

Lecture 8

1. OLS linear regression: Proc Reg
2. Residual plots, ODS plots
3. Model reduction, subset selection
4. Predictions

1

How can we be sure our SAS code is correct?

- Send problem observations to separate dataset for further investigation.
- Check code with test data.
- Print input data and output data and compare randomly chosen observations.
- Break complex data step into parts and check results after each part.
- Perform a check for the specific problem you suspect may occur.

On course website: *Seeing Red: Tips for Debugging the SAS Data Step* by M. Lee

2

Data were analyzed using the SAS System for Windows (release 9.1; SAS Institute Inc, Cary, NC) and SUDAAN (release 9.0; Research Triangle Institute, Research Triangle Park, NC) software programs. All analyses included sample weights that account for the unequal probabilities of selection because of oversampling and nonresponse. All variance calculations incorporate the sample weights and account for the complex sample design. We replicated the main analyses with 2 separate SAS programs written independently by 2 of us (K.M.F., B.I.G.).

OGTT problem from HW 2: one IF statement for each row of table.

```
data A;
  set pubh.ogtt_hw2;
  if (sex="F") then sex="f";
  missing = (min000=. or min030=. or min060=. or min090=. or min120=.);
  length OGTT_category $20. ;
  OGTT_category = "problem"; * overwritten when calculated;
  middle = max(min030, min060, min090);
  if ( (0 LE min000 < 100) and (0 LE min120 < 140) and (middle < 200))
    then OGTT_category="1_NGT";
  if ((100 LE min000 LE 125) and (0 LE min120 < 140) and (middle < 200))
    then OGTT_category="2_NGT+IFG";
  if ((0 LE min000 < 100) and (0 LE min120 < 140) and (middle GE 200))
    then OGTT_category="3_NGT-Indt";
```

```

if ( (0 LE min000 < 100) and (140 LE min120 < 200))
  then OGTT_category="4_IGT";
if ( (100 LE min000 LE 125) and (140 LE min120 < 200))
  then OGTT_category="5_IGT+IFG";

if ( (0 LE min000 < 100) and (min120 GE 200))
  then OGTT_category="6_CFRD no FH";
if ( (100 LE min000 LE 125) and (min120 GE 200))
  then OGTT_category="7_CFRD no FH+IFG";
if ( (min000 GE 125) and (min120 GE 200))
  then OGTT_category="8_CFRD+FH";

```

Result: 991 patients in categories, 28 problem/missing observations

5

Student questions:

1. *NGT and NGT-IFG both require all three 30-60-90 mins to be <200. We weren't sure how to do this beyond a series of AND statements so that it would only be included if all three were under 200 (and not missing).*

In your code you used the MAX function, which we read as finding the largest number in the set and making a constant for that observation. That constant was then compared with 200. However, if there were missing data, wouldn't that be overlooked by the max function?

6

2. You first labeled all *OGGT_Category* as "problem" and then overwrote that with new categories as you defined them.

We looked over each individual piece, but we still weren't getting the same numbers. We were suspicious that perhaps some observations could somehow fall into more than one category and so be overwritten twice. To test this, we ran the code after the *OGGT_Category* = "problem" in the reverse order. We ended up with 6 more categorized observations.

7

```
data C;
  set pubh.ogtt_hw2;
  if (sex="F") then sex="f";
  missing = (min000=. or min030=. or min060=. or min090=. or min120=.);
  length OGTT_category $20. ;
  OGTT_category = "problem";
  middle = max(min030, min060, min090);
  if ( (min000 GE 125) and (min120 GE 200)) then OGTT_category="8_CFRD+FH";
  if ( (100 LE min000 LE 125) and (min120 GE 200)) then OGTT_category="7_CFRD no FH+IFG";
  if ( (0 LE min000 < 100) and (min120 GE 200)) then OGTT_category="6_CFRD no FH";
  if ( (100 LE min000 LE 125) and (140 LE min120 < 200)) then OGTT_category="5_IGT+IFG";
  if ( (0 LE min000 < 100) and (140 LE min120 < 200)) then OGTT_category="4_IGT";
  if ( (0 LE min000 < 100) and (0 LE min120 < 140) and (middle GE 200))
    then OGTT_category="3_NGT-Indt";
  if ( (100 LE min000 LE 125) and (0 LE min120 < 140) and (middle < 200))
    then OGTT_category="2_NGT+IFG";
  if ( (0 LE min000 < 100) and (0 LE min120 < 140) and (middle < 200))
    then OGTT_category="1_NGT";
```

Result: 991 patients in categories, 28 problem/missing observations
(same as before)

How can we tell whether observations are falling into 2 categories?

