# Computation for variable selection

**David Rossell[1] and Omiros Papaspiliopoulos[2]**

[1]: University of Warwick (UK)
[2]: ICREA and UPF

Recall

$$\mathbf{t} = \mathbf{X}\mathbf{w} + \boldsymbol{\epsilon}, \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}, q\mathbf{I})$$

- $\mathbf{X}$ is $n \times p$, $\mathbf{w}$ is $p \times 1$
- Models $M_1, \ldots, M_K$ $(K = 2^p)$
- $(\mathbf{X}_k, \mathbf{w}_k)$ subsets of $(\mathbf{X}, \mathbf{w})$ for variables included by $M_k$

Posterior model probabilities

$$p(M_k \mid \mathbf{t}) = \frac{p(\mathbf{t} \mid M_k)p(M_k)}{p(\mathbf{t})}$$

Issue: if $p$ large we cannot enumerate all models

# Outline

Projection-based methods

Heuristics

Markov Chain Monte Carlo

Shrinkage priors

Idea: transform $\mathbf{x}_i \in \mathbb{R}^p$ into $\mathbf{z}_i = g(\mathbf{x}_i) \in \mathbb{R}^{\tilde{p}}$ such that $\mathbf{Z}^T\mathbf{Z}$ is diagonal ($\tilde{p} \leq p$)

$$\mathbf{t} = \mathbf{Z}\mathbf{w} + \boldsymbol{\epsilon}, \text{ where } \boldsymbol{\epsilon} \sim N(\mathbf{0}, q\mathbf{I})$$

Example: Principal components regression

Consider eigendecomposition $\mathbf{X}^T\mathbf{X} = \mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^T$

- Columns in $\mathbf{E}$ are eigenvectors, $\mathbf{E}^T\mathbf{E} = \mathbf{I}$
- $\boldsymbol{\Lambda}$ is diagonal with eigenvalues $\geq 0$

Define $\mathbf{Z} = \mathbf{X}\mathbf{E}$ (projection of $\mathbf{X}$ on $\mathbf{E}$)

$$\mathbf{Z}^T\mathbf{Z} = \mathbf{E}^T\mathbf{X}^T\mathbf{X}\mathbf{E} = \mathbf{E}^T\mathbf{E}\boldsymbol{\Lambda}\mathbf{E}^T\mathbf{E} = \boldsymbol{\Lambda}$$

# Simplifications under orthogonalization

Assume variance $q$ is known and that $p(\mathbf{w}_k) = \prod_{j=1}^{d_k} p(w_{kj})$

$$p(\mathbf{t} \mid M_k, q) = \int \frac{1}{(2\pi q)^{\frac{n}{2}}} e^{-\frac{1}{2q}(\mathbf{t}-\mathbf{Z}_k^T\mathbf{w}_k)^T(\mathbf{t}-\mathbf{Z}_k^T\mathbf{w})} p(\mathbf{w}_k) d\mathbf{w}_k =$$

$$\frac{e^{-\frac{1}{2q}\mathbf{t}^T\mathbf{t}}}{(2\pi q)^{\frac{n}{2}}} \prod_{j=1}^{d_k} e^{\frac{1}{2q}\frac{\hat{w}_{kj}^2}{\mathbf{z}_{kj}^T\mathbf{z}_{kj}}} \int e^{-\frac{v_{kj}}{2q}\left(w_{kj}-\hat{w}_{kj}\right)^2} p(w_{kj}) dw_{kj} =$$

$$\frac{e^{-\frac{1}{2q}\mathbf{t}^T\mathbf{t}}}{(2\pi q)^{\frac{n}{2}}} \prod_{j=1}^{d_k} m(\hat{w}_{kj}, v_{kj})$$

where $\hat{w}_{kj} = \frac{\mathbf{z}_{kj}^T\mathbf{t}}{(\mathbf{z}_{kj}^T\mathbf{z}_{kj})}$ is the least squares estimate and $v_{kj} = 1/(\mathbf{z}_{kj}^T\mathbf{z}_{kj})$

# Consequences

We can pre-compute $m(\hat{w}_j, v_j)$ for $j = 1, \ldots, p$

- Posterior model probabilities

$$p(M_k \mid \mathbf{t}, q) \propto p(M_k) \prod_{j \in M_k} m(\hat{w}_j, v_j)$$

- Marginal probabilities. If $p(w_1 \neq 0, \ldots, w_p \neq 0)$ is exchangeable

$$p(w_i \neq 0 \mid \mathbf{t}, q) = \frac{m(\hat{w}_j, v_j) r}{m(\hat{w}_j, v_j) r + e^{\frac{\hat{w}_j^2}{2 q v_j}} e^{-\frac{v_j \hat{w}_j^2}{2q}}}$$

where $r = p(w_j \neq 0) / p(w_j = 0)$

Consequences

1. Marginal probabilities computed with $O(p)$ computations!
2. Highest $p(M_k \mid \mathbf{t})$ corresponds to including $p(w_i \neq 0 \mid \mathbf{t}) > \frac{1}{2}$

