SAS

Most popular Statistical software worldwide.

SAS claims that its products are used at over 40,000 sites, including at 90% of the Fortune 500.

This will not be all SAS as they make other products, such as JMP (a menu and dialogue based stat package)

Huge in the pharmaceutical industry. There is a belief, though 100% FALSE, that the FDA requires the analysis of all clinical trials must be done if SAS.

You can do it in anything, just need to document it as part of your approval submission.

SAS, the company, has the reputation of being a fantastic place to work as well.

Extremely powerful package. You name it, it probably does it. If it doesn't, they are probably working on it.

It is available for many platforms including Windows, Unix, Macintosh, mainframes (z/OS, CMS, VSE, VMS, MVS)

However they don't keep all versions updated at the same rate.

It's a program based package. You need to write a program for your analysis.

These programs will have a block structure, with each block corresponding to a different part of the analysis.

Each block will usually start with a different PROC statement, such as PROC REGRESS, PROC SORT, PROC LOGISTIC, etc.

Within each block, command will be given, options set, etc.

Programming features

Case doesn't matter – PROC, proc, & PrOc all mean the same thing.

All commands must end with a '`;`'. This allows commands to be split across lines.

When commenting your code, you can either do it with

```
* comments ;
```

or

```
/* comments */
```

The following is a program which performs a one way Analysis of variance, does some calculations with the residuals, and then gives some diagnostic plots.

```
* Sample SAS program
* Data set is from Dean and Voss (1999) Design and Analysis of
* Experiments.  Problem 3, page 129.

options linesize=75;   /* set the output width to 75 characters */

data temp;
  infile 'p147.3';
  input brand time;
  invtime=1/time;

* print the data to see if everything is ok
proc print data=temp;
  title 'Margarine Experiment';

* Run the ANOVA
proc glm;
  classes brand;
  model invtime = brand;
  estimate 'marg vs but' brand 1 1 1 -3/divisor=3;
  output out=resids predicted=pred residual=z;
run;

* Switch from data file temp to data file resids
data;
  set resids;

*Standardize the residuals to have standard deviation 1
proc standard std=1.0;
  var z;

* Calculate the normal scores with Blom's adjustment
proc rank normal=blom;
  var z;
  ranks nscores;

proc print;

* Generate diagnostic plots
proc plot;
  plot invtime*brand z*pred z*brand z*nscores;

run;

quit;  /* Ends the program */
```

This program will generate a lot of output. SAS tends to do that.

Also SAS will generate two files when running a program, your results (`.lst`) and a log file (`.log`), describing what commands have been run along with comments and error messages.

Also, the default file extension for programs is `.sas`.

However you can use any file extension you want.

With the results, the default is to return plain text.

It is possible to create your output in other formats, such as HTML (with the `ODS` command)

Obs	brand	time	invtime
1	1	167	.005988024
2	1	171	.005847953
3	1	178	.005617978
4	1	175	.005714286
5	1	184	.005434783
6	1	176	.005681818
7	1	185	.005405405
8	1	172	.005813953
9	1	178	.005617978
10	1	178	.005617978
11	2	231	.004329004
12	2	233	.004291845
13	2	236	.004237288
14	2	252	.003968254
15	2	233	.004291845
16	2	225	.004444444
17	2	241	.004149378
18	2	248	.004032258
19	2	239	.004184100
20	2	248	.004032258
21	3	176	.005681818
22	3	168	.005952381
23	3	171	.005847953
24	3	172	.005813953
25	3	178	.005617978
26	3	176	.005681818
27	3	169	.005917160
28	3	164	.006097561
29	3	169	.005917160
30	3	171	.005847953
31	4	201	.004975124
32	4	199	.005025126
33	4	196	.005102041
34	4	211	.004739336

Obs	brand	time	invtime
35	4	209	.004784689
36	4	223	.004484305
37	4	209	.004784689
38	4	219	.004566210
39	4	212	.004716981
40	4	210	.004761905

