ENV 710: Lecture 12

model selection

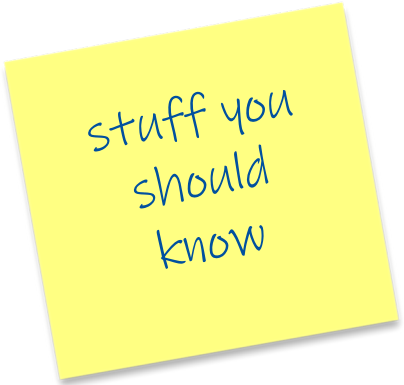
Duke | NICHOLAS SCHOOL OF THE ENVIRONMENT

linear models

model selection

learning goals

- what is model selection and why do it?
- understand the terminology of model selection
  - null model, saturated model, maximal model, minimum adequate model
  - parsimony
  - nested models
- what is the process of model selection
  - forward and backward model selection
  - model selection criteria: t-test, F-test, adjusted R^2, AIC
- more specifics about linear models
  - building models with interactions
  - interpretation of interaction effects



explore your data

head off problems before they occur and know what to look out for as you proceed

- look for surprises (outliers, typos, etc.)
- look for gross deviations from assumptions (e.g., lack of normality, lack of linear relationship between DV and IV's, strong multicollinearity
- determine correct model based on type of DV (e.g., continuous or discrete variable)

build your model

based on hypotheses, study design, & DV's

- determine model structure
- should IV's be treated as continuous or nominal?
- do you need to standardize IV's?
- do you need to include interactions, exponentials or polynomials?
- do you need to include random effects?
- do you need to transform DV's?

refine your model

determine best model based on hypotheses

- do you need the most parsimonious model?
- remove non-significant terms?
- use F-test for comparing nested models
- use AIC for non-nested models

verify your model

evaluate model assumptions

- check residuals plots for normality, homoscedasticity and influential data points
- check for other assumptions, such as overdispersion (Poisson GLM)
- if assumptions aren't met, need to adapt by potentially transforming data, standardizing IV's or restructuring the model

interpret your model

make your conclusions

- is omnibus H0 statistically significant?
- does model explain much variance in DV?
- assess effect sizes (β's)
- graph results to demonstrate effects
- accept/reject hypotheses
- make predictions

example

cognitive test scores

Can the cognitive test scores of 3 & 4-year olds (kid.score) be predicted from characteristics of their mothers?

- mother's IQ (`mom.iq`),
- completion of high school (`mom.hs`): 1 = completed, 0 = failed
- number of years of work (`mom.work`).

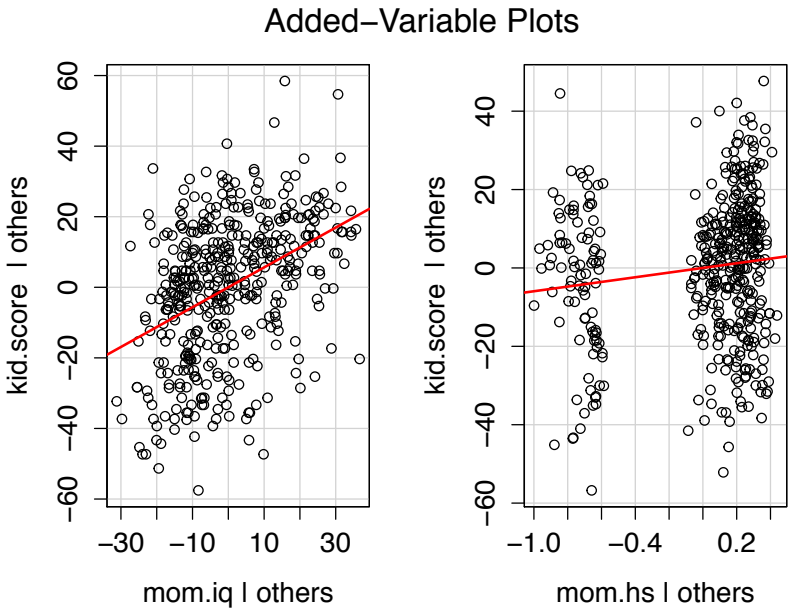
play along… download `kidiq.csv`

(a) run the model
(b) interpret the results
(c) plot the results in a single plot of number of years worked against cognitive test scores, with individual regressions for each level of mother's high school

