Cumulative Sum and Moving Average Charts

Introduction

The Shewhart charts support actions that are based on the last sample observation: whether or not it crosses the control limits.

They respond quickly to large changes, but slowly if ever to small changes.

The Cusum and EWMA charts use the last observation, but also bring in information from past observations, and are more sensitive to small changes.
Cumulative Sum Control Chart

Simulated example

A hypothetical process, when in control, is normally distributed with mean \( \mu = 10 \) and standard deviation \( \sigma = 1 \). We observe 20 observations from \( N(10, 1) \) followed by 10 observations from \( N(11, 1) \): a 1\( \sigma \) out-of-control shift in the mean.

\[
x \leftarrow \text{read.csv("Data/Table-09-01.csv")}$x
\]
\[
\text{summary(qcc(x, "xbar.one", center = 10, limits = 10 + c(-3, 3)))}
\]

No points outside the control limits (although two runs are flagged).
Now plot the cumulative sum of (observation - center):

\[
xc <- \text{cumsum}(x - 10) \\
\text{plot}(xc, \text{type} = "b") \\
\text{abline}(h = 0, v = 20.5, \text{lty} = 2)
\]

The plot changes radically after the shift. But this is not a control chart: how to decide when action is needed?
V-Mask

source("R-code/vmask.R")
vmask(x - 10)

As each observation is added to the chart, place the V-mask over it. If any earlier point falls outside the “V”, declare the process out of control.

In this case, the last two points indicate loss of control.

Design choices:
- length of vertical;
- slope of control lines.
Tabular Cusum

The V-mask is convenient for a paper chart and a cardboard mask, but inconvenient for computation. The **tabular cusum** leads to the same decisions, algorithmically.

Choose a target value $\mu_0$ and a **reference value** $K$, and let

$$C_i^+ = \max \left[ 0, x_i - (\mu_0 + K) + C_{i-1}^+ \right]$$
$$C_i^- = \max \left[ 0, (\mu_0 - K) - x_i + C_{i-1}^- \right]$$

starting with $C_0^+ = C_0^- = 0$.

The process is declared out of control if either $C_i^+$ or $C_i^-$ exceeds a **decision interval** $H$. 
Notes

If a change of process mean to $\mu_1$ needs to be detected quickly, choose

$$K = \frac{|\mu_1 - \mu_0|}{2}.$$ 

Often $\mu_1 = \mu_0 \pm 1\sigma$ (a “one-sigma shift”), so $K = \sigma/2$.

Also, a decision interval $H = 5\sigma$ is often used.
In R:
The `qcc` package provides a function `cusum()` to make a chart showing $C^+_i$ and $C^-_i$ and control limits at $\pm H$, with these values by default:

```R
summary(cusum(x, center = 10, std.dev = 1))
```

The function `cusum()` makes the chart, but has no option to produce the actual table. However, the value returned by `cusum()` contains all the necessary information to make the table:

```R
with(cusum(x, center = 10, std.dev = 1, plot = FALSE),
     cbind(data, "Ci+" = pos, "Ci-" = -neg))
```
Cusum design

Because successive points on the cusum chart are not independent, its properties are difficult to describe.

Write $K = k\sigma$ and $H = h\sigma$. The $ARL_0$ has been calculated for various $k$ and $h$, and $ARL_1$ has been calculated for for various $k$, $h$, and shifts in the mean. See Tables 9.3 and 9.4.

In R:

```
library(spc)
xcusum.arl(k = 0.5, h = 5, mu = 0, sided = "two")
xcusum.arl(k = 0.5, h = 5, mu = 1, sided = "two")
```
Optimal design

Suppose we want a cusum chart with a specified ARL$_0$, say 370, and a low ARL$_1$ for a one-sigma shift.

In R:

```r
o <- optimize(function(k)
    xcusum.arl(k = k, h = xcusum.crit(k, 370, sided = "two"),
    mu = 1, sided = "two"),
    interval = c(0, 1))

print(o)
xcusum.crit(o$minimum, 370, sided = "two")
```

The optimal $k$ is indeed very close to 1/2, and the optimal $h$ (4.77) is close to 5, with the optimal ARL$_1 = 9.92$. 
Standardized cusum

Cusum charts are often based on standard\(\textit{ized}\) variables:

\[
z_i = \frac{x_i - \mu_0}{\sigma}.
\]

The \texttt{cusum()} function in the \texttt{qcc} package makes a standardized cusum chart.
Head start

To improve detection of an out-of-control condition at the start of the chart, initialize $C_0^+$ and $C_0^-$ at some positive value instead of 0, typically $H/2$ (a “50% head start”).

The `spc` package gives ARL calculations with the head start option, but the `cusum()` function in the `qcc` package has no provision for a head start in version 2.6, but it is supported in version 2.8.

Optimization shows that for $ARL_0 = 370$ and a 50% head start, $ARL_1$ for a one-sigma shift is minimized by $k = 0.40$ and $h = 5.68$. The optimal $ARL_1$ is 5.98, whereas with no head start it is 9.92.