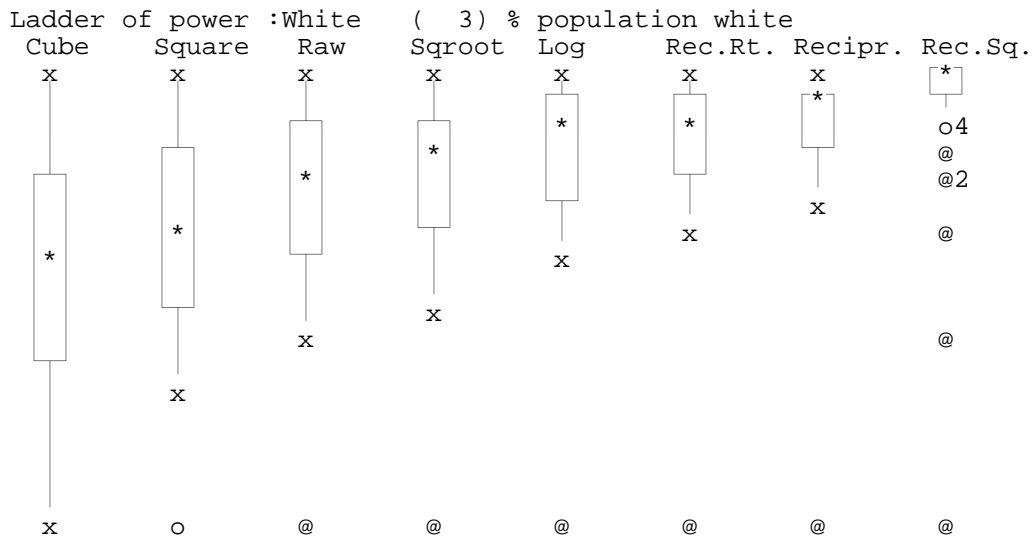
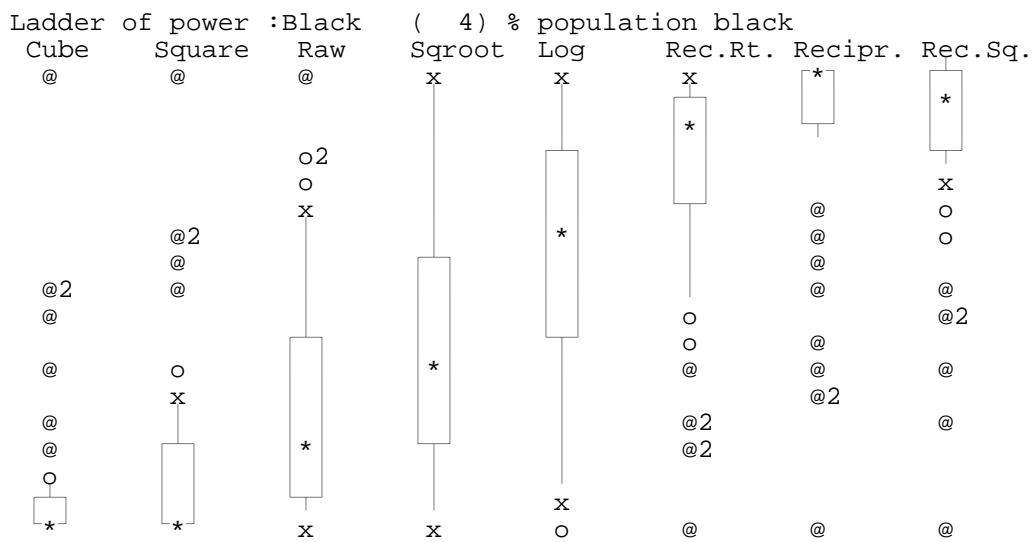


