimations. **Plot** method is available if the input space pertains to R^2.
Currently **cwmEm** assumes that local models are Gaussian, even if cluster weighted methodology may be extended to mixtures of non - Gaussian distributions.
The initialization process use k-means algorithm to assign each observation to one of the nc groups on which initial parameters are estimated.

The cwrEm function

The cwrEm function is the core function of **cwmEm package**. It receives data inputs and returns the block of estimates bundled in a **cwrObj** object. Optional parameters regulates eventual estimation constraints or the estimation stop rules.
The compulsory parameters are:

1. X: the independent sample observations, a $N * nc$ numeric data matrix (or data frame).
2. Y: the dependent sample observations, a $N * 1$ numeric data matrix (or data frame).
3. nc: the number of clusters to estimate.

An example of cwrEm call is:

```
helloCwr=cwrEm(X=dataSet[,1:2], Y=dataSet[,3], nc=3)
```

The cwrEm function

The output of cwrEm function are cwrEm objects. Useful informations contained in their slots are:

1. **muX, muY, sigmaX, sigmaY**: matrix or vectors of local groups parameters estimates
2. **beta0**: an array of intercepts, whose structure depend by the number of input variables and the number of groups.
3. **priorC, posterior, group**: priors and posterior probabilities, final group membership
4. **logLik, nPar, aic, bic, N**: informations used to estimated goodness of fit indices based on log-likelihood.

Summary, print and plot methods are available for cwrEm objects. They can simply be called by the syntax: print(cwrObject), summary(cwrObject), plot(cwrObject). A call to plot method will return error if the sample space is bigger than R^2.

The stepCwr function

EM algorithm achieves only local maxima. Therefore in practice it is restarted and the better solution in terms of log likelihood is elected. At each restart the cwrEm parameters are initialized by means of the parameters estimated in the previous step, if they gained a likelihood increase. **stepCwr** function applies such method to cluster weighed modeling framework.
The multiple restarts solution has been introduced for the first time in the CWM framework, following the proposal implemented in the **flexmix** ([Leish2004]) package.

The stepCwr function

The compulsory arguments of stepCwr function are the same of cwrEm function. Other arguments specify the proportion of the sample set to be used (to obtain quicker computation), the maximum number of iteration and if a change of the sampled data set is allowed.

```
> args(cwrEm)
function (X, Y, nc, prop = 0.1, nIter = 10,
changeTrainingSet = FALSE)
NULL
```

Kwown issues

Known issues are:

- Numerical problems that may arise during estimation process. The occurrence of such problems depends by the empirical given matrix.
- Slowness of the estimation process on big dataset.

Next steps

Following improvements have been planned:

- Swicht to S4 programming framework.
- Rewrite the EM kernel in C++ to obtain faster estimations.
- Eventually optimize the computational kernel, by optimizing the EM algorithm and by improving the initialization process.
- Mixed variables, non-linear local models and multilevel data structures.

