BDrive®
Power for Intelligent Collaborative Infrastructures™

Analytics Process Modules for NetCharts Designer/Server

About Process Modules

A typical user of advanced charting software such as NetCharts often wants to perform some rapid analysis of the data to gain a general understanding of what the numbers mean. The analytics process modules developed for use with NetCharts Designer/Server are designed to help such a user gain valuable insight into what the data represents without having to transport it to a separate analytics tool.

The process modules described here are divided into three groups: core, functional and math. The core modules perform very basic operations on one or more columns of numbers. The functional modules are designed to perform more specialized operations of univariate and bivariate analysis such as Histogram, Pareto, Regression and Integration. In many cases the functional modules may utilize one or more of the core modules. Math modules support mathematical operations like complex number arithmetic, vector manipulations, matrix methods and quaternion math.

All the core and functional modules are described in detail in the section below. Unit tests with sample data are provided in order to illustrate the operation of each module.

The Appendix provides additional information about Regression, Fourier analysis and Univariate Metrics.

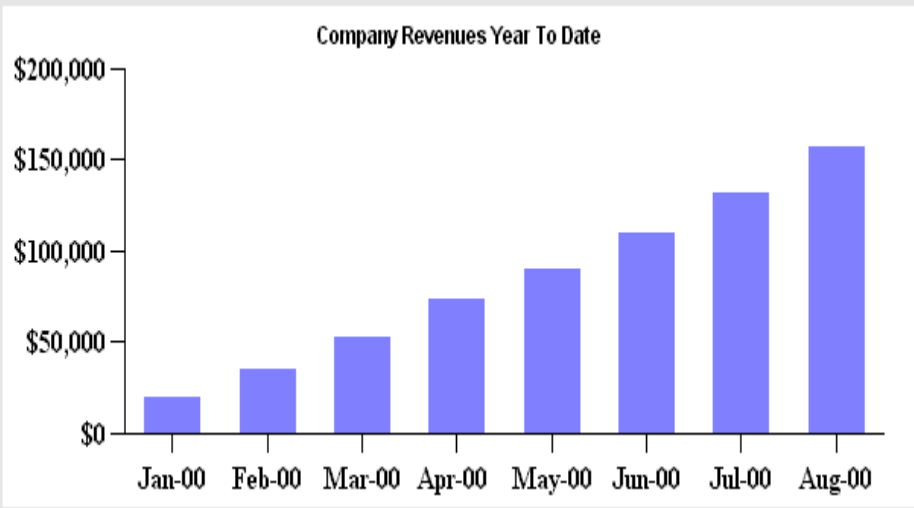
Description of Process Modules

<table>
<tr><td>Module Name</td><td>Aggregate Column</td><td>Module Type</td><td>Core</td></tr>
<tr><td>Description</td><td colspan="3">Cumulative sum of the numbers in a column of data from top to bottom returned in a column of equal length. The line-by-line aggregate may also be displayed as a percentage of the total sum of all the numbers.</td></tr>
<tr><td>Parameters</td><td colspan="3">
<table>
<tr><th>Parameter Name</th><th>Description</th></tr>
<tr><td>Column number</td><td>Column on which to perform the aggregate operation. Column numbers begin with zero</td></tr>
<tr><td>Option</td><td>Display result as cumulative SUM or PERCENTAGE</td></tr>
<tr><td>Include Original Data</td><td>If “Yes” include the original dataset in the result table</td></tr>
</table>
</td></tr>
</table>

BDrive®
Power for Intelligent Collaborative Infrastructures™

Unit Tests

Input	Parameter	Result
Month,Revenue Jan-00, 20000 Feb-00, 15000 Mar-00, 18000 Apr-00 , 21000 May-00, 16000 Jun-00, 20000 Jul-00, 22000 Aug-00, 25000	Column number = 1 Option = SUM Include Original Data = No	Result 20000.0 35000.0 53000.0 74000.0 90000.0 110000.0 132000.0 157000.0



Module Name	Bin Data	**Module Type**	Core
Description	Bin column numbers to obtain frequency of numbers in a certain range. If a column of data contains numbers that range from 1 to 1000 for instance, create bins of a certain width (eg.100) and count how many column items fall into each range (i.e. 1-100, 101-200 and so on). The bin width may be set automatically or be specified by the user. The user may provide the actual width of each bin or, alternatively, specify the total number of bins.		