8

```
data A problems1; make 2 datasets with separate output statements
  (lines of code);
  length OGTT_category $20. ;
  (lines of code);
  IF (OGTT_category = "problem" or missing=1) then output problems1;
  ELSE output A;
```

```
data C problems2; make 2 datasets
  (lines of code);
  length reverse_OGTT $20. ;
  (lines of code reverse order);
  IF (reverse_OGTT = "problem" or missing=1) then output problems2;
  ELSE output C;
```

9

```
proc sort data=A; by id;
proc sort data=C; by id;
```

```
data check;
  merge A C;
  by id;
  if (reverse_OGTT NE OGTT_category); keep mismatches only
```

```
proc print data=check;
```


Another approach to finding observations in more than one category:

```
data A;
  set pubh.ogtt_hw2;
  if (sex="F") then sex="f";
  missing = (min000=. or min030=. or min060=. or min090=. or min120=.);
  length OGTT_category $20. ;
  OGTT_category = "problem";
  count = 0;
  middle = max(min030, min060, min090);
  if ( (0 LE min000 < 100) and (0 LE min120 < 140) and (middle < 200))
    then do ; OGTT_category="1_NGT"; count = count+1; end ;
  (same do-loop for other categories)
```

13

```
data check;
  set A;
  if (count > 1);

proc print data=check;
```

This would find the same observation that was erroneously put in 2 categories.

Do we need RETAIN count ?

14

Ordinary least squares linear regression

- Proc GLM (General Linear Model) assumes response is continuous, predictors can be categorical or continuous
- Proc Reg assumes response and all predictors are continuous

SAS Help > SAS/STAT > SAS/STAT User's Guide > The REG Procedure

15

Minnesota math scores 2000 example:

- `district` school district number
- `school` ID number
- `LEP_pct` percent of students with Limited English Proficiency
- `Special_Ed_pct` percent of students in Special Education
- `free_lunch_pct` percent of students receiving free or reduced-price lunch
- `mobility_pct` mobility index (percent)
- `drop_out_pct` percent of students dropping out of school
- `total_8th_graders` total eighth grade enrollment
- `total_students` kindergarten through 12th grade enrollment
- `operating_budget` district operating expenditure per student (*district level*)
- `total_budget` district total expenditure per student (*district level*)

16

Format for Proc Reg:

Proc Reg;

```
MODEL response = list of predictors ;
```

We have taken logs of several predictors, and dropped horizontal outliers from our *trimmed* data set, B:

```
Proc Reg data=B;
```

```
model mathscore = log_lep log_lunch  
Special_Ed_pct log_mobility drop_out_pct  
log_8th_grade log_total_students operating_budget  
total_budget;
```

17

The SAS System

1

The REG Procedure

Model: MODEL1

Dependent Variable: mathscore

Number of Observations	Read	417
Number of Observations	Used	413
Number of Observations	with Missing Values	4

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	3416.69187	379.63243	61.09	<.0001
Error	403	2504.52363	6.21470		
Corrected Total	412	5921.21550			

Root MSE	2.49293	R-Square	0.5770
Dependent Mean	53.67070	Adj R-Sq	0.5676
Coeff Var	4.64486		

18

Root MSE = square root of mean square error

= estimate of error standard deviation, $\hat{\sigma}$

R-Square = $SS(\text{Model}) / SS(\text{Total})$, proportion of variability “explained” by model

Adjusted R Squared adjusts for the number of predictors in the model, because R^2 always increases as predictors are added whether they help or not.

$$\text{Adjusted R Squared} = 1 - \frac{(n-1)(1-R^2)}{n-p},$$

where n is the number of observations and p is the number of predictors in the model. What if $p = 1$?

