432 Class 26 Slides

github.com/THOMASELOVE/432-2018

2018-04-19

Getting Started

Preliminaries

```r
library(skimr)
library(rms)
library(aplore3) # for a data set
library(ResourceSelection) # for Hosmer-Lemeshow test
library(bestglm) # for a demonstration of all subsets
library(broom)
library(tidyverse)
```

Today's Agenda

- Loose Ends
  - Spending Degrees of Freedom in Project 2, and in Life
  - Logistic Regression
    - The Hosmer-Lemeshow test
    - Variable Selection with "All Subsets" / `bestglm`

Spending Degrees of Freedom

For Project 2

Remember that for Project 2, I've simplified your decision-making. See section 6.19 of the instructions or the Class 26 README.

- Basically, I want you to restrain yourself on the size of the model you wind up with.
- Even if you have a big sample, don't fit project 2 models with more than 20 degrees of freedom or more than 8 predictors, if you can possibly help it.
    - My brain can't hold even that many degrees of freedom together at one time. Anything more will leave me a quivering mess.

On Fitting Regression Models

Some things are always true. . .

- Being honest about your findings is important.
- Identifying a stable phenomenon to study is crucial.
- Measuring that phenomenon well is crucial.
- Being transparent about your work is important. Describing your work so it can be replicated, and then actually replicating it, are good things. Sharing your data and your materials is a good idea.
- Having a large sample size (n) is helpful in fitting models.

but the impact of lots of other things changes depending on why you're fitting a regression model.

The Key Point

Decide what your research question is, and use it to help you think about what's important in your modeling.

- Models that account for only a few of the possibly important dimensions of a problem don't lead to causal conclusions, but can help screen out important from less important areas for future work.
- Model development strategies that work well with large sample sizes (n) and small numbers of predictors to consider don't necessarily work well when the situation is reversed.
- Most problems involve missing data, and problems in measurement. Getting those issues settled effectively is often overlooked.
- The sample size you need to fit a regression model changes depending on your aims.

On Fitting Models

What are some of the reasons you might fit a linear (or generalized linear) model?

1. All prediction, no explanations
2. All description, external validity is irrelevant
3. Clinical/Scientific Prediction with strong priors
4. Above all else, simplicity
5. Causal inference
6. Risk Adjustment

On Fitting Models

What are some of the reasons you might fit a linear (or generalized linear) model?

1. (All prediction, no explanations) Because you want to make predictions about an outcome in new data based on some training data you have, but you're happy to take those predictions as emerging from a mysterious magic "black box" that cannot be peered into without spoiling the surprise. You don't care if your results are a little biased, so long as the predictions are strong. Parsimony doesn't matter to you.

- Some especially useful tools here include: variable (feature) selection through cross-validation, stepwise approaches, AIC and BIC, machine learning tools like regression trees, and other means of quickly searching through many possible models.
- Sample size is rarely a big issue here. The big problem is having more variables than you can possibly plot at once. You usually have enough data to partition into separate development and test samples.

On Fitting Models

2. (All description, external validity is irrelevant) Because you want to describe, as accurately as possible, the nature of the associations you observe in the available data, but you don't care much at all whether the conclusions you draw will hold up in new data.

- Confidence intervals for coefficients (slopes, mostly), and sometimes you'll run the model on clinically relevant cutpoints rather than continuous predictors to see what's happening more simply. Simple polynomial models can be appealing, and you'll sometimes want to build this in the ANCOVA context, where you're looking for the impact of specific pre-specified interactions.
- Residual plots play a big role here in deciding whether the model "fits" well enough, or identifying cases when it doesn't.
- Cross-validation is useful, but not a big part of model selection or convincing people that the model is "right" or not.

On Fitting Models

3. (Clinical prediction) You want to do an excellent job predicting an outcome in new data, but you have a lot of prior knowledge about the predictors under consideration, and want to use that information to help produce prediction rules as effectively as possible. You welcome the fact that most relationships are non-linear, but would like to be parsimonious if possible, as data are often expensive.

