



## DixonTest.CriticalValues: A Computer Code to Calculate Critical Values for the Dixon Statistical Data Treatment Approach

Mahendra P. Verma

Instituto de Investigaciones Electricas

Mario Cesar Suarez A.

Universidad Michoacana-UMSNH

---

### Abstract

The computer program **DixonText.CriticalValues** is written in VB.NET to extend the quadrature approach to calculate the critical values with accuracy up to 6 significant digits for Dixon's ratios. Its use in creating the critical values tables in **Excel** is illustrated.

*Keywords:* critical value, Dixon ratio, **Excel**, quadrature, VB.NET.

---

## 1. Introduction

Statistical tests are used for the identification and rejection of extremums, commonly called outliers in a dataset and their application for analytical data treatment is common in all branches of science (Lohginger 1999; Verma, Quiroz-Ruiz, and Diaz-Gonzalez 2008). Recently, Komsta (2006) developed a computer software package, **outliers**, for analytical data processing using various statistical tests in the programming language R (R Core Team 2013). Dixon's ratio or Q-test is one of the methods for identifying extremums in a sample, taken from a normal distribution (Dixon 1950, 1951; Dean and Dixon 1951; Rorabacher 1991). The implementation of this approach requires the calculation of critical values. Dixon (1951) created the critical values tables for sample data points up to 30 with uncertainty in the third digit. Komsta (2006) used the Dixon's critical values. Now, many researchers have calculated the critical values for a large number of data points with higher accuracy (Lohginger 1999; Miller and Miller 2005; McBane 2006; Verma *et al.* 2008). Two types of numerical analysis methods are used to evaluate the integral equation: *Monte Carlo simulation* (Efstathiou 1992; Verma *et al.* 2008) and *quadrature* (Dixon 1950, 1951; McBane 2006).

In this article the quadrature approach is extended by programming the generating functions for the polynomials like Hermite (Riley, Hobson, and Bence 2006; Weisstein 2011), Legendre

(Riley *et al.* 2006; Weisstein 2011) and half-range Hermite (Steen, Byrne, and Geldard 1969; Gautschi 1994; Ball 2003), which are used for the evaluation of definite integrals through the numerical quadrature approximation method. The program, **DixonTest.CriticalValues** is written as an object library (DLL; dynamic link library) in VB.NET, which permits its use in any other programming environment on any platform, where the **.NET framework** and the virtual environment called as Common Language Runtime (CLR) are installed (Venkat 2012). The CLR contains a component called Just In-Time Compiler (JIT), which converts the intermediate language into native code for the underlying operating system. A demonstration program, **DixonDemo**, is written in VB.NET. Similarly, the creation of critical values tables in **Excel** is illustrated.

## 2. Theoretical aspects

For the sake of completeness, the procedure of Dixon (1951) and McBane (2006) is summarized here. The joint probability density function for a set of  $n$  random observations  $(x_1, x_2, \dots, x_n)$  from a normal distribution with mean ( $\mu$ ) and standard deviation ( $\sigma$ ) is defined as

$$P(x_1, x_2, \dots, x_n) = \prod_{i=1}^n \phi(x_i), \quad (1)$$

where  $\phi(x_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp \left\{ -\frac{1}{2} \left( \frac{x_i - \mu}{\sigma} \right)^2 \right\}$ .

For the ordered set of observations  $(x_1 \leq x_2 \leq \dots < x_n)$ , Dixon (1951) defined the ratio,  $r_{j,i-1} = (x_n - x_{n-j}) / (x_n - x_i)$ , where subscripts  $i$  and  $j$  indicate the number of suspected outliers at the lower and upper ends of the data set, respectively. The ratios are independent of  $\mu$  and  $\sigma$  of the distribution; therefore, the joint density function is equally valid for the standard normal distribution (i.e.,  $\mu = 0$  and  $\sigma = 1$ ).

