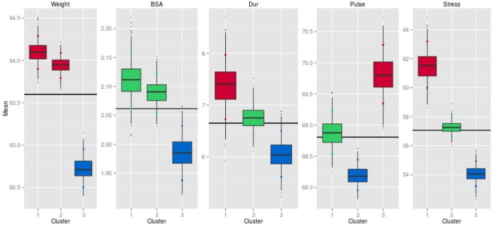
Using Profile Regression Mixture Models and Dirichlet Processes to explore the combined effect of risk factors; the R package PReMiuM

Silvia Liverani and Michail Papathomas



British-German meeting on classification - UCL
November 2013

Outline

- Motivation
- Method
- The R package PReMiuM
- Examples

People

- David Hastie
- John Molitor
- Sylvia Richardson

Multicollinearity

- Goal of epidemiological studies is to investigate the joint effect of different covariates / risk factors on a phenotype...
- ... but highly correlated risk factors create collinearity problems!

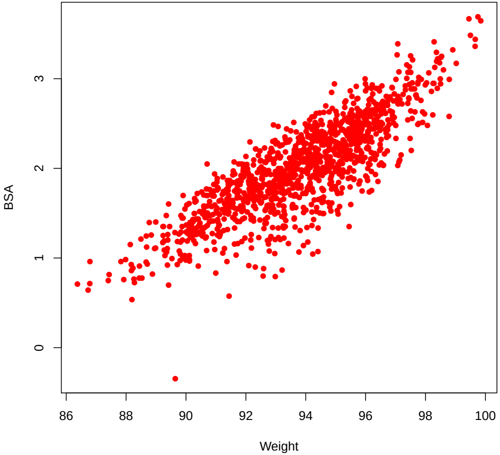
Multicollinearity

- Goal of epidemiological studies is to investigate the joint effect of different covariates / risk factors on a phenotype...
- ... but highly correlated risk factors create collinearity problems!

Example

Researchers are interested in determining if a relationship exists between **blood pressure** (y = BP, in mm Hg) and

- **weight** (x_1 = Weight, in kg)
- **body surface area** (x_2 = BSA, in sq m)
- duration of hypertension (x_3 = Dur, in years)
- basal pulse (x_4 = Pulse, in beats per minute)
- stress index (x_5 = Stress)



Weight = x_1, BSA = x_2

- Highly correlated risk factors create collinearity problems, causing instability in model estimation

Model	$\hat{\beta}_1$	SE $\hat{\beta}_1$	$\hat{\beta}_2$	SE $\hat{\beta}_2$
$y \sim x_1$	2.64	0.30	–	–
$y \sim x_2$	–	–	3.34	1.33
$y \sim x_1 + x_2$	6.58	0.53	-20.44	2.28

- Highly correlated risk factors create collinearity problems, causing instability in model estimation

Model	$\hat{\beta}_1$	SE $\hat{\beta}_1$	$\hat{\beta}_2$	SE $\hat{\beta}_2$
$y \sim x_1$	2.64	0.30	–	–
$y \sim x_2$	–	–	3.34	1.33
$y \sim x_1 + x_2$	6.58	0.53	-20.44	2.28

- Effect 1: the estimated regression coefficient of any one variable depends on which other predictor variables are included in the model.
- Effect 2: the precision of the estimated regression coefficients decreases as more predictor variables are added to the model.

When it is of interest to detect interactions between covariates, standard regression modelling may become problematic.

- In a classical setting, fitting a linear model with many parameters sometimes **requires an impractically large vector of observations** to produce valid inferences (Burton et al., IJE, 2009). Also, **identifiability and collinearity** problems are often present.
- In Bayesian model comparison, the space of models becomes vast, and model search algorithms like the Reversible Jump approach (Green, Bka 1995) require **an impractically large number of iterations** before they converge (Dobra and Massam, St Meth 2010).