- Some especially useful tools here include: scatterplot matrices (when the number of predictors is modest), cross-validation, assessments of discrimination and calibration, Spearman's ρ^2 plot to point the way to non-linear terms that might be impactful if present, restricted cubic splines, polynomial functions, and graphical tools like nomograms
- Most stepwise tools aren't helpful here. We try to not "peek" at the outcome-predictor relationships to maintain unbiased estimates of the relationship without extensive validation.

On Fitting Models

4. (Above all else, simplicity) You want the problem to look like one in a statistics textbook, where everything is fit with the simplest possible model, where every term adds statistically significant predictive value, and where obtaining an unbiased estimate of the outcome is especially important. You still care a bit about what happens in new data, but you're mostly concerned about parsimonious model development.

    - Some especially useful tools include best subsets, stepwise approaches, and methods for pruning a set of predictors with clustering or principal components analysis. These models usually make the (often incorrect) assumption that relationships are linear.
    - Often this approach is used by people who are trying to pre-specify their entire model in advance, and want to be sure they can "explain" the result when they are done. That may not be a reasonable thing to hope for. This is a place where the "rule of 20" can be very helpful in setting expectations in advance.

On Fitting Models

5. (Causal inference) You want to identify whether a particular causal pathway you have pre-specified matches up well with what you see in new data.
6. (Risk Adjustment) You want to identify the impact of a particular exposure/predictor on an outcome, while controlling for the effects of a series of additional predictors. Perhaps you've done a randomized experiment / clinical trial, and want to identify whether particular results meet a standard for statistical (as well as clinical) significance. Power is very important.

- In either case, bias is very important, and you want to avoid it. Careful design of a comparison group (like I teach in 500) is a very good way to go about this work, but it's also true that there's a lot of epidemiology that goes into drawing causal conclusions, or even thinking hard about an association.
- Often the details of modeling take a back seat here to the details of designing the study (and the comparison groups) in the first place.

Assessing the Quality of a Logistic Regression Model

A Quick Example

SOURCE: Hosmer and Lemeshow (2000) Applied Logistic Regression: Second Edition. These data are copyrighted by John Wiley & Sons Inc. and must be acknowledged and used accordingly. Data were collected at Baystate Medical Center, Springfield, Massachusetts during 1986.

```r
# uses aplore3 package for data set
lbw <- aplore3::lowbwt
head(lbw,3)
```

```
  id      low age lwt  race smoke  ptl  ht  ui       ftv
1  4 < 2500 g  28 120 Other   Yes  One  No Yes      None
2 10 < 2500 g  29 130 White    No None  No Yes Two, etc.
3 11 < 2500 g  34 187 Black   Yes None Yes  No      None
   bwt
1  709
2 1021
3 1135
```

Fit a logistic regression model

```
model_10 <- glm(low ~ lwt + ptl + ht,
                data = lbw, family = binomial)
model_10
```

```

Call:  glm(formula = low ~ lwt + ptl + ht, family = binomial,

Coefficients:
 (Intercept)          lwt       ptlOne  ptlTwo, etc.
     1.17016     -0.01851      1.74219       0.15105
       htYes
     1.91234

Degrees of Freedom: 188 Total (i.e. Null);  184 Residual
Null Deviance:      234.7
Residual Deviance: 207.4    AIC: 217.4
```

The Hosmer-Lemeshow Test of Goodness of Fit

See this link from Jonathan Bartlett.

- The Hosmer-Lemeshow goodness of fit test is based on dividing the sample into g groups (g is often chosen to be 10) according to their predicted probabilities.
- We then calculate the expected number of Y $= 1$ outcomes (based on the model probabilities of Y $= 1$ in the group) in each of the 10 groups, and compare those results to what is observed in the data using a Pearson goodness of fit statistic.
- A significant result indicates statistically significant “lack of fit” but it’s easy to criticize the test (among other things, if you change the choice of g, you can materially change the *p* value.)

Running the Hosmer-Lemeshow Test

```r
# requires ResourceSelection package
hos <- hoslem.test(model_10$y, fitted(model_10), g = 10)
hos
```

```

    Hosmer and Lemeshow goodness of fit (GOF) test

data:  model_10$y, fitted(model_10)
X-squared = 5.6459, df = 8, p-value = 0.6868
```

So there's no evidence of poor fit. We can also get a table of observed vs. expected.