The Dixon's ratios are functions of three variables ( $x_i, x_{n-j}$  and  $x_n$ ) of the  $n$  data values and the corresponding joint probability density function is obtained by integrating Equation 1 over all except the three points. Then the variables are changed from  $(x_i, x_{n-j}, x_n)$  to  $(t, u, r)$  according to the transformations as  $t^2 = (1 + r^2)v^2/2$  and  $u^2 = 3x^2/2$ , where  $x = x_n$ ,  $v = x_n - x_i$  and  $r = (x_n - x_{n-j})/v$ , and the joint density function is integrated for  $x$  and  $v$  over their ranges  $(-\infty < x < \infty, 0 \leq v < \infty)$  to obtain the function of  $r$  alone as

$$P(r) = \frac{n!}{(i-1)!(n-j-i-1)!(j-1)!} \int_{-\infty}^{\infty} \int_0^{\infty} \left[ \int_{-\infty}^{x-v} \phi(t) dt \right]^{i-1} \times \left[ \int_{x-v}^{x-rv} \phi(t) dt \right]^{n-j-i-1} \left[ \int_{x-rv}^x \phi(t) dt \right]^{j-1} \phi(x-v) \phi(x-rv) \phi(x) v dv dx. \quad (2)$$

For simplicity,  $J(x, r, v)$  is defined as

$$J(x, r, v) = \left[ \int_{-\infty}^{x-v} \phi(t) dt \right]^{i-1} \left[ \int_{x-v}^{x-rv} \phi(t) dt \right]^{n-j-i-1} \left[ \int_{x-rv}^x \phi(t) dt \right]^{j-1}. \quad (3)$$

On substituting values of  $\phi$ 's and Equation 3 in the Equation 2, we get

$$P(r) = N \int_{-\infty}^{\infty} \exp \left( -\frac{3x^2}{2} \right) \int_0^{\infty} \exp \left\{ -\frac{(1+r^2)v^2}{2} \right\} J(x, v, r) \exp \{ (1+r)xv \} v dv dx, \quad (4)$$

where  $N = \frac{n!}{(i-1)!(n-j-i-1)!(j-1)!} (2\pi)^{-3/2}$ .

After further simplification, the Equation 4 is converted to the integral form suitable for the Gauss-Hermite quadrature approximation approach as

$$P(r) = N \sqrt{\frac{2}{3}} \sqrt{\frac{2}{1+r^2}} \int_{-\infty}^{\infty} \exp(-u^2) \int_0^{\infty} \exp(-t^2) J(x(u), v(t, r), r) \times \exp\left\{\frac{2ut(1+r)}{\sqrt{3(1+r^2)}}\right\} v(t, r) dt du, \quad (5)$$

where  $x(u) = u\sqrt{2/3}$  and  $v(t, r) = t\sqrt{2/(1+r^2)}$ .

The numerical integration for  $u$  with limits  $(-\infty$  to  $+\infty)$  is performed with the Gauss-Hermite quadrature. The integration limits for  $t$  are from 0 to  $+\infty$ . The above integration for  $t$  may be performed with the half-range Hermite polynomials (Steen *et al.* 1969; Gautschi 1994; Ball 2003). The half-range Hermite polynomials are developed differently by different people using the recurrence formula approach. The instability in the approach creates fluctuations after a certain degree of the polynomials (Steen *et al.* 1969), which was overcome with the brute force method of using arbitrary precision arithmetic (Gautschi 1994; Ball 2003). Ball (2003) presented a new approach using the asymptotic initial Gauss values and refining the values from higher to lower side (backward) by iteration procedure and obtained the identical values of abscissa and weights of half-range Hermite as proposed by Gautschi (1994). In this work the approaches of Steen *et al.* (1969) and Ball (2003) for the generation of half-range Hermite polynomials are implemented in order to present a comparative study in the calculation of total probability density and critical values.

According to the quadrature integration approach, the approximate solution of Equation 5 for the joint probability density  $P(r)$  is expressed as

$$P(r) \simeq N \sqrt{\frac{2}{3}} \sqrt{\frac{2}{1+r^2}} \sum_{k=1}^{nH} w_{nH}(u_k) \sum_{l=1}^{nHH} w_{nHH}(t_l) J(x(u_k), v(t_l, r), r) \times \exp\left\{\frac{2u_k t_l (1+r)}{\sqrt{3(1+r^2)}}\right\} v(t, r), \quad (6)$$

where  $u_k$  is the value of the  $k$ th abscissa of the  $nH$ -degree Hermite polynomial (written in short as  $n$ -Hermite) and  $w_{nH}(u_k)$  is the corresponding weight. Similarly,  $t_l$  and  $w_{nHH}(t_l)$  are the corresponding parameters for the  $nHH$ -degree half-range Hermite polynomial either of Gautschi (1994) and Ball (2003) denoted as half Hermite Ball or of Steen *et al.* (1969) denoted as half Hermite SBG. The variable  $r$  has limits 0 to 1.