The GLM Procedure

Class Level Information

Class	Levels	Values
brand	4	1 2 3 4

Number of observations 40

The GLM Procedure

Dependent Variable: invtime

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	0.00001782	0.00000594	208.33	<.0001
Error	36	0.00000103	0.00000003		
Corrected Total	39	0.00001884			

R-Square	Coeff Var	Root MSE	invtime Mean
0.945537	3.294115	0.000169	0.005125

Source	DF	Type I SS	Mean Square	F Value	Pr > F
brand	3	0.00001782	0.00000594	208.33	<.0001

Source	DF	Type III SS	Mean Square	F Value	Pr > F
brand	3	0.00001782	0.00000594	208.33	<.0001

Parameter	Estimate	Standard Error	t Value	Pr > |t|
marg vs but	0.00044184	0.00006165	7.17	<.0001

Obs	brand	time	invtime	pred	z	nscores
1	1	167	.005988024	.005674016	1.93577	2.15636
2	1	171	.005847953	.005674016	1.07227	0.97574
3	1	178	.005617978	.005674016	-0.34546	-0.48629
4	1	175	.005714286	.005674016	0.24825	0.28402
30	3	171	.005847953	.005837574	0.06399	0.18802
31	4	201	.004975124	.004794041	1.11633	1.08144
32	4	199	.005025126	.004794041	1.42457	1.20084
33	4	196	.005102041	.004794041	1.89873	1.74638
34	4	211	.004739336	.004794041	-0.33723	-0.34951

Obs	brand	time	invtime	pred	z	nscores
35	4	209	.004784689	.004794041	-0.05765	0.00000
36	4	223	.004484305	.004794041	-1.90943	-2.15636
37	4	209	.004784689	.004794041	-0.05765	0.00000
38	4	219	.004566210	.004794041	-1.40451	-1.34037
39	4	212	.004716981	.004794041	-0.47505	-0.63114
40	4	210	.004761905	.004794041	-0.19811	-0.21972

```
                      Margarine Experiment                           47
                                    17:01 Wednesday, December 10, 2003

   Plot of invtime*brand.  Legend: A = 1 obs, B = 2 obs, etc.

invtime |
 0.0065 +
        |
        |
        |
        |                                    A
 0.0060 +  A                                 A
        |                                    B
        |  B                                 C
        |  B                                 B
        |  C                                 A
 0.0055 +
        |  B
        |
        |
        |                                                     A
 0.0050 +                                                     B
        |
        |                                                     C
        |                                                     B
        |                                                     A
 0.0045 +                                                     A
        |                 A
        |                 C
        |                 B
        |                 A
 0.0040 +                 C
        |
        ---+----------------+----------------+----------------+--
           1                2                3                4

                                  brand
```

```
        Plot of z*pred.  Legend: A = 1 obs, B = 2 obs, etc.

  z |
    |
  2 +                                                         A
    |                        A
    |                                                              A
    |        A                A
    |
    |                         A
  1 +                                                         A
    |        A                                                A
    |        B                                                     A
    |                                                                  B
    |        A
    |                                                         A
  0 +        A                B                               A    B
    |                         A                                    A
    |        A                A                               C
    |                         A
    |
    |
 -1 +        B                                                     B
    |
    |        A                A                                    A
    |                                                         A
    |                                                         A
    |                         A
 -2 +
    |
    ---+-------+-------+-------+-------+-------+-------+-------+-------+--
    0.0040  0.0043  0.0045  0.0048  0.0050  0.0053  0.0055  0.0058  0.0060

                                      pred
```

```
    Plot of z*brand.  Legend: A = 1 obs, B = 2 obs, etc.

 z |
   |
 2 +  A
   |                                                        A
   |                                     A
   |                  A                                     A
   |
   |                                                        A
 1 +  A
   |  A               A
   |                  B                  A
   |                                     B
   |                  A
   |  A
 0 +  A               A                  B                  B
   |                                     A                  A
   |  C               A                                     A
   |                                                        A
   |
   |
-1 +                  B                  B
   |
   |                  A                  A                  A
   |  A
   |  A
   |                                                        A
-2 +
   |
   ---+------------------+------------------+------------------+--
      1                  2                  3                  4

                                  brand
```