Parameters

Parameter Name	Description
Column number	Column on which to perform the bin operation. Column numbers begin with zero

Bin type	This parameter determines how the bins will be set up. The options are Automatic, By Number, By Width
Bin size	If Bin type is “Automatic”, a bin size will be calculated automatically and need not be specified by the user If Bin type is “By Number”, bin size is the number of bins into which to the total data range is divided If Bin type is “By Width”, bin size refers to the width of the data range into which each column item will be binned
Min range	Beginning point of first data range. Will be calculated if not provided by user.
Max range	End point of last data range. Will be calculated if not provided by user.
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter
X, Y 20,55 100,32 63,45 5,23 55,24 26,72 75,75 10,110 20,221	Column number = 1 Bin type = By Width Bin size = 50 Min range = 0 Max range = 250 Include Original Data = No

Result

Bin Name	Bin Min	Bin Max	Bin Frequency
Less	-1.0E30	0.0	0
Bin1	0.0	50.0	4
Bin2	50.0	100.0	3
Bin3	100.0	150.0	1
Bin4	150.0	200.0	0
Bin5	200.0	250.0	1
More	250.0	1.0E30	0
null	null	null	null
null	null	null	null

<table>
<tr><td>Module Name</td><td>Grade Column</td><td>Module Type</td><td>Core</td></tr>
<tr><td>Description</td><td colspan="3">The relative importance or weight of a number in a data set with reference to the whole set is calculated by percentage of the total sum, by probability (normalized so that total sum = 1), by percentile (normalized so that the maximum value = 100) or by rank.</td></tr>
<tr><td>Parameters</td><td colspan="3">
<table>
<tr><th>Parameter Name</th><th>Description</th></tr>
<tr><td>Column number</td><td>Column on which to perform the grade column operation. Column numbers begin with zero</td></tr>
<tr><td>Option</td><td>PERCENTAGE, PROBABILITY, PERCENTILE or RANK</td></tr>
<tr><td>Include Original Data</td><td>If “Yes” include the original dataset in the result table</td></tr>
</table>
</td></tr>
<tr><td>Unit Tests</td><td colspan="3">
<table>
<tr><th>Input</th><th>Parameter</th><th>Result</th></tr>
<tr><td>x
1
2
3
4
5</td><td>Column number = 1

Option = Various
Include Original Data = Yes</td><td>
<table>
<tr><th>x</th><th>Result</th><th>Result</th><th>Result</th><th>Result</th></tr>
<tr><td>1</td><td>0</td><td>0.0666667</td><td>6.6666667</td><td>5</td></tr>
<tr><td>2</td><td>25</td><td>0.1333333</td><td>13.333333</td><td>4</td></tr>
<tr><td>3</td><td>50</td><td>0.2</td><td>20</td><td>3</td></tr>
<tr><td>4</td><td>75</td><td>0.2666667</td><td>26.666667</td><td>2</td></tr>
<tr><td>5</td><td>100</td><td>0.3333333</td><td>33.333333</td><td>1</td></tr>
</table>

The above table shows the results for various options for the same dataset. The first result column corresponds to option Percentile, second column to Probability, third column to Percentage and fourth column to Rank.
</td></tr>
</table>
</td></tr>
</table>

Module Name	Delete Column/s	**Module Type**	Core
Description	Delete selected columns from data set.		

Parameters

Parameter Name	Description
Column to delete	FIRST, LAST, OTHER
Column index	Comma separated list of column indices if delete column choice is OTHER

Unit Tests

Input	Parameter	Result
X Y A B -0.5 0.5 -0.5 0.0 0.5 0.5 0.5 1.0 0.5 -0.5 0.5 2.0 -0.5 -0.5 -0.5 3.0 -0.5 0.5 -0.5 4.0	Column to delete = OTHER Column index = 2	X Y B -0.5 0.5 0.0 0.5 0.5 1.0 0.5 -0.5 2.0 -0.5 -0.5 3.0 -0.5 0.5 4.0

Module Name	Delete Row/s	**Module Type**	Core
Description	Delete selected rows from data set.		

Parameters

Parameter Name	Description
Row to delete	FIRST, LAST, OTHER
Row index	Comma separated list of row indices if delete row choice is OTHER

Unit Tests

Input	Parameter	Result
X Y A B -0.5 0.5 -0.5 0.0 0.5 0.5 0.5 1.0 0.5 -0.5 0.5 2.0 -0.5 -0.5 -0.5 3.0 -0.5 0.5 -0.5 4.0	Row to delete = LAST	X Y A B -0.5 0.5 -0.5 0.0 0.5 0.5 0.5 1.0 0.5 -0.5 0.5 2.0 -0.5 -0.5 -0.5 3.0

Module Name	Include Headers	Module Type	Core
Description	Add new header row or replace existing data column headers with user-specified comma-separated list.		

