Module 22: Bayesian Methods
Lectures 6: Model selection and averaging

Ken Rice

Department of Biostatistics
University of Washington

Outline

Model selection

Stochastic search

Model selection and averaging

Diabetes example:

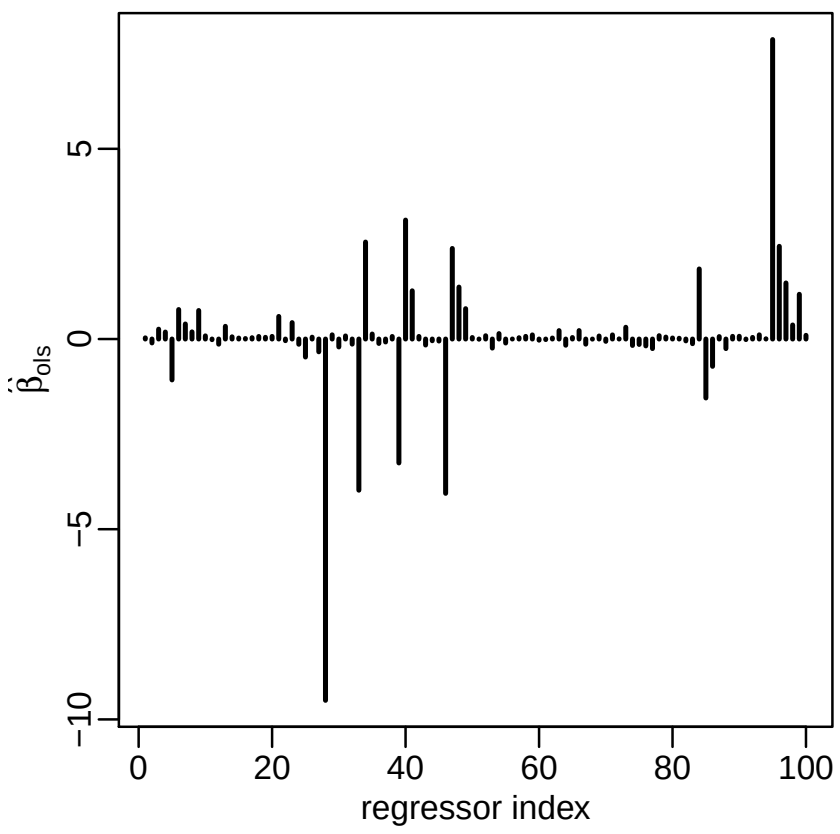
- 342 subjects
- y_i = diabetes progression
- $\mathbf{x}_i$ = explanatory variables.

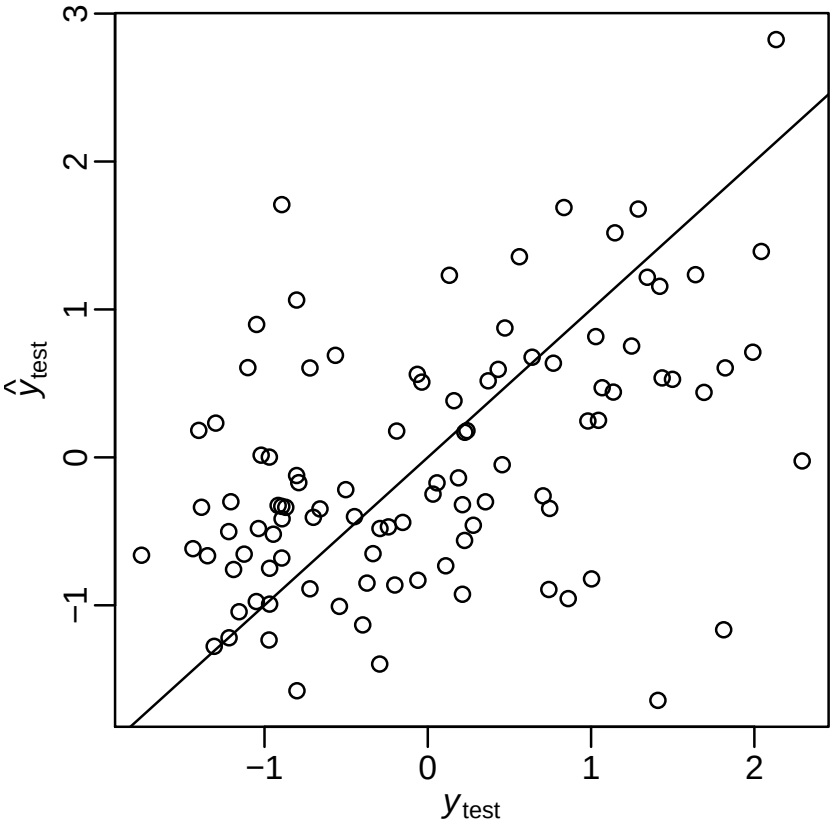
Each $\mathbf{x}_i$ includes

- 13 subject specific measurements $(x_{\text{age}}, x_{\text{sex}}, \ldots)$;
- $78 = \binom{13}{2}$ interaction terms $(x_{\text{age}} \cdot x_{\text{sex}}, \ldots)$;
- 9 quadratic terms (x_{sex} and three genetic variables are binary)

100 explanatory variables total!

