Model Selection

ISLR Chapter 6, GH 6 Chapter 24

September 30, 2019

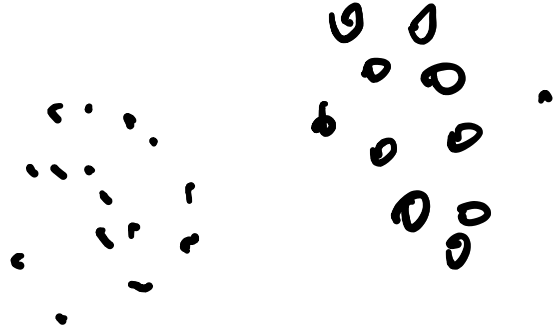
Voting model with interactions and a subset of predictors

```r
# see code for variable coding
nes1992 = dplyr::select(nes1992, race,  black,
                        gender, educ, income, partyid,
                        ideo, vote)
vote.glm = glm(vote ~ (. -race)^2, data=nes1992,
               family="binomial")
```

Output

	Estimate	Std. Error
(Intercept)	-9.0E+14	2.2E+07
blackTRUE	-1.0E+15	2.4E+07
genderfemale	8.7E+14	2.1E+07
educhigh school graduate	-2.5E+15	2.5E+07
educsome college	-1.7E+15	2.7E+07
educcollege graduate	-1.9E+15	3.7E+07
educmissing	-1.7E+15	6.6E+07
income2	-2.1E+15	2.4E+07
income3	-2.0E+15	3.3E+07
income4	-3.0E+15	7.3E+07
income5	9.3E+14	8.0E+07
incomemissing	-1.0E+15	4.0E+07
partyidindependents	3.6E+15	4.7E+07
partyidrepublicans	7.6E+15	2.7E+07
partyidapolitical	-1.2E+15	1.0E+08
partyidmissing	6.0E+14	1.3E+08

Problems



quasi-
separation

- large coefficients

Quasi-Separation (in Binary Data)

Collinearity

Problems

- large coefficients
- large standard errors! instability

Quasi-Separation (in Binary Data)

Collinearity

Problems

- large coefficients
- large standard errors! instability
- very small p-values

Quasi-Separation (in Binary Data)

Collinearity

Problems

- large coefficients
- large standard errors! instability
- very small p-values
- lots of NA's

Quasi-Separation (in Binary Data)

Collinearity

Problems

- large coefficients
- large standard errors! instability
- very small p-values
- lots of NA's
- warnings glm.fit: algorithm did not converge

Quasi-Separation (in Binary Data)

Collinearity

Problems

- large coefficients
- large standard errors! instability
- very small p-values
- lots of NA's
- warnings glm.fit: algorithm did not converge
- warnings glm.fit: fitted probabilities numerically 0 or 1 occurred

Quasi-Separation (in Binary Data)

Collinearity

Problems

- large coefficients
- large standard errors! instability
- very small p-values
- lots of NA's
- warnings glm.fit: algorithm did not converge
- warnings glm.fit: fitted probabilities numerically 0 or 1 occurred
- still have over-dispersion

Folk theorem

Quasi-Separation (in Binary Data)

Collinearity

numerical instability $\leftrightarrow$

variance of estimator is

crazy high

Possible Solutions

- Variable Selection: reduce the number of predictors

Distinguish between goals of good predictions and learning the "true" model

Possible Solutions

- Variable Selection: reduce the number of predictors
  - best subset selection of 2^p models (exhaustive enumeration)

Distinguish between goals of good predictions and learning the "true" model

Possible Solutions

$X \in \mathbb{R}^p$

how many subsets of variables

- Variable Selection: reduce the number of predictors
  - best subset selection of 2^p models (exhaustive enumeration)
  - step-wise selection (forward, backwards, step-wise, MCMC)

