



New York State Agricultural Experiment Station
Geneva, New York

Computer Centre: ref 420(replaces 328) / 5-Sep-07

SAS 9.1

Version 9.1 of SAS is available on hostname `stat.nysaes.cornell.edu`. To use SAS, a username must be obtained from the Computer Centre. Any SSH client can be used for network connexion (e.g. **Terminal** on a Macintosh, **PUTTY** on Windows).

Using SAS

Non-interactive usage from UNIX

If a SAS program has been set up in a file by means of a text editor (either directly on `stat` or transferred to `stat` after preparation on a desktop), it may be executed by typing

```
sas filename
```

It is recommended that the program name be of the form *filename.sas*. SAS usually produces two output files: *filename.lst* will contain the output from the SAS procedures, and *filename.log* will contain a log of the run including a record of any errors that occurred.

When preparing material on a desktop, don't forget to save the file as text, and transfer it as text.

Interactive usage - line mode

Type

```
sas
```

and SAS will respond with a line number followed by a ? prompt.

Using data files with SAS

Input files

SAS accepts standard text files as data. To read a data file, simply precede a SAS **input** statement with an **infile** statement. For example, to read fields *variety*, *block*, and *yield* from the file *spraydata1985*, one could use the following data step.

```
data;  
  infile 'spraydata1985';  
  input variety block yield;
```

A general pathname can also be used:

```
data;  
  infile '~jb/sastest/spraydata1985';  
  input variety block yield;
```

If data in the file is tab-delimited, for example it was prepared on a Macintosh with Excel and then saved as text, use `expandtabs` on the `infile` statement

```
infile 'spraydata1985' expandtabs;
```

SAS datasets

Datasets generated by SAS last only for the duration of the session unless specific steps are taken. To make the dataset in the above example into a permanent file in the current directory, use the sequence

```
libname myset '.';
data myset.t;
  infile 'spraydata1985';
  input variety block yield;
```

In this example, the symbol `myset` is used in a two part file specification and results in the dataset being stored in the current directory as *t.sas7bdat*.

To access the dataset in a subsequent SAS session, use the similar arrangement

```
libname myset '.';
proc print data=myset.t;
```

Output files

The following reads data from *spraydata1985* and copies it into *spraydata1985copy*.

```
data;
  infile 'spraydata1985';
  input variety block yield;
  file 'spraydata1985copy';
  put variety block yield;
```