OLS regression





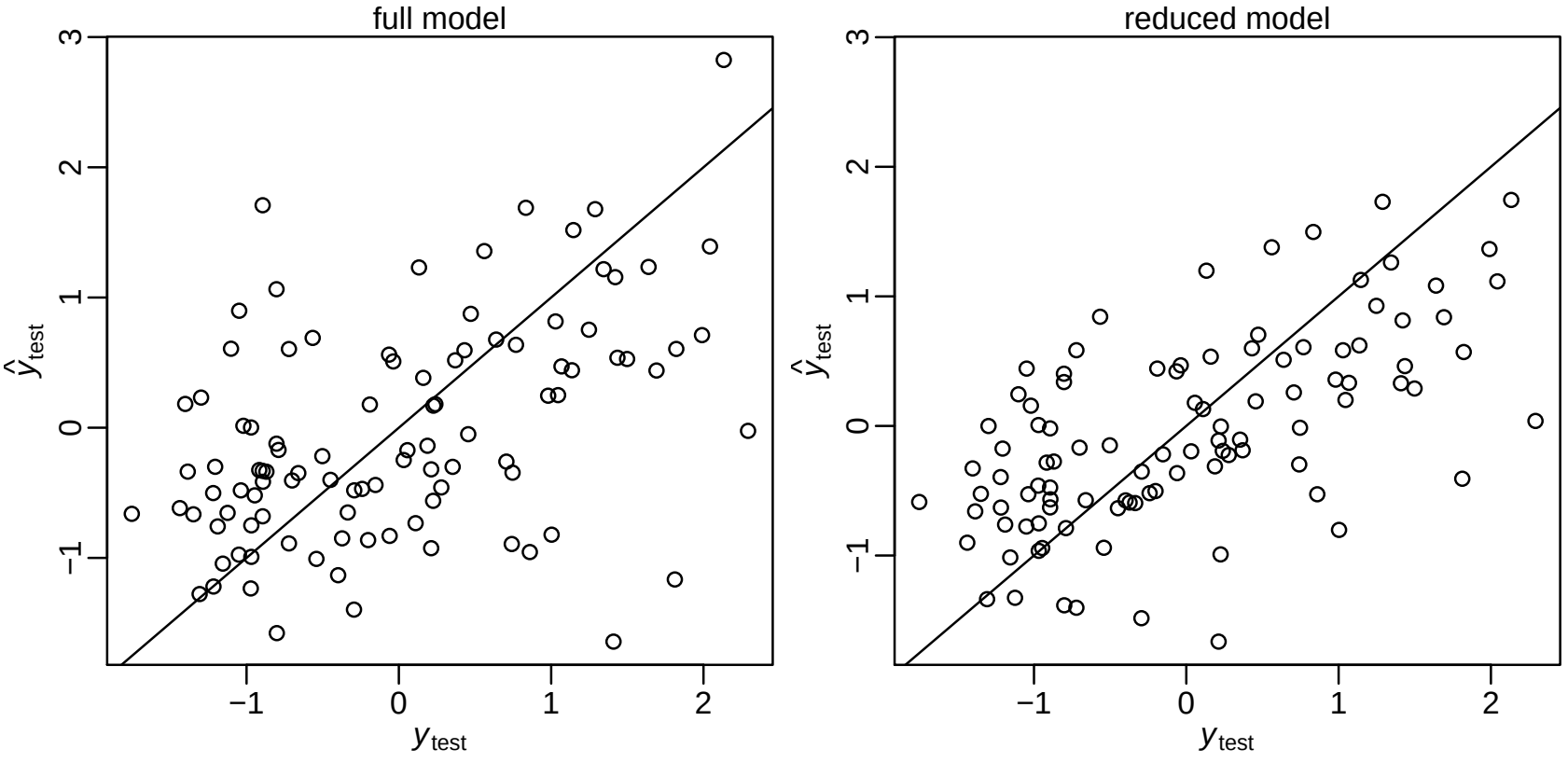
$$\frac{1}{100}\sum(y_{\text{test},i} - \hat{y}_{\text{test},i})^2 = 0.9263277$$

$$\frac{1}{100}\sum(y_{\text{test},i} - 0)^2 = \sum y_{\text{test},i}^2 = 1.0094689$$

Backwards elimination

1. Obtain the estimator $\hat{\boldsymbol{\beta}}_{\text{ols}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ and its t-statistics.
2. If there are any regressors j such that $|t_j| < t_{\text{cutoff}}$,
   2.1 find the regressor $j_{\min}$ having the smallest value of $|t_j|$ and remove column $j_{\min}$ from $\mathbf{X}$.
   2.2 return to step 1.
3. If $|t_j| > t_{\text{cutoff}}$ for all variables j remaining in the model, then stop.

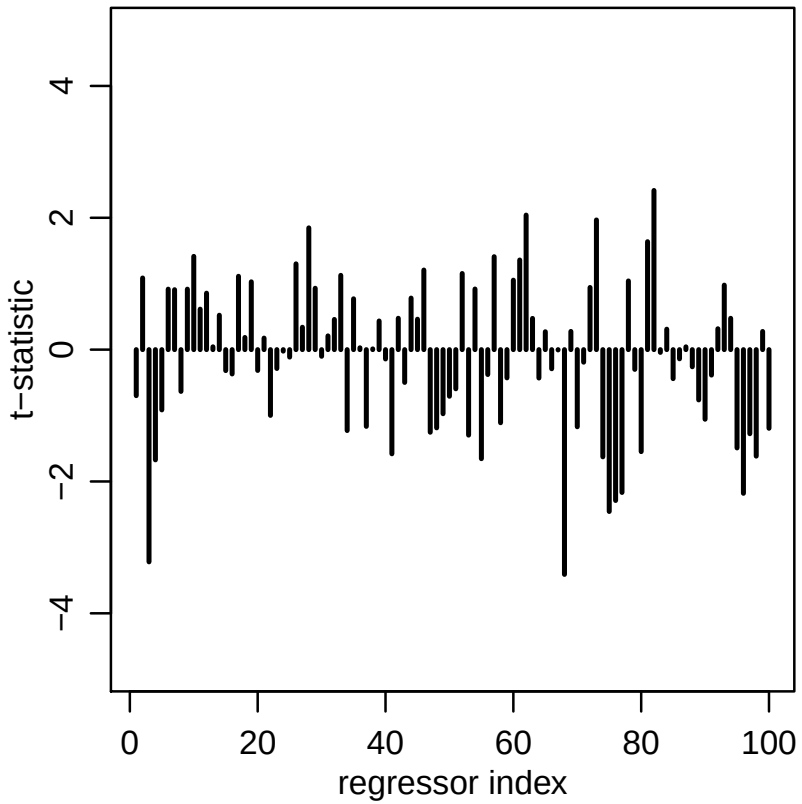
Backwards elimination



$$\frac{1}{100}\sum(y_{\text{test},i} - \hat{y}_{\text{test}^{bel},i})^2 = 0.6392334$$

