Statistical Case Studies

Wolfgang Härdle, Yuichi Mori, Philippe Vieu

May 6, 2004

Contents

Part I

Biostatistics

1 Discriminant analysis based on continuous and discrete variables: application to systematic zoology

Avner Bar-Hen, *Jean Jacques Daudin*

Expected length of the paper: 15-20 pages

1.1 Abstract

In discrimination, as in many multivariate techniques, computation of a distance between two populations is often useful. Mahalanobis'Δ^2 has become the standard measure of distance when the observations are quantitative and Hotelling derived its distribution for normal populations. The aim of this article is to adapt these results to the case where the observed characteristics are a mixture of quantitative and qualitative variables. In the first section we use Kullback-Leibler divergence to obtain a generalization of the Mahalanobis distance and we study distributional properties.

A problem frequently encountered by the practitioner in Discriminant Analysis is how to select the best variables. In mixed discriminant analysis (MDA), i.e., discriminant analysis with both continuous and discrete variables, the problem is more difficult because of the different nature of the variables. In section 2, we propose a selection variable strategy. Stopping rules are established from distributional results and penalized likelihood.

One of the aims of discriminant analysis is the allocation of unknown entities to populations that are known *a priori*. A preliminary matter for consideration before an outright or probabilistic allocation is made for an unclassified entity X is to test the assumption that X belongs to one of the predefined groups.

In section 3, for the general parametric case, a test is proposed to verify the hypothesis that X is coming from a new population.

In section 4, we apply our results to discriminate between three populations of kangaroos based on sex and eighteen continuous measurements (Andrews DF, Hertzberg AM, DATA: A Collection of Problems from Many Fields for the Student and Research Worker, 1985. New York: Springer-Verlag).

2 Estimation of linear regression models Longitudinal data

Jörg Breitung, Rémy Slama and Axel Werwatz

2.1 Introduction

2.1.1 Motivations

It has become common in economics and in epidemiology to make studies in which subjects are followed over time (longitudinal data) or the observations are structured into groups sharing common unmeasured characteristics (hierarchical data). These studies may be more informative than simple cross-sectional data, but they need an appropriate statistical modeling, since the 'classical' regression models of the GLM family (Fahrmeir and Tutz, 1994) assume statistical independence between the data, which is not the case when the data are grouped or when some subjects contribute for two or more observations.

Hierarchical regression models allow to analyze such surveys. Their main difference with classical regression models consist in the introduction of a group specific variable that is constant within each group, but differs between groups. This variable can be either a fixed-effect (classical) variable, or a random effect variable. From a practical point of view, the fixed or random-effect variable may be regarded as allowing to a certain extent to take into account unobserved characteristics (genetic, behavioral, ...) shared by the observations belonging to a given group. From a statistical point a view, the introduction of the group-level variable 'absorbs' the correlation between the different observations of a given group, and allow the residuals of the model to remain uncorrelated.

We will present here the fixed- and random-effect models in the case of linear regression, and their implementation in XploRe. A particular attention will be given to the case of unbalanced longitudinal data, that is studies in which the number of observations per group is not the same for all groups. This is an important issue in that the implementation of models adapted to such data needs some adaptation compared to the balanced case and since the elimination of the groups with only one observation could yield selection biases. The models will be applied to an epidemiological study about reproductive health, where women were asked to describe the birth of weight of all their children born in a given calendar period.

2.1.2 Example

```
{panfix,panrand,panopt} = seq(seqlist1,seqlist2)
```
estimates linear panel data models

We want to describe the influence of tobacco consumption by the woman during her pregnancy on the birth weight of her baby. We conducted a study among a cross-sectional sample of $N = 1,037$ women living in 2 French areas and asked them to describe retrospectively all their pregnancies leading to a livebirth during the 15 years before interview, and, for each baby, to indicate the number of cigarettes smoked during the first term of pregnancy (exposure, noted x).

The influence of cigarette exposure could be studied by linear regression on birth weight (dependent variable, noted y). Given the amount of information lying in the other pregnancies and the cost of data collection, it is tempting to try to make use of all the available information. Using all the pregnancies ($N\bar{T}$, where $\bar{T}$ is the mean number of pregnancies per woman) in a linear regression model may not be appropriate, since the estimation of the linear regression model

$$y_j = \mu + x_j^\top \beta + u_j, \quad j = 1, \ldots, N\bar{T} \tag{2.1}$$

by the ordinary least squares (OLS) method makes the assumption that the residuals u_j are independent random variables. Indeed, there may be correlation between the birth weights of a the children of a given woman, since the

corresponding pregnancies may have been influenced by the genetic characteristics of the woman and some occupational or behavioral exposures remaining constant over the woman's reproductive life.

A possible way to cope with this correlation is to use hierarchical modelling. The 2-levels structure of the data (woman or *group* level, and pregnancy or *observation* level) must be made explicit in the model. If we index by i the woman and t the pregnancies of a given woman, then a hierarchical linear regression model for our data can be written:

$$y_{it} = \mu + x_{it}^{\top}\beta + \alpha_i + u_{it}, \quad i = 1, \ldots, N \quad t = 1, \ldots, T_i \tag{2.2}$$

where y_{it} is the birth weight of the pregnancy number t of woman i. The number of pregnancies described by the woman i is a value T_i between 1 and say 12 and can vary between women. Of course, x_{it}, the mean number of cigarettes smoked daily, can vary between women and between the various pregnancies of a woman. The main difference with model *linearreg* is that the model now contains the α_i variables ($i = 1, \ldots, N$) defined at the group (or woman) level.

The random-effect model can be estimated using the command

```
p=panrand(id,y,x)
```

where id stands for i and x for the independent variables. This allows to obtain the output shown in Table 2.4.2

Table 2.1: Tobacco consumption by the woman during the first term of pregnancy

Parameters	Estimate	SE	t-value	p-value
Tobacco	-9.8389	2.988	-3.292	0.001
Sex(Girl=1)	-157.22	18.18	-8.650	0.000
(...)Constant	3258.1	83.48	39.027	0.000
St.dev of a(i): 330.16			St.dev of e(i,t): 314.72	
R2(without): 0.2426				

The model was adjusted for other variables, like duration of pregnancy, mother's

alcohol consumption, sex of the baby, which are not shown in this output. The random-effect model estimates that, on average, tobacco consumption by the woman during the first term of pregnancy is associated with a decrease by 9.8 grams (95% confidence interval: $[-15.7; -4.0]$) of the birth weight of the baby per cigarette smoked daily.

2.1.3 Definitions and notations

The cross-section unit (e.g. individual, household, hospital, cluster etc.) will be denoted group and be indexed by i, whereas t indexes the different observations of the group i. The t index can correspond to time, if a subject is followed and observed at several occasions like in a cohort study, but it may also be a mere identifying variable, for instance in the case of therapeutical trial about a new drug, realized in several hospitals. In this case, it may be appropriate to use a hierarchical model, with i standing for the hospital, and t indexing each subject within the hospital.

We will use indifferently the terms of *panel* or preferably *longitudinal* data to design data sets with a hierarchical structure, whatever the sampling method (cross-sectional or cohort surveys) although the term of panel study is sometimes used exclusively in the case of cohort studies. The data set is said unbalanced when the number of observations T_i is not the same for all groups, $i = 1, 2, \ldots, N$, and balanced when $T_i = T$ for all i. The explained quantitative variable will be denoted y_i, which is a vector of dimension T_i. The average number of observations is denoted as $\bar{T} = N^{-1} \sum_{i=1}^{N} T_i$.

In the first section, we will present the theoretical bases of the fixed and random effect models, and give explicit formulas for the parameters and options of the `panfix` and `panrand` quantlets. This technical section can however be skipped by the readers non-familiar to statistical notations.

2.2 Theoretical aspects

2.2.1 The fixed-effect model

The model

For individual (or groups) i at time t we have

$$y_{it} = \alpha_i + x_{it}^\top \beta + u_{it}, \quad i = 1, \ldots, N, \quad t = 1, \ldots, T \tag{2.3}$$

This model is also called the *analysis of covariance model*. It is a *fixed effects* model in the sense that the individual specific intercepts α_i are assumed to be non-stochastic. The vector of explanatory variables x_{it} is assumed independent of the errors u_{it} for all i and t. The choice of the fixed-effect model (as opposed to a random effect model) implies that statistical inference is conditional on the individual effects α_i.

Writing (2.3) for each observation gives

$$\underbrace{\begin{bmatrix} \boldsymbol{y}_1 \\ \boldsymbol{y}_2 \\ \vdots \\ \boldsymbol{y}_N \end{bmatrix}}_{NT \times k} = \underbrace{\begin{bmatrix} \mathbf{1}_{T_1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_{T_2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \cdots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{1}_{T_N} \end{bmatrix}}_{NT \times N} \underbrace{\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{bmatrix}}_{N \times 1} + \underbrace{\begin{bmatrix} \boldsymbol{x}_1^\top \\ \boldsymbol{x}_2^\top \\ \vdots \\ \boldsymbol{x}_N^\top \end{bmatrix}}_{NT \times k} \boldsymbol{\beta} + \underbrace{\begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \\ \vdots \\ \boldsymbol{u}_N \end{bmatrix}}_{NT \times 1} \tag{2.4}$$

or, in matrix notation,

$$y = D_N \alpha + X\beta + u. \tag{2.5}$$

Parameters estimation

The matrix D_N can be seen as a matrix of N dummy variables. Therefore, the least-squares estimation of (2.3) is often called "least-squares dummy-variables estimator" (Hsiao, 1986). The coefficient estimates results as:

$$\widehat{\boldsymbol{\beta}}_{WG} = \left(\boldsymbol{X}^\top \boldsymbol{W}_n \boldsymbol{X}\right)^{-1} \boldsymbol{X}^\top \boldsymbol{W}_n \boldsymbol{y} \tag{2.6}$$

$$\widehat{\boldsymbol{\alpha}} = (\boldsymbol{D}_N^\top \boldsymbol{D}_N)^{-1} \boldsymbol{D}_N^\top (\boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}_{WG}) \tag{2.7}$$

$$= \begin{bmatrix} T_1^{-1} \sum_{t=1}^{T} (y_{1t} - \boldsymbol{x}_{1t}^\top \widehat{\boldsymbol{\beta}}_{WG}) \\ \vdots \\ T_N^{-1} \sum_{t=1}^{T} (y_{Nt} - \boldsymbol{x}_{Nt}^\top \widehat{\boldsymbol{\beta}}_{WG}) \end{bmatrix} \tag{2.8}$$

where

$$\boldsymbol{W}_n = \boldsymbol{I}_{NT} - \boldsymbol{D}_N (\boldsymbol{D}_N^\top \boldsymbol{D}_N)^{-1} \boldsymbol{D}_N^\top$$

transforms the regressors to the deviation-from-the-sample-means form, where $\boldsymbol{l}_T$ is the unit vector of size T. Accordingly, β_{WG} can be written as the "Within-Group" (WG) estimator:

$$\widehat{\boldsymbol{\beta}}_{WG} = \left[\sum_{i=1}^{N} \sum_{t=1}^{T} (\boldsymbol{x}_{it} - \bar{\boldsymbol{x}}_i)(\boldsymbol{x}_{it} - \bar{\boldsymbol{x}}_i)^\top \right]^{-1} \left[\sum_{i=1}^{N} \sum_{t=1}^{T} (\boldsymbol{x}_{it} - \bar{\boldsymbol{x}}_i)(y_{it} - \bar{y}_i) \right] , \tag{2.9}$$

where the individual means are defined as

$$\bar{y}_i = \frac{1}{T_i} \sum_{t=1}^{T_i} y_{it} \ , \quad \bar{\boldsymbol{x}}_i = \frac{1}{T_i} \sum_{t=1}^{T_i} \boldsymbol{x}_{it}.$$

To estimate the average of the individual effects $\bar{\alpha} = N^{-1} \sum_{i=1}^{N} \alpha_i$, the individual means can be corrected by the sample means $\bar{y} = (N\bar{T})^{-1} \sum_{i=1}^{N} \sum_{t=1}^{T_i} y_{it}$ and $\bar{\boldsymbol{x}}$ is defined accordingly. The least-squares estimates of β and $\bar{\alpha}$ is obtained from the equation

$$y_{it} - \bar{y}_i + \bar{y} = \bar{\alpha} + (\boldsymbol{x}_{it} - \bar{\boldsymbol{x}}_i + \bar{\boldsymbol{x}}_i)^\top \boldsymbol{\beta} + \tilde{u}_{it} \ . \tag{2.10}$$

It is important to notice, from (2.9), that cross section units with only one observation do not contribute to the estimation $\widehat{\boldsymbol{\beta}}$ of the parameters associated to the explaining variables $\boldsymbol{x}$; that is, the same estimate results if these cross

section units would be excluded from the data set. The groups with $T_i = 1$ only play a role in the estimation of the mean intercept.

Adequation of the model to the data

In complement to the parameter estimation, the degree of explanation of the model and the variance of the error terms can be estimated. It is also possible to test if the introduction of a group-specific variable makes sense with the data used, by means of a F-statistic test presented below.
There are two different possibilities to compute the degree of explanation R^2. First, one may be interested in the fraction of the variance that is explained by the explanatory variables comprised in x_{it}. In this case R^2 is computed as the squared correlation between y_{it} and $\boldsymbol{x}_{it}^\top \widehat{\boldsymbol{\beta}}_{WG}$. On the other hand, one may be interested to assess the goodness of fit when the set of regressors is enhanced by the set of individual specific dummy variables. Accordingly, the R^2 is computed as the squared correlation between y_{it} and $\boldsymbol{x}_{it}^\top \widehat{\boldsymbol{\beta}}_{WG} + \widehat{\alpha}_i$. In the output table of the `panfix` quantlet, the former goodness-of-fit statistic is referred to as "R^2 (without effects)", whereas the latter is indicated by "R^2 (with effects)".

In practical applications the individual specific constants may have similar size so that it is preferable to specify the model with the same constant for all groups. This assumption can be tested with an F statistic for the hypothesis $\alpha_1 = \alpha_2 = \cdots = \alpha_N$. In the output table of the `panfix` pantlet the p-value of this test statistic is presented in the line "F(no eff.)".

In order to assess the importance of the individual specific effects, their "variances" are estimated. Literally, it does not make much sense to compute a variance of α_i if we assume that these constants are deterministic. Nevertheless, the variance of α_i is a measure of the variability of the individual effect and can be compared to the variance of the error u_{it}. The formula for estimating the variance of the fixed effects is similar to the computation of variances in the random-effects model. However, the residuals are computed using the within-group estimator $\widehat{\boldsymbol{\beta}}_{WG}$ (Amemiya, 1981).

Options for the fixed-effects model

a) Robust standard errors
Arelano and Bond (1987) suggests an estimator of the standard errors for $\widehat{\boldsymbol{\beta}}_{WG}$

that is robust to heteroskedastic and autocorrelated errors u_{it}:

$$\widetilde{Var}(\widehat{\boldsymbol{\beta}}_{WG}) = \left(\sum_{i=1}^{N} \widetilde{\boldsymbol{X}}_i^\top \widetilde{\boldsymbol{X}}_i\right)^{-1} \left(\sum_{i=1}^{N} \widetilde{\boldsymbol{X}}_i^\top \widehat{\boldsymbol{u}}_i \widehat{\boldsymbol{u}}_i^\top \widetilde{\boldsymbol{X}}_i\right) \left(\sum_{i=1}^{N} \widetilde{\boldsymbol{X}}_i^\top \widetilde{\boldsymbol{X}}_i\right)^{-1},$$

where

$$\widetilde{\boldsymbol{X}}_i = \begin{bmatrix} \boldsymbol{x}_{i1}^\top - \bar{\boldsymbol{x}}_i^\top \\ \boldsymbol{x}_{i2}^\top - \bar{\boldsymbol{x}}_i^\top \\ \vdots \\ \boldsymbol{x}_{iT}^\top - \bar{\boldsymbol{x}}_i^\top \end{bmatrix} \quad \text{and} \quad \tilde{\boldsymbol{u}}_i = \begin{bmatrix} y_{i1} - \bar{y}_i - (\boldsymbol{x}_{i1} - \bar{\boldsymbol{x}}_i)^\top \widehat{\boldsymbol{\beta}}_{WG} \\ y_{i2} - \bar{y}_i - (\boldsymbol{x}_{i2} - \bar{\boldsymbol{x}}_i)^\top \widehat{\boldsymbol{\beta}}_{WG} \\ \vdots \\ y_{iT} - \bar{y}_i - (\boldsymbol{x}_{iT} - \bar{\boldsymbol{x}}_i)^\top \widehat{\boldsymbol{\beta}}_{WG} \end{bmatrix}.$$

It should be noted that the estimation of this covariance matrix requires two steps. In the first step the within-group estimator is used to estimate β. In the second step, the covariance matrix is computed by using the residuals of the fixed-effects model. Therefore, the computation time is roughly doubled.

b) Test for autocorrelation

The test for autocorrelation tests the null hypothesis: $H_0 : E(u_{it}u_{i,t-1}) = 0$. Since the residuals of the estimated fixed-effect model are correlated, a test for autocorrelation has to adjust for a correlation that is due to the estimated individual effect. Define

$$\tilde{u}_{i,t-1} = y_{i,t-1} - \boldsymbol{x}_{i,t-1}^\top \widehat{\boldsymbol{\beta}}_{WG} - (T-1)^{-1} \sum_{s=1}^{T-1} y_{is} - \boldsymbol{x}_{is}^\top \widehat{\boldsymbol{\beta}}_{WG}.$$

It is not difficult to verify that under the null hypothesis

$$E\Big\{(y_{it} - \boldsymbol{x}_{it}^\top \widehat{\boldsymbol{\beta}}_{WG})\tilde{u}_{i,t-1}\Big\} = -\sigma_u^2/(T-1),$$

where $\sigma_u^2 = E(u_{it}^2)$. The test statistic is therefore constructed as

$$\tilde{\rho} = \frac{\sum_{i=1}^{N}\sum_{t=2}^{T}\Big[(y_{it} - \boldsymbol{x}_{it}^\top \widehat{\boldsymbol{\beta}}_{WG})\tilde{u}_{i,t-1}/\widehat{\sigma}_u^2 + 1/(T-1)\Big]}{\sqrt{\sum_{i=1}^{N}\sum_{t=2}^{T}\tilde{u}_{i,t-1}^2}}.$$

Under the null hypothesis, the limiting distribution has a standard normal limiting distribution.

c) Estimates of the individual effects
The mean intercept is estimated by:

$$\widehat{\mu} = \bar{y} - \widehat{\boldsymbol{\beta}}^{\top} \bar{\boldsymbol{x}}. \tag{2.11}$$

It is also possible to estimate the group variables α_i:

$$\widehat{\alpha}_i = \bar{y}_i - \widehat{\mu} - \widehat{\boldsymbol{\beta}}^{\top} \bar{\boldsymbol{x}}_{\boldsymbol{i}}. \tag{2.12}$$

2.2.2 The random effects model

The model

For the random effects model it is assumed that the individual specific intercept α_i in the model

$$y_{it} = \boldsymbol{x}_{it}^\top \boldsymbol{\beta} + \alpha_i + u_{it}, \quad i = 1, \ldots, N, \quad t = 1, \ldots, T \tag{2.13}$$

is a random variable with $E(\alpha_i) = 0$ and $E(\alpha_i^2) = \sigma_\alpha^2$. Furthermore we assume that

$$\begin{array}{ll} E(\alpha_i u_{it}) = 0 & \text{for all } i, t, \\ E(\alpha_i \boldsymbol{x}_{it}) = \boldsymbol{0} & \text{for all } i, t. \end{array}$$

In general the vector $\boldsymbol{x}_{it}$ includes a constant term.

The composed error term is written as $v_{it} = \alpha_i + u_{it}$ and the model assumptions imply that the vector $\boldsymbol{v}_i = [v_{i1}, \ldots, v_{iT}]^\top$ has the covariance matrix

$$E(\boldsymbol{v}_i \boldsymbol{v}_i^\top) = \Psi \ .$$

The model (2.13) can be efficiently estimated by using the GLS estimator

$$\widehat{\boldsymbol{\beta}}_{GLS} = \left(\sum_{i=1}^{N} \boldsymbol{X}_i^\top \Psi^{-1} \boldsymbol{X}_i \right)^{-1} \left(\sum_{i=1}^{N} \boldsymbol{X}_i^\top \Psi^{-1} \boldsymbol{y}_i \right) \ , \tag{2.14}$$

where $\boldsymbol{X}_i = [\boldsymbol{x}_{i1}, \ldots, \boldsymbol{x}_{iT}]^\top$ and $\boldsymbol{y}_i = [y_{i1}, \ldots, y_{iT}]^\top$. This estimator is equivalent to a least-squares estimator of the transformed model

$$y_{it} - \psi \bar{y}_i = (\boldsymbol{x}_{it} - \psi \bar{\boldsymbol{x}}_i)^\top \boldsymbol{\beta} + e_{it} \ , \tag{2.15}$$

where

$$\psi = \sqrt{\frac{\sigma_u^2}{\sigma_u^2 + T\sigma_\alpha^2}} \tag{2.16}$$

and $e_{it} = v_{it} - \psi \bar{v}_i$.

In general, the variances σ_u^2 and σ_α^2 are unknown and must be replaced by estimates. To this end several different estimators were suggested (Baltagi, 1995). The `panrand` quantlet employs the estimator suggested by Swamy and Arora (1972), which is based on two different regressions. First, the model is estimated by using the within-group estimator. The estimated error variance

(corrected by the degrees of freedom) is an unbiased estimator for σ_u^2. The second regression is based on the individual means of the data

$$\bar{y}_i = \bar{\boldsymbol{x}}_i^\top \beta + \bar{v}_i \ . \tag{2.17}$$

Since $E(\bar{v}_i^2) = \sigma_\alpha^2 + \sigma_u^2/T$, an estimator for σ_α^2 is obtained from the estimated residual variance of (2.17). Let $\hat{\sigma}_1^2$ denote the estimated residual variance of the between-group regression (2.17), which results from dividing the residual sum of squares by $(N-K-1)$. The estimated variance of the individual effect results as $\hat{\sigma}_\alpha^2 = (\hat{\sigma}_1 - \hat{\sigma}_u^2)/T$. A serious practical problem is that the resulting estimator of $\hat{\sigma}_\alpha^2$ may become negative. In this case $\hat{\sigma}_\alpha^2$ is set to zero.

Table 2.2: Nama table

id	y	x_1	x_2
1	3409	38	0
1	3755	41	1
2	1900	32	1
3	4200	41	1
3	4050	40	0
3	4300	41	1
...	...	...	...
100	3000	39	0
100	2850	39	1

2.3 Computing fixed and random-effect models

2.3.1 Data preparation

Suppose we want to regress a quantitative variable y over explanatory variables noted x. The variable indexing the group will be noted id. This is for instance how the data set should look like in the case of two x variables:

If you have a balanced data set (same number of observations per group) sorted by group, then the id variable is not necessary. You will have to give the number of observations per subject instead of the id vector, that XploRe will then build for you.

2.3.2 Fixed and random-effect linear regression

The fixed-effect linear regression model can be estimated using the following syntax:

```
library("metrics")
p=panfix(id,y,x{,opt})
```

The random-effect linear regression model can be estimated using the following syntax:

```
library("metrics")
p=panrand(id,y,x{,opt})
```

2.3.3 Options for `panfix`

The options must be defined by the `panopt` quantlet according to the syntax:

```
opt=panopt(optname,optvalue)
```

where `optname` is the name of the option, and `optvalue` the value associated to the option. The name of the option has to be given as a string. You may define several options at the same time according to the following syntax:

```
opt=panopt(optname1,optvalue1,optname2,optvalue2,optname3,optvalue3)
```

The following options can be defined:

`alpha:` If equal to 1, asks for the individual effect parameter to be estimated and stored. The estimation is done assuming that the sum of all alpha parameters is zero.

`autoco:` If equal to 1, an autocorrelation test is performed (only if the number of observations is at least 2 for each group). Default is no test performed.

`ci:` If this parameter is set to the value pval, then the confidence intervals will be given at the level (100-pval)%. By default, no ci are given.

`notab:` If this parameter is set to 1, then no table of results is displayed.

`robust:` The robust estimates of variance given in (Arelano and Bond, 1987) are used. These should be more valid than the classical variance estimates in the case of heteroscedasticity. Default is the standard variance estimates.

`xlabel:` Label of the explanatory variables, to make the output table more explicit. This option must be given as a vertical array of the k strings corresponding to the labels (constant term excluded). Maximum label length is 11 characters. $(k \times 1)$ vector.

For example, if $\boldsymbol{x}$ is a vector of 2 columns containing the independent variables tobacco and alcohol consumption, you may type:

```
lab="tobacco"|"alcohol"
opt=panopt("xlabel",lab)
p=panfix(id,y,x,opt)
```

In the output table, the parameters associated to the first and second variables will be labelled by the indicated names. Unspecified options will be set at their default value, and the order in which the options are given is not important.

2.3.4 Options for `panrand`

The options must be defined by the `panopt` quantlet according to the syntax:

```
opt=panopt(optname,optvalue)
```

where `optname` is the name of the option, and `optvalue` the value associated to the option.

The following options can be defined:

`opt.shf:` Allows you to see the various steps of the estimation procedure.

`opt.xlabel:` Label of the explanatory variables, to make the output table more explicit. This option must be given as a vertical array of the k strings corresponding to the labels (constant term excluded). Maximum label length is 11 characters and $(k \times 1)$ vector.

2.4 Application

In this section, we illustrate the use of the `panfix` and `panrand` quantlets presented above, with some estimations based on real data.

Table 2.3: Nama table

Variable	Mean	Std Dev	$5-95^{th}$ percentiles
Birth weight (g)	3409	510	2610-4250
Gestational length (days)	283	11.8	261-294
Mother's age (years)	27.2	4.4	20.1-35.1
Proportion of parous women	0.60		
Sex of the offspring (proportion of boys)	0.50		

2.4.1 Presentation of the data

The data come from an epidemiologic study about human reproductive life events. Briefly, a cross-sectional sample of 1089 women from Bretagne and Normandie were questioned during spring 2000 about the birth weight of all their children born between 1985 and 2000. We present here the association between the birth weight (dependent variable), the gestational length, the age, and the parity (previous history of livebirth, no/yes) of the mother (independent variables). There was a total of 1963 births in the study period (1.8 pregnancy per woman) and the data can be considered as longitudinal data with a hierarchical structure, the woman being the first level, and the pregnancy the second level. The use of fixed or random effect models allows to take into account all the pregnancies who took place in the study period described by the woman. In such epidemiological studies about human reproduction, the exclusion of couples with only one pregnancy may give rise to selection bias, since the couples with only one pregnancy are more likely than those with two or more pregnancies to have difficulties in conceiving. Here is a brief description of the data set:

2.4.2 Results

First, we will describe briefly our data using the `panstat` quantlet:

```
library("metrics")
z=read("birthweight.dat")
panstat(z[,1:cols(z)])
```

The first column of `z` contains the identified variable, whereas the next columns contain the dependent variables, and then the independent variables. If the panel is balanced and sorted by group, the first argument `id` can be replaced by a scalar indicating the number of observations per group. We obtain the following output:

Table 2.4: Nama table

	Minimum	Maximum	Mean	Within Var.%	Std.Error
Variable 1	750	5300	3409	23.8	509.6
Variable 2	-98	21	-5.715	27.56	11.76
Variable 3	14.37	45.71	27.18	26.77	4.366
Variable 4	0	1	0.595	66.82	0.491
Variable 5	0	1	0.5028	45.7	0.5001

The column `Within Var.%` gives the value of the variance of the residuals of the withing-group estimator, divided by the overall variance.

We can then estimate a fixed-effect regression model. The program:

```
z=read("birthweight.dat")
p=panfix(z[1,],z[2,],z[3:6])
```

gives the following estimates:

Thus, on average, an increase in 1 day of the duration of pregnancy was associated with a gain of weight of 18.4 grams `(beta[1])`, and girls are 145 g lighter than boys at birth `(beta[4])`, with a 95% confidence interval of [-186;-103] g. Moreover, women who already had a child have a tendency to give birth to heavier babies (77 g on average). There is a non-significant tendency to an increase in birth weight with mother's age.