$$Y = \beta_0 + \beta_1 mom.iq + \beta_2 mom.hs$$

example

cognitive test scores



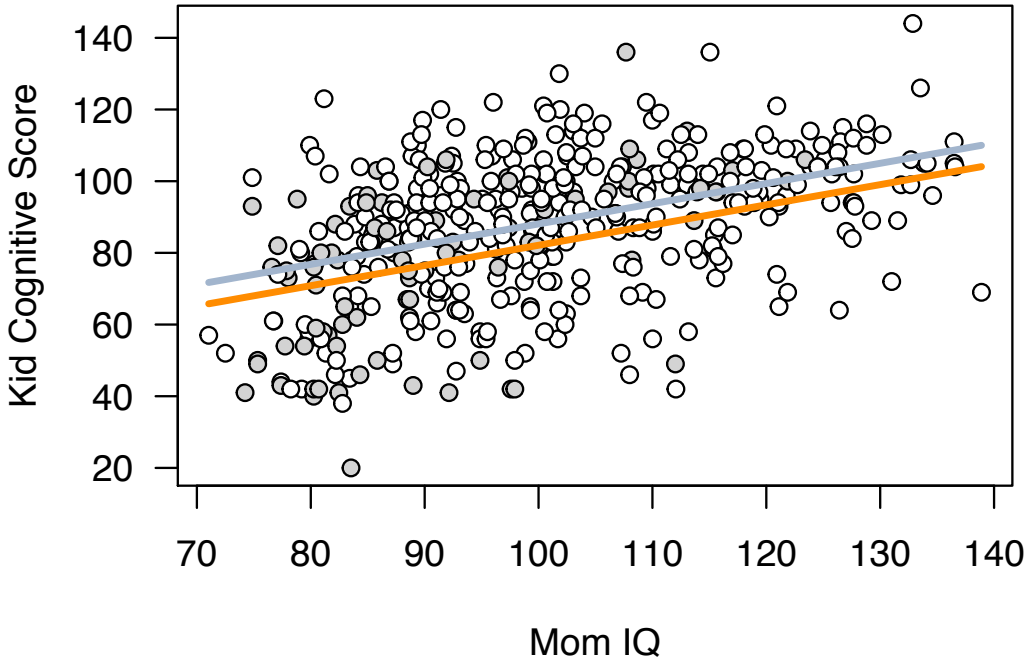

```
ex1 <- lm(kid.score ~ mom.iq + factor(mom.hs))
  avPlots(ex1)
```

```
Call:lm(formula = kid.score ~ mom.iq + mom.hs)

Coefficients:  Estimate Std. Error t value Pr(>|t|)
(Intercept)   25.73154    5.87521   4.380 1.49e-05 ***
mom.iq         0.56391    0.06057   9.309  < 2e-16 ***
mom.hs         5.95012    2.21181   2.690  0.00742 **

Residual standard error: 18.14 on 431 degrees of freedom
Multiple R-squared:  0.2141, Adjusted R-squared:  0.2105
F-statistic: 58.72 on 2 and 431 DF,  p-value: < 2.2e-16
```

example

cognitive test scores

- plot Mom IQ over Kid score, for moms that completed and did not complete high school



model selection

model selection is the way to identify the model that is best supported by the data ('best model') from among a set of candidate models… can be used to identify the hypothesis that is best supported by observations

