Class Exercise 2

This exercise is based upon Chapter 2 of the SAS Certification Prep Guide. We will learn how to perform advanced queries using the SQL procedure.

We will use the same data for this class exercise that we used in Class Exercise 1 (Fall 2009 follow-up to ten percent subsample of Fall 2008 admissions). Import the Excel spreadsheet into SAS (name it sample) and then run the following commands.

proc sql feedback;

select \*

from sample;

quit;

Review the SAS log. The feedback option writes the list of columns to the SAS log.

You can limit the number of rows displayed in the output by using the INOBS= option. Let’s limit the number of input rows to 10.

proc sql inobs=10;

select \*

from sample;

quit;

We can check for duplicates using the sort procedure or the DISTINCT keyword in the SELECT clause in PROC SQL. Compare results for approaches below.

proc sort data=sample out=sample2 nodupkey; by degree;

run;

proc sort data=sample out=sample2 nodup; by degree;

run;

proc sql;

select distinct degree, class, cltotgpa, instcd, gender, country, stcd, cntyid, satv, satm, regstat, enroll, ipeds, housing

from sample;

quit;

proc sql;

select distinct degree from sample;

quit;

Let’s subset the data to rows where the degree is B.S. We will keep the columns degree, class, gender, and satm.

proc sql;

select degree, class, gender, satm

from sample

where degree eq 'B.S. ';

quit;

We can change the appearance of the output by adding a title and labels;

proc sql;

title 'Degree, Class, Gender, and SAT Math Score';

select degree label='Degree',

class label='Class',

gender label='Gender',

satm label='Math SAT'

from sample

where degree eq 'B.S.';

quit;

Modify the above code so that all B.S. degrees, not simply those labelled 'B.S.', are captured. You generated a list of such degrees with the **select distinct** syntax you used earlier in this exercise.

For each row in **sample**, we will determine the sum of the math and verbal SAT scores (satv and satm). What happens when the following code is run? Does the code execute?

proc sql;

title;

select degree, class, gender, sum(satv, satm) as satt

from sample

where satt > 1250;

quit;

We will now add the keyword CALCULATED in the where clause. Does the code execute? How were missing values handled?

proc sql;

select degree, class, gender, sum(satv, satm) as satt

from sample

where calculated satt > 1250;

quit;

Modify the above code to identify students from South Carolina with Total SAT greater than 1250. Again modify the code to generate a count of these students.

Let’s group the data by degree, class, and gender and determine the average math SAT score for each group.

proc sql;

select degree, class, gender, avg(satm) as avgsatm

from sample

group by degree, class, gender

order avgsatm;

quit;

Finally, let’s look at an example of a noncorrelated subquery. Be sure to describe the steps in the subquery

proc sql;

select degree, class, gender, avg(satm) as avgsatm

from sample

group by degree, class, gender

having avg(satm) >

(select avg(satm)

from sample)

order avgsatm;

quit;

Gradute students should create a subquery that selects degree, class, gender, and SATV for Continuing students (See column N—REGSTAT) whose SATV is greater than the highest SATV for New Transfer students (See column N—REGSTAT).