Replication ("Cross-validation")

```
/dos/brunner/442f09/lecture > ls  *.data
exploremath.data  replicmath.data

/dos/brunner/442f09/lecture > head exploremath.data
      1   2   2    0   78.0    65   80    39   English   Female   3   3   1
      2   2   6    2   66.0    54   75    57   English   Female   3   3   1
      3   2   4    4   80.2    77   70    62   English   Male     5   6   1
      4   2   5    2   81.7    80   67    76   English   Female   2   2   1
      5   2   4    4   86.8    87   80    86   English   Male     5   5   1
      6   2   3    1   76.7    53   75    60   English   Male     3   3   1
      7   2   3    2   85.8    86   81    54   Other     Female   2   2   1
      8   2   4    3   73.0    75   77    17   English   Male     4   5   1
      9   2   6    2   72.3    63   60     2   English   Male     4   4   1
     10   2   8    6   90.3    87   88    76   English   Male     4   4   1

/dos/brunner/442f09/lecture > head replicmath.data
     1   2   4    0   80.8    78   85    56   Other     Female   16   16   2
     2   .   .    .   96.8    98   93    97   English   Male     14   14   2
     3   4   5    3     .      .    .     .         .        .    .    .   2
     4   .   .    .   71.8    69   62     .   English   Male     18   13   2
     5   2   6    3   84.7    89   80    50   English   Female    8    8   2
     6   2   3    4   81.3     0   77    67   English   Female   16   16   2
     7   2   6    6   87.8    82   90    70   Other     Female    8    1   2
     8   2   4    3   85.0    87   83    50   English   Female    6    6   2
     9   2   5    1   72.2    61   74     8   English   Male     14   14   2
    10   2   8    2   83.0    71   73    44   English   Female   16   16   2
```

The program `readmath2.sas` starts with the `proc format` from the front of `readmath.sas`, and then creates two SAS data sets, `mathex` and `mathrep`. After the `proc format`, `readmath2.sas` has:

```
data mathex;
     infile 'exploremath.data';
     input id course precalc calc gpa calculus english mark lang $ sex $
           nation1 nation2 sample;
```

Then it goes on just like `readmath.sas`, except that it includes the dummy variables for `ethnic` and `course`. Then we have

```
/********* Now create the data set mathrep, exactly parallel but from a
           different raw data file (random split) **********/

data mathrep;
     infile 'replicmath.data';
     input id course precalc calc gpa calculus english mark lang $ sex $
           nation1 nation2 sample;
```

Following this is an exact copy-paste of all the computed variables, labels and so on from the first part. The result is that when we `%include 'readmath2.sas'`, *both* data sets are available for analysis.

```
/* boncross.sas */
%include 'readmath2.sas';
title2 'Bonferroni-protected cross-validation for the regression model';

proc reg data=mathex;
     title3 'Exploratory sample:  Final model';
     model grade = hsgpa hscalc hsengl  totscore  mtongue;

proc reg data=mathrep;
     title3 'Replication sample:  T-test is significant if p < 0.05/5 = 0.01';
     model grade = hsgpa hscalc hsengl  totscore  mtongue;
```

Part of boncross.lst

Exploratory sample: Final model

R-Square 0.4635

Parameter Estimates

Variable	Label	DF	t Value	Pr > |t|
Intercept	Intercept	1	-5.93	<.0001
hsgpa	High School GPA	1	7.20	<.0001
hscalc	HS Calculus	1	2.13	0.0339
hsengl	HS English	1	-2.45	0.0150
totscore	Total # right on diagnostic test	1	3.86	0.0001
mtongue		1	-2.21	0.0280

Replication sample: T-test is significant if p < 0.05/5 = 0.01

R-Square 0.3925

Parameter Estimates

Variable	Label	DF	t Value	Pr > |t|
Intercept	Intercept	1	-4.79	<.0001
hsgpa	High School GPA	1	4.53	<.0001
hscalc	HS Calculus	1	1.71	0.0890
hsengl	HS English	1	-0.35	0.7297
totscore	Total # right on diagnostic test	1	5.99	<.0001
mtongue		1	-0.68	0.4960

.

Final conclusion: Among students who finish university Calculus, those with a higher high school grade point average tend to do better and those with good scores on the diagnostic test tend to do better.