- find the smallest set of variables that provides an adequate description of the data
- available explanatory variables are candidate variables
- if we have k candidate variables, there are potentially $2k$ models to consider (each term being in or out of a model)

model selection

model selection is a trade-off between

complexity and fit

reflects conflicting interests…

- describe the data reasonably well
- build a model simple enough to be interpretable

the model…

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \varepsilon$$

statistical modeling basics

- null model: the mean is the only parameter → no explanatory power

$$Y = \beta_0 + \varepsilon$$

- saturated model: includes a parameter for every data point ($k = n$) → no explanatory power

- maximal model: contains all factors, interactions and covariates of interest

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \varepsilon$$

- minimum adequate model: only includes explanatory variables that improve the fit of the model to the data

prefer the most parsimonious model

nested models

- two models are nested if one model contains all the terms of the other, and at least one additional term
  - larger model → full model
  - smaller model → restricted model

[1] $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2 + \varepsilon$

[2] $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$

[3] $Y = \beta_0 + \beta_1 X_1 + \varepsilon$

[4] $Y = \beta_0 + \beta_2 X_2 + \varepsilon$

[5] $Y = \beta_0 + \beta_1 X_1 + \beta_4 X_4 + \beta_5 X_1 X_4 + \varepsilon$

stepwise model selection

forward model selection – start with the null model and add variables one by one

backwards model selection – start with a full model and remove variables one by one

backwards selection procedure

- fit maximal model, then simplify:
  - remove non-significant interaction terms
  - remove non-significant quadratic terms
  - remove non-significant explanatory variables
  - group together factor levels that do not differ from one another

- caveat: reductions should make sense and not lead to significant reductions in explanatory power

- decide between a full model and a restricted model

H_0: reduced model is adequate

H_a: full model is better

model selection – one term

- when the full model has exactly one more term than the reduced model, can use a *t*-test:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 X_2$$

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2$$

- H_0: "reduced model is adequate" equivalent to: $H_0 : \beta_3 = 0$

```
Call:lm(formula = kid.score ~ mom.iq + mom.hs)

Coefficients:  Estimate   Std. Error   t value    Pr(>|t|)
(Intercept)    25.73154   5.87521      4.380      1.49e-05 ***
mom.iq         0.56391    0.06057      9.309      < 2e-16 ***
mom.hs         5.95012    2.21181      2.690      0.00742 **
```

model selection – multiple terms

- for nested models, if full model has more than 1 additional term than the restricted model, then use the partial *F*-test
- compare the error sum of squares (SSE) for the reduced model to the SSE for the full model
- SSE_F for the full model will always be less than the SSE_R of the reduced model, the question is by how much?

partial F-test

$$F = \frac{SSE_R - SSE_F}{df_R - df_F} / \frac{SSE_F}{df_F}$$

$$df_R - df_F = (n - k_R) - (n - k_F)$$

$$df_F = n - k_F - 1$$

tests the ratio of increase in sum of squares to increase in df's

partial F-test

Of the two models compared below, which is the full model? Which is the reduced model?

What hypothesis does this analysis test?

Should we accept Model 1 or Model 2?

significantly less variance is explained by the reduced model (Model 2) compared to the fuller model (Model 1)

$$F = \frac{144137 - 141757}{1} / \frac{141757}{431} = 7.236$$

```
anova(ex1, ex2)
```

```
Analysis of Variance Table
Model 1: kid.score ~ mom.iq + factor(mom.hs)
Model 2: kid.score ~ mom.iq

  Res.Df    RSS    Df  Sum of Sq    F       Pr(>F)
1    431   141757
2    432   144137  -1   -2380.2   7.2369   0.007419 **
```

procedure in detail

- deviance is the "difference between two models" = RSS in linear models
  - RSS is the same as SSE (sum of squared errors)
  - small RSS indicates a tight fit of the model to the data

- order matters if explanatory variable are correlated
  - significance will depend on the order it is added or omitted from the model
  - test objectively by model simplification