# Remaining issues

- Model prob require proportionality constant

  $$p(M_k \mid \mathbf{t}, q) = \frac{p(M_k)p(\mathbf{t} \mid M_k, q)}{p(\mathbf{t} \mid q)} = \frac{p(M_k)p(\mathbf{t} \mid M_k, q)}{\sum_{k=1}^{K} p(\mathbf{t} \mid M_k, q)}$$

  If $K$ is large we cannot enumerate all models!
- In practice $q$ is not known. We need

  $$p(M_k \mid \mathbf{t}) = \int p(M_k \mid \mathbf{t}, q)p(q \mid M_k)dq$$

  But then $p(M_k \mid \mathbf{t})$ doesn't factor anymore!

These are exciting research questions

# Final thoughts on projections

PCA defines new uncorrelated $\mathbf{Z} = \mathbf{XE}$

- Linear in $\mathbf{X}$
- Maximizing explained variance in $\mathbf{X}$

Issues

- $\mathbf{Z}$ are not defined to predict $\mathbf{t}$. Why not seek $\tilde{z}_i \in \mathbb{R}^{\tilde{p}}$

$$\min \sum_{i=1}^{n} \left( t_i - \sum_{j=1}^{\tilde{p}} \tilde{w}_j^T \tilde{z}_{ij} \right)^2$$

- Why not consider non-linear projections (projection pursuit)

$$\min \sum_{i=1}^{n} \left( t_i - \sum_{j=1}^{\tilde{p}} f_j(\tilde{w}_j^T \tilde{z}_{ij}) \right)^2$$

- Even if BMS selects only one $\mathbf{z}_j$, this is a linear comb of all $\mathbf{x}_j$'s

# Outline

Projection-based methods

Heuristics

Markov Chain Monte Carlo

Shrinkage priors

Projection methods select $\mathbf{z}_j$'s, but we're often interested in $\mathbf{x}_j$'s.

Suppose we seek $M_{k^*} = \text{argmax}_k p(M_k \mid \mathbf{t})$ (HPM). Equivalently let $\gamma_j = I(w_j \neq 0)$, $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_p)$, we seek

$$\gamma^* = \text{argmax}_{\boldsymbol{\gamma}} \log\left(p(\mathbf{t} \mid \boldsymbol{\gamma})\right) + \log p\left(\boldsymbol{\gamma}\right) = \text{argmax}_{\boldsymbol{\gamma}} C_{\boldsymbol{\gamma}}$$

- This is an integer programming optimization problem
- Potentially many local modes
- Many algorithms out there, let's start with stepwise methods

# Heuristic 1: Stepwise methods

Stepwise forward

1. Start with null model (no predictors)
2. Add single variable $j^*$ providing best $C_\gamma$
3. Continue until $\min\{p, n\}$ variables in the model

Stepwise backward

1. Start with full model (all predictors, assumes $p \leq n$)
2. Drop variable $j^*$ resulting in best $C_\gamma$
3. Continue until no variables in the model

Hybrid stepwise: consider forward & backward moves, choose the best of the two

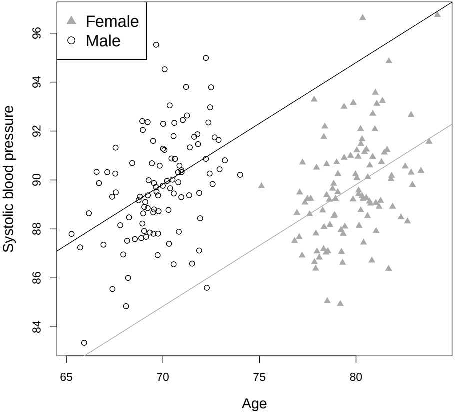
# Pros & Cons of Stepwise methods

| | Pros | Cons |
|---|---|---|
| Forward | Works for $p > n$ | Doesn't consider full Corr($\mathbf{X}$) |
| Backward | Considers full Corr($\mathbf{X}$) | Works for $p \leq n$ |
| Hybrid | Local optima less likely | Higher CPU cost |

Example

- $\mathbf{t}$: Systolic Blood pressure
- $\mathbf{x}_1$: age
- $\mathbf{x}_2$: gender

Issue: women are older than men. Correlated predictors!