Observed vs. Expected Table from H-L test

```
cbind(hos$observed, hos$expected)
```

```
               y0 y1     yhat0     yhat1
[0.0305,0.14] 16  3 17.226390  1.773610
(0.14,0.2]    16  4 16.600448  3.399552
(0.2,0.225]   20  4 18.773041  5.226959
(0.225,0.252] 12  2 10.599250  3.400750
(0.252,0.263] 16  4 14.817304  5.182696
(0.263,0.288] 14  3 12.265960  4.734040
(0.288,0.322] 12  6 12.521738  5.478262
(0.322,0.384] 11  8 12.395653  6.604347
(0.384,0.637]  7 12  9.658303  9.341697
(0.637,0.947]  6 13  5.141914 13.858086
```

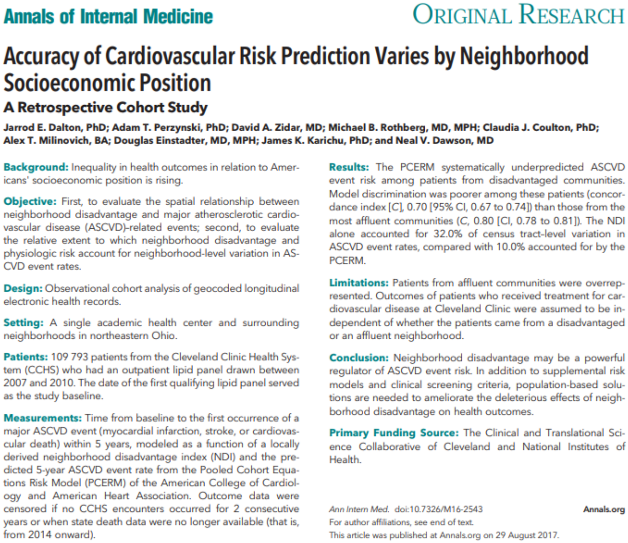
Change the number of groups, *g*?

```
for (i in 5:15) {
  print(hoslem.test(model_10$y,
                    fitted(model_10), g=i)$p.value)
}
```

```
[1] 0.288379
[1] 0.1888862
[1] 0.6682719
[1] 0.7830653
[1] 0.6564895
[1] 0.6868256
[1] 0.4444073
[1] 0.4520327
[1] 0.6381505
[1] 0.6437715
[1] 0.07773936
```

Building a Plot to Assess Model Accuracy in Logistic Regression

Motivation



Annals of Internal Medicine

ORIGINAL RESEARCH

Accuracy of Cardiovascular Risk Prediction Varies by Neighborhood Socioeconomic Position

A Retrospective Cohort Study

Jarrod E. Dalton, PhD; Adam T. Perzynski, PhD; David A. Zidar, MD; Michael B. Rothberg, MD, MPH; Claudia J. Coulton, PhD; Alex T. Milinovich, BA; Douglas Einstadter, MD, MPH; James K. Karichu, PhD; and Neal V. Dawson, MD

Background: Inequality in health outcomes in relation to Americans' socioeconomic position is rising.

Objective: First, to evaluate the spatial relationship between neighborhood disadvantage and major atherosclerotic cardiovascular disease (ASCVD)-related events; second, to evaluate the relative extent to which neighborhood disadvantage and physiologic risk account for neighborhood-level variation in ASCVD event rates.

Design: Observational cohort analysis of geocoded longitudinal electronic health records.

Setting: A single academic health center and surrounding neighborhoods in northeastern Ohio.

Patients: 109 793 patients from the Cleveland Clinic Health System (CCHS) who had an outpatient lipid panel drawn between 2007 and 2010. The date of the first qualifying lipid panel served as the study baseline.

Measurements: Time from baseline to the first occurrence of a major ASCVD event (myocardial infarction, stroke, or cardiovascular death) within 5 years, modeled as a function of a locally derived neighborhood disadvantage index (NDI) and the predicted 5-year ASCVD event rate from the Pooled Cohort Equations Risk Model (PCERM) of the American College of Cardiology and American Heart Association. Outcome data were censored if no CCHS encounters occurred for 2 consecutive years or when state death data were no longer available (that is, from 2014 onward).