1. fit the maximal model
2. begin model simplification
   - inspect parameter estimates and remove non-significant terms, starting with interactions
   - if deletion causes an insignificant increase in deviance
     - leave the term out of the model
   - if deletion causes significant increase in deviance
     - put the term back in the model
3. keep removing terms from the model

model selection

- t-test can be used when the full model has one more term than the reduced model
- partial *F*-test works for nested models when the full model has more than one additional terms than the restricted model
- if models are not nested, compare adjusted R^2 and residual standard error or use AIC
- AIC is the relative good fit of a statistical model
  - cannot tell how well data fit a model in any absolute sense
  - selects a model from a set of models
  - seeks a model that has a good fit to the truth, but few parameters (favors small residual error, but penalizes for including additional predictor variables)

Akaike's Information Criterion (AIC)

- for multiple regression, the formula is:

$$AIC = n \cdot log(\frac{RSS}{n}) + 2k$$

- where *k* is the number of parameters, *n* is the sample size, and RSS is the residual sum of squares
- minimize AIC
- if $n/k < 40$, use AICc

example

cognitive test scores

Can the cognitive test scores of 3 & 4-year olds (`kid.score`) be predicted from characteristics of their mothers?

- mother's IQ (`mom.iq`),
- completion of high school (`mom.hs`): 1 = completed, 0 = failed
- number of years of work (`mom.work`)

full model

```
[1] lm(kid.score ~ mom.hs*mom.iq*mom.work)
```

```
[2] lm(kid.score ~ mom.hs + mom.iq + mom.work + mom.hs*mom.iq
+ mom.hs*mom.work + mom.iq*mom.work + mom.hs*mom.iq*mom.work)
```

$$Y = \beta_0 + \beta_1(IQ) + \beta_2(HS) + \beta_3(WK) + \beta_4(IQ)(HS) + \beta_5(HS)(WK) +$$

$$\beta_6(IQ)(WK) + \beta_7(IQ)(HS)(WK)$$

potential models

```
fit.1 <- lm(kid.score ~ mom.hs*mom.iq*mom.work)
```

$$Y = \beta_0 + \beta_1(IQ) + \beta_2(HS) + \beta_3(WK) + \beta_4(IQ)(HS) + \beta_5(HS)(WK) + \beta_6(IQ)(WK) + \beta_7(IQ)(HS)(WK)$$

```
fit.2 <- update(fit.1,.~.-mom.hs:mom.iq:mom.work)
```

$$Y = \beta_0 + \beta_1(IQ) + \beta_2(HS) + \beta_3(WK) + \beta_4(IQ)(HS) + \beta_5(HS)(WK) + \beta_6(IQ)(WK)$$

```
fit.3 <- update(fit.2,.~.-mom.iq:mom.work)
```

$$Y = \beta_0 + \beta_1(IQ) + \beta_2(HS) + \beta_3(WK) + \beta_4(IQ)(HS) + \beta_5(HS)(WK)$$

```
fit.4 <- update(fit.3,.~.-mom.hs:mom.work)
```

$$Y = \beta_0 + \beta_1(IQ) + \beta_2(HS) + \beta_3(WK) + \beta_4(IQ)(HS)$$

```
fit.5 <- update(fit.4,.~.-mom.work)
```

$$Y = \beta_0 + \beta_1(IQ) + \beta_2(HS) + \beta_4(IQ)(HS)$$

```
fit.6 <- update(fit.5,.~.-mom.hs:mom.iq)
```

$$Y = \beta_0 + \beta_1(IQ) + \beta_2(HS)$$

1. run each model & record the RSS and adjusted R^2
2. run the F-test
3. calculate AIC for all models

backwards selection

full model → reduced model
H_0: reduced model is adequate
H_a: fuller model is better

```
anova(fit.1, fit.2, fit.3, fit.4, fit.5, fit.6)
```

```
Analysis of Variance Table

Model 1: kid.score ~ mom.hs * mom.iq * mom.work
Model 2: kid.score ~ mom.hs + mom.iq + mom.work + mom.hs:mom.iq + mom.hs:mom.work +
mom.iq:mom.work
Model 3: kid.score ~ mom.hs + mom.iq + mom.work + mom.hs:mom.iq + mom.hs:mom.work
Model 4: kid.score ~ mom.hs + mom.iq + mom.work + mom.hs:mom.iq
Model 5: kid.score ~ mom.hs + mom.iq + mom.hs:mom.iq
Model 6: kid.score ~ mom.hs + mom.iq
```