The R^2 value of 0.22 indicates that only a small fraction of the variability of the data is explained by the model, and that other variables should be included (for instance height and weight of the mother before pregnancy, information on health,...).

Table 2.5: Nama table

Parameters	Estimate	SE	t-value	p-value
beta[1]	18.548	1.17	15.8908	0.0000
beta[2]	7.964	4.61	1.7263	0.0843
beta[3]	75.239	25.97	2.8970	0.0038
beta[4]	-144.51	21.27	-6.7931	0.0000
Constant	3326.1	115.3	28.8350	0.0000
St.dev of a(i): 321.47			St.dev of e(i,t):318.47	
Log-Likelihood: 22627.617			R2(without) : 0.2203	
F(no eff.) p-val: 0.0000			R2(with eff) : 0.8272	

In this case, there are some groups with only one observation (cf. output above); we cannot therefore perform an autocorrelation-test, nor obtain robust confidence-intervals estimates. In the case of a data set with all groups having at least 2 observations, this can be obtained by the following syntax:

```
z=read("birthweight_2.dat")
opt=panopt("robust",1,"autoco",1,"ci",10)
p=panfix(z[1,],z[2,],z[3:6],opt)
```

For the data, the a-priori choice between the fixed-effect and the random-effect model would be the random-effect model, because the included women were randomly selected from two French rural areas, and we wish to infer the model estimates on the women who conceived between 1985 and 2000 in the whole area.

We can obtain the random-effect model estimates by the program:

```
z=read("birthweight.dat")
p=panrand(z[1,],z[2,],z[3:6,],opt)
```

which gives the following estimates:

On the whole, these estimates are consistent with those of the fixed-effect model. You can notice that for variable [2] (mother's age), the estimates from the two models differ (7.8 with a standard error of 4.6 for the fixed-effect model, and 4.6 with a standard error of 2.6 for the random effect model). In such a

Table 2.6: Nama table

Parameters	Estimate	SE	t-value	p-value	95% CI	
beta[1]	18.927	0.8286	22.844	0.000	17.3	20.55
beta[2]	4.5912	2.638	1.740	0.082	-0.58	9.76
beta[3]	88.389	18.89	4.678	0.000	51.36	125.4
beta[4]	-152.53	17.46	-8.735	0.000	-186.8	-118.3
Constant	3413.3	68.94	49.509	0.000	3278.0	3548.0
St.dev of a(i): 337.9			St.dev of e(i,t): 312.19			
R2(without): 0.2206						

case, where the number of observations is small for many units, it is not rare that both models yield different parameter estimates.

Bibliography

Amemiya, T. (1981). Qualitative response models: A survey, *Journal of Economic Literature* **19**: 1483-1536.

Arellano, M. and Bond, S.R. (1987). Computing robust standard errors for within-groups estimators, *Oxford Bulletin of Economics and Statistics* **49**: 431-434.

Baltagi B.H., (1995). *Econometrics analysis of panel data*, Wiley, Chichester.

Breitung J., (2000). *XploRe Learning Guide*, Springer, Berlin.

Fahrmeir and Tutz, (1994). *Multivariate Statistical Modelling Based on Generalized Linear Models*, Springer Series in Statistics.

Hsiao, (1986). *Analysis of Panel data, Cambridge University Press*, Cambridge.

Swami, P.A. and Arora, S.S., (1972). The exact finite sample properties of the estimators ofcoefficients in the error components regression models, *Econometrica* **40**: 261-275.

3 Conditional quantiles with functional covariates: an application to ozone pollution forecasting

Hervé Cardot, *Christophe Crambes*, *Pascal Sarda*

Expected length of the paper: 20 pages

3.1 Abstract

This work deals with the study of pollution data with the aim of forecasting the Ozone pollution in the city of Toulouse. The ORAMIP ("Observatoire Régional de l'Air en Midi-Pyrénées") provided data which are hourly measures of pollutants as well as hourly measurements of meteorological covariates. The nature of these data allows us to deal with them as curves, known in some discretization points, which are called *functional data* in the literature.

Our goal is then to give a prediction of the maximum of Ozone one day knowing one or several of these functional variables the day before. To do this, we consider two models. The first one bases the prediction on the conditional mean, and the second one on the conditional median. In each case, we have functional covariates and we introduce a spline estimator of the functional coefficient which minimizes a least square type (in the first model) or a least absolute value type (in the second model) penalized criterion. The minimization criterion corresponding to the first model has an explicit solution, contrary to the second one which is solved by an iterative weighted least square algorithm. These two approaches are illustrated with the ORAMIP data and we make a comparison of the prediction of these models with different covariates.

4 Nonparametric functional methods: new tools for chemiometrical analysis

Frédéric Ferraty, *Aldo Goia*, *Philippe Vieu*

Expected length of the paper: 20 pages

4.1 Abstract

Spectrometric is an usual technique for chemiometric analysis. Spectrometric data are consisting in continuous spectra of some components to be analysed. From a statistical point of view these data are clearly of functional (continuous) nature. We will center our purpose around a food industry spectrometric real data set, which is a set of absorbances spectra observed on several pieces of meat. The aim of this contribution is to show how the recent nonparametric methodology for functional data may provide interesting results in this setting. Concretely, we will present two functional nonparametric methods, corresponding to two different statistical problem. The first one is the problem of predicting some real response variable (percentage of fatness) corresponding to some given continuous absorbance spectra, and we will decsribe a Nonparametric Functional Regression method. The second one will be the question of discrimating these spectra according to some categorical response, and we will describe a Nonparametric Curves Discrimination method.

It is worth being noted that, even if our presentation will be centered aroubnd this spectrometric food industry example, both the methodology and the programs will be presented in a general way. This will allow for posible application of the proposed methods in many other fields of applied statistics in which functional data have to be treated (environmetrics, econometrics, biometrics, ...).

5 Polychotomous regression: application to landcover prediction

Frédéric Ferraty, *Martin Paegelow*, *Pascal Sarda*

Expected length of the paper: 20 pages

5.1 Abstract

The aim of this work is to predict landcover for a given area: analyzing at first the evolution of landcover in the past one wants to produce a map of landcover in the future. The data analyzed comes from a mountainous area in the Pyrénées which name is Garrotxes. The size of the area is 8570 hectares divided in pixels: for each pixel we have the value for a categorical variable indicating the nature of vegetation for this pixel (with eight levels) and the values of environmental variables (slope, ...). We have three sets of such data for the years 1980, 1989 and 2000.

The problem is then to predict at time t and for each pixel a categorical response (the value of vegetation for this pixel) given both categorical and scalar predictors *i.e.* the nature of vegetation at time $t-1$ and the value of environmental variables in the neighbouring pixels. For this we use the *multiple logistic* (or *polychotomous*) regression model. To estimate the parameters of this (linear) model, we use a penalized log-likelihood estimator: penalization allows numerical stability of the solution whereas, for reasonable small values of the penalization parameters, it does not affect the value of the estimators. A Newton-Raphson algorithm is used to achieve the numerical mawimization of the penalized log-likelihood. The first step of the procedure consists in an estimation step based on the first two maps. The third map is then used to validate

the choice of the size of the neighbourhood and the value of the penalization parameter. Prediction of the map is done using the estimated parameters of the model obtained at the second step.

6 A kernel method in analysis of replicated micro-array experiments

Ali Gannoun, Benoît Liquet, Jérôme Saracco, Wolfgang Urfer

Expected length of the paper: around 20 pages

6.1 Abstract

Microarrays are part of a new class of biotechnologies which allow the monitoring of expression levels of thousands of genes simultaneously. In microarray data analysis, the comparison of gene expression profiles with respect to different conditions and the selection of biologically interesting genes are crucial tasks. Multivariate statistical methods have been applied to analyze these large data sets. To identify genes with altered expression under two experimental conditions, we describe in this chapter a new nonparametric statistical approach. Specifically, we propose estimating the distributions of a t-type statistic and its null statistic, using kernel methods. A comparison of these two distributions by means of a likelihood ratio test can identify genes with significantly changed expressions. A method for the calculation of the cut-off point and the acceptance region is also derived. This methodology is applied to a leukemia data set containing expression levels of 7129 genes. The corresponding results are compared to the traditional t-test and the normal mixture model.

7 Kernel Estimates of Hazard Functions for Biomedical Data Sets

Ivana Horová, *Jiří Zelinka*

Expected length of the paper: 20 pages

7.1 Abstract

The purpose of this chapter is to present a nonparametric method for censored samples. This is a common situation in survival analysis problem. We will use the model of random censorship where the data are censored from the right. This type of censorship is often met in many applications, especially in clinical research or in life testing of complex technical systems. In summarizing the survival data there are two functions of central interest, namely the survival function and the hazard function.

The well-known product-limit estimator of the survival function was proposed by Kaplan and Meier. A single sample of survival data may be also summarized through the hazard function which shows the dependence of the instantaneous risk of death time. We focus on nonparametric estimates of the hazard functions and their derivatives.

Among nonparametric estimates methods of kernel estimates represent one of the most effective methods. These methods are simple enough which makes the numerical calculation easy and fast and the possibilities for mathematical analysis of properties of obtained estimates are very good, too.

The kernel estimates depend on a bandwidth, a kernel and on order of a kernel. The procedure of choosing these three parameters is dealing with. As far as the

biomedical application is concerned the attention will be paid not only to the estimate of hazard function but also to the estimate of the second derivative of this function since the dynamics of the underlying cause is often of a great interest. For this reason the attention is also paid to the detection of the points where the most rapid changes of the hazard functions occur.

The aforementioned method is applied to the real biomedical data sets. The method of kernel estimate seems to be suited as an exploratory tool for analyzing in hazard rates. Moreover this method offers the consistent estimate and graphical representation can give suitable explanation of the detection of points of the most rapid change as well.

8 Partially Linear Models

Wolfgang Härdle, *Hua Liang*

8.1 Introduction

Partially linear models (PLM) are regression models in which the response depends on some covariates linearly but on other covariates nonparametrically. PLMs generalize standard linear regression techniques and are special cases of additive models. This chapter covers the basic results and explains how PLMs are applied in the biometric practice. More specifically, we are mainly concerned with least squares estimators of the linear parameter while the nonparametric part is estimated by e.g. kernel regression, spline approximation, piecewise polynomial and local polynomial techniques. When the model is heteroscedastic, the variance functions are approximated by weighted least squares estimators. Numerous examples illustrate the implementation in practice.

`plmest = plmk(x,t,y,h)`
estimates the parameters with kernel regression

`plmest = plmlp(x,t,y,h,p)`
estimates the parameters with p-order local polynomial

`plmest = plmp(x,t,y,m,mn)`
estimates the parameters with piecewise polynomial approximation

`plmest = plmls(x,t,y,m)`
estimates the parameters with least squares spline

`plmest = plmhetexog(x,t,y,w.h.h1)`
estimates the parameters when the variance is a function of exogenous variables

`plmest = plmhett(x,t,y,h,h1)`
estimates the parameters when the variance is an unknown function of the nonparametric variable

`plmest = plmhetmean(mn,x,t,y,h)`
estimates the parameters when the variance is an unknown function of the mean

Partially linear models (PLM) are defined by

$$Y = X^\top \beta + g(T) + \varepsilon, \tag{8.1}$$

where X and T are d-dimensional and scalar regressors, β is a vector of unknown parameters, $g(\cdot)$ an unknown smooth function and ε an error term with mean zero conditional on X and T.

The PLM is a special form of the additive regression models (Hastie and Tibshrani, 1990) and (Stone, 1985), which allows easier interpretation of the effect of each variables and may be preferable to a completely nonparametric regression since the well-known reason "curse of dimensionality".

On the other hand, PLMs are more flexible than the standard linear models since they combine both parametric and nonparametric components.

Several methods have been proposed to consider PLM. Suppose n observations

(Engle, Granger, Rice and Weiss, 1986), (Heckman, 1986) and (Rice, 1986) used spline smoothing and defined estimators of β and g as the solution of

$$\arg\min_{\beta,g} \frac{1}{n}\sum_{i=1}^{n}\{Y_i - X_i{}^{\top}\beta - g(T_i)\}^2 + \lambda\int\{g''(u)\}^2 du. \tag{8.2}$$

Speckman (1988) estimated the nonparametric component by $\mathcal{W}\gamma$, where $\mathcal{W}$ is a $(n \times q)-$matrix of full rank and γ is an additional parameter. PLM may be rewritten in a matrix form

$$Y = X\beta + \mathcal{W}\gamma + \varepsilon. \tag{8.3}$$

The estimator of β based on (8.3) is

$$\widehat{\beta}_S = \{X^{\top}(I - P_{\mathcal{W}})X\}^{-1}\{X^{\top}(I - P_{\mathcal{W}})Y\}, \tag{8.4}$$

where $P_{\mathcal{W}} = \mathcal{W}(\mathcal{W}^{\top}\mathcal{W})^{-1}\mathcal{W}^{\top}$ is a projection matrix and $\mathbf{I}$ is a $d-$order identity matrix. Green, Jennison and Seheult (1985) proposed another class of estimates

$$\widehat{\beta}_{\mathrm{GJS}} = \{X^{\top}(I - \mathcal{W}_h)X)\}^{-1}\{X^{\top}(I - \mathcal{W}_h)Y)\}$$

by replacing $\mathcal{W}$ in (8.4) by another smoother operator $\mathcal{W}_h$. Chen (1988) proposed a piecewise polynomial to approximate nonparametric function and then derived the least squares estimator which is the same form as (8.4). Recently Härdle, Liang, and Gao (2000) have systematically summarized the up-to-date results about PLM.

No matter which regression method is used for the nonparametric part, the forms of the estimators of β may always be written as $\{X^{\top}(I-W)X\}^{-1}X^{\top}(I-W)Y$, where W is a projection operation. The estimators are asymptotically normal under appropriate assumptions.

The next section will be concerned with several nonparametric fit methods for $g(t)$ because of their popularity, beauty and importance in nonparametric statistics. In Section 8.4, a real data-set is investigated for illustrating the theories and techniques.

8.2 Estimation and Nonparametric Fits

As stated in the previous section, different ways to approximate the nonparametric part may give the corresponding estimators of β. The popular nonparametric methods includes kernel regression, local polynomial, piecewise polynomial and smoothing spline. Related works are referred to Wand and Jones (1995), Eubank (1988), and Fan and Gijbels (1996). Härdle (1990) gives an extensive discussion of various nonparametric statistical methods based on the kernel estimator. This section mainly mentions the estimation procedure for β when one adapts these nonparametric methods and explains how to use XploRe quantlets to calculate the estimates.

8.2.1 Kernel Regression

Let $K(\cdot)$ be a kernel function satisfying certain conditions and h_n be a bandwidth parameter. The weight function is defined as

$$\omega_{ni}(t) = K\Big(\frac{t - T_i}{h_n}\Big) \Big/ \sum_{j=1}^{n} K\Big(\frac{t - T_j}{h_n}\Big).$$

Let $g_n(t, \beta) = \sum_{i=1}^{n} \omega_{ni}(t)(Y_i - X_i^\top \beta)$ for a given β. Substitute $g_n(T_i, \beta)$ into (8.1) and use least square criterion. Then the least squares estimator of β is obtained as

$$\widehat{\beta}_{\mathrm{KR}} = (\widetilde{\mathbf{X}}^\top \widetilde{\mathbf{X}})^{-1} \widetilde{\mathbf{X}}^\top \widetilde{\mathbf{Y}},$$

where $\widetilde{\mathbf{X}}^\top = (\widetilde{X}_1, \ldots, \widetilde{X}_n)$ with $\widetilde{X}_j = X_j - \sum_{i=1}^{n} \omega_{ni}(T_j) X_i$ and $\widetilde{\mathbf{Y}}^\top = (\widetilde{Y}_1, \ldots, \widetilde{Y}_n)$ with $\widetilde{Y}_j = Y_j - \sum_{i=1}^{n} \omega_{ni}(T_j) Y_i$. The nonparametric part $g(t)$ is estimated by:

$$\widehat{g}_n(t) = \sum_{i=1}^{n} \omega_{ni}(t)(Y_i - X_i^\top \widehat{\beta}_{\mathrm{KR}}).$$

When $\varepsilon_1, \ldots, \varepsilon_n$ are identically distributed, their common variance σ^2 may be estimated by $\widehat{\sigma}_n^2 = (\widetilde{\mathbf{Y}} - \widetilde{\mathbf{X}}\widehat{\beta}_{\mathrm{KR}})^\top (\widetilde{\mathbf{Y}} - \widetilde{\mathbf{X}}\widehat{\beta}_{\mathrm{KR}})$.

The detailed discussions on asymptotic theories of these estimators are referred to Härdle, Liang, and Gao (2000) and Speckman (1988). A main result in literature on the estimator $\widehat{\beta}_{\mathrm{KR}}$ may be described as follows.

THEOREM 8.1 *Suppose (i)* $\sup_{0\leq t\leq 1} E(\|X\|^3|t) < \infty$ *and* $\Sigma = Cov\{X - E(X|T)\}$ *is a positive definite matrix.*
(ii) $g(t)$ *and* $E(x_{ij}|t)$ *are Lipschitz continuous; and (iii) the bandwidth* $h \approx \lambda n^{-1/5}$ *for some* $0 < \lambda < \infty$. *Then*

$$\sqrt{n}(\widehat{\beta}_{\mathrm{KR}} - \beta) \xrightarrow{\mathcal{L}} N(0, \sigma^2\Sigma^{-1}).$$

In XploRe the quantlet `plmk` calculates the estimates $\widehat{\beta}_{\mathrm{KR}}$, $\widehat{\sigma}_n^2$ and $\widehat{g}_n(t)$. Its syntax is the following:

```
plmest=plmk(x,t,y,h)
```

Input parameters are

`x` : the linear regressors,

`t` : represents the non-linear regressors,

`y` : the response, and

`h` : determines the bandwidth.

Output parameters are

`plmest.hbeat` : estimate the parameter of X,

`plmest.hsigma` : estimate the variance of the error, and

`plmest.hg` : estimate the nonparametric part.

We now give an example of XploRe code to generate a sample from the PLM model, and then show the calculation results.

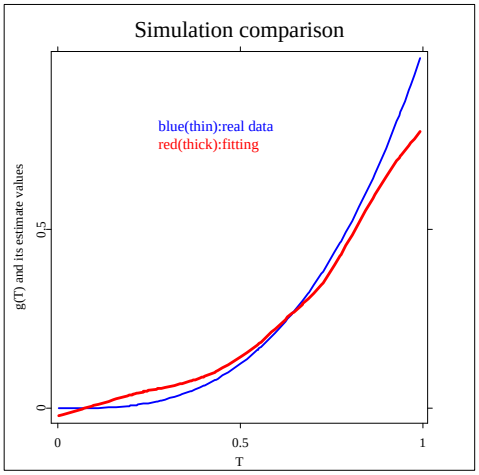

Figure 8.1: The simulation results for nonparametric function via quantlet `plmk`. Thin line: real data; Thick line: the fitting.

`XCSplm01.xpl`

8.2.2 Local Polynomial

The kernel regression (or local constant) can be improved by using local linear, more generally, local polynomial smoothers since they have appealing asymptotic bias and variance terms that are not adversely affected at the boundary (Fan and Gijbels, 1996)., see

Suppose that the $(p+1)$-th derivative of $g(t)$ at the point t_0 exists. We then approximate the unknown regression function $g(t)$ locally by a polynomial of order p. A Taylor expansion gives, for t in a neighborhood of t_0,

$$\begin{aligned} g(t) &\approx g(t_0) + g'(t_0)(t-t_0) + \frac{g^{(2)}(t_0)}{2!}(t-t_0)^2 + \cdots + \frac{g^{(p)}(t_0)}{p!}(t-t_0)^p \\ &\stackrel{\text{def}}{=} \sum_{j=0}^{p} \alpha_j (t-t_0)^j. \end{aligned} \tag{8.5}$$

To estimate β and $g(t)$, we first estimate α_j as the functions of β, denoted as $\alpha_j(\beta)$, by minimizing

$$\sum_{i=1}^{n}\left\{Y_i - X_i^\top\beta - \sum_{j=0}^{p}\alpha_j(T_i - t_0)^j\right\}^2 K_h(T_i - t_0), \tag{8.6}$$

where h is a bandwidth controlling the size of the local neighborhood, and $K_h(\cdot) = K(\cdot/h)/h$ with K a kernel function. Minimize

$$\sum_{i=1}^{n}\left\{Y_i - X_i^\top\beta - \sum_{j=0}^{p}\alpha_j(\beta)(T_i - t_0)^j\right\}^2. \tag{8.7}$$

Denote the solution of (8.7) by β_n. Let $\alpha_j(\beta_n)$ be the estimate of α_j, and denote by $\widehat{\alpha}_{jn}$ $j = 0, \ldots, p$. It is clear from the Taylor expansion in (8.5) that $\nu!\widehat{\alpha}_{jn}$ is an estimator of $g^{(j)}(t_0)$ for $j = 0, \ldots, p$. To estimate the entire function $g^{(j)}(\cdot)$ we solve the above weighted least squares problem for all points t_0 in the domain of interest.

It is more convenient to work with matrix notation. Denote by $\mathbf{Z}$ the design matrix of T in problem (8.6). That is,

$$\mathbf{Z} = \begin{Bmatrix} 1 & (T_1 - t_0) & \ldots & (T_1 - t_0)^p \\ \vdots & \vdots & \vdots & \vdots \\ 1 & (T_n - t_0) & \ldots & (T_n - t_0)^p \end{Bmatrix}.$$

Set $\mathbf{Y} = (Y_1, \cdots, Y_n)^\top$ and $\alpha(\beta) = \{\alpha_0(\beta), \cdots, \alpha_p(\beta)\}^\top$. Let $\mathbf{W}$ be the $n \times n$ diagonal matrix of weights: $\mathbf{W} = \text{diag}\{K_h(T_i - t_0)\}$. The weighted least squares problems (8.6) and (8.7) can be rewritten as

$$\min_{\beta}(\mathbf{Y} - \mathbf{X}\beta - \mathbf{Z}\alpha)^\top\mathbf{W}(\mathbf{Y} - \mathbf{X}\beta - \mathbf{Z}\alpha),$$
$$\min_{\alpha}\{\mathbf{Y} - \mathbf{X}\beta - \mathbf{Z}\alpha(\beta)\}^\top\{\mathbf{Y} - \mathbf{X}\beta - \mathbf{Z}\alpha(\beta)\},$$

with $\alpha(\beta) = \{\alpha_0(\beta), \ldots, \alpha_p(\beta)\}^\top$. The solution vectors are provided by weighted least squares and are given by

$$\begin{aligned}\widehat{\beta}_{\mathrm{LP}} &= [\mathbf{X}^\top\{\mathbf{I}-\mathbf{Z}(\mathbf{Z}^\top\mathbf{W}\mathbf{Z})^{-1}\mathbf{Z}^\top\mathbf{W}\}\mathbf{X}]^{-1}\mathbf{X}^\top\{\mathbf{I}-\mathbf{Z}(\mathbf{Z}^\top\mathbf{W}\mathbf{Z})^{-1}\mathbf{Z}^\top\mathbf{W}\}\mathbf{Y} \\ \widehat{\alpha} &= (\mathbf{Z}^\top\mathbf{W}\mathbf{Z})^{-1}\mathbf{Z}^\top\mathbf{W}(\mathbf{Y}-\mathbf{X}\widehat{\beta}_{\mathrm{LP}})\end{aligned}$$

Theoretically the asymptotic normality is still valid under the conditions similarly to those of Theorem 8.1. More detailed theoretical discussions are referred to Hamilton and Truong (1997).

The quantlet `plmp` is assigned to handle the calculation of $\widehat{\beta}_{\mathrm{LP}}$ and $\widehat{\alpha}$ in XploRe. Its syntax is similar to that of the quantlet `plmk`:

```
plmest=plmp(x,t,y,h,{p})
```

where x, t, y, h are the same as in the quantlet `plmk`. p is the local polynomial order. The default value is $p = 1$, meaning the local linear.

As a consequence, the estimate of the parameter equals $(1.2019, 1.2986, 1.3968)$ and the estimates of the nonparametric function is shown in Figure 8.2. There exists obvious differences between these results from the quantlet `plmk` and `plmp`. More specifically, the results for parametric and nonparametric estimation from the quantlet `plmp` are preferable to these from the quantlet `plmk`.

8.2.3 Piecewise Polynomial

We assume g are Hölder continuous smooth of order $p = (m + r)$, that is, let r and m denote nonnegative real constants $0 < r \leq 1$, m is nonnegative integer such that

$$|g^{(m)}(t') - g^{(m)}(t)| < M|t' - t|^r, \text{ for } t, t' \in [0, 1].$$

Piecewise polynomial approximation for the function $g(\cdot)$ on $[0, 1]$ is defined as follows. Given a positive M_n, divide $[0, 1]$ in M_n intervals with equal length $1/M_n$. The estimator has the form of a piecewise polynomial of degree m based on the M_n intervals, where the $(m+1)M_n$ coefficients are chosen by the method of least squares on the basis of the data. The basic principle is concisely stated as follows.

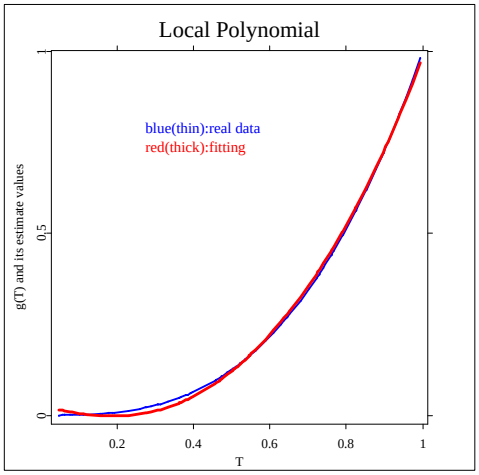

Figure 8.2: The simulation results for nonparametric function via quantlet `plmp`. Thin line: real data; Thick line: the fitting.

`XCSplm02.xpl`

Let $I_{n\nu}(t)$ be the indicator function of the ν-th interval, and d_ν be the midpoint of the ν-th interval, so that $I_{n\nu}(t) = 1$ or 0 according to $t \in [(\nu-1)/M_n, \nu/M_n)$ for $\nu = 1, \ldots, M_n$ and $[1 - 1/M_n, 1]$ or not. $P_{n\nu}(t)$ be the m-order Taylor expansion of $g(t)$ at the point d_ν. Denote

$$P_{n\nu}(t) = \sum_{j=0}^{m} a_{ju} t^j \text{ for } t \text{ in the } \nu\text{th interval}$$

Consider the piecewise polynomial approximation of g of degree m given by

$$g_n^*(t) = \sum_{\nu=1}^{M_n} I_\nu(t) P_{n\nu}(t).$$

Suppose we have n observed data $(X_1, T_1, Y_1), \ldots, (X_n, T_n, Y_n)$. Denote

$$\mathbf{Z} = \begin{Bmatrix} I_{n1}(T_1) & \ldots & I_{n1}(T_1)T_1^m & \ldots & I_{nM_n}(T_1) & \ldots & I_{nM_n}(T_1)T_1^m \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ I_{n1}(T_n) & \ldots & I_{n1}(T_n)T_n^m & \ldots & I_{nM_n}(T_n) & \ldots & I_{nM_n}(T_n)T_n^m \end{Bmatrix}$$

and

$$\eta_{\mathbf{g}} = (a_{01}, \ldots, a_{m1}, a_{02}, \ldots, a_{m2}, \ldots, a_{0M_n}, \ldots, a_{mM_n})^\top$$

Then

$$\begin{pmatrix} g_n^*(T_1) \\ \vdots \\ g_n^*(T_n) \end{pmatrix} = \begin{pmatrix} \sum_{u=1}^{M_n} I_{nu}(T_1)P_{n\nu}(T_1) \\ \vdots \\ \sum_{u=1}^{M_n} I_{nu}(T_n)P_{n\nu}(T_n) \end{pmatrix} = \mathbf{Z}\eta_{\mathbf{g}}.$$

Hence we need to find β and η_g to minimize

$$(\mathbf{Y} - \mathbf{X}\beta - \mathbf{Z}\eta_{\mathbf{g}})^\top (\mathbf{Y} - \mathbf{X}\beta - \mathbf{Z}\eta_{\mathbf{g}}).$$

Suppose that the solution of minimization problem exists. The estimators of β and η_g are

$$\widehat{\beta}_{\mathrm{PP}} = \{\mathbf{X}^\top(\mathbf{I} - \mathbf{P})\mathbf{X}\}^{-\mathbf{1}}\mathbf{X}^\top(\mathbf{I} - \mathbf{P})\mathbf{Y}$$

and $\eta_{ng} = \mathbf{A}(\mathbf{Y} - \mathbf{X}\widehat{\beta}_{\mathrm{PP}})$, where $\mathbf{A} = (\mathbf{Z}^\top\mathbf{Z})^{-\mathbf{1}}\mathbf{Z}^\top$ and $\mathbf{P} = \mathbf{ZA}$. The estimate of $g(t)$ may be described

$$g_n(t) = z(\mathbf{Z}^\top\mathbf{Z})^{-\mathbf{1}}\mathbf{Z}^\top(\mathbf{Y} - \mathbf{X}\widehat{\beta}_{\mathrm{PP}})$$

for a suitable z.

