The Calculation of Average Run Length for a Nonparametric CUSUM Procedure

Dennis W. King, STATKING Consulting Inc.
Michael T. Longnecker, Texas A&M University

ABSTRACT

A method of calculating the probability mass function of the Wilcoxon signed rank statistic is discussed. The algorithm presented, written using PROC IML, is based on the work of Milton (1970). A brief summary of a newly developed error bound for this algorithm is given.

For process control schemes based on the Wilcoxon signed rank statistic, the calculation of the Average Run Length (ARL) of the scheme requires the evaluation of the probability mass function of the Wilcoxon statistic under a shift in the location parameter. The application of the algorithm's output to the calculation of ARL is shown in detail.

1. INTRODUCTION

The use of statistical procedures to monitor repetitive manufacturing processes has become quite widespread. A characteristic of the process is observed over time. The observed random variables may be measurements of the weight of rolled coils of processed steel or the measured dimension of a die which was cut to a specified length.

The main thrust is to track whether the manufactured product is near a management specified goal value, say $\theta_0$. In statistical terms this means monitoring the mean of the process random variables.

Many authors have discussed sequential statistical tests to indicate when a process goes "out of control", i.e., when the process mean drifts away from $\theta_0$. Some of the authors who have addressed this topic are Shewhart (1931), Page (1954) and Lucas (1976). All of these articles center on the calculation of

$$S_n = \sum_{i=1}^{n} w_{n-i} \cdot (v(X_i) - k), \quad (1.1)$$

at each time point $n$ in the process. The components of this sum are $w_{n-i}$, the weight placed on the observation at each time point; $v(X_i)$, some condensation of the data vector $X_i = (X_{i1}, \ldots, X_{ig})$ observed at time point $i$; and $k$, a reference value. For a CUSUM procedure, we set all $w_i = 1$. The choice of $v(X_i)$ is arbitrary but a common choice is $X - \theta_0$. We test $H_0 : \theta = \theta_0$ and reject $H_0$ in favor of $H_\alpha : \theta > \theta_0$ when $S_i \geq h$ where

$$S_n = \max(0, S_{n-1} + v(X_n) - k), \quad (1.2)$$

where $S_0 = 0$. The reference value $k$ is chosen so that the decision interval $h$ does not depend on the time point, i.e., we have a constant rejection region. Note that we are making an unknown number of statistical tests since the procedure continues indefinitely until we reject $H_0$. The random variable $N$ will be used to denote the number of tests performed before the process is
When the condensation statistic, \( v(X_i) \), is a continuous random variable, such as \( X - \theta_0 \), then the ARL is given by an integral equation (see Lucas (1976)) which must be solved by methods of numeric integration. When \( v(X_i) \) is a discrete random variable, Brook and Evans (1972) have shown that the ARL is given by

\[
E(N) = (I - P)^{-1} \cdot 1 \tag{1.3}
\]

where \( P \) is the probability transition matrix of a Markov chain whose states are the values of the discrete CUSUM.

2. THE DISTRIBUTION OF THE WILCOXON STATISTIC

When \( g > 1 \), we may use in place of \( X - \theta_0 \) in (1.1),

\[
v(X_i) = SR_i = \sum_{j=1}^{g} \Psi_j a(R_j^+) \tag{2.1}
\]

where \( R_j^+ \) is the rank of \(|X_j - \theta_0|\) among \(|X_1 - \theta_0|, ..., |X_g - \theta_0|\), \( \Psi_j = \Psi(X_j - \theta_0) \) and \( \Psi(t) = -1, 1 \) as \( t <, \geq 0 \). For the set of scores, \( a(j) = j \), \( j = 1, ..., g \), the signed linear rank statistic in (2.1) becomes the Wilcoxon Signed Rank Statistic, denoted henceforth as \( W \).

Since this condensation statistic is discrete, the ARL for this type of CUSUM will be given exactly by (1.3). The elements of the probability transition matrix in (1.3) will be formed from the probability distribution of \( W \). The states of the Markov chain for any particular step are the values of \( W, 0,1, ..., h \). So then the transient state probabilities are

\[
p_{ij} = P(S_i = j - i) = P(W = k + j - i)
\]

(2.2)

To obtain the distribution of \( W \), we first find the distribution of \( W^+ = \sum_{j=1}^{g} \Psi_j a(R_j^+) \) where \( a(j) \) are as in (2.1) and \( \Psi_j(t) = 0,1 \) as \( t <, \geq 0 \). Then applying the transformation \( W = 2(W^+ - E(W^+)) = 2W^+ - g(g + 1)/2 \),

\[
P(W = w) = P(W^+ = w/2 + E(W^+)). \tag{2.3}
\]

When the process is in control, i.e. \( \theta = \theta_0 \), all orderings of the ranks are equally likely and it can be shown (see Randles and Wolfe (1979), p. 52) that

\[
P_0(W^+ = c) = d_g(c)/2^g, \tag{2.4}
\]

where \( d_g(c) \) is the number of subsets of the integers \( 1, ..., g \) for which the sum of the elements in the subset equals \( c \) and \( c = 0,1, ..., g(g + 1)/2 \).

