Model Selection

Peter Hoff
Duke STA 610

The model selection problem

Test-based selection

Consistent model selection

Modeling choices

Model: A *statistical model* is a set of probability distributions for your data.

- In HLM, the model is a specification of fixed effects and random effects.
- Once we select a model, we can estimate the parameters in the model and make further inference.

```
nels[1:5,]

##   school enroll flp public urbanicity hwh   ses mscore
## 1   1011      5   3      1      urban   2 -0.23  52.11
## 2   1011      5   3      1      urban   0  0.69  57.65
## 3   1011      5   3      1      urban   4 -0.68  66.44
## 4   1011      5   3      1      urban   5 -0.89  44.68
## 5   1011      5   3      1      urban   3 -1.28  40.57
```

What kinds of effects could we include?

- fixed effects: `enroll,flp,public,urbanicity,hwh,ses`
- random effects: `1,hwh,ses`
- fixed effect interactions: `enroll*flp, public*flp`,. . .
- random effect interactions: `hwh*ses`
- higher order terms: $\texttt{ses}^2$,. . .

Model selection

We would like a procedure that can identify the "best" model from the data.

- "best=true" if the truth is one of the potential models.
- "best" means giving the best prediction or description otherwise.

Setup: Let $M_1, M_2, \dots, M_K$ be candidate models. For example, maybe

- M_1: `y ~ flp`
- M_2: `y ~ flp + ses`
- M_3: `y ~ flp + ses + (ses|school)`

Model selection procedure: A procedure that takes data $(\mathbf{y}, \mathbf{X})$ as input and outputs a model.

$$\mathtt{msel}(\mathbf{y}, \mathbf{X}) \in \{M_1, \dots, M_K\}$$

Consistent model selection

As our data are subject to sampling variability, we can't expect a model selection procedure to select the best model with probability 1. However, we do expect that

$$\Pr(\texttt{msel}(\mathbf{y}, \mathbf{X}) = M_k) \text{ is large if } M_k \text{ is correct.}$$

As more data comes in, a good procedure should have an increasingly large chance of selecting the right model. Such a procedure is *consistent*.

Consistency: $\texttt{msel}(\mathbf{y}, \mathbf{X}))$ is consistent if

$$\text{when } M_k \text{ is true, then } \Pr(\texttt{msel}(\mathbf{y}, \mathbf{X}) = M_k) \to 1 \text{ as } n, m \to \infty.$$

Unfortunately, model selection based on p-values is *not consistent*.

Backwards elimination

Diabetes example:

- 442 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!

Backwards elimination

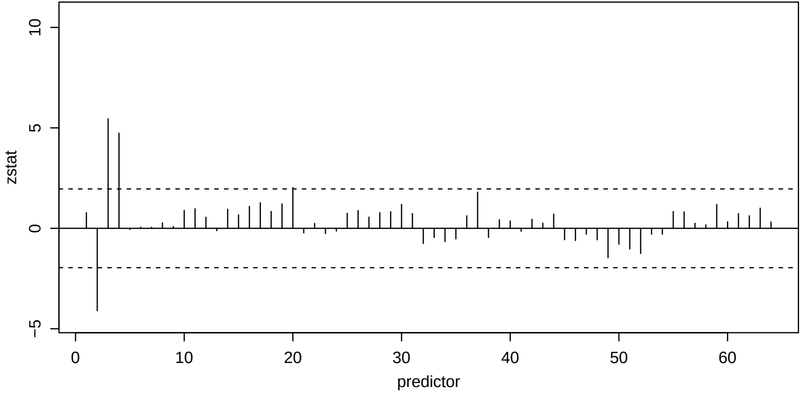
1. Obtain the estimator $\hat{\boldsymbol{\beta}}_{ols} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ and its t-statistics.
2. If there are any regressors j such that $|t_j| < t_{cutoff}$,
   - 2.1 find the regressor j_{min} having the smallest value of $|t_j|$;
   - 2.2 remove column j_{min}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

Backwards elimination

```
### backwards elimination
ZSTATS<-NULL ; zmin<-0 ; zcut<-qnorm(.975)
while(zmin< zcut)
{
  fit<-lm(y~ -1+XS)
  zscore<-summary(fit)$coef[,3]

  zmin<-min(abs(zscore))
  if(zmin<zcut)
  {
    jmin<-which.min(abs(zscore))
    XS<-XS[,-jmin]
  }

  zs<-rep(0,ncol(X))
  zs[ match(substr(names(zscore),3,9),colnames(X)) ] <-zscore
  ZSTATS<-rbind(ZSTATS,zs)
}
###
```