Issues caused by

- correlated risk factors
- interacting risk factors

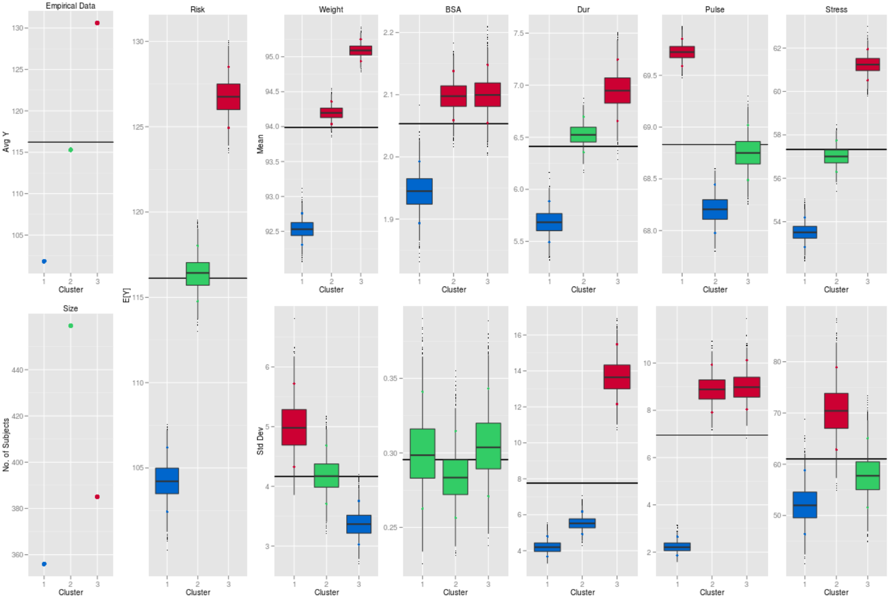
Issues caused by

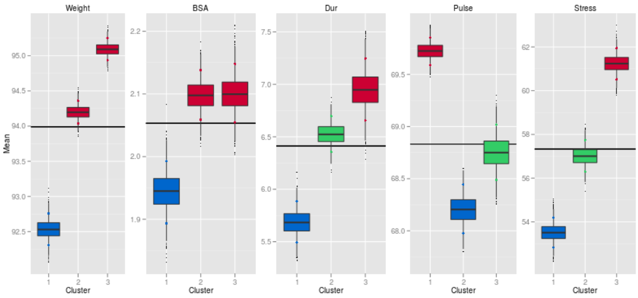
- correlated risk factors
- interacting risk factors

$\downarrow$

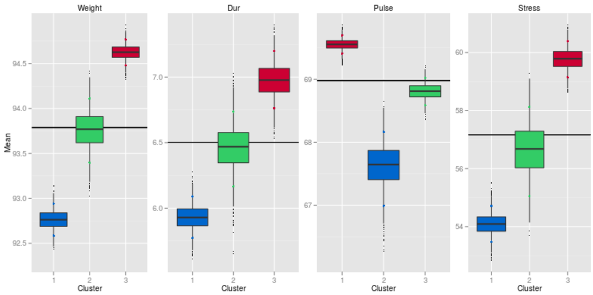
Profile regression

- **partitions the subjects into groups according to covariate profile**
- investigation of the joint effects of multiple risk factors
- jointly models the covariate patterns and health outcomes
- flexible but tractable Bayesian model


Empirical Data
Avg Y
130
125
120
115
110
105
Size
No. of Subjects
440
420
400
380
360
Risk
E[Y]
130
125
120
115
110
105
Weight
Mean
95.0
94.5
94.0
93.5
93.0
92.5
Std Dev
6
5
4
3
BSA
2.2
2.1
2.0
1.9
0.35
0.30
0.25
Dur
7.5
7.0
6.5
6.0
5.5
16
14
12
10
8
6
4
Pulse
69.5
69.0
68.5
68.0
10
8
6
4
2
Stress
62
60
58
56
54
80
70
60
50
1
2
3
Cluster



x_1 and x_2



only x_1

Notation

For individual *i*

y_i	outcome of interest
$\mathbf{x}_i = (x_{i1}, \ldots, x_{iP})$	covariate profile
$\mathbf{w}_i$	fixed effects
$z_i = c$	the allocation variable indicates the cluster to which individual *i* belongs

Statistical Framework

- Mixture model for the covariates

$$f(\mathbf{x}_i|\phi,\psi) = \sum_c \psi_c f(\mathbf{x}_i|z_i = c, \phi_c)$$

Statistical Framework

- Mixture model for the covariates

$$f(\mathbf{x}_i|\phi,\psi)=\sum_c \psi_c f(\mathbf{x}_i|z_i=c,\phi_c)$$

- For example for discrete covariates

$$f(\mathbf{x}_i|z_i=c,\phi_c)=\prod_{j=1}^{J}\phi_{z_i,j,x_{i,j}}$$

Statistical Framework

- Mixture model for the covariates

$$f(\mathbf{x}_i|\phi,\psi) = \sum_c \psi_c f(\mathbf{x}_i|z_i = c, \phi_c)$$

- For example for discrete covariates

$$f(\mathbf{x}_i|z_i = c, \phi_c) = \prod_{j=1}^{J} \phi_{z_i,j,x_{i,j}}$$

Statistical Framework

- Mixture model for the covariates

$$f(\mathbf{x}_i|\phi,\psi) = \sum_c \psi_c f(\mathbf{x}_i|z_i = c, \phi_c)$$

Statistical Framework

- Mixture model for the covariates

$$f(\mathbf{x}_i|\phi,\psi) = \sum_c \psi_c f(\mathbf{x}_i|z_i = c, \phi_c)$$

- Prior model for the mixture weights ψ_c
  - stick-breaking priors (constructive definition of the Dirichlet Process)