When the process is out of control \( \theta > \theta_0 \), not all permutations of the ranks are equally likely and additional notation is necessary. Without loss of generality take \( \theta_0 = 0 \). Let \( X_{(1)} \leq X_{(2)} \leq ... \leq X_{(g)} \) be the ordered values of \( X_1, X_2, ..., X_g \). Denote \( Z_{g} = 1 \) if the \( i \)th smallest observation, \( X_{(i)} \), is nonnegative and 0 if the \( i \)th smallest is negative. The rank configurations are then \( Z_g = (Z_{g1}, Z_{g2}, ..., Z_{gg}) \). Thus, \( W = \sum jZ_{gj} \). Then \( P(W^+ = c) \) is obtained by summing \( P_0(Z_g = z_g) \) for all possible rank configurations \( Z_g \) for which \( \sum jZ_{gj} = c \). For example, with \( g = 5 \), \( P_0(W^+ = 12) = P_0(Z_5 = 3) \).
Klotz (1963) has shown the expression for \( P_\theta (Z_\theta = z_\theta) \) is

\[
g^1 \int_0^\infty \int_0^{t_2} \cdots \int_0^{t_2} f_0(t_i - s_i \theta) dt_i \quad (2.5)
\]

where \( f_0 \) is the pdf of \( P_\theta \) and \( s_i = 2z_i - 1 \).

Denote this integral as \( I_2 \).

Milton (1970) recognized that the region of integration, depicted in Figure 1 in the appendix, for this particular problem allows a convenient approximation formula. In Figure 1, we see that if we use a simple midpoint formula for numeric integration in any one dimension the region we need will be given by

\[
I_2 \doteq m^2 \left( \frac{f_{11}f_{21} + f_{12}f_{22} + f_{13}f_{23}}{2} 
+ f_{11}f_{22} + f_{11}f_{23} + f_{12}f_{23} \right)
\]

where \( m = (b-a)/3 \) for \( M=3 \) subintervals and \( f_{ij} \) is the pdf \( f_0 \) in (2.5) evaluated at the midpoint of its respective subinterval. This generalizes to \( M \) subintervals in 2 dimensions as

\[
I_2 \doteq m^2 \left( \sum_{i=1}^M f_{1i}f_{2i} \right)
\quad (2.6)
\]

where \( m = (b-a)/3 \) for \( M=3 \) subintervals and \( f_{ij} \) is the pdf \( f_0 \) in (2.5) evaluated at the midpoint of its respective subinterval. This generalizes to \( M \) subintervals in 2 dimensions as

\[
I_2 \doteq m^2 \left( \sum_{i=1}^M f_{1i}f_{2i} \right)
\quad (2.6)
\]

We can then further generalize to \( g \) dimensions and state the above in matrix terms. This allows us to program the algorithm using PROC IML. This method is much more easily programmed than the quadrature methods and requires only \( M^g \) storage locations in the computer.

A brief description of the error bounding for this algorithm is now given. For the details, see King and Longnecker (1990). The method of numeric integration used here is subject to two sources of error. The error introduced by truncating an infinite region of integration will be denoted by \( \epsilon_{\text{trunc}} \). The error introduced by using a Newton-Cotes formula to approximate the integral over the finite region will be denoted \( \epsilon_{\text{calc}} \). The total error is then

\[
\epsilon = \epsilon_{\text{trunc}} + \epsilon_{\text{calc}}. \quad (2.8)
\]

When using the algorithm we first specify a truncation error, say \( \epsilon_0 \). It has been shown by King and Longnecker (1990), that if we choose the truncation boundary, \( C \), as shown in Figure 1, such that

\[
C \geq F_\theta^{-1}(1 - \epsilon_0) + \theta. \quad (2.9)
\]

then \( \epsilon_0 \leq \epsilon_{\text{trunc}} \). Once \( C \) has been determined, we then select \( m \), the width of the interval in the midpoint numerical integration procedure, by iterating on \( m \) until

\[
((f_{\text{max}}(m) + D(m))^g - (f_{\text{max}}(m))^g) * \text{NADDS} < \epsilon_{\text{calc}} \quad (2.10)
\]

where \( \text{NADDS} \), the total number of additions performed, \( = (M^g-1) \), where \( M = C/m \). The value \( f_{\text{max}}(m) \) is height of the highest rectangle in the integration region and \( D(m) \) is the maximum value of the integration error over the region. These values are given in King and Longnecker (1990) for specific symmetric distributions. If we follow this two-step procedure then the total error will be \( \leq \epsilon \). The above error bounding procedure has been implemented in the SAS Macros to be described below.