$Z_j \in \{0, 1\}$ depending on inclusion of variable j

$p = 20$, $2^{20} > 1{,}000{,}000$

1) computation

2) Overfitting

3) interpret/selection

Distinguish between goals of good predictions and learning the "true" model

Possible Solutions

- Variable Selection: reduce the number of predictors
  - best subset selection of 2^p models (exhaustive enumeration)
  - step-wise selection (forward, backwards, step-wise, MCMC)
- Shrinkage: use all predictors, but the coefficients are shrunk towards 0

Distinguish between goals of good predictions and learning the "true" model

Possible Solutions

- Variable Selection: reduce the number of predictors
  - best subset selection of 2^p models (exhaustive enumeration)
  - step-wise selection (forward, backwards, step-wise, MCMC)
- Shrinkage: use all predictors, but the coefficients are shrunk towards 0 shrinkage estimators
  - some shrinkage methods shrink coefficients to zero allowing variable selection (ad hoc)

good prediction but set $\hat{\beta}_j = 0$ for many j

Distinguish between goals of good predictions and learning the "true" model

Possible Solutions

- Variable Selection: reduce the number of predictors
  - best subset selection of 2^p models (exhaustive enumeration)
  - step-wise selection (forward, backwards, step-wise, MCMC)
- Shrinkage: use all predictors, but the coefficients are shrunk towards 0
  - some shrinkage methods shrink coefficients to zero allowing variable selection (ad hoc)
- Shrinkage + variable selection

Distinguish between goals of good predictions and learning the "true" model

Possible Solutions

- Variable Selection: reduce the number of predictors
  - best subset selection of 2^p models (exhaustive enumeration)
  - step-wise selection (forward, backwards, step-wise, MCMC)
- Shrinkage: use all predictors, but the coefficients are shrunk towards 0
  - some shrinkage methods shrink coefficients to zero allowing variable selection (ad hoc)
- Shrinkage + variable selection
- Dimension Reduction: create new variables which are linear combinations of orig.

Distinguish between goals of good predictions and learning the "true" model

$x_i \in \mathbb{R}^p$ $\quad V_i = C x_i$

$v_i \in \mathbb{R}^d$ $\quad d \ll p$

Balancing Goodness of Fit and Model Complexity

Adjusted Deviance: deviance $+$ number of parameters

- adding a variable with a parameter that is zero is expected to decrease the deviance by 1

Balancing Goodness of Fit and Model Complexity

error + complexity

how measure

Adjusted Deviance: deviance + $\underline{\text{number of parameters}}$

- adding a variable with a parameter that is zero is expected to decrease the deviance by 1
- adding k variables (all with zero coefficients) is expected to reduce the deviance by k ($\mathrm{E}[\chi_k^2]$ variable)

Balancing Goodness of Fit and Model Complexity

Adjusted Deviance: deviance + number of parameters

- adding a variable with a parameter that is zero is expected to decrease the deviance by 1
- adding k variables (all with zero coefficients) is expected to reduce the deviance by k ($\mathrm{E}[\chi^2_k]$ variable)
- needs to be greater than 1

Balancing Goodness of Fit and Model Complexity

Adjusted Deviance: deviance $+$ number of parameters

- adding a variable with a parameter that is zero is expected to decrease the deviance by 1
- adding k variables (all with zero coefficients) is expected to reduce the deviance by k ($\mathrm{E}[\chi^2_k]$ variable)
- needs to be greater than 1
- How much bigger to improve predictions?

Akaike Information Criterion

AIC: deviance + 2 (number of parameters) + each predictor needs to reduce the deviance by 2 to improve the fit to new data

- True data generating model $f(y)$

$\mathcal{M}_1$ = all variables $\neq$ $p(y \mid \hat{\theta}, \mathcal{M})$

$\mathcal{M}_2$ = include $1, \ldots, p-1$

$\neq$

assume $\mathcal{M}$, and $\hat{\theta}$ $\rightarrow$

index model

sampling dist of y

$$KL(f, \hat{p}_{\mathcal{M}}) = \int \log\left[\frac{f(y)}{p(y \mid \hat{\theta}, \mathcal{M})}\right] f(y)\, dy$$