$$\mathbb{P}(Z_i = c|\boldsymbol{\psi}) = \psi_c$$

Statistical Framework

- Mixture model for the covariates

$$f(\mathbf{x}_i|\phi,\psi) = \sum_c \psi_c f(\mathbf{x}_i|z_i = c, \phi_c)$$

- Prior model for the mixture weights ψ_c
  - stick-breaking priors (constructive definition of the Dirichlet Process)

$$\mathbb{P}(Z_i = c|\boldsymbol{\psi}) = \psi_c \qquad \psi_1 = V_1$$

$$\psi_c = V_c \prod_{l<c}(1 - V_l) \qquad V_c \sim \mathrm{Beta}(1,\alpha)$$

Statistical Framework

- Mixture model for the covariates

$$f(\mathbf{x}_i|\phi,\psi) = \sum_c \psi_c f(\mathbf{x}_i|z_i = c, \phi_c)$$

- Prior model for the mixture weights ψ_c
  - stick-breaking priors (constructive definition of the Dirichlet Process)

$$\mathbb{P}(Z_i = c|\boldsymbol{\psi}) = \psi_c \qquad \psi_1 = V_1$$

$$\psi_c = V_c \prod_{l<c}(1 - V_l) \qquad V_c \sim \mathrm{Beta}(1, \alpha)$$

  - larger concentration parameter α the more evenly distributed is the resulting distribution.
  - smaller concentration parameter α the more sparsely distributed is the resulting distribution, with all but a few parameters having a probability near zero

Statistical Framework

- Joint covariate and response model

$$f(\mathbf{x}_i, y_i|\phi, \theta, \psi, \beta) = \sum_c \psi_c f(\mathbf{x}_i|z_i = c, \phi_c) f(y_i|z_i = c, \theta_c, \beta, \mathbf{w}_i)$$

Statistical Framework

- Joint covariate and response model

$$f(\mathbf{x}_i, y_i|\phi, \theta, \psi, \beta) = \sum_c \psi_c f(\mathbf{x}_i|z_i = c, \phi_c) f(y_i|z_i = c, \theta_c, \beta, \mathbf{w}_i)$$

- Mixture model jointly for covariate and response
- For example, for Bernoulli outcome

$$\text{logit}\{p(y_i = 1|\theta_c, \beta, \mathbf{w}_i)\} = \theta_c + \beta^T \mathbf{w}_i$$

Statistical Framework

- Joint covariate and response model

$$f(\mathbf{x}_i, y_i|\phi, \theta, \psi, \beta) = \sum_c \psi_c f(\mathbf{x}_i|z_i = c, \phi_c) f(y_i|z_i = c, \theta_c, \beta, \mathbf{w}_i)$$

- Mixture model jointly for covariate and response
- For example, for Bernoulli outcome

$$\text{logit}\{p(y_i = 1|\theta_c, \beta, \mathbf{w}_i)\} = \theta_c + \beta^T \mathbf{w}_i$$

- The association of the profiles with the response are characterised by the risk effect parameters θ_c

Statistical Framework

- Joint covariate and response model

$$f(\mathbf{x}_i, y_i|\phi, \theta, \psi, \beta) = \sum_c \psi_c f(\mathbf{x}_i|z_i = c, \phi_c) f(y_i|z_i = c, \theta_c, \beta, \mathbf{w}_i)$$

- Mixture model jointly for covariate and response
- For example, for Bernoulli outcome

$$\text{logit}\{p(y_i = 1|\theta_c, \beta, \mathbf{w}_i)\} = \theta_c + \beta^T \mathbf{w}_i$$

- The association of the profiles with the response are characterised by the risk effect parameters θ_c
- Note, the above framework, adopted in Molitor et al. (Biostatistics, 2010) is similar to the one in Bigelow and Dunson (JASA, 2009).

Implementation

We have implemented profile regression in C++ (now wrapped in an R package) for

- binary, binomial, categorical, Normal and Poisson outcome
- Normal and discrete covariates

It can do

- dependent or independent slice sampling (Kalli et al., 2011)
- or truncated Dirichlet process model (Ishwaran and James, 2001)

as well as

- handles missing data
  - we check which cluster the subject is allocated to and then sample
- includes some label switching moves

Characterising the "best partition"

- Construct a score matrix $\mathbf{S} = S_{ij}$ that records the probability that individuals i and j are assigned to the same cluster.
- Task is to find a partition $\mathbf{z}^{best}$ that best represents $\mathbf{S}$
  - susceptible to Monte Carlo error
  - Dahl (2006) suggests using a least square distance criterion
  - Alternatively, we process the similarity matrix through the Partitioning Around Medoids algorithm (Kaufman & Rousseu, 1990) to derive $\mathbf{z}^{best}$.