True model: $\mathbf{t} = 50 + 0.5x_1 + 5x_2 + \epsilon$

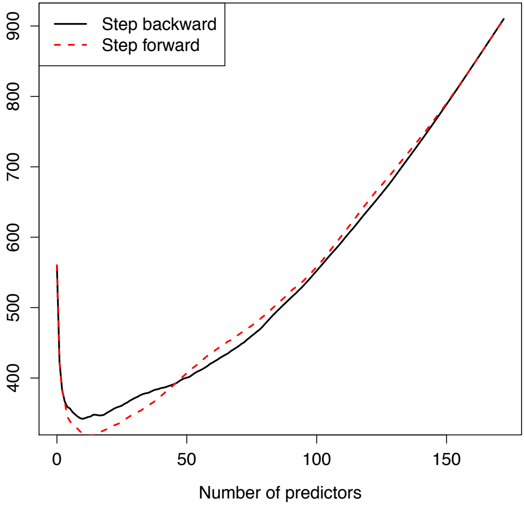
Suppose we set uniform model prior probabilities $P(\gamma)$

| | $C_\gamma$ | $P(\gamma \mid \mathbf{t})$ |
|---|---:|---:|
| Intercept | -281.4 | 1e-12 |
| Age only | -284.1 | 7e-14 |
| Gender only | -283.0 | 2.3e-13 |
| Age & Gender | -253.9 | 1 |

- Stepwise forward: start from $\gamma = (0, 0)$ including any variable decreases $C_\gamma$. Solution is $\gamma^* = (0, 0)$
- Stepwise backward: solution is $\gamma^* = (1, 1)$

# Colon cancer data (p=172,n=262)

Plotting $-C_\gamma$ (looking for minimum)



# Colon cancer data (p=172,n=262)

Assess leave-one-out cross-validated $R^2$ between $(\hat{t}, t)$

| | $p$ | $R^2$ (cross-val) |
|---|---|---|
| Least squares | 172 | 0.374 |
| Hybrid forw/back | 13 | 0.577 |

Recall that with BMA we were getting $R^2 = 0.56$ (this required some numerical approx to be discussed later on)

- Most pairwise correlations $\leq 0.5$, stepwise methods may suffer under stronger correlation
- If $\mathbf{x}_i^T \mathbf{x}_j = 0$ for all $i \neq j$ (orthogonal) and $q$ known then stepwise methods find the global maximum

Stepwise methods find $\gamma^*$, not $p(\gamma^* \mid \mathbf{t})$. Hard to assess uncertainty

# Heuristic 2: Restrict the model space

Idea: instead of considering $2^p$ models, use some fast algorithm to focus on “promising” ones (e.g. stepwise forward)

Example

1. Use LASSO to find sequence of models $\gamma^{(1)}, \ldots, \gamma^{\min\{n,p\}}$ with $0, 1, \ldots, \min\{p, n\}$ variables
2. Compute $p(\mathbf{t} \mid \gamma^{(k)})p(\gamma^{(k)})$ for each model to find HPM
3. Lower bound for $p(\gamma^* \mid \mathbf{t})$ given by

$$\frac{p(\mathbf{t} \mid \gamma^*)p(\gamma^*)}{\sum_k p(\mathbf{t} \mid \gamma^{(k)})p(\gamma^{(k)})}$$

# Using the LASSO

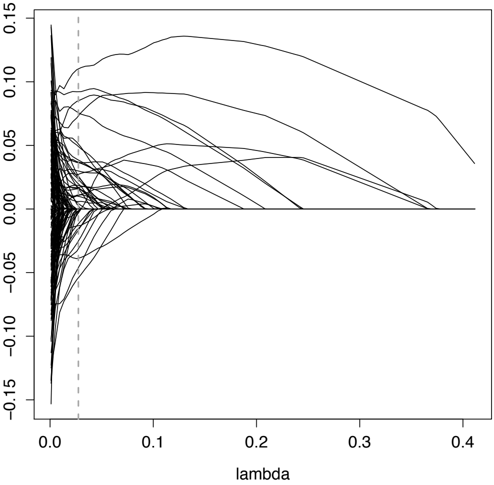
Let $\lambda > 0$ be a penalization parameter, the goal is

$$\min_{\mathbf{w}}(\mathbf{t} - \mathbf{X}\mathbf{w})^T(\mathbf{t} - \mathbf{X}\mathbf{w}) + \lambda \sum_{j=1}^{p} |w_j|$$

Convex (sum of 2 convex functions) $\Rightarrow$ unique minimum. Many optimization algorithms

- Optimize each $w_j$ with other $w_k$ fixed ($k \neq l$). Quadratic in $w_j$
- Least Angle Regression (LAR) algorithm: find $\hat{\mathbf{w}}$ for all possible $\lambda$ in $O(\min\{n, p\})$ steps
- $\ldots$