Parameters

Parameter Name	Description
Column Action	MODIFY or ADD
Column names	Comma-separated list of new column headers

Unit Tests

Input	Parameter	Result
X Y A B -0.5 0.5 -0.5 0.0 0.5 0.5 0.5 1.0 0.5 -0.5 0.5 2.0 -0.5 -0.5 -0.5 3.0 -0.5 0.5 -0.5 4.0	Column Action = ADD Column names = 1,2,3,4	1 2 3 4 X Y A B -0.5 0.5 -0.5 0.0 0.5 0.5 0.5 1.0 0.5 -0.5 0.5 2.0 -0.5 -0.5 -0.5 3.0 -0.5 0.5 -0.5 4.0

Module Name	Count Group Names	Module Type	Core
Description	Group a column of strings or names by content and count the number of occurrences of each string. The result is returned in two columns listing each distinct string and the corresponding number of occurrences side by side. The columns are sorted by ascending or descending count.		

Parameters

Parameter Name	Description
Column	Column on which to perform the group operation. Column numbers begin with zero
Sort	Sort data by ascending (ASC) or descending (DESC) count. Select No Sort to do no sorting
Include Original	If “Yes” include the original

	Data: dataset in the result table

Unit Tests

Input	Parameter	Result
Col0 ,Hours Good,5 Good,3 Better,2 Best,10 Good,1 Better,1	Column = 0 Option = ASC Include Original Data = Yes	Col0 Hours Name Count Good 5 Best 1 Good 3 Better 2 Better 2 Good 3 Best 10 null null Good 1 null null Better 1 null null

Module Name	Aggregate Group Names	**Module Type**	Core

Description

Group a column of strings or names by content. For each string, sum the corresponding numbers in another column. The result is returned in two columns listing each distinct string and the corresponding sum side by side. The columns are sorted by ascending or descending sum.

Parameters

Parameter Name	Description
Column	Column used to group by string. Column numbers begin with zero
Column to Aggregate	Column used to sum numbers corresponding to content of first column. Column numbers begin with zero
Option	Sort data by ascending (ASC) or descending (DESC) count or No Sort
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
Col0 ,Hours Good,5 Good,3 Better,2 Best,10 Good,1 Better,1	Column = 0 Column to Aggregate = 1 Option = DESC Include Original Data = Yes	Col0 Hours Name Sum Good 5 Best 10.0 Good 3 Good 9.0 Better 2 Better 3.0 Best 10 null null Good 1 null null Better 1 null null

Example with no sort

Input	Parameter	Result
Location,Count Sunnyvale,1 Sunnyvale,2 San Jose,4 Sunny,1 SJSU,1 San Francisco,1 Alameda,10 Cupertino,1 Sunnyvale,2	Column = 0 Column to Aggregate = 1 Sort = No Sort Include Original Data = No	Name Sum Sunnyvale 5.0 San Jose 4.0 Sunny 1.0 SJSU 1.0 San Francisco 1.0 Alameda 10.0 Cupertino 1.0

Module Name	Sort 1 Column	**Module Type**	Core

Description

Sort a column of numbers in ascending or descending order and return the result in a column of equal length

Parameters

Parameter Name	Description
Column number	Column on which to perform the sort operation. Column numbers begin with zero
Option	Specify ascending (ASC) or descending (DESC) sort
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
X, Y 20,55 100,32 63,45 5,23 55,24 26,72 75,75 10,110 20,221	Column number = 1 Option = DESC Include Original Data = No	Result 221 110 75 72 55 45 32 24 23

<table>
<tr><td>Module Name</td><td>Two Column Operation</td><td>Module Type</td><td>Core</td></tr>
<tr><td>Description</td><td colspan="3">Perform item by item addition, subtraction, multiplication or division with two columns of data of equal length and return the result in a column of the same length as the input columns</td></tr>
<tr><td>Parameters</td><td colspan="3">
<table>
<tr><th>Parameter Name</th><th>Description</th></tr>
<tr><td>Column 1</td><td>First column on which to perform the two-column operation. Column numbers begin with zero</td></tr>
<tr><td>Column 2</td><td>Second column on which to perform the two-column operation. Column numbers begin with zero</td></tr>
<tr><td>Operation</td><td>ADD, SUBTRACT, MULTIPLY or DIVIDE</td></tr>
<tr><td>Result Column Name</td><td>Name of result column</td></tr>
<tr><td>Include Original Data</td><td>If “Yes” include the original dataset in the result table</td></tr>
</table>
</td></tr>
<tr><td>Unit Tests</td><td colspan="3">
<table>
<tr><th>Input</th><th>Parameter</th><th>Result</th></tr>
<tr><td>X,Y
0,0
1,1
2,0
3,-1
4,0</td><td>Column 1 = 0