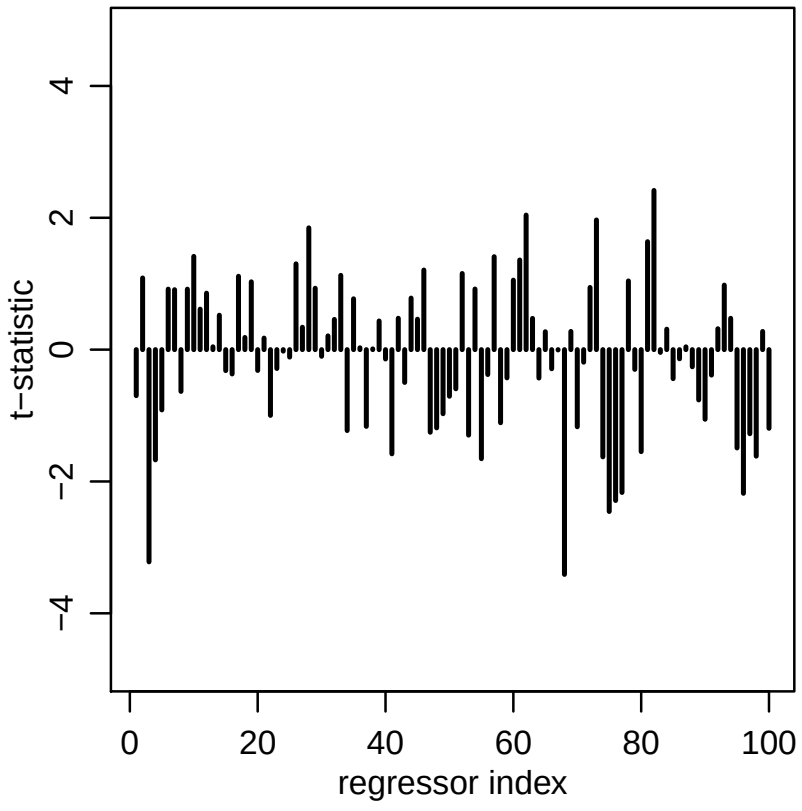
Spurious associations

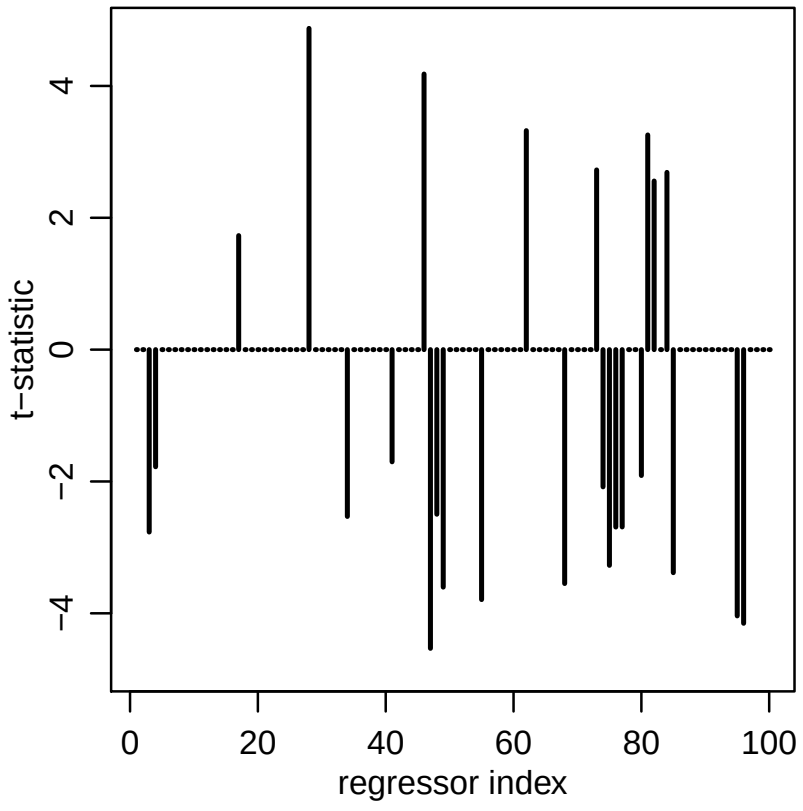
Now try modeling permuted $y_{\pi(i)} = \boldsymbol{\beta}^T \mathbf{x}_i + \epsilon_i$ (and backwards-select)



Spurious associations

Now try modeling permuted $y_{\pi(i)} = \beta^T \mathbf{x}_i + \epsilon_i$





Spurious associations

```
sum(abs(t.bslperm)>2 )

## [1] 21

sum(abs(t.bslperm)>3 )

## [1] 12

sum(abs(t.bslperm)>4 )

## [1] 5
```

- 21 regressors have t-stats > 2 ($p \approx 0.05$)
- 12 regressors have t-stats > 3 ($p \approx 0.003$)
- 5 regressors have t-stats > 4 ($p \approx 0.00006$)

Often want some way to pick a sparse model – but this approach is not smart

Bayesian model selection

Prior belief: $\beta_j \approx 0$ for many j's.

Formulation: Write $\beta_j = z_j \times b_j$, where $z_j \in \{0, 1\}$ and $b_j \in \mathbb{R}$.

$$y_i = z_1 b_1 x_{i,1} + \cdots + z_p b_p x_{i,p} + \epsilon_i.$$

For example, in the FTO experiment,

$$\begin{aligned}
\mathrm{E}[Y|\mathbf{x}, \mathbf{b}, \mathbf{z} = (1,0,1,0)] &= b_1 x_1 + b_3 x_3 \\
&= b_1 + b_3 \times \text{age} \\
\mathrm{E}[Y|\mathbf{x}, \mathbf{b}, \mathbf{z} = (1,1,0,0)] &= b_1 x_1 + b_2 x_2 \\
&= b_1 + b_2 \times \text{group} \\
\mathrm{E}[Y|\mathbf{x}, \mathbf{b}, \mathbf{z} = (1,1,1,0)] &= b_1 x_1 + b_2 x_2 + b_3 x_3 \\
&= b_1 + b_2 \times \text{group} + b_3 \times \text{age}.
\end{aligned}$$

Can think of each value of $\mathbf{z} = (z_1, \ldots, z_p)$ representing a *different model*.

Bayesian model selection

Or, think of z_j as unknown components in one (big) model – written informally as;

$$\begin{aligned} z_j &\overset{\text{iid}}{\sim} Bern(0.5) \\ b_j &\sim p(b_j) \\ \epsilon_i &\overset{\text{iid}}{\sim} N(0, \sigma^2) \\ \sigma^2 &\sim p(\sigma^2) \\ y_i &= z_1 b_1 x_{i,1} + \cdots + z_p b_p x_{i,p} + \epsilon_i \end{aligned}$$

Each of the 2^p possible values of of $\mathbf{z}$ has a posterior probability. (In the prior we treat them as a 'coin toss', equally likely to be 'in' or 'out'.)

Bayesian model comparison

Posterior probability

$$p(\mathbf{z}|\mathbf{y},\mathbf{X}) = \frac{p(\mathbf{z})p(\mathbf{y}|\mathbf{X},\mathbf{z})}{p(\mathbf{y}|\mathbf{X})}$$

Model comparison

$$\begin{aligned} \frac{p(\mathbf{z}_a|\mathbf{y},\mathbf{X})}{p(\mathbf{z}_b|\mathbf{y},\mathbf{X})} &= \frac{p(\mathbf{z}_a)}{p(\mathbf{z}_b)} \times \frac{p(\mathbf{y}|\mathbf{X},\mathbf{z}_a)}{p(\mathbf{y}|\mathbf{X},\mathbf{z}_b)} \\ \text{posterior odds} &= \text{prior odds} \times \text{“Bayes factor”} \end{aligned}$$

Note that the Bayes Factor (BF) does not depend on the prior for $\mathbf{z}$ – so the ‘coin toss’ prior is not crucial for this approach.

Parsimony

The formula for $p(\mathbf{y}|\mathbf{X}, \mathbf{z})$ is messy, but

$$\frac{p(\mathbf{y}|\mathbf{X}, \mathbf{z}_a)}{p(\mathbf{y}|\mathbf{X}, \mathbf{z}_b)} = (1+n)^{(p_{z_b} - p_{z_a})/2} \left(\frac{s^2_{z_a}}{s^2_{z_b}}\right)^{1/2} \times \left(\frac{s^2_{z_b} + \mathrm{SSR}^{z_b}_g}{s^2_{z_a} + \mathrm{SSR}^{z_a}_g}\right)^{(n+1)/2}.$$

A model $\mathbf{z}_a$ is penalized if;

- it is too complex (nuber of covariates p_A is large)
- it doesn't fit well (SSR^a_g is large)