Results: The PCERM systematically underpredicted ASCVD event risk among patients from disadvantaged communities. Model discrimination was poorer among these patients (concordance index [*C*], 0.70 [95% CI, 0.67 to 0.74]) than those from the most affluent communities (*C*, 0.80 [CI, 0.78 to 0.81]). The NDI alone accounted for 32.0% of census tract-level variation in ASCVD event rates, compared with 10.0% accounted for by the PCERM.

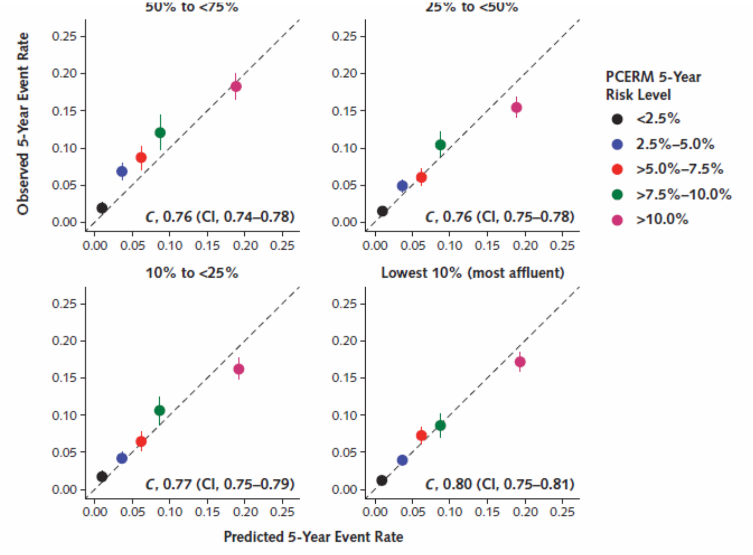
Limitations: Patients from affluent communities were overrepresented. Outcomes of patients who received treatment for cardiovascular disease at Cleveland Clinic were assumed to be independent of whether the patients came from a disadvantaged or an affluent neighborhood.

Conclusion: Neighborhood disadvantage may be a powerful regulator of ASCVD event risk. In addition to supplemental risk models and clinical screening criteria, population-based solutions are needed to ameliorate the deleterious effects of neighborhood disadvantage on health outcomes.

Primary Funding Source: The Clinical and Translational Science Collaborative of Cleveland and National Institutes of Health.

Ann Intern Med. doi:10.7326/M16-2543 **Annals.org**
For author affiliations, see end of text.
This article was published at Annals.org on 29 August 2017.

Dalton et al. (2017) Figure 2 (partial)



Dalton et al. (2017) Caption

Perfect calibration of the PCERM is represented along the line y = x; points above this line indicate underestimation of risk by the PCERM in relation to observed event rates, and points below it indicate overestimation of risk. Concordance indices (C) and corresponding 95% CIs are displayed within each panel. The C ranges from 0.5 to 1.0, where a value of 0.5 represents no discrimination of events from nonevents and a value of 1.0 represents complete separation of outcomes.

Evaluating a Logistic Model's Accuracy

Dividing into 5 groups via quintiles of the predicted response (.fitted)...