THEOREM 8.2 *There exist positive definite matrices Σ_{00} and Σ_{01} such that both $Cov(X|t) - \Sigma_{00}$ and $\Sigma_{01} - Cov(X|t)$ are nonnegative definite for all $t \in [0,1]$. Suppose that $\lim_{n\to\infty} n^{-\lambda}M_n = 0$ for some $\lambda \in (0,1)$ and $\lim_{n\to\infty} \sqrt{n}M_n^{-p} = 0$. Then $\sqrt{(}\widehat{\beta}_{\mathrm{PP}} - \beta)$ converges to $N(0, \sigma^2\Sigma^{-1})$.*

The quantlet `plmp` evaluates the estimates $\widehat{\beta}_{\mathrm{PP}}$ and $g_n(t)$ stated above. Its syntax is similar to those of the two previous quantlets:

```
plmest=plmp(x,t,y,m,mn)
```

where m and mn represent m and M_n, respectively. We now use the quantlet `plmp` to investigate the example considered in the quantlet `plmk`. We assume $m = 2$ and $M_n = 5$ and compute the related estimates via the quantlet `plmp`. The implementation works as follows.


`XCSplm03.xpl`

The result for parameter β is `plmest.hbeta`$= (1.2, 1.2999, 1.3988)^\top$. Alternatively the estimates for nonparametric part are also given.

8.2.4 Least Square Spline

This subsection introduces least squares spline. We only state its algorithm rather than the theory, which can be found in Eubank (1988) for an overall discussion.

Suppose that g has $m-1$ absolutely continuous derivatives and m-th derivative that is square integrable and satisfies $\int_0^1 \{g^{(m)}(t)\}^2 dt < C$ for a specified $C > 0$. Via a Taylor expansion, the partially linear model can be rewritten as

$$Y = X^\top \beta + \sum_{j=1}^{m} \alpha_j T^{j-1} + Rem(T) + \varepsilon$$

where $Rem(s) = (m-1)!^{-1} \int_0^1 \{g^{(m)}(t)(t-s)_+^{m-1}\}^2 \, dt$. By using a quadrature rule, $Rem(s)$ can be approximate by a sum of the form $\sum_{j=1}^{k} d_j (t - t_j)_+^{m-1}$ for some set of coefficients $d_1, \ldots, d_k$ and points $0 < t_1, \ldots, < t_k < 1$. Take a basis $V_1(t) = 1$, $V_2(t) = t$, $\ldots$, $V_m(t) = t^{m-1}$, $V_{m+1}(t) = (t - t_1)^{m-1}$, $\ldots$, $V_{m+k}(t) = (t - t_k)^{m-1}$ and set $\eta = (\alpha_1, \ldots, \alpha_m, d_1, \ldots, d_k) \stackrel{\mathrm{def}}{=} (\eta_1, \ldots, \eta_{m+k})^\top$. The least squares spline estimator is to minimize

$$\arg\min_{\beta,\eta} \frac{1}{n} \sum_{i=1}^{n} \left\{ Y_i - X_i^\top \beta - \sum_{j=1}^{m+k} \eta_j V_j(T_i) \right\}^2 .$$

Conveniently with matrix notation, denote $\mathbf{Z} = (Z_{ij})$ with $Z_{ij} = \{V_j(T_i)\}$ for $i = 1, \ldots, n$ and $j = 1, \ldots, m + k$ and $\mathbf{X} = (X_1, \ldots, X_n)^\top$. The least squares spline estimator is equivalent to the solution of the minimizing problem

$$(\mathbf{Y} - \mathbf{X}\beta - \mathbf{Z}\eta)^\top (\mathbf{Y} - \mathbf{X}\beta - \mathbf{Z}\eta).$$

If the problem has an unique solution, its form is the same as $(\widehat{\beta}_{\mathrm{PP}}, \eta_{ng})$ in the subsection about piecewise polynomial. Otherwise, we may use ridge idea to modify the estimator. `plmls` is concerned with implementation of the above algorithm in XploRe.

```
plmest=plmls(x,t,y,m,knots)
```

XCSplm04.xpl

Input parameters are

`x` : $n \times d$ matrix of the linear design points

`t` : $n \times 1$ vector of the non-linear design points

`y` : $n \times 1$ vector of the response variables

`m` : the order of spline, and

`knots` : $k \times 1$ vector of knot sequence knots.

Output parameters are

`plmest.hbeat` : $d \times 1$ vector of the estimate of the parameter and

`plmest.hg` : the estimate of the nonparametric part.

8.3 Heteroscedastic Cases

When the variance function given covariates (X, T) is non-constant, the estimators of β proposed in former section is inefficient. The strategy of overcoming this drawback is to use weighted least squares estimation. Three cases will be briefly discussed. Let $\{(Y_i, X_i, T_i), i = 1, \ldots, n\}$ denote a sequence of random samples from

$$Y_i = X_i^\top \beta + g(T_i) + \sigma_i \xi_i, , i = 1, \ldots, n, \tag{8.8}$$

where X_i, T_i, T_i are the same as those in model (8.1). ξ_i are i.i.d. with mean 0 and variance 1, and σ_i^2 are some functions, whose concrete forms will be discussed later.

In general, the least squares estimator $\widehat{\beta}_{\text{LS}}$ is modified to a weighted least squares estimator

$$\beta_W = \Big(\sum_{i=1}^n \gamma_i \widetilde{X}_i \widetilde{X}_i^\top\Big)^{-1} \Big(\sum_{i=1}^n \gamma_i \widetilde{X}_i \widetilde{Y}_i\Big) \tag{8.9}$$

for some weight γ_i $i = 1, \ldots, n$. In our model (8.8) we take $\gamma_i = 1/\sigma_i^2$. In principle the weights γ_i (or σ_i^2) are unknown and must be estimated. Let $\{\widehat{\gamma}_i,\, i = 1, \ldots, n\}$ be a sequence of estimators of γ. One may define an estimator of β by substituting γ_i in (8.9) by $\widehat{\gamma}_i$. Let

$$\widehat{\beta}_{\text{WLS}} = \left(\sum_{i=1}^n \widehat{\gamma}_i \widetilde{X}_i \widetilde{X}_i^\top\right)^{-1} \left(\sum_{i=1}^n \widehat{\gamma}_i \widetilde{X}_i \widetilde{Y}_i\right)$$

be the estimator of β.

Under suitable conditions, the estimator $\widehat{\beta}_{\text{WLS}}$ is asymptotically equivalent to that supposed the function σ_i^2 to be known. Therefore $\widehat{\beta}_{\text{WLS}}$ is more efficient than the estimators given in the previous section. The following subsections present three variance functions and construct their estimators. Three nonparametric heteroscedastic structures will be studied. In the remainder of this section, $H(\bullet)$ is always assumed to be unknown Lipschitz continuous.

8.3.1 Variance is a Function of Exogenous Variables

Suppose $\sigma_i^2 = H(W_i)$, where $\{W_i; i = 1, \ldots, n\}$ are design points, which are assumed to be independent of ξ_i and (X_i, T_i) and defined on $[0, 1]$ in the case where $\{W_i; i = 1, \ldots, n\}$ are random design points. Let $\widehat{\beta}_{\mathrm{LS}}$ and $\widehat{g}_n(\cdot)$ be initial estimators of β and $g(\cdot)$, for example, given by kernel regression in Section 8.2.1. Define

$$\widehat{H}_n(w) = \sum_{j=1}^{n} \widetilde{W}_{nj}(w)\{Y_j - X_j^\top \widehat{\beta}_{\mathrm{LS}} - \widehat{g}_n(T_i)\}^2$$

as the estimator of $H(w)$, where $\{\widetilde{W}_{nj}(t); i = 1, \ldots, n\}$ is a sequence of weight functions satisfying appropriate assumptions. Then let $\widehat{\sigma}_{ni}^2 = H_n(W_i)$.

Quantlet `plmhetexog` performs the weighted least squares estimate of the parameter. In the procedure of estimating the variance function, the estimate given by `plmk` is taken as the primary one.

`XCSplm05.xpl`

Correspondingly the following output are shown in XploRe output window. `plmest.hbetals` is $d \times 1$ vector of LS estimate of parameter, `plmest.hbeta` is $d \times 1$ vector of the estimate $\widehat{\beta}_{\mathrm{WLS}}$, `plmest.hg0` is the estimate of nonparametric function based on `plmest.hbetals`, and `plmest.hg` is the estimate of nonparametric function based on `plmest.hbeta`.

8.3.2 Variance is an Unknown Function of T

Suppose that the variance σ_i^2 is a function of the design points T_i, i.e., $\sigma_i^2 = H(T_i)$, with $H(\cdot)$ an unknown Lipschitz continuous function. Similarly to subsection 8.3.1, we define the estimator of $H(\cdot)$ as

$$\widehat{H}_n(t) = \sum_{j=1}^{n} \widetilde{W}_{nj}(t)\{Y_j - X_j^\top \widehat{\beta}_{\mathrm{LS}} - \widehat{g}_n(T_i)\}^2.$$

Quantlet `plmhett` calculates the weighted least squares estimate of the parameter in this case. In the procedure of estimating the variance function, the

estimate given by `plmk` is taken as the primary one.

```
plmest=plmhett(x,t,y,h,h1)
```

XCSplm06.xpl

8.3.3 Variance is a Function of the Mean

We consider the model (8.8) with $\sigma_i^2 = H\{X_i^\top \beta + g(T_i)\}$, which means that the variance is an unknown function of the mean response.

Since $H(\cdot)$ is assumed to be completely unknown, the standard method is to get information about $H(\cdot)$ by replication, i.e., we consider the following "improved" partially linear heteroscedastic model

$$Y_{ij} = X_i^\top \beta + g(T_i) + \sigma_i \xi_{ij}, \quad j = 1, \ldots, m_i; \quad i = 1, \ldots, n,$$

where Y_{ij} is the response of the j^{th} replicate at the design point (X_i, T_i), ξ_{ij} are i.i.d. with mean 0 and variance 1, β, $g(\cdot)$ and (X_i, T_i) are the same as before.

We compute the predicted value $X_i^\top \widehat{\beta}_{\mathrm{LS}} + \widehat{g}_n(T_i)$ by fitting the least squares estimator $\widehat{\beta}_{\mathrm{LS}}$ and nonparametric estimator $\widehat{g}_n(T_i)$ to the data and the residuals $Y_{ij} - \{X_i^\top \widehat{\beta}_{\mathrm{LS}} + \widehat{g}_n(T_i)\}$, and estimate σ_i^2 by

$$\widehat{\sigma}_i^2 = \frac{1}{m_i} \sum_{j=1}^{m_i} [Y_{ij} - \{X_i^\top \widehat{\beta}_{\mathrm{LS}} + \widehat{g}_n(T_i)\}]^2,$$

where each m_i is unbounded.

Quantlet `plmhetmean` executes the above algorithm in XploRe. For calculation simplicity, we use the same replicate in practice. The estimate given by `plmk` is taken as the primary one.

```
plmest=plmhetmean(mn,x,t,y,h)
```

The following simulated data show us how to run the quantlet `plmhetmean`.

 `XCSplm07.xpl`

8.4 Real Data Examples

In this section we provide some biometrics data sets and illustrate the calculation results when using the quantlets introduced in Section 8.2 to consider these examples. The detailed descriptions are given in following.

EXAMPLE 8.1 *We use the data from the* Framingham Heart Study*, which consists of a series of exams taken two years apart, to illustrate one of the applications of PLM in biometrics. There are* 1615 *men, aged between* 31 *to* 65*, in this data set. The outcome Y represents systolic blood pressure (SBP). Covariates employed in this example are a patient's age (T) and the serum cholesterol level (X). Empirical study indicates that SBP linearly depends upon the serum cholesterol level but nonlinearly on age. For this reason, we apply PLM to investigate the function relationship between Y and (T, X). Specifically, we estimate β and $g(\cdot)$ in the model*

$$Y_i = X_i\beta + g(T_i) + \varepsilon_i, \quad i = 1, \cdots, 1615.$$

For nonparametric fitting, we use a Nadaraya-Watson weight function with quartic kernel

$$(15/16)(1 - u^2)^2 I(|u| \leq 1)$$

and choose the bandwidth using cross-validation.

The estimate value of the linear parameter equals to 10.617*, and the estimate of $g(T)$ is given in Figure 8.3. The figure shows that with the age increasing, SBP increases but looks like a straight line. The older the age, the higher the SBP is.*

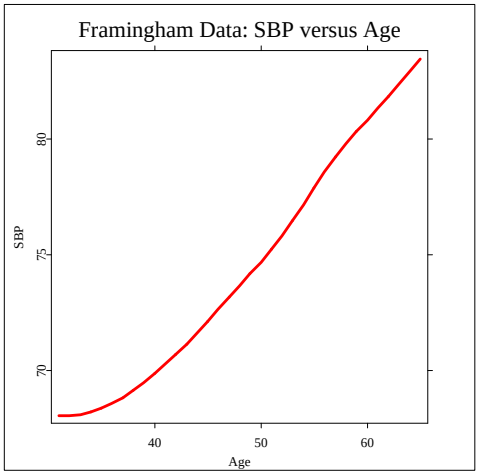

Figure 8.3: Relationship SBP and serum cholesterol level in Framingham Heart Study.

XCSplm08.xpl

EXAMPLE 8.2 *This is an example of using PLM to analyze* NHANES Cancer data. *This data set is a cohort study originally consisting of* 8596 *women, who were interviewed about their nutrition habits and when later examined for evidence of cancer. We restrict attention to a sub-cohort of* 3145 *women aged* $25 - 50$ *who have no missing data the variables of interest. The outcome* Y *is saturated fat, while the predictors include age, body mass index (BMI), protein and vitamin A and B intaken. Again it is believable that* Y *depends nonlinearly on age but linear upon other dummy variables.*

In this example we give an illustration of the `plmls` *for the real data. We select* $m = 3$ *and the knots at* $(35, 46)$. *As a consequence, the estimates of linear parameters are* $(-0.162, 0.317, -0.00002, -0.0047)$, *and the nonparametric estimated are shown in Figure 8.4. The curve of the nonparametric part in this data set is completely different from that of the above example and looks like arch-shape. The pattern reaches to maximum point at about age* 35.

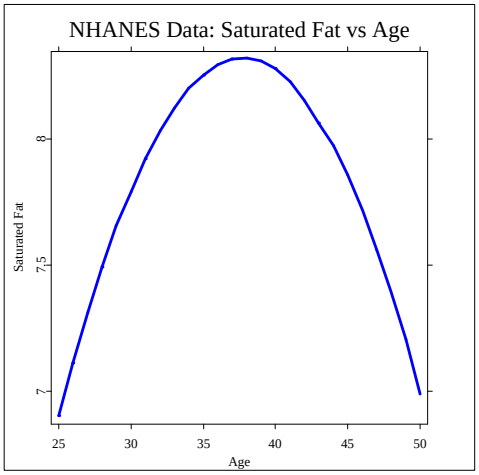

Figure 8.4: NHANES regression of saturated fat on age.

XCSnhanes.xpl

We also run other quantlets for these two data sets. We found that the estimates of nonparametric parts from different quantlets have similar shapes, although differences in the magnitude of the estimates from different estimation methods are visible.

Bibliography

de Boor, C. (1978). *A Practical Guide to Splines*. New York:Springer-Verlag.

Chen, H.(1988). Convergence rates for parametric components in a partly linear model. *Annals of Statistics*, **16**, 136-146.

Engle, R. F., Granger, C. W. J., Rice, J. and Weiss, A. (1986). Semiparametric estimates of the relation between weather and electricity sales. *Journal of the American Statistical Association*, **81**, 310-320.

Eubank, R. L. (1988). *Spline Smoothing and Nonparametric Regression*. New York: Marcel Dekker.

Fan, J. and Gijbels, I. (1996). *Local Polynomial Modelling and Its Applications*. Vol. 66 of *Monographs on Statistics and Applied Probability*, Chapman and Hall, New York.

Green, P., Jennison, C. and Seheult, A. (1985). Analysis of field experiments by least squares smoothing. *Journal of the Royal Statistical Society, Series B*, **47**, 299-315.

Härdle, W. (1990). *Applied Nonparametric Regression.* Cambridge University Press, New York.

Härdle, W., Liang, H. and Gao, J. T. (2000). *Partially Linear Models*. Springer-Physica-Verlag, Heidelberg.

Hastie, T. J. and Tibshirani, R. J. (1990). *Generalized Additive Models*. Vol. 43 of *Monographs on Statistics and Applied Probability*, Chapman and Hall, London.

Hamilton, S. A. and Truong, Y. K. (1997). Local linear estimation in partly linear models. *Journal of Multivariate Analysis*, **60**, 1-19.

Heckman, N.E. (1986). Spline smoothing in partly linear models. *Journal of the Royal Statistical Society, Series B*, **48**, 244-248.

Rice, J.(1986). Convergence rates for partially splined models. *Statistics & Probability Letters*, **4**, 203-208.

Robinson, P.M.(1988). Root-n-consistent semiparametric regression. *Econometrica*, **56**, 931-954.

Speckman, P. (1988). Kernel smoothing in partial linear models. *Journal of the Royal Statistical Society, Series B*, **50**, 413-436.

Stone, J. C. (1985). Additive regression and other nonparametric models. *Annals of Statistics*, **13**, 689-705.

Wand, M.P. and Jones, M. C.(1995). *Kernel Smoothing*. Vol. 60 of *Monographs on Statistics and Applied Probability*, Chapman and Hall, London.

9 Analysis of contingency tables

Masahiro KURODA

9.1 Abstract

This chapter presents log-linear modelings that are useful for analyzing contingency tables. The log-linear models are easy to describe the dependence among the variables of statistical models and are usually used to the analysis of contingency tables. Given the observed data in a contingency table, the log-linear analysis estimates the model parameters and selects a best model that can explain the relationship among variables.

First section introduces the log-linear modelings for two- and three-ways contingency tables and extend them to multidimensional contingency tables. Second section provides the statistical inference for log-linear models that is to find maximum likelihood estimates of model parameters and select a log-linear model. Third section shows the computational algorithm to perform the log-linear analysis. Final section illustrates numerical examples using XploRe.

10 Identifying Coexpressed Genes

Qihua Wang

Some gene expression data contain outliers and noise because of experiment error. In clustering, outliers and noise can result in false positives and false negatives. This motivates us to develop a weighting method to adjust the expression data such that the outlier and noise effect decrease, and hence result in a reduction in false positives and false negatives in clustering.

In this paper, we describe the weighting adjustment method and apply it to a yeast cell cycle data set. Based on the adjusted yeast cell cycle expression data, the hierarchical clustering method with a correlation coefficient measure performs better than that based on standardized expression data. The clustering method based on the adjusted data can group some functionally related genes together and yields higher quality clusters.

10.1 Introduction

In order to explore complicated biological systems, microarray expression experiments have been used to generate large amounts of gene expression data (Schena *et al.* (1995), DeRisi *et al.* (1997), Wen *et al.* (1998), Cho *et al.* (1998)). An important type of those experiments is to monitor each gene multiple times under some conditions (Spellman *et al.* (1998), Cho *et al.* (1998), Chu *et al.* (1998)). Those of this type have allowed for the identification of functionally related genes due to common expression patterns (Brown *et al.* (2000), Eisen *et al.* (1998), Wen *et al.* (1998), Roberts *et al.* (2000)). Because of the large number of genes and the complexity of biological networks, clustering is a useful exploring technique for analysis of gene expression data. Different clustering methods including the hierarchical clustering algorithm (Eisen *et al.* (1998), Wen *et al.* (1998)), the CAST algorithm (Ben-Dor *et al.*, 2000) and self-organizing maps (Tamayo *et al.*, 1999) have been employed to analyze ex-

pression data.

Given the same data set, different clustering algorithms can potentially generate very different clusters. A biologist with a gene expression data set is faced with the problem of choosing an appropriate clustering algorithm or developing a more appropriate clustering algorithm for his or her data set. Cho *et al.* (1998) recently published a 17-point time course data set measuring the expression level of each of 6601 genes for the yeast Saccharomyces Cerevisiae obtained from using an Affymetrix hybridization array. Cells in a yeast culture were synchronized, and cultured samples were taken at 10-minutes intervals until 17 observations were obtained. Heyer, Kruglyak and Yooseph (1999) presented a systematic analysis procedure to identify, group, and analyze coexpressed genes based on this 17-point time course data.

An important problem for clustering is to select a suitable pairwise measure of coexpression. Possible measures include the Euclidean distance, correlation and rank correlation. Euclidean distances and pattern correlation have a clear biological meaning: Euclidean distances are used when the interest is in looking for identical patterns, whereas correlation measures are used in the case of the trends of the patterns.

In the clustering, most measures scored curves with similar expression patterns well, but often gave high scores to dissimilar curves or low scores to similar ones. We will refer to a pair that is dissimilar, but receives a high score from the similarity measure as a false positive (Heyer, Kruglyak and Yooseph, 1999), and a pair that is similar, but receives a low score as a false negative. As pointed out by Heyer, Kruglyak and Yooseph (1999) that the correlation coefficient performed better than the other measures, but resulted in many false positives. It is noted that the reason for false positive to occur is outlier effect. Hence, Heyer, Kruglyak and Yooseph (1999) proposed a new measure called jackknife correlation. For a data set with t observations, the jackknife correlation J_{ij} is defined as $J_{ij} = \min\{\rho_{ij}^{(1)}, \rho_{ij}^{(2)}, \cdots, \rho_{ij}^{(t)}, \rho_{ij}\}$, where ρ_{ij} denotes the correlation of the gene pair i, j and $\rho_{ij}^{(l)}$ denotes the correlation of the pair i, j computed with the lth observation deleted. An advantage of this method is that it results in a reduction in false positives. However, this method might be radical and lead to false negatives since it takes the least value of these correlation coefficients as the measure of the similarity. On the other hand, the method may lose much valuable information since it works by deleting data. Also, the jackknife correlation is only robust to a single outlier. For n outliers, a more general definition of jackknife correlation is needed. For this case, however, this method is computationally intensive for even small values of n and can result in the

loss of much valuable information since it deletes n data points.

If the expression level of a gene at each time point is viewed as a coordinate, then the standardized expression level of each gene at all t time points describes a point in the t dimensional space, and the Euclidean distance between any two points in this space can be computed. Euclidean distances are more affected by small variations in the patterns and produce less interpretable clusters of sequences. As pointed by Heyer, Kruglyak and Yooseph (1999) the two points for which the distance is minimized are precisely the points that have the highest correlation. However, the opposite is not true. That is, a pair of genes that are dissimilar and have large Euclidean distance may have high correlation because of outlier effect and hence receive a high score from the similarity measure of correlation coefficient.

This shows that the Euclidean distance measure with standardized data performs better than the correlation coefficient measure in the sense of resulting in less false positive. However, the Euclidean distance measure still result in many false negatives due to the effect of outliers. If the expression levels of two genes are close to each other but one of the time points, and one of the two genes has a high peak or valley at the remaining time point, then the Euclidean distance may be large and hence the pair which closes to each other except for the outlier may be considered as dissimilarity.

It seems difficult to avoid outlier effect by selecting similarity measure. A possible method to reduce the outlier effect is to adjust the expression data.

Wang (2002) proposes a weighting adjustment method and apply it to the 17 time-point time course data such that a similarity measure assigns higher score to coexpressed gene pairs and lower scores to gene pairs with unrelated expression patterns, and hence results in not only a reduction of false positives but also a reduction of false negatives in clustering. Here, we present the work.

10.2 Methodology and Implementation

We consider the Saccharomyces cerevisiae data set by Cho *et al.* (1998). This data set measures the expression level of each of the 6601 genes of Saccharomyces cerevisiae at 17 time points, sampled every ten minutes during roughly two complete cell cycles. Before giving and applying our method to the data set, we first filter away the genes that were either expressed at very low levels or did not vary significantly across the time points (Heyer, Kruglyak and

Yooseph, 1999). The reason to do so is that the fluctuations were more likely noise than signal if the expression levels were below a detection threshold or that the gene that showed so little variation over time may be inactive or not involved in regulation. We remove the genes whose expression values across all the time points are less than 250 and those whose maximum expression levels are not larger than 1.1 times of their average expression levels. After filtering, 3281 genes remained in the data set.

10.2.1 Weighting Adjustment

Many of the false positives and false negatives occurred due to the effect of outliers by standard clustering methods. A possible method to reduce the effect of outliers is to adjust the raw data by a certain method. It is noted that the expression level of a gene at any time point is closely related to the expression levels of the gene at the time points in the nearest neighbor of this point. It is likely that the closer the two time points, the higher the relationship between the two expression levels at the two time points.

This leads us to use a weighting method to adjust the expression values so that not only the effect of outliers decreases but also data analysis is less sensitive to small perturbation in the data. The data have been standardized by subtracting the mean and dividing by the standard deviation. Let $x_{i,j}$ be the standardized expression level of the ith gene at the j time point for $i = 1, 2, \ldots, 3281$ and $j = 1, 2, \ldots, 17$. We get the following adjusted expression level

$$x'_{i,j} = \begin{cases} \frac{1}{2}x_{i,j} + \frac{1}{3}x_{i,j+1} + \frac{1}{6}x_{i,j+2}, & \text{if } j = 1 \\ \frac{1}{5}x_{i,j-1} + \frac{1}{2}x_{i,j} + \frac{1}{5}x_{i,j+1} + \frac{1}{10}x_{i,j+2}, & \text{if } j = 2 \\ \frac{1}{12}x_{i,j-2} + \frac{1}{6}x_{i,j-1} + \frac{1}{2}x_{i,j} + \frac{1}{6}x_{i,j+1} + \frac{1}{12}x_{i,j+2}, & \text{if } 3 \le j \le 15 \\ \frac{1}{10}x_{i,j-2} + \frac{1}{5}x_{i,j-1} + \frac{1}{2}x_{i,j} + \frac{1}{5}x_{i,j+1}, & \text{if } j = 16 \\ \frac{1}{2}x_{i,j} + \frac{1}{3}x_{i,j-1} + \frac{1}{6}x_{i,j-2}. & \text{if } j = 17 \end{cases}$$

It is easily seen that the adjusted expression level of a gene at the jth time point is the weighting average of the expression levels of the gene at the time points in the nearest neighbor of the j time point for $j = 1, 2, \ldots, 17$.

Actually, the adjusted expression level of the jth time point is given by assigning weight $1/2$ to jth time point and total weight $1/2$ to other points. The symmetric points about the jth time point such as $j+1$ and $j-1$ are assigned the equal weights for $j = 3, 2, \ldots, 15$.

The weights which are assigned to time points k with $|k - j| > 2$ are zero and

to time point $j+2$ or $j-2$ is 1/2 time of that for the time point $j+1$ or $j-1$. One intuitive method for seeing how the weighting method behaves is to plot the expression data and the adjusted ones for some gene pairs.