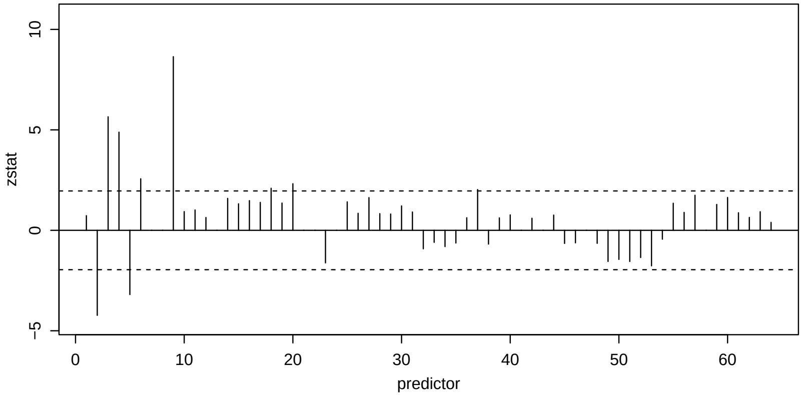
Backwards elimination

Initial *z*-scores:



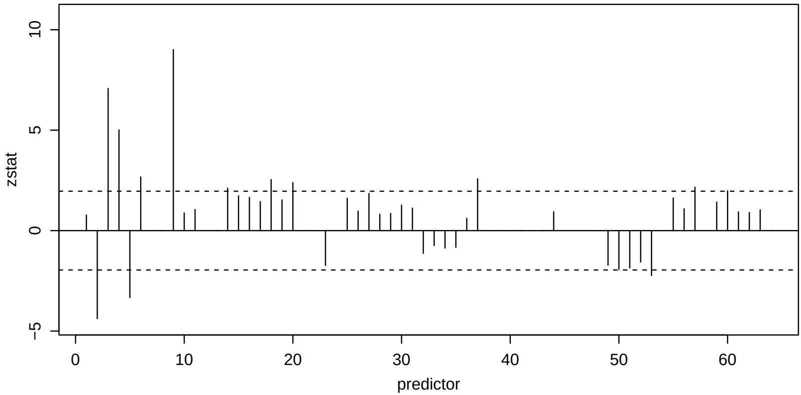
Backwards elimination

After ten iterations:



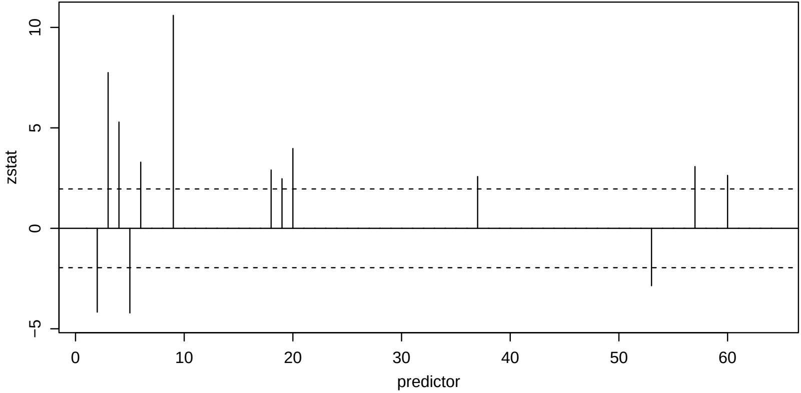
Backwards elimination

After twenty iterations:



Backwards elimination

Final solution:



Final solution

```
summary(fit)

##
## Call:
## lm(formula = y ~ -1 + XS)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.05779 -0.49533 -0.02017  0.40202  1.86086
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## XSsex     -0.15026    0.03603  -4.171 3.67e-05 ***
## XSbmi      0.30789    0.03972   7.752 6.62e-14 ***
## XSmap      0.19982    0.03777   5.290 1.95e-07 ***
## XStc      -0.44478    0.10561  -4.211 3.09e-05 ***
## XSldl      0.32683    0.09924   3.293  0.00107 **
## XSltg      0.57384    0.05415  10.598  < 2e-16 ***
## XSltg^2    0.30735    0.10591   2.902  0.00390 **
## XSglu^2    0.08227    0.03332   2.469  0.01393 *
## XSage:sex  0.13101    0.03297   3.974 8.29e-05 ***
## XSbmi:map  0.08699    0.03373   2.579  0.01024 *
## XStc:ltg  -0.45086    0.15781  -2.857  0.00448 **
## XSldl:ltg  0.37997    0.12363   3.073  0.00225 **
## XShdl:ltg  0.16663    0.06323   2.635  0.00871 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6752 on 429 degrees of freedom
## Multiple R-squared:  0.5565, Adjusted R-squared:  0.5431
## F-statistic: 41.41 on 13 and 429 DF,  p-value: < 2.2e-16
```

How would you interpret the *p*-values, standard errors, CIs?

A problem with backwards selection

Let $\mathbf{y}_\pi$ be a permutation of $\mathbf{y}$, eg.

$$\begin{aligned} \mathbf{y} &= (2.2, -1.2, 0.5, \ldots, -0.7) \\ \mathbf{y}_\pi &= (0.5, -0.7, 2.2, \ldots, -1.2) \end{aligned}$$

Question: What is the relationship between $\mathbf{y}_\pi$ and $\mathbf{X}$?

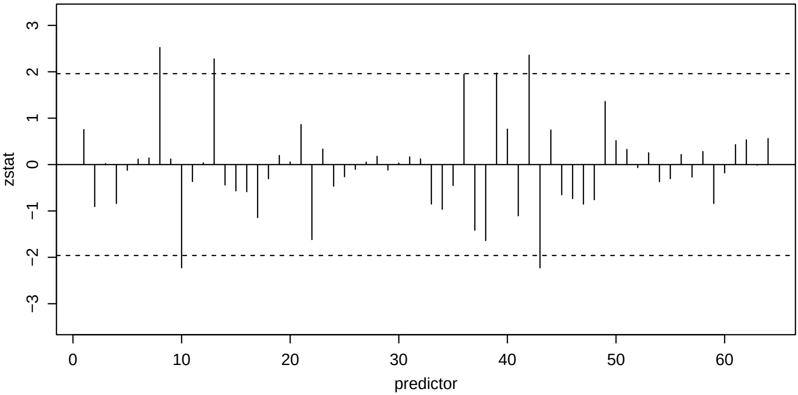
Question: What would happen if we did backwards elimination on $\mathbf{y}_\pi \sim \mathbf{X}$?

Backwards elimination on permuted data

```
yp<-sample(y)
XS<-X

### backwards elimination
ZSTATS<-NULL ; zmin<-0 ; zcut<-qnorm(.975)
while(zmin< zcut)
{
  fit<-lm(yp˜ -1+XS)
  zscore<-summary(fit)$coef[,3]

  zmin<-min(abs(zscore))
  if(zmin<zcut)
  {
    jmin<-which.min(abs(zscore))
    XS<-XS[,-jmin]
  }

  zs<-rep(0,ncol(X))
  zs[ match(substr(names(zscore),3,9),colnames(X)) ] <-zscore
  ZSTATS<-rbind(ZSTATS,zs)
}
###
```

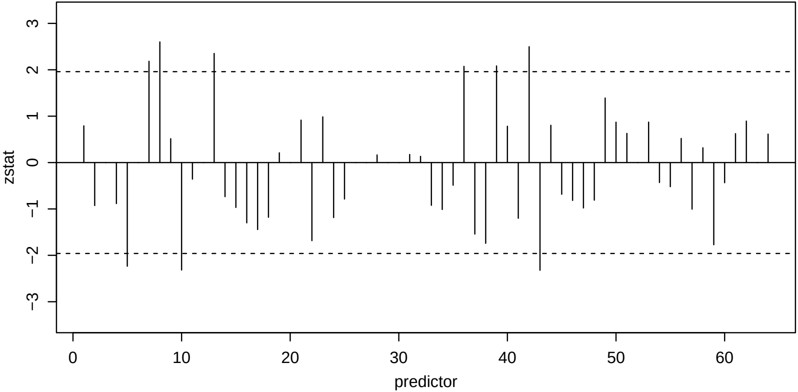
Backwards elimination

Initial z-scores:



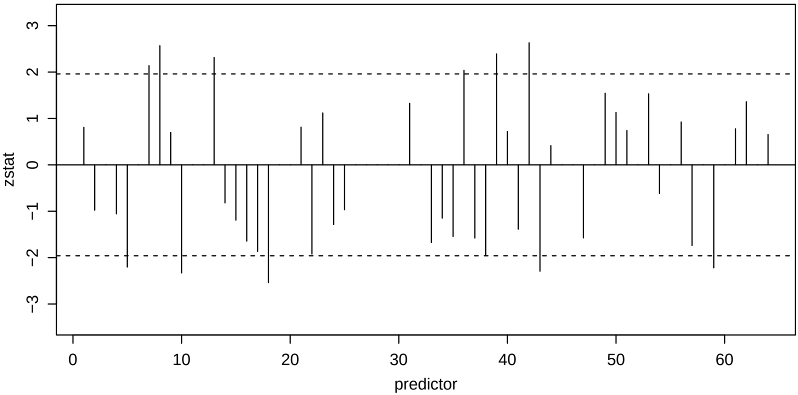
Backwards elimination

After 10 iterations:



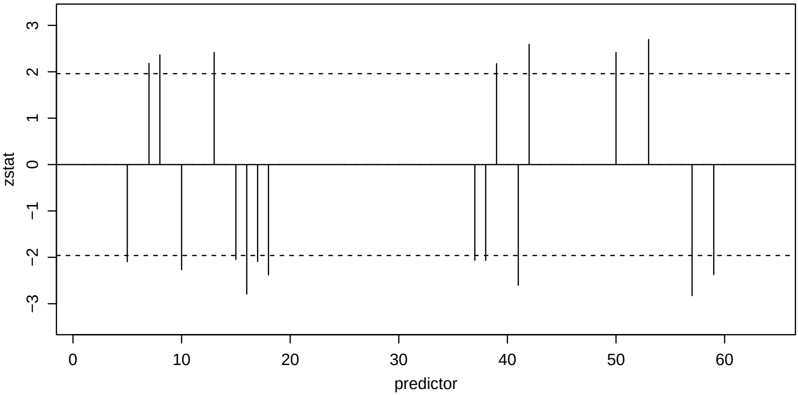
Backwards elimination

After twenty iterations:



Backwards elimination

Final solution:



Final solution

```
summary(fit)

##
## Call:
## lm(formula = yp ~ -1 + XS)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.8058 -0.7964 -0.1466  0.6645  2.4560
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## XStc      -0.28628    0.13675  -2.094  0.03690 *
## XShdl      0.43316    0.19864   2.181  0.02976 *
## XStch      0.53841    0.22773   2.364  0.01852 *
## XSglu     -0.12160    0.05366  -2.266  0.02395 *
## XSmap^2    0.12926    0.05345   2.418  0.01601 *
## XSldl^2   -0.58442    0.28590  -2.044  0.04156 *
## XShdl^2   -0.41785    0.14968  -2.792  0.00548 **
## XStch^2   -0.35026    0.16769  -2.089  0.03732 *
## XSltg^2   -0.24849    0.10444  -2.379  0.01779 *
## XSbmi:map -0.12095    0.05857  -2.065  0.03953 *
## XSbmi:tc  -0.44804    0.21700  -2.065  0.03956 *
## XSbmi:ldl  0.53181    0.24448   2.175  0.03016 *
## XSbmi:tch -0.33768    0.12969  -2.604  0.00954 **
## XSbmi:ltg  0.33771    0.13029   2.592  0.00987 **
## XStc:ldl   0.76928    0.31857   2.415  0.01617 *
## XStc:ltg   0.41443    0.15371   2.696  0.00729 **
## XSldl:ltg -0.43629    0.15446  -2.825  0.00496 **
## XShdl:tch -0.58784    0.24778  -2.372  0.01812 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9799 on 424 degrees of freedom
```

Inconsistency of backwards elimination

Backwards elimination (and forwards selection) generally rely on a comparison of models based on a p-value.

M_1: `y ~ x1 + x2 + x3`

M_0: `y ~ x1 + x2`

Variable `x3` is eliminated if

- its z-score is < 1.96 in absolute value
- (more or less) equivalently, if the p-value from the LRT is > 0.05.