Riley *et al.* (2006) and Weisstein (2011) explained the estimation of error in the numerical quadrature; however, the error calculation must be performed in each step of the program (Kumar 2011). Here we will implement a simple procedure to estimate the accuracy (precision) in the critical values for the Dixon's ratios by calculating the accuracy in the total probability density (or cumulative density function, *CDF*).

The calculation of critical values for the Dixon ratios is performed by integrating the Equation 6 for  $r = 0$  to  $R$  and the equation is evaluated with the numerical quadrature approach using the Legendre polynomials (McBane 2006). The generating functions for the Legendre polynomials are taken from Riley *et al.* (2006) and Weisstein (2011). The numerical solution

is performed here in two steps: (i) integrating for the total probability density (i.e., changing the limits from  $r = 0$  to  $1$  to  $z = -1$  to  $+1$ ) and (ii) integrating for the cumulative probability density up to  $R$  (i.e., changing the integration limits from  $r = 0$  to  $R$  to  $z = -1$  to  $+1$ ). For a given value of  $CDF$ , the critical value  $R$  is calculated by the bisection method (Press, Teukolsky, Vetterling, and Flannery 1992) using the initial values,  $CDF = 0$ ,  $R = 0$  and  $CDF = 1$ ,  $R = 1$ .

In this work the program of McBane (2006) in FORTRAN is rewritten in VB.NET. The programming of special functions (Hermite, Legendre and half range Hermite) is included which makes the program versatile for comparative studies. Similarly, the present program, **Dixon-Test.CriticalValues** provides better accuracy up to 6 decimal places in the critical values and can be used in Microsoft Windows-based software including **Excel**.

### 3. Implementation

#### 3.1. Critical values calculation

The present computer program, **DixonTest.CriticalValues**, is written as an object library (DLL) in VB.NET. A namespace `DixonTest` is created which will be used in future to include all the programs associated with the Dixon statistical test methods. Presently, it contains a class 'CriticalValues'. The properties of the class are given in Table 1. A class encapsulates the data and methods, and serves as blue-print for creating objects. A class may be extended for its functionality without knowing its code. The use of classes in **Excel** will be presented later.

#### 3.2. DixonDemo: A demonstration program

To illustrate the calculation of critical values, a demonstration program, **DixonDemo**, is written in VB.NET. Figure 1 shows the graphical user interface of the program. The user provides the values of `nDixon`, `iDixon`, `jDixon`, `Alpha`,  $n$ -Hermite,  $n$ -HHermite, half Hermite (Ball),  $n$ -Legendre and accuracy (`RAccuracy`). On pressing the button, `Calc`, the values of total probability, critical value, and corresponding cumulative probability are calculated as shown in Figure 1. The accuracy indicates the number of accurate digits after the decimal point in the critical value. One should check the accurate digits in the total probability in order to assign the value of `RAccuracy`. Indeed, the program works for any value of `RAccuracy`; but the accuracy in the critical value is one digit less than the accurate digits in the total probability. If the check-box of half Hermite (Ball) is selected, the program uses the half-range Hermite polynomials of Ball (2003), otherwise of Steen *et al.* (1969). The upper limit for the degree of  $n$ -HHermite SGB is 15; there is some instability in the polynomials after this degree (Steen *et al.* 1969).

Figure 1 shows the calculation for `nDixon` = 7, `iDixon` = 2, `jDixon` = 1, `Alpha` = 0.1 (i.e., for the ratio  $r_{11}$  or  $r_{1,1}$  with 90% confidence interval) using the half Hermite SBG. There are seven accurate digits in the total probability density, so the critical value is 0.532944 (i.e., accurate at least up to six decimal places). However, the total probability is 0.9996055, when calculated using the half Hermite Ball polynomial while keeping the same values for the other parameters. Thus the accuracy reduces to 3 digits instead of 7 in the total probability and