Post-processing the output

- Once $\mathbf{z}^{best}$ is defined, it is important to have a measure of uncertainty/ stability
  - This can be done by model averaging quantities related to $\mathbf{z}^{best}$ (effects and cluster related parameters) through post-processing
  - If $\mathbf{z}^{best}$ is a consistent clustering, then subgroup parameters will vary little from iteration to iteration, leading to narrow posterior credible intervals.
- Predicting
  - At each iteration of the sampler, calculate the probability that i' belongs to each of the clusters

$$p(z_{i'} = c|\mathbf{x}_{i'}, \psi_c, \phi) \propto p(z_{i'}|c, \psi_c)p(\mathbf{x}_{i'}|z_{i'} = c, \phi)$$

and sample θ_c according to these probabilities

R package PReMiuM

Program to implement Dirichlet Process Bayesian Clustering

- Allows user to exclude the response from the model
- Allows user to run with a fixed alpha or update alpha (default)
- Allows users to run predictive scenarios (basic or Rao-Blackwellised predictions)
- Handling of missing data
- Adaptive MCMC where appropriate
- Can generate simulation data
- Performs post processing
- Functions for flexible plotting

R package PReMiuM

- Can be downloaded from CRAN

  `http://cran.r-project.org/web/packages/PReMiuM/`
- or

```r
install.packages("PReMiuM")
```

Example: Simulated data

The profiles are given by

y : outcome, Bernoulli
x : 5 covariates, all discrete with 3 levels
w : 2 fixed effects, continuous or discrete

Example: Simulated data

The profiles are given by

y : outcome, Bernoulli
x : 5 covariates, all discrete with 3 levels
w : 2 fixed effects, continuous or discrete

```
> profRegr(yModel="Bernoulli",
    xModel="Discrete",
    nSweeps=1000,
    nBurn=1000,
    data="dataMatrix.txt",
    output="output/output",
    nCovariates=5,
    covNames=c("Var1","Var2","Var3","Var4","Var5"),
    nFixedEffects=2,
    fixedEffectsNames=c("FE1","FE2")
    xLevels=c(3,3,3,3,3))
```

R package DiPBaC: Output

The programme creates a number of output files:

- output_z.txt → allocations z

R package DiPBaC: Output

The programme creates a number of output files:

- output_z.txt → allocations z
- output_nClusters.txt → cluster sizes

R package DiPBaC: Output

The programme creates a number of output files:

- output_z.txt → allocations z
- output_nClusters.txt → cluster sizes
- output_theta.txt → cluster-specific parameter for outcome: θ

R package DiPBaC: Output

The programme creates a number of output files:

- output_z.txt → allocations z
- output_nClusters.txt → cluster sizes
- output_theta.txt → cluster-specific parameter for outcome: θ
- output_phi.txt → cluster-specific parameter for covariates: ϕ

R package DiPBaC: Output

The programme creates a number of output files:

- output_z.txt $\rightarrow$ allocations *z*
- output_nClusters.txt $\rightarrow$ cluster sizes
- output_theta.txt $\rightarrow$ cluster-specific parameter for outcome: θ
- output_phi.txt $\rightarrow$ cluster-specific parameter for covariates: ϕ
- output_alpha.txt $\rightarrow$ parameter of the Dirichlet process α

R package DiPBaC: Output

The programme creates a number of output files:

- output_z.txt → allocations *z*
- output_nClusters.txt → cluster sizes
- output_theta.txt → cluster-specific parameter for outcome: θ
- output_phi.txt → cluster-specific parameter for covariates: ϕ
- output_alpha.txt → parameter of the Dirichlet process α
- output_beta.txt → fixed effects parameter β

R package DiPBaC: Output

The programme creates a number of output files:

- output_z.txt → allocations *z*
- output_nClusters.txt → cluster sizes
- output_theta.txt → cluster-specific parameter for outcome: θ
- output_phi.txt → cluster-specific parameter for covariates: ϕ
- output_alpha.txt → parameter of the Dirichlet process α
- output_beta.txt → fixed effects parameter β
- output_entropy.txt
- output_nMembers.txt
- output_logPost.txt
- output_psi.txt
- output_betaProp.txt
- output_log.txt
- output_thetaProp.txt
- output_alphaProp.txt
- output_input.txt

R package PReMiuM: Postprocessing

- Retuns file with output info

```
> runInfoObj<-readRunInfo(fileStem="output",
    directoryPath="output")
```

- Computes the score matrix that records the probability that each pair of individuals are in the same clusters

```
> dissimObj<-calcDissimilarityMatrix(runInfoObj)
```

- Computes the optimal partition

```
> clusObj<-calcOptimalClustering(dissimObj)
```