Now we want to see how well we can predict final marks. In regression courses, you may learn about ``prediction intervals. '' When prediction intervals are based on exploratory analyses, they are suspect for exactly the same reasons that confidence intervals and significance tests are suspect. The best way to assess prediction is to try it on a new data set and see how well you do. That is, use your exploratory data set to generate predictions for your replication data set.

We'll do this here, but how? One obvious way is to produce Y-hat values from the regression equation, and see how close they are to the observed Y values in the replication data set. But for grades, what's important is how close the *letter* grades are. For example, if you predict 40% and the student gets a 20%, you're correct – an F is an F. But if you predict 80% and the student gets 60%, that 20 point error is the difference between A- and C-.

When I first tried to create a letter grade and predicted letter grade, I did it like this:

```
if 90 <= grade <= 100 then lgrade = 'A+';
   else if 85 <= grade <= 89 then lgrade = 'A ';
   else if 80 <= grade <= 84 then lgrade = 'A-';
   else if 77 <= grade <= 79 then lgrade = 'B+';
   else if 73 <= grade <= 76 then lgrade = 'B ';
   else if 70 <= grade <= 72 then lgrade = 'B-';
   else if 67 <= grade <= 69 then lgrade = 'C+';
   else if 63 <= grade <= 66 then lgrade = 'C ';
   else if 60 <= grade <= 62 then lgrade = 'C-';
   else if 57 <= grade <= 59 then lgrade = 'D+';
   else if 53 <= grade <= 56 then lgrade = 'D ';
   else if 50 <= grade <= 52 then lgrade = 'D-';
   else if  0 <= grade <= 49 then lgrade = 'F ';
```

The code is correct, but there is a problem. When `proc freq` composes a table, it sorts the values of the variables in alphabetical order, with 'A ' coming before 'A+ '. The result is a table that's hard to read. So as you will see, I gave numerical values to `lgrade`, and then used `proc format` to set up a printing format with the letters. This caused `proc freq` to sort the values by the internal (numerical) value, which is what we want. Incidentally, `proc glm` sorts by the formatted (printed); this default behaviour can be over-ridden by the `order=internal` option on the `proc glm` line.

```
/* predictmathgrade.sas */
%include 'readmath2.sas';
title2 'Predict Grade';
/* The mathrep data step continues. */

pregrade = -66.75516 + 1.58918*hsgpa + 0.21759*hscalc - 0.30024*hsengl
                     + 0.97213*totscore - 4.69657*mtongue;
pregrade = round(pregrade);
label pregrade = 'Predicted Grade';

dgrade = grade-pregrade;

proc means n mean std min max t probt;
     title3 'Means of Predicted vs. Observed grades';
     var pregrade grade dgrade;

proc corr;
     title3 'Correlation between predicted and observed grades';
     var grade pregrade;
proc plot;
     title3 'Observed by Predicted grade';
     plot grade*pregrade;


proc format;
     value gfmt 1 =  'F '
                2 =  'D-'  3 =  'D '   4 =  'D+'
                5 =  'C-'  6 =  'C '   7 =  'C+'
                8 =  'B-'  9 =  'B '  10 = 'B+'
                11 = 'A-' 12 =  'A '  13 = 'A+';

data replic2;
     set mathrep;
     if dgrade ne .; /* Must have both grade and predicted grade */
     if         90 <= grade <= 100 then lgrade = 13;
        else if 85 <= grade <= 89  then lgrade = 12;
        else if 80 <= grade <= 84  then lgrade = 11;
        else if 77 <= grade <= 79  then lgrade = 10;
        else if 73 <= grade <= 76  then lgrade = 9;
        else if 70 <= grade <= 72  then lgrade = 8;
        else if 67 <= grade <= 69  then lgrade = 7;
        else if 63 <= grade <= 66  then lgrade = 6;
        else if 60 <= grade <= 62  then lgrade = 5;
        else if 57 <= grade <= 59  then lgrade = 4;
        else if 53 <= grade <= 56  then lgrade = 3;
        else if 50 <= grade <= 52  then lgrade = 2;
        else if  0 <= grade <= 49  then lgrade = 1;
```

```
     if         90 <= pregrade <= 100 then plgrade = 13;
        else if 85 <= pregrade <= 89  then plgrade = 12;
        else if 80 <= pregrade <= 84  then plgrade = 11;
        else if 77 <= pregrade <= 79  then plgrade = 10;
        else if 73 <= pregrade <= 76  then plgrade = 9;
        else if 70 <= pregrade <= 72  then plgrade = 8;
        else if 67 <= pregrade <= 69  then plgrade = 7;
        else if 63 <= pregrade <= 66  then plgrade = 6;
        else if 60 <= pregrade <= 62  then plgrade = 5;
        else if 57 <= pregrade <= 59  then plgrade = 4;
        else if 53 <= pregrade <= 56  then plgrade = 3;
        else if 50 <= pregrade <= 52  then plgrade = 2;
        else if  0 <= pregrade <= 49  then plgrade = 1;
     label lgrade = 'Letter Grade'
           plgrade = 'Predicted Letter Grade';
     format lgrade plgrade gfmt.;

options pagesize=500;
proc freq;
     tables lgrade*plgrade / norow nopercent;

proc sort;
     by dgrade;

proc print;
     var hsgpa totscore plgrade lgrade pregrade grade dgrade;

/* Produce one final prediction equation for future use: Both data sets.
   Use merge to add variables.  */

data both;
     set mathex mathrep;
proc reg data=both;
     title3 'Final Prediction Equation';
     model grade = hsgpa totscore;
```