In comparing regression models, many prefer adjusted R^2 , because adding a non-significant predictor may *reduce* adjusted R^2 .

19

Estimated regression coefficients:

Parameter Estimates					
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	71.49920	2.07620	34.44	<.0001
log_lep	1	-0.65463	0.13643	-4.80	<.0001
log_lunch	1	-1.89919	0.27951	-6.79	<.0001
Special_Ed_pct	1	-0.07866	0.03944	-1.99	0.0468
log_mobility	1	-1.53474	0.29114	-5.27	<.0001
drop_out_pct	1	-0.56735	0.15351	-3.70	0.0002
log_8th_grade	1	0.21923	0.30184	0.73	0.4681
log_total_students	1	-0.72168	0.39947	-1.81	0.0716
operating_budget	1	-0.46716	0.30285	-1.54	0.1237
total_budget	1	-0.05989	0.23671	-0.25	0.8004

Test of $H_0: \beta_j = 0$ is $t = \hat{\beta}_j / SE(\hat{\beta}_j)$, with df from Error

Pr > |t| gives 2-sided p -value

20

Plots from Proc Reg

There are at least three ways to get decent plots from Proc Reg

1. Using plots options

```
ods graphics on;
Proc Reg plots = ( list of keywords
    RstudentsByPredicted(label) CooksD(label) );
ID variable to use to label observations ;
model y = x1 x2;
run;
ods graphics off;
```

21

2. Using plot statement

```
Proc Reg ;
model y = x1 x2;
plot rstudent. * predicted. ; dot is part of the name
plot cookd. * obs. ;
```

22

3. Write out to a dataset the studentized residuals, fitted values, etc.

Use Proc Gplot with this output dataset to make plots.

```
Proc Reg data = M;  
  model y = x1 x2;  
  OUTPUT out = Z  
  rstudent = rhat predicted = yhat CookD = cook_distance ;
```

Z is the name of the output dataset.

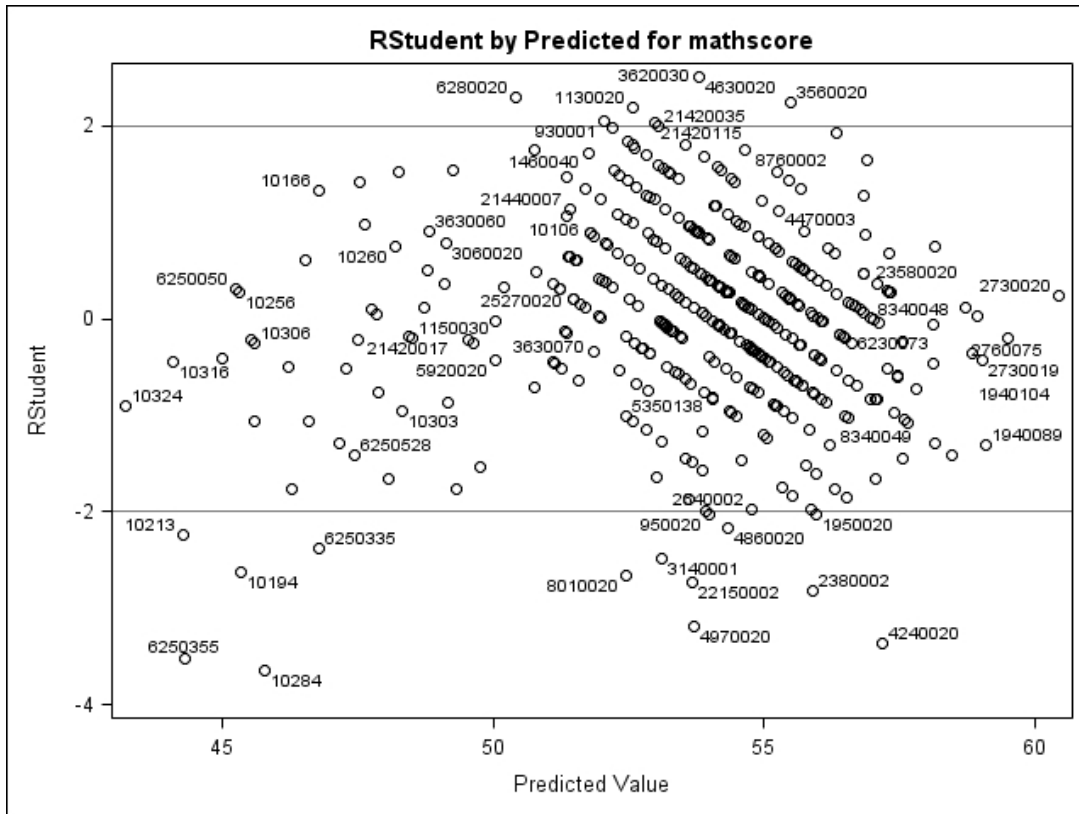
Z includes all variables of original data (M) plus the variables listed on OUTPUT