FTO example

$$\begin{aligned} \mathrm{E}[Y_i|\boldsymbol{\beta}, \mathbf{x}_i] &= \beta_1 x_{i,1} + \beta_2 x_{i,2} + \beta_3 x_{i,3} + \beta_4 x_{i,4} \\ &= \beta_1 + \beta_2 \times \mathsf{grp}_i + \beta_3 \times \mathsf{age}_i + \beta_4 \times \mathsf{grp}_i \times \mathsf{age}_i \,. \end{aligned}$$

effect of group $\Leftrightarrow$ one of more of β_2, β_4 not zero

z	model	$\log p(\mathbf{y}\|\mathbf{X}, \mathbf{z})$	$p(\mathbf{z}\|\mathbf{y}, \mathbf{X})$
(1,0,0,0)	β_1	-71.82	0
(1,1,0,0)	$\beta_1 + \beta_2 \times \mathsf{grp}_i$	-70.04	0
(1,0,1,0)	$\beta_1 + \beta_3 \times \mathsf{age}_i$	-67.04	0
(1,1,1,0)	$\beta_1 + \beta_2 \times \mathsf{grp}_i + \beta_3 \times \mathsf{age}_i$	-61.19	0.63
(1,1,1,1)	$\beta_1 + \beta_2 \times \mathsf{grp}_i + \beta_3 \times \mathsf{age}_i + \beta_4 \times \mathsf{grp}_i \times \mathsf{age}_i$	-61.72	0.37

$$\begin{aligned} \Pr(\beta_2 \text{ or } \beta_4 \neq 0) &= 0.60 \\ \Pr(\beta_2 \text{ or } \beta_4 \neq 0|\mathbf{y}, \mathbf{X}) &\approx 1 \end{aligned}$$

High dimensional regression

Diabetes example: $p = 100 \Rightarrow 2^{100} \approx 10^{30}$ models to consider.

We can't compute $p(\mathbf{z}|\mathbf{y}, \mathbf{X})$ for each $\mathbf{z}$. Instead, we hope to

- search for models $\mathbf{z}$ with high posterior probability;
- approximate $\beta_j = z_j \times b_j$ for each j;
- build a predictive model for $\mathbf{y}$.

This can be achieved via a Monte Carlo method known as *Gibbs sampling*.

The Gibbs sampler

Goal: A *Monte Carlo* approximation to $p(x, y, z)$

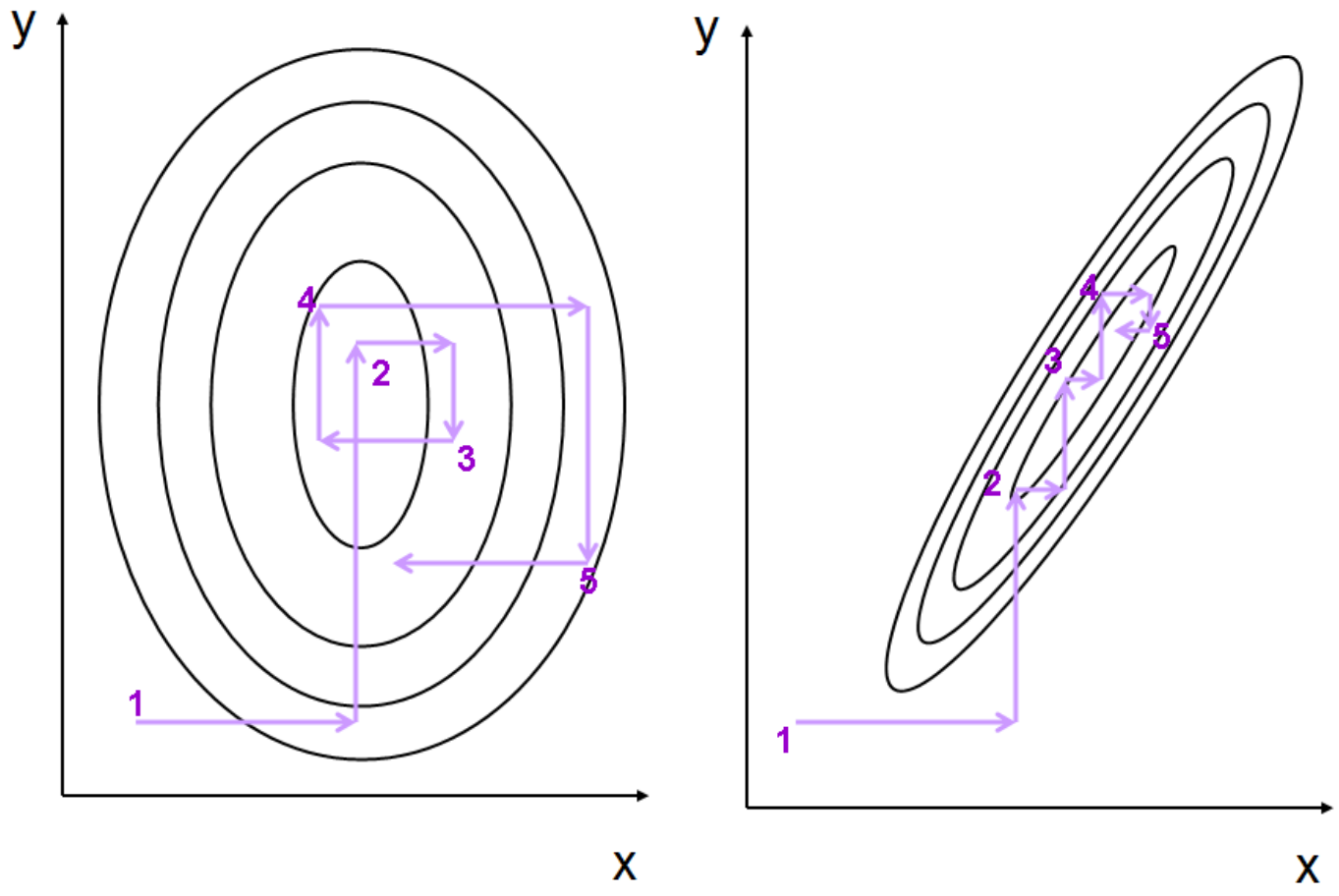
Given $\{x^{(s)}, y^{(s)}, z^{(s)}\}$,

1. simulate $x^{(s+1)} \sim p(x|y^{(s)}, z^{(s)})$,
2. simulate $y^{(s+1)} \sim p(y|x^{(s+1)}, z^{(s)})$,
3. simulate $z^{(s+1)} \sim p(z|x^{s+1)}, y^{(s+1)})$.

