

# QUEEN MARY, UNIVERSITY OF LONDON

MAS 314

Design of Experiments

Practical 3

25 January 2007

---

In this practical we shall concentrate on ways of looking at data and ways of declaring factors. We shall do this using the scabbiness data from Assignment 2.

**1 (Scabbiness Data)** The data are in the file `scabby.dat`. To load this data:

Data → Load → ASCII file

Click on **Browse ...** and find the file `DOETUTOR/STU/SCABBY.DAT`. It contains 2 columns of 32 values. The first column is treatment coded 1–7 according to the following scheme:

treatment	1	2	3	4	5	6	7
amount of sulphur	0	300	600	1200	300	600	1200
timing	N/A	autumn	autumn	autumn	spring	spring	spring

The second column is an index of scabbiness of the potatoes grown in the 32 plots. While loading, choose to name the two columns `tmt` and `scabs` respectively.

Notice some bad practice in the data file. There is no column (or subset of columns) which enables you to uniquely identify the each plot *without* knowing the treatment on it.

**2 (Reading factors)** One of Genstat's strengths is its distinction between variates and factors. However, sometimes it is hard work to get this right.

When Genstat 7.2 loads data from a file it assumes that each column of numbers is a variate and every other column contains text. In the data in `scabby.dat` the treatments are coded 1, ..., 7, so Genstat assumes that `tmt` is a variate. How can it be changed into a factor? Each of the following methods should work.

**Method 1** If you have the data in a spreadsheet, right-click on the appropriate column and choose `Convert to Factor`.

**Method 2** Using the mouse,

Spread → Calculate → Factor

Then choose/type

**Data:**   
**Save In:**

**Method 3** Create a factor called `tmtfac` with the same levels as `tmt` by typing the lines

```
factor [levels=7] tmtfac  
calculate tmtfac=tmt
```

in the **Input Window** and submitting them.

**3 (Looking at the data so far)** It is often useful to look at all the variates and factors that have been defined so far, so that you can check for silly mistakes. Here are two methods. Try them both and decide which one works best for you.

**Method 1** Type

```
print tmt, scabs
```

in the **Input Window**, submit the line and view the result in the **Output Window**. After you have extra variates or factors to look at, do the same thing again, looking at the new ones side-by-side with the old (you can edit the command and submit it again).

**Method 2** Using the mouse:

```
Spread → New → Data in Genstat
```

then choose to put `tmt` and `scabs` into the spreadsheet and look at the spreadsheet. After you have extra variates or factors to look at, you will need to add them to the spreadsheet before you can look at them. You do this with

```
Spread → Add → Data in Genstat
```

**4 (Analysis of scabbiness data)** Get an overview of the data drawing a boxplot of the data for each treatment. Then analyse the data and check that the output agrees with what was done by hand in Assignment 2.

**5 (Creating a new factor from an old one)** In what follows, I assume that `tmt` is factor. If you used Method 3 above then replace `tmt` by `tmtfac`.

The description of the scabbiness data on page 1 shows that the factor `tmt` can be used to define three new factors: `sulphur`, `timing` and `control`. The last one is a 2-level factor that has level 1 on the control treatment (`tmt = 1`) and level 2 otherwise (`tmt = 2, ..., 7`).

**Method 1** We will create the factor `control` by using the mouse.

Data → Calculations

Tick the box to make factors available. Click on `Functions ...` and then choose/type

<b>Function class:</b>	<code>Transformations</code>
<b>Function:</b>	<code>Convert factor</code>
<b>Factor:</b>	<code>tmt</code>
<b>Values for new levels:</b>	<code>!(1, 2, 2, 2, 2, 2, 2)</code>

Click on `OK`, notice what Genstat has written in the calculation box, and tell it to save the result in `control`. After clicking on `OK` again, look at the values of `control`. If necessary, convert it to a factor.

Notice that the `NEWLEVELS` function has converted values in `tmt` into values in `control` according to the following rule:

<code>tmt</code>	1	2	3	4	5	6	7
<code>control</code>	1	2	2	2	2	2	2

**Method 2** If you make the new factor by typing in commands, the general rule is to declare the factor first, with its number of levels, and then calculate its values. An advantage is that you can declare names for the levels of the factor (these are called LABELS) rather than levels. Create the factor `timing` by typing the lines

```
factor [labels = !T(na, autumn, spring)] timing
calculate timing = newlevels(tmt; !(1, 2, 2, 2, 3, 3, 3))
```

in the **Input Window** and submitting them. Then inspect the values of `timing`. Do you need to convert it to a factor?

In the spreadsheet, you can edit column attributes to change the values of a factor to more meaningful labels (these must be text) or levels (these must be numbers). Try it.

**Method 3** We will create the factor `sulphur` by copying `tmt` and editing its levels. Select the `tmt` column by left-clicking. Then

Spread → Factor → Recode

Call the new factor `sulphur` and edit its levels appropriately. Now check that all three new factors have the levels that you expect.

**6 (Checking factors)** Click on the `timing` factor in the spreadsheet. Then

Spread → Factor → Edit Levels and Labels

Experiment with changing the levels or the labels. What does the column headed 'Counts' show? (In a recent practical, one student was able to use this feature to find out what mistake she had made in entering her data.) Do the same thing with the other factors.

**7 (Further analysis)** Now that you have created the extra factors, analyse the scabby data again using various different inputs to `Treatment Structure`:

```
tmt
control
control/tmt
```

By comparing the outputs, see if you can work out what Genstat is doing in each case.

Save the spreadsheet with all these extra factors, so that you can use it again later.

**8 (Factors with values in standard order)** Factors which describe the experimental units usually come with their values in standard order. In Practical 2 we did this by inserting a new column into the spreadsheet, right-clicking on it and choosing `Fill`. Alternatively, they can be declared using the command `GENERATE`. Use `Help` to find out about `GENERATE` and write details below. You may need to know that Genstat calls its commands `DIRECTIVES`.