Inconsistency of backwards elimination

Now suppose M_0 is true. What is the probability of selecting M_1?

$$\begin{aligned}\Pr(\texttt{bsel}(\mathbf{y},\mathbf{X}) = M_1 | M_0) = & \ \Pr(\text{reject } M_0|M_0) \\ &= \text{type I error rate} \\ &= \Pr(p - \text{value} > 0.05|M_0) = 0.05\end{aligned}$$

This does not change as $m, n \to \infty$.

(Actually, for the LRT the probability gets closer to 0.05 as $m, n \to \infty$).

Problems with backwards elimination

There are other problems with backwards elimination (and forwards selection):

Problem 1: The method doesn't search over all possible models.

Problem 2: The resulting *p*-values and standard errors may be misleading.

Problem 3: The model selection procedure is *not consistent*

Problems 1-2 are issues for any model selection procedure.

However, some model selection procedures do not have problem 3.

Building a better model selection procedure

Suppose only two models are under consideration, M_0 and M_1.

Maximize the likelihoods under each model:

$$l_1 = \log p(\mathbf{y}|\hat{\theta}_1)$$
$$l_0 = \log p(\mathbf{y}|\hat{\theta}_0)$$

If l_1 is much bigger than l_0, then it makes sense to prefer M_1 to M_0.

However, recall that if

- M_0 is nested in M_1, or
- M_0 has many fewer parameters than M_1,

then l_1 will always/typically be larger than l_0.

Building a better model selection procedure

Idea: Prefer M_1 to M_0 if

- l_1 is bigger than l_0 by an amount that depends on p_0, p_1.
- $l_1 - l_0 > c_{p_0,p_1}$

This should remind you of the LRT, where we prefer M_1 to M_0 if

$$\lambda = 2 \times (l_1 - l_0) > q_{p_0,p_1},$$

where $q_{p_0,,p_1}$ is a quantile of the appropriate null distribution.

Exercise: Show that the LRT procedure has the above form.

LRT as a model selection procedure

LRT: Reject M_0, favor M_1 if

$$\begin{aligned} \lambda = 2 \times (l_1 - l_0) &> \chi^2_{p_1 - p_0, .95} \\ l_1 - l_0 &> \tfrac{1}{2}\chi^2_{p_1 - p_0, .95} = c_{p_1, p_0} \end{aligned}$$

Problem: If M_0 is true, probability of selecting M_1 is ≈ 0.05, regardless of m, n.

Model selection via hypotheses test is *not consistent*.

Modified selection criteria

Consider *any* procedure that prefers M_1 to M_0 if

$$l_1 - l_0 > c_{p_0,p_1},$$

where c_{p_0,p_1} is constant in m, n.

Any such procedure corresponds to a LRT for some particular type I error rate, and hence will not be consistent.

Solution: Have the cutoff c depend on m, n - favor M_1 over M_0 if

$$l_1 - l_0 > c_{p_0,p_1,m,n}$$

Modified selection criteria

Question: How should c change with $N = m \times n$? Go up, or go down?

Answer:

- The inconsistency comes from rejecting M_0 too often.
- The threshold for favoring M_1 over M_0 should go up.
- We will still be able to select M_1 correctly if M_1 is true - as N increases our ability to distinguish M_1 from M_0 increases as well.

Selection criteria: Favor M_1 over M_0 if

$$l_1 - l_0 > c_{p_0,p_1,m,n},$$

where $c_{p_0,p_1,m,n}$ is increasing in m, n.

BIC - Bayes information criteria

$$
\begin{aligned}
b_0 &= l_0 - \tfrac{1}{2} p_0 \log N \\
b_1 &= l_1 - \tfrac{1}{2} p_1 \log N
\end{aligned}
$$

Model selection via BIC: Favor M_1 over M_0 if $b_1 > b_0$.

Exercise: Rewrite this procedure to have the form used previously.

$$b_1 > b_0 \Leftrightarrow l_1 - l_0 > \tfrac{1}{2}\big((p_1 - p_0) \times \log N\big)$$

Notice: The cutoff

- is increasing in $p_1 - p_0$,
- is increasing in $N = m \times n$.