# Colon cancer data. LASSO solution path: $\hat{w}_j$ vs. $\lambda$



# Refining the approximation

LASSO seeks small

$$(\mathbf{t}-\mathbf{X}\mathbf{w})^T(\mathbf{t}-\mathbf{X}\mathbf{w})+\lambda\sum_{j=1}^{p}|w_j|$$

BMS seeks high $\log p(\mathbf{t} \mid M_k)+\log p(M_k)\approx$

$$-\frac{1}{2}(\mathbf{t}-\mathbf{X}_k\hat{\mathbf{w}}_k)^T(\mathbf{t}-\mathbf{X}_k\hat{\mathbf{w}}_k)-\frac{d_k}{2}\log|\mathbf{X}_k^T\mathbf{X}_k|+\log p(M_k)$$
$$\approx -\frac{1}{2}(\mathbf{t}-\mathbf{X}\hat{\mathbf{w}})^T(\mathbf{t}-\mathbf{X}\hat{\mathbf{w}})-\frac{1}{2}h\left(d_k\right)$$

- $\sum_{j=1}^{p}|w_j|$ is $L_1$ norm, $d_k=\sum_{j=1}^{p}I(w_j\neq 0)$ is $L_0$ norm
- Better approximations for $L_0$ norm possible (*e.g.* adaptive LASSO)

Again, pretty much open research...

# Heuristic 3: pre-screening

1. Use a quick rule to select $\tilde{p} \ll p$ variables. $\mathbf{X} \rightarrow \tilde{\mathbf{X}}$
2. Run full BMS on $\tilde{\mathbf{X}}$

Example: univariate predictive effect followed by FDR

1. For $j = 1, \ldots, p$ fit $\mathbf{t} = w_j \mathbf{x}_j + \epsilon$
2. Obtain P-value for $\hat{w}_j$, or perhaps $P(w_j \neq 0 \mid \mathbf{t})$
3. Adjust so that FDR$< \alpha$ (*e.g.* Benjamini-Hochberg) or $E(\text{FDP} \mid \mathbf{t}) < \alpha$ (Bayesian FDR)
4. Let $\tilde{\mathbf{X}}$ contain all variables passing FDR criterion

Iterative screening methods also available (*Sure Independence Screening, Iterative Sure Independence Screening...*)

# Outline

Projection-based methods

Heuristics

Markov Chain Monte Carlo

Shrinkage priors

# Markov Chain Monte Carlo (MCMC)

Idea: $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_p)$ is a random variable with distribution $p(\boldsymbol{\gamma} \mid \mathbf{t})$

1. Obtain a sample $\boldsymbol{\gamma}^{(1)}, \ldots, \boldsymbol{\gamma}^{(L)}$ from $p(\boldsymbol{\gamma} \mid \mathbf{t})$
2. Estimate $\hat{p}(\boldsymbol{\gamma} = \mathbf{g} \mid \mathbf{t}) = \frac{1}{L} \sum_{l=1}^{L} I(\boldsymbol{\gamma}^{(l)} = \mathbf{g})$

MCMC: family of methods to sample from $p(\boldsymbol{\gamma} \mid \mathbf{t})$

- Choose an arbitrary initial $\boldsymbol{\gamma}^{(0)}$
- Transition from $\boldsymbol{\gamma}^{(l)} \rightarrow \boldsymbol{\gamma}^{(l+1)}$ using Markov Chain
- Set $p(\boldsymbol{\gamma}^{(l+1)} \mid \boldsymbol{\gamma}^{(l)})$ so that $p(\boldsymbol{\gamma} \mid \mathbf{t})$ is the stationary distribution

The larger $p(\boldsymbol{\gamma} = \mathbf{g} \mid \mathbf{t})$ the more likely we visit $\mathbf{g}$. We cannot enumerate all models, so we focus on those with high $p(\boldsymbol{\gamma} = \mathbf{g} \mid \mathbf{t})$

# Gibbs sampling

Let $\boldsymbol{\gamma}_{-j}$ be $\boldsymbol{\gamma}$ after excluding $\gamma_j$. Set $\boldsymbol{\gamma}^{(l)} = \boldsymbol{\gamma}^{(l-1)}$, then