Gender, Ethnicity and Math performance 1
Predict Grade
Means of Predicted vs. Observed grades

The MEANS Procedure

Variable	Label	N	Mean	Std Dev	Minimum
pregrade	Predicted Grade	382	57.3115183	11.7014400	31.0000000
grade	Final mark (if any)	380	56.4421053	18.2574470	1.0000000
dgrade		288	-1.8888889	14.2719291	-64.0000000

Variable	Label	Maximum	t Value	Pr > |t|
pregrade	Predicted Grade	97.0000000	95.73	<.0001
grade	Final mark (if any)	97.0000000	60.26	<.0001
dgrade		31.0000000	-2.25	0.0255

Gender, Ethnicity and Math performance 2
Predict Grade
Correlation between predicted and observed grades

The CORR Procedure

2 Variables: grade pregrade

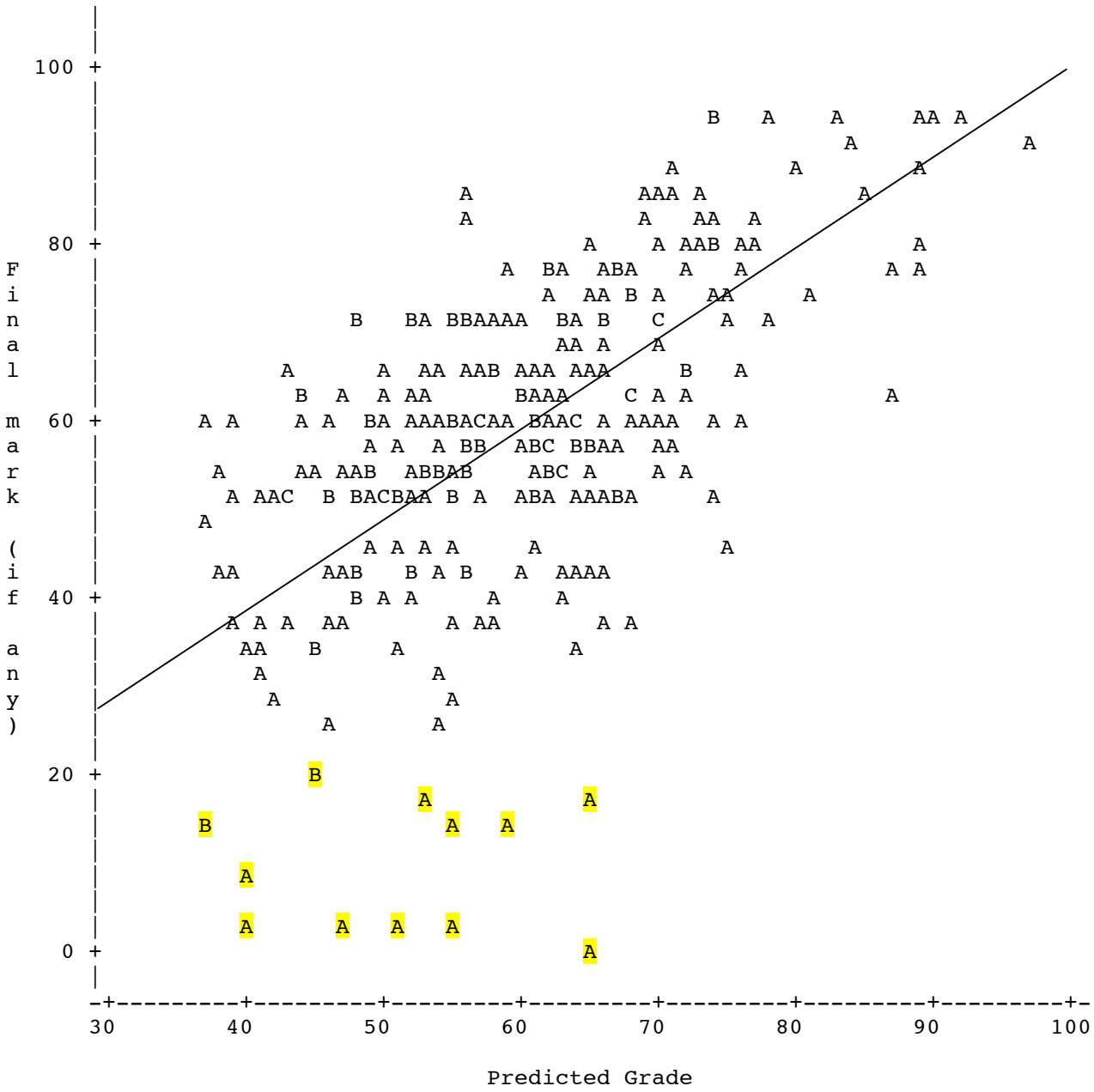
Variable	Label
grade	Final mark (if any)
pregrade	Predicted Grade