23

plots option

```
ods graphics on;  
Proc Reg data=B plots=( rstudentsbypredicted(label) CooksD(label) );  
  id school;  
  model mathscore = log_lep log_lunch  
    Special_Ed_pct log_mobility drop_out_pct  
    log_8th_grade log_total_students operating_budget  
    total_budget;  
run;  
ods graphics off;
```

24



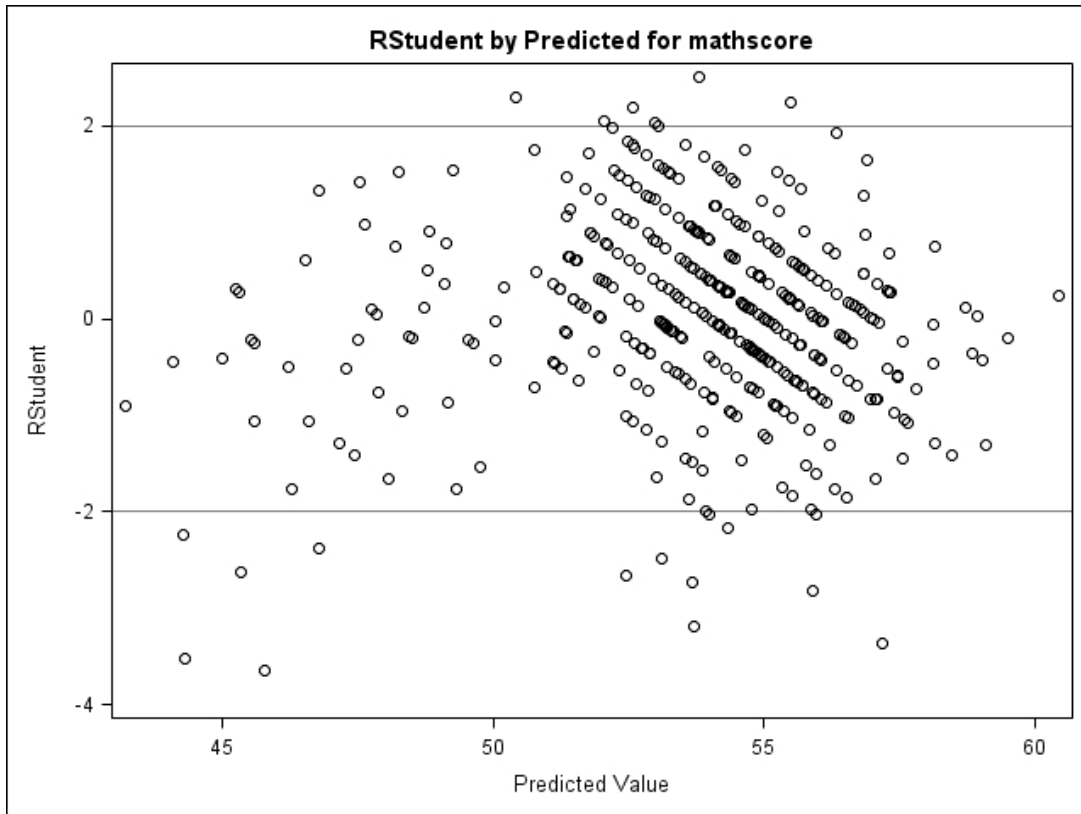
25

Without the labels:

```
ods graphics on;
Proc Reg data=B plots=( rstudentsbypredicted );
  model mathscore = log_lep log_lunch
    Special_Ed_pct log_mobility drop_out_pct
    log_8th_grade log_total_students operating_budget
    total_budget;
run;
ods graphics off;
```