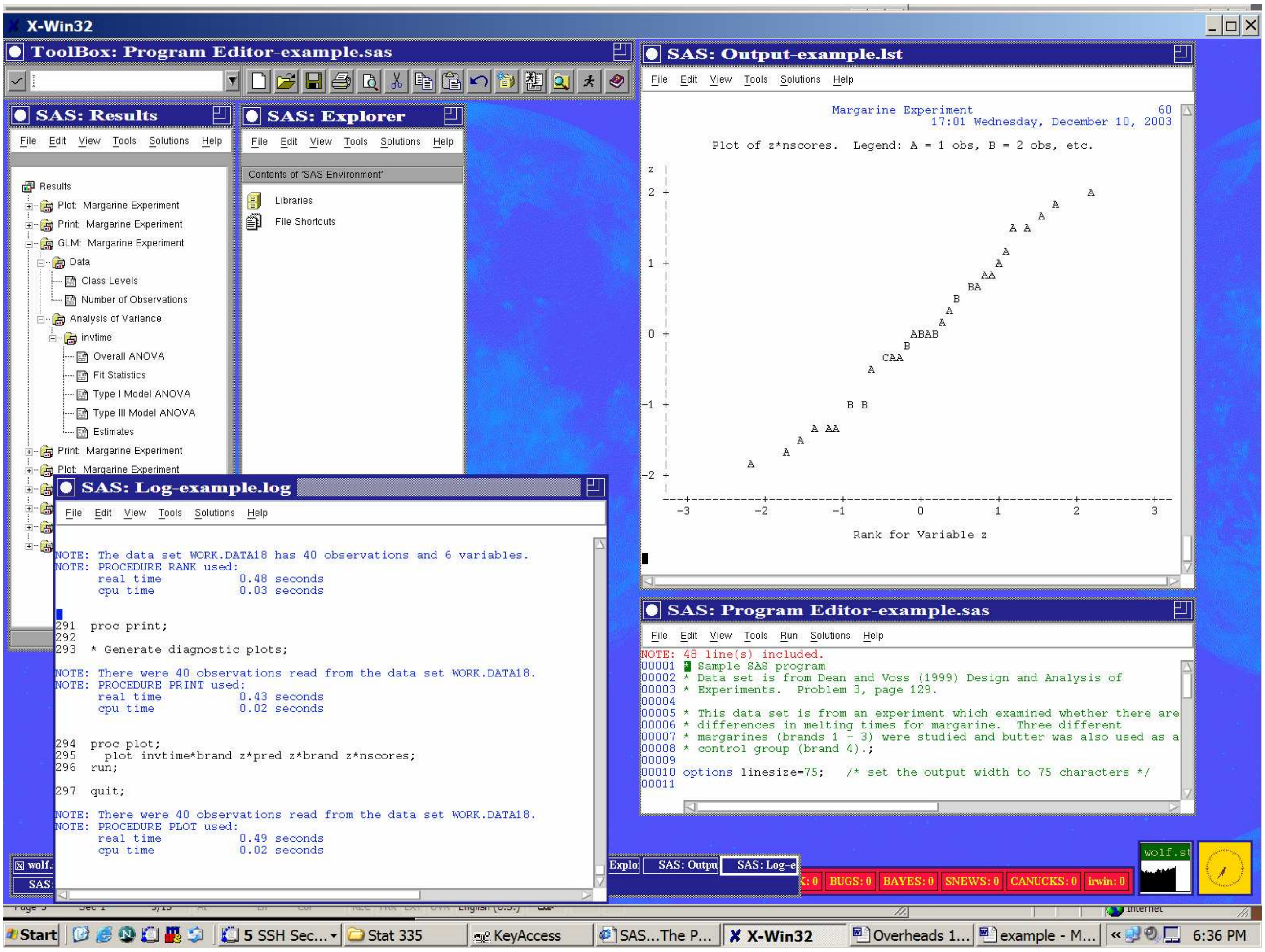
```
                           Margarine Experiment                          50
                                         17:01 Wednesday, December 10, 2003

          Plot of z*nscores.  Legend: A = 1 obs, B = 2 obs, etc.

 z |
   |
 2 +                                                           A
   |                                                      A
   |                                                    A
   |                                                A A
   |
   |                                               A
 1 +                                              A
   |                                            AA
   |                                          BA
   |                                        B
   |                                       A
   |                                      A
 0 +                                  ABAB
   |                                 B
   |                              CAA
   |                            A
   |
   |
-1 +                         B B
   |
   |                    A AA
   |                  A
   |                A
   |           A
-2 +
   |
   ---+----------+----------+----------+----------+----------+----------+--
     -3         -2         -1          0          1          2          3

                              Rank for Variable z
```