This generates $\{x^{(s+1)}, y^{(s+1)}, z^{(s+1)}\}$ – and then 'go round' again, many times.
Repeated many times, this generates $\{x^{(1)}, y^{(1)}, z^{(1)}\}, \ldots, \{x^{(S)}, y^{(S)}, z^{(S)}\}$

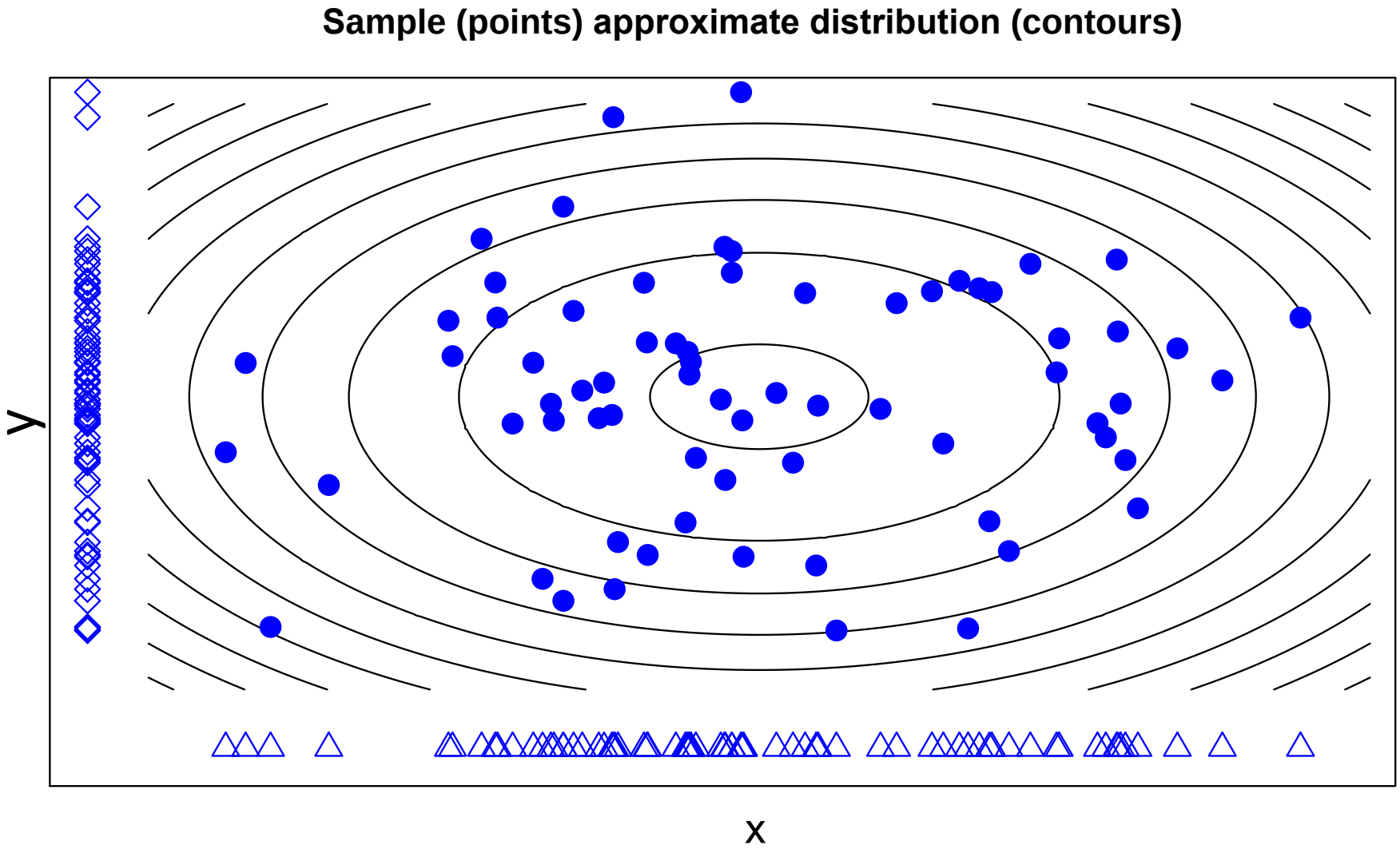
The Gibbs sampler

For a couple of two-dimensional examples;



The Gibbs sampler

Output from a short sampler;



The Gibbs sampler

Repeated many times, this generates $\{x^{(1)}, y^{(1)}, z^{(1)}\}, \ldots, \{x^{(S)}, y^{(S)}, z^{(S)}\}$

The distribution of this sequence *approximates* $p(x, y, z)$:

$$\begin{aligned}
\frac{1}{S}\sum x^{(s)} &\approx \mathrm{E}[x] = \int x\ p(x,y,z)\ dx\ dy\ dz \\
\frac{\#(x^{(s)} \in A)}{S} &\approx \Pr(x \in A) = \int\int\int_A p(x,y,z)\ dx\ dy\ dz \\
\frac{\#(\{x^{(s)}, y^{(s)}, z^{(s)}\} \in B)}{S} &\approx \int\int_B\int p(x,y,z)\ dx\ dy\ dz
\end{aligned}$$

By necessity, the sequence will frequently visit regions where $p(x, y, z)$ is large.

Gibbs sampling for model selection

Goal Approximate $p(z_1, \ldots, z_p | \mathbf{y}, \mathbf{X})$.

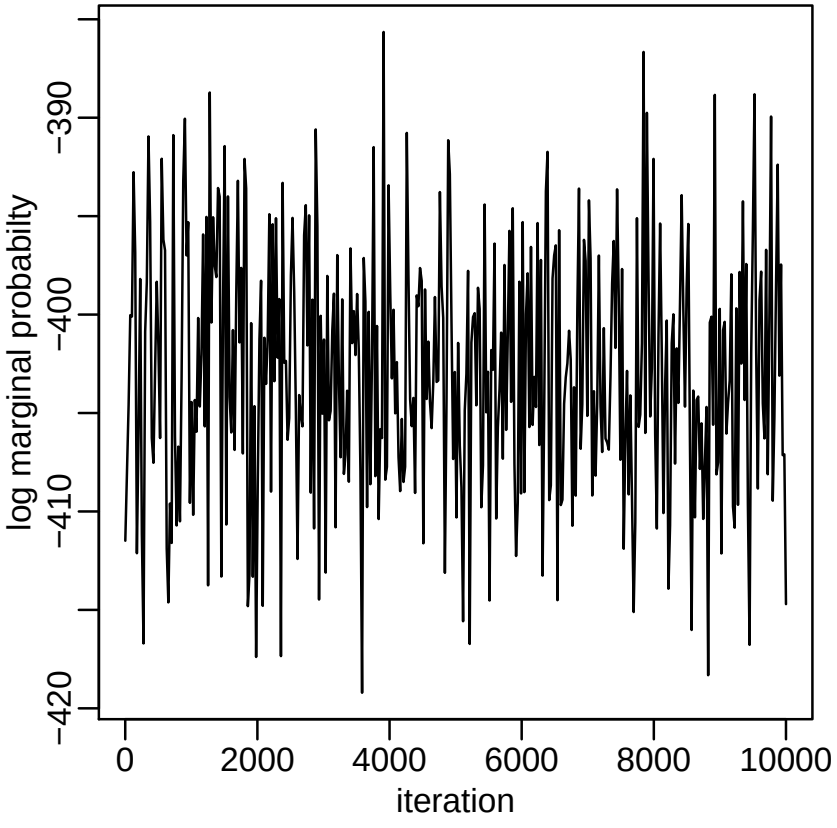
Gibbs sampler: Given $\mathbf{z}^{(s)} = (z_1^{(s)}, \ldots, z_p^{(s)})$,