Name	Access	Description
nDixon	Read-write	An integer value which denotes the total number of data points.
iDixon	Read-write	An integer value which denotes the number of suspected lower outliers.
jDixon	Read-write	An integer value which denotes the number of suspected upper outliers.
nHermPoly	Write only	An integer value which denotes the degree of Hermite polynomial.
nHHermPoly	Write only	An integer value which denotes the degree of half-range Hermite polynomial.
nLegPoly	Write only	An integer value which denotes the degree of Legendre polynomial.
HalfHermiteBall	Write only	A Boolean value which denotes the type of half-range Hermite ( <b>True</b> – Ball (2003) and <b>False</b> – Steen <i>et al.</i> (1969)).
Alpha	Read-write	A normalized value to represent the confidence interval (e.g., for 95% confidence interval, Alpha = 0.05).
RAccuracy	Write only	An integer value which denotes the calculated accurate digits in the values of rCritical after the decimal point.
TotalProbability	Read only	Total probability for a given value of nDixon, iDixon, and jDixon.
rCritical	Read only	Critical value for a given value of nDixon, iDixon, and jDixon.
rProbabilty	Read only	Cumulative probability for a given value of nDixon, iDixon, jDixon and Alpha

Table 1: Properties of class ‘CriticalValues’.

consequently up to 2 digits in the critical values.

Figure 2 presents a comparative study for the calculation of total probability using the half Hermite Ball and half Hermite SBG polynomials. The values of the other parameters were kept the same for both cases. On the  $x$ -axis the degree of the half Hermite polynomial is plotted, whereas the total probability is plotted on the  $y$ -axis, which was multiplied by  $1 \times 10^6$  and then  $9 \times 10^5$  was subtracted from it in order to amplify the probability axis scale. The values of common parameters are nDixon = 7, iDixon = 2, jDixon = 1, Alpha = 0.0,  $n$ -Legendre = 12. The value of  $n$ -half Hermite for SBG is up to 15 as there is instability in the polynomial for higher values. The value of  $n$ -Hermite is changed to 5, 10, 15 and 20 in the respective figures. For  $n$ -Hermite = 5 and 10, there are similar fluctuating behaviors in the total probability for both half Hermite Ball and half Hermite SBG, whereas for  $n$ -Hermite = 15, the behavior is getting stable for half Hermite SBG and is stable for  $n$ -Hermite = 20. But there is the same fluctuating behavior in the case of half Hermite Ball even for  $n$ -Hermite = 15 and 20. Thus the half Hermite polynomial of Steen *et al.* (1969) provides better results for these calculations, when the value of nDixon is less than 10. Similarly, the adequate accuracy in the critical value with minimum expense is obtained by  $n$ -half Hermite = 10 for the SBG polynomials.

Figure 1: Graphical user interface of the demonstration program to illustrate the calculation of critical values for Dixon's ratios.

nDixon	Dixon (1951)	Rorabacher (1991)	Lohginger (1999)	McBane (2006)	Verma <i>et al.</i> (2008)	This work crit. values	This work total prob.
5	0.642	0.642	0.642	0.642	0.64228	0.642357	0.99999970
10	0.412	0.412	0.412	0.412	0.41184	0.411860	0.99999966
15	0.338	0.338	0.338	0.339	0.33856	0.338539	0.99999999
20	0.300	0.300	0.300	0.300	0.30044	0.300499	0.99999999
25	0.277	0.277	0.277	0.276	0.27646	0.276421	1.00000002
30	0.260	0.260	0.260	0.259	0.25942	0.259449	1.00000005
40					0.23669	0.236563	1.00000006
50					0.22145	0.221436	0.99999943
60					0.21043	0.210451	0.99999980
70					0.20203	0.201984	1.00000138
80					0.19510	0.195187	1.00000043
90					0.18957	0.189568	0.99999730
100					0.18482	0.184711	0.99999642

Table 2: Comparison of critical values for the ratio  $r_{10}$  for  $\text{Alpha} = 0.05$  (i.e., confidence interval = 95%). For  $\text{nDixon} > 50$ , there is accuracy up to at least 4 digits.

Figure 3 presents the behavior of total probability for different values of  $\text{nDixon}$  with keeping the same values of other parameters (i.e.,  $i\text{Dixon} = 2$ ,  $j\text{Dixon} = 1$ ,  $n\text{-Hermite} = 30$ ,  $n\text{-Legendre} = 12$ ). For  $\text{nDixon} = 12$ , the behavior of total probability for both types of  $n$ -half Hermite polynomials is the same and stable; however, on further increasing  $\text{nDixon}$  there is fluctuation in the first part of the behavior. The highest degree of half Hermite SBG is 15, so the total probability and critical value calculation has higher uncertainty. In the case of  $\text{nDixon} = 100$ , the behavior is stable for the degree of  $n$ -half Hermite Ball ( $\geq 45$ ). Thus the half Hermite Ball provides better results for  $\text{nDixon} > 12$ .