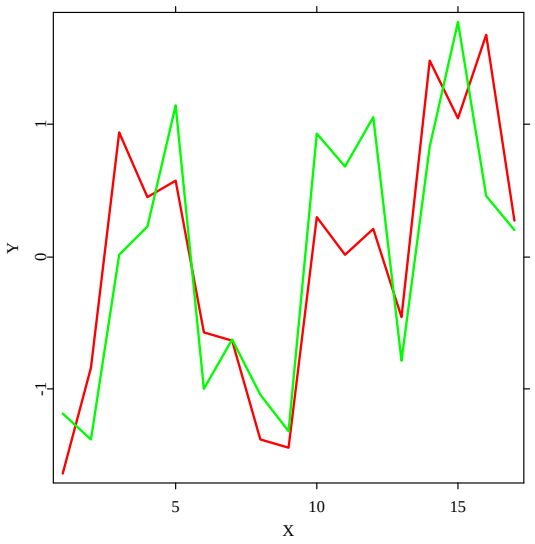

Figure 10.1: Standardized expression level curves for YDR224c/HTB1 and YDR225w/HTA1_i. The gene pair has a correlation coefficient of 0.8094 based on the standardized data.

`XCSclust01.xpl`

¿From Figure 10.1 to Figure 10.4, it seems that the curves of the functionally related gene pairs with coexpression become more similar to each other after adjustment. ¿From Figures 10.5 and 10.6, the unrelated gene pair which is not coexpressed seems to become further away from each other. Another more exact method is to compare the correlation coefficients of gene pairs or Euclidean distances of them based on the expression data with those based on the adjusted ones. It is interesting to find that the correlation coefficients of the most of highly correlated gene pairs become larger and those of lowly correlated gene pairs become smaller after the expression values are adjusted.

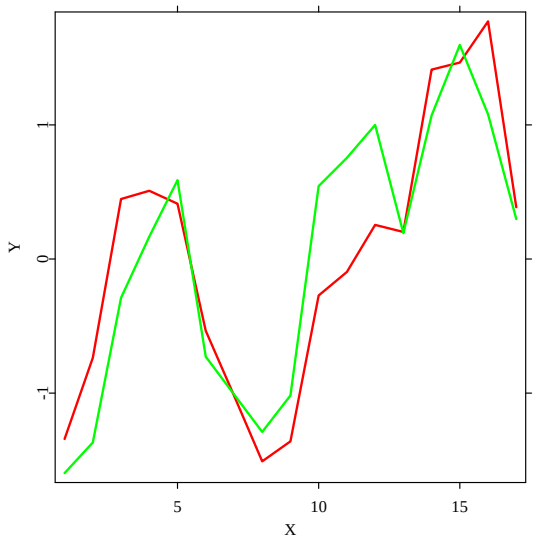

Figure 10.2: Adjusted expression level curves for YDR224c/HTB1 and YDR225w/HTA1_i. The gene pair has a correlation coefficient of 0.8805 based on the adjusted data.

XCSclust02.xpl

This can be seen from Table 10.1 and Figure 10.7.

¿From Figure 10.7, it is easy to see that the correlation coefficients of the most gene pairs whose correlation coefficients are larger than 0.6 before adjustment become larger after adjustment, and those whose correlation coefficients are less than 0.2 before adjustment become less after adjustment. That is, this method gives higher score to similar gene pairs and lower score to dissimilar ones. This may be due to the fact that the weighting adjustment method can lead to a reduction of effect of outliers and noise in expression data. From Figure 10.7 and Table 10.1, we also see that some of the highly correlated pairs are given lower correlation coefficient score after the expression data are adjusted. The reason may be that outliers or data noise lead to the high correlation between

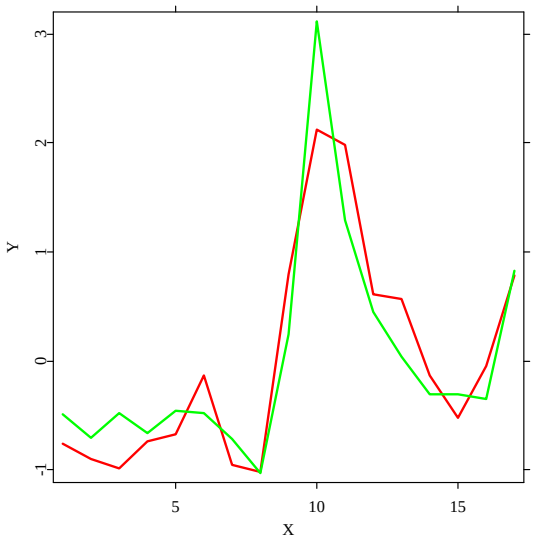

Figure 10.3: Standardized expression level curves for YDL179w/PCL9 and YLR079w/SIC1. The gene pair has a correlation coefficient of 0.9106 based on the standardized data.

XCSclust03.xpl

these gene pairs, or that, randomly, some of them display very similar pattern before adjustment. After weighting adjustment for the expression values, the correlation coefficients for these pairs will decrease since the adjustment method leads to a reduction of effect of outliers, data noisy and randomization. Also, it is observed that some lowly correlated gene pairs are given much higher correlation coefficient score after the expression data are adjusted. The reason may be that only one of a gene pair contains outliers at the same time points or one of the two genes has high peaks and another gene have high valleys at the same time points, and these outliers lead to the low correlation between the gene pair. After adjustment, effect of outliers decreases and hence the correlation coefficient for the gene pair will increase. This, for example, can be seen from Figures 10.8 and 10.9, which contain plots of the expression level

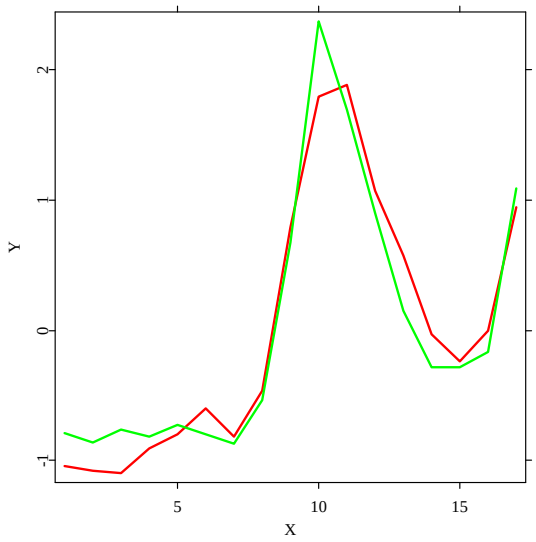

Figure 10.4: Standardized expression level curves for YDL179w/PCL9 and YLR079w/SIC1. The gene pair has a correlation coefficient of 0.9675 based on the adjusted data.

XCSclust04.xpl

curves for gene pair YAR002w and YBL102w/SFT2 based on standardized expression data and adjusted ones, respectively. From Figure 10.8, we see that YBL102w/SFT2 and YAR002w seem to be overly expressed at two different time points of 90 minutes and 150 minutes, respectively. At the time point of 90 minutes, only YBL102w/SFT2 has a high peak. At the time point of 150 minutes, YAR002w has a high peak and YBL102w/SFT2 has a low valley. If one removes the expression values of the two genes at the two time points, the correlation coefficient of the two genes increase to 0.6036 from 0.3150. This shows that the two special expression values lead to a low correlation between the two genes. From Figure **??**, it is easily seen that the weighting adjustment method leads to a reduction of effect of the expression values at the two time

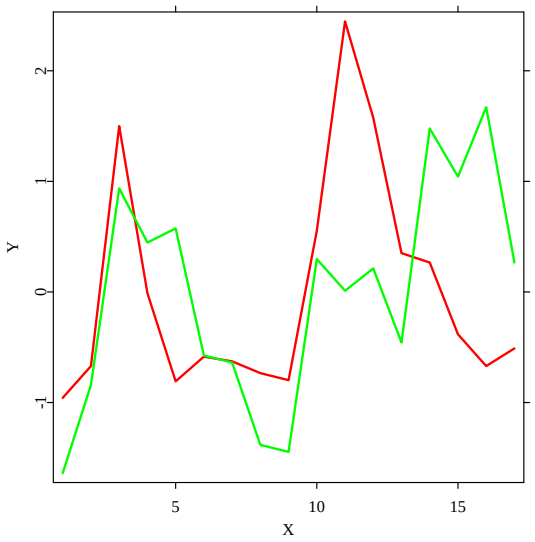

Figure 10.5: Standardized expression level curves for YDL227c/HO and YDR224c/HTB1. The gene pair has a correlation coefficient of 0.3172 based on the standardized data based on the standardized expression data.

XCSclust05.xpl

points so that the correlation coefficient of the two genes increase to 0.8113.

By the above important features, we can expect that this adjustment method will lead to a reduction of both the false positives and false negatives when Pearson correlation coefficient clustering algorithm is used.

10.2.2 Clustering

We clustered the gene expression time series according to the Pearson correlation coefficient since it not only conforms well to the intuitive biological notion

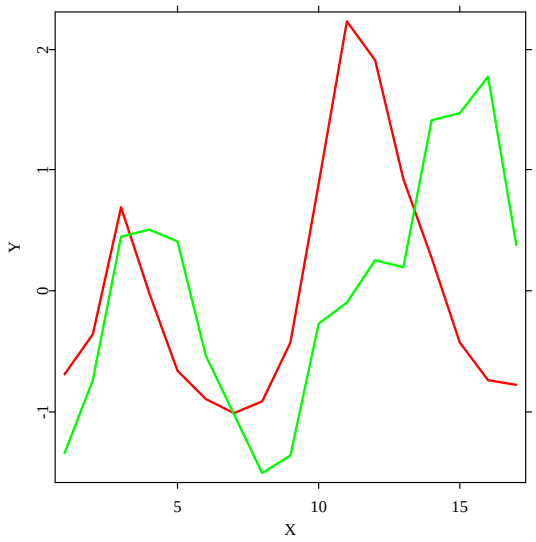

Figure 10.6: Standardized expression level curves for YDL227c/HO and YDR224c/HTB1. The gene pair has a correlation coefficient of 0.1401 based on the adjusted expression data.

XCSclust06.xpl

and performs better than other measures, but also the correlation coefficient measure has the important features described in Section 2.1 for the adjusted expression data.

The clustering method that we use is the popular hierarchical method. This method computes a dendrogram that assembles all the genes into a single tree. Starting with N clusters containing a single gene each, at each step in the iteration the two closest clusters are merged into a larger cluster by calculating an upper-diagonal similarity matrix by the metric described above and scanning the matrix to identify the highest value. Similarity measure between clusters is defined as that between their average expression pattern. After $N-1$ steps, all the genes are merged together into an hierarchical tree.

Table 10.1: Correlation coefficients for some gene pairs based on the standarized and adjusted expression data

Gene	Pairs	BA	AA
YKL130c/SHEI	YNL053w/MSG5	0.8047	0.8474
YDL179w/PCL9	YLR079w/SICI	0.9106	0.9676
YJL157c/FAR1	YKL185w/ASH1	0.9293	0.9535
YJR092w/BUD4	YLR353w/BUD8	0.6904	0.9684
YIL009w/FAA3	YLL040c/VPS13	0.7519	0.8798
YJL196c/EL01	YJR148w/TWT2	0.6815	0.7433
YBL023c/MCM2	YBR202w/CDC47	0.7891	0.8383
YHR005c/GPA1	YJL157c/FAR1	0.8185	0.8320
YOR373w/NUD	YJL157c/FAR	-0.1256	-0.2090
YOR373w/NVD1	YAL040c/c	-0.1133	-0.2222
YDR225w/HTA1$\tilde{\text{i}}$	YLL022c	0.3493	0.0673
YJR018w	YJR068w/RFe2	0.9046	0.8968
YJR068/RFC2	YJR132w/NMD5	0.8700	0.7121

BA: Before Adjustment, AA: After Adjustment

Once the tree is constructed, the data can be partitioned into any number of clusters by cutting the tree at the appropriate level. For large data sets, however, it is not easy to choose an appropriate location for cutting the tree. We will not address this problem here since our purpose in this paper is to show how our weighting adjustment method improves the classification results. To evaluate how 'good' our clustering is, let us identify some applications.

YDR224c/HTB1 and YDR225w/HTA1 are late G1 and G2 regularly genes which have the same biological function (DNA replication, (Cho *et al.*, 1998)). A natural question is: Can the two genes be grouped together based on the adjusted expression levels? To answer this question, let us find the clusters including the two genes.

In our hierarchical tree, it can be found the smallest cluster including YDR224c/HTB1 contains two genes, YDR224c/HTB1 and YDR225w/HTA1. It is interesting to note that the two genes are just known functionally related by Cho *et al.* (1998).

The above result implies that this cluster is also the smallest one which includes the two genes. This shows that our method can group the two functionally

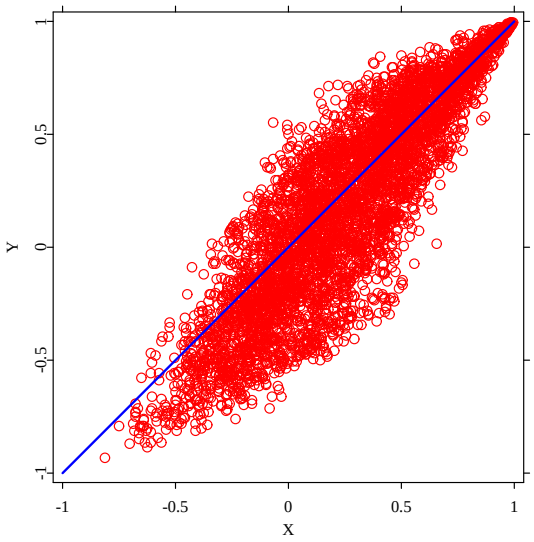

Figure 10.7: Correlation coefficients of 10,000 gene pairs. ρ and ζ are the correlation coefficients based on the standardized expression data and the adjusted expression data, respectively. 869 gene pairs have correlation coefficients which are larger than 0.6. The correlation coefficients of 556 pairs of them become larger after adjustment. 2303 gene pairs have correlation coefficients which are less than 0.2. The correlation coefficients of 1520 pairs of them become less after adjustment.

XCSclust07.xpl

related genes together.

Another intuitive method to evaluate objectively the quality of the clustering for the particular application is to plot the expression data for the genes in the clustering and determine whether the plots look similar and how the plots look similar. Figure 5.1 plots the expression level curves for the two genes. By Figure 5.1, their expression patterns are indeed similar to each other.

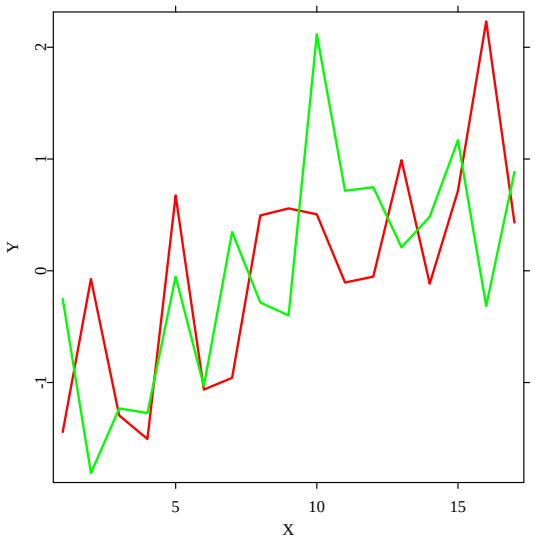

Figure 10.8: Standardized expression level curves for YAR002w and YBL102w/SFT2. The gene pair has a correlation coefficient of 0.3750 based on the standardized expression data.

XCSclust08.xpl

It is known there are 19 late G1 regulatory genes and two of them are just YDR224c/HTB1 and YDR225w/HTA1 (Cho *et al.*, 1998). In our clustering tree, the cluster including the two genes whose gene number is the closest to 19 contains 17 genes, 4 of them are known to be late G1 regulary and functionally related with DNA replication. It is known that some unrelated genes also may have similar expression patterns. Hence, the remaining 13 genes are not necessarily functionally related with the 4 genes even though they are coexpressed. However, the 13 genes provide excellent candidates for further study.

We also try to find the smallest cluster including the 19 genes in late G1 group.

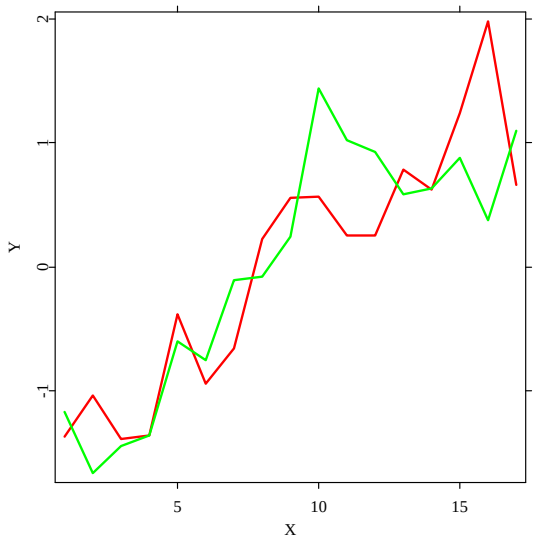

Figure 10.9: Standardized expression level curves for YAR002w and YBL102w/SFT2. The gene pair has a correlation coefficient of 0.8113 based on the adjusted expression data.

XCSclust09.xpl

Unfortunately, this cluster contains 2930 genes and hence is not of high quality since it contains many lowly related genes. This reason may be that some gene pairs in the late G1 group are lowly related. For example, the correlation coefficient of the gene pair, YDR225w/HTA1 and YPR175w/DPB2, in the late G1 group is 0.0471.

Another problem we should answer is whether the adjustment method improves the classification result compare to the corresponding hierarchical method based on standardized expression data. Let us consider the above example again and see how the clustering method based on standard expression data behaves. From the hierarchical tree based on the standardized data without adjustment, the smallest cluster including YDR224c/HTB1 is {YDR224c/HTB1, YDR134C/_f}.

However, YDR134C/_f is not known to be functionally related with YDR224c/HTB1 though it provides a possible candidate. Figure 5.10 plots the expression level curves of the two genes.

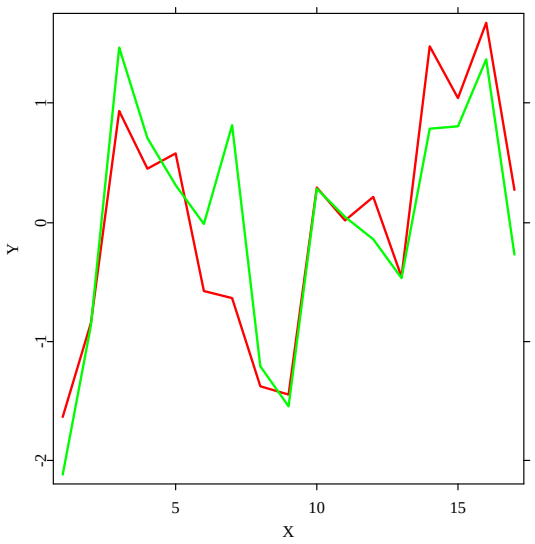

Figure 10.10: Standardized expression level curves for YDR224c/HTB1 and YDR134c/_f in the clustering tree based on the standardized expression data.

XCSclust10.xpl

In the clustering tree based on the standardized expression data without adjustment, the cluster including all the 3281 genes is the only one including both YDR224c/HTB1 and YDR225w/HTA1. This shows that this method cannot group the two functionally related genes together.

Both YDR224c/HTB1 and YDR225w/HTA1 are also in the late G1 group mentioned above, which contains 19 genes. Hence, the cluster including the 3281 genes are also the only one including the 19 genes. This shows that this clustering method with standardized expression data yields much lower quality

clusters and also cannot group the 19 genes together.

Let us consider another example. YJR092W/BUD4 and YLR353W/BUD8 are M regulatory genes which are functionally related to directional growth (Cho *et al.*, 1998). In our clustering tree, the smallest cluster including the two genes contains four genes. The other two genes are YNL066W and YOR025W/HST3. Figure 10.11 plots the standardized expression level curves of the four genes.

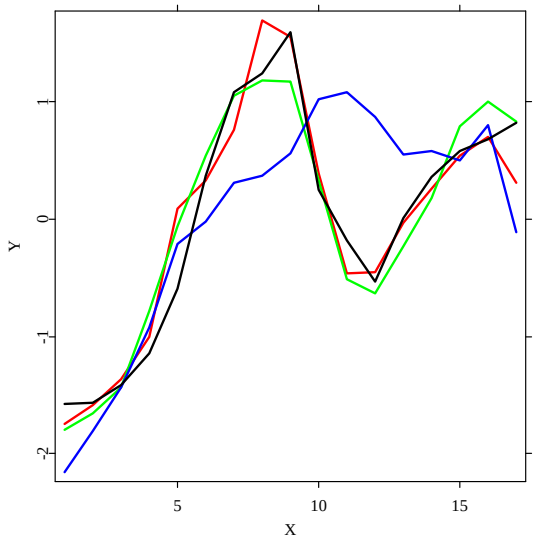

Figure 10.11: Standardized expression level curves for the genes in the smallest cluster including YJR092w/HUD4 and YLR353w/HUD8 in the clustering tree based on the adjusted data. The remaining two genes are YNL066w and YOR025w/HST3.

XCSclust11.xpl

¿From Figure 10.11, all the expression level curves are similar to each other except YNL066W. It is easy to see that YOR025W/HST3 is coexpressed with YJR092W/HUD4 and YLR353W/BUD8. Hence, YOR025W/HST3 provides an excellent candidate for further study whether it is functionally related with

YJR092W/HUD4 and YLR353W/BUD8.

Let us apply the clustering method with standardized data without adjustment to the above example. The smallest cluster including YJR092W/BUD4 and YLR353W/BUD8 contains 11 genes.

YJL157c/FAR1 is an early G1 regulary gene which is functionally related with mating pathway. In our hierarchical tree, the smallest cluster including this gene contains two genes,YJL157c/FAR1 and YGR183C/QCR9_ex1. From Figure 10.12, we can see that YGR183C/QCR9_ex1 is coexpressed with YJL157c/FAR1 though it is not known to be early G1 regularly gene which is functionally related with YJL157c/FAR1. The second smallest cluster contains 5 genes in addition to YJL157c/FAR1. One of them is YKL185w/ASH1, which is known to be functionally related with YJL157c/FAR1. Actually, this cluster is also the smallest one including the two functionally related genes.

For the clustering method with standardized expression data, the smallest cluster including YJL157c/FAR1 contains 6 genes. The second smallest cluster contains 7 genes. No genes in the two clusters are known to be functionally related. The smallest cluster including the two functionally related genes, YJL157c/FAR1 and YKL185w/ASH1, contains 96 genes.

It is known that YIL140w/SRO4 is the only one which is known to be S regulatory and to be related with directional growth. Are there any functionally related genes with it? Which genes are coexpressed with it? In our clustering tree, the first smallest cluster including this gene is {YIL140w/SRO4, YPL163c/SVS1}. The second smallest cluster contains another gene, YOR373w/NUD1, in addition to the above two genes. From the standardized data without adjustment, different clusters are obtained. The smallest cluster including YIL140w/SRO4 is {YIL140w/SRO4, YLR326w}. The second smallest cluster contains YNL243w/SLA2 and YPR052c/NHP6A in addition to the two genes in the smallest cluster. Figures 10.13 and 10.14 plot the expression level curves for the genes in the two second smallest clusters, respectively.

¿From Figures 10.13 and 10.14, we see that the expression level curves for the genes in the cluster by our method are more closer to each other. This also can be seen by their correlation coefficients. In our cluster, other genes are more highly related with YIL140w/SRO4. This shows that our clusters have higher quality for this special example. From Figure 10.14, the cluster based on the standardized expression data is of much more lowly quality since it contains some lowly related genes.

¿From the above examples, we see that the clustering method based on the

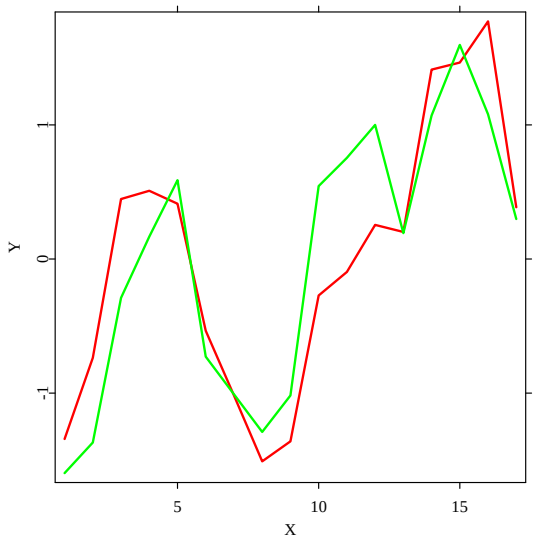

Figure 10.12: Standardized expression level curves for the genes in the smallest cluster including YJL157c/FAR1 in the clustering tree based on the adjusted data.

XCSclust12.xpl

adjusted expression data behave better than that based on the standardized expression data without adjustment. Our method can group coexpression genes and some functionally related genes together. However, We have not found that any known functionally related genes can be in the same clusters with high quality in the clustering tree based on the standard expression data. Figure 10.13 shows that two functionally related gene pairs, YJR092w/BUD4 and YLR353w/BUD8, are in a cluster with 11 genes. However, this cluster is clearly not of high quality since it contains some lowly related genes with YJR092w/BUD4 and YLR353w/BUD8.

It should be pointed out some functionally related genes cannot also group together based on the adjusted data. This reason may be that some functionally

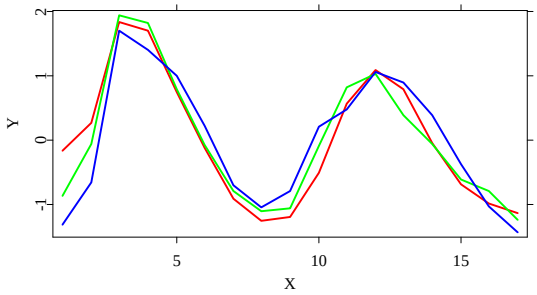

Figure 10.13: Standardized expression level curves for the genes in the second smallest cluster including YIL140w/SRO4 in the clustering tree based on adjusted expression data.

XCSclust13.xpl

related genes are not coexpressed. Also, genes in the same high quality cluster are not necessarily functionally related since some functionally unrelated genes have similar expression patterns. Because there is a connection between coexpression and functional relation, the clusters are an exploratory tool that meant to identify candidate functionally related genes for further study though they do not reveal the final answer whether these genes in the same clusters are functionally related.

10.3 Concluding Remarks

Our purpose to use the weighting method to adjust the expression data is to decrease the the effect of the outliers and noise. It is reasonable to assign a weight of 1/2 to the point that one hopes to adjust and a total weight of 1/2 to other points which are located in its nearest neighbor. This method of assigning weights used in this paper can effectively result in a reduction of effect of outliers and noise and does not change the normal expression levels too much. Also, the weighting method is robust for the slight change of the

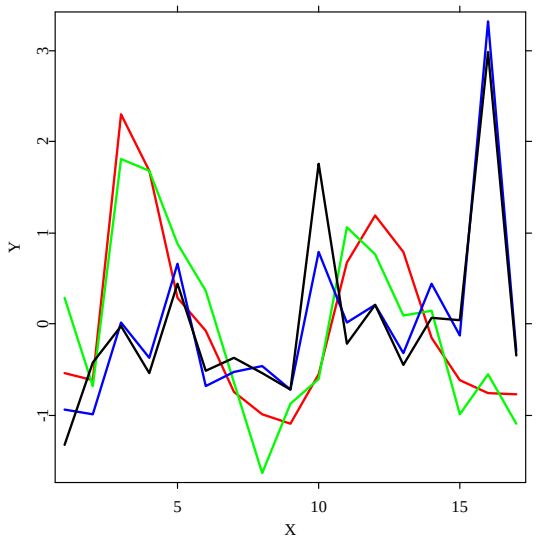

Figure 10.14: Standardized expression level curves for the genes in the second smallest cluster including YIL140w/SRO4 in the clustering tree based on the standardized expression data.

XCSclust14.xpl

weights. If one assigns a much larger weight than 1/2 to the point which is adjusted, effect of outlier or noise may not decrease effectively. If one assigns a much less weight than 1/2 to the point, the adjusted expression level may not provide correct information and hence the weighting method may result in wrong clustering results since such a method changes the expression levels too much. It should be pointed out that the weighting adjustment method can be applied to any analysis procedures for any gene expression data to decrease the effect of outlier and noise though we apply it only to a hierarchical clustering for the yeast cell cycle data in this paper.