Column 2 = 1

Operation = ADD

Result Column = Sum

Include Original Data = No</td><td>Sum
0
2
2
2
4</td></tr>
</table>
</td></tr>
</table>

Module Name	N Column Operation	**Module Type**	Core
Description	Perform item by item addition and multiplication on multiple columns of data of equal length and return the result in a column of the same length as the input columns		

Parameters

Parameter Name	Description
Columns	Comma separated list of columns on which to perform the operation
Operation	ADD or MULTIPLY
Result Column Name	Name of result column
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
Col0,Col1,Col2 1,2,3 4,5,6	Columns = 0,1,2 Operation = ADD/MULIPLY Result Column = Sum Include Original Data = Yes	Col0 Col1 Col2 ADD012 MULT012 1 2 3 6.0 6.0 4 5 6 15.0 120.0

Module Name	Univariate Metrics	**Module Type**	Functional
Description	Calculate basic metrics such as count, sum, average, minimum, maximum, variance and standard deviation for a column of numbers. The above metrics are calculated when the option BASIC is selected. The user may also select the option ALL to calculate kurtosis, skewness and standard error of the mean in addition to the above metrics. The formulas used for the calculations are provided in the appendix. Any null values in the input data are ignored.		

Parameters

Parameter Name	Description
Input Column	Column for which to calculate basic metrics. Column numbers begin with zero
Option	The option ALL or BASIC determines exactly what metrics will be calculated
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
X 1 6 3 0 5 9 5 2 3 1 7	Input Column = 0 Option = ALL Include Original Data = Yes	X Metric Value 1 Count 11.0 6 Sum 42.0 3 Average 3.8181818181818183 0 Min 0.0 5 Max 9.0 9 Median 3.0 5 Variance 7.963636363636364 2 SDev 2.821991559809555 3 Kurtosis -1.1069562201106586 1 Skewness 0.34487183398873555 7 Standard Error 0.8508624695852793

Example with null data

Input	Parameter	Result
X 101 null 29 23 34 55 221 150 160 null null	Input Column = 0 Option = Basic Include Original Data = Yes	X Metric Value 101 Count 8.0 null Sum 773.0 29 Average 96.625 23 Min 23.0 34 Max 221.0 55 Median 78.0 221 Variance 4750.234375 150 SDev 68.9219440744383 160 null null null null null null null null

<table>
<tr><td>Module Name</td><td>Histogram Distribution</td><td>Module Type</td><td>Functional</td></tr>
<tr><td>Description</td><td colspan="3">Similar to the Bin Data module, Histogram Distribution also bins column numbers to obtain frequency of numbers in a certain range. In addition, it provides the option to bin data by probability (total probability summed over whole range of data is 1) or by percentage (total percentage summed over whole range of data is 100). The bin width may be set automatically or be specified by the user. The user may provide the actual width of each bin or, alternatively, specify the total number of bins.</td></tr>
<tr><td>Parameters</td><td colspan="3">
<table>
<tr><th>Parameter Name</th><th>Description</th></tr>
<tr><td>Column number</td><td>Column on which to perform the histogram operation. Column numbers begin with zero</td></tr>
<tr><td>Bin type</td><td>This parameter determines how the bins will be set up. The options are Automatic, By Number, By Width</td></tr>
<tr><td>Histogram size</td><td>If Bin type is “Automatic”, a histogram size will be calculated automatically and need not be specified by the user

If Bin type is “By Number”, histogram size is the number of bins into which to the total data range is divided

If Bin type is “By Width”, histogram size refers to the width of the data range into which each column item will be binned</td></tr>
<tr><td>Min range</td><td>Beginning point of first data range. Will be calculated if not provided by user.</td></tr>
<tr><td>Max range</td><td>End point of last data range. Will be calculated if not provided by user.</td></tr>
<tr><td>Histogram type</td><td>Histogram result is returned By Number, Percentage, or Probability</td></tr>
<tr><td>Include Original Data</td><td>If “Yes” include the original dataset in the result table</td></tr>
</table>
</td></tr>
</table>

Unit Tests

Input	Parameter
val 1 2 2 3 3 3 4 4 4 4 5 5 5 6 6 7	Column number = 0 Bin type = By Width Bin size = 1 Histogram type = By Number, Percentage, Probability (displayed in successive columns) Include Original Data = Yes