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

	grade	pregrade
grade	1.00000	0.60611
Final mark (if any)		<.0001
	380	288
pregrade	0.60611	1.00000
Predicted Grade	<.0001	
	288	382

Predict Grade
Observed by Predicted grade

Plot of grade*pregrade. Legend: A = 1 obs, B = 2 obs, etc.



NOTE: 291 obs had missing values.

The line Y = X has been added by hand.

Locate and investigate the highlighted observations. Did they take the final exam?

Predict Grade
Observed by Predicted grade

The FREQ Procedure

Table of lgrade by plgrade

lgrade(Letter Grade) plgrade(Predicted Letter Grade)

Frequency Col Pct	F	D-	D	D+	C-	Total
F	29 48.33	7 31.82	12 29.27	4 21.05	2 6.90	65
D-	11 18.33	6 27.27	3 7.32	1 5.26	4 13.79	32
D	7 11.67	1 4.55	8 19.51	2 10.53	5 17.24	30
D+	1 1.67	1 4.55	2 4.88	0 0.00	4 13.79	15
C-	8 13.33	3 13.64	6 14.63	5 26.32	4 13.79	37
C	1 1.67	2 9.09	3 7.32	3 15.79	5 17.24	23
C+	1 1.67	0 0.00	0 0.00	0 0.00	1 3.45	10
B-	2 3.33	2 9.09	5 12.20	3 15.79	1 3.45	23
B	0 0.00	0 0.00	0 0.00	0 0.00	1 3.45	14
B+	0 0.00	0 0.00	0 0.00	1 5.26	2 6.90	7
A-	0 0.00	0 0.00	1 2.44	0 0.00	0 0.00	14
A	0 0.00	0 0.00	1 2.44	0 0.00	0 0.00	9
A+	0 0.00	0 0.00	0 0.00	0 0.00	0 0.00	9
Total	60	22	41	19	29	288

(Continued)

Table of lgrade by plgrade

lgrade(Letter Grade) plgrade(Predicted Letter Grade)

Frequency Col Pct	C	C+	B-	B	B+	Total
F	9 20.93	1 5.88	0 0.00	1 5.56	0 0.00	65
D-	3 6.98	3 17.65	0 0.00	1 5.56	0 0.00	32
D	5 11.63	0 0.00	2 9.52	0 0.00	0 0.00	30
D+	4 9.30	1 5.88	2 9.52	0 0.00	0 0.00	15
C-	5 11.63	2 11.76	2 9.52	2 11.11	0 0.00	37
C	2 4.65	3 17.65	3 14.29	0 0.00	0 0.00	23
C+	5 11.63	0 0.00	2 9.52	1 5.56	0 0.00	10
B-	5 11.63	0 0.00	3 14.29	1 5.56	1 25.00	23
B	4 9.30	4 23.53	1 4.76	2 11.11	0 0.00	14
B+	0 0.00	1 5.88	1 4.76	1 5.56	0 0.00	7
A-	1 2.33	1 5.88	2 9.52	6 33.33	2 50.00	14
A	0 0.00	1 5.88	3 14.29	1 5.56	0 0.00	9
A+	0 0.00	0 0.00	0 0.00	2 11.11	1 25.00	9
Total	43	17	21	18	4	288