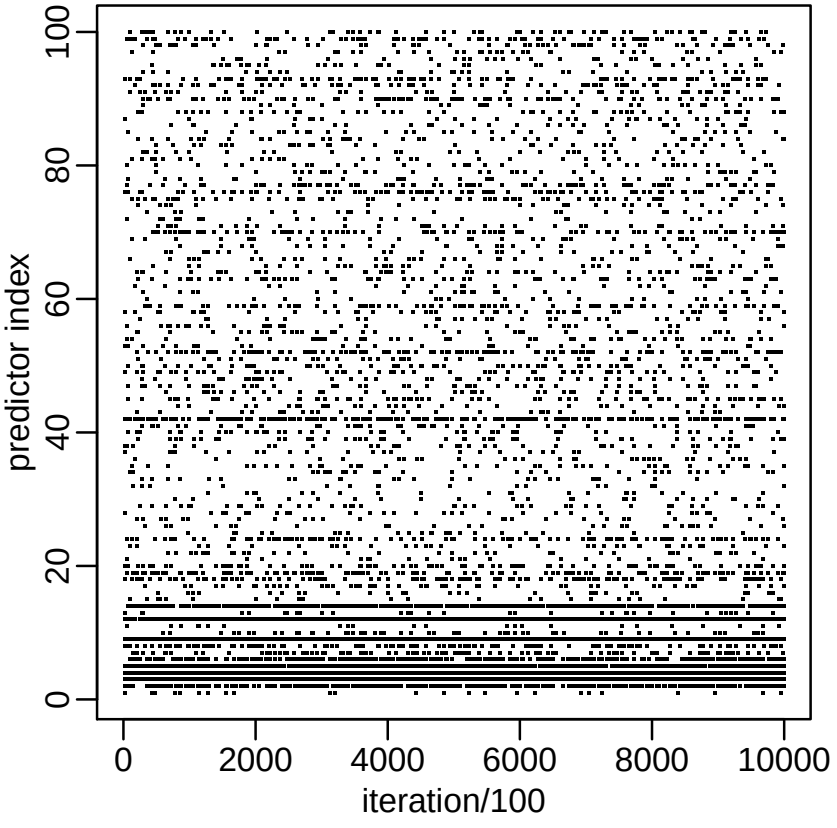
$$
\begin{aligned}
z_1^{(s+1)} &\sim p(z_1 | z_2^{(s)}, \ldots, z_p^{(s)}, \mathbf{y}, \mathbf{X}) \\
z_2^{(s+1)} &\sim p(z_2 | z_1^{(s+1)}, z_3^{(s)}, \ldots, z_p^{(s)}, \mathbf{y}, \mathbf{X}) \\
&\vdots \\
z_p^{(s+1)} &\sim p(z_p | z_1^{(s+1)}, \ldots, z_{p-1}^{(s+1)}, \mathbf{y}, \mathbf{X})
\end{aligned}
$$

This generates $\mathbf{z}^{(s+1)}$ from $\mathbf{z}^{(s)}$.

Repeating this generates $z^{(1)}, \ldots, z^{(S)}$ with which to approximate $p(\mathbf{z} | \mathbf{y}, \mathbf{X})$.

Diabetes example





Marginal inference

What is the estimate of $\boldsymbol{\beta}$?

Recall

$$\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p) = (b_1 z_1, \ldots, b_p, z_p)$$

Our Monte Carlo samples are

$$\begin{array}{lcccccc}
\boldsymbol{\beta}^{(1)} & = & (0 & -.299 & 0 & .427 & \cdots & .845) \\
\boldsymbol{\beta}^{(2)} & = & (0 & -.235 & .834 & .374 & \cdots & 0) \\
\vdots & & & & \vdots & & & \\
\boldsymbol{\beta}^{(S)} & = & (0 & -.315 & 0 & .536 & \cdots & 0)
\end{array}$$

A posterior mean for $\boldsymbol{\beta}$ is obtained in the usual way:

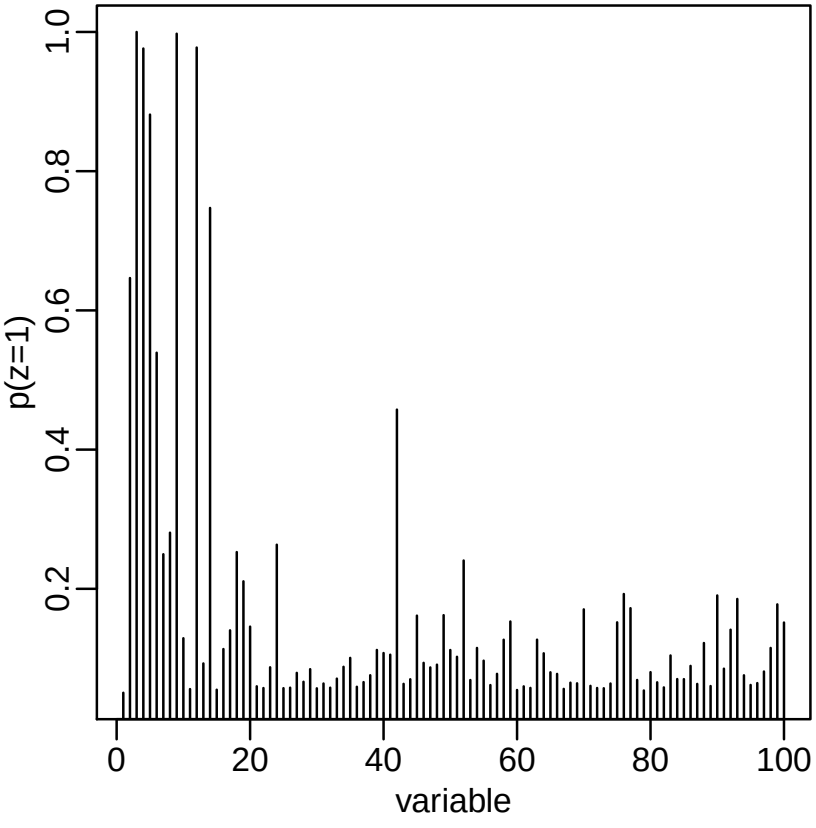
$$\hat{\boldsymbol{\beta}}^{\text{bayes}} = \frac{1}{S}\sum \boldsymbol{\beta}^{(s)} \approx \mathrm{E}[\boldsymbol{\beta}|\mathbf{y}, \mathbf{X}]$$