Result

X	null	Bin	Histogram	Bin	Histogram	Bin	Histogram	Bin	Histogı
1	null	1.0	0.375	1.0	0.0625	1.0	0.1875	1.0	18
2	null	2.0	0.4375	2.0	0.125	2.0	0.1875	2.0	18
2	null	3.0	0.1875	3.0	0.1875	3.0	0.25	3.0	2
3	null	null	null	4.0	0.25	4.0	0.1875	4.0	18
3	null	null	null	5.0	0.1875	5.0	0.125	5.0	1
3	null	null	null	6.0	0.125	6.0	0.0625	6.0	6
4	null	null	null	7.0	0.0625	null	null	null	
4	null	null	null	null	null	null	null	null	
4	null	null	null	null	null	null	null	null	
4	null	null	null	null	null	null	null	null	
5	null	null	null	null	null	null	null	null	
5	null	null	null	null	null	null	null	null	
5	null	null	null	null	null	null	null	null	
6	null	null	null	null	null	null	null	null	
6	null	null	null	null	null	null	null	null	
7	null	null	null	null	null	null	null	null	

Module Name	Pareto Analysis	**Module Type**	Functional

Description

Given a set of causes that lead to problems in a certain area, the Pareto Principle states that in many cases only a "vital few" factors (about 20%) are responsible for producing most (about 80%) of the problems. To help identify the major components in a set of data, Pareto analysis sorts a column of data and accompanying labels in descending order of magnitude and provides the cumulative percentage contribution that each additional item makes to the total. The (x,y) values for an 80% y line and a 20% x line are also provided. Plotting the points at which the cumulative percentage line crosses the 80% and 20% lines allows the user to note what percentage of items contribute to 80% of the total and what percentage of the total is made up of 20% of the items respectively.

In the output two additional columns called LHS and RHS are provided. These columns provide information about the min and max values for the Left scale and right scale respectively.

Parameters

Parameter Name	Description
Category Names Column	Column that holds item labels. Column numbers begin with zero
Occurrences Column	Column that holds item values. Column numbers begin with zero
Top Extent	Number of items to be returned in the result. Setting this parameter to a small number results in data being provided for only the first few major items

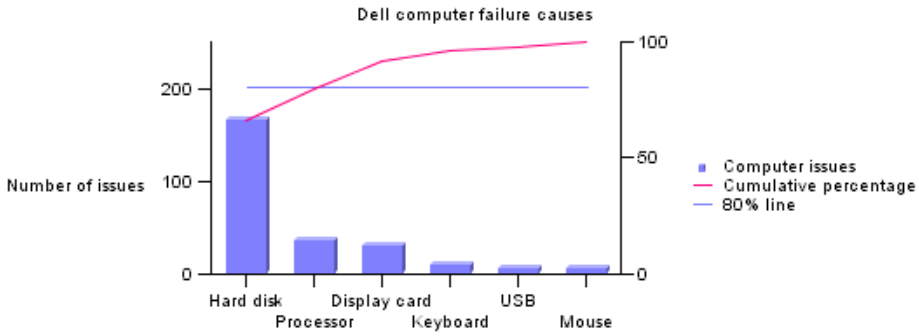
Unit Tests

Input	Parameter
Cause,Issues Processor, 35 Display card, 30 Keyboard, 10 Hard disk, 165 Mouse,5 USB, 5	Category Names Column = 0 Occurrences Column = 1 Top Extent = 6

Result

Cause	Issues	Cumulative Percentage	Line 80 X	Line 80 Y	Line 20 X	Line 20 Y	LHS	RH
Processor	35	66.0	0.0	80.0	1.7	0.0	0.0	0.
Display card	30	80.0	1.0	80.0	1.7	16.6666666 66666668	250.0	100.
Keyboard	10	92.0	2.0	80.0	1.7	33.3333333 33333336	null	nu
Hard disk	165	96.0	3.0	80.0	1.7	50.0	null	nu
Mouse	5	98.0	4.0	80.0	1.7	66.6666666 6666667	null	nu
USB	5	100.0	5.0	80.0	1.7	100.0	null	nu

The above results, except for the 20% line, are plotted below.