The following is part of the log file, just to give you a flavour (The rest is on the web site in example1.log)

```
NOTE: Copyright (c) 1999-2001 by SAS Institute Inc., Cary, NC, USA.
NOTE: SAS (r) Proprietary Software Release 8.2 (TS2M0)
      Licensed to OHIO STATE UNIVERSITY -ACADEMIC TECHNOLOGY SERVICE,
Site 00163
55002.
NOTE: This session is executing on the SunOS 5.8 platform.



This message is contained in the SAS news file, and is presented upon
initialization.  Edit the files "news" in the "misc/base" directory to
display site-specific news and information in the program log.
The command line option "-nonews" will prevent this display.

NOTE: SAS initialization used:
      real time           2:12.18
      cpu time            1.05 seconds

     * Sample SAS program
     * Data set is from Dean and Voss (1999) Design and Analysis of
     * Experiments.  Problem 3, page 129.

     * This data set is from an experiment which examined whether there
are
     * differences in melting times for margarine.  Three different
     * margarines (brands 1 - 3) were studied and butter was also used as
a
     * control group (brand 4).;

     options linesize=75;   /* set the output width to 75 characters */



     data temp;
       infile 'p147.3';
       input brand time;
       invtime=1/time;

     * print the data to see if everything is ok;

NOTE: The infile 'p147.3' is:
      File Name=/home/irwin/SAS/p147.3,
      Owner Name=irwin,Group Name=parstaff,
      Access Permission=rw-------,
      File Size (bytes)=240

NOTE: 40 records were read from the infile 'p147.3'.
      The minimum record length was 5.
      The maximum record length was 5.
NOTE: The data set WORK.TEMP has 40 observations and 3 variables.
NOTE: DATA statement used:
      real time           1.75 seconds
```

```
      cpu time            0.01 seconds


proc print data=temp;
  title 'Margarine Experiment';

* Run the ANOVA;

NOTE: There were 40 observations read from the data set WORK.TEMP.
NOTE: PROCEDURE PRINT used:
      real time           0.61 seconds
      cpu time            0.01 seconds


proc glm;
  classes brand;
  model invtime = brand;
  estimate 'marg vs but' brand 1 1 1 -3/divisor=3;
  output out=resids predicted=pred residual=z;
run;


* Switch from data file temp to data file resids;

NOTE: The data set WORK.RESIDS has 40 observations and 5 variables.
NOTE: PROCEDURE GLM used:
      real time           1.32 seconds
      cpu time            0.09 seconds


data;
  set resids;

* Standardize the residuals to have standard deviation 1;

NOTE: There were 40 observations read from the data set WORK.RESIDS.
NOTE: The data set WORK.DATA13 has 40 observations and 5 variables.
NOTE: DATA statement used:
      real time           0.69 seconds
      cpu time            0.01 seconds
```

When running the program, the default is to open the windowing environment. The following is from a run on a Sun under unix (Solaris 5.8 under Xwindows). Things will look different on different platforms, but the same set of windows should be available.



SAS can also be run in batch mode (at least with unix and other mainframe operating systems).

To do it in unix, give the command

sas program.sas

This will create the files program.lst (output) and program.log (log file)

SAS System Options

There are many options you can use the alter the way that SAS runs the following are useful for setting up the way SAS generates it output. In the previous output I showed you, I used the defaults, except for linesize, which I switched from the default 109.