Set $\gamma_j^{(l)} = 1$ with probability $p(\gamma_j = 1 \mid \boldsymbol{\gamma}_{-j}^{(l)}, \mathbf{t}) = \frac{p(\gamma_j=1, \boldsymbol{\gamma}_{-j}^{(l)} | \mathbf{t})}{p(\boldsymbol{\gamma}_{-j}^{(l)} | \mathbf{t})} =$

$$\frac{p(\mathbf{t} \mid \gamma_j = 1, \boldsymbol{\gamma}_{-j}^{(l)}) p(\gamma_j = 1, \boldsymbol{\gamma}_{-j}^{(l)})}{p(\mathbf{t} \mid \gamma_j = 1, \boldsymbol{\gamma}_{-j}^{(l)}) p(\gamma_j = 1, \boldsymbol{\gamma}_{-j}^{(l)}) + p(\mathbf{t} \mid \gamma_j = 0, \boldsymbol{\gamma}_{-j}^{(l)}) p(\gamma_j = 0, \boldsymbol{\gamma}_{-j}^{(l)})}$$

else set $\gamma_j^{(l)} = 0$. Repeat for $j = 1, \ldots, p$, $l + 1, \ldots, L$.

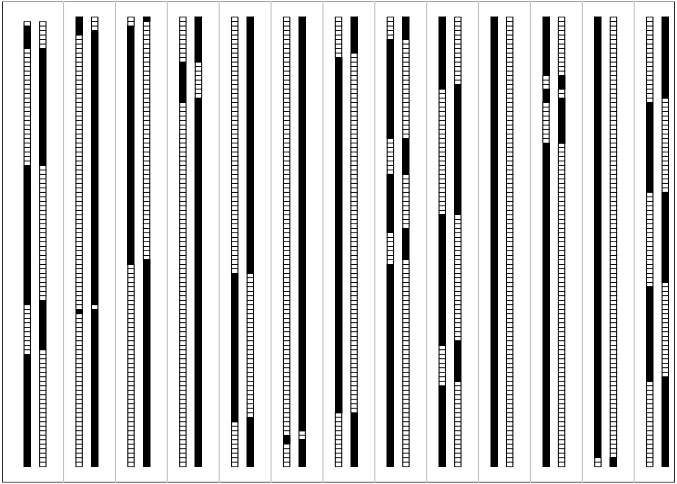
- Each update considers only 2 models
- Similar to stepwise methods with probabilistic updates

# A bivariate example

| Model | $p(M_k \mid \mathbf{t})$ |
|---|---|
| $\gamma_1 = 0, \gamma_2 = 0$ | 0.005 |
| $\gamma_1 \neq 0, \gamma_2 = 0$ | 0.49 |
| $\gamma_1 = 0, \gamma_2 \neq 0$ | 0.49 |
| $\gamma_1 \neq 0, \gamma_2 \neq 0$ | 0.005 |

Run 10,000 iterations with

$$p(\gamma_1 = 1 \mid \gamma_2 = 0) = 0.49/0.495 = 0.989$$
$$p(\gamma_1 = 1 \mid \gamma_2 = 1) = 0.005/0.495 = 0.01$$
$$p(\gamma_2 = 1 \mid \gamma_1 = 0) = 0.49/0.495 = 0.989$$
$$p(\gamma_2 = 1 \mid \gamma_1 = 1) = 0.005/0.495 = 0.01$$



$\hat{p}(M_1 \mid \mathbf{y}) = 0.005, \hat{p}(M_2 \mid \mathbf{y}) = 0.498, \hat{p}(M_3 \mid \mathbf{y}) = 0.492, \hat{p}(M_4 \mid \mathbf{y}) = 0.004$

# Other MCMC methods

Many other algorithms available

- Scan variables in random order (random scan Gibbs)
- Update several variables at a time (block Gibbs)
- Consider multiple moves
- Jointly sample $(\boldsymbol{\gamma}, \mathbf{w})$
- ...

Example: Metropolized-Gibbs

Suppose $\gamma_j^{(l)} = g_j$. Set $\gamma_j^{(l+1)} = 1 - g_j$ with probability $\min\{1, u\}$,

$$u = \frac{p(\mathbf{t} \mid \gamma_j = 1 - g_j, \boldsymbol{\gamma}_{-j}^{(l)}) p(\gamma_j = 1 - g_j, \boldsymbol{\gamma}_{-j}^{(l)})}{p(\mathbf{t} \mid \gamma_j = g_j, \boldsymbol{\gamma}_{-j}^{(l)}) p(\gamma_j = g_j, \boldsymbol{\gamma}_{-j}^{(l)})}$$

Increases chance of moving from $g_j$ to $1 - g_j$ (lower $\mathrm{Cor}(\boldsymbol{\gamma}^{(l)}, \boldsymbol{\gamma}^{(l+1)})$)

# MCMC convergence

For large enough $l$ we sample from $p(\gamma \mid \mathbf{t})$. Ideally, large means that the chain has converged