Heyer, Kruglyak and Yooseph (1999) proposed a jackknife correlation measure to resolve false positive. As pointed out before, this method may be

radical and may lead to false negatives. An improved method which can avoid the false negatives may be to use another jackknife correlation $\rho_{ij,JK} = \frac{1}{n}\sum_{k=1}^{n}(n\rho_{ij} - (n-1)\rho_{ij}^{(k)})$ based on the adjusted data, where ρ_{ij} and $\rho_{ij}^{(k)}$ are as defined in Introduction. On the other hand, the clustering method with the jackknife correlation measure $\rho_{ij,JK}$ based on the standardized expression data without adjustment may be conservative and cannot avoid the occurring of false positives very well. Based on the adjusted data, however, the jackknife correlation measure may avoid the false positives and false negatives very well. We will investigate the measure in future work.

Bibliography

Ben-Dor, A., Bruhn, L., Friedman, N., Nachman, I., Schummer, M. and Yakhini, Z. (2000). Tissue classification with gene expression profiles. In *Proceedings of the 4th Annual International Conference on Computational Molecular Biology (RECOMB)* Universal Academy Press, Tokyo.

Brown,M.P.S., Grundy, W.N., Lin,D., Cristianini,N., Sugnet, C.W., Furey,T.S., Jr, M.A. and Haussler, D. (2000). Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proc. Natl Acad. Sci. USA*, **96**, 9112-9217.

Cho,R.J., Cambell,M.J., Winzeler,E.A., Steinmetz,E.A., Conway,A., Wodicka,L., Wolfsberg, T.J., Gabrielian, A.E., Landsman,D., Lockhart,D.J. and Davis, R.W. (1998). A genome-wide transcriptional analysis of the mitotic cell cycle. *Mol. Cell.,* **2**, 65-73.

Chu,S., J.DeRisi, M. eisen, J. Mulholland, D. Botstein, P. Brown, and I. Herskowize (1998). The transcriptional program of sporulation in budding yeast. *Science* **282**, 699-705.

DeRisi,J.L., Iyer,V.R.and Brown, P. O. (1997)Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, **278**, 680-686.

Eisen,M.B., Spellman, P.T., Brown,P.O. and Botstein,D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proc. Natl Acad.Sci USA*, **95**, 14863-14868.

Heyer, L.J., Kruglyak,S. and Yooseph,S. (1999). Exploring expression data: Identification and Analysis of coexpressed genes. *Genome Research*, **9**, 1106-1115.

Roberts,C., Nelson, B., Marton, M., Stoughton,R., Meyer, M., Bennett, H., He, Y., Dai, H., Walker,W., Hughes, T., Tyers, M., Boone,C. Friend, S.

(2000). Signaling and circuitry of mutiple maps pathways revealed by a matrix of global gene expression profiles. *Science*, **287**, 873-880.

Schena,M., Shalon,D., Davis,R.W. andBrown,P.O.(1995)Quantitative monitoring of gene expression patterns with a DNA microarray. *Science*, **210**, 467-470.

Spellman, P.T., Sherlock, G., Zhang, M.Q., Iyer,V.R., Anders, K., Eisen, M.B., Brown,P.O., Botstein, D. and Futcher,B. (1998)Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell*, **9**, 3273-3297.

Tamayo, P., Slonim,D., Mesirov,J., Zhu,Q., Kitareewan,S., Dmitrovsky,E., Lander,E. and Golub,T. (1999)Interpreting patterns of gene expression with sely-organizing maps. *Proc. Natl. Acad. Sci. USA*, **96**, 2907-2912.

Wang, Q. H. ((2002)). Identifying Coexpressed Genes with Adjusted Time Course Macroarray Data using Weighting Method. Unpublished manuscript.

Wen,X., Fuhrman, S., Michaels, G. S., Carr, D. B., Smith, S., Barker, J. L. and Somogyi, R. (1998). Large-scale temporal gene expression mapping of central nervous system development. *Proc. Natl Acad. Sci. USA*, **95**, 334-339.

11 Calculating Odds Ratios in Generalized Additive Models including interactions. Application to post-operative infection data.

Javier Roca-Pardiñas, Carmen Cadarso-Suarez, Wenceslao Gonzalez-Manteiga

Expected length of the paper: 20 pages

11.1 Abstract

In many biomedical studies, there is often interest in calculating effect measures in presence of interactions between two exposures that have been measured in a continuous scale. Traditional approaches based on parametric regression models are limited by the degree of arbitrariness involved in transforming these exposures into categorical variables or in imposing a given parametric functional form on the regression function. Recently generalized additive models (GAMs) were proposed to overcome these shortcomings, but there is currently no analytical methods with which to calculate GAM-based association estimates for interactions among continuous exposures. In this work we considered a modified version of the local scoring (with backfitting) algorithm that allows nonparametric estimation of association curves through GAMs with iterations. Procedures for testing second-order interaction terms were also suggested. Backfitting theory is difficult in this context, and bootstrap procedures are therefore provided for estimating the distribution of the test statistics and for the construction of pointwise confidence bands for association curves. Given

the high computational cost involved, binning techniques were used to speed up the computation in the estimation and testing process. The validity of the new methods is supported by the results of a simulation study, and they are illustrated using binary data from a study of possible risk factors for post-operative infection.

12 Survival Trees

Carmela Cappelli and Heping Zhang

12.1 Introduction

Survival trees are a useful regression tool to model the relationship between a survival time and a set of covariates. Survival or censored data are particularly common in medical research, and they also arise from many different areas of scientific and clinical research. For example, in the social sciences, we may be interested in the school drop-out rates and the turnover in a labor market. Tree based methods, due to their nonparametric nature and flexibility, have become very popular in the last two decades as an alternative to the traditional proportional hazard model.

The term *survival data* refers to any data that deals with time to the occurrence of an event of interest. Although the methods developed to cope with survival data are primarily related to medical and biological research, they have their root in insurance statistics and, in general, they are widely used in the social and economic sciences, as well as in engineering. In economics we may study the "survival" of firms or the "survival" of products. For quality control purposes it is a common practice to study the "survival" of electronic components (reliability data analysis, failure time analysis, see Meeker and Escobar (1998)).

In medical research, the event of interest is usually the time to death of a patient after the diagnosis but it might be the time to recovery or remission as well. The main feature of survival data is the presence of incomplete data, which are referred to as *censored observations* and often provide the most relevant information about the phenomenon under study. Censoring can arise from several reasons: the observation time is limited and the study ends before the event is observed for all the subjects, some of the subjects may be lost to follow up the study, subjects are entered at fixed times and the event occurred

before recording. In all these cases, the exact time of the event is not observed. Depending on the direction of the censoring, censored data can be classified into *right censored* when the survival time exceeds the observed one, and *left censored* when the survival time is less than the observed one. Left censoring is particularly important in studies on infectious diseases such hepatitis or HIV (human immunodeficiency) but it will not be discussed here. In the realm of right censored data, a distinction can be made among three different types of censoring:

- Type I censoring: the subjects enter the study at the same time, at a given date the study ends and some of them are lost to follow up or the event is not occurred;
- Type II censoring: the subjects enter the study at the same time, the end of the study is not initially fixed and it is carried on until the event occurs for a certain proportion of subjects;
- Type III censoring: the subjects enter the study at different times.

Figure 12.1 depicts these situations.

Note that Type II is *nonrandom censoring*, whereas Type I and III are *random censoring*.

The circumstance that the survival time cannot be fully observed for all the subjects under study can be formally expressed as follows. Let Y be the observed time and T be the survival time. Without censoring, $Y = T$, i.e., the observed time is the true survival time. With censoring, the observed time is the censoring time denoted by U. A censoring indicator δ takes into account the time being censored, so that $\delta = 1$ if $Y = T$ and $\delta = 0$ otherwise. For the latter, $Y = \min(T, U)$.

There are several important issues involved in the analysis of survival data. They include the comparison of the survival distributions among two or more groups and the identification of predictive variables of survival time. To these ends, parametric, semiparametric and nonparametric methods have been developed. Briefly, parametric methods require specifying a distribution for the survival times (for example Exponential or Weibull). The semi-parametric methods make no assumptions concerning the distributions of the survival times but assume a known form for the effects of the covariates on survivorship. Non-parametric methods make no assumptions on distributions of the survival

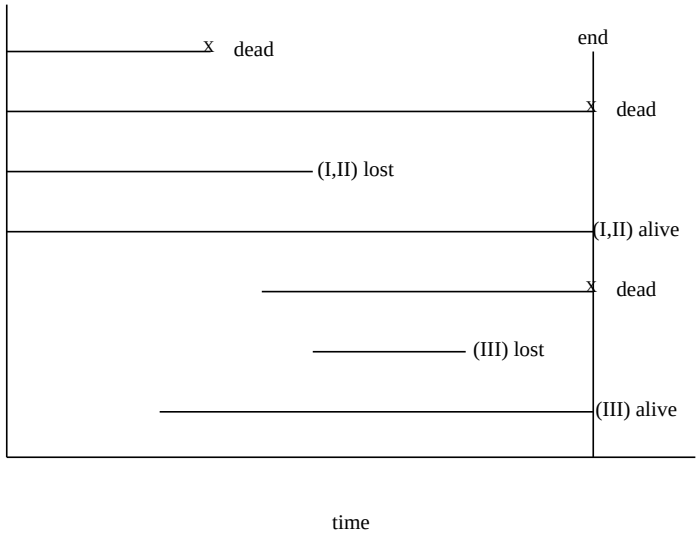

Figure 12.1: The various types of censor data

times. General discussions on various methods can be found in textbooks such as Lee (1992) and Miller (1998).

Among nonparametric methods, tree based methods have become a very popular tool for survival data analysis thanks to the fact that multiple covariates may be associated with the survival time and researchers are commonly interested in identifying subgroups of subjects with similar survival distributions as determined by the covariates.

The XploRe quantlib `hazreg` provides a number of quantlets for the analysis of survival data. We will describe here the quantlet `stree`, which implements the tree based regression method for survival data developed by Zhang (1995) and Zhang and Singer (1999), providing a complete tool to grow, prune and display survival trees.

This chapter is a tutorial for the XploRe `stree` quantlet in the XploRe quantlib

`hazreg` Grund and Yang (2000, Chapter 5), which represent the XploRe implementation of the methodology decribed by Zhang (1995) and Zhang and Singer (1999) and a modification of Heping Zhang's program called STREE. In Section 1, we describe censored survival data. In Section 2, the survival tree methodology is presented. In Section 3, the syntax of the quantlets `stree` is illustrated with some examples.

12.2 Methodology

Any tree based method involves two main steps:

1. *growing* the tree, i.e., partitioning the data (internal nodes) according to a splitting criterion which allows to select the best covariate and cut point along it to split any node;
2. *pruning* the tree, i.e, removing retrospectively some of the branches in order to get a shorter and more accurate tree.

With censored data the survival time is not completely observed for all the subjects and therefore it involves two response variables: the observed time and the censoring indicator defined above. As a consequence, the data are triplets $\{y_i, \delta_i, \mathbf{x}_i\}$, $i = 1, \ldots, n$ where y_i is the observed time for the i^th subjects, δ_i indicates whether y_i is censored and $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})$ is the vector of the p covariates associated with the i-th subject. The events $y_i = t_i$ are called *event times* or *failure times*. It is noteworthy that in this approach, the censoring is assumed to be random (Type I and Type III), so that, given the values of the covariates, the conditional distributions of the survival time and the censoring time are independent.

12.2.1 Splitting criteria

The growing phase is led by the objective of forming a number of homogeneous subsets with respect to the response variable. In order to achieve this aim, the quantlet `stree` allows three splitting criterion as described by Zhang (1995). Two of them are based on an extension of the *impurity measure* introduced by Breiman, Friedman, Olshen and Stone (1984) and the other one is based on the log-rank test statistic.

Impurity based criteria In order to discuss the impurity-based splitting criteria, it is useful to recall some basic concepts and notation in the classification of a multi-class response. Consider a candidate split s of a node t into two offsprings t_l and t_r and let $p(t_l)$ and $p(t_r)$ be the proportions of observations sent by s into node t_l and t_r, respectively. The impurity at node t, denoted by $i(t)$, measures the impurities based on the within-class probabilities. Then, a natural way to evaluate the performances of a candidate split is the change in impurity given by:

$$\Delta i(s,t) = i(t) - \{p(t_l)i(t_l) + p(t_r)i(t_r)\} \tag{12.1}$$

The quantity $\Delta i(s,t)$ is used as a partitioning criterion. This notion of impurity in the case of censored survival data cannot be used as it stands because, although the outcome we are interested in is the survival time, this involves two response variables: the observed time y_i (continuous) and the censoring indicator δ_i (binary). In this respect, a pure node would contain subjects whose observed times are similar and who are in most part censored or uncensored. In other words, a suitable impurity measure for censored data must take account of both observed time and censoring. Therefore the impurity of a node can be expressed as:

$$i(t) = w_1 i_y(t) + w_2 i_\delta(t), \tag{12.2}$$

where w_1 and w_2 are pre-specified weights and $i_y(t)$ and $i_\delta(t)$ denote the impurity of node t for the observed time and censoring, respectively. In particular, the impurity for the time is given by

$$i_y(t) = \sum_{i=1}^{n(t)} \frac{\{y_i - \bar{y}(t)\}^2}{\sum y_i^2} \tag{12.3}$$

where $n(t)$ is the number of observations in node t and $\bar{y}(t)$ is the average of the observed times. The denominator is needed to be normalized with respect to the other component of the impurity. When the summation in the denominator is over node t observations the criterion is called *adaptive normalization*. When it is over the whole sample it is called *global normalization*.

For the impurity of the censoring indicator, it is measured by the entropy measure:

$$i_\delta(t) = -p_t \log(p_t) - (1 - p_t) \log(1 - p_t), \tag{12.4}$$

where p_t denotes the proportion of censored data in node t. Among all the candidate splits at a given node, one split is chosen to maximize the reduction in impurity as measured by 12.1. This simple adaptation of the impurity criterion provides a straightforward way to combine the continuous and categorical outcomes that characterize censored data.

Log-rank statistic criterion The log-rank test statistic is commonly used in the analysis of censored survival data to compare the survival distributions of different groups. For a given covariate and a split point, a 2×2 contingency table is created of the form

Table 12.1: Contingency table for the log-rank statistic.

	Event	
	Yes	No
$x_{ij} \leq s$	a_i	n_i
$x_{ij} > s$	d_i	K_i

where x_{ij} is the value of the j-th covariate for the i-th observation, s is a split point, and K_i is the risk set at time y_i. The log-rank test statistic is defined as:

$$LR(s) = \frac{\sum_i (a_i - E_i)}{\sqrt{\sum_i V_i}} \tag{12.5}$$

where

$$E_i = \frac{d_i n_i}{K_i} \tag{12.6}$$

and

$$V_i = \left\{ \frac{d_i (K_i - n_i) n_i}{K_i (K_i - 1)} \right\} (1 - \frac{d_i}{K_i}). \tag{12.7}$$

Given that the log-rank statistic testes the significance of the difference between two survival distributions, it represents, in a way, a natural choice for splitting the data into two groups with different survivals and it is widely adopted as the splitting criterion Segal (1998), LeBlanc and Crowley (1993), Ciampi and Thiffault (1986).

At a given node t, for every covariate and split point, the log-rank test statistic is computed and the best split s^* is chosen if

$$LR(s^*, t) = \max LR(s, t). \tag{12.8}$$

12.2.2 Pruning

Tree growing, or recursive partitioning, is only one aspect of the tree construction. Tree pruning generally follows tree growing, because of the following two concerns:

1. **complexity** – the long resulting structure tends to be very large; this is especially the case with binary trees since an attribute may reappear (although in a restricted form) many times down the tree;
2. **overfitting** – several branches, especially the terminal ones, reflect particular features of the data arising from the sampling procedure rather than modeling the underlying relationship between the response variable and the covariates.

Therefore, after a large tree T_{max} is grown, a pruning step is carried out in order to simplify the structure and avoid overfitting as discussed in Cappelli, Mola and Siciliano (2002). The quantlet `stree` implements a practical bottom up pruning procedure following the proposal suggested by Segal (1998), which can be described as follows. A statistic S_t (say the log-rank test statistic) is assigned to each internal node t of T_{max}. These statistics are ordered in an increasing order. A threshold is then selected and any internal node whose statistic does not reach the threshold is changed into a terminal node.

The threshold can be fixed by simply considering a significance level. Cutting off the branches stemming from the internal nodes that do not reach the threshold results in a single final pruned tree. A more effective approach that allows

insights into the pruning process is to generate a sequence of nested pruned subtrees of T_{max} in the spirit of the pruning procedure proposed in the CART book (see the XploRe CART tutorial). The sequence is created by iterating the process of locating the minimum value of the statistic and pruning the offsprings of the node(s) that reaches this minimum value. The threshold and therefore the final tree, is selected by plotting the minimal statistics against the size (number of terminal nodes) of the corresponding subtree.

The inspection of the plot allows to select the final tree, in particular, usually the plot shows a "kink" where the pattern changes suggesting that the corresponding tree could be the final one. An important point in the pruning process concerns the assignment of the statistic to the internal nodes. This assignment involves two steps: first, the statistic is computed for all internal nodes; next, the assigned value is replaced with the maximum over the node offsprings if the latter is greater. The sequence, therefore, is created considering the maximized values. In this way the pruning process tends to retain branches that contain sub-branches with higher values of the statistic.

12.3 The Quantlet stree

12.3.1 Syntax

The quantlet `stree` has the following syntax:

```
streeout = stree (covars, time, censor, covartypes, method)
```

grows, prunes and plots the survival tree

with input variables:

`covars` : A $n \times p$ matrix containing observations of covariates,

`time` : A $n \times 1$ vector containing observations of survival time

`censor` : A $n \times 1$ vector containing the censoring indicator,

`covartypes` : specifies the type of covariates

`method` : indicates the splitting criterion.

The arbitrary name `streeout` has been used to indicate the output which includes the following output variables:

`nodenum` : the node number,

`cases` : the number of observations falling into the node,

`dnleft` : the left descendant node number ,

`dnright` : the right descendant node number,

`median` : the median survival time,

`splitvar` : the splitting variable chosen to split the node

`splitval,` splitting values or categories; observations having the variable `splitvar` larger than the value in `splitval` are sent to the right daughter node, otherwise to the left daughter node. For categorical variables `splitval` reports the categories for cases sent into the right descendant.

Optional parameters allows to modify the output presentation. Note that the output of `stree` is shown both in the form of a table and a graphical display.

12.3.2 Example

In order to illustrate the quantlet `stree` the Early Lung Cancer Detection data has been considered; this data set is available at the *Statlib* archive (http://lib.stat.cmu.edu/datasets/csb).

The following variables were recorded: **patient ID** (integer); **institution** (0=Memorial Sloan Kettering, 1=Mayo Clinic, 2=John Opkins); **group** (0=study, 1=controls); **means of detection** (0= routine cytology, 1=routine X-ray, 2=both X-ray and cytology, 3=interval); **cell type** (0=epidermoid, 1=adenocarcinoma, 2=large cell, 3=oat cell, 4= other); **stage**, this variable involves four covariates: **overall stage** (three levels), **tumor** (three levels), **lymph nodes** (three levels), **distant metastases**(two levels), **operated** (0=no, 1=yes); **survival** (days from detection to last date known alive); **survival category**(0=alive, 1=dead of lung cancer, 2=dead of other causes).

The analysis has been restricted to the study group, discarding the controls; also, in the study group, patients dead for other causes than the lung cancer has not been considered so that the subset consist of $n = 475$ patients. The

Table 12.2: Global Normalization before Prune

node #	cases	left nodes	right nodes	median value	split var #	split value
1	475	2	3	805.00	4	3,2
2	183	4	5	1719.00	1	2,1
3	292	6	7	516.50	1	2
4	47	8	9	2282.00	5	2
5	136	10	11	1339.50	8	1
6	78	12	13	1479.50	2	3,2
7	214	14	15	415.50	4	3
8	25	16	17	2772.00	3	4,3
9	22	18	19	1586.00	2	3,2
10	23	20	21	1208.00	5	2
11	113	22	23	1343.00	3	4
12	31	24	25	1336.00	6	2
13	47	26	27	1617.00	6	2
14	28	28	29	490.00	5	2
15	186	30	31	405.00	3	4,1
22	56	32	33	1002.00	2	3,2
23	57	34	35	1720.00	1	2
25	17	36	37	1331.00	3	3
27	36	38	39	1826.50	3	3,2


`XCSstree01.xpl`

following XploRe code reads the original data (file `lung.dat`), deletes the patient ID, creates the subset and the input variables for the quantlet `stree` and runs the quantlet considering as splitting criterion the global normalization.

The results are displayed in Table 12.2 and 12.3, moreover the pruned tree is displayed in Figure 12.2.


`XCSstree01.xpl`

The first discriminant variable selected by the global normalization splitting criterion is the overall stage of the lung cancer, followed by the institution at

Table 12.3: Global Normalization after Prune

node #	cases	left nodes	right nodes	median value	split var #	split value
1	475	2	3	805.00	4	3,2
2	183	4	5	1719.00	1	2,1
3	292	6	7	516.50	1	2
4	47	8	9	2282.00	5	2
5	136	10	11	1339.50	8	1
7	214	14	15	415.50	4	3
11	113	22	23	1343.00	3	4
23	57	34	35	1720.00	1	2


XCSstree01.xpl

both nodes 2 and 3. For example, the split of node 2 separates patients of the Mayo Clinic and of John Hopkins, who are sent to node 5, from patients of The Memorial Sloan Kittering. By setting in the above code the input variable `method=‘‘adaptnorm’’` and `method=‘‘logrank’’` , the other available criteria are used to grow the survival tree, adaptive normalization and log-rank statistic, respectively. Since the different splitting criteria affect the structure of the tree, it is advisable to try them all, selecting the final tree on the basis of scientific judgement.

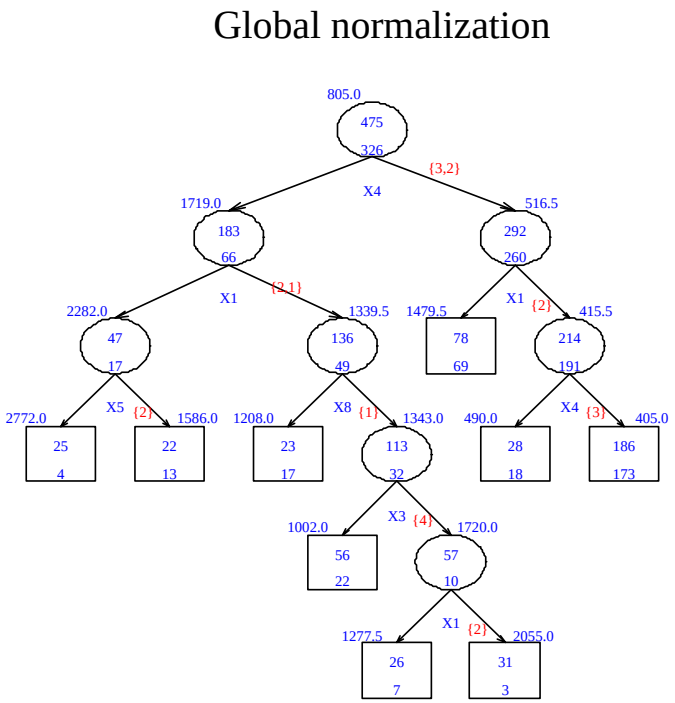

Figure 12.2: The survival tree for Early Lung Cancer Detection Data

Bibliography

Breiman L., Friedman J. H., Olshen R. A. and Stone C. J. (1984). *Classification and Regression Trees*, Wadsworth, Belmont CA.

Cappelli C., Mola F.and Siciliano R.(1984), A statistical approach to growing a honest reliable tree. *Computational Statistics and Data Analysis*, **38** (3), pp. 285–299.

Ciampi, A.and Thiffault, J. (1986), Stratification by stepwise regression, correspondence analysis and recursive partitioning: a comparison of three methods of analysis for survival data with covariates. *Computational Statistics and Data Analysis*, vol **4**, pp. 185–204.

Grund, B. and Yang, L. (2000), Hazard Regression in XploRe, in Härdle, Hlávka and Klinke (eds), *XploRe Application Guide*, Springer.

LeBlanc, M. and Crowley, J. (1993), Survival Trees by Goodness of Split. *Journal of the American Statistical Association*, vol **88**, pp. 457–467.

Lee, E.T. (1989). *Statistical Methods for Survival Data Analysis*. Wiley, New York.

Meeker, W.Q. and Escobar, L. A., (1998). *Statistical Methods for Reliability Data* . John Wiley and Sons, Inc.

Miller, R.G, (1998). *Survival Analysis*. Wiley.

Segal, M. (1998), Regression Trees for Censored Data. *Biometrics*, vol **44**, pp. 35–48.

Zhang, H.P. (1995), Splitting Criteria in Survival Trees, in *Proceedings of the 10-th International Workshop on Statistical Modelling, Innsbruck Austria, July 1995*, pp. 305–314.

Zhang, H.P., and Singer, B. (1999). *Recursive Partitioning in the Health Science*, Springer.

13 Variable Selection in Principal Component Analysis

Yuichi Mori, *Masaya Iizuka*, *Tomoyuki Tarumi* and *Yutaka Tanaka*

While there exist several criteria by which to select a reasonable subset of variables in the context of PCA, we introduce herein variable selection using criteria in Tanaka and Mori (1997)'s modified PCA (M.PCA) among others.

In order to perform such variable selection via XploRe, the quantlib `vaspca`, which reads all the necessary quantlets for selection, is first called, and then the quantlet `mpca` is run using a number of selection parameters.

In the first four sections we present brief explanations of variable selection in PCA, an outline of M.PCA and flows of four selection procedures, based mainly on Tanaka and Mori (1997), Mori (1997), Mori, Tarumi and Tanaka (1998) and Iizuka *et al.* (2002a). In the last two sections, we illustrate the quantlet `mpca` and its performance by two numerical examples.

13.1 Introduction

Consider a situation in which we wish to select items or variables so as to delete the redundant variables or to make a small dimensional rating scale to measure latent traits. Validity requires that all of the variables be included. On the other hand, practical application requires that the number of variables be as small as possible.

There are two types of examples: a clinical test and a plant evaluation. As for the former case, a clinical test for ordinary persons is sometimes not suitable for handicapped persons because the number of checkup items are too large. It is desirable to reduce the number of variables (checkup items) and obtain global scores which can reproduce the information of the original test. As for the latter

case, there are a large number of sensors (checkpoints) in a plant that are used to measure some quantity at each point and evaluate the performance of the entire plant. Exact evaluation requires evaluation based on data measured at all points, but the number of points may be too large to obtain the result within a limited time for temporary evaluation. Therefore, appropriately reducing the number of points to be used in temporary analysis is helpful. For such cases, we meet the problem of variable selection in the context of principal component analysis (PCA).

Let us show another example. In Figure 13.1, the left-hand plot is a scatter plot of the first and second principal components (PCs) obtained based on all 19 original variables, and the right-hand plot is based on seven selected variables. There are not so many differences between the two configurations of PCs. This illustrates the meaningfulness of variable selection in PCA since selected variables can provide almost the same result as the original variables if the goal of the analysis is to observe the configuration of the PCs.

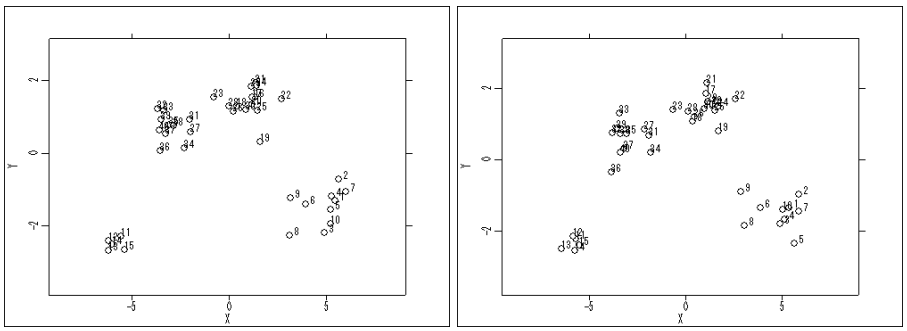

Figure 13.1: Scatter plots of principal component scores based on 19 variables (left) and based on 7 selected variables {3, 7, 13, 15, 16, 17, 18} (right).

vaspca01.xpl

Furthermore we can perform variable selection in PCA as a prior analysis, for example, when the number of original variables is too large for the desired analysis, or as a posterior analysis, for example, when some clusters are obtained and typical variables must be selected from among those in each cluster.