Kullback-Leibler or relative entropy

$$= \int \log(f(y)) f(y)\, dy - \int \log(p(y \mid \hat{\theta}, \mathcal{M})) f(y)\, dy$$

$$= C - \underline{\int \log(p(y \mid \hat{\theta}, \mathcal{M})) f(y)\, dy}$$

$\not\equiv$ $\rightarrow$ no dependence on $\theta, \mathcal{M}$

Akaike Information Criterion

AIC: deviance + 2 (number of parameters) + each predictor needs to reduce the deviance by 2 to improve the fit to new data

- True data generating model $f(y)$
- Candidate Model $p(y \mid \theta, \mathcal{M})$; estimate $p(y \mid \hat{\theta}, \mathcal{M})$

$$
\begin{aligned}
KL(f, \hat{p}_{\mathcal{M}}) &= \int \log\left[\frac{f(y)}{p(y \mid \hat{\theta}, \mathcal{M})}\right] f(y)\, dy \\
&= \int \log(f(y)) f(y)\, dy - \int \log(p(y \mid \hat{\theta}, \mathcal{M})) f(y)\, dy \\
&= C - \int \log(p(y \mid \hat{\theta}, \mathcal{M})) f(y)\, dy
\end{aligned}
$$

Akaike Information Criterion

AIC: deviance + 2 (number of parameters) + each predictor needs to reduce the deviance by 2 to improve the fit to new data

- True data generating model $f(y)$
- Candidate Model $p(y \mid \theta, \mathcal{M})$; estimate $p(y \mid \hat{\theta}, \mathcal{M})$
- measure closeness of candidate to truth by Kullback Leibler divergence

$$
\begin{aligned}
KL(f, \hat{p}_{\mathcal{M}}) &= \int \log\left[\frac{f(y)}{p(y \mid \hat{\theta}, \mathcal{M})}\right] f(y)\, dy \\
&= \int \log(f(y)) f(y)\, dy - \int \log(p(y \mid \hat{\theta}, \mathcal{M})) f(y)\, dy \\
&= C - \int \log(p(y \mid \hat{\theta}, \mathcal{M})) f(y)\, dy
\end{aligned}
$$

can you ?

compute !

Estimating

Naive estimate of integral

$$
\begin{aligned}
K(f, \hat{p}_{\mathcal{M}}) &= C - \boxed{\int \log(p(y \mid \hat{\theta}, \mathcal{M})) f(y)\, dy} \\
&\approx C - \boxed{\frac{1}{n} \sum_i \log(p(y_i \mid \hat{\theta}, \mathcal{M})))} \\
&= C - \frac{\ell(\hat{\theta}; \mathcal{M})}{n}
\end{aligned}
$$

← plugin estimg for

$$\lim_{n \to \infty} \frac{1}{n} \sum_i \log(p(y_i \mid \hat{\theta}, \mathcal{M})) = \int \log p(y \mid \ldots) f(y)\, dy$$

Akaike showed that the bias was approximately $\boxed{p_{\mathcal{M}}/n}$

Correcting for bias, minimizing KL divergence is the same as minimizing

$$-\frac{\ell(\hat{\theta}; \mathcal{M})}{n} + \frac{p_{\mathcal{M}}}{n}$$

\# param. in model

or multiplying by $2n$ we get the deviance $+\ 2p_{\mathcal{M}}$

$$-2\ell(\hat{\theta}; \mathcal{M}) + 2p_{\mathcal{M}}$$

Bayes Information Criterion (BIC or Schwarz Criterion)

Consider models $\mathcal{M}_1, \ldots \mathcal{M}_K$

Bayes Theorem: probability of model $\mathcal{M}$

(a)

$$p(\mathcal{M}_j \mid \underbrace{Y_1, \ldots, Y_n}_{\text{data}}) = \frac{p(Y_1, \ldots, Y_n \mid \mathcal{M}_j)p(\mathcal{M}_j)}{\sum_k p(Y_1, \ldots, Y_n \mid \mathcal{M}_k)p(\mathcal{M}_k)}$$

Pick model that has highest posterior probability

What happened to θ?