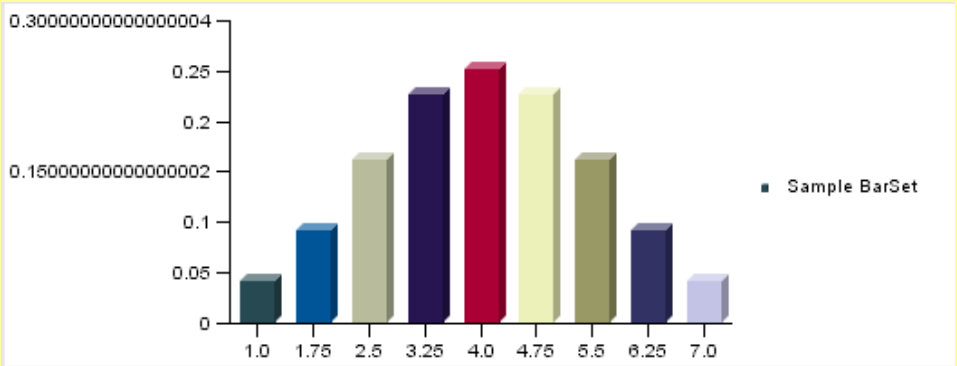


Module Name	Normal Distribution	**Module Type**	Functional
Description	Given a column of data, the Normal Distribution module calculates the normal probability distribution or “bell curve” obtained from the mean and standard deviation of the data set for the default analysis type “Probability”. If analysis type is “Cumulative”, the cumulative probability over the range of data is returned in the result columns. Analysis type “Inverse” is identical to cumulative except for the fact that the X and Y columns are interchanged for ease of plotting a horizontally oriented cumulative probability.		
Parameters			

Parameter Name	Description
Column	Column for which to calculate the normal distribution. Column numbers begin with zero
Number of Points	The number of points within the data range for which to evaluate the normal distribution
Analysis Type	Cumulative, Inverse or Probability
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
val 1 2 2 3 3 3 4 4 4 4 5 5 5 6 6 7	Column = 0 Number of points = 9 Original dataset = No Analysis Type = Probability	Nx Ny 1 0.0417071 1.75 0.0916678 2.5 0.160882 3.25 0.2254665 4 0.2523133 4.75 0.2254665 5.5 0.160882 6.25 0.0916678 7 0.0417071 The plot below shows the above Normal distribution superposed on a histogram of the same data set



Module Name	Area	**Module Type**	Functional
Description	Perform cumulative area-under-the-curve computation with two columns of data of equal length. The result will be returned in two columns (X and resultant Area) of the same length as the input columns.		

Parameters

Parameter Name	Description
Column X	X column for the area operation. Column numbers begin with zero
Column Y	Y column for the area operation. Column numbers begin with zero
Area Option	AGGREGATE/INDIVIDUAL
Result Column X Name	Name of result column X
Result Column Y Name	Name of result column Y containing the Area computation
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
x,y 0,1 1,1 2,1 3,1 4,1	Column X = 0 Column Y = 1 Area Option = AGGREGATE (agg) or INDIVIDUAL (sliver) Result Column X Name = sliverx/aggx Result Column Y Name = slivery/aggy Include Original Data = YES	x y sliverx slivery aggx aggy 0 1 0.0 0.0 0.0 0.0 1 1 1.0 1.0 1.0 1.0 2 1 2.0 1.0 2.0 2.0 3 1 3.0 1.0 3.0 3.0 4 1 4.0 1.0 4.0 4.0

Module Name	Differentiation	**Module Type**	Functional
Description	Perform cumulative differentiation computation with two columns of data of equal length. The result will be returned in a column of the same length as the input columns.		

Parameters

Parameter Name	Description
Column X	X column for the differentiation operation. Column numbers begin with zero
Column Y	Y column for the differentiation operation. Column numbers begin with zero
Result Column X Name	Name of result column X
Result Column Y Name	Name of result column Y
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
X,Y 0,0 0.5,0.5 1,1 1.5,1.5 2,2	Column X = 0 Column Y = 1 Result Column X Name = DX Result Column Y Name = DY Include Original Data = Yes	X Y DX DY 0 0 0.0 1.0 0.5 0.5 0.5 1.0 1 1 1.0 1.0 1.5 1.5 1.5 1.0 2 2 2.0 1.0

Module Name	Integration	**Module Type**	Functional
Description	Perform cumulative numerical integration with two columns of data of equal length. The result will be returned in a column of the same length as the input columns. The user-provided constant value will be added to the result.		

Parameters

Parameter Name	Description
Column X	X column for the integration operation. Column numbers begin with zero
Column Y	Y column for the integration operation. Column numbers begin with zero
Constant for Integration	Constant value to be added to the result of integration
Result Column X Name	Name of result column X
Result Column Y Name	Name of result column Y
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
X,Y 0,0 1,1 2,0 3,-1 4,0	Column X = 0 Column Y = 1 Result Column X Name = IX Result Column Y Name = IY Include Original Data = Yes	X Y IX IY 0 0 0.0 0.0 0.5 0.5 0.5 0.125 1.0 1.0 1.0 0.5 1.5 1.5 1.5 1.125 2.0 2.0 2.0 2.0

Module Name	Curve Length	**Module Type**	Functional
Description	Perform cumulative computation of the length of a 2 dimensional curve assuming that the x and y points respectively are provided in two columns of equal length. The result will be returned in a column of the same length as the input columns.		

