where $a < b$.

See Also

Chapter III-12, **Statistics** for a function and operation overview; the **StatsRectangularCDF** and **StatsInvRectangularCDF** functions.

StatsResample

StatsResample /N=*numPoints* [*flags*] *srcWave*

The StatsResample operation resamples *srcWave* by drawing (with replacement) *numPoints* values from *srcWave* and storing them in the wave W_Resampled or M_Resampled if /MC is used. You can iterate the process and compute various statistics on the data samples.

Flags

/ITER=*n* Repeats the resampling for *n* iterations, which is useful only when combined with /WS or /SQ.

The /ITER flag is ignored by the Jack-Knife analysis (/JCKN).

/JCKN=*ufunc* Performs Jack-Knife analysis. Here ufunc is a user function of the format:

```
Function ufunc(inWave)
    wave inWave
    ... compute some statistic for inWave
    return someValue
End
```

The results are stored in the wave W_JackKnifeStats in the current data folder. Use

```
Edit W_JackKnifeStats.ld
```

to display the wave with dimension labels.

In the Jack-Knife method the operation runs N iterations where N is the number of points in *srcWave*. In each iteration the opeation calls the user-defined function ufunc(*inWave*) passing it an internal wave which contains (N-1) samples from *srcWave*. The function computes some user-defined statistic, say "Z", and stores it in *inWave*. At the end of iterations the operation uses the Z values in *inWave* to compute various Jack-Knife estimates. The standard estimator is defined as:

$$Z = ufunc(srcWave).$$

The Jack-Knife estimator is simply:

$$\hat{z} = \frac{1}{n}\sum_{i=1}^{n} z_i .$$

The Jack-Knife t-estimator is slightly less biased. It is given by:

$$t = nZ - (n-1)\hat{z},$$

The estimate of the standard error is given by:

$$\hat{\sigma}_{\hat{z}} = \sqrt{\frac{n-1}{n}\sum_{i=1}^{n}(z_i - \hat{z})^2} .$$

The Jack-Knife analysis ignores the /N and /ITER flags. The number of points and the number of iterations are determined by the number of points in *srcWave*.

/K — Kills W_Resampled after passing it to WaveStats. When /ITER is used, W_Resampled is not saved.

/MC — Use /MC when you want to sample random (complete) rows from a multi-column 2D *srcWave*. The combination of /N=n with /MC results in the wave M_Resampled in the current data folder. M_Resampled will have n rows, the same number of columns and the same data type as *srcWave*.

/N=*numPoints* — Specifies the number of points sampled from *srcWave*.

The /N flag is ignored by the Jack-Knife analysis (/JCKN).

/Q — No information printed in the history area.

/SQ=*m* — Uses StatsQuantiles to compute the data quartiles. The methods are:

m=0: Tukey (default).
m=1: Minitab.
m=2: Moore and McCabe.
m=3: Mendenhall and Sincich.

See Details for information about how the results are stored.

The default trim value is 25%.

/WS=*m* — Uses WaveStats operation to calculate data statistics.

m=0: Creates a new wave containing the samples (default).
m=1: Creates the new wave and passes it to `WaveStats/Q/M=1`.
m=2: Creates the new wave and passes it to `WaveStats/Q/M=2`.

See Details for information about how the results are stored.

/Z — Ignores any errors.

Details

StatsResample can perform Bootstrap Analysis, permutations tests, and Monte-Carlo simulations. It draws the specified number of data points (with replacement) from *srcWave* and places them in a destination wave W_Resampled.

Specify /WS or /SQ to use the WaveStats or StatsQuantiles operations, respectively, to compute results directly from the data. StatsResample normally creates the wave W_Resampled and, optionally, the M_WaveStats and W_StatsQuantiles waves. Both options also create various V_ variables described below. If you use more than one iteration, StatsResample creates instead the waves M_WaveStatsSamples and M_StatsQuantilesSamples for the results.

M_WaveStatsSamples (with /WS) contains a column for each iteration. Each column is equivalent to the contents of M_WaveStats for that iteration. You can use the command

```
Edit M_WaveStatsSamples.ld
```

to display the results in a table using row labels, and, for example, to display a graph of the rms of the samples as a function of iteration number execute:

```
Display M_WaveStatsSamples[5][]
```

M_StatsQuantilesSamples (with /SQ) contains a column for each iteration. Each column consists of the contents of W_StatsQuantiles for the corresponding data. Here again you can execute the command

```
Edit M_StatsQuantilesSamples.ld
```

to display the wave in a table using row labels. To display a graph of the median as a function of iteration execute:

```
Display M_statsQuantilesSamples[2][]
```

Output Variables

StatsResample creates the following variables: V_Median, V_Q25, V_Q75, V_IQR, V_min, V_max, V_numNaNs, V_numINFs, V_avg, V_sdev, V_rms, V_adev, V_skew, V_kurt, and V_Sum.

These variables are valid only if you use either /SQ or /WS, but not both, and only if you do not use /ITER. Unused variables are set to NaN.