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	426	138279				
2	427	138280	-1	-0.89	0.0027	0.958377
3	428	138486	-1	-206.52	0.6362	0.425518
4	429	138872	-1	-385.48	1.1875	0.276441
5	430	138879	-1	-6.98	0.0215	0.883512
6	431	141757	-1	-2878.44	8.8677	0.003068 **

model selection with AIC

full model → reduced model

```
AIC(fit.1, fit.2, fit.3, fit.4, fit.5, fit.6)
```

	df	AIC	ΔAIC
fit.1	9	3751.207	
fit.2	8	3749.210	1.997
fit.3	7	3747.858	1.352
fit.4	6	3747.064	0.794
fit.5	5	3745.086	1.978
fit.6	4	3751.989	-6.903

rule of thumb:

Δ **AIC < 2** → models more or less equivalent
Δ **AIC 4-7** → models clearly distinguishable
Δ **AIC > 10** → models definitely different

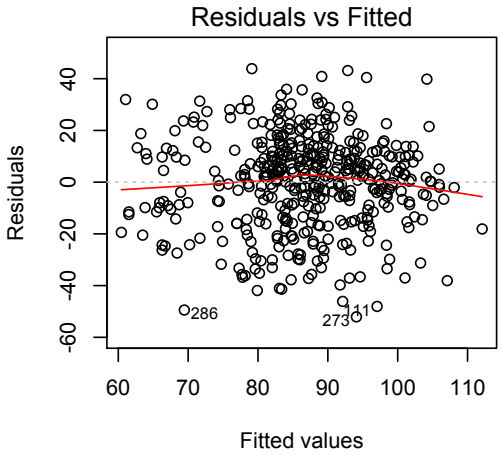
* simply dropping models with Δ **AIC > 2 risks discarding** useful models

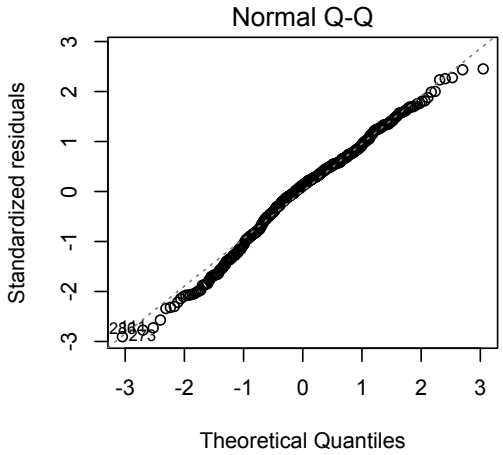
minimum adequate model - best model

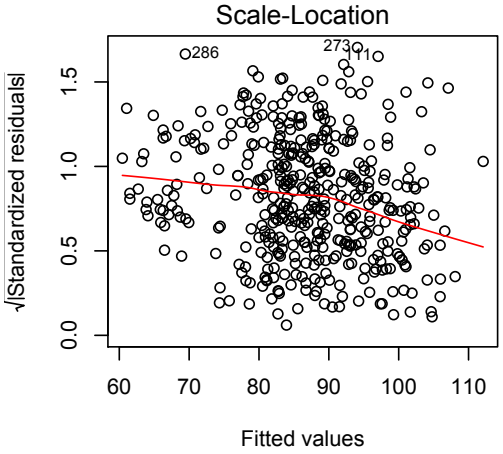
$$Y = -11.48 + 51.27(HS) + 0.97(IQ) - 0.48(HS)(IQ)$$

