SAS Built-in Functions

Mathematical Functions

\[ \text{Log} (x) \quad \text{ln}(x) \]
\[ \text{Exp} (x) \quad e^x \]
\[ \text{SQRT} (x) \quad \sqrt{x} \]
\[ \text{ABS} (x) \quad \text{absolute value of } x \]
\[ \text{CEIL} (x) \quad \text{smallest integer } \geq x \]
\[ \text{FLOOR} (x) \quad \text{largest integer } \leq x \]
\[ \text{INT} (x) \quad \text{integer part of } x \]

\[ \text{ROUND}(x, \text{precision}) \quad \text{rounding} \]
\[ \text{ROUND}(x, .1) \]
\[ \text{ROUND}(x, .01) \]
\[ \text{ROUND}(x, 1) \]
\[ \text{ROUND}(x, 5) \]
\[ \text{MAX (variable list)} \quad \text{must be at least 2 variables} \]

\[ \text{MAX (x, y, z)} \]

\[ \text{MIN (variable list)} \]

\[ \text{MIN (A, B, C)} \]

\[ \text{MOD (x, y)} \quad \text{modulus arithmetic} \]

\[
\begin{array}{c|c|c|c}
  \text{x} & \text{y} & \text{R} \\
\end{array}
\]

\[
\begin{array}{c|c|c|c}
  \text{y} & \text{is the divisor} \\
  \text{x} & \text{is the dividend} \\
  \text{R} & \text{is the remainder} \\
\end{array}
\]

\[ \text{mod(x, y) same as x mod y in textbooks} \]
\[
\sin(x) \quad \text{X in radians}
\]
\[
\cos(x)
\]
\[
\tan(x)
\]
\[
\log_{10}(x) \quad \text{log in base 10}
\]
\[
\log_2(x) \quad \text{log in base 2}
\]
\[
\arcsin(x) \quad \text{arcsin}(x)
\]
\[
\arccos(x)
\]
\[
\arctan(x)
\]

Note: Automatic SAS system variable -N- is incremented by 1 each time the data step is executed.

Caution: p.101
Statistical Functions

RAND ('dist', parm1, ..., parmk)

will generate a random number from the specified distribution. The 'dist' must be enclosed in quotes. Valid distributions include: 'BERNOULLI'
'BETA'
'BINOMIAL'
'CHISQUARE'
'EXPONENTIAL'
'F'
The parameters parml, \( \theta \) and others

The parameters parml, \( \theta \) are shape, location, or scale parameters appropriate for the specified distribution.
EX:  \[ b1 = \text{RAND}('\text{BINOMIAL}', .5, 20); \]
\[ x = \text{RAND}('\text{CHISQUARE}', 6); \]
\[ y = \text{RAND}('F', \text{ndf}, \text{ddf}); \]
\[ y = \text{RAND}('\text{GAMMA}', a); \]
\[ z = \text{RAND}('\text{NORMAL}'); \]
\[ x = \text{RAND}('\text{NORMAL}', \mu, \sigma); \]
\[ h = \text{RAND ('POISSON', } \lambda \text{)}; \]

\[ t = \text{RAND ('T', df)}; \]

\[ X = \text{RAND ('UNIFORM')} ; \]

returns \( 0 < X < 1 \)

See SAS documentation (RAND function)
for additional info
More Statistical Functions

\( m = \text{median}(\text{list}); \)
computes median

\( a = \text{mean}(\text{list}); \)
computes mean of nonmissing values

\( s = \text{skeewness}(\text{list}); \)

\( s = \text{sum}(\text{list}); \)

\( s = \text{std}(\text{list}); \)

\( v = \text{var}(\text{list}); \)

\( \text{num} = n(\text{list}); \)
returns \# of nonmissing values in the variable list
EX: AVE = MEAN (OF EXAM1 - EXAM4);
    AVE = MEAN (EXAM1, EXAM2,
                EXAM3, EXAM4);

Could also compute
NEWAVE = (EXAM1 + EXAM2 + EXAM3 + EXAM4)/4;
But if any exam score is missing,
then NEWAVE = . also

Using the MEAN fcn (and other
builtin fcnstoo) will compute
average of nonmissing values
in the variable list.
The INPUT function

Used to convert a character string of digits to its numeric equivalent, and can also specify an informat for the numeric variable.

General Form

\[
\text{variable1} = \text{INPUT}(\text{variable2}, \text{informat});
\]

**EX:**

```
DATA CHAR2NUM;
SET ZIP.CODES;
* ZIPCODE IS $5. IN ZIP.CODES;
ZIPNUM= INPUT(ZIPCODE, 5.);
RUN;
```
PUT function

Used to convert a number to its equivalent character string value.

Can also specify an informat for the character variable.

General Form

```
var1 = PUT (var2, informat);
```

**EX:**

```
var char
    numeric
    var
```

5.11
LAG function

When we computed
\( X = \text{MEAN(OF } X_1 - X_3) \);
the values of \( X_1, X_2, X_3 \) were
all on the same observation:

name  \( X_1 \)  \( X_2 \)  \( X_3 \)

Builtin funcs like \texttt{MEAN}, \texttt{SUM},
etc. perform "within-observation"
processing.

But what if the data set had
one observation for each \( X \):

name  number   \( X \)

Joe    1       50
Joe    2       84
Joe    3       67
Trying to compute the mean of 3 values of $x$, one each on three observations, is difficult to do in a data step (this is easy to do using PROC MEANS however).

Can use the LAG fcn to accomplish this in a data step

**General Form**

```
LAG (variable)
LAG2 (variable)
LAGN (variable);
```
Extract "Substring" from a Character Variable's value

Use the SUBSTR fcn (substring) and the LENGTH fcn (don't get this confused with the LENGTH statement, which can also be useful for character extraction).

SUBSTR fcn

SUBSTR(char-var, start, length)

EX1 /* ZIPSTATE values 26505WV ;
DATA; SET ZIP.Codes;
Length State $2.;
State = SUBSTR(ZIPSTATE, 6, 2);
RUN;
Length  int
    Length ( char variable )
returns position of the right-most non-blank character, i.e., the actual number of non-blank characters.

Don't confuse this with the
Length stmt.
INDEX func

INDEX(char variable, 'string');

returns the beginning position in the char variable that contains the 'string'; or it returns 0 if the 'string' is not contained in the value of the char variable.

ex: Suppose NAME has current value 'EINSTEIN'

x = INDEX(NAME, 'N'); (x=3)
y = INDEX(NAME, 'A'); y=0
z = INDEX(NAME, 'n'); z=0
DATA CHECKING

Can use VERIFY function to make sure that a character variable contains valid values.

VERIFY (char variable, 'verify-string')

returns 0 if all characters in the character variable are specified as OK in the verify string; otherwise it returns first position of first
Char. that is not in the verifystring.
Removing Blanks or Other Characters from a String

Compress Fcn

Compress ( char., 'removechars');

Ex: Phone = Compress( telephone, 'Y( )-');
removes Y, ( ), - from the string values of telephone

Ex: NAME = Compress ( LASTNAME);
If 2nd argument is omitted, blank spaces only are removed.
Concatenating Strings
Can 'join' together two or more char. strings, using concatenation operator `||` (two vertical bars).
TRIM fn removes trailing blanks
LEFT fcn removes leading blanks.