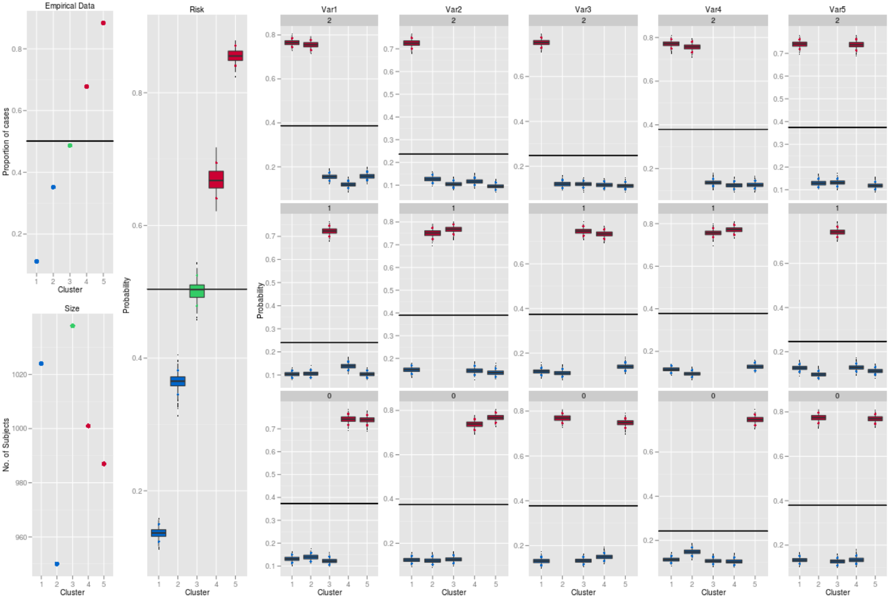
R package PReMiuM: Postprocessing

- Computes the average risk and profile for each cluster

```
> riskProfileObj<-calcAvgRiskAndProfile(clusObj)
```

- Plots

```
> clusterOrderObj<-plotRiskProfile(riskProfileObj,
   "output/summary.png")
```


Empirical Data
Proportion of cases
0.2
0.4
0.6
0.8
Cluster
1
2
3
4
5
Size
No. of Subjects
960
980
1000
1020
Cluster
Risk
Probability
0.2
0.4
0.6
0.8
Cluster
Var1
Var2
Var3
Var4
Var5
2
1
0
Probability
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
Cluster

Predictions

```
> profRegr(yModel="Bernoulli",
    xModel="Discrete",
    nSweeps=1000,
    nBurn=1000,
    data="dataMatrix.txt",
    output="output/output",
    nCovariates=5,
    covNames=c("Var1","Var2","Var3","Var4","Var5"),
    nFixedEffects=2,
    fixedEffectsNames=c("FE1","FE2")
    xLevels=c(3,3,3,3,3),
    predict="predict.scenarios.txt")
```