(a)

$$\begin{aligned} p(Y_1, \ldots, Y_n \mid \mathcal{M}) &= \int p(Y_1, \ldots, Y_n \mid \theta, \mathcal{M})p(\theta \mid \mathcal{M})\, d\theta \\ &= \int \mathcal{L}(\theta)p(\theta \mid \mathcal{M})\, d\theta \end{aligned}$$

Continue

Maximizing $p(\mathcal{M}_j \mid Y_1, \ldots, Y_n)$ is equivalent to picking $\mathcal{M}$ that maximizes

$$\log(p(Y_1, \ldots, Y_n \mid \mathcal{M}_j)) + \log(p(\mathcal{M}_j))$$

marginal likelihood | model j

Taylor's series expansion of likelihood can be used to show this is approximately

$$\approx \ell_{\mathcal{M}_j}(\hat{\theta}) - \frac{p_{\mathcal{M}_j}}{2} \log(n)$$

Multiply by -2 to obtain BIC = deviance + log(n) (number of parameters)

Not necessarily the best predictive model! But the model that is most likely to be true given the data out of the collection of models under consideration.

$\mathcal{M}_1, \ldots, \mathcal{M}_K$

R Packages/Functions

$LOO(M_1)$

$LOO(M_2)$

$\vdots$

$LOO(M_K)$

- `step` (base R, step-wise)

R Packages/Functions

- `step` (base R, step-wise)
- `leaps::regsubsets` exhaustive Leaps & Bounds search AIC, BIC linear models

R Packages/Functions

- `step` (base R, step-wise)
- `leaps::regsubsets` exhaustive Leaps & Bounds search AIC, BIC linear models
- `bestglim::bestglm` GLM's for AIC, BIC, LOOCV, others

R Packages/Functions

step-wise

- `step` (base R, step-wise)
- `leaps::regsubsets` exhaustive Leaps & Bounds search AIC, BIC linear models
- `bestglim::bestglm` GLM's for AIC, BIC, LOOCV, others
- `BAS:bas.lm` or `BAS:bas.glm` AIC, BIC, more with exhaustive and MCMC as well as model averaging

R Packages/Functions

- `step` (base R, step-wise)
- `leaps::regsubsets` exhaustive Leaps & Bounds search AIC, BIC linear models
- `bestglim::bestglm` GLM's for AIC, BIC, LOOCV, others
- `BAS:bas.lm` or `BAS:bas.glm` AIC, BIC, more with exhaustive and MCMC as well as model averaging
- `BMA` samples based on leaps and MCMC

Stepwise

```r
best.step = step(vote.glm, k=2)   # AIC
```

```
Start:  AIC=11197.27
vote ~ ((race + black + gender + educ + income + partyid +
    race)^2

                 Df Deviance     AIC
- educ:income     19    665.8   867.8
- educ:ideo       12    679.8   895.8
- educ:partyid     8    674.6   898.6
- income:ideo     15  10164.3 10374.3
- gender:partyid   2  10164.3 10400.3
- gender:income    5  10308.5 10538.5
<none>                10957.3 11197.3
- partyid:ideo     6  11461.9 11689.9
- black:partyid    2  12110.7 12346.7
- income:partyid  10  12254.8 12474.8
- black:educ       4  12326.9 12558.9
```

Final Model

```r
summary(best.step)
```

```
Call:
glm(formula = vote ~ black + gender + income + partyid + id
    black:income + gender:partyid, family = "binomial", dat

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4090  -0.3516  -0.2055   0.4019   3.3471

Coefficients: (1 not defined because of singularities)
                      Estimate Std. Error z va
(Intercept)           -3.64935    0.42549  -8.
blackTRUE            -17.30639  612.84355  -0.
genderfemale           0.75432    0.31208   2.
income2                0.21476    0.37663   0.
income3                0.07647    0.35021   0.
```