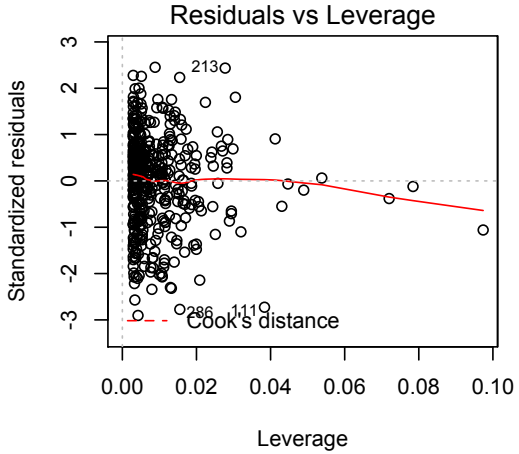
```
summary(fit.5)
```

```
Call:
lm(formula = kid.score ~ mom.hs + mom.iq + mom.hs:mom.iq)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   -11.4820    13.7580  -0.835 0.404422
mom.hs         51.2682    15.3376   3.343 0.000902 ***
mom.iq          0.9689     0.1483   6.531 1.84e-10 ***
mom.hs:mom.iq  -0.4843     0.1622  -2.985 0.002994 **
---

Residual standard error: 17.97 on 430 degrees of freedom
Multiple R-squared:  0.2301,    Adjusted R-squared:  0.2247
F-statistic: 42.84 on 3 and 430 DF,  p-value: < 2.2e-16
```


Residuals vs Fitted
Residuals
Fitted values
286
273
111


Normal Q-Q
Standardized residuals
Theoretical Quantiles


Scale-Location
√|Standardized residuals|
Fitted values
286
273
111


Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
213
286
111

what do the coefficients mean?

```
Call:
lm(formula = kid.score ~ mom.hs + mom.iq + mom.hs:mom.iq)

Residuals:
    Min      1Q  Median      3Q     Max
-52.092 -11.332   2.066  11.663  43.880

Coefficients:
              Estimate   Std. Error t value Pr(>|t|)
(Intercept)    -11.4820    13.7580  -0.835 0.404422
mom.hs          51.2682    15.3376   3.343 0.000902 ***
mom.iq           0.9689     0.1483   6.531 1.84e-10 ***
mom.hs:mom.iq   -0.4843     0.1622  -2.985 0.002994 **
---
```

Questions?

Code slide 8

```
cols <- c("lightgrey", "white")

with(kdat, plot(mom.iq, kid.score, las = 1, ylab = "Mom IQ",
   xlab = "Kid Cognitive Score", pch = 21, bg =
     cols[factor(mom.hs)]))

x <- with(kdat, seq(min(mom.iq), max(mom.iq), 100))

kcoef <- coef(lm1)

curve(kcoef[1] + kcoef[2]*x + kcoef[3]*0, col = "darkorange",
   add = T, lwd = 3)

curve(kcoef[1] + kcoef[2]*x + kcoef[3]*1, col =
   "lightsteelblue3", add = T,
     lwd = 3)
```

Code slide 8

```
g1 <- ggplot(kdat, aes(mom.iq, kid.score)) +
       geom_point(aes(group = mom.hs, colour = factor(mom.hs)),
        size = 3, show.legend = F, alpha = 0.3) +
       xlab("Mom IQ") + ylab("Kid cognitive score") +
       theme_bw()

 cc <- data.frame(sl = c(coef(ex1)[2], coef(ex1)[2]),
                  int = c(coef(ex1)[1],coef(ex1)[1]+coef(ex1)[3]),
                  col = c(1,0))

 g2 <- g1 + geom_abline(data = cc, aes(slope =sl, intercept = int,
                                       colour = factor(col)), size = 1) +
    scale_colour_brewer(palette="Set1") + theme(legend.position="none")

 g2
```