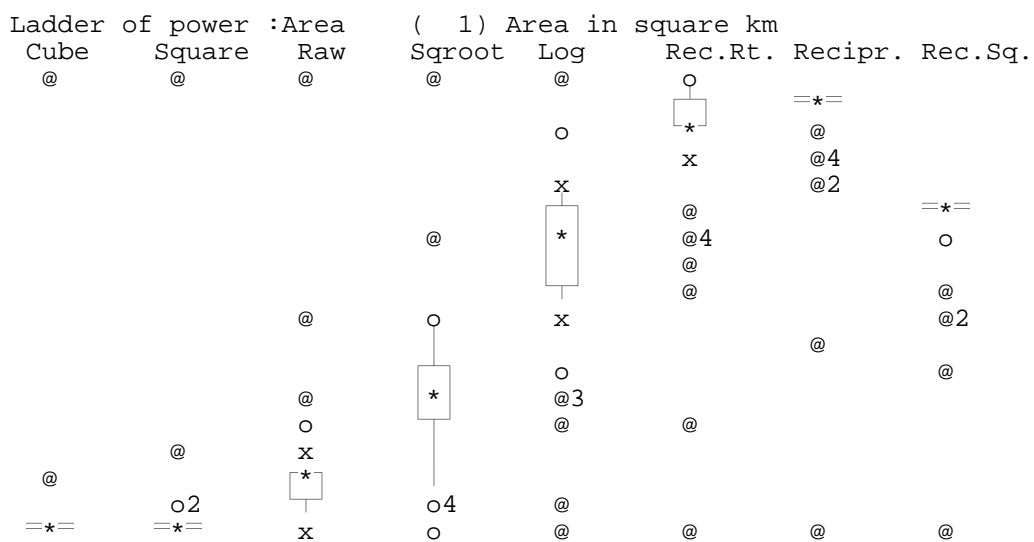
Ex. 5



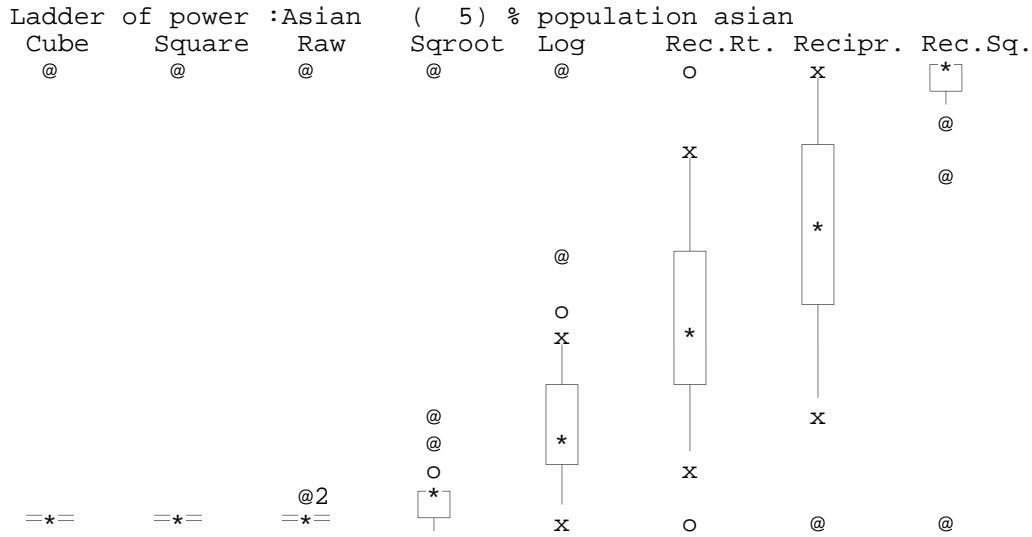
Ex. 6



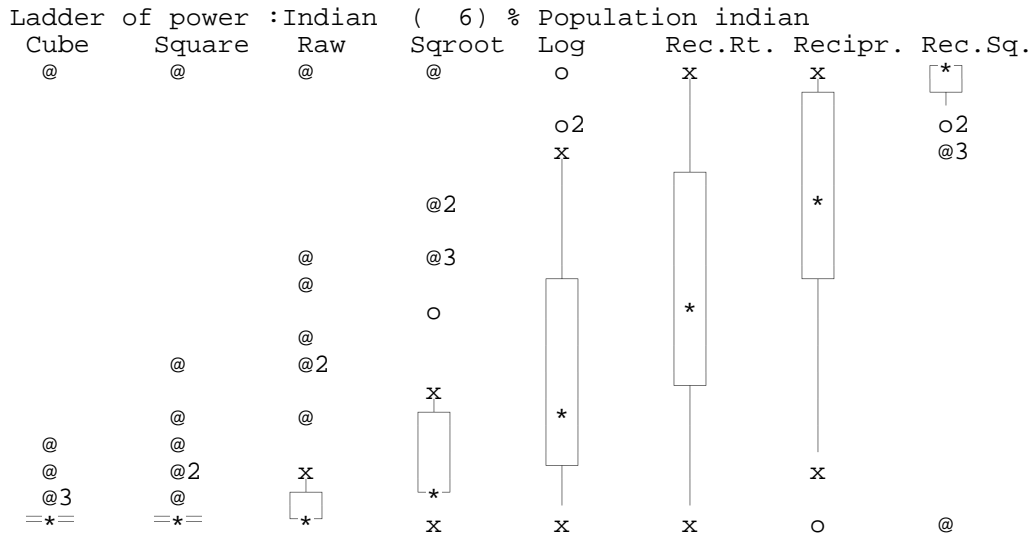
Ex. 7



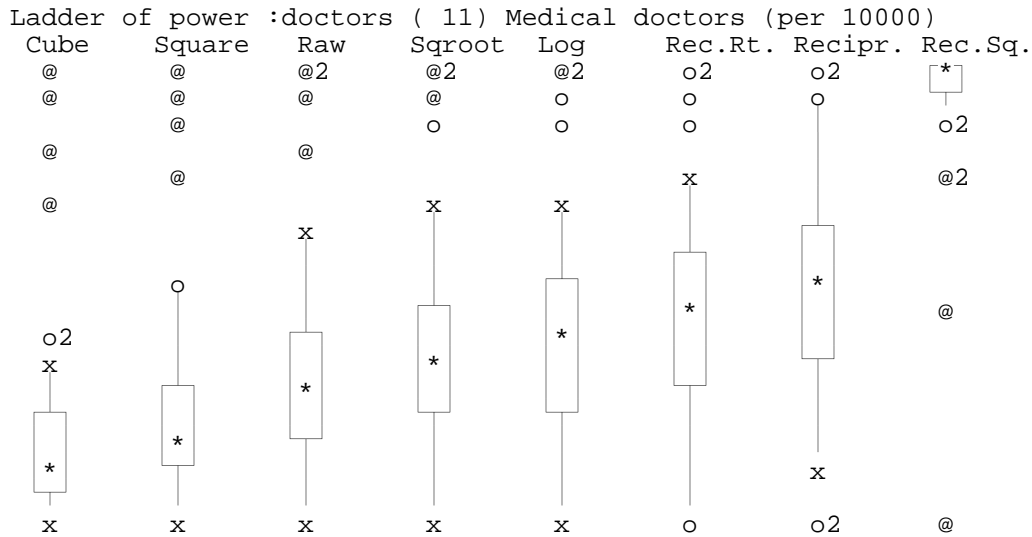
Ex. 8

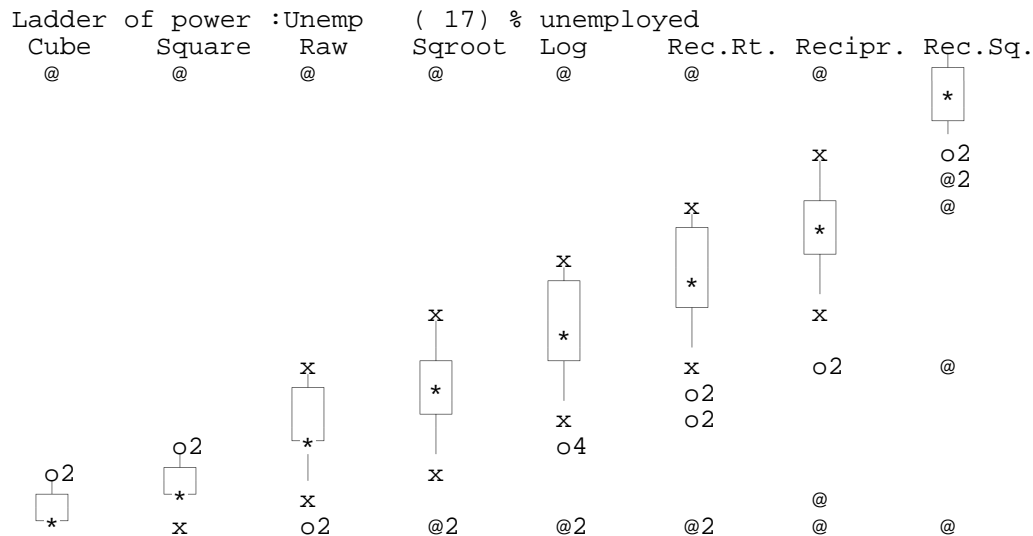


Ex. 9



Ex.10





Re-expression commands

The BOXPLOT command has a LADDER option showing Tukey's ladder of powers, as in

```
>BOXPLOT 1 LADDER
```

Note however that this produces only a display showing how reexpressions would affect the variable, i.e. this is not a command actually re-expressing the variable.