Stepwise

(1) collinearities

- each step pick the lowest IC model

Does not do exhaustive search

Stepwise

- each step pick the lowest IC model
- add/drop until no improvement

greedy search

Does not do exhaustive search

Stepwise

$\delta_1 \to 100$

$\delta_1 + 10^{-8} \to 10$

- each step pick the lowest IC model
- add/drop until no improvement
- output is the final model

what tolerance

Does not do exhaustive search

Stepwise

- each step pick the lowest IC model
- add/drop until no improvement
- output is the final model
- possible that forward, backwards, both lead to different final models.

Does not do exhaustive search

Example with bestglm (exhaustive)

```
library(bestglm)
nes1992sub = dplyr::select(nes1992, -race) %>%
            filter(partyid != "apolitical")
vote.AIC = bestglm(Xy=nes1992sub, family=binomial,
                   IC="AIC", RequireFullEnumerationQ = T)
```

Notes: dataframe limited to variables under consideration with the response last

Best AIC

```
blackTRUE             -2.1791     0.4419  -4.931 8.20e-07 **
partyidindependents    1.5648     0.2876   5.440 5.32e-08 **
partyidrepublicans     3.8305     0.2037  18.801  < 2e-16 **
partyidmissing         1.0224     1.2645   0.809 0.418765
ideomoderate           0.5971     0.3590   1.663 0.096257 .
ideoconservative       1.6459     0.2215   7.431 1.07e-13 **
ideomissing            1.4722     0.4063   3.624 0.000291 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1767.29  on 1302  degrees of freedom
Residual deviance:  799.31  on 1295  degrees of freedom
AIC: 815.31

Number of Fisher Scoring iterations: 6
```

Best BIC

```
blackTRUE              -2.1791     0.4420  -4.931 8.20e-07 **
partyidindependents     1.5648     0.2876   5.440 5.32e-08 **
partyidrepublicans      3.8305     0.2037  18.801  < 2e-16 **
partyidapolitical     -12.2197   535.4112  -0.023 0.981791
partyidmissing          1.0224     1.2645   0.809 0.418765
ideomoderate            0.5971     0.3590   1.663 0.096257 .
ideoconservative        1.6459     0.2215   7.431 1.07e-13 **
ideomissing             1.4722     0.4063   3.624 0.000291 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1768.36  on 1303  degrees of freedom
Residual deviance:  799.31  on 1295  degrees of freedom
AIC: 817.31

Number of Fisher Scoring iterations: 12
```