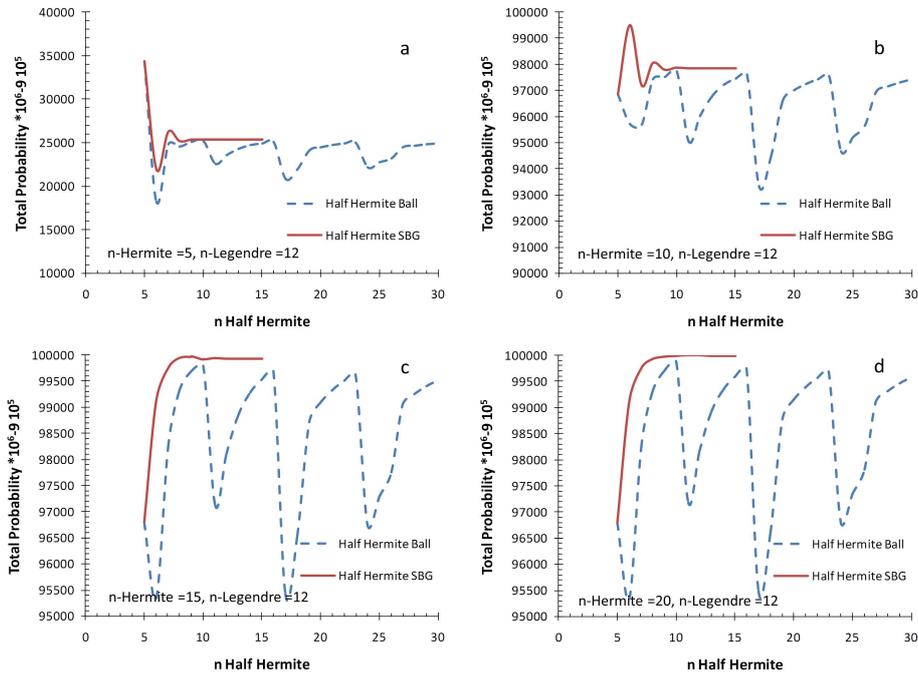


Figure 2: Variation of total probability with the degree of half Hermite polynomial to study the variation of the degree of Hermite polynomials ( $n$ -Hermite = 5, 10, 15 and 20).

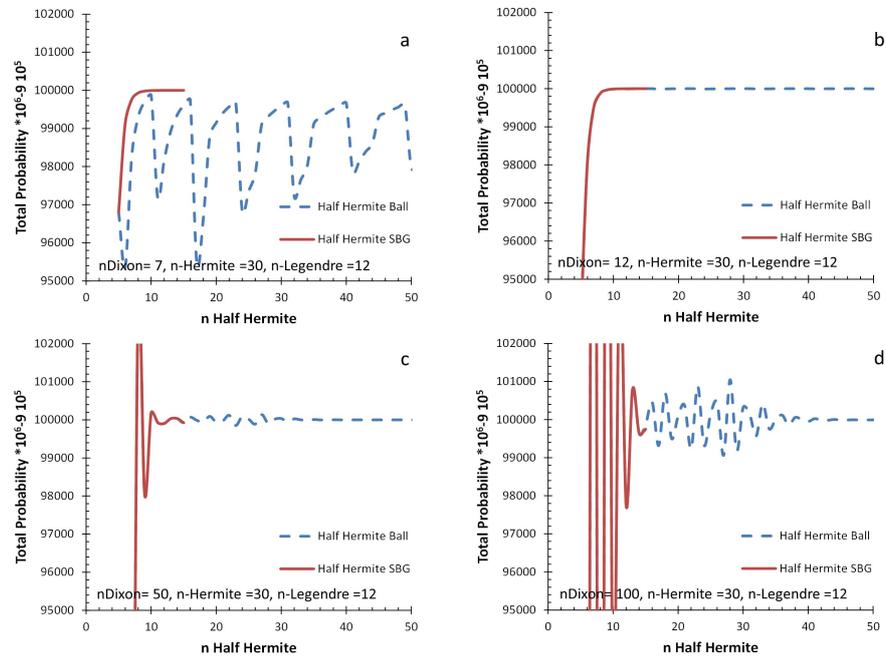


Figure 3: Variation of total probability with the degree of half Hermite polynomial to study the variation of total number of data points (i.e.,  $n$ Dixon = 7, 12, 50 and 100).