BIC - standard form

$$BIC_0 = -2 \times l_0 + p_0 \log N$$
$$BIC_1 = -2 \times l_1 + p_1 \log N$$

Model selection via BIC: Favor M_1 over M_0 if $BIC_1 < BIC_0$.

This is the same as favoring M_1 over M_0 if $b_1 < b_0$:

$$BIC_0 = -2 \times b_0$$
$$BIC_1 = -2 \times b_1$$

Do we trust BIC?

$$y_{i,j} = \beta_1 + \beta_2 x_{i,j} + a_{1,j} + \epsilon_{i,j}$$
$$a_{1,1}, \ldots, a_{1,m} \sim \text{ i.i.d. } N(0, \tau^2)$$

Consider selecting from among the following four models:

M_{00}**:** $\beta_2 = 0$, $\tau^2 = 0$

M_{10}**:** $\beta_2 \neq 0$, $\tau^2 = 0$

M_{01}**:** $\beta_2 = 0$, $\tau^2 \neq 0$

M_{11}**:** $\beta_2 \neq 0$, $\tau^2 \neq 0$

Question: What are the number of parameters in each model?

M_{11} $p = 4$

M_{01} $p = 3$

M_{10} $p = 3$

M_{00} $p = 2$

Comment: Which models could be compared with LRT?

Simulation study

```
m<-50 ; n<-5 ; g<-rep(1:m,times=rep(n,m))

BIC.RES<-NULL

for(t2 in c(0,1)){
for(beta2 in c(0,1)) {

  BIC.SIM<-NULL
  for(s in 1:100)
  {
    b<-rnorm(m,0,sqrt(t2) )
    x<-rnorm(m*n)

    y<- 1 + beta2*x + b[g] + rnorm(m*n)

    fit.00<-lm(y~1)
    fit.01<-lm(y~x)

    fit.10<-lmer(y ~ 1 + (1|g), REML=FALSE )
    fit.11<-lmer(y ~ x + (1|g), REML=FALSE )

    BIC.SIM<-rbind(BIC.SIM,c(BIC(fit.00),BIC(fit.01),BIC(fit.10),BIC(fit.11)))
  }

  BIC.RES<-rbind(BIC.RES,(table( c(1:4,apply(BIC.SIM,1,which.min))  ) -1))
}}
```

Simulation study

```
BIC.RES
```

```
##       1   2   3   4
## [1,] 99   0   1   0
## [2,]  0 100   0   0
## [3,]  0   0 100   0
## [4,]  0   0   0 100
```

A harder simulation study

```
m<-10 ; n<-5 ; g<-rep(1:m,times=rep(n,m))

BIC.RES<-NULL

for(t2 in c(0,.5)){
for(beta2 in c(0,.5)) {

  BIC.SIM<-NULL
  for(s in 1:100)
  {
    b<-rnorm(m,0,sqrt(t2) )
    x<-rnorm(m*n)

    y<- 1 + beta2*x + b[g] + rnorm(m*n)

    fit.00<-lm(y~1)
    fit.01<-lm(y~x)

    fit.10<-lmer(y ~ 1 + (1|g), REML=FALSE )
    fit.11<-lmer(y ~ x + (1|g), REML=FALSE )

    BIC.SIM<-rbind(BIC.SIM,c(BIC(fit.00),BIC(fit.01),BIC(fit.10),BIC(fit.11)))
  }

  BIC.RES<-rbind(BIC.RES,(table( c(1:4,apply(BIC.SIM,1,which.min))  ) -1))
}}
```

Simulation study

```
BIC.RES

##       1  2  3  4
## [1,] 92  7  1  0
## [2,]  6 93  0  1
## [3,] 30  1 66  3
## [4,]  5 28  5 62
```

Model selection for NELS data

```
fit.full<-lmer( mscore ~
  as.factor(flp) + as.factor(urbanicity) + public +
  ses + ses:public + (ses|school) , data=nels,REML=FALSE)

summary(fit.full)$coef

##                                  Estimate Std. Error      t value
## (Intercept)                   53.72704978  0.4672579 114.98371763
## as.factor(flp)2               -1.73548708  0.4026467  -4.31019849
## as.factor(flp)3               -4.45001943  0.4379125 -10.16189084
## as.factor(urbanicity)suburban -0.02067462  0.3833574  -0.05393039
## as.factor(urbanicity)urban    -0.94654261  0.4193025  -2.25742178
## public                        -0.84372430  0.4425283  -1.90659944
## ses                            3.41745532  0.2586162  13.21438763
## public:ses                     0.90865289  0.2946272   3.08407716
```