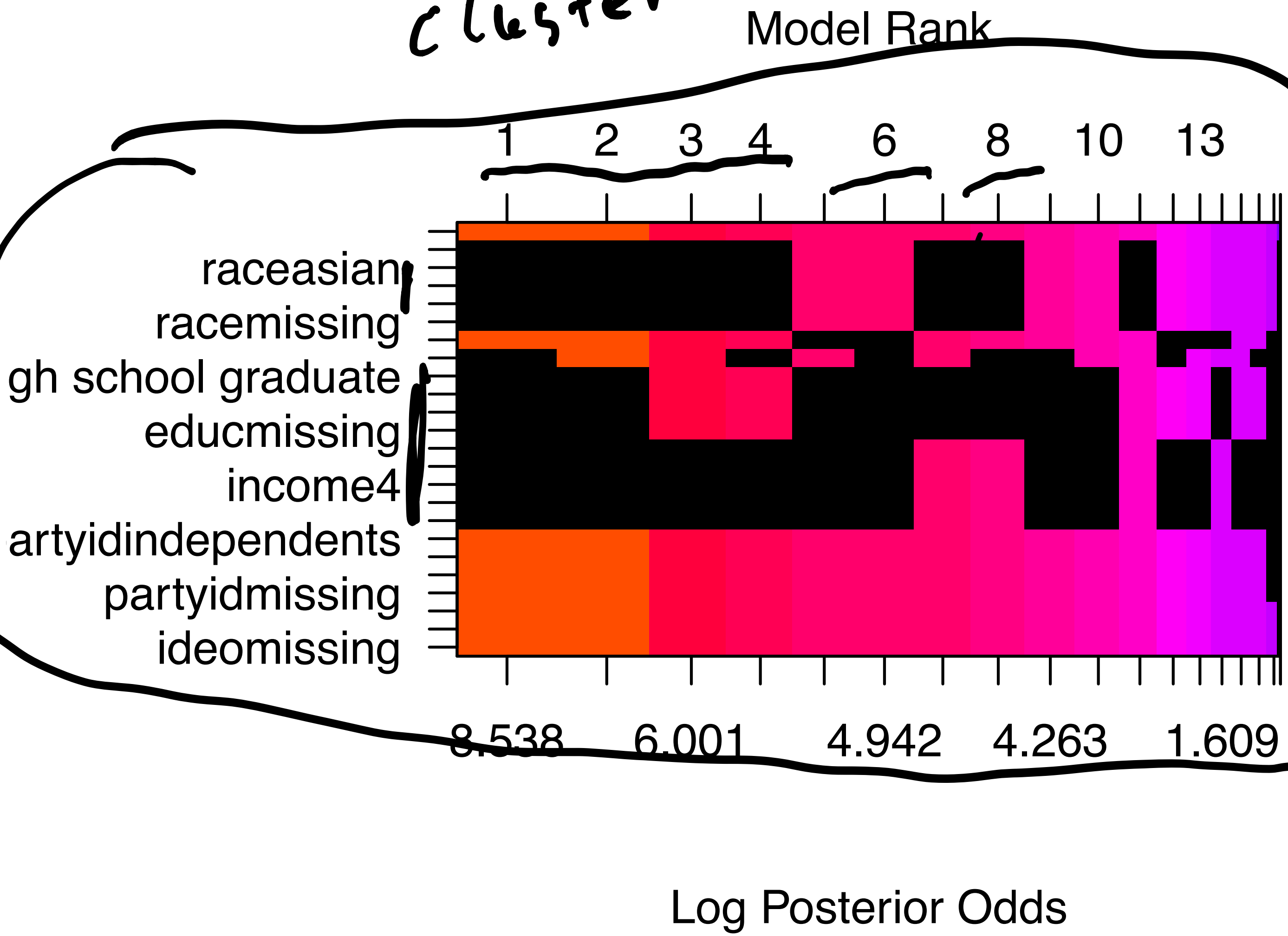
BAS with AIC

```r
library(BAS)
# nes1992 = mutate(nes1992, vote=as.numeric(vote))
vote.BAS.AIC = bas.glm(vote ~ ., data=nes1992,
                   family=binomial(),
                   method="MCMC",
                   n.models=256, MCMC.iterations=10000,
                   betaprior=aic.prior(),
                   modelprior=uniform())
```