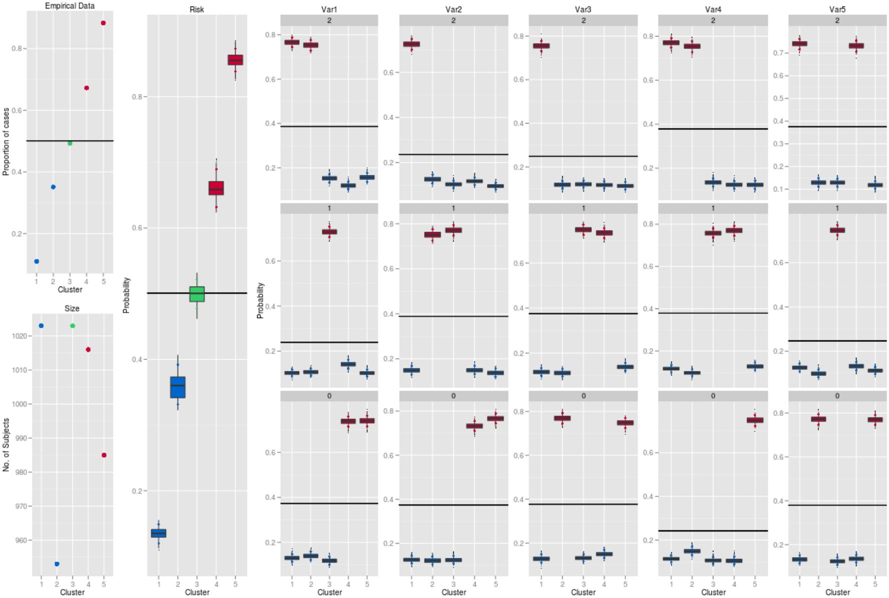
Predictions

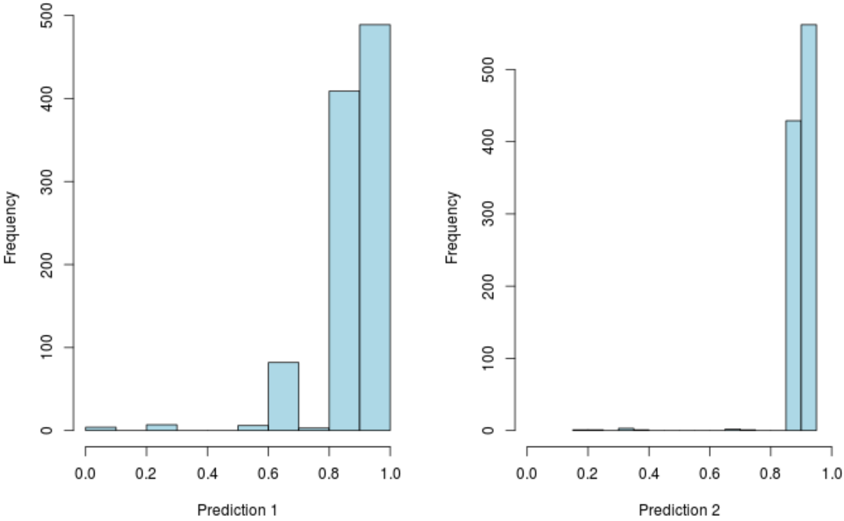
- The file `predict.scenarios.txt` contains the predictive scenarios for the 5 covariates and the 2 fixed effects.

```
0 1 0 0 0 -0.348608422340999  0.786982503814148
0 0 1 0 0  3.832601433577385 -0.67410633721989
```

- Postprocessing follows as before
- Predictions are postprocessed separately as follows

```
> predictions<-calcPredictions(riskProfileObj,
   doRaoBlackwell=F,
   fullSweepPredictions=T,
   fullSweepLogOR=T)
```


Empirical Data
Proportion of cases
0.2
0.4
0.6
0.8
1
2
3
4
5
Cluster
Size
No. of Subjects
960
970
980
990
1000
1010
1020
Cluster
Risk
Probability
0.2
0.4
0.6
0.8
Cluster
Var1
Var2
Var3
Var4
Var5
2
1
0
Probability
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
Cluster

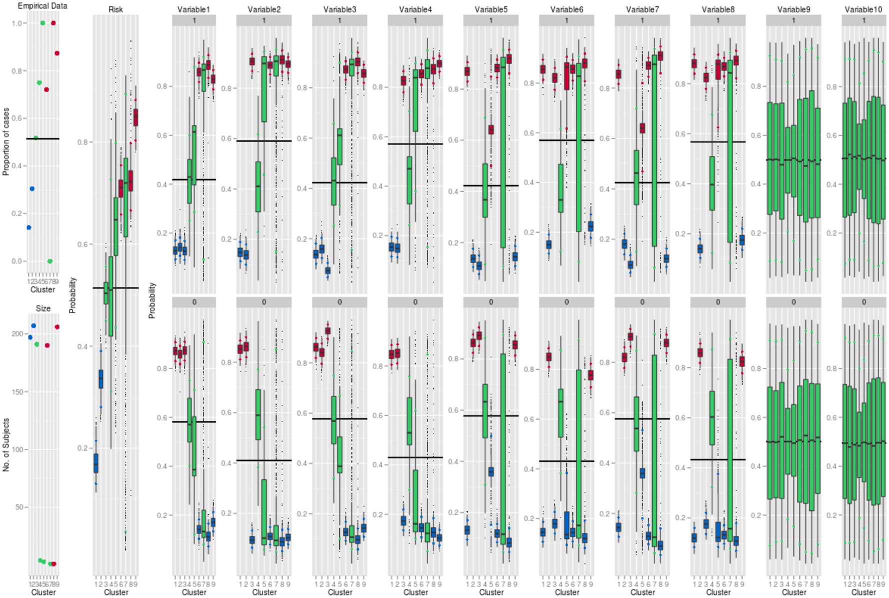

Frequency
0
100
200
300
400
500
0.0
0.2
0.4
0.6
0.8
1.0
Prediction 1
Frequency
0
100
200
300
400
500
0.0
0.2
0.4
0.6
0.8
1.0
Prediction 2

Variable selection

- In many applications, the number of predictors is large and the full profiles become difficult to interpret
  - ⇒ Aim is to identify the predictors that contribute more than others to the formation of sub-populations
- Use statistical procedures embedding variable selection within profile regression (Papathomas et al., 2012), inspired from Tadesse et al. (2005) and Chung and Dunson (2009)

Variable selection

- Context of application: genetic epidemiology where there is interest in investigating “gene-gene interactions”
  - As the number of markers genotyped is huge, blind search for interactions followed by adjustment for multiple testing has very low power → more focussed strategies are recommended
- In MP et al. (2012) we investigate how profile regression combined with variable selection can highlight combinations of SNPs associated with higher disease risk in a genetic association study of lung cancer (Hung et al 2008).