Model selection for NELS data

```
BIC(fit.full)

## [1] 92472.76
```

```
fit.r1<-lmer( mscore ~
  as.factor(flp) + as.factor(urbanicity) + public +
  ses + (ses|school) , data=nels,REML=FALSE)

BIC(fit.r1)

## [1] 92472.71
```

```
fit.r2<-lmer( mscore ~
  as.factor(flp) + as.factor(urbanicity) +
  ses + (ses|school) , data=nels,REML=FALSE)

BIC(fit.r2)

## [1] 92464.98
```

Futher reductions

```
fit.r3<-lmer(mscore~ as.factor(flp) + ses + (ses|school) , data=nels,REML=FALSE)

BIC(fit.r3)

## [1] 92454.31
```

Futher reductions

```
fit.r4a<-lm( mscore ~ as.factor(flp) + ses , data=nels)
BIC(fit.r4a)

## [1] 93151.9
```

```
fit.r4b<-lmer( mscore ~ ses + (ses|school) , data=nels,REML=FALSE)
BIC(fit.r4b)

## [1] 92597.89
```

```
fit.r4c<-lmer( mscore ~ (ses|school) , data=nels,REML=FALSE)
BIC(fit.r4c)

## [1] 93267.56
```

Where does BIC come from?

Suppose there are only two models M_0 and M_1.

In a Bayesian analysis, one would be able to compute

$$\Pr(M_1|\mathbf{y}) = \frac{\Pr(M_1)p(\mathbf{y}|M_1)}{\Pr(M_1)p(\mathbf{y}|M_1) + \Pr(M_0)p(\mathbf{y}|M_0)}$$

Alternatively, the odds that M_1 is true are

$$\frac{\Pr(M_1|\mathbf{y},\mathbf{X})}{\Pr(M_0|\mathbf{y},\mathbf{X})} = \frac{\Pr(M_1)}{\Pr(M_0)} \times \frac{p(\mathbf{y}|M_1)}{p(\mathbf{y}|M_0)}$$

If $Pr(M_1) = \Pr(M_0)$, then

$$\frac{\Pr(M_1|\mathbf{y},\mathbf{X})}{\Pr(M_0|\mathbf{y},\mathbf{X})} = \frac{p(\mathbf{y}|M_1)}{p(\mathbf{y}|M_0)}$$

Where does BIC come from?

We would select M_1 if $\frac{p(\mathbf{y}|M_1)}{p(\mathbf{y}|M_0)} > 1$, or equivalently

$$\log \frac{p(\mathbf{y}|M_1)}{p(\mathbf{y}|M_0)} = \log p(\mathbf{y}|M_1) > p(\mathbf{y}|M_0).$$

It can be shown that in many cases for large N,

$$\begin{aligned} \log p(\mathbf{y}|M_1) &\approx \log p(\mathbf{y}|\hat{\theta}_1) - \tfrac{1}{2} p_1 \log N \\ \log p(\mathbf{y}|M_0) &\approx \log p(\mathbf{y}|\hat{\theta}_0) - \tfrac{1}{2} p_0 \log N \end{aligned}$$

and so we prefer M_1 to M_0 if

$$\begin{gathered} \log p(\mathbf{y}|\hat{\theta}_1) - \tfrac{1}{2} p_1 \log N > \log p(\mathbf{y}|\hat{\theta}_0) - \tfrac{1}{2} p_0 \log N \\ -2 \log p(\mathbf{y}|\hat{\theta}_1) + p_1 \log N < -2 \log p(\mathbf{y}|\hat{\theta}_0) + p_0 \log N \\ BIC(M_1) < BIC(M_0) \end{gathered}$$

Comments

Other information criteria: AIC, TIC, GIC.
See Müller, Sealy and Welsh (2013) for a review.

Don't do the following:

- $BIC(M_1) = 100$, but has many parameters;
- $BIC(M_0) = 101$, but has few parameters.

"Since the BICs are close, and M_1 has more parameters, I'll go with M_0."

M_1 has *already* been penalized for its number of parameters.
The BIC selection rule would be to select M_1.