Table 2 presents the comparison of critical values for the ratio  $r_{10}$ . The total probability up to 8 decimal digits is also shown in the table. One has to take into account that the accuracy decreases with increasing total number of data points in the quadrature approach and is at least 4 significant digits for `nDixon` = 100. It can be observed that there is good agreement up to 4 decimal places between the calculated values using the Monte Carlo simulation (Verma *et al.* 2008) and the quadrature method (this work). However, the execution time for the calculation of critical values is in the order of seconds, whereas the Monte Carlo simulation requires months of execution time.

The requirement of accuracy in the critical values needed for analytical data treatment depends on two aspects: *total number of data points* and *precision of confidence interval*. Let us consider an example for the Dixon ratio  $r_{10}$ : `nDixon` = 30, `iDixon` = 1, `jDixon` = 1, `Alpha` = 0.05 (i.e., 95% confidence interval). The values of *n-Hermite*, *n-half Hermite*, *HermiteBall*, and *n-Legendre* are 30, 15, `False` and 10, respectively and the permitted uncertainty in the confidence interval be  $\pm 0.5\%$  (i.e., confidence interval is between 94.5% and 95.5%). The calculated critical values are 0.253756, 0.259449 and 0.265625 for the confidence intervals 94.5%, 95.0%, and 95.5%, respectively. Thus the critical values with two significant digits for the Dixon ratio test  $r_{10}$  are sufficient for a dataset with 30 data points.

Now, we repeat the above calculations for `nDixon` = 100 while keeping the same values for the other parameters. In this case the accuracy of the present quadrature approach is up to 4 significant digits. The calculated critical values are 0.180418, 0.184811 and 0.189590 for the confidence intervals 94.5%, 95.0%, and 95.5%, respectively. Thus the critical values with at least three significant digits for the Dixon ratio test  $r_{10}$  are needed for a dataset with 100 data points.

### 3.3. Using DixonTest.CriticalValues in Excel

The library (`DixonTest.dll`) can be used in any computer programming language in the Windows environment. However, its use in **Excel** to create critical values tables will be explained here; since most of the geochemists use **Excel** for geochemical data management and calculations.

The installation and uninstallation of library, `DixonTest.dll`, is explained in file `ReadMe.PDF` which is available as supplementary material.

Verma (2003) explained the procedure for writing a function in the personal workbook of **Excel**. A macro stored in this location will be available for all the workbooks. For using the class '`CriticalValues`' in the library **DixonTest** it is needed to perform the following steps:

1. *Add reference to library*: press the "Developer" tab on the **Excel** ribbon and then press "Visual Basic" button. It will open the Visual Basic Environment. Now, press menu Tools→References and set the references to the library, `DixonTest.tlb`, which sets the reference on the library, `DixonTest.dll`.
2. *Creating functions*: write the following code in a module in the personal workbook (`PERSONAL.XLSB`). There are two functions, `TotalProbability` and `RCriticalValue`. Save the personal workbook. Now, the function can be used in any workbook.

```
Public Function RCriticalValues(nDixon As Integer, iDixon As Integer, _
    jDixon As Integer, alpha As Double, nH As Integer, _
```

```
HH As Boolean, nHH As Integer, nL As Integer, _
RAcc As Integer) As Double
Dim DTest As New DixonCriticalValues2013.CriticalValues
With DTest
    .nDixon = nDixon
    .iDixon = iDixon
    .jDixon = jDixon
    .AlphaValue = alpha
    .nHermPoly = nH
    If HH = True Then
        .HalfHermiteBall = True
        .nHHermPoly = nHH
    Else
        .HalfHermiteBall = False
        If nHH > 15 Then
            .nHHermPoly = 15
        Else
            .nHHermPoly = nHH
        End If
    End If
    .nLegPoly = nL
    .RAccuracy = RAcc
End With
RCriticalValues = DTest.RCritical
End Function

Public Function TotalProbability(nDixon As Integer, iDixon As Integer, _
jDixon As Integer, nH As Integer, _
HH As Boolean, nHH As Integer, _
nL As Integer) As Double
Dim DTest As New DixonCriticalValues2013.CriticalValues
With DTest
    .nDixon = nDixon
    .iDixon = iDixon
    .jDixon = jDixon
    .AlphaValue = 1
    .nHermPoly = nH
    If HH = True Then
        .HalfHermiteBall = True
        .nHHermPoly = nHH
    Else
        .HalfHermiteBall = False
        If nHH > 15 Then
            .nHHermPoly = 15
        Else
            .nHHermPoly = nHH
        End If
    End If
End With
End Function
```