Empirical Data
Proportion of cases
1.0
0.8
0.6
0.4
0.2
0.0
123456789
Cluster
Size
No. of Subjects
200
150
100
50
123456789
Cluster
Risk
Probability
0.8
0.6
0.4
0.2
1 2 3 4 5 6 7 8 9
Cluster
Probability
Variable1
Variable2
Variable3
Variable4
Variable5
Variable6
Variable7
Variable8
Variable9
Variable10
1
0
0.8
0.6
0.4
0.2
1 2 3 4 5 6 7 8 9
Cluster

Using 0-1 variable selection switches

- Consider cluster specific binary switches, $\gamma_{c,\,j}$, so that
  - $\gamma_{c,\,j} = 1$ when predictor x_j is **important** for allocating subjects to cluster c, and in this case the associated probabilities are $\phi_{c,\,j}(k), 1 \leq k \leq m_j$
  - When $\gamma_{c,\,j} = 0$ the associated probability is $\phi_{0,\,j}(k)$, a common probability for all individuals for whom $x_{ij} = k$.

  So the discrete covariate model is modified accordingly to

$$f(\mathbf{x}_i \mid \phi_c, \gamma_c) = \prod_{j:\ \gamma_{c,\,j}=1} \phi_{z_i,\,j}(x_{ij}) \prod_{j:\ \gamma_{c,\,j}=0} \phi_{0,\,j}(x_{ij})$$

- Prior for switches: given ρ_j, $\gamma_{c,\,j} \sim \text{Bernoulli}(1, \rho_j)$.
- We consider a sparsity inducing prior for ρ with an atom at zero:

$$\rho_j \sim 1_{\{w_j=0\}}\delta_0(\rho_j) + 1_{\{w_j=1\}}\text{Beta}(\alpha_\rho, \beta_\rho)$$

  where $w_j \sim \text{Bernoulli}(0.5)$.

Similar to Chung and Dunson (JASA, 2009), but in their set up, covariate observations contribute to the likelihood through a regression model. In our case, covariate observations contribute directly to the likelihood, and we introduce $\phi_{0,j}(x_{ij})$.

Using latent selection probabilities

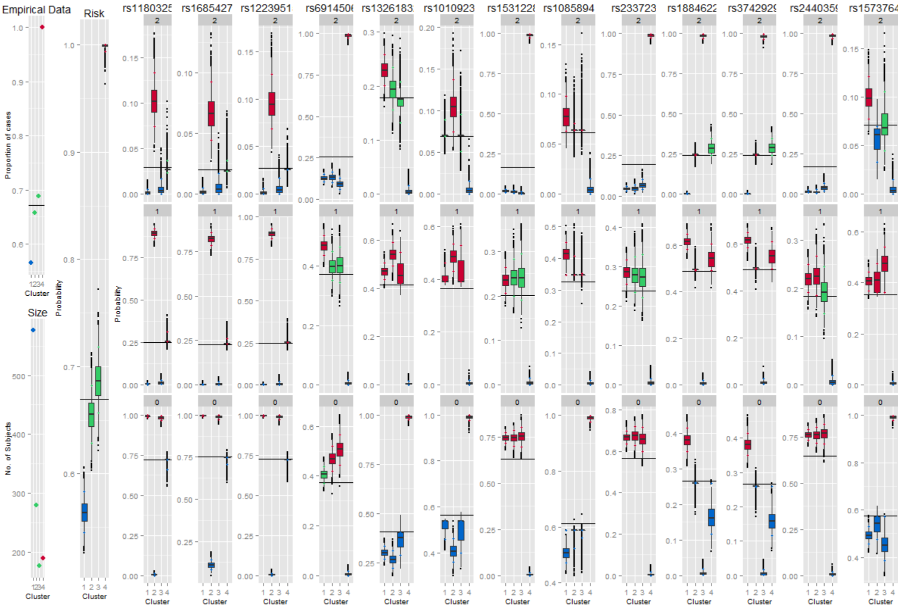
- The 0-1 variable selection switches model exhibits sometimes poor mixing, requiring long runs
- We introduce an new parametrisation involving continuous selection probabilities ζ_j, taking values in $[0, 1]$
- Each $\phi_{c,\,j}$ is replaced by a hybrid parameter,

$$\phi^*_{c,\,j}(x) = \zeta_j \times \phi_{c,\,j}(x) + (1 - \zeta_j) \times p_j(x) \tag{1}$$

  where $p_j(k)$ is the observed proportion of individuals for whom $x_{ij} = k$.
- Parameters ζ_j can be viewed as proxies for the probability that predictor j supports the cluster structure:
  $\rightarrow$ important predictors for the clustering will have values of ζ_j close to 1.
- Similarly, use a sparsity inducing prior for ζ_j with an atom at zero.