CENTER | NOCENTER: Output is either centered or left justified

DATE | NODATE: Should today's date appear at the top of each page

NUMBER | NONUMBER: Should the output pages be numbers

LINESIZE = *n*: The maximum width of output lines (range = 64 to 256)

PAGESIZE = *n*: The maximum number of lines per page of output (range = 15 to 32767)

PAGENO = *n*: Starts numbering output pages at zero

YEARCUTOFF = *yyyy*: Specifies the first year in a hundred-year span for interpreting two-digit dates.

So to left justify your output and set the pagesize to 50, you can use the command

```
options nocenter pagesize=50;
```

or you can split it across multiple commands

```
options nocenter;

options pagesize=50;
```

Data Entry

SAS can read in data file from a wide range of formats, text files (space, comma, or tab delimited), Excel, Access, DBF, Lotus 1-2-3, etc.

Two approaches to reading data, `DATA` step, or `PROC IMPORT`.

`DATA` step:

Can be used for a wide range of text files. Often used with space delimited files, but can be used with fixed format files and files with other delimiters. The example earlier was with a space delimited file.

Also I believe that variable names cannot be included in the file. They are given with the `input` part of the `DATA` step

Now suppose that the data set for the example had brand in columns 1 – 3 and time was in columns 4 - 8. The data could also have been read in with

```
data temp;
  infile 'p147.3';
  input brand 1-3 time 4-8;
  invtime=1/time;
```

You can also a combination of this. For example

```
data temp;
  infile 'p147.3';
  input brand 1-3 time;
  invtime=1/time;
```

would get brand from columns 1-3, and then start looking for time in column 4, but with no restriction on where it ends.

To use a different delimiter, say a comma, modify your `infile` command like

```
  infile ‘p147.3’ dlm = ‘,’
```

For tab delimited files, use `dlm =’09’x` (the tab character is character 9 in ASCII). The `x` indicates that the character is being indicated with a hex code.

In the input line, string variables must be indicated by a `$`. For example

```
  input country $ GPA government $
```

would make `country` and `government` string variables and `GPA` numeric.

`PROC IMPORT:`

This is the approach for reading in non text files and text files with the variable names in the first row. An example is as follows

```
proc import datafile = 'p1473.txt'
out = temp dbms = tab replace;
run;

data;
  set temp;
  invtime=1/time;
run;
```

In this set of commands, the file `p1473.txt` is read in, a data set `temp` is created. The `dbms` indicates that a tab delimited text file is being read in. The `replace` option says to overwrite the data set temp if it exists.

Usefile options for `dbms` are `dlm` (space delimited), `cvs` (comma delimited), `excel`, `dbf` (dBase).

The `dbms` option is not needed if the datafile name has the standard file extension in the name.

The `data` step after it creates a new data file, using a default name (probably `data1`), reading in the data set `temp` created in `proc import` and adds the variable `invtime`.

Writing out data files

This can either be done with a DATA step or PROC export.

`DATA` step:

This is one approach to writing out your data as a text file. An example is the following

```
data _NULL_;
  set finaldata;
  file 'DV.dat';
  put brand time invtime pred z nscores;
run;
```

This will write the variables into the file `DV.dat`, with each variable separated by a space. Other delimiters can be use the `dlm` option.

The DATA step can also be to create a SAS binary data file.

For example the command

```
data 'DV2';
  set finaldata;
run;
```

creates a data file DV2.sas7bdat. The quotes indicate that this should be a permanent file and not a temporary data set. It can be accessed with

```
data test2;
  set 'DV2';
  drop nscores;
```

The `drop nscores` piece says to drop that variable from the data set.

keep brand time invtime

would only keep these three variables and drop the rest.

`PROC EXPORT`:

It's basically the opposite of PROC IMPORT, taking most of the same options. The main differences are `outfile` is used instead of `datafile` and `data` is used instead of `out`.

```
proc import outfile = 'p1473.txt'

data = finaldata dbms = tab;
```

Most of the time, something like this will be used.

Useful PROCS

`UNIVARIATE` (BASE): Univariate summary statistics

`CORR` (BASE): Correlations

`MEANS` (BASE): Similar to Univariate, but adds in some simple testing as well.