Critical Values Table								
			nHermite	30				
iDixon	1		nHHermite	50				
jDixon	1		HHermiteBall	1				
			nLegendre	12				
			Raccuracy	6				
nDixon\alpha	0.30	0.25	0.02	0.10	0.05	0.02	0.01	Total Prob
5	0.373193	0.409925	0.729263	0.558094	0.642357	0.729263	0.780991	0.99999970
10	0.220620	0.245187	0.481339	0.348949	0.411860	0.481339	0.526268	0.99999966
15	0.176155	0.196609	0.399002	0.284395	0.338539	0.399002	0.438454	0.99999999
20	0.153690	0.171956	0.355995	0.251136	0.300499	0.355995	0.392395	0.99999999
25	0.139697	0.156560	0.328682	0.230169	0.276421	0.328682	0.363083	1.00000002
30	0.129950	0.145814	0.309387	0.215437	0.259449	0.309387	0.342354	1.00000005
35	0.122669	0.137777	0.294815	0.204359	0.246658	0.294815	0.326683	1.00000007
40	0.116967	0.131474	0.283291	0.195635	0.236563	0.283291	0.314285	1.00000006
45	0.112343	0.126360	0.273876	0.188529	0.228327	0.273876	0.304146	0.99999982
50	0.108496	0.122099	0.265991	0.182592	0.221436	0.265991	0.295647	0.99999943
55	0.105227	0.118478	0.259251	0.177532	0.215551	0.259251	0.288383	0.99999933
60	0.102405	0.115348	0.253395	0.173148	0.210451	0.253395	0.282066	0.99999980
65	0.099934	0.112608	0.248249	0.169301	0.205967	0.248249	0.276512	1.00000068
70	0.097747	0.110180	0.243668	0.165887	0.201984	0.243668	0.271561	1.00000138
75	0.095793	0.108010	0.239559	0.162828	0.198414	0.239559	0.267120	1.00000135
80	0.094031	0.106053	0.235844	0.160068	0.195187	0.235844	0.263107	1.00000043
85	0.092433	0.104278	0.232464	0.157557	0.192253	0.232464	0.259453	0.99999888
90	0.090974	0.102655	0.229370	0.155262	0.189568	0.229370	0.256104	0.99999730
95	0.089635	0.101166	0.226521	0.153152	0.187096	0.226521	0.253029	0.99999633
100	0.088399	0.099791	0.223885	0.151201	0.184811	0.223885	0.250179	0.99999642

Figure 4: **Excel** worksheet to illustrate the calculation of critical values tables. The values for  $nDixon = 5$  and 10 are calculated using half Hermite SBG.

```

End If
End If
.nLegPoly = nL
End With
TotalProbability = DTest.TotalProbability
End Function

```

Figure 4 shows the calculation of critical values tables in an **Excel** worksheet. The user has to change the values of `iDixon` and `jDixon` to get the corresponding critical values table. Similarly, the values of `nDixon` can be changed according to the need. Once, the table is calculated, one has to copy and save it in another worksheet with “save as value type”. The **Excel** workbook is also available as supplementary material. Similarly, the calculation of total probability to know the accurate digit in the critical values is shown in the last column.

## 4. Conclusions

The present computer program **DixonTest.CriticalValues** is fast, flexible and efficient for the calculation of critical values with accuracy up to 6 significant digits for the Dixon statistical tests. The accuracy in the critical values is estimated through the calculation of the total probability density and is considered as one digit less than the total accurate digits in the total probability. The accuracy of three digits in the critical values is sufficient for the statistical treatment of an analytical data set up to 100 points. The program can be easily transported to other environments like **Excel**.

## Acknowledgments

The first author highly appreciates George C. McBane for explaining all the minute details of his computer program for the calculation of the critical values and providing all the relevant information for the development of the present computer program and Surendra P. Verma for introducing him with this subject. The authors appreciate the reviewers for critical reading and providing constructive comments to improve the content of this article.