Thus, specifying a subset of variables in the context of PCA is useful in many

practical applications.

13.2 Variable selection in PCA

The problem of variable selection in PCA has been investigated by Jolliffe (1972, 1973), Robert and Escoufier (1976), McCabe (1984), Bonifas *et al.* (1984), Krzanowski (1987a, 1987b), Falguerolles and Jmel (1993), and Mori, Tarumi and Tanaka (1994), among others. These studies sought to obtain ordinary principal components (PCs) based on a subset of variables in such a way that these PCs retain as much information as possible compared to PCs based on all the variables: Jolliffe (1972, 1973)'s methods consider PC loadings, and the methods of McCabe (1984) and Falguerolles and Jmel (1993) use a partial covariance matrix to select a subset of variables, which maintains information on all variables to the greatest extent possible. Robert and Escoufier (1976) and Bonifas *et al.* (1984) used the RV-coefficient and Krzanowski (1987a, 1987b) used Procrustes analysis to evaluate the closeness between the configuration of PCs computed based on selected variables and that based on all variables. Tanaka and Mori (1997) discuss a method called the "modified PCA" (M.PCA) to derive PCs which are computed using only a selected subset of variables but which represent all of the variables, including those not selected. Since M.PCA naturally includes variable selection procedures in the analysis, its criteria can be used directly to detect a reasonable subset of variables (e.g. see Mori (1997, 1998), and Mori, Tarumi and Tanaka (1998)). Furthermore, other criteria can be considered, such as criteria based on influence analysis of variables using the concept reported in Tanaka and Mori (1997) and criteria based on predictive residuals using the concept reported in Krzanowski (1987b) (for details, see Mori *et al.* (1999), Mori and Iizuka (2000) and Iizuka *et al.* (2003).)

Thus, the existence of several methods and criteria is one of the typical characteristics of variable selection in multivariate methods without external variables such as PCA (here the term "external variable" is used as a variable to be predicted or to be explained using the information derived from other variables). Moreover, the existing methods and criteria often provide different results (selected subsets of variables), which is regarded as another typical characterirstic. This occurs because each criterion or PC procedure has its own reasonable purpose of selecting variables. Therefore, we can not say that one is better than the other. These characteristics are not observed in multivariate methods with external variable(s), such as multiple regression analysis.

In practical applications of variable selection, it is desirable to provide computation environment where those who want to select variables can apply a suitable method for their own purposes of selection without difficulties and/or they can try various methods and choose the best method by comparing the results. However, previously, we had no device by which to perform any method easily. In order to provide useful tools for variable selection in PCA, we have developed computation environments in which anyone can easily perform variable selection in PCA using any existing criteria. A windows package "VASPCA (VAriable Selection in PCA)" was initially developed (Mori, 1997) and has been converted to functions for use in general statistical packages, such as R and XploRe. In addition, we have also constructed web-based software using the functions as well as the document pages of variable selection in PCA, see Mori *et al.* (2000a), Iizuka *et al.* (2002a) and also either of the URLs, http://face.f7.ems.okayama-u.ac.jp/~masa/vaspca/indexE.html or http://mo161.soci.ous.ac.jp/vaspca/indexE.html.

13.3 Modified PCA

M.PCA (Tanaka and Mori, 1997) is intended to derive PCs which are computed using only a selected subset but which represent all of the variables, including those not selected. If we can find such PCs which represent all of the variables very well, we may say that those PCs provide a multidimensional rating scale which has high validity and is easy to apply practically. In order to find such PCs we can borrow the concepts of Rao (1964)'s PCA of instrumental variables and Robert and Escoufier (1976)'s RV-coefficient-based approach.

Suppose we obtain an $n \times p$ data matrix Y. If the original data set of Y consists of categorical variables, the data set should be quantified in an appropriate manner (Mori, Tanaka and Tarumi, 1997). Let Y be decomposed into an $n \times q$ submatrix Y_1 and an $n \times (p-q)$ submatrix Y_2 $(1 \leq q \leq p)$. We denote the covariance matrix of $Y = (Y_1, Y_2)$ as $S = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix}$, Y is represented as accurately as possible by r PCs, where r is the number of PCs and the PCs are linear combinations of a submatrix Y_1, i.e. $Z = Y_1 A$ $(1 \leq r \leq q)$. In order to derive $A = (a_1, \ldots, a_r)$, the following criteria can be used:

(Criterion 1) The prediction efficiency for Y is maximized using a linear predictor in terms of Z.

(Criterion 2) The RV-coefficient between Y and Z is maximized. The RV-coefficient is computed as $RV(Y,Z) = \mathtt{tr}(\tilde{Y}\tilde{Y}^\top\tilde{Z}\tilde{Z}^\top)/\{\mathtt{tr}(\tilde{Y}\tilde{Y}^\top)\mathtt{tr}(\tilde{Z}\tilde{Z}^\top)\}^{1/2}$, where $\tilde{Y}$ and $\tilde{Z}$ are centered matrices of Y and Z, respectively.

The maximization criteria for the above (Criterion 1) and (Criterion 2) are given by the proportion P

$$P = \sum_{j=1}^{r} \lambda_j / \mathtt{tr}(\mathtt{S}), \tag{13.1}$$

and the RV-coefficient

$$RV = \left\{ \sum_{j=1}^{r} \lambda_j^2 / \mathtt{tr}(\mathtt{S}^2) \right\}^{1/2}, \tag{13.2}$$

respectively, where λ_j is the j-th eigenvalue, in order of magnitude, of the eigenvalue problem (EVP)

$$[(S_{11}^2 + S_{12}S_{21}) - \lambda S_{11}]\boldsymbol{a} = 0. \tag{13.3}$$

When the number of variables in Y_1 is q, Y_1 should be assigned by a subset of q variables (Y_2 by a subset of $p-q$ remaining variables) which provides the largest value of P in (13.1) for (Criterion 1) or the largest value of RV in (13.2) for (Criterion 2), and the solution is obtained as a matrix A, the columns of which consist of the eigenvectors associated with the largest r eigenvalues of EVP (13.3).

Obviously, these criteria can be used to select a reasonable subset of size q, that is, "variable selection using criteria in M.PCA" is to find a subset of size q by searching for that which has the largest value of the above criterion P or RV among all possible subsets of size q.

13.4 Selection procedures

Although the best method by which to find a subset of variables of size q provides the optimum value for a specified criterion among all possible ${}_pC_q$ combinations of variables, this method is usually impractical due to the high computational cost of computing criterion values for all possible subsets. Therefore,

as practical strategies, Tanaka and Mori (1997) introduced the two-stage *Backward elimination* procedure, and later Mori (1997) proposed three procedures, *Forward selection*, *Backward-forward stepwise selection* and *Forward-backward stepwise selection*, in which only one variable is removed or added sequentially. These procedures allow automatic selection of any number of variables.

Let V be the criterion value P or RV obtained by assigning q variables to Y_1.

Backward elimination

Stage A. Initial fixed-variable stage

- **A-1** Assign q variables to subset Y_1, usually $q := p$.
- **A-2** Solve the EVP (13.3).
- **A-3** Look carefully at the eigenvalues, determine the number r of PCs to be used.
- **A-4** Specify kernel variables which should always be involved in Y_1, if necessary. The number of kernel variables is less than q.

Stage B. Variable selection stage (Backward)

- **B-1** Remove one variables from among q variables in Y_1, make a temporary subset of size $q-1$, and compute V based on the subset. Repeat this for each variable in Y_1, then obtain q Vs. Find the best subset of size $q-1$ which provides the largest V among q Vs and remove the corresponding variable from the present Y_1. Put $q := q-1$.
- **B-2** If the V or q is larger (or smaller) than the preassigned values, go to B-1. Otherwise stop.

Forward selection

Stage A. Initial fixed-variable stage

- **A-1** $\sim$ **3** Same as A-1 to 3 in Backward elimination.
- **A-4** Redefine q as the number of kernel variables (here, $q \geq r$). If you have kernel variables, assign them to Y_1. If not, put $q := r$, find the best subset of q variables which provides the largest V among all possible subsets of size q and assign it to Y_1.

Stage B. Variable selection stage (Forward)

Basically the opposites of Stage B in Backward elimination

Backward-forward stepwise selection

Stage A. Initial fixed-variable stage

A-1 $\sim$ **4** Same as A-1 to 4 in Backward elimination.

Stage B. Variable selection stage (Backward-forward)

B-1 Put $i := 1$.

B-2 Remove one variable from among q variables in Y_1, make a temporary subset of size $q-1$, and compute V based on the subset. Repeat this for each variable in Y_1, then obtain q Vs. Find the best subset of size $q-1$ which provides the largest V (denoted by V_i) among q Vs and remove the corresponding variable from the present Y_1. Set $q := q - 1$.

B-3 If the V or q is larger (or smaller) than preassigned values, go to B-4. Otherwise stop.

B-4 Remove one variable from among q variables in Y_1, make a temporary subset of size $q-1$, and compute V based on the subset. Repeat this for each variable in Y_1, then obtain q Vs. Find the best subset of size $q-1$ which provides the largest V (denoted by V_{i+1}) among q Vs and remove the corresponding variable from the present Y_1. Set $q := q - 1$.

B-5 Add one variable from among $p - q$ variables in Y_2 to Y_1, make a temporary subset of size $q + 1$ and compute V based on the subset. Repeat this for each variable, except for the variable removed from Y_1 and moved to Y_2 in B-4, then obtain $p - q - 1$ Vs. Find the best subset of size $q+1$ which provides the largest V (denoted by V_{temp}) among $p - q - 1$ Vs.

B-6 If $V_i < V_{temp}$, add the variable found in B-5 to Y_1, set $V_i := V_{temp}$, $q := q + 1$ and $i := i - 1$, and go to B-5. Otherwise set $i := i + 1$ and go to B-3.

Forward-backward stepwise selection

Stage A. Initial fixed-variable stage

A-1 to **4** Same as A-1 to 4 in Forward selection.

Stage B. Variable selection stage (Forward-backward)

Basically the opposites of Stage B in Backward-forward stepwise selection

Mori, Tarumi and Tanaka (1998) showed that criteria based on the subsets of variables selected by the above procedures differ only slightly from those based on the best subset of variables among all possible combinations in the case of variable selection using criteria in M.PCA. Mori, Tarumi and Tanaka (1998) also reported that stepwise-type selections (Backward-forward and Forward-backward) can select better subsets than single-type selections (Backward and Forward) and that forward-type selections (Forward and Forward-backward) tend to select better subsets than backward-type selections (Backward and Backward-forward).

13.5 Quantlet

`mpca (x{ ,r})`
performs variable selection using criteria in M.PCA

Before calling the quantlet `mpca`, load quantlib `metrics` by typing:

```
library("metrics")
```

in the input line. This quantlib includes main quantlets such as `mpca` which select subsets of variables automatically and sub quantlets which are used in main quantlets: `geigensm` (solves the generalized EVP), `divide` (divides a matrix Y into two submatrices Y_1 and Y_2), `delcol` (deletes specified columns from the original matrix and generates a new matrix) and other necessary modules for selection.

The quantlet `mpca` has a required argument, a data set X, and an optional argument, the r-number of PCs. If the number of PCs of the data is unknown, type the quantlet only using the first argument, e.g. `mpca(data)`. If known, type the quantlet with both arguments, e.g. `mpca(data, 2)` and then the specification of the second parameter (the number of PCs) will be skipped.

When the `mpca` starts, four parameters are required for selection: a matrix type (covariance or correlation), the number r of PCs ($1 \leq r < p$), a criterion (the

proportion P or the RV-coefficient) and a selection procedure (Backward, Forward, Backward-forward, Forward-backward or All-possible at q). We optionally implemented the All-possible selection procedure at a particular number q of variables to obtain the best subset of that size. Note that computation may take a long time.

After computation based on the specified parameters, two outputs are displayed: a list which indicates the criterion values and variable numbers to be assigned to Y_1 and Y_2 for every number q of selected variables ($r \leq q \leq p$) and a graph which illustrates the change of the criterion value. See the practical actions in the next section.

Note that this quantlet has no function to specify initial variables and the number of variables at the first stage. This quantlet simply selects a reasonable subset of variables automatically as q changes from p to r (or from r to p). In addition, `mpca` performs All-possible selection at the first stage of Forward and Forward-backward procedures to find the initial subset of size r.

13.6 Examples

13.6.1 An artificial data

Here, we apply variable selection using M.PCA criteria to an artificial data set which consists of 87 individuals 20 variables. Suppose the file name of the artificial data set is `artif.dat` and the data set is saved in the folder in which XploRe is installed. Although this data set was generated artificially, the data set was modified in a clinical test (87 observations on 25 qualitative variables) to make the data meaningful.


`XCSvaspca01.xpl`

Based on the specified parameters variable selection is performed. Here, we apply variable selection with the following parameters: correlation matrix, two PCs, the proportion P criterion and Backward procedure. After calculation, the process of removing variables is output in the output window: the criterion value and variable numbers of Y_1 and Y_2 separated by "|" for every number q of selected variables ($q = p, p-1, \ldots, r = 20, 19, \ldots, 2$).

Table 13.1: Variable Selection in Principal Component Analysis
using selection criteria in Modified PCA
Correlation matrix, Proportion P, Backward, r: 2

q	Criterion Value	Y1—Y2																			
20	0.74307	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20|
19	0.74262	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	17	18	19	20|	16
18	0.74212	1	2	3	4	5	6	7	9	10	11	12	13	14	15	17	18	19	20|	8	16
17	0.74160	1	2	3	4	5	6	7	10	11	12	13	14	15	17	18	19	20|	8	9	16
16	0.74105	1	2	4	5	6	7	10	11	12	13	14	15	17	18	19	20|	3	8	9	16
15	0.74026	1	2	4	5	7	10	11	12	13	14	15	17	18	19	20|	3	6	8	9	16
14	0.73931	1	2	4	5	7	10	11	13	14	15	17	18	19	20|	3	6	8	9	12	16
13	0.73826	1	2	4	5	7	10	11	13	14	15	17	18	20|	3	6	8	9	12	16	19
12	0.73650	1	2	4	5	7	11	13	14	15	17	18	20|	3	6	8	9	10	12	16	19
11	0.73467	1	2	4	5	7	11	13	14	17	18	20|	3	6	8	9	10	12	15	16	19
10	0.73276	1	4	5	7	11	13	14	17	18	20|	2	3	6	8	9	10	12	15	16	19
9	0.73010	1	4	5	7	13	14	17	18	20|	2	3	6	8	9	10	11	12	15	16	19
8	0.72736	1	4	7	13	14	17	18	20|	2	3	5	6	8	9	10	11	12	15	16	19
7	0.72372	1	4	7	13	14	17	18|	2	3	5	6	8	9	10	11	12	15	16	19	20
6	0.71891	1	4	7	13	14	17|	2	3	5	6	8	9	10	11	12	15	16	18	19	20
5	0.71329	1	7	13	14	17|	2	3	4	5	6	8	9	10	11	12	15	16	18	19	20
4	0.70133	7	13	14	17|	1	2	3	4	5	6	8	9	10	11	12	15	16	18	19	20
3	0.68264	13	14	17|	1	2	3	4	5	6	7	8	9	10	11	12	15	16	18	19	20
2	0.65580	13	17|	1	2	3	4	5	6	7	8	9	10	11	12	14	15	16	18	19	20

r: number of principal components, q: number of selected variables,
Y_1: subset of variables to be selected, and Y_2: subset of variables to be deleted

The graph of criterion values is also displayed (Figure 13.2). You can observe the change in the criterion visually using this graph.

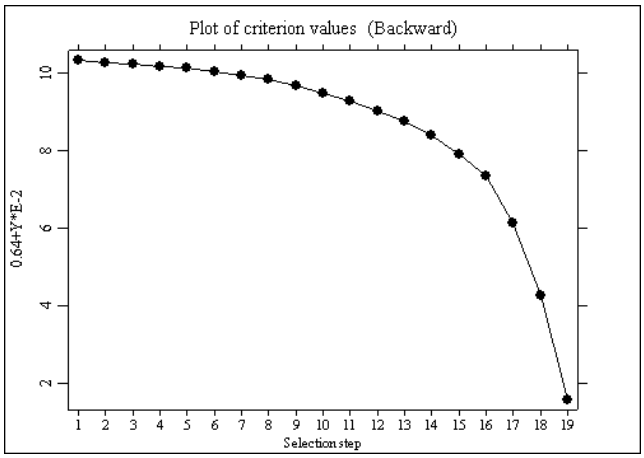

Figure 13.2: Index plot of the proportion Ps as q changes from 20 to 2. (Artificial data, r=2, correlation matrix, the proportion P and Backward)

These outputs show that the proportion P changes slightly until the number of variables is six (at step 15). The range of the proportion P's is only 0.02416 ($= 0.74307 - 0.71891$). This means that 14 of the 20 variables are almost redundant for composing PCs to be used to reproduce the original variables. Furthermore, if a subset of size 11 or more is selected, the difference between the proportion based on the selected subset and that based on all of the variables is less than 0.01.

Looking at the results, a subset of any number of variables displayed as `Y1` can be selected in the output list.

Here, we show another result obtained by applying Forward-backward stepwise selection to the same data set. The index plot of the criterion value is illustrated in Figure 13.3 and selected variables are

Table 13.2: Variable Selection in Principal Component Analysis
using selection criteria in Modified PCA
Correlation matrix, Proportion P, Forward-backward, r: 2

q	Criterion Value	Y1 — Y2																		
2	0.65658	4	16|	1	2	3	5	6	7	8	9	10	11	12	13	14	15	17	18	19	20
3	0.68161	4	11	15|	1	2	3	5	6	7	8	9	10	12	13	14	16	17	18	19	20
4	0.69858	4	11	13	15|	1	2	3	5	6	7	8	9	10	12	14	16	17	18	19	20
5	0.71210	3	10	11	15	20|	1	2	4	5	6	7	8	9	12	13	14	16	17	18	19
6	0.72047	3	5	10	11	15	20|	1	2	4	6	7	8	9	12	13	14	16	17	18	19
7	0.72514	3	5	9	10	11	15	20|	1	2	4	6	7	8	12	13	14	16	17	18	19
8	0.72944	1	3	4	5	10	11	15	20|	2	6	7	8	9	12	13	14	16	17	18	19
9	0.73298	1	3	4	5	10	11	15	16	20|	2	6	7	8	9	12	13	14	17	18	19
10	0.73480	1	3	4	5	10	11	13	15	16	20|	2	6	7	8	9	12	14	17	18	19
11	0.73660	1	3	4	5	7	10	11	13	15	16	20|	2	6	8	9	12	14	17	18	19
12	0.73766	1	3	4	5	8	9	10	11	13	15	16	20|	2	6	7	12	14	17	18	19
13	0.73886	1	3	4	5	8	9	10	11	13	15	16	19	20|	2	6	7	12	14	17	18
14	0.73995	1	3	4	5	6	8	9	10	11	13	15	16	19	20|	2	7	12	14	17	18
15	0.74087	1	3	4	5	6	8	9	10	11	13	15	16	18	19	20|	2	7	12	14	17
16	0.74131	1	3	4	5	6	8	9	10	11	12	13	15	16	18	19	20|	2	7	14	17
17	0.74179	1	2	3	4	5	6	8	9	10	11	12	13	15	16	18	19	20|	7	14	17
18	0.74223	1	2	3	4	5	6	7	8	9	10	11	12	13	15	16	18	19	20|	14	17
19	0.74262	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	17	18	19	20|	16
20	0.74307	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20|

r: number of principal components, q: number of selected variables
Y_1: subset of variables to be selected, Y_2: subset of variables to be deleted

r : number of principal components, q : number of selected variables Y1: subset of variables to be selected, Y2: subset of variables to be deleted

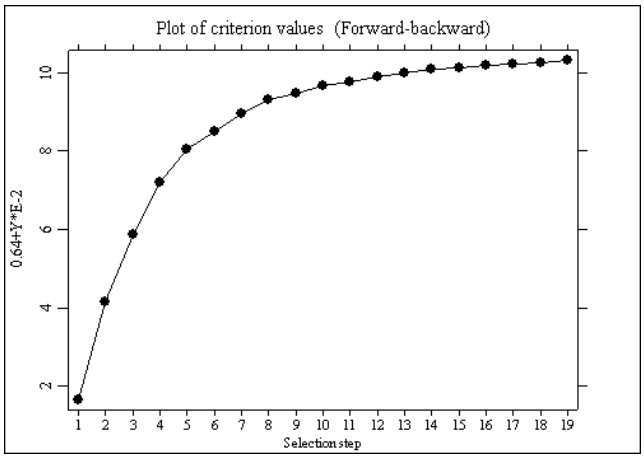

Figure 13.3: Index plot of the proportion Ps as q changes from 2 to 20. (Artificial data, r=2, correlation matrix, the proportion P and Forward-backward)

The outputs are displayed in selected order (in reverse order of backward-type selection). Although stepwise-type selection takes longer than single-type selection, stepwise-type selection can provide more reasonable results. In fact, when the number of variables is six, for example, the selected subset {3, 5, 10, 11, 15, 20} is the same result as that obtained by All-possible selection (see the result of All-possible selection described below).

If you choose All-possible at a specified q in the fourth selection box, one additional box opens to specify the number of variables to be investigated (Figure **??**). Then, the best subset of the specified size q is displayed in the output window:

Table 13.3: Variable Selection in Principal Component Analysis
using selection criteria in Modified PCA
Correlation matrix, Proportion P, All-possible at a specified q, r: 2

q	Criterion Value	Y1 — Y2
6	0.72047	3 5 10 11 15 20| 1 2 4 6 7 8 9 12 13 14 16 17 18 19

r: the number of principal components, q: the number of selected variables
Y_1: a subset of variables to be selected, Y_2: a subset of variables to be deleted

If `mpca` is called using the second argument, for example, `mpca(artif, 2)`, solving the prior EVP and the second selection to specify the number of PCs are skipped.

13.6.2 Application data

As the second numerical example, we analyze a data set of alate adelges (winged aphids), which was analyzed originally by Jeffers (1967) using ordinary PCA and later by various authors, including Jolliffe (1973) and Krzanowski (1987a, 1987b), using PCA with variable selection functions. We applied our variable selection method to the data set given in Krzanowski (1987a). The data set consists of 40 individuals and 19 variables. Eigenvalues and their cumulative proportions of the data are 13.8379 (72.83%), 2.3635 (85.27%), 0.7480 (89.21%), ..., therefore we use two PCs as in previous studies. Since Jeffers (1967) found four clusters by observing the plot of PCs obtained by ordinary PCA based on the correlation matrix of whole variables, we choose the RV-coefficient as a selection criterion to detect a subset providing the close configuration of PCs to the original configuration. Here, we apply Forward-backward stepwise selection based on the correlation matrix to the data.

The results of (Y_1, Y_2) for every q are obtained as the following output and their RV-coefficients changes as shown in Figure 13.4.

Table 13.4: Variable Selection in Principal Component Analysis
using selection criteria in Modified PCA
Correlation matrix, RV-coefficient, Forward-backward, r: 2

q	Criterion Value	Y1 — Y2																	
2	0.97069	5	13|	1	2	3	4	6	7	8	9	10	11	12	14	15	16	17	18	19
3	0.98413	5	13	18|	1	2	3	4	6	7	8	9	10	11	12	14	15	16	17	19
4	0.98721	5	7	13	18|	1	2	3	4	6	8	9	10	11	12	14	15	16	17	19
5	0.99066	3	7	13	17	18|	1	2	4	5	6	8	9	10	11	12	14	15	16	19
6	0.99198	3	7	13	16	17	18|	1	2	4	5	6	8	9	10	11	12	14	15	19
7	0.99311	3	7	13	15	16	17	18|	1	2	4	5	6	8	9	10	11	12	14	19
8	0.99371	3	4	7	13	15	16	17	18|	1	2	5	6	8	9	10	11	12	14	19
9	0.99435	3	4	5	10	13	15	16	17	18|	1	2	6	7	8	9	11	12	14	19
10	0.99496	3	4	5	10	11	13	15	16	17	18|	1	2	6	7	8	9	12	14	19
11	0.99540	1	5	6	9	10	11	13	15	17	18	19|	2	3	4	7	8	12	14	16
12	0.99593	1	4	5	6	9	10	11	13	15	17	18	19|	2	3	7	8	12	14	16
13	0.99634	1	4	5	6	9	10	11	13	15	16	17	18	19|	2	3	7	8	12	14
14	0.99671	1	4	5	6	8	9	10	11	13	15	16	17	18	19|	2	3	7	12	14
15	0.99693	1	3	4	5	6	8	9	10	11	13	15	16	17	18	19|	2	7	12	14
16	0.99704	1	2	3	4	5	6	8	9	10	11	13	15	16	17	18	19|	7	12	14
17	0.99712	1	2	3	4	5	6	8	9	10	11	12	13	15	16	17	18	19|	7	14
18	0.99723	1	2	3	4	5	6	7	8	9	10	11	12	14	15	16	17	18	19|	13
19	0.99726	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19|

r: number of principal components, q: number of selected variables
Y_1: subset of variables to be selected, Y_2: subset of variables to be deleted

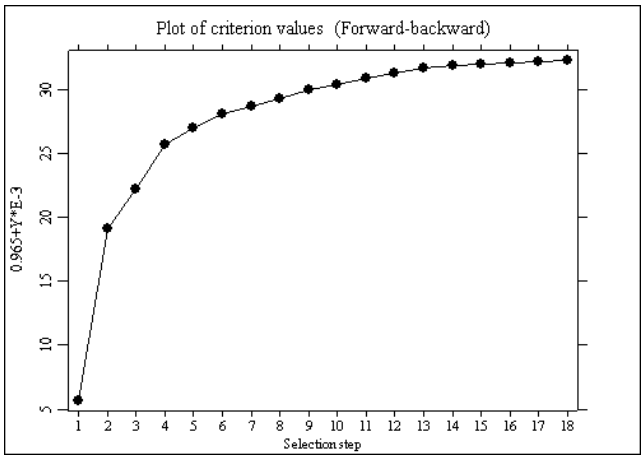

Figure 13.4: Index plot of the RV-coefficients as q changes from 2 to 19. (Alate data, r=2, correlation matrix, the RV-coefficient and Forward-backward)

The results illustrate that the RV-coefficient changes slightly when the number of variables is over five (at step 4). In particular, the sequential difference is less than 0.0007 when the number of variables is over 7 (step 6).

Here, we draw a scatter plot of PC scores based on the seven selected variables {3, 7, 13, 15, 16, 17, 18} and compare this plot with that based on the 19 original variables.

XCSvaspca02.xpl

Using these arguments in the quantlet `geigensm` to solve the generalized EVP (13.3), we obtain the sorted eigenvalues `mevp.values` and the associated eigenvectors `mevp.vectors`. Thus, the modified PC scores `mpc` are obtained after scale adjustment. The last block draws two scatter plots of the first two PC scores. These are shown in Figure 13.1 in Section 13.1 (The figures can be rotated and the first three PCs can be observed as the three-dimensional display by mouse operation. Note, however, that the modified PCs were calculated as the number of PCs is two).

As the plots illustrate, little difference exists between the two configurations, i.e. the use of only seven among 19 variables is sufficient to obtain PCs that

provide almost the same information as the original PCs.

Bibliography

Bonifas, I., Escoufier, Y., Gonzalez, P.L. et Sabatier, R. (1984).

Choix de variables en analyse en composantes principales, *Rev. Statist. Appl.*, **23**: 5-15.

Falguerolles, A. De et Jmel, S. (1993). Un critere de choix de variables en analyse en composantes principales fonde sur des modeles graphiques gaussiens particuliers, *Rev. Canadienne Statist.*, **21**(3): 239-256.