**Def.** Let $p^l(\gamma^{(l)} \mid \gamma^{(0)})$ be the distribution of $\gamma^{(l)}$. If

$$p^l(\gamma = \mathbf{g} \mid \gamma^{(0)}) = p(\gamma = \mathbf{g} \mid \mathbf{t}) \tag{1}$$

for any $\mathbf{g}$ (in a set of probability one) we say the chain has converged

We cannot check convergence from (1) (rhs not available)

- Monitor characteristics of $\gamma^{(l)}$, check they "stabilized"
- Run multiple independent chains and compare results

# Example in R (p=50,n=100)

Simulate data

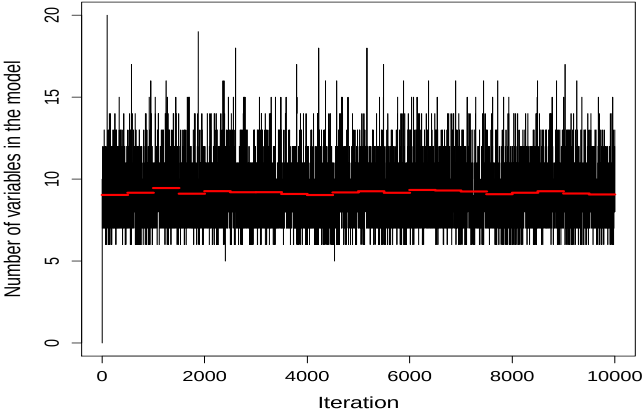
```
library(mvtnorm)
w <- c(rep(0,40),rep(.5,5),rep(1,5))
sigma <- diag(length(w))
sigma[upper.tri(sigma)] <- 0.75
sigma[lower.tri(sigma)] <- 0.75
x <- rmvnorm(100,sigma=sigma)
y <- x %*% matrix(w,ncol=1) + rnorm(nrow(x))
```

Run Bayesian model selection

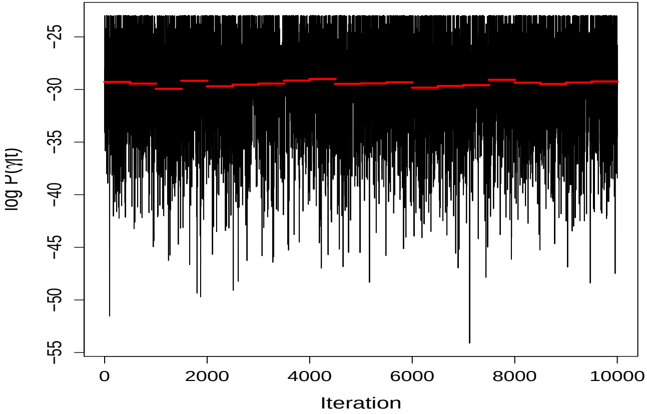
```
library(mombf)
fit1 <- modelSelection(y=y,x=x,niter=10^4,
  priorCoef=zellnerprior(tau=nrow(x)),
  priorDelta=modelbbprior(),burnin=0)
```

# Monitor model size $\sum_{j=1}^{p} \gamma_j^{(l)}$

The chain started at $\gamma_1^{(0)} = \ldots = \gamma_p^{(0)} = 0$

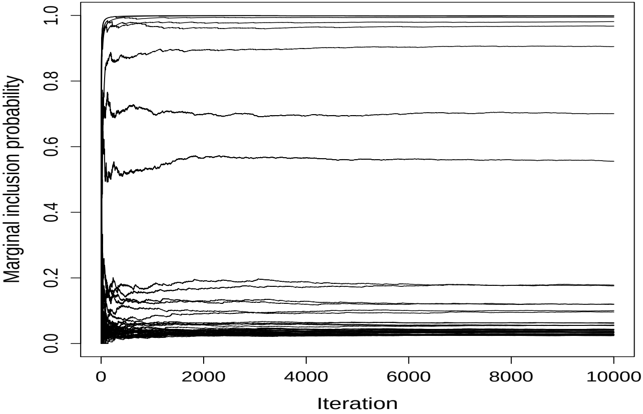


# Monitor $p(\mathbf{t} \mid \gamma^{(l)})p(\gamma^{(l)})$



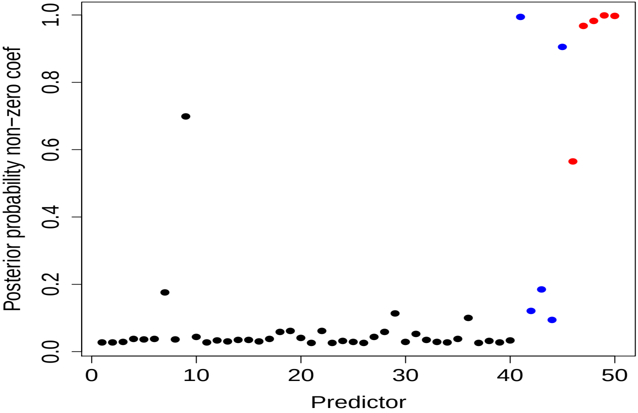
We can also monitor largest $p(\mathbf{t} \mid \gamma^{(l)})p(\gamma^{(l)})$ so far

