

NAME

boxcar – Smooth a time series by computing its uniformly-weighted moving average.

SYNOPSIS

```
cat infile | ${HOME}/bin/exe/boxcar width > outfile
```

DESCRIPTION

If no command-line parameter is given, the program writes usage information and stops with return code 1. If a parameter is given, the program reads it to obtain the width W of the interval in X over which values of X and Y are to be averaged; if $W \leq 0$ it writes an error message and stops with return code 2.

The input X values must be in non-decreasing order. The program reads input coordinates $(X(I), Y(I))$ until the difference $|X(I) - X(1)| > W$; then it computes $XAVG$, the average of the X values preceding $X(I)$, and $YAVG$, the average of the Y values preceding $Y(I)$, and writes $(XAVG, YAVG)$ to the output file. Next it shifts the $X(I)$ and $Y(I)$ values one element left and appends one or more input values to X and Y to continue the process. The shifting of data left can be viewed as moving the boxcar right by one data point, but because the width of the boxcar is a difference between X values the boxcar might include different numbers of data points at different steps of the filtering process if the X values are not uniformly spaced. The first element in the output timeseries appears only after enough input data have been processed to fill the boxcar; thereafter each input element produces one output element. When the end of the input data is reached the program outputs the averages of the X and Y values that are in the boxcar even if they do not fill it.

This program considers the most realistic smoothed representation of the input data set to be that which associates each average of Y values with the average of the corresponding X values; usually this average X value will not be one of the $X(I)$ in the interval over which the averages are computed.

UNITS

- 5 input time series read from standard-in
- 6 output moving average written to standard-out

DIAGNOSTICS

These are the possible return codes.

- 0 all went well
- 1 usage information was written
- 2 specified boxcar width is not a positive number
- 3 the program's workspace is too small
- 4 the input X values are not in non-decreasing order
- 5 bad input data were encountered

EXAMPLE

```
unix[1] cat infile | ${HOME}/bin/exe/boxcar 0.3 > outfile
final boxcar has width 2.496061225410395D-01
unix[2] echo $?
0
unix[3]
```

AUTHOR

Michael Kupferschmid