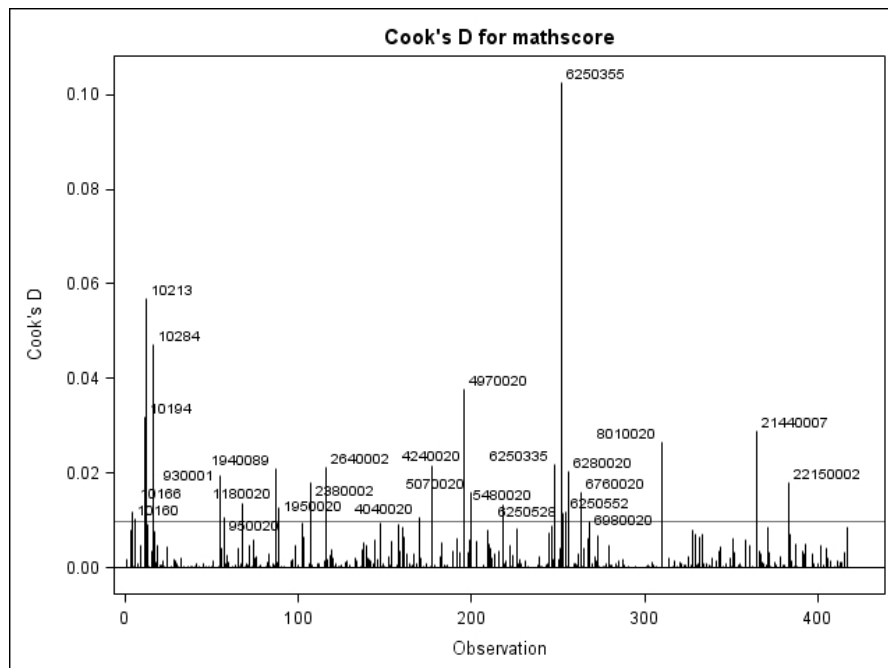
These plots open in a SAS Graphics window; save as image, or copy/paste, or use screen capture.

26



27

Cook's distance estimates change in regression coefficients when one observation is omitted. Large values are close to 1.