## References

- Ball JS (2003). “Half-Range Generalized Hermite Polynomials and the Related Gaussian Quadratures.” *SIAM Journal of Numerical Analysis*, **40**(6), 2311–2317.
- Dean RB, Dixon WJ (1951). “Simplified Statistics for Small Numbers of Observations.” *Analytical Chemistry*, **23**(4), 636–638.
- Dixon WJ (1950). “Analysis of Extreme Values.” *The Annals of Mathematical Statistics*, **21**(4), 488–506.
- Dixon WJ (1951). “Ratios Involving Extreme Values.” *The Annals of Mathematical Statistics*, **22**(1), 68–78.
- Efstathiou CE (1992). “Stochastic Calculation of Critical Q-Test Values for the Detection of Outliers in Measurements.” *Journal of Chemical Education*, **69**(9), 733–736.
- Gautschi W (1994). “Algorithm 726: **ORTHPOL** – A Package of Routines for Generating Orthogonal Polynomials and Gauss-Type Quadrature Rules.” *ACM Transactions on Mathematical Software*, **20**(1), 21–62.
- Komsta L (2006). “Processing Data for Outliers.” *R News*, **6**(2), 10–13. URL <http://CRAN.R-project.org/doc/Rnews/>.
- Kumar PBS (2011). *Numerical Methods and Programming*. IIT Madras NPTEL Course, India. URL <http://freevidelectures.com/Course/2663/Numerical-Methods-and-Programing.html>.
- Lohglinger H (1999). *Teach/Me – Data Analysis*. Springer-Verlag.
- McBane GC (2006). “Programs to Compute Distribution Functions and Critical Values for Extreme Value Ratios for Outlier Detection.” *Journal of Statistical Software*, **16**(3), 1–9. URL <http://www.jstatsoft.org/v16/i03/>.
- Miller JN, Miller JC (2005). *Statistics and Chemometrics for Analytical Chemistry*. 5th edition. Prentice Hall, Essex.
- Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1992). *Numerical Recipes in C: The Art of Scientific Computing*. 2nd edition. Cambridge University Press, Cambridge.
- R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>.

- Riley KF, Hobson MP, Bence SJ (2006). *Mathematical Methods for Physics and Engineering*. 3rd edition. Cambridge University Press, Cambridge.
- Rorabacher DB (1991). “Statistical Treatment for Rejection of Devian Values Critical Values of Dixon’s  $Q$  Parameter and Related Subrange Ratios at the 95% Confidence Level.” *Analytical Chemistry*, **63**(2), 139–146.
- Steen NM, Byrne GD, Geldard EM (1969). “Gaussian Quadratures for the Integrals  $\int_0^\infty \exp(-x^2) dx$  and  $\int_0^b \exp(-x^2) f(x) dx$ .” *Mathematics of Computation*, **23**(107), 661–671.
- Venkat (2012). *.NET Program Execution*. URL <http://csharp-video-tutorials.blogspot.co.uk/2012/07/net-program-execution-part-1.html>.
- Verma MP (2003). “Steam Tables for Pure Water as an ActiveX Component in Visual Basic 6.0.” *Computers & Geosciences*, **29**(9), 1155–1163.
- Verma SP, Quiroz-Ruiz A, Diaz-Gonzalez L (2008). “Critical Values for 33 Discordancy Test Variants for Outliers in Normal Samples up to Sizes 1000, and Applications in Quality Control in Earth Sciences.” *Revista Mexicana de Ciencias Geologicas*, **25**(1), 82–96.
- Weisstein EW (2011). *Hermite-Gauss Quadrature*. From *MathWorld – A Wolfram Web Resource*. URL <http://mathworld.wolfram.com/Hermite-GaussQuadrature.html>.

**Affiliation:**

Mahendra P. Verma  
Geotermia  
Instituto de Investigaciones Electricas  
Reforma 113, Col. Palmira, Cuernavaca, Mor., C.P. 62490  
México  
E-mail: [mahendra@iie.org.mx](mailto:mahendra@iie.org.mx)  
URL: <http://www.iie.org.mx/>

Mario Cesar Suarez A.  
Facultad de Ciencias-UMSNH  
Universidad Michoacana  
Morelia, México  
URL: <http://www.fismat.umich.mx/~marioc/>