# Monitor $\hat{p}(\gamma_j = 1 \mid \mathbf{t})$



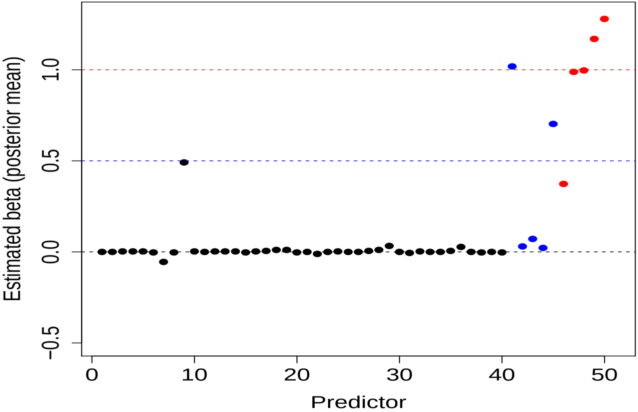
# What do results look like?

Marginal inclusion probabilities $\hat{p}(\gamma_j = 1 \mid \mathbf{t})$



# What do results look like?

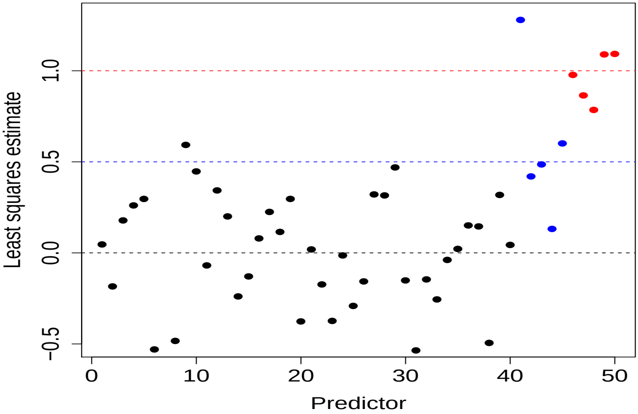
BMA $\hat{E}(w_j \mid \mathbf{t})$



Root mean square error $E^{\frac{1}{2}}\left(\sum_{j=1}^{p}(E(w_j \mid \mathbf{t}) - w_j)^2\right) \approx 1.17$

# What do results look like?

Least-squares estimate $\hat{w}_j$



Root mean square error $E^{\frac{1}{2}}\left(\sum_{j=1}^{p}(\hat{w}_j - w_j)^2\right) \approx 2.02$

# Digression: computer implementation

MCMC may revisit previous $\gamma$

- Convenient to store $C_\gamma = \log p(\mathbf{t} \mid \gamma) + \log p(\gamma)$
- If $p(\gamma \in A \mid \mathbf{t}) \approx 1$ for a small set $A$, upon convergence MCMC spends most time revisiting models.
- $\gamma = (0, 1, 0, \ldots, 0)$ is the binary code for integer

$$i(\gamma) = 0 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + \ldots + 0 \times 2^p - 1$$

We can store $C_\gamma$ in a quickly accessible vector

Parallel computing for multiple chains / moves

Computing $p(\mathbf{t} \mid \gamma^{(l+1)})$ starting from $p(\mathbf{t} \mid \gamma^{(l)})$ (*e.g.* matrix inversion)

# Outline

Projection-based methods

Heuristics

Markov Chain Monte Carlo

Shrinkage priors

# Shrinkage priors

Idea: Instead of considering $2^p$ models, focus on single model with $p$ variables and encourage that $E(w_j \mid \mathbf{t})$ is shrunk to 0

Suppose we seek the posterior mode

$$(\mathbf{w}, q) = \text{argmax}_{(\mathbf{w},q)} \log\left(p(\mathbf{w}, q \mid \mathbf{t})\right) =$$