Parameters

Parameter Name	Description
Column X	X column for the curve length operation. Column numbers begin with zero
Column Y	Y column for the curve length operation. Column numbers begin with zero
Result Column X Name	Name of result column X
Result Column Y Name	Name of result column Y
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
X,Y -0.5,0.5 0.5,0.5 0.5,-0.5 -0.5,-0.5 -0.5,0.5	Column X = 0 Column Y = 1 Result Column X Name = CLX Result Column Y Name = CLY Include Original Data = Yes	X Y CLX CLY -0.5 0.5 -0.5 0.0 0.5 0.5 0.5 1.0 0.5 -0.5 0.5 2.0 -0.5 -0.5 -0.5 3.0 -0.5 0.5 -0.5 4.0

Module Name	Surface Metrics	**Module Type**	Functional
Description	Compute data point count, curve length and surface area of a 2 dimensional curve assuming that the x and y points respectively are provided in two columns of equal length.		

Parameters

Parameter Name	Description
Column for X	X column for the surface metrics operation. Column numbers begin with zero
Column for Y	Y column for the surface metrics operation. Column numbers begin with zero
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
X Y -0.5 0.5 0.5 0.5 0.5 -0.5 -0.5 -0.5 -0.5 0.5	Column for X = 0 Column for Y = 1 Include Original Data = Yes	X Y Metric Value -0.5 0.5 Count 5 0.5 0.5 Curve Length 4.0 0.5 -0.5 Surface Area 1.0 -0.5 -0.5 null null -0.5 0.5 null null

Module Name	Regression	**Module Type**	Functional

Description

Regression is performed in order to find a mathematical equation that closely represents a set of paired (x,y) data. This allows one to estimate y values for x points other than the ones present in the original data set. This module fits two columns of related data (x,y pairs) to an equation of the form y = f(x), where the function f(x) is determined by the type and order of regression selected. The result x column returns a new set of x values (the original set or a smoothened one). The result y column returns the function f(x) evaluated at the new x points. The new (x,y) pairs need not include the original ones.

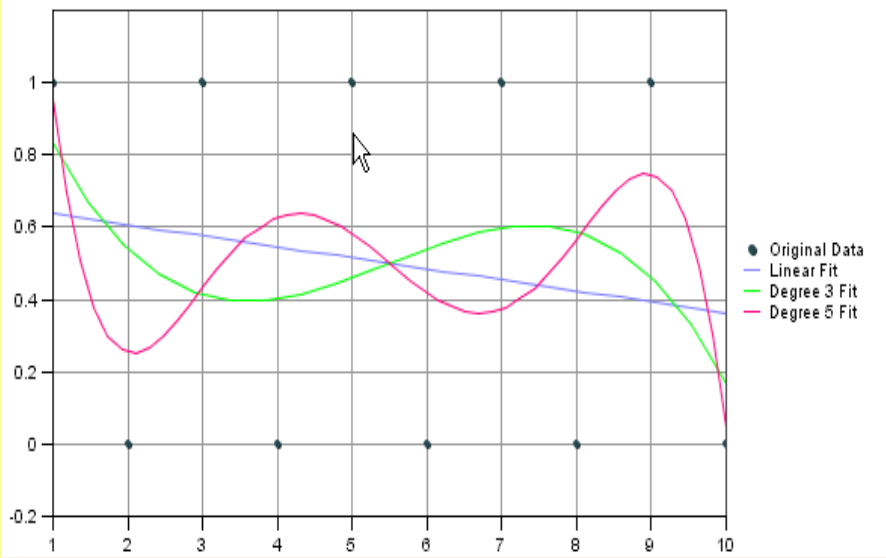
Parameters

Parameter Name	Description
Column X	X column for the regression operation. Column numbers begin with zero
Column Y	Y column for the regression operation. Column numbers begin with zero
Operation	Type of regression equation to which the data will be fit. POLYNOMIAL , EXPONENTIAL , HYPERBOLA , LOGARITHMIC, INVERSE, INVERSE_POWER, POLYEXP, POWER (*Refer to Appendix for regression equation*)
Degree	Polynomial degree applicable only to those regression equations that are expressed as a polynomial series (POLYNOMIAL, INVERSE, LOGARITHMIC, POLYEXP)
Smoothen	Total number of (x,y) pairs to be returned in the result columns. This number should be larger than the original data set.
Result Column X Name	Name of result column X
Result Column Y Name	Name of result column Y
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
X,Y,Z 0,0,1 1,1,0 2,2,1 3,3,0 4,4,1 5,5,0	Column X = 0 Column Y = 2 Operation = POLYNOMIAL Degree = 1, 3 or 5 Smoothen = 20	See Plot displayed below.