Iizuka, M., Mori, Y., Tarumi, T. and Tanaka, Y. (2002a). Implementation of variable selection program for principal component analysis to WWW, *Proceedings of the Institute of Statistical Mathematics*, **49**(2): 277-292. (in Japanese).

Iizuka, M., Mori, Y., Tarumi, T. and Tanaka, Y. (2002b). Statistical software VASMM for variable selection in multivariate methods, In: W. Härdle and B. Rönz (eds.) *COMPSTAT2002 Proceedings in Computational Statistics*, Springer-Verlag, pp.563-568.

Iizuka, M., Mori, Y., Tarumi, T. and Tanaka, Y. (2003). Computer intensive trials to determine the number of variables in PCA, *Journal of the Japanese Society of Computational Statistics*, **14**(2) (Special Issue of ICNCB). (to appear).

Jeffers, J. N. R. (1967). Two case studies in the application of principal component analysis, *Applied Statistics*, **16**: 225–236.

Jolliffe, I. T. (1972). Discarding variables in a principal component analysis I - Artificial data -, *Applied Statistics*, **21**: 160-173.

Jolliffe, I. T. (1973). Discarding variables in a principal component analysis II - Real data -, *Applied Statistics*, **22**: 21-31.

Krzanowski, W. J. (1987a). Selection of variables to preserve multivariate data structure, using principal components, *Applied Statistics*, **36**: 22-33.

Krzanowski, W. J. (1987b). Cross-validation in principal component analysis, *Biometrics*, **43**: 575-584.

McCabe, G. P. (1984). Principal variables, *Technometrics*, **26**: 137-44.

Mori,Y. (1997). Statistical software VASPCA - Variable selection in PCA -, *Bulletin of Okayama University of Science,* **33**(A): 329-340.

Mori, Y. (1998). Principal component analysis based on a subset of variables - Numerical investigation using RV-coefficient criterion -, *Bulletin of Okayama University of Science*, **34**(A): 383-396. (in Japanese).

Mori, Y. and Iizuka, M. (2000). Study of variable selection methods in data analysis and its interactive system, *Proceedings of ISM Symposium - Recent Advances in Statistical Research and Data Analysis* -: 109-114.

Mori, Y., Iizuka, M. Tarumi, T. and Tanaka, Y. (1999). Variable selection in "Principal Component Analysis Based on a Subset of Variables", *Bulletin of the International Statistical Institute* (*52nd Session Contributed Papers Book2*): 333-334.

Mori, Y., Iizuka, M. Tarumi, T. and Tanaka, Y. (2000a). Statistical software "VASPCA" for variable selection in principal component analysis, In: W. Jansen and J.G. Bethlehem (eds.) *COMPSTAT2000 Proceedings in Computational Statistics* (*Short Communications*), pp.73-74.

Mori, Y., Iizuka, M., Tarumi, T. and Tanaka, Y. (2000b). Study of variable selection criteria in data analysis, *Proceedings of the 10th Japan and Korea joint Conference of Statistics*: 547-554.

Mori, Y., Tarumi, T. and Tanaka, Y. (1994). Variable selection with RV-coefficient in principal component analysis, In: R. Dutter and W. Grossman (eds.), *COMPSTAT1994 Proceedings in Computational Statistics* (*Short Communications*), pp.169-170.

Mori, Y., Tanaka, T. and Tarumi, T. (1997). Principal component analysis based on a subset of qualitative variables, In: C. Hayashi `et al.` (eds.), *Proceedings of IFCS-96: Data Science, Classification and Related Methods*, Springer-Verlag, pp. 547-554.

Mori, Y., Tarumi, T. and Tanaka, Y. (1998). Principal Component analysis based on a subset of variables - Numerical investigation on variable selection procedures -, *Bulletin of the Computational Statistics of Japan*, **11**(1): 1-12. (in Japanese)

Rao, C. R. (1964). The use and interpretation of principal component analysis in applied research, *Sankhya*, **A26**: 329-358.

Robert, P. and Escoufier, Y. (1976). A unifying tool for linear multivariate statistical methods: the RV-coefficient, *Appl. Statist.*, **25**: 257-65.

Tanaka, Y. and Mori, Y. (1997). Principal component analysis based on a subset of variables: Variable selection and sensitivity analysis, *American Journal of Mathematics and Management Sciences*, **17**(1&2): 61-89.

14 A semiparametric approach to estimate reference curves for biophysical properties of the skin

Saracco Jérôme, *Ali Gannoun*, *Christiane Guinot*, *Benoît Liquet*

Expected length of the paper: 20 pages

14.1 Abstract

Reference curves which take one covariable into account such as the age, are often required in medicine, but simple systematic and efficient statistical methods for constructing them are lacking. Classical methods are based on parametric fitting (polynomial curves). In this chapter, we propose a new methodology for the estimation of reference intervals for data sets, based on nonparametric estimation of conditional quantiles. The derived method should be applicable to all clinical or more generally biological variables that are measured on a continuous quantitative scale. To avoid the curse of dimensionality when the covariate is multidimensional, a new semiparametric procedure is also proposed. This procedure is based on dimension-reduction and nonparametric estimation of conditional quantiles as previously introduced. This semiparametric approach combines sliced inverse regression (SIR) and a kernel estimation of conditional quantiles. The usefulness of these nonparametric and semiparametric estimation procedures are illustrated on a real data set collected in order to establish reference curves for biophysical properties of the skin of healthy French women.

15 Survival Analysis

Makoto TOMITA

.

15.1 Abstract

This chapter explains the technique of the fundamental survival time analysis using XploRe. Kaplan-Meier estimator is mentioned as the typical technique of non-parametric survival time analysis. The most common estimate of the survival distribution, the Kaplan-Meier estimate, is a product of survival proportions. It produces non-parametric estimates of failure probability distributions for a single sample of data that contains the exact time of failure, or contains data is right censored. It calculates about a proportion surviving, and a survival time, then it is plotted a Kaplan-Meier survival curve.

Some methods are proposed about approval of the difference of the survival time of two groups. Log-rank test is the approval method which applied Kaplan-Meier estimate. This tests the difference of survival proportions as the whole.

And Cox regression using a proportional hazard rate is indispensable in this latest field. It is one of semi-parametric survival time analyzing methods. Cox's proportional hazard model is multiple linear regression analysis considered by survival time can be taken to the response variable Y and explanatory variable the factor X. And hazard rate is applied to variable Y. An effect of treatment is given by a coefficient β on multiple linear regression analysis. Then we want to evaluate β.

These techniques are explained applying to data using XploRe.

Part II

Geostatistics

16 Spatial Statistics

Pavel Čížek, Wolfgang Härdle and Jürgen Symanzik

16.1 Introduction

In spatial statistics, we deal with spatial data, i.e., data collected in a particular region such as a country, multiple states, etc. Examples for such data sets are economic data, environmental data, or medical data. For spatial data, we typically consider a spatial correlation; observations from nearby locations are similar. The field of spatial statistics provides techniques that allow to deal with spatially correlated data. Our intention is to provide with this text an introduction into spatial statistics. Here, we describe practical problems in the field of medicine, and agronomics. For a detailed overview of spatial statistics, the reader is referred to Ripley (1981) or Cressie (1993).

We concentrate on two areas of specialization: (i) Spatial interpolation, smoothing, and kriging; and (ii) Spatial point process analysis.

16.2 Spatial Interpolation, Smoothing, and Kriging

```
myres = SPKRsurfls (np, xmat)
    fits a trend surface, i.e., a polynomial regression
    surface, by least squares

myres = SPKRsurfgls (np, covmod, xmat, nx, dval, alpha,
    se {, D}
    fits a trend surface by generalized least squares

covvals = SPKRexpcov (r, d, alpha, se)
    spatial covariance function for use with SPKRsurfgls

covvals = SPKRgaucov (r, d, alpha, se)
    spatial covariance function for use with SPKRsurfgls

covvals = SPKRsphercov (r, d, alpha, se {, D})
    spatial covariance function for use with SPKRsurfgls

mygrid = SPKRtrmat (obj, xl, xu, yl, yu, n)
    evaluates a trend surface over a grid

mygrid = SPKRprmat (obj, xl, xu, yl, yu, n)
    evaluates a kriging surface over a grid

mygrid = SPKRsemat (obj, xl, xu, yl, yu, n {, se})
    evaluates a kriging standard error of prediction
    surface over a grid

corres = SPKRcorrelogram (krig, nint)
    computes spatial correlograms of spatial data or
    residuals

varres = SPKRcorrelogram (krig, nint)
    computes spatial (semi-)variograms of spatial data or
    residuals

cont = SPKRmultcontours (disp, pos1, pos2, obj, start,
    end, step)
    draws multiple contour lines of a spatial object of
    type "trmat", "prmat", or "semat"
```

Quantlets related to spatial interpolation, smoothing, and kriging start with the letters SPKR. The presented spatial statistics quantlets have been adapted from Venables and Ripley (1999). In fact, the C-code from Venables and Ripley (1999) has been linked to XploRe through a DLL. The sample data sets, `topo.dat` and `pines.dat` have been taken from that book as well. Please check this reference and the related Web site at http://www.stats.ox.ac.uk/pub/MASS3/ for more details.

The examples in this Section show how to do the computations and produce the graphics from Sections 14.1 and 14.2 in Venables and Ripley (1999) using XploRe.

16.2.1 Trend Surfaces

Our first example shows how to fit trend surfaces of order np, i.e., polynomial regression surfaces, to a data set. In this example, we calculate trend surfaces of order 2, 3, 4, and 6 for the `topo.dat` data set. The results are displayed as contours on Figure 16.1, which is similar to Figure 14.1 in Venables and Ripley (1999). Obviously, the higher-order surfaces show considerable problems near the edges due to extrapolation.

This XploRe code results in the following graphical display that is similar to Figure **??** in Venables and Ripley (1999).

16.2.2 Kriging

In the next step, we look at a trend surface based on least squares, a trend surface based generalized least squares, and a kriged surface and its standard error of prediction, based on the `topo.dat` data set. The results are displayed as contours. The XploRe result displayed in Figure is similar to Figure 14.5 in Venables and Ripley (1999).

16.2.3 Correlogram and Variogram

We now show how to construct a correlogram and variogram for the residuals of the `topo.dat` data set, based on a least squares quadratic trend surface. The result on Figure 16.3 is similar to Figure 14.6 in Venables and Ripley (1999).

Next, we look at two different covariance structures, see Figure 16.4. In the

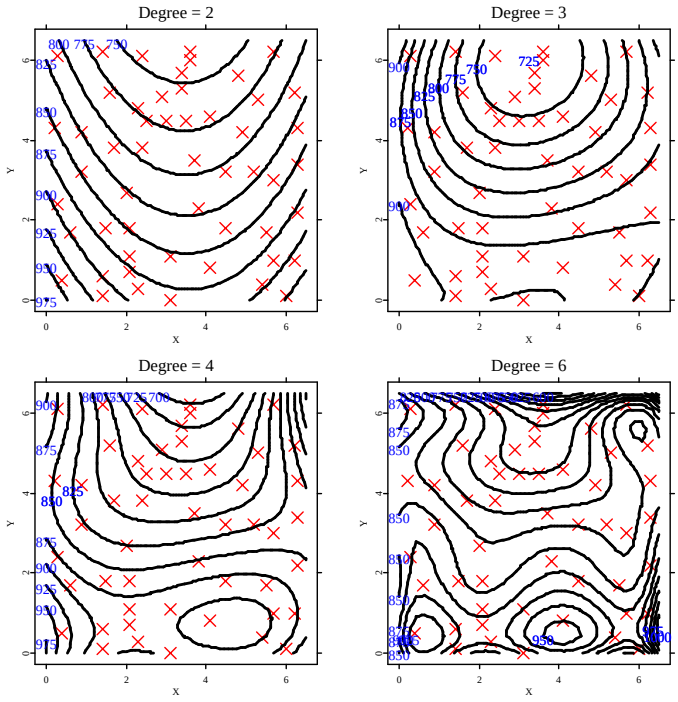

Figure 16.1: Trend surfaces of various orders for `topo.dat`

XCSspa01.xpl

left plot, we construct two correlograms for the `topo.dat` data set - one with residuals from a quadratic trend surface showing an exponential covariance (solid blue line) and a Gaussian covariance (dashed cyan line) function. The right plot shows the raw `topo.dat` data set with a fitted Gaussian covariance function. The discussed plot is again similar to Figure 14.7 in Venables and Ripley (1999).

Finally, we look at two more kriged surfaces and standard errors of prediction for the `topo.dat` data set, see Figure 16.5, which is similar to Figure 14.8 in Venables and Ripley (1999). In the top row, we use a quadratic trend surface

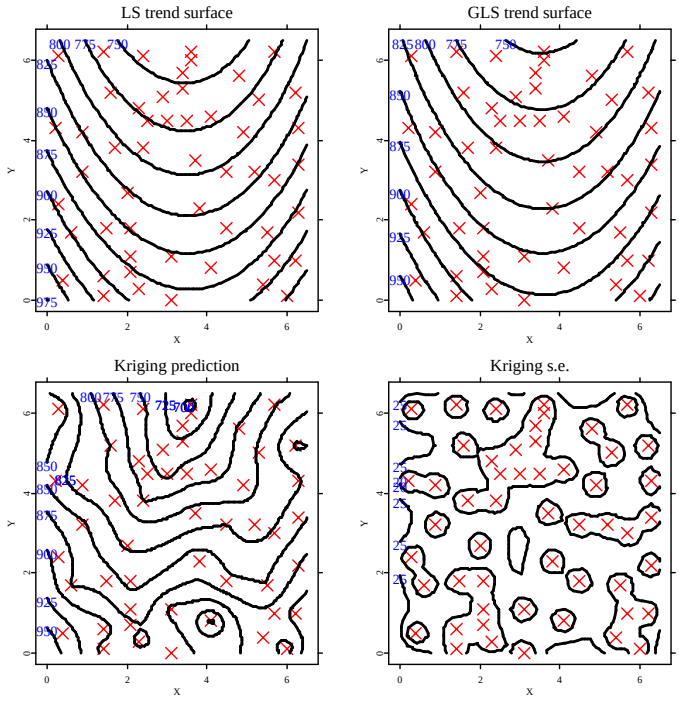

Figure 16.2: Trend and kriged surfaces for `topo.dat`

`XCSspa02.xpl`

and a nugget effect. The bottom row is without a trend surface.

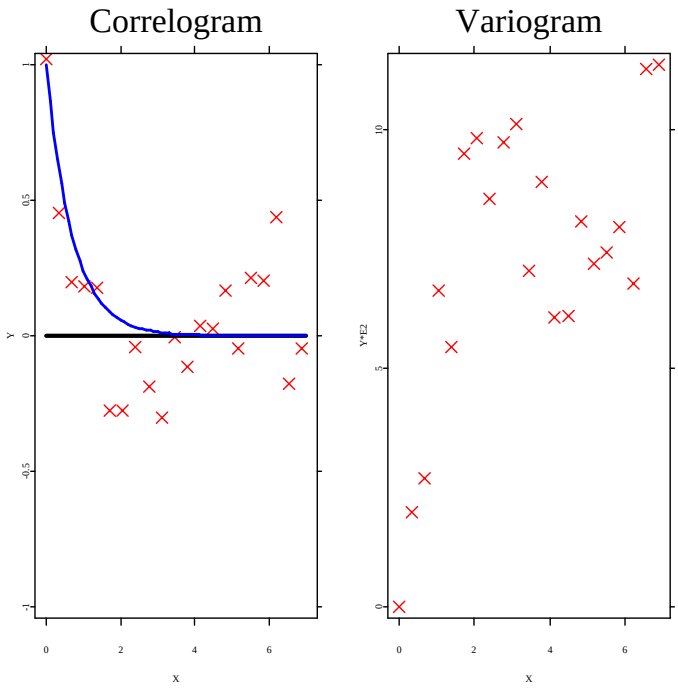

Figure 16.3: Correlogram and Variogram for `topo.dat` data set

`XCSspa03.xpl`

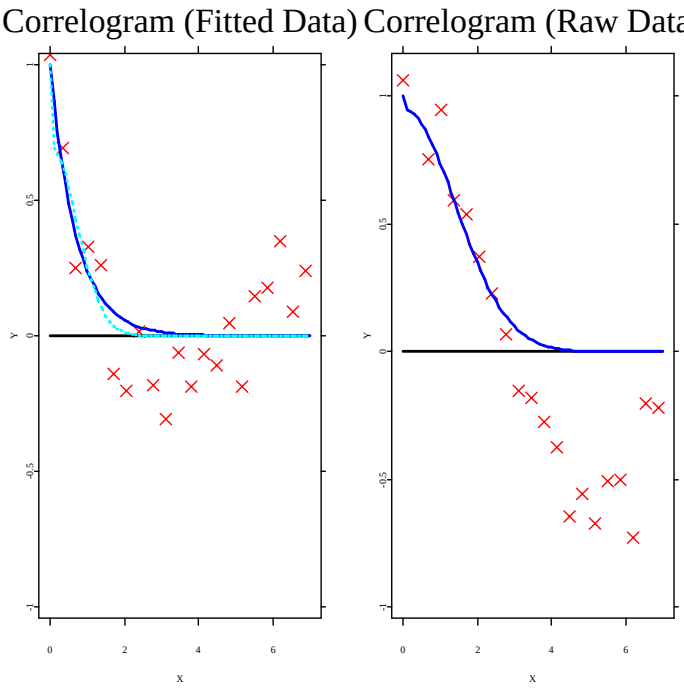

Figure 16.4: Correlogram for fitted and raw data `topo.dat`

XCSspa04.xpl

16.3 Spatial Point Process Analysis

```
ppobj = SPPPinit (pp, xl, xu, yl, yu, fac)
    creates a point process object and calls
    SPPPsetregion to set the rectangular spatial domain

SPPPinitrandom (rstart)
    resets the random number generator for point
    processes

SPPPsetregion (pp)
    sets the rectangular spatial domain for spatial point
    pattern analysis

area = SPPPgetregion ()
    retrieves the rectangular spatial domain that
    previously has been set by SPPPinit or SPPPsetregion

ppkfn = SPPPkfn (pp, fs, k)
    computes K-fn of a point pattern

ppsim = SPPPpsim (nsim, n)
    simulates a Binomial (Poisson) spatial point process

ppstrauss = SPPPstrauss (nsim, n, c, r)
    simulates a Strauss spatial point process

ppssi = SPPPssi (nsim, n, r)
    simulates a SSI (sequential spatial inhibition) point
    process
```

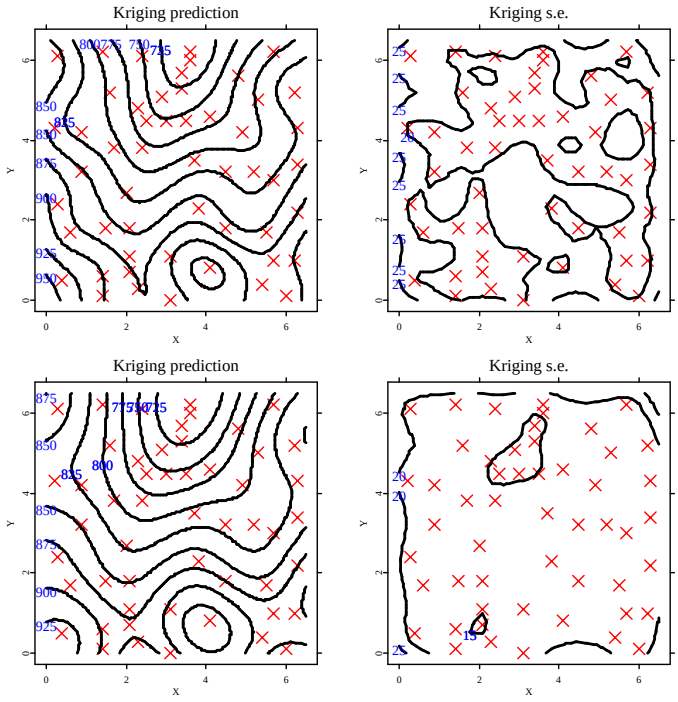

Figure 16.5: Trend and kriged surfaces for `topo.dat`

`XCSspa05.xpl`

Quantlets related to spatial point process analysis start with the letters SPPP.

The examples in this Section show how to do the computations and produce the graphics from Section 14.3, i.e., Figure 14.9, in Venables and Ripley (1999), using XploRe.

After initializing the `pines.dat` data set, we first draw the raw data in the upper left plot (see Figure 16.6).

We now make 100 simulation runs of a Binomial process with 72 observations that inhibit the same spatial domain as the original data. We draw the result

of the first simulation run in the upper center plot and the envelope of L(t) of these simulation runs in the upper right plot. Here, $L(t) = \sqrt{K(t)/\pi}$, where K represents Ripley's K function. For a Poisson process $K(t) = \pi t^2$, thus $L(t)$ will be linear for a Poisson process. The solid black line is $L(t)$ for the `pines.dat` data set. Obviously, a Binomial process does not fit the `pines.dat` data set.

In the middle plots, we consider a Strauss process as a possible alternative. The middle left plot shows a one run of a $Strauss(72, 0.15, 0.7)$ simulation. In the middle center plot, we draw the envelope of $L(t)$ of 100 of these simulation runs. The solid black line is $L(t)$ for the `pines.dat` data set. The solid cyan lines represents the averages of the simulation runs.

In addition, we also conduct 100 simulation runs of a $Strauss(72, 0.2, 0.7)$ process. Similarly, we draw the envelope of $L(t)$ of these simulation runs, the averages, and the result from the `pines.dat` data set. Obviously, both processes describe the data reasonably well.

Just for illustrative purposes, we also look at 100 simulation runs of Matern's sequential spatial inhibition (SSI) process in the bottom row. The result of the first simulation run are displayed in the lower left plot. The lower center and lower right plots display the envelope of $L(t)$ of 100 of these simulation runs. The solid black line is $L(t)$ for the `pines.dat` data set. The solid cyan lines represents the averages of the simulation runs.

Obviously, this $SSI(72, 0.7)$ does not fit the `pines.dat` data set.

And here is the result of all these plots:

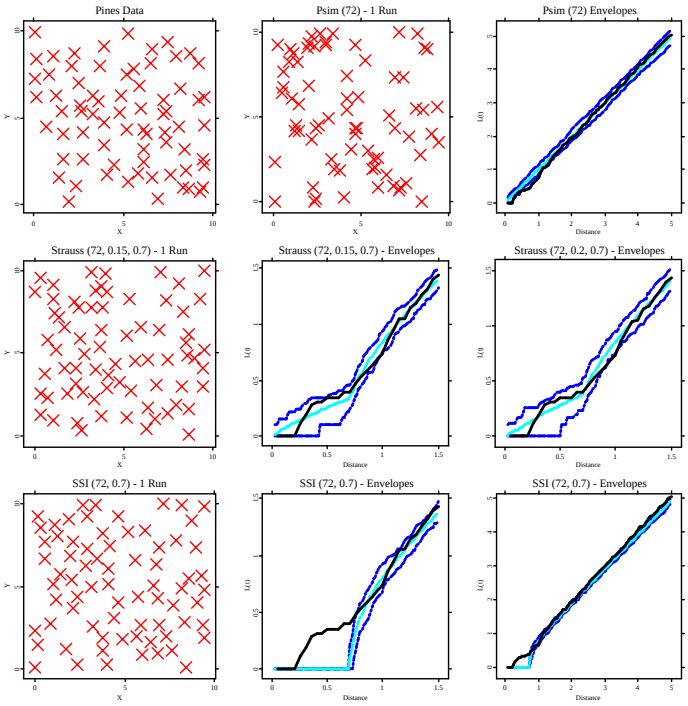

Figure 16.6: Spatial analysis of `pines.dat`

XCSspa06.xpl

Bibliography

Cressie, N. A. C. (1993). *Statistics for Spatial Data (Revised Edition)*, Wiley, New York, NY.

Ripley, B. D. (1981). *Spatial Analysis*, Wiley, New York, NY.

Venables, W.N. and Ripley, B.D. (1999). *Modern Applied Statistics with S-Plus (Third Edition)*, Springer, New York, NY.

17 Functional Data Analysis

Yoshihiro Yamanishi

17.1 Introduction

Functional data analysis (FDA) has been developed for analyzing functional (or curve) data. In FDA, we treat the data that consist of functions not of vectors. We take samples at time points $t_1, t_2, \ldots$ and regard $\{x(t_j), j = 1, 2, \ldots\}$ as multivariate observations. In this sense the original functional $x(t)$ can be regarded as the limit of $\{x(t_j)\}$ as the sampling interval tends to zero and the dimension of multivariate observations tends to infinity. Ramsay and Silverman (1997) have discussed several methods for analyzing functional data, including functional regression analysis, functional principal component analysis (PCA), and functional canonical correlation analysis (CCA). These methodologies look attractive, because we often meet the cases where we wish to apply regression analysis and principal component analysis to such data. In the following we describe how to use the FDA tools for applying functional data analysis.

17.1.1 Basis Expansion

In practice we usually obtain sampled data such as $\{x(t_j), j = 1, 2, \ldots\}$. So at the first stage of a general functional data analysis we must transform the data into a functional form such as $\{x(t)\}$. That is, we have to estimate a function on the basis of sampled observations with noise by using an appropriate smoothing method. Most methods for functional data are based on an approximation with truncated basis function expansions (Ramsay and Silverman, 1997). In other words, it is assumed that functional data unit can be expressed with sufficient accuracy by a linear combination of finite terms of basis functions. Suppose we use a sets of basis functions $\phi(s) = (\phi_1(s), \ldots, \phi_K(s))^\top$, where K is the

number of basis functions. Then observed functions $x_i(s)$ can be expanded as

$$x_i(s) = \sum_{k=1}^{K} \mathbf{C}_{ij}\phi_k(s) = \mathbf{C}_i^{\top}\phi(s), \quad i = 1, \cdots, N,$$

where N is the number of observations, K is the number of basis functions, and $\mathbf{C}$ is N by K coefficient matrix.

17.1.2 Basic Statistics in Functional Context

Suppose we have N data functions, which are denoted by $x_1(t), x_2(t), \cdots, x_N(t)$. Similarly as in ordinary statistical theory, the basic statistics in functional context are defined as follows:

Mean function:

$$\bar{x}(t) = N^{-1}\sum_{i=1}^{N} x_i(t)$$

Variance function:

$$\mathrm{Var}(x(t)) = N^{-1}\sum_{i=1}^{N}[x_i(t) - \bar{x}(t)]^2$$

Covariance function:

$$\mathrm{Cov}(s,t) = N^{-1}\sum_{i=1}^{N}\{x_i(s) - \bar{x}(s)\}\{x_i(t) - \bar{x}(t)\}$$

17.1.3 Representing the Functional Data

`coef = fouriertrans (tmat, nbasis)`
Calculates the coefficients in applying a basis expansion by using Fourier series

`phi = fouriereval (nbasis, nresol, period)`
Evaluates the basis functions of Fourier series

The quantlet `Fouriertrans` calculates the coefficient matrix of functional data in applying the basis expansion by using Fourier series. And the quantlet `Fouriereval` evaluates the basis functions of Fourier series based on the period and the number of the points where functions are evaluated.

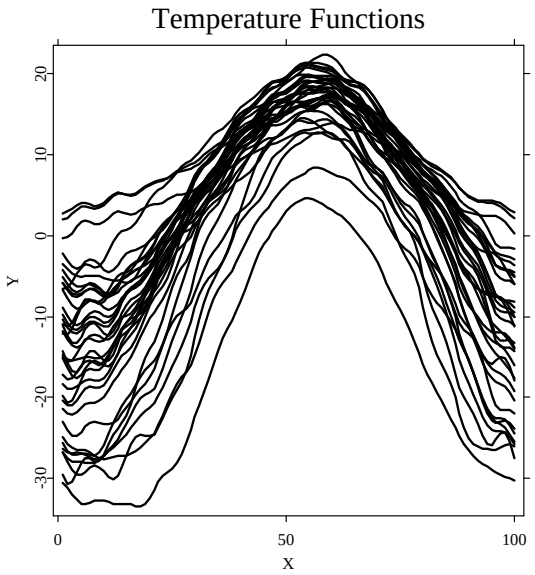

Figure 17.1: Example of Functional Data (Temperature).