Top models

```r
image(vote.BAS.AIC, rotate=F)
```

cluster models



BAS with BIC

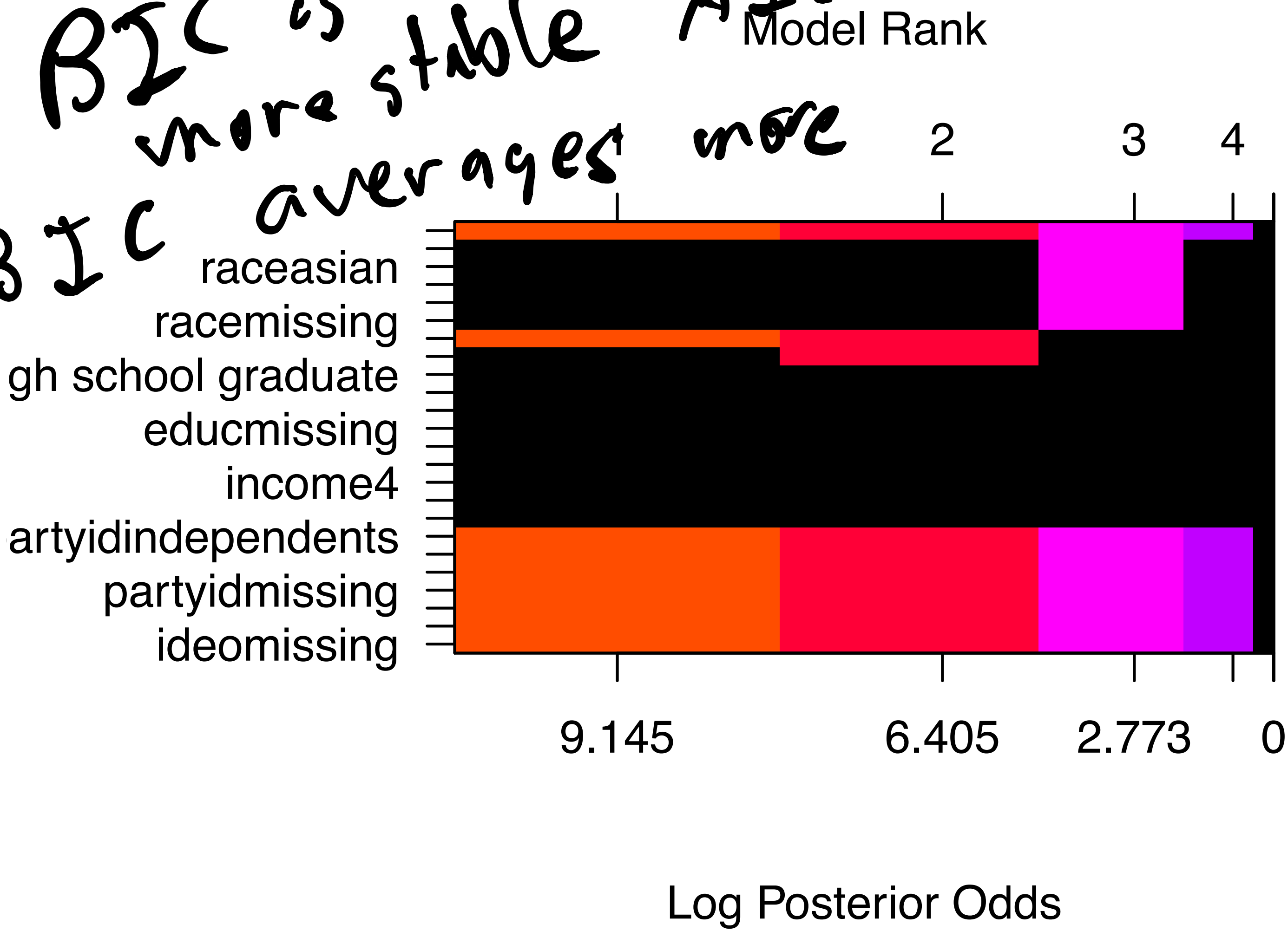
```r
nes1992 = mutate(nes1992, vote=as.numeric(vote))
vote.BAS.BIC = bas.glm(vote ~ ., data=nes1992,
                       family=binomial(),
                       method='MCMC',
                       n.models=256,  MCMC.iterations=10000,
                       betaprior=bic.prior(n = nrow(nes1992)),
                       modelprior=uniform())
```

Top models

```
image(vote.BAS.BIC, rotate=F)
```

BIC is more stable AIC

BIC averages more



Summary

- Various model selection criteria may not all agree on best model

Summary

- Various model selection criteria may not all agree on best model
- competing goals of finding the "true" model versus best for prediction

Summary

- Various model selection criteria may not all agree on best model
- competing goals of finding the "true" model versus best for prediction
- exhaustive search is not always possible for big p

Summary

- Various model selection criteria may not all agree on best model
- competing goals of finding the "true" model versus best for prediction
- exhaustive search is not always possible for big p
- Stochastic Search