Variable Selection: R commands

```
> profRegr(yModel="Bernoulli",
    xModel="Discrete",
    nSweeps=1000,
    nBurn=1000,
    data="dataMatrix.txt",
    output="output/output",
    nCovariates=5,
    covNames=paste("Var",seq_along(1:10),sep=""),
    nFixedEffects=0,
    xLevels=rep(2,10),
    varSelect="BinaryCluster")
```


Empirical Data
Proportion of cases
1.0
0.9
0.8
0.7
0.6
1234
Cluster
Size
No. of Subjects
500
400
300
200
1234
Cluster
Risk
Probability
1.0
0.9
0.8
0.7
0.6
1 2 3 4
Cluster
rs1180325
rs1685427
rs1223951
rs691450
rs1326183
rs1010923
rs153122
rs1085894
rs233723
rs1884622
rs374292
rs244035
rs1573764
2
1
0
Probability
1 2 3 4
Cluster

Other features

Other inputs for `profRegr`:

hyper Set hyperparameters

nClusInit Set initial number of clusters, equal to integer

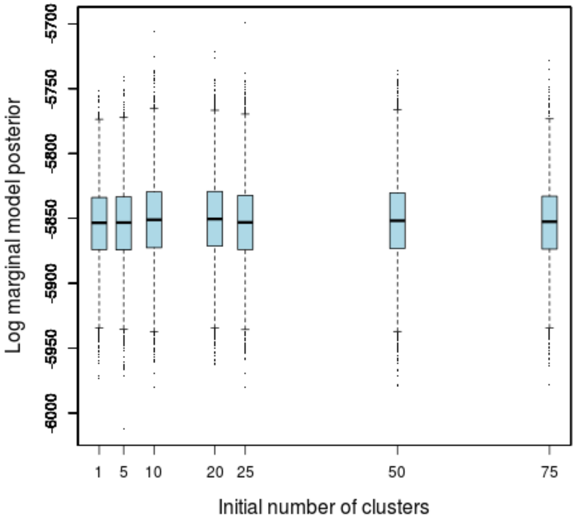
sampler Set sampler: Truncated, SliceDependent or SliceIndependent

alpha Can fix the value of α

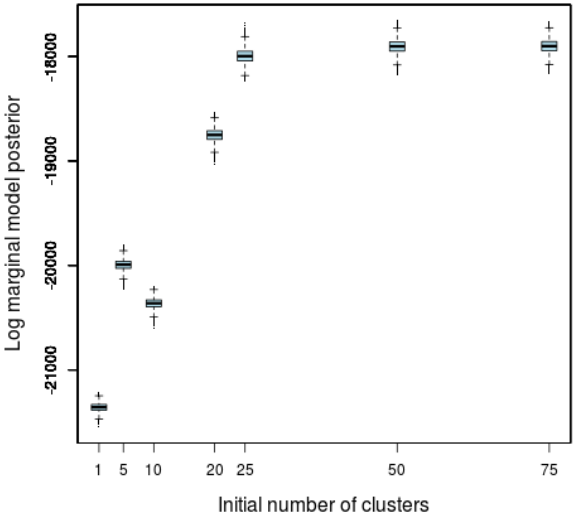
excludeY

extraYVar

$p(Z|X, Y, \alpha)$ - simulated data



$p(Z|X, Y, \alpha)$ - real data



Further work

Further work

- Mixing properties of the algorithm: further moves, tempering
- Extension to include spatial and time components
- New epidemiological and clinical applications and gene-gene and gene-environment interaction search

References

- Bigelow and Dunson (2009). Bayesian Semiparametric Joint Models for Functional Predictors. JASA, 104, 26-36.
- J. T. Molitor, M. Papathomas, M. Jerrett and S. Richardson (2010) Bayesian Profile Regression with an Application to the National Survey of Children's Health, Biostatistics, 11, 484-498.
- M. Papathomas, J. Molitor, S. Richardson, E. Riboli and P. Vineis (2011) Examining the joint effect of multiple risk factors using exposure risk profiles: lung cancer in non smokers. Environmental Health Perspectives, 119, 84-91.
- Papathomas, M , Molitor, J, Hoggart, C, Hastie, D and Richardson, S (2012) Exploring data from genetic association studies using Bayesian variable selection and the Dirichlet process: application to searching for gene-gene patterns. Genetic Epidemiology. 36:663-674
- Hastie, D. I., Liverani, S. and Richardson, S. (2013) Sampling from Dirichlet process mixture models with unknown concentration parameter: Mixing issues in large data implementations. Available at http://uk.arxiv.org/abs/1304.1778
- Liverani, S., Hastie, D. I., Papathomas, M. and Richardson, S. (2013) PReMiuM: An R package for Profile Regression Mixture Models using Dirichlet Processes. Available at http://uk.arxiv.org/abs/1303.2836