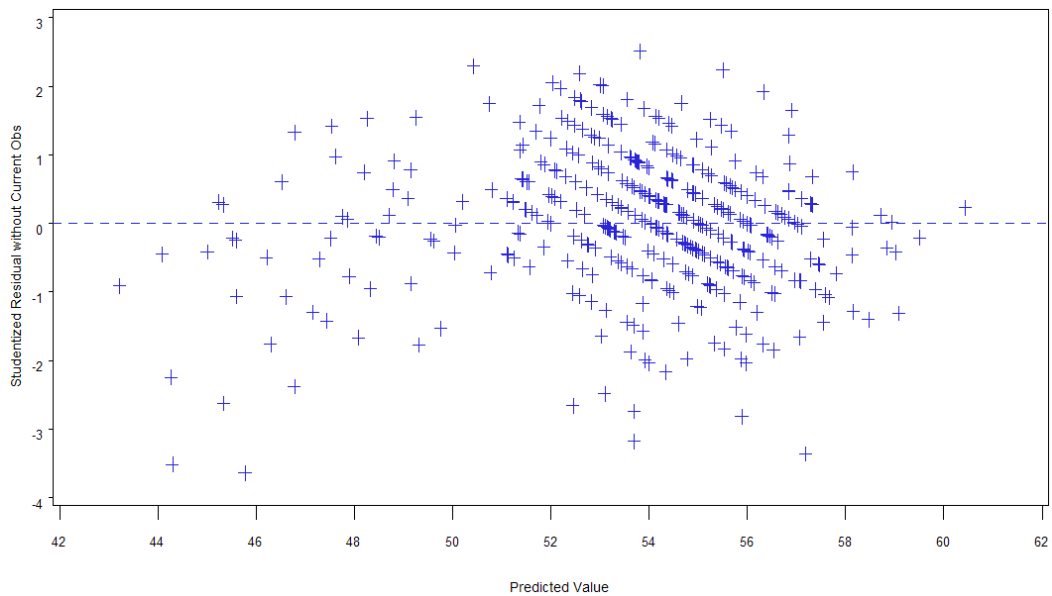


28

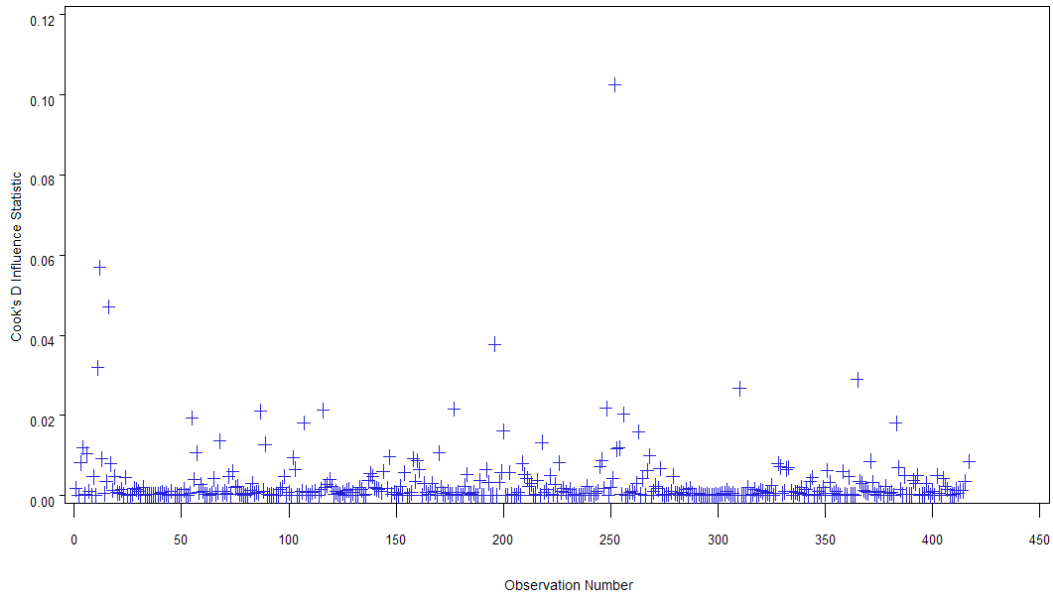
Plot statement

```
Proc Reg data=B;  
  model mathscore = log_lep log_lunch  
    Special_Ed_pct log_mobility drop_out_pct  
    log_8th_grade log_total_students operating_budget  
    total_budget;  
  plot rstudent. * predicted. ; dot is part of the name  
  plot cookd. * obs. ;
```

29



30



31

Model reduction

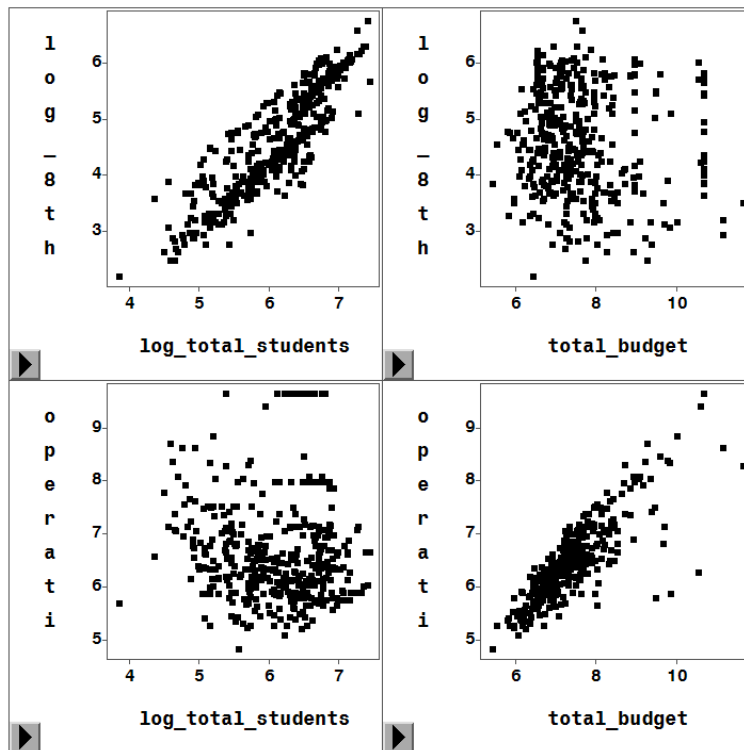
Idea: use the fitted regression coefficients to decide which $H_0 : \beta_j = 0$ are really true.