3. SAS MACROS FOR WILCOXON CUSUM

The programming necessary to implement the procedures discussed in section 2 is threefold: (a) calculate \( C \) and \( m \) necessary to achieve the specified error bound, \( \epsilon \), for
each probability in the probability density function of $W$ for the given underlying distribution and group size, (b) compute the probability density function of $W$ under specified shift according to the Milton Algorithm and (c) calculate the ARL of the Wilcoxon CUSUM using the pdf of $W$. This is done with three SAS Macros, whose call statements are of the form

```
%BOUND(P=,SHIFT=,DIST=,DF=);  
%WALTDIST(P=,SHIFT=,DIST=,DF=);  
%ARLWILC(P=,SHIFT=,DIST=,DF=,H=,K=  
```

where

- **DIST** = name of the underlying distribution of the data (NORM, LOGS, DEXP, T)
- **P** = sample size at each time point
- **H** = decision interval
- **K** = reference value
- **DF** = degrees of freedom for the T distribution
- **SHIFT** = the shift from the target mean in standard deviation units

Note that you may run only the error bounding macro or the error bound and the distribution of the Wilcoxon Statistic or all 3 macros. The outputs for these macros are shown in the appendix.

### 4. ARL COMPARISONS

The ARL under various shifts of size $\theta$ can then be calculated. Table 1 in the appendix shows the ARL for a particular Wilcoxon CUSUM for various underlying distributions. The errors in using the estimated probabilities from above is < 1%. Note the underlying distribution must be specified when using nonparametric statistic, $v(X_i)$, under a shift $\theta$. The ARL's can then be compared to a comparable CUSUM procedure (in terms of in-control ARL) where $v(X_i) = X - \theta_0$.

Table 1 shows that when the data comes from very heavy tailed distributions the ARL is shorter for the Wilcoxon CUSUM than the parametric CUSUM. For the other distributions, the loss in ARL is small so the nonparametric procedure can be considered a global procedure to protect against non Normal data in the small sample case.

### 5. A DATA EXAMPLE

As an example, consider data generated from the soft drink industry. At the end of each hour of production, five bottles of soft drink are sampled and their fill levels are recorded. The data values shown in Table 2 are the deviations of the fill levels relative to the target value of 8 ounces. The data are assumed to be sampled from a Normal universe with $\sigma = 1$ ounce. An upward shift in the process mean of .25 ounces has occurred. As we can see, the nonparametric CUSUM detects this shift at time point 5 and the parametric procedure signals slightly more quickly at time point 4. This is consistent with the ARL results of Table 1. The calculations necessary to implement the nonparametric scheme are not that difficult and could be carried out by technicians or line personnel.

### 6. DISCUSSION

The procedure discussed above can calculate the values of the probability density function of the Wilcoxon signed rank statistic to 4 decimal place accuracy for most symmetric densities. It can also calculate the
ARL for the Wilcoxon CUSUM procedure described above. However, the code does have some limitations.

The procedure is inadequate for the scaled $T$ distribution with $\leq 1$ degree of freedom. This is due to the fact that the step size, $m$, cannot be varied across the region of integration. This, in turn, makes the computer storage and cpu requirements in this situation infeasible, at least for implementation on most microcomputers.

For other symmetric distributions, the time necessary to calculate the ARL of the Wilcoxon CUSUM still prohibits generation of large tables of ARL. Thus, the procedure is more of a research tool which allows comparison of the ARL's for this procedure versus ARL's for parametric approaches.

REFERENCES


Table 1. Comparison of Parametric and Nonparametric Exact ARL's for Various Underlying Distributions, $g = 5$