XCSfda01.xpl

For example, the quantlets are applied to the daily temperature data of 35 weather stations in Canala (Ramsay and Silverman, 1997). We use Fourier series as basis functions. Figure 17.1 shows temperature functions. Evidently these data look like curves.

17.2 Functional Principal Component Analysis

Functional principal component analysis (PCA) is an important technique to extract a few major and typical features from complex data functions. In this section, we explain the outline of the functional PCA and how to use our quantlets.

17.2.1 Ordinary Functional Principal Component Analysis

Suppose we have a set of functional data $\{x_i(s)\}_{i=1}^{N}$. Weight function $\xi(s)$ is chosen in such a way that it maximizes the variance

$$PCASV = \int \int \xi(s)v(s,t)\xi(t)dsdt,$$

where $v(s,t)$ indicates the covariance function based on the functional data set. Note that the right hand side just corresponds to the quadratic form representing the variance of a linear combination of multivariate random vectors in classical multivariate analysis. The maximization of PCASV under the constraints

$$\int \xi_l(t)^2 dt = 1 \ , \ \int \xi_l(t)\xi_m(t)dt = 0 \ \ (l < m)$$

leads to an integral eigenequation as follows:

$$\int v(s,t)\xi(t)dt = \rho\xi(t).$$

17.2.2 Penalized Functional Principal Component Analysis

Here penalty function is introduced to incorporate smoothing into the principal components (PCs). Suppose ξ satisfies periodic boundary conditions, that is, the second and third derivatives of ξ satisfy periodic boundary conditions on $\mathcal{T}$. Then the most popular form of the penalty for ξ is given by

$$PEN_2(\xi) = \| \ D^2\xi \ \|^2 = \int \xi(t)D^4\xi(t)dt.$$

In this case the penalized variance can be expressed by

$$PCAPSV = \frac{PCASV}{\| \ \xi \ \|^2 + \lambda \times PEN_2(\xi)},$$

where λ is a smoothing parameter. This expression means that the trade-off between maximizing the sample variance and smoothing ξ is controlled by a smoothing parameter λ. The solution ξ is obtained as the eigenfunction associated with the largest eigenvalue of the following penalized eigenequation

$$\int v(s,t)\xi(t)dt = \rho(I + \lambda D^4)\xi(s).$$

17.2.3 Algorithm

Suppose we use a set of basis functions $\phi(s) = (\phi_1(s), \cdots, \phi_K(s))^\top$. Then a data function $x_i(s)$ and a weight function $\xi(s)$ can be expanded as

$$x_i(s) = \sum_{k=1}^{K} \mathbf{C}_{ik}\phi_k(s) = \mathbf{C}_i^\top \phi(s), \qquad \xi(s) = \sum_{k=1}^{K} y_k\phi_k(s) = y^\top \phi(s),$$

where K is the number of basis functions and $\mathbf{C}$ is N by K coefficient matrix. Define $\mathbf{V}$ as the covariance matrix of coefficient $\mathbf{C}_i$ and let $\mathbf{J}_\phi = \int \phi(s)\phi(s)^\top ds$, $\mathbf{K}_\phi = \int (D^2\phi(s))(D^2\phi(s))^\top ds$. Then the functional eigenequation is transformed to the following matrix eigenvalue problem

$$(\mathbf{J}_\phi \mathbf{V} \mathbf{J}_\phi)y = \rho(\mathbf{J}_\phi + \lambda \mathbf{K}_\phi)y.$$

By using Cholesky factorization $\mathbf{L}\mathbf{L}^\top = \mathbf{J}_\phi + \lambda\mathbf{K}_\phi$, the above generalized eigenvalue problem leads to an eigenvalue problem of a symmetrical matrix as

$$(\mathbf{S}\mathbf{J}_\phi \mathbf{V} \mathbf{J}_\phi \mathbf{S}^\top)(\mathbf{S}^{-T}y) = \rho(\mathbf{S}^{-T}y),$$

where $\mathbf{S} = \mathbf{L}^{-1}$ and $\mathbf{S}^{-T} = (\mathbf{S}^{-1})^T$. After solving y, we transform back to eigenfunction $\xi(s)$.

17.2.4 Applying Functional PCA

```
fpcaresult = FDApca (fdcoef, period, lambda{, npc})
```
carries out a penalized functional PCA

The quantlet `FDApca` enables us to apply penalized functional PCA. The input parameters of this quantlets are the coefficient matrix for functional data, the

period, the smoothing parameter, and the number of principal components to be kept. The result of the application is assigned to the variable `fpcaresult` which is a list containing the following output:

`fpcaresult.values` : the eigenvalues

`fpcaresult.varprop` : the proportion of variance explained by each eigenfunction

`fpcaresult.scores` : the PC scores

`fpcaresult.harmcoef` : the coefficient matrix of eigenfunctions

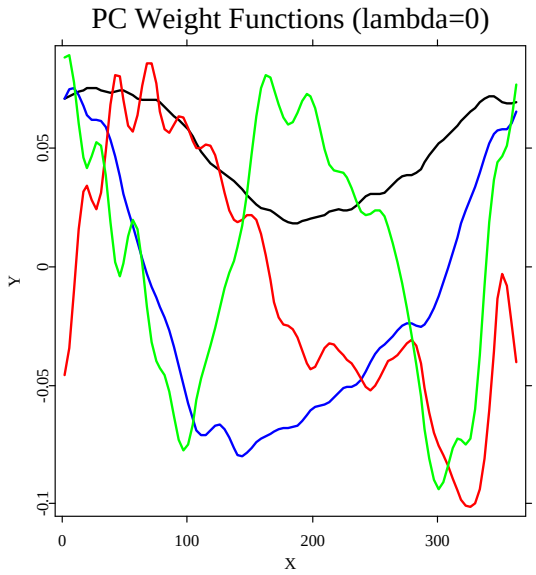

Figure 17.2: Weight Functions for PCs when $\lambda = 0$.

`XCSfda02.xpl`

For example, the functional PCA is applied to the daily temperature data. We use Fourier series as basis functions. Figure 17.2 shows the PC weight functions

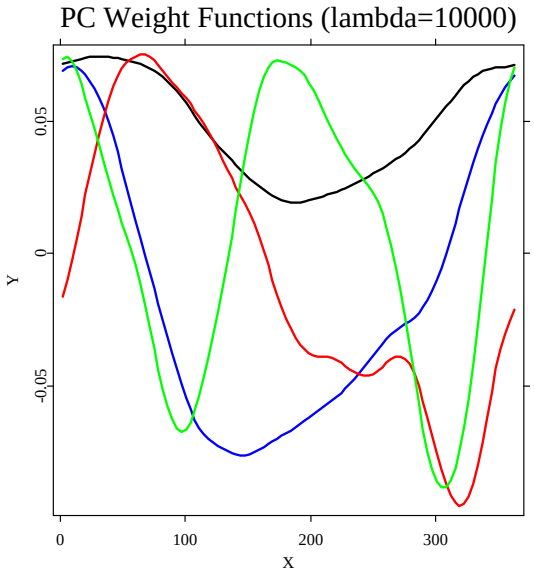

Figure 17.3: Weight Functions for PCs when $\lambda = 10000$.

XCSfda03.xpl

when $\lambda = 0$, while Figure 17.3 shows the PC weight functions when $\lambda = 10000$. The black curve indicates the weight function for the first PC, the blue curve indicates the weight function for the second PC, the red curve indicates the weight function for the third PC, and the green curve indicates the weight function for the fourth PC. You can see that the penalized method removes the roughness in the raw PC curves as λ increases. Its effect makes it easier for users to interpret the result at the expense of the decrease of PC variance. In the case of ordinary functional PCA, the eigenvalues are shown in Table 17.1.

On the other hand, in the case of penalized functional PCA, the eigenvalues are shown in Table 17.2.

Table 17.1: Eigenvalues of ordinary functional PCA

Eigenvalues
804.69
73.104
27.11
7.122

Table 17.2: Eigenvalues of penalized functional PCA

Eigenvalues
802.45
71.01
22.883
5.0654

17.2.5 Interpretation

Looking at these weight functions, we can interpret the PCs as follows: The first PC is a measure of overall temperatures throughout the year, because it is positive throughout the year. In particular, the temperature in winter has the greatest variability between the observations. The second PC represents the contrast between the temperatures in summer and in winter, so it is a measure of uniformity of temperature through the year. The third PC represents the contrast between the temperatures in the first part and in the last part of the year. So it is a time shift effect almost through the year. Finally, the fourth PC consists of a positive contribution for the winter and summer months and a negative contribution for the spring and autumn months, therefore it corresponds to an effect on the onset of spring and autumn.

Bibliography

Besse, P. and Ramsay, J. O. (1988). Principal components analysis of sampled functions, *Psychometrika* **51**: 285–311.

Härdle, W., Klinke, S., and Müller, M. (2000). *XploRe Learning Guide*, Springer.

Ramsay, J. O. and Dalzell, C. (1991). Some tools for functional data analysis (with discussion), *J. Royal Statist. Soc.* **B 53**: 539-572.

Ramsay, J. O. and Silverman, B. W. (1997). *Functional Data Analysis*, Springer.

18 Statistical Analysis of Failure Time with microearthquakes applications

Graciela Estévez-Pérez, Alejandro Quintela del Rio

Expected length of the paper: 15-20 pages

18.1 Abstract

This chapter is devoted to the Statistical Analysis of Failure Time by means of nonparametric estimation of hazard function, and more specifically to its application for analyzing temporal data on earthquake occurrences. We first present the method of estimation and the framework: *kernel estimation of hazard function under a general dependence assumption on the sample data*. In this situation, the asymptotic optimality properties (consistency and asymptotic normality) are established and the controversial problem of bandwidth selection is approached. In fact, we prove the asymptotic optimality of the cross-validation procedure (both global and local version) and we get their convergence rates. In addition, the global rate of convergence is used to motivate the introduction of a penalized version of the cross-validation procedure, which gives better estimations than the ordinary cross-validation bandwidth.

On the other hand, an important part of chapter is devoted to study the occurrence process of earthquakes in some geographic regions making use of the previous tools and showing the corresponding functions implemented in XploRe. Our analysis, based on the information provided by the data and on the universally accepted assumption of temporal grouping of earthquakes, confirm this grouping and characterize both, the occurrence process of main shocks and the aftershock sequences (clusters).

19 Fuzzy Clustering

Hizir Sofyan

Fuzzy clustering is one of the non-hierarchical clustering methods. The purpose of clustering is to construct groups in such a way that the profiles of objects in the same groups are relatively homogenous whereas the profiles of objects in different groups are relatively heterogeneous.

In conventional clustering, sample is either assigned to or not assigned to a group. Fuzzy clustering which apply the concept of fuzzy sets to cluster analysis give belongedness to groups at each point of data set by a membership function. Its advantage can adapt to noisy data and classes that are not well separated. In this paper, we handled with biomedical data.

19.1 Introduction

The basic problem of clustering is to begin with a sample of n p-dimensional points and then to classify the points into groups purely from their location in p-dimensional space. The aim of clustering is to form groups in which the observation characteristics in one group is relatively homogeneous whereas the observation characteristics among different groups are relatively heterogeneous. In general, clustering methods can be divided into two categories: hierarchical clustering and non-hierarchical clustering. One of non-hierarchical clustering methods is fuzzy clustering.

The use of fuzzy set theory is becoming popular because it produces not only crisp decision when necessary but also corresponding degree of membership. Usually, membership functions are defined based on a distance function, such that membership degrees express proximities of entities to cluster centers.

19.2 Basic Concepts

19.2.1 Probability and Fuzziness

An event E in τ-field in probability theory is defined as a subset of the sample space Ω. Space Ω is a collection of possibilities or sample points and the realization of these sample points indicates its occurrence. However, many real world events that encounter daily are perceived to be vague or ill-defined rather than being a probabilistic problem.

For example, if we are asked whether it will rain tomorrow or not then we will answer by expressing that the probability of rain tomorrow is 40% with the implication 60% not rain. While the fact is that we are not able to give an exact answer. We prefer to use the linguistically expression like *it is a very likely to rain* or *the chance of rain is about 40%*.

The concept of fuzzy sets was first introduced by Zadeh (1965) to represent vagueness. Fuzzy sets extend to clustering in that object of the data set may be fractionally assigned to multiple clusters. This allows for ambiguity in the data and yields detailed information about the structure of the data. One of the uses of fuzzy sets is fuzzy clustering, that will be discussed in the next section.

19.2.2 Distance Measures

The distances between points play an important role in clustering. A distance between two p-dimensional observations $x = (x_1, x_2, ..., x_p)^\top$ and $y = (y_1, y_2, ..., y_p)^\top$ is denoted in matrix notation as:

$$d(x, y) = \sqrt{(x - y)^\top (x - y)} \tag{19.1}$$

and it is called Euclidean distance. The statistical distance between these two observations is

$$d(x, y) = \sqrt{(x - y)^\top A (x - y)} \tag{19.2}$$

where $A = S^{-1}$ is the inverse of S, the matrix of sample variances and covariances. It is often called Mahalanobis distance.

The distance measure or metric should be chosen with care. The Euclidean metric should not be used where different attributes have widely varying aver-

age values and standard deviations, since large numbers in one attribute will prevail over smaller numbers in another. With the diagonal and Mahalanobis metrics, the input data are transformed before use. Choosing the Mahalanobis metric results in transformation of the data set to one in which all attributes have zero mean and unit variance. Correlations between variables are taken into account. Choosing the diagonal metric results in transformation of the data set to one in which all attributes have equal variance.

19.3 Fuzzy Clustering

19.3.1 Fuzzy C-means Method

```
v = xcfcme(x, c, m, e, alpha)
     Performs a fuzzy C-means cluster analysis
```

The idea of fuzzy clustering came from the Hard C-Means (HCM) founded by Ruspini (1969). He introduced a notion of fuzzy partition to describe the cluster structure of a data set and suggested an algorithm to compute the optimum fuzzy partition. Dunn (1973) generalized the minimum-variance clustering procedure to a Fuzzy ISODATA clustering technique. Bezdek (1981) generalized Dunn's approach to obtain an infinite family of algorithms which is called the Fuzzy C-Means (FCM) algorithm defined as follows:

$$J_m(U,V) = \sum_{k=1}^{n}\sum_{i=1}^{c} u_{ik} d^2(x_k, v_i), \tag{19.3}$$

where $X = (x_1, x_2, ..., x_n)$ is n data sample vectors, U is a partition of X in c part, $V = (v_1, v_2, ..., v_c)$ are cluster centers in R^p, $d^2(x_k, v_i)$ is an inner product induced norm on R^p, and u_{ik} is referred to as the grade of membership of x_k to the cluster i, in this case the member of u_{ik} is 0 or 1.

One approach to fuzzy clustering is the fuzzy C-Means (Bezdek, 1981). Before Bezdek, Dunn (1973) had developed the fuzzy C-Means Algorithm. The idea of Dunn's algorithm is to extend the classical within groups sum of squared error objective function to a fuzzy version by minimizing this objective function. Bezdek generalized this fuzzy objective function by introducing the weighting

exponent m, $1 \leq m < \infty$:

$$J_m(U,V) = \sum_{k=1}^{n}\sum_{i=1}^{c}(u_{ik})^m d^2(x_k, v_i), \tag{19.4}$$

where U is a partition of X in c part, $V = v = (v_1, v_2, ..., v_c)$ are the cluster centers in R^p, and A is any $(p \times p)$ symmetric positive definite matrix defined as the following:

$$d(x_k, v_i) = \sqrt{(x_k - v_i)^\top (x_k - v_i)} \tag{19.5}$$

where $d(x_k, v_i)$ is an inner product induced norm on R^p, u_{ik} is referred to as the grade of membership of x_k to the cluster i. This grade of membership satisfies the following constraints:

$$0 \leq u_{ik} \leq 1, \quad \text{for } 1 \leq i \leq c, 1 \leq k \leq n, \tag{19.6}$$

$$0 < \sum_{k=1}^{n} u_{ik} < n, \qquad \text{for } 1 \leq i \leq c, \tag{19.7}$$

$$\sum_{i=1}^{c} u_{ik} = 1, \qquad \text{for } 1 \leq k \leq n. \tag{19.8}$$

The fuzzy C-Means (FCM) uses an iterative optimization of the objective function, based on the weighted similarity measure between x_k and the cluster center v_i.

Steps of the fuzzy C-Means algorithm, according to **?** follow:

Algorithm

1. Given a data set $X = \{x_1, x_2, ..., x_n\}$, select the number of clusters $2 \leq c < N$, the maximum number of iterations T, the distance norm $d^2(x_k, v_i)$, the fuzziness parameter $m > 1$, and the termination condition $\varepsilon > 0$.
2. Give an initial value $U^{(0)}$.
3. For $t = 1, 2, ..., T$

a) Calculate the c cluster centers $\{v_{i,t}\}, i = 1, ..., c$

$$v_{i,t} = \frac{\sum_{k=1}^{n} u_{ik,t-1}^{m} x_k}{\sum_{k=1}^{n} u_{ik,t-1}^{m}} \tag{19.9}$$

b) Update the membership matrix. Check the occurrence of singularities. Let $I = \{1, ..., c\}$,

$$I_{k,t} = \{i | 1 \leq i \leq c, d_{ik,t} = \parallel x_k - v_{i,t} \parallel = 0\},$$

and $\bar{I}_{k,t} = \{1, 2, ..., c\}/I_{k,t}$
Then calculate the following

$$u_{ik,t} = \sum_{j=1}^{c} \left(\frac{d_{ik,t}}{d_{jk,t}} \right)^{\frac{2}{m-1}}, \text{ if } \Upsilon_{\mathrm{k,t}} = 0 \tag{19.10}$$

Choose $a_{ik,t} = 1/\#(\Upsilon_{k,t}), \forall i \in \Upsilon; \#(.)$ denotes the ordinal number.

4. If $E_t = \parallel U_{t-1} - U_t \parallel \leq \varepsilon$ then stop otherwise return to step 3.

This procedure converges to a local minimum or a saddle point of J_m. The FCM algorithm computes the partition matrix U and the clusters' prototypes in order to derive the fuzzy models from these matrices.

The syntax of this algorithm in XploRe is

```
fcm=xcfcme(x,c,m,e)
```

The inputs are the following; `x` is a $n \times p$ matrix of n row points to be clustered, `c` is the number of clusters, `m` is an exponent weight factor ($m > 1$), `e` is termination tolerance, and `u` is $n \times p$ matrix of initialized uniform distribution.

19.3.2 Fuzzy Gustafson Kessel

Gustafson and Kessel (GK) is an extension of FCM. Different distributions and size of clusters usually load to sub optimal results with FCM. In order to adopt to different structures in data, GK used the covariance matrix to capture ellipsoidal properties of clusters.

Gustafson and Kessel (1979) extended the fuzzy C-Means algorithm for an inner-product metric norm

$$d(x_k, v_i) = \sqrt{(x_k - v_i)^\top M_i (x_k - v_i)}, \tag{19.11}$$

where M_i is a positive definite matrix adapted according to the actual shapes of the individual clusters, described approximately by the cluster covariance matrices F_i.

$$F_i = \frac{\sum_{k=1}^{N} (\mu_{i,k})^m (x_k - v_i)(x_k - v_i)^\top}{\sum_{k=1}^{N} (\mu_{i,k})^m} \tag{19.12}$$

It can be shown that the distance inducing matrix M_i is calculated as the normalized inverse of the cluster covariance matrix

$$M_i = \det(F_i)^{\frac{1}{n}} F_i^{-1}. \tag{19.13}$$

The normalization by the determinant of F_i is involved in order to constraint M_i. Without this constraint, the objective function

$$J_m(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik})^m d^2(x_k, v_i), \tag{19.14}$$

which is linear with respect to M_i could be made as small as desired by making M_i less positive definite.

Algorithm

Given a data set X, we choose the number of clusters $1 < c < N$, the weighting exponent $m > 1$ and the termination tolerance $\epsilon > 0$. Initialize the fuzzy partition matrix $U^{(0)}$ randomly, such that it satisfies the conditions

$$\sum_{i=1}^{c} \mu_{i,k} = 1, k = 1, ..., N \tag{19.15}$$

and

$$0 < \sum_{k=1}^{N} \mu_{i,k} < N, i = 1, ..., c. \tag{19.16}$$

Say about iteration Repeat for $t = 1, 2, ...$

Step 1: Compute the cluster centers: V_i, t,

$$v_{i,t} = \frac{\sum_{k=1}^{n} u_{ik,t-1}^{m} x_k}{\sum_{k=1}^{n} u_{ik,t-1}^{m}} \tag{19.17}$$

Step 2: Compute the cluster covariance matrices:

$$F_i = \frac{\sum_{k=1}^{N} (u_{i,k})^m (x_k - v_i)(x_k - v_i)^\top}{\sum_{k=1}^{N} (u_{i,k})^m} \tag{19.18}$$

Step 3: Compute the distances

$$d^2(x_k, v_{i,t}) = (x_k - v_{i,t})^\top [\det(F_i)^{1/n} F_i^{-1}](x_k - v_{i,t}) \tag{19.19}$$

Step 4: Update the fuzzy partition matrix:

$$u_{ik,t} = \sum_{j=1}^{c} \left\{ \frac{d_{ik,t}}{d_{jk,t}} \right\}^{\frac{2}{m-1}}, \tag{19.20}$$

if $d_{ik} = 0$ for some $i = s$, set $u_{ks} = 1$ and $u_{ik} = 0$.

Until $E_t = \| U_{t-1} - U_t \| \leq \varepsilon$.

19.3.3 Fuzzy Gath-Geva

Gath Geva combines FC Means and FMLE. It ignores the objective function $J(X, V)$ and simply replaces u_{i_k} by posterior probability $\mathrm{P}(C_k/x_i)$ of class C_k given the observation x_i. Although it does not give optimal partition in cases of variable cluster shapes and densities. Using an "exponential distance" including the fuzzy covariance matrix (FMLE) results in optimal partition even

when a great variability of cluster shapes and densities is present. Given a data set X, we choose the number of clusters $1 < c < N$, the weighting exponent $m > 1$ and the termination tolerance $\epsilon > 0$. Initialize the fuzzy partition matrix $U^{(0)}$ randomly, such that it satisfies the conditions

$$\sum_{i=1}^{c} \mu_{i,k} = 1, k = 1, ..., N \tag{19.21}$$

and

$$0 < \sum_{k=1}^{N} \mu_{i,k} < N, i = 1, ..., c. \tag{19.22}$$

Then repeat for $t = 1, 2, ...$

Step 1: Compute the cluster centers $v_{i,t}$:

$$v_{i,t} = \frac{\sum_{k=1}^{n} u_{ik,t-1}^{m} x_k}{\sum_{k=1}^{n} u_{ik,t-1}^{m}} \tag{19.23}$$

Step 2: Compute the cluster covariance matrices:

$$F_i = \frac{\sum_{k=1}^{N} (\mu_{i,k})^m (z_k - v_i)(z_k - v_i)^\top}{\sum_{k=1}^{N} (\mu_{i,k})^m} \tag{19.24}$$

Step 3: Compute the prior probability

$$P_i = \frac{\sum_{k=1}^{n} u_{ik,t-1}^{m}}{\sum_{k=1}^{n} u_{ik,t-1}^{m}} \tag{19.25}$$

Step 4: Compute the distances:

$$(d^2(x_k, v_{i,t}) = \frac{1}{P_i} \sqrt{F_i} \exp 1/2(x_k - v_{i,t} F_i^{-1}(x_k - v_{i,t})) \tag{19.26}$$

Step 5: Update the fuzzy partition matrix:

$$u_{ik,t} = \sum_{j=1}^{c} (\frac{d_{ik,t}}{d_{jk,t}})^{\frac{2}{m-1}}, \tag{19.27}$$

if $d_{ik} = 0$ for some $i = s$, set $u_{ks} = 1$ and $u_{ik} = 0$.

Until $E_t = \| U_{t-1} - U_t \| \leq \varepsilon$.

19.4 Cluster Validity

In practical applications, we need a cluster validity method to measure the quality of clustering result. The quality of a clustering process depends on many factors, such as the method of initialization, the choice of the number of classes c, and the clustering method. The method of initialization requires a good estimate of the clusters and its application dependent, so the cluster validity problem is reduced to the choice of an optimal number of classes c. Several cluster validity measures have been developed in the past. In this section, we describe only four of these measures.

19.4.1 The Partition Coefficient

The partition coefficient is defined as

$$F(U, c) = \frac{1}{n} \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^2 \tag{19.28}$$

Suppose that ω_c represents the clustering result, then the optimal choice of c is given by

$$\max_{c} \left\{ \max_{\Omega_c} F(U, c) \right\}, c = 2, ..., n-1. \tag{19.29}$$

The partition coefficient measures the closeness of all input samples to their corresponding cluster centers. If each sample is closely associated with only one cluster, that is, if for each k, u_ik is large for only one i value, then the uncertainty of the data is small, which corresponds to a large $F(U, c)$ value.

19.4.2 The Partition Entropy

The partition entropy is defined as

$$H(U,c) = -\frac{1}{n}\sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik} log(u_{ik}). \tag{19.30}$$

The optimal choice of c is given by

$$\min_{c}\left\{\min_{\Omega_c} H(U,c)\right\}, c = 2, ..., n-1. \tag{19.31}$$

When all u_{ik}'s have values close to 0.5, which represents a high degree of fuzziness of the clusters, $H(U,c)$ is large and thus indicates a poor clustering result. On the other hand, if all u_{ik}'s have values close to 0 or 1, $H(U,c)$ is small and indicates a good clustering result.

19.4.3 The Compactness and Separation Validity

The compactness and separation validity function is defined as:

$$S(U,c) = \frac{\frac{1}{n}\sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^2 |x_k - v_i|^2}{\min |v_i - v_j|^2} \tag{19.32}$$

The optimal choice of c is given by

$$\min_{c}\left\{\min_{\Omega_c} S(U,c)\right\}, c = 2, ..., n-1, where \tag{19.33}$$

$S(U,c)$ is the ratio between the average distance of input samples to their corresponding cluster centers and the minimum distance between cluster centers. A good cluster procedure should make all input samples as close to their cluster centers as possible and all cluster centers separated as far as possible.

19.5 Illustrative Example

remotely sensing data will be used as an application.

Bibliography

Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algoritmss*, Plenum Press, New York.

Bezdek, J. C. and Pal, S. K. (1992). *Fuzzy Models for Pattern Recognition*, IEEE Press, New York.

Dunn, J. C. (1973). A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, *Journal of Cybernetics* **3**: 32–57.

Flury, B. and Riedwyl, H. (1988). *Multivariate Statistics, A Practical Approach*, Cambridge University Press.

Everitt, B. S. (1993). *Cluster Analysis*, Edward Arnold, London.

Gower, J. C. (1967). A comparison of some methods of cluster analysis, *Biometrics* **23**: 623–628.

Härdle, W., Klinke, S., and Turlach, B.A.(1995). *XploRe: An Interactive Statistical Computing Environment*, Springer Verlag, New York.

Härdle, W. and Simar, L. (2000). Applied Multivariate Statistical Analysis, *http://www.md-stat.com*, Humboldt Universität zu Berlin.

Hellendorn, H. and Driankov, D. (1998). Fuzzy: Model Identification, *Springer Verlag*, Heidelberg.

Johnson, R. A., and Wichern D. W. (1992). *Applied Multivariate Statistical Analysis*, Prentice-Hall.

MacQueen, J. B. (1967). Some Methods for classification and Analysis of Multivariate Observations, *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, University of California Press, **1**: 281–297.

Mucha, H. J. (1992). Clusteranalyse mit Microcomputern, *Akademie Verlag*, Berlin.

Mucha, H. J. (1995). Clustering in an Interactive Way, *Discussion Paper 9513*, Institut für Statistik und Ökonometrie, Humboldt-Universität zu Berlin.

Mucha, H. J. (1996). CLUSCORR: Cluster Analysis and Multivariate Graphics under MS-EXCEL, *Report No. 10*, WIAS Institut, Berlin.

Ruspini, E. H. (1969). A New Approach to Clustering, *Information Control* **15**: 22–32.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function, *Journal of Amer. Statist. Assoc.* **58**: 236–244.

Zadeh, L. A. (1965). Fuzzy Sets, *Information Control* **8**: 338–353.