Standard expressions let you perform operations like

```
>LET #10=LOG(#10)
```

The transformation language is powerful enough to provide illimited possibilities. For the most common transformation and reexpression tasks, EDA however offers many specific commands, as well as a specialized module REEXPRES, simplifying considerably your work proposing a re-expression centered environment and taken care of some specific problems (like logs for negative numbers³).

Commands like

```
>STANDARDIZE 1
```

i.e. variable 1 is standardized (by default: remove median and divide by midspread; you find other commands for adding up variables, creating dummy variables, recoding some variable and so forth.

The *REEXPRES* command has been designed as an interactive tool to hunt for interesting and useful transformations before you actually transform variables in the WA. ("Play module")

REEXPRES is a special module with its own syntax. Basically you enter the module with a variable; it will then show a boxplot, a density line as well as the 5-number summary for the variable; if you type a reexpression command you will see the same summaries for the variable after the transformation. Both sets of summaries will remain on the screen, i.e. you will be able to compare them.

Here's an example of what you might see at some stage...

3. The LET command would simply tell you that logs of negative numbers are impossible and aborts; REEXPRES will add automatically a suitable constant and tell you what it did.

```

Reexpress InfMor(10) Infant Mortality
Current reexpression:SquareRoot
O: 2.00 13.30
H: 3.61 9.25
M: 6.16
139* 61827231*1335 1412225411 63 3253413212124 42213425341 421 2 1 1

```

```

Reexpress InfMor(10) Infant Mortality
Current reexpression:Log
O: 1.39 5.18
H: 2.57 4.45
M: 3.64
1 3 9 * 6 1 8 2 43 2313711234111332 2551 612328442226 534684431211

```

This example shows a screen, when you asked for a log-re-expression after having done a square-root transformation on the previous step.

Within REEXPRESS you will find a number of commands (type HELP within REEXPRESS for details); here the remaining examples only deal with Tukey-Style power transformations (Ladder of power); the image of the ladder provides a convenient way of thinking about successive steps of re-expression to achieve more symmetry.

Here's a commented sequence of commands, illustrating the use of some REEXPRESS commands.

```

>REEXPRESS 1 ! Enter the REEXPRESS module (prompt changes)
Reexpress:UP ! Go one step up the ladder (here: Square up from Raw)
Reexpress:UP ! Go one step further up (here: Cube from Square)
Reexpress:DOWN ! Back down one step (back to square)
Reexpress:LADDER ! Show the ladder
Reexpress:LOG ! You may also enter directly the transformation
Reexpress:APPLY ! Ok. Replace the variable in the WA and quit.
> ! Back to EDA (normal mode)

```

The second command sequence illustrates the use of REEXPRESS with several variables, i.e. going the first to the last

```

>REEXPRESS 1,5,9,10
Reexpress:UP ! Here we work on variable of var 1
Reexpress:APPLY ! Apply reexpression to it
Reexpress:DOWN ! Next variable (5)
Reexpress:DOWN
Reexpress:NEXT ! Next variable (9), 5 was left unchanged
Reexpress:STANDARDIZE
Reexpress:LADDER
Reexpress:Q ! Quit now; 9 unchanged; forget about 10
> ! Back to normal EDA mode.

```

The TRACES command offers a number of options for transforming each group. Here's a commented command sequence

```

>TRACES 5 REMOVEMEDIAN ! Remove groupe medians and show
>TRACES 5 STANDARDIZE ! Standardize groups and show
>TRACES 5 REEXPRESS=2 ! Reexpress, using square.

```

Note that for the power transformations, you have to indicated directly the requested power, i.e. in this example we are using the square.⁴

If you need to transform a relationship for *linearity*, you will need to go back to PI (Plot inspect).

```

pi:P XUP Y
pi:P XUP YDOWN
pi:P XLOG YSQRT

```

4. In fact you can use any power; REEXPRESS=0 commands a log transformation.