Outline

Description and purpose of simulations

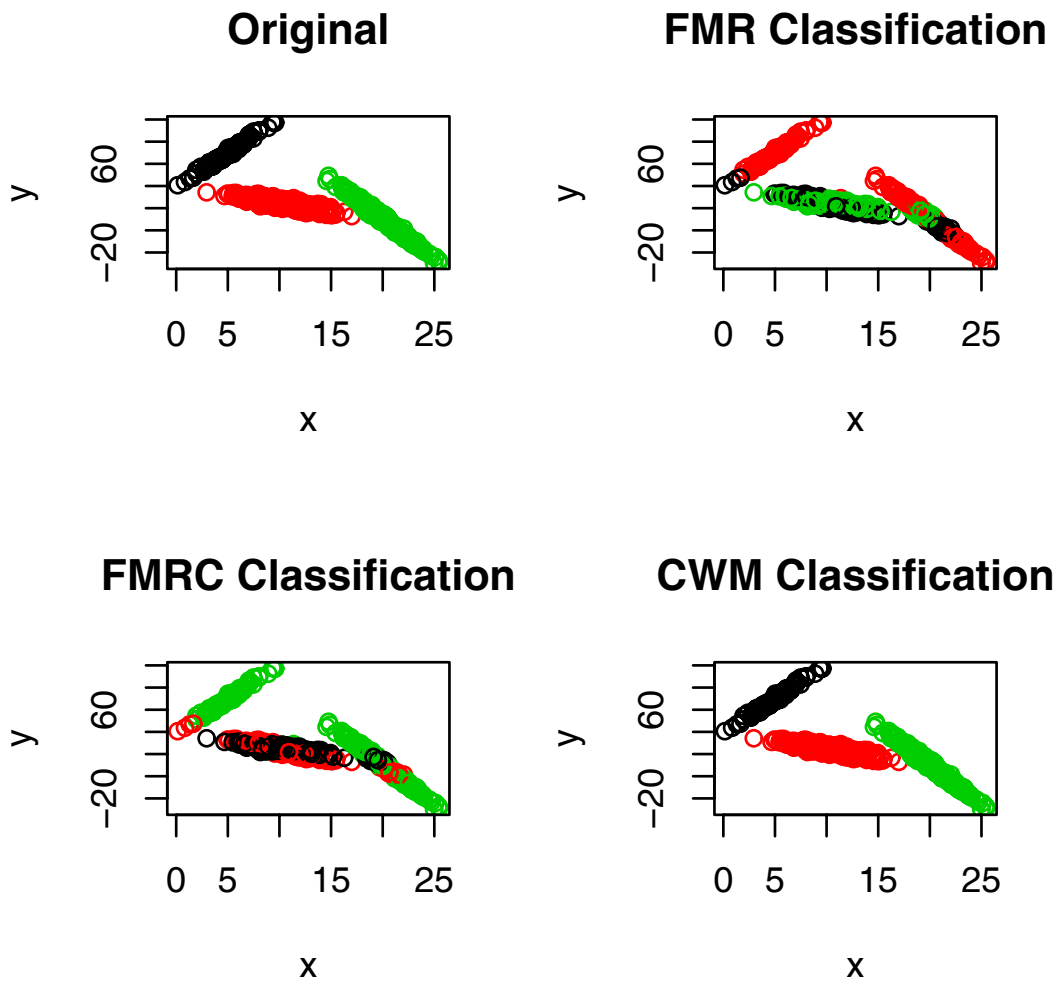
Next slides show CWM methodology applied on a simulated data set. CWREM package has been used. Comparison with FMR and FMRC as estimated by ([Leish2004]) is discussed.
The artificial data set has been generated as follows:

- tree G groups, whose sample dimension was: $N_1 = 100, N_2 = 600, N_3 = 300$.
- The samples were generated according to the parameters for $p(x|\Omega_g)$ and $p(y|x, \Omega_g)$

$$X_{G_1} \sim N(5, 2) \rightarrow y = 40 + 6x + N(0, 2)$$
$$X_{G_2} \sim N(10, 2) \rightarrow y = 40 - 1.5x + N(0, 2)$$
$$X_{G_3} \sim N(20, 2) \rightarrow y = 150 - 7x + N(0, 2)$$

- The estimation process has been repeitee 100 times and goodness of classification statistics have been recorded.

Classification graphs: FMR, FMRC, CWM compared



Classification results

Following conclusions may be drawn from the application CWM, FMR and FMRC frameworks on the data generating process:

1. FMR and FMRC do not correctly classify the original observations.
2. FMR results shows that in 38 out of 100 trials the misclassification rate varies from 7.3% to 11.1%, while in 62 out of 100 trials the misclassification rate varies from 25.6% to 49.5%.
3. As for FMRC is concerned, in 49 out of 100 trials the misclassification rate is null, while in 51 out of 100 trials the misclassification rate varies from 24.5% to 50%.
4. Finally, in CWM the misclassification rate is 0.1% in only 5 out of 100 trials, while in the other cases no misclassifications are reported.

Outline

The bibliography


Minotti, S.C.; Vittadini, G.: Local Multilevel Modeling for Comparisons of Institutional Performance: *Data Analysis and Classification: from the exploratory to the confirmatory approach. Springer, Berlin, in press, 2009*


Minotti S.C., Ingrassia S., Vittadini G.: Local Statistical Modeling by Cluster-Weighted: *Quaderno di Dipartimento QD 2009/26 (Febbraio 2009), Dipartimento di Statistica, Università degli Studi di Milano-Bicocca*


http://people.cs.ubc.ca/ murphyk/Software/index.html


Chambers, J. M.: Sofware for data analysis. Programming with R: *Springer, 2008*


Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM-algorithm: *JRSS(B) 39, 1-38, 1977*


Gershenfeld, N. (1999) : itshape The Nature of Mathematical Modelling, *Cambridge University Press, Cambridge, 101-130.*


Leisch, F. (2004). Flexmix: A general framework for finite mixture models and latent class regression in R. *Journal of Statistical Software*, **11***(8), 1-18*.