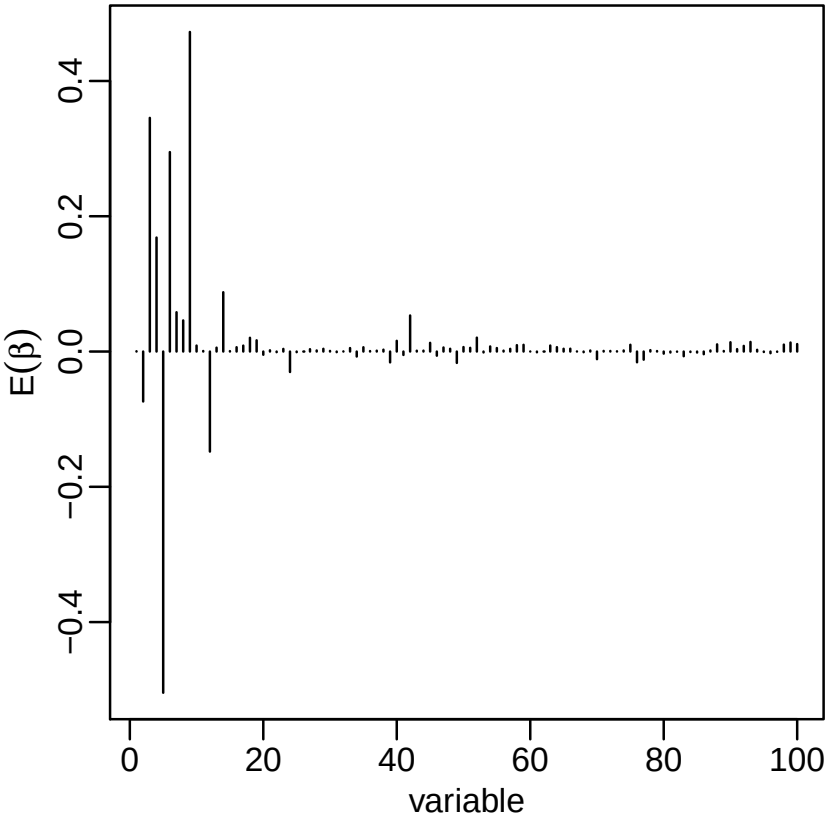
Out of sample predictions can be made with $\hat{\boldsymbol{\beta}}_{\text{bayes}}$:

$$\hat{y}^{bayes}_{\text{test,i}} = \hat{\boldsymbol{\beta}}^{T}_{\text{bayes}} \mathbf{x}_{\text{test,i}}$$

Out of sample prediction error: $\frac{1}{S}\sum (y_{\text{test,i}} - \hat{y}^{bayes}_{\text{test,i}})^2 = 0.4852529$

Marginal inference





Important variables

```
colnames(X)[ order(z.pmean,decreasing=TRUE)[1:10] ]

##  [1] "bmi"     "ltg"     "g2"      "map"     "tc"      "sex.age" "sex"
##  [8] "ldl"     "ltg.age" "tch"

colnames(X)[ order(b.pmean,decreasing=TRUE)[1:10] ]

##  [1] "ltg"     "bmi"     "ldl"     "map"     "sex.age" "hdl"     "ltg.age"
##  [8] "tch"     "glu.bmi" "map.sex"
```

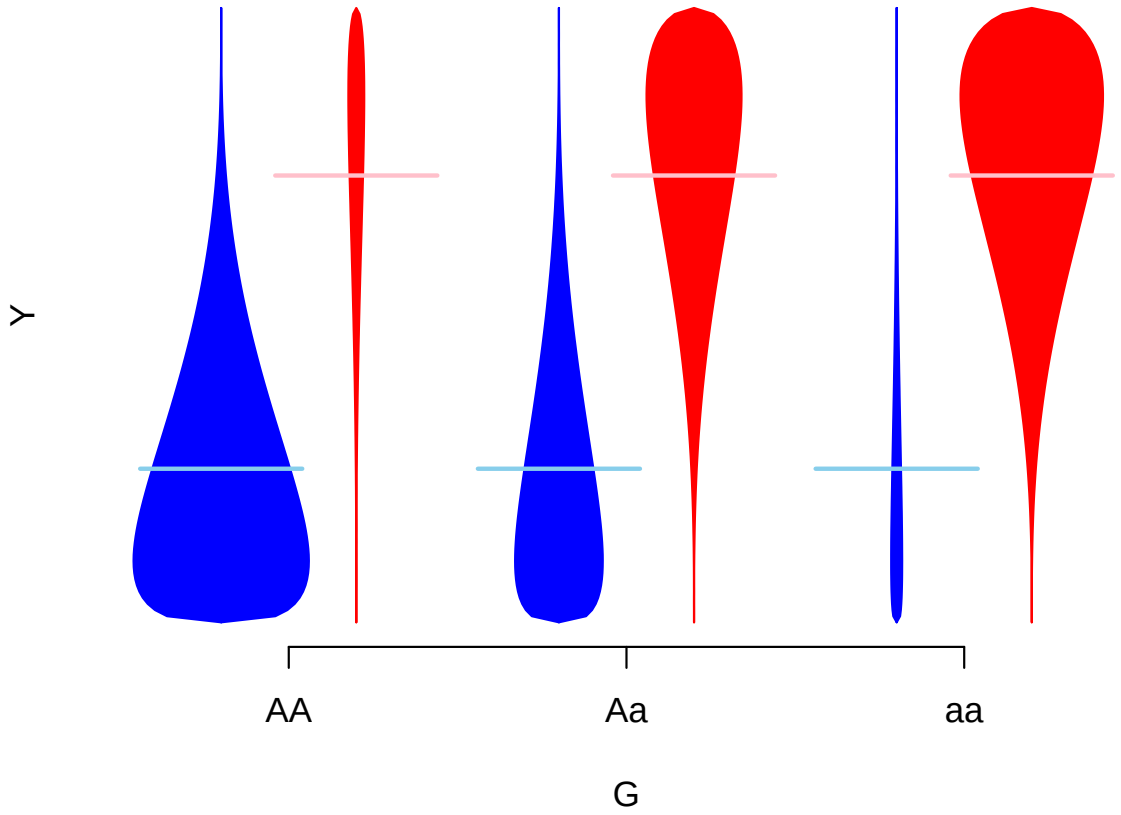
Other approaches, briefly

Model-averaging in this way gives an honest statement of uncertainty. But;

- Not all variables are in the model for the same reason – may want to 'force' some covariates into the model
- When selecting a single, parsimonious model, may want to maximize its ability to predict – not its probability of being true