`FREQ` (BASE): Creates and analyzes contingency tables

`TABULATE` (BASE): Descriptive statistics in tabular format

`TTEST` (STAT): 1 and 2 sample t tests

`ANOVA` (STAT): Analysis of variance for balanced designs

`REG` (STAT): Regression

`GLM` (STAT): General linear model. Similar to `lm()` in S-Plus/R,

`LOGISTIC` (STAT): Logistic Regression

Common struction for SAS PROCS

```
PROC whatever <options>;
command1;
command2;
…
run;
```

Options will usually include `DATA=`. Also common are options of the form `OUT=`. Note the actual name can change and there may be more than one.

Common commands for many PROC are `by`, `class`, `freq`, `output`, `var`, and `weight`. For example, here are the possible commands that are part of `PROC UNIVARIATE`:

```
PROC UNIVARIATE <option(s)>;
  BY <DESCENDING> variable-1
  <...<DESCENDING> variable-n>
  <NOTSORTED>;
  CLASS variable-1<(variable-option(s))>
  <variable-2<(variable-option(s))>>
  </ KEYLEVEL='value1'|('value1' 'value2')>;
  FREQ variable;
  HISTOGRAM <variable(s)> </ option(s)>;
  ID variable(s);
  INSET <keyword(s) DATA=SAS-data-set> </
```

```
option(s)>;
OUTPUT <OUT=SAS-data-set> statistic-
keyword-1=name(s)
<... statistic-keyword-n=name(s)>
<percentiles-specification>;
PROBPLOT <variable(s)> </ option(s)>;
QQPLOT <variable(s)> </ option(s)>;
VAR variable(s);
WEIGHT variable;
```

The number of options for this PROC is huge. Go to the documentation on the web to see them all.

PROC UNIVARIATE example:

```
options linesize=75;

DATA swiss;
  infile '/home/irwin/SAS/swiss.dat';
  input fertility agriculture examination
    education catholic infantmort;
  title 'Swiss Data Set';

proc univariate data = swiss
  normal
  cibasic
  plots
  alpha = 0.01;
  var education;
run;
```

And the output is

```
                              Swiss Data Set                                    1
                                        13:26 Thursday, December 11, 2003

                          The UNIVARIATE Procedure
                            Variable:  education

                                   Moments

N                                47    Sum Weights                 47
Mean                     10.9787234    Sum Observations           516
Std Deviation            9.61540739    Variance            92.4560592
Skewness                 2.42076401    Kurtosis            7.43408377
Uncorrected SS                 9918    Corrected SS        4252.97872
Coeff Variation          87.5821991    Std Error Mean      1.40255132


                         Basic Statistical Measures

            Location                    Variability

    Mean     10.97872     Std Deviation            9.61541
    Median    8.00000     Variance                92.45606
    Mode      7.00000     Range                   52.00000
                          Interquartile Range      6.00000


                Basic Confidence Limits Assuming Normality

     Parameter          Estimate      99% Confidence Limits

     Mean               10.97872       7.21005    14.74740
     Std Deviation       9.61541       7.55881    13.03220
     Variance           92.45606      57.13564   169.83834


                         Tests for Location: Mu0=0

       Test           -Statistic-    -----p Value------

       Student's t    t   7.82768    Pr > |t|    <.0001
       Sign           M      23.5    Pr >= |M|   <.0001
       Signed Rank    S       564    Pr >= |S|   <.0001


                            Tests for Normality

  Test                  --Statistic---    -----p Value------

  Shapiro-Wilk          W     0.748203    Pr < W      <0.0001
  Kolmogorov-Smirnov    D     0.246538    Pr > D      <0.0100
  Cramer-von Mises      W-Sq  0.656644    Pr > W-Sq   <0.0050
  Anderson-Darling      A-Sq  3.587264    Pr > A-Sq   <0.0050

                         Quantiles (Definition 5)

                          Quantile      Estimate

                          100% Max            53
                          99%                 53
```

```
                    The UNIVARIATE Procedure
                      Variable:  education

                    Quantiles (Definition 5)

                     Quantile      Estimate

                     95%                 29
                     90%                 28
                     75% Q3              12
                     50% Median           8
                     25% Q1               6
                     10%                  3
                     5%                   2
                     1%                   1
                     0% Min               1


                      Extreme Observations

              ----Lowest----        ----Highest---

              Value      Obs        Value      Obs

                  1       25           28       18
                  2       33           29       46
                  2       31           29       47
                  2       16           32       42
                  3       37           53       45

        Stem Leaf                     #             Boxplot
           5 3                        1                *
           4
           4
           3
           3 2                        1                *
           2 899                      3                0
           2 0                        1                |
           1 59                       2                |
           1 00122222333             11             +--+--+
           0 55666677777778888999    20             *-----*
           0 12223333                 8                |
             ----+----+----+----+
         Multiply Stem.Leaf by 10**+1
```

```
                   Swiss Data Set                             3
                               13:26 Thursday, December 11, 2003

              The UNIVARIATE Procedure
                Variable:  education

                   Normal Probability Plot
52.5+                                               *
    |
    |
    |                                                  +
    |                                           * +++++
27.5+                                      **+++++
    |                                   ++*++
    |                              +++++**
    |                        ++++** * *
    |                ** ****** **
 2.5+   *   *   ** *++++
     +----+----+----+----+----+----+----+----+----+----+
         -2        -1         0        +1        +2
```