Drop these unnecessary variables to get a simpler model, with lower error variance.

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	71.49920	2.07620	34.44	<.0001
log_lep	1	-0.65463	0.13643	-4.80	<.0001
log_lunch	1	-1.89919	0.27951	-6.79	<.0001
Special_Ed_pct	1	-0.07866	0.03944	-1.99	0.0468
log_mobility	1	-1.53474	0.29114	-5.27	<.0001
drop_out_pct	1	-0.56735	0.15351	-3.70	0.0002
log_8th_grade	1	0.21923	0.30184	0.73	0.4681
log_total_students	1	-0.72168	0.39947	-1.81	0.0716
operating_budget	1	-0.46716	0.30285	-1.54	0.1237
total_budget	1	-0.05989	0.23671	-0.25	0.8004

32

These four predictors are related in pairs.



33

Compute the Pearson correlation

```
Proc Corr data=B;  
  var log_8th_grade log_total_students operating_budget  
      total_budget;
```

34

Pearson Correlation Coefficients
 Prob > |r| under H0: Rho=0
 Number of Observations

	log_8th_ grade	log_total_ students	operating_ budget	total_ budget
log_8th_grade	1.00000 413	0.88067 <.0001 413	-0.13430 0.0063 413	-0.07663 0.1200 413
log_total_students	0.88067 <.0001 413	1.00000 417	-0.07471 0.1277 417	-0.02422 0.6219 417
operating_budget	-0.13430 0.0063 413	-0.07471 0.1277 417	1.00000 417	0.89177 <.0001 417
total_budget	-0.07663 0.1200 413	-0.02422 0.6219 417	0.89177 <.0001 417	1.00000 417

35

Predicted values

To get predictions from any of the regression procedures, create a new data set:
 values of explanatory variables for the new cases.

SET this new data on top of the original data.

Because the new rows are missing the response, they are omitted when fitting the model.

36

There are two kinds of predictions from a regression: a predicted *mean* and a predicted *observation*. The predicted values are identical, but the SE is larger for a predicted observation.

Usually we are interested in a predicted mean:

```
OUTPUT  OUT = fitted_values
        PREDICTED = yhat  LCLM = lower  UCLM = upper  STDP = se_mean ;
```

37

In Florida vote example we want a new observation (at Palm Beach).

Write predictions out to a data set:

```
OUTPUT  OUT = fitted_values
        PREDICTED = yhat  LCL = lower  UCL = upper  STDI = se_new_obs ;
```

Why is $SE(\text{predicted mean}) > SE(\text{predicted observation})$?

Additional variability of observation around mean.

38

Simple linear regression with one predictor x , SE of the predicted **mean** at x_p is

$$\sqrt{\hat{\sigma}^2 \left(\frac{1}{n} + \frac{(x_p - \bar{x})^2}{(n-1)SD(x)^2} \right)}$$

so predictions farther from center of the data \bar{x} have bigger $(x_p - \bar{x})^2$, bigger SE. $\hat{\sigma}^2$ is MS(Error).

When we predict a new observation, the variability is essentially the sum of the variability of the mean plus the variability around the mean.

Standard error of prediction for a **new observation** at x_p

$$\sqrt{\hat{\sigma}^2 + \hat{\sigma}^2 \left(\frac{1}{n} + \frac{(x_p - \bar{x})^2}{(n-1)SD(x)^2} \right)}$$