In the plot shown below, the original data is displayed along with

	polynomial fits of degree 1, 3 and 5. 

Module Name	Fourier	**Module Type**	Functional
Description	Fourier regression is performed in order to find a mathematical equation that closely represents a set of paired (x,y) data that is periodic and can be well represented by a Fourier series. This allows one to estimate y values for x points other than the ones present in the original data set. This module fits two columns of related data (x,y pairs) to an equation of the form ($f(x) = b_0 + b_1\cos(x) + b_2\sin(x) + b_3\cos(2*x) + b_4\sin(2*x) + ...$). The result x column returns a new set of x values (the original set or a smoothened one). The result y column returns the function f(x) evaluated at the new x points. The new (x,y) pairs need not include the original ones.		

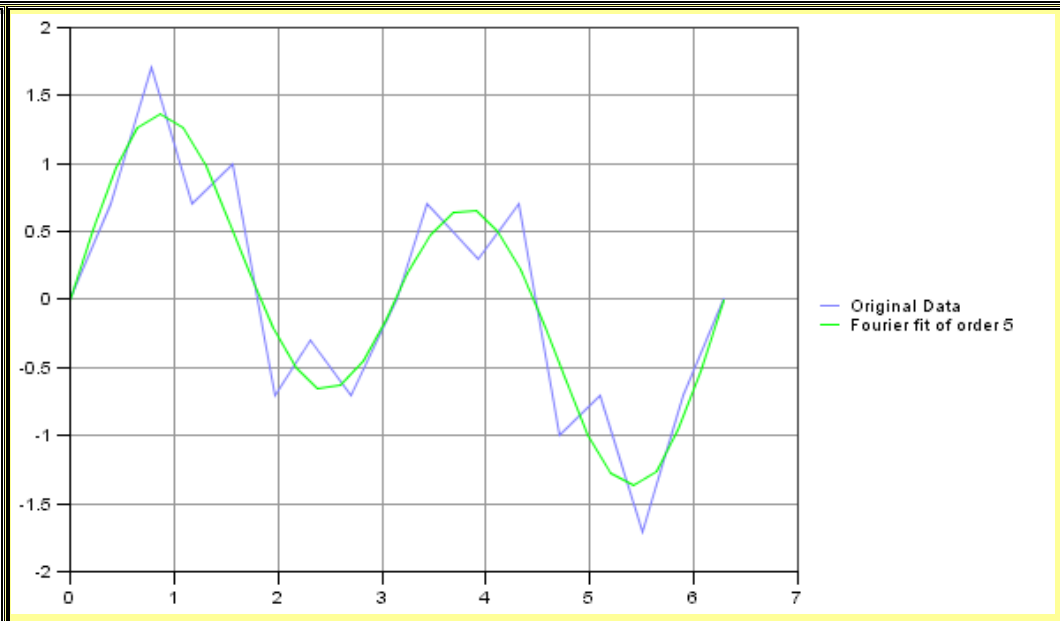
Parameters

Parameter Name	Description
Column X	X column for the Fourier regression operation. Column numbers begin with zero
Column Y	Y column for the Fourier regression operation. Column numbers begin with zero
Order	For a selected order of n, the Fourier equation fit to the data will extend up to the terms including cos(nx) and sin(nx)
Smoothen	Total number of (x,y) pairs to be returned in the result columns. This number should be larger than the original data set.
Result Column X Name	Name of result column X
Result Column Y Name	Name of result column Y
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input		Parameter	Result
angle	values	Column X = 0	See plot below.
0	0		
0.392699	0.707107	Column Y = 1	
0.785398	1.707107		
1.178097	0.707107	Order = 5	
1.570796	1		
1.963495	-0.70711	Smoothen = 30	
2.316594	-0.29289		
2.709293	-0.70711	Result Column X Name = For X	
3.141593	0		
3.433292	0.707107		
3.926991	0.29289	Result Column Y Name = For Y	
4.31969	0.707107		
4.712389	-1		
5.105088	-0.70711		
5.497787	-1.70711		
5.890486	-0.70711		
6.283185	0		

In the plot shown below, the original data is displayed along with the results of Fourier regression.



Module Name	Filter	Module Type	Functional
Description	The filter module fits a set of periodic data to a Fourier series of selected order and filters out unwanted orders (e.g. remove noise from a signal). Two columns of related data (x,y pairs) are fit to an equation of the form ($f(x) = b_0 + b_1\cos(x) + b_2\sin(x) + b_3\cos(2*x) + b_4\sin(2*x) + ...$). The result x column returns a new set of x values (the original set or a smoothened one). The result y column returns the function f(x) evaluated at the new x points with only terms of the desired orders included.		