PROC GLM:

Has a similar function to lm() in S-Plus/R.

PROC GLM < options > **;**
  CLASS variables **;**
  MODEL dependents=independents < / options > **;**
  ABSORB variables **;**
  BY variables **;**
  FREQ variable **;**
  ID variables **;**
  WEIGHT variable **;**
  CONTRAST 'label' effect values < ... effect values > < / options > **;**
  ESTIMATE 'label' effect values < ... effect values > < / options > **;**
  LSMEANS effects < / options > **;**
  MANOVA < test-options >< / detail-options > **;**
  MEANS effects < / options > **;**
  OUTPUT < **OUT=**SAS-data-set >
    keyword=names < ... keyword=names > < / option > **;**
  RANDOM effects < / options > **;**
  REPEATED factor-specification < / options > **;**
  TEST < **H=**effects > **E=**effect < / options > **;**

```
options linesize=75;

proc import datafile = '93cars.txt'
   out = cars;
run;

data cars93;
  set cars;
  highfuel = 100/highmpg;
  cityfuel = 100/citympg;
run;

proc glm data = cars93;
  classes domestic;
  model highfuel = weight domestic
     weight*domestic / solution clm;
  output out=resids predicted=pred
     residual=resids;
run;
```

In the `model` statement, the structure is similar to S-Plus/R, in that interactions are denoted with *. However when an interaction term is included, the main effect are not automatically included. They must be explicitly added as I did above.

Also it takes a wide range of options. For example you must explicitly ask for the parameter estimates.

The classes statement indicates which variables in the model should be treated as categorical factors. The default is to treat as variables in the model as numeric.

When fitting models with categorical variables, SAS deals with the parametrization and contrasts a bit differently. It actually fits an over parametrized model and uses and generalized inverse instead of a regular inverse, since its $X^T X$ matrix is singular.

There is a wide range of summary statistics that can be output. They include, the fits, residuals, leverages, Cook's D, prediction intervals and confidence intervals, DFFITS, etc.

The GLM Procedure

Class Level Information

Class	Levels	Values
Domestic	2	0 1

Number of observations 93

The GLM Procedure

Dependent Variable: highfuel

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	3	22.51693938	7.50564646	69.45	<.0001
Error	89	9.61827575	0.10807051		
Corrected Total	92	32.13521513			

R-Square	Coeff Var	Root MSE	highfuel Mean
0.700694	9.282533	0.328741	3.541499

Source	DF	Type I SS	Mean Square	F Value	Pr > F
Weight	1	22.00269643	22.00269643	203.60	<.0001
Domestic	1	0.02634815	0.02634815	0.24	0.6227
Weight*Domestic	1	0.48789481	0.48789481	4.51	0.0364