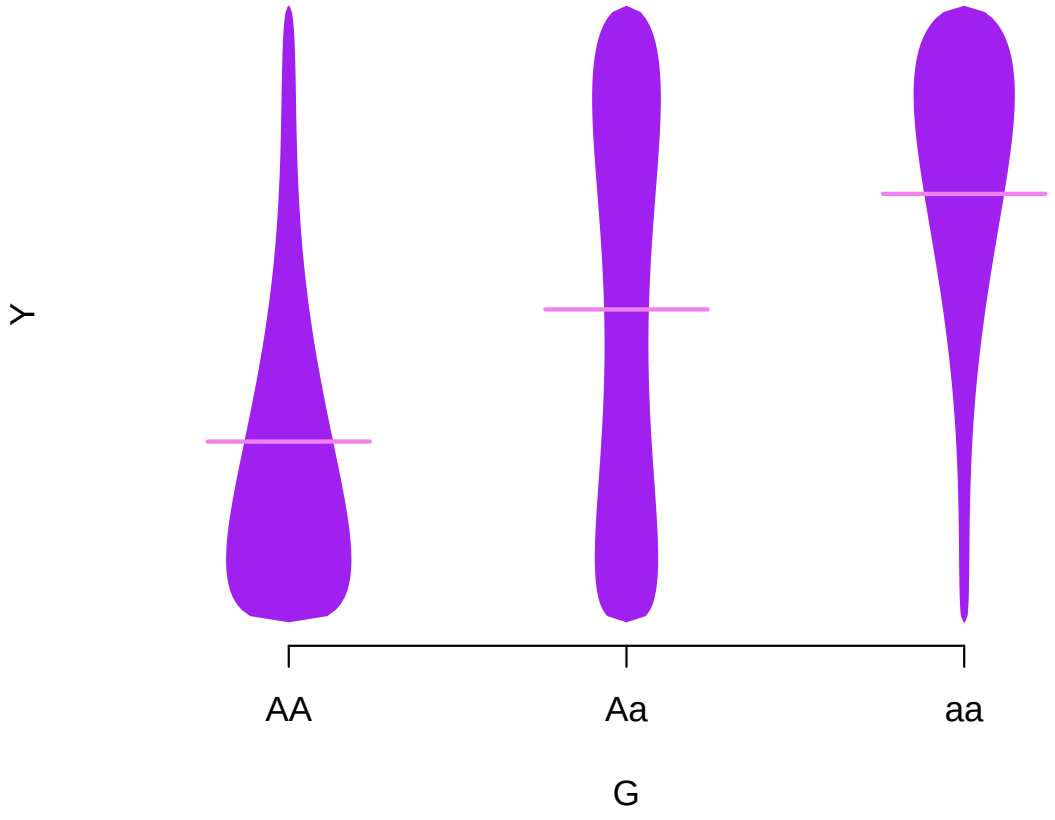
Confounding

'Confounding' means not being able to distinguish between a signal of interest, and some other cause. Here's a genetic 'signal';



Confounding

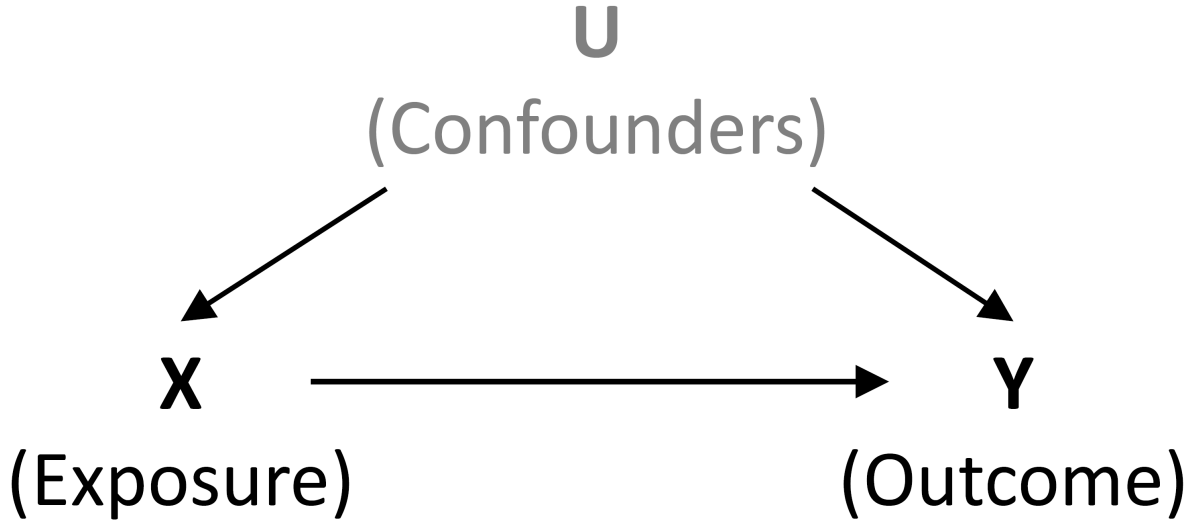
...which can be explained by ancestry, i.e. is confounded by ancestry



However, analysis that adjusts for ancestry would be of interest – even if models without it are better-supported.

Confounding

Directed Acyclic Graphs (DAGs) are a general language for confounding;



Arrows indicate causal relationships; confounding means 'backdoor paths' exist; these can be removed by adjustment for confounders. In genetic association work, typically ancestry is the only plausible confounder - expression and methlyation work is more complex.

Confounding

Bayesian Adjustment for Confounding (BAC, Wang et al 2012) specifies a model with

1. Dependence of outcome on the exposure and the set of confounders
2. Dependence of exposure on the set of confounders
3. Dependence between these models, making variable inclusion in (1) more likely if it is included in (2)

So BAC fits two set of **z** indicators, and links them. Modeling exposures is unusual – doing it well takes careful work.

The method is implemented in BEAU, a stand-alone R package, using approximate calculations for the posterior.

Prediction

Understanding causes (and confounding) is often very important – but ability to predict can matter too;

- Remaining lifetime
- Drug response
- Telling 'good' genotyping from 'bad'

To pick a model here, it's reasonable to ask how well it would predict *in similarly-collected data*. This choice may not be the same as asking what the causes are, e.g. TV ownership rates predict child mortality but are not a cause.

Cross-validation

A natural way to assess how well a fitted model predicts is to fit it, and predict!

SSR is a common measure of predictive accuracy

Cross-validation

1. *SSR* (squared error loss) is not the only option – need to consider the *loss* (*utility*) of particular predictions
2. For categorical outcomes, could also weight misclassification rates (e.g. $P(1|0)$ and $P(0|1)$) – some mistakes may be worse than others
3. Trickier still for dependent outcomes
4. 10-fold cross-validation is typical
5. Fitting multiple models with Gibbs sampling, and cross-validating each can be too slow

Approximate prediction measures

The standard 'score' is log *posterior predictive density*

$$\log p_{\text{ppost}}(y) = \log \int p(y|\theta)p(\theta|y)_{\text{obs}}d\theta).$$

Expected out-of-sample accuracy (over new datasets $\tilde{y}$) is defined as

$$elpd = E(\log p_{\text{ppost}}(\tilde{y})) = \int \log p_{\text{ppost}}(\tilde{y})q(\tilde{y})d\tilde{y}$$

for true density $q(\tilde{y})$. A natural way to estimate this is through the 'in sample accuracy',

$$lpd = \log \int p(y_{\text{obs}}|\theta)p(\theta|y)_{\text{obs}}d\theta,$$

but its double-use of the posterior leads to bias – worse with more parameters.

Approximate prediction measures

- Akaike's Information Criterion (AIC) approximates *lpd* by $\log p(y_{\text{obs}}|\hat{\theta}_{MLE})$ – so is not Bayesian, and adds bias-correction k, the number of parameters
- Deviance Information Criterion (DIC) approximates *lpd* by $\log p(y_{\text{obs}}|E(\theta|y_{\text{obs}}))$ and adds the *effective number of parameters*,

$$p_D = 2(\log p(y_{\text{obs}}|E(\theta|y_{\text{obs}})) - E_\theta[\log p(y_{\text{obs}}|\theta)])$$

For either, in large samples – and under some conditions – choosing the model with the lowest value is equivalent to doing cross-validation.

NB Several other versions are available; AIC, DIC2, WAIC...

DIC examples

- Shriner and Yi 2009 use DIC in the context of multiple QTL Mapping – to select how many QTLs there are, and their locations
- Yu et al, 2012 use DIC studying gene×environment interactions, with a model that 'clusters' nearby* variants, so they have similar interaction effects. DIC is used to choose how many clusters

* ...using the Potts model