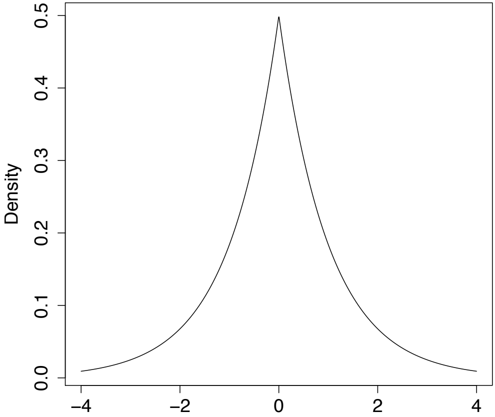
$$\propto -\frac{n}{2}\log(q) - \frac{1}{2q}\sum_{i=1}^{n}(\mathbf{y}_i - \mathbf{x}_i'\mathbf{w})^2 + \log(p(\mathbf{w}, q))$$

- Terms 1-2 are the likelihood function
- $\log(p(\mathbf{w}, q))$ reinforces certain param values (i.e. penalty)

Posterior mode equivalent to maximizing penalized likelihood!

# Double exponential (Laplace) distribution

Consider prior $p(w_j \mid q) = \frac{\lambda}{2q}\exp(-\frac{\lambda}{q}|w_j|)$



# Bayesian LASSO

Then maximizing $\log\left(p(\mathbf{w}, q \mid \mathbf{t})\right) \propto$

$$-\frac{n}{2}\log p(q) - \frac{1}{2q}\sum_{i=1}^{n}(\mathbf{y}_i - \mathbf{x}_i'\mathbf{w})^2 - \lambda\frac{1}{q}\sum_{j=1}^{p}|w_j| + \log p(q)$$

with respect to $\mathbf{w}$ is equivalent to minimizing

$$\frac{1}{2}\sum_{i=1}^{n}(\mathbf{y}_i - \mathbf{x}_i'\mathbf{w})^2 + \lambda\sum_{j=1}^{p}|w_j|$$

The posterior mode is equivalent to the LASSO solution

# Shrinkage priors

Stronger shrinkage than LASSO is possible

- Double-exponential places higher prior prob on $w_j \approx 0$ than Normal
- One can assign even larger prior prob, *e.g.* $\lim_{w_j \to 0} p(w_j) = \infty$
- We can shrink further by introducing prior dependence across $w_j$'s

Properties of shrinkage priors

- Posterior mode may be sparse, but $E(\mathbf{w} \mid \mathbf{t})$ is not
- It doesn't make sense to compute $P(w_j \neq 0 \mid \mathbf{t})$. Hard to assess uncertainty in the selected model
- Relative to regular LASSO, Bayesian LASSO gives posterior credibility intervals (uncertainty in the parameter estimates)
- Obtaining more than just the mode can be computationally demanding

# TGFB study ($n = 262$)

Predict TGFB from

- $p =$172 promising genes
- $p =$10,172 genes

Compare

- BMS: MOM, Hyper-g, Benchmark prior + Beta-Binomial(1,1)
- Bayesian LASSO
- Penalized likelihood: LASSO, adaptive LASSO, SCAD

Evaluate

- Mean number of predictors
- $R^2$ between $(y_i, \hat{y}_i)$ (leave-one-out cross-validation)
- CPU time (Mac laptop, single core)

# TGFB study ($n = 262$)

| | $p = 172$ | | $p =$10,172 | | |
|---|---|---|---|---|---|
| | $\bar{p}$ | $R^2$ | $\bar{p}$ | $R^2$ | CPU time |
| MOM ($10^7$ updates) | 4.3 | 0.566 | 6.5 | 0.617 | 1m 52s |
| Hyper-g ($10^7$ updates) | 11.3 | 0.562 | 26.4 | 0.522 | 11m 49s |
| BenchP ($10^7$ updates) | 4.2 | 0.562 | 3.0 | 0.586 | 1m 23s |
| BLASSO ($2.5 \cdot 10^5$ iter) | $104^*$ | 0.580 | $100^*$ | 0.598 | 3.6h |
| SCAD (10-fold CV) | 28 | 0.560 | 81 | 0.535 | 17s |
| LASSO (10-fold CV) | 42 | 0.586 | 159 | 0.570 | 24s |
| AdaLASSO (10-fold CV) | 24 | 0.569 | 10 | 0.536 | 2m 49s |

$^*$: $|E(\theta_j \mid Y)| > 0.01$

R package `mombf` (MOM,iMOM, BenchP), hyper-g (`BAS`), BLASSO (`BLR`), LASSO, SCAD (`ncvreg`), AdaLASSO (`parcor`)

# Final thoughts

Extending BMS to truly high-dimensions is an open challenge

- Can we characterize when it leads to better estimates?
- When can we expect to recover the "true model"
- Can we address computational bottlenecks?

Some (many?) leading experts state BMS computationally unfeasible and propose alternatives, but

*An approximate answer to the right question is worth a great deal more than a precise answer to the wrong question*

John W. Tukey