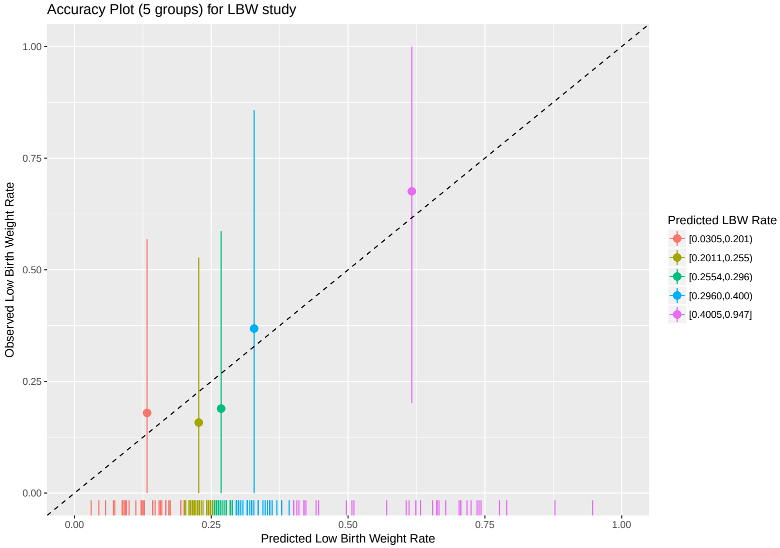
```
m10_aug <- augment(model_10, type.predict = "response")
m10_aug$.obs <- model_10$y
m10_aug$.cat5 <- Hmisc::cut2(m10_aug$.fitted, g = 5)

(perform.1 <- m10_aug %>% select(low, .obs, .fitted, .cat5))
```

```
          low .obs    .fitted          .cat5
1    < 2500 g    1 0.66613977 [0.4005,0.947]
2    < 2500 g    1 0.22503748 [0.2011,0.255)
3    < 2500 g    1 0.40625849 [0.4005,0.947]
4    < 2500 g    1 0.94688994 [0.4005,0.947]
5    < 2500 g    1 0.40048220 [0.4005,0.947]
6    < 2500 g    1 0.16703190 [0.0305,0.201)
7    < 2500 g    1 0.34850570 [0.2960,0.400)
8    < 2500 g    1 0.63243376 [0.4005,0.947]
9    < 2500 g    1 0.65447500 [0.4005,0.947]
```

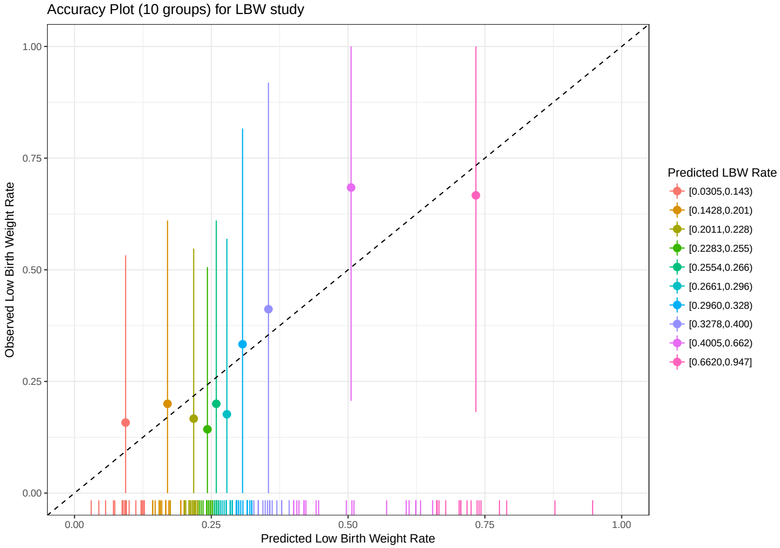
Building the Accuracy Plot

```r
perform.1 %>%
    group_by(.cat5) %>%
    summarize(obs_mean = mean(.obs), obs_sd = sd(.obs),
              fit_mean = mean(.fitted)) %>%
 ggplot(., aes(x = fit_mean, y = obs_mean, col = .cat5)) +
  geom_point(size = 3) +
  geom_linerange(aes(x = fit_mean,
                     ymin = pmax(0, obs_mean - obs_sd),
                     ymax = pmin(1, obs_mean + obs_sd))) +
  geom_abline(slope = 1, linetype = "dashed") +
  geom_rug(data = perform.1, aes(x = perform.1$.fitted,
                 y = perform.1$.obs), sides = "b") +
  scale_color_discrete(name = "Predicted LBW Rate") +
  lims(y = c(0,1), x = c(0, 1)) +
  labs(y = "Observed Low Birth Weight Rate",
      x = "Predicted Low Birth Weight Rate",
      title = "Accuracy Plot (5 groups) for LBW study")
```

The Logistic Regression Accuracy Plot



Accuracy Plot with 10 deciles



Can we use “best subsets” or “all subsets” in logistic regression?

Using the bestglm package to do "best subsets" with logistic models

There are several available tools. If you're interested, I'd start with this link.