(Continued)

Table of lgrade by plgrade

lgrade(Letter Grade) plgrade(Predicted Letter Grade)

Frequency Col Pct	A-	A	A+	Total
F	0 0.00	0 0.00	0 0.00	65
D-	0 0.00	0 0.00	0 0.00	32
D	0 0.00	0 0.00	0 0.00	30
D+	0 0.00	0 0.00	0 0.00	15
C-	0 0.00	0 0.00	0 0.00	37
C	0 0.00	1 14.29	0 0.00	23
C+	0 0.00	0 0.00	0 0.00	10
B-	0 0.00	0 0.00	0 0.00	23
B	1 25.00	1 14.29	0 0.00	14
B+	0 0.00	1 14.29	0 0.00	7
A-	0 0.00	1 14.29	0 0.00	14
A	1 25.00	2 28.57	0 0.00	9
A+	2 50.00	1 14.29	3 100.00	9
Total	4	7	3	288

Obs	hsgpa	totscore	plgrade	lgrade	pregrade	grade	dgrade
1	81.3	9	C	F	65	1	-64
2	75.7	5	D	F	55	4	-51
3	79.5	7	C	F	65	16	-49
4	75.2	6	D-	F	51	3	-48
5	82.3	7	D+	F	59	13	-46
6	73.0	6	F	F	47	4	-43
7	78.8	7	D	F	55	15	-40
8	71.7	11	D	F	53	16	-37
9	71.8	5	F	F	40	4	-36
10	72.2	6	F	F	40	8	-32
11	76.5	15	C+	F	68	36	-32
12	81.7	6	C	F	64	33	-31
13	77.5	7	D	F	54	25	-29
14	76.5	18	B	F	75	46	-29
15	83.7	15	C	F	66	38	-28

Skipping

270	84.0	17	B-	A	70	86	16
271	73.0	4	F	D	38	55	17
272	84.8	13	C+	A	69	86	17
273	87.5	7	B-	A	71	88	17
274	78.3	7	D	B-	53	70	17
275	76.0	3	F	C-	44	62	18
276	79.0	11	D+	B+	59	77	18
277	87.7	12	B	A+	74	93	19
278	70.8	5	F	C	44	63	19
279	78.7	7	D-	B-	52	71	19
280	78.8	7	D-	B-	52	71	19
281	84.8	12	B	A+	74	94	20
282	73.2	4	F	C-	39	60	21
283	74.8	11	F	B-	48	70	22
284	75.3	5	F	B-	48	70	22
285	69.7	6	F	C-	37	60	23
286	70.0	10	F	C+	43	67	24
287	75.7	13	D	A-	56	82	26
288	82.7	3	D	A	56	87	31

Predict Grade
Final Prediction Equation

The REG Procedure
Model: MODEL1
Dependent Variable: grade Final mark (if any)

Number of Observations Read	1158
Number of Observations Used	603
Number of Observations with Missing Values	555

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	2	79663	39832	201.58	<.0001
Error	600	118556	197.59341		
Corrected Total	602	198219			

Root MSE	14.05679	R-Square	0.4019
Dependent Mean	58.93367	Adj R-Sq	0.3999
Coeff Var	23.85189		

Parameter Estimates

Variable	Label	DF	Parameter Estimate	Standard Error
Intercept	Intercept	1	-69.37989	7.89356
hsgpa	High School GPA	1	1.42465	0.10158
totscore	Total # right on diagnostic test	1	1.59291	0.16913

Parameter Estimates

Variable	Label	DF	t Value	Pr > |t|
Intercept	Intercept	1	-8.79	<.0001
hsgpa	High School GPA	1	14.02	<.0001
totscore	Total # right on diagnostic test	1	9.42	<.0001

$$\widehat{Y} = -69.37989 + 1.42465\ \texttt{hsgpa} + 1.59291\ \texttt{totscore}$$