<table>
<thead>
<tr>
<th>Procedure</th>
<th>Distribution</th>
<th>0</th>
<th>.25</th>
<th>.5</th>
<th>1.0</th>
<th>2.0</th>
<th>3.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUSUM</td>
<td>Normal</td>
<td>100.9</td>
<td>30.6</td>
<td>15.8</td>
<td>7.9</td>
<td>5.1</td>
<td>5.0</td>
</tr>
<tr>
<td>Wilcoxon</td>
<td>Normal</td>
<td>100.9</td>
<td>36.1</td>
<td>17.9</td>
<td>8.8</td>
<td>5.5</td>
<td>5.0</td>
</tr>
<tr>
<td>CUSUM</td>
<td>Double Exp.</td>
<td>100.5</td>
<td>31.0</td>
<td>15.5</td>
<td>8.5</td>
<td>5.0</td>
<td>5.0</td>
</tr>
<tr>
<td>Wilcoxon</td>
<td>Double Exp.</td>
<td>100.9</td>
<td>27.3</td>
<td>14.1</td>
<td>8.0</td>
<td>5.7</td>
<td>5.2</td>
</tr>
<tr>
<td>CUSUM</td>
<td>Logistic</td>
<td>100.5</td>
<td>30.0</td>
<td>16.0</td>
<td>8.5</td>
<td>5.0</td>
<td>5.0</td>
</tr>
<tr>
<td>Wilcoxon</td>
<td>Logistic</td>
<td>100.9</td>
<td>33.2</td>
<td>16.4</td>
<td>8.4</td>
<td>5.6</td>
<td>5.1</td>
</tr>
<tr>
<td>CUSUM</td>
<td>$T^* 2 , df$</td>
<td>99.5</td>
<td>42.0</td>
<td>19.0</td>
<td>8.5</td>
<td>5.0</td>
<td>5.0</td>
</tr>
<tr>
<td>Wilcoxon</td>
<td>$T^* 2 , df$</td>
<td>100.9</td>
<td>26.0</td>
<td>13.3</td>
<td>7.8</td>
<td>5.8</td>
<td>5.4</td>
</tr>
</tbody>
</table>

FIG. 1 Region of Integration for the Milton Algorithm
# Table 2

<table>
<thead>
<tr>
<th>Sample Data</th>
<th>Signed Ranks</th>
<th>Wilcoxon Statistic</th>
<th>Parametric CUSUM</th>
<th>Wilcoxon CUSUM</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2, -0.5, 1, 1.0, 0.8, 0.4</td>
<td>-5, -2, 4, 3, 1</td>
<td>1</td>
<td>0.00000</td>
<td>0</td>
</tr>
<tr>
<td>0.2, 0.1, 1.4, 0.0, -1</td>
<td>3, 2, 5, 1, -4</td>
<td>7</td>
<td>0.00000</td>
<td>2</td>
</tr>
<tr>
<td>1.7, 2.1, -1, 1.1, 0.2</td>
<td>4.5, -3, 2, 1</td>
<td>9</td>
<td>0.66293</td>
<td>6</td>
</tr>
<tr>
<td>-1, 0.2, 0.2, 1.9, 1.2</td>
<td>-4, 2, 1, 5, 3</td>
<td>7</td>
<td>0.95244 *</td>
<td>8</td>
</tr>
<tr>
<td>1.7, -0.0, 0.5, -3, 1.1</td>
<td>5, -1, 3, -2, 4</td>
<td>9</td>
<td>1.43964 *</td>
<td>12 *</td>
</tr>
</tbody>
</table>

## Output from BOUND Macro

<table>
<thead>
<tr>
<th>g</th>
<th>SHIFT DIST</th>
<th>DF</th>
<th>C</th>
<th>EPSCALC</th>
<th>EPSTRUNC</th>
<th>NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1</td>
<td>NORM</td>
<td>4</td>
<td>5.265</td>
<td>0.011</td>
<td>.00001</td>
</tr>
</tbody>
</table>

## Output from WALTDIST Macro

<table>
<thead>
<tr>
<th>OBS</th>
<th>c</th>
<th>Wilcoxon pdf</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0.00010</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.00013</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0.00018</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>0.00054</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>0.00089</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>0.00287</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
<td>0.00419</td>
</tr>
<tr>
<td>8</td>
<td>7</td>
<td>0.00671</td>
</tr>
<tr>
<td>9</td>
<td>8</td>
<td>0.01319</td>
</tr>
<tr>
<td>10</td>
<td>9</td>
<td>0.02571</td>
</tr>
<tr>
<td>11</td>
<td>10</td>
<td>0.04143</td>
</tr>
<tr>
<td>12</td>
<td>11</td>
<td>0.05509</td>
</tr>
<tr>
<td>13</td>
<td>12</td>
<td>0.11568</td>
</tr>
<tr>
<td>14</td>
<td>13</td>
<td>0.10470</td>
</tr>
<tr>
<td>15</td>
<td>14</td>
<td>0.20697</td>
</tr>
<tr>
<td>16</td>
<td>15</td>
<td>0.42155</td>
</tr>
</tbody>
</table>

## Output from ARLWILC Macro

<table>
<thead>
<tr>
<th>OBS</th>
<th>ARL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8.81426</td>
</tr>
</tbody>
</table>