- The bestglm package does an exhaustive (all subsets) search through a glm (slowly) using AIC, BIC or cross-validation, for example.
- This expects the data to be in a certain form, for instance, the outcome must be named *y* and you can have no extraneous variables present.
- The tutorial you'll find at the link above is pre-tidyverse. I don't know how well bestglm works with the tidyverse, but it's not likely to be much worse than leaps does.
- This approach can also be used with linear models which include multi-categorical predictors.
- bestglm is going to do an exhaustive search, which will be slow with large data sets, or large pools of predictors.
- In addition to bestglm there are several other appealing tools for looking at large numbers of potential models quickly.

Preparing the Data Set

```r
lbw_for_bestglm <- lbw %>%
    mutate(y = low) %>%
    select(age, lwt, race, smoke, ptl, ht, ui, ftv, y)
```

Must include only the variables we want to include in the search, and then the outcome, which must be labeled y, in that order.

lbw_for_bestglm data: First few observations

```
head(lbw_for_bestglm)
```

```
  age lwt  race smoke  ptl  ht  ui       ftv        y
1  28 120 Other   Yes  One  No Yes      None < 2500 g
2  29 130 White    No None  No Yes Two, etc. < 2500 g
3  34 187 Black   Yes None Yes  No      None < 2500 g
4  25 105 Other    No  One Yes  No      None < 2500 g
5  25  85 Other    No None  No Yes      None < 2500 g
6  27 150 Other    No None  No  No      None < 2500 g
```

Search through all subsets, with AIC as criterion

```r
res.best.logistic <-
    bestglm(Xy = lbw_for_bestglm,
            family = binomial,
            IC = "AIC",
            method = "exhaustive")
```

```
Morgan-Tatar search since family is non-gaussian.
Note: factors present with more than 2 levels.
```

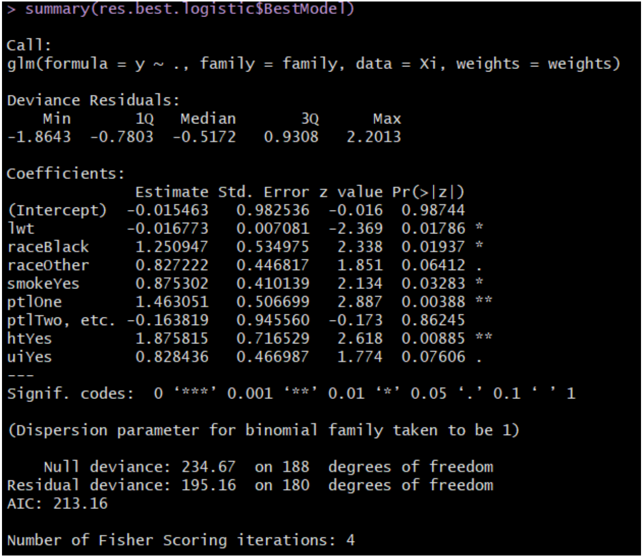
This approach (with a gaussian family) can also be used for OLS.

Show top 5 models chosen by AIC

```r
res.best.logistic$BestModels
```

```
    age  lwt  race smoke  ptl   ht    ui   ftv Criterion
1 FALSE TRUE  TRUE  TRUE TRUE TRUE  TRUE FALSE  211.1647
2  TRUE TRUE  TRUE  TRUE TRUE TRUE  TRUE FALSE  211.9063
3 FALSE TRUE  TRUE  TRUE TRUE TRUE FALSE FALSE  212.2697
4  TRUE TRUE  TRUE  TRUE TRUE TRUE FALSE FALSE  212.6828
5  TRUE TRUE FALSE FALSE TRUE TRUE  TRUE FALSE  213.4539
```

Show result for best model, according to AIC



```
> summary(res.best.logistic$BestModel)

Call:
glm(formula = y ~ ., family = family, data = Xi, weights = weights)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-1.8643  -0.7803  -0.5172   0.9308   2.2013  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)   
(Intercept)  -0.015463   0.982536  -0.016  0.98744   
lwt          -0.016773   0.007081  -2.369  0.01786 * 
raceBlack     1.250947   0.534975   2.338  0.01937 * 
raceOther     0.827222   0.446817   1.851  0.06412 . 
smokeYes      0.875302   0.410139   2.134  0.03283 * 
ptlOne        1.463051   0.506699   2.887  0.00388 **
ptlTwo, etc. -0.163819   0.945560  -0.173  0.86245   
htYes         1.875815   0.716529   2.618  0.00885 **
uiYes         0.828436   0.466987   1.774  0.07606 . 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 195.16  on 180  degrees of freedom
AIC: 213.16

Number of Fisher Scoring iterations: 4
```