Source	DF	Type III SS	Mean Square	F Value	Pr > F
Weight	1	21.19245842	21.19245842	196.10	<.0001
Domestic	1	0.42981420	0.42981420	3.98	0.0492
Weight*Domestic	1	0.48789481	0.48789481	4.51	0.0364

Parameter		Estimate		Standard Error	t Value	Pr > |t|
Intercept		1.368412490	B	0.27519967	4.97	<.0001
Weight		0.000706864	B	0.00008484	8.33	<.0001
Domestic	0	-0.742054400	B	0.37209083	-1.99	0.0492
Domestic	1	0.000000000	B	.	.	.
Weight*Domestic	0	0.000252874	B	0.00011901	2.12	0.0364
Weight*Domestic	1	0.000000000	B	.	.	.

NOTE: The X'X matrix has been found to be singular, and a generalized inverse was used to solve the normal equations. Terms whose estimates are followed by the letter 'B' are not uniquely estimable.

The GLM Procedure

Observation	Observed	Predicted	Residual
1	3.22580645	3.22245051	0.00335595
2	4.00000000	4.04302685	-0.04302685
3	3.84615385	3.86547524	-0.01932140
4	3.84615385	3.89426740	-0.04811355
5	3.33333333	4.11980592	-0.78647259
6	3.22580645	3.40418148	-0.17837502
57	4.00000000	3.40480080	0.59519920
58	3.44827586	3.42879426	0.01948160
59	4.00000000	4.00943601	-0.00943601
60	3.84615385	3.10022986	0.74592399

Observation	95% Confidence Limits for Mean Predicted Value	
1	3.11742228	3.32747873
2	3.90169213	4.18436157
3	3.74451763	3.98643286
4	3.77029337	4.01824142
5	3.96857802	4.27103383
6	3.29594990	3.51241305
57	3.30711142	3.50249018
58	3.33135039	3.52623814
59	3.87225027	4.14662174
60	2.94315247	3.25730724

Observation	Observed	Predicted	Residual
61	3.84615385	3.92019237	-0.07403852
62	3.03030303	2.82895775	0.20134528
63	4.16666667	4.20618238	-0.03951572
91	4.00000000	3.32322304	0.67677696
92	3.57142857	3.49117726	0.08025131
93	3.57142857	3.74070925	-0.16928068

Observation	95% Confidence Limits for Mean Predicted Value	
61	3.80282367	4.03756106
62	2.68401721	2.97389830
63	4.04325020	4.36911457
91	3.22340681	3.42303927
92	3.39354703	3.58880749
93	3.63115860	3.85025990

```
                    The GLM Procedure

Sum of Residuals                              -0.00000000
Sum of Squared Residuals                       9.61827575
Sum of Squared Residuals - Error SS           -0.00000000
PRESS Statistic                               10.56191586
First Order Autocorrelation                    0.07667443
Durbin-Watson D                                1.84367064
```

Here is the R output for the same model

```
> anova(weight.domestic.int.lm)
Analysis of Variance Table

Response: Highfuel
                Df  Sum Sq Mean Sq  F value  Pr(>F)
Weight           1 22.0027 22.0027 203.5957 < 2e-16 ***
Domestic         1  0.0263  0.0263   0.2438 0.62269
Weight:Domestic  1  0.4879  0.4879   4.5146 0.03638 *
Residuals       89  9.6183  0.1081

> summary(weight.domestic.int.lm)

Call:
lm(formula = Highfuel ~ Weight * Domestic, data = cars93)

Residuals:

     Min       1Q   Median       3Q      Max
-0.78647 -0.21346 -0.03952  0.17163  0.99145

Coefficients:

                       Estimate Std. Error t value Pr(>|t|)
(Intercept)           6.264e-01  2.504e-01   2.501   0.0142 *
Weight                9.597e-04  8.347e-05  11.498   <2e-16***
DomesticDomestic      7.421e-01  3.721e-01   1.994   0.0492 *
Weight:DomesticDomes -2.529e-04  1.190e-04  -2.125   0.0364 *
```

Notice that the summaries are similar. Note that R generates the Type I (Sequential SS)