Some Simple SAS program files

There are many ways to use SAS. One way is to write your own programs. In SAS three main windows: the program editor (where you write and edit your code), the log window (where the program compiler information is provided when you run the program), and the output window (where the program output is printed). Read up a bit on SAS on one of the webpages listed on the ST640 Introduction to SAS webpage and then try running the simple programs listed below.

1. Two columns of data are entered into a SAS data set named ‘cows’. The data set is printed using proc print (to see if SAS has understood your input statements correctly). Then the second column of data is described using proc univariate.

```sas
data cows;
   input group prod;
   cards;
   1 12.6
   1 14.3
   1 19.2
   2 11.2
   2 12.7
   2 13.1;
   proc print;
      var group prod;
   proc univariate;
      var prod;
   run;
```

2. The data of problem 3.6 in Ott (ST511/512 textbook) is read off of the server’s h: drive. It is described using proc univariate. The ‘plot’ option is added to proc univariate to get some additional plot output. Proc chart is used to get a vertical bar chart (vbar).

```sas
data fluoride;
   infile 'h:\data\st511\ch3_6.dat';
   input fluor;
   proc print;
   proc univariate plot;
   proc chart;
      vbar fluor;
   run;
```

Try rerunning the same program after changing the word ‘chart’ to ‘gchart’. Proc gchart makes plots that look nicer than the character plots of proc chart.

3. This example reads three columns of data. It then creates a new column that is the difference between columns two and three. Proc means is used to calculate some means and standard deviations on the new columns of differences. Proc plot is used to plot column b versus column a. Proc reg is used to fit a line through the data relating columns b to column a (We will study this later).
data ducks;
  input pair a b;
  diff=a-b;
cards;
1 5 7
2 4 3
3 7 9
4 3 5
5 3 2;
proc print;
proc means maxdec=2;
  var diff;
proc plot;
  plot b*a;
proc reg;
  model b=a;
run;

Try rerunning the above program without the maxdec statement. MAXDEC controls the number of digits printed.

You can also try running proc gplot in place of proc plot to get a graphics plot in place of a character plot.

4. An example of Do-loops. Remember that SAS is a programming language. In this example, we use a Do-loop to construct a small table of cut-offs for the t-test.

* This is a comment. It begins with a star and ends with a semi-colon.
This program finds the number such that the absolute value of a random variable having the t-distribution will have probability 0.01, 0.05 and 0.10 of exceeding that number.

Here 'df' is the degrees of freedom and the three variables C01, C05 and C10 refer to the three probability levels.

/* This is another way to make a comment. It is useful if your comment contains some semi-colons: ;;;; */
data ttable;
do df=2 to 30;
  C01=tinv(.005,df);
  C05=tinv(.025,df);
  C10=tinv(.05,df);
  output;
end;
proc print;
  var df C01 C05 C10;
run;