Search all subsets, using BIC instead of AIC

```r
res.best.logistic2 <-
    bestglm(Xy = lbw_for_bestglm,
            family = binomial,
            IC = "BIC",
            method = "exhaustive")
```

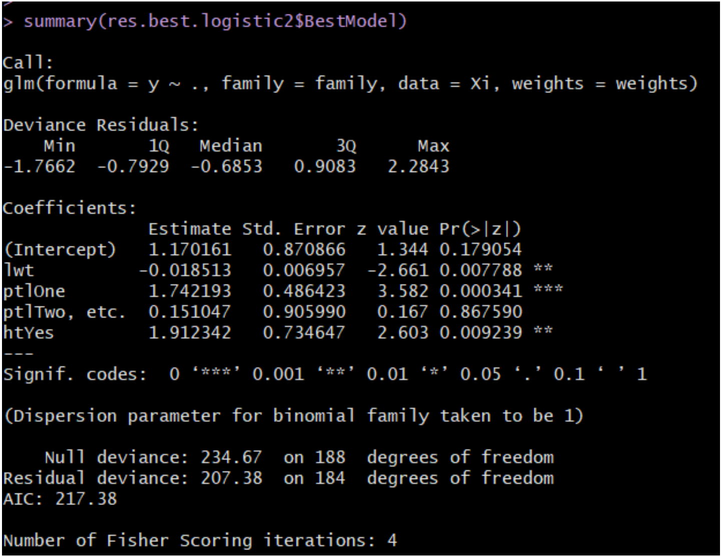
```
Morgan-Tatar search since family is non-gaussian.
Note: factors present with more than 2 levels.
```

Show top 5 models by BIC

```
res.best.logistic2$BestModels
```

```
    age   lwt  race smoke  ptl    ht    ui   ftv Criterion
1 FALSE  TRUE FALSE FALSE TRUE  TRUE FALSE FALSE  228.3477
2 FALSE FALSE FALSE FALSE TRUE FALSE FALSE FALSE  230.1914
3  TRUE FALSE FALSE FALSE TRUE FALSE FALSE FALSE  230.2738
4 FALSE  TRUE FALSE FALSE TRUE FALSE FALSE FALSE  230.3406
5 FALSE  TRUE FALSE FALSE TRUE  TRUE  TRUE FALSE  230.3604
```

Show result for best model, according to BIC



```
> summary(res.best.logistic2$BestModel)

Call:
glm(formula = y ~ ., family = family, data = Xi, weights = weights)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7662  -0.7929  -0.6853   0.9083   2.2843

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.170161   0.870866   1.344 0.179054
lwt          -0.018513   0.006957  -2.661 0.007788 **
ptlOne        1.742193   0.486423   3.582 0.000341 ***
ptlTwo, etc.  0.151047   0.905990   0.167 0.867590
htYes         1.912342   0.734647   2.603 0.009239 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 207.38  on 184  degrees of freedom
AIC: 217.38

Number of Fisher Scoring iterations: 4
```

Best Models, by Criterion

There are 8 predictors under consideration.

- age, lwt, race, smoke, ptl, ht, ui and ftv

Model (by AIC)	Rank	Model (by BIC)
lwt, smoke, ptl, ht, ui	1	lwt, ptl, ht
all except ftv	2	ptl
lwt, race, smoke, ptl, ht	3	age, ptl
all except ui, ftv	4	lwt, ptl
age, lwt, ptl, ht, ui	5	lwt, ptl, ht, ui

Search all subsets, using Cross-Validation

Can be done when all predictors are continuous or binary, but **not** if you have categorical predictors with more than two levels.

```
# not run here because we have
# some multi-categorical predictors
set.seed(432)
res.best.logistic_cv <-
    bestglm(Xy = lbw_for_bestglm,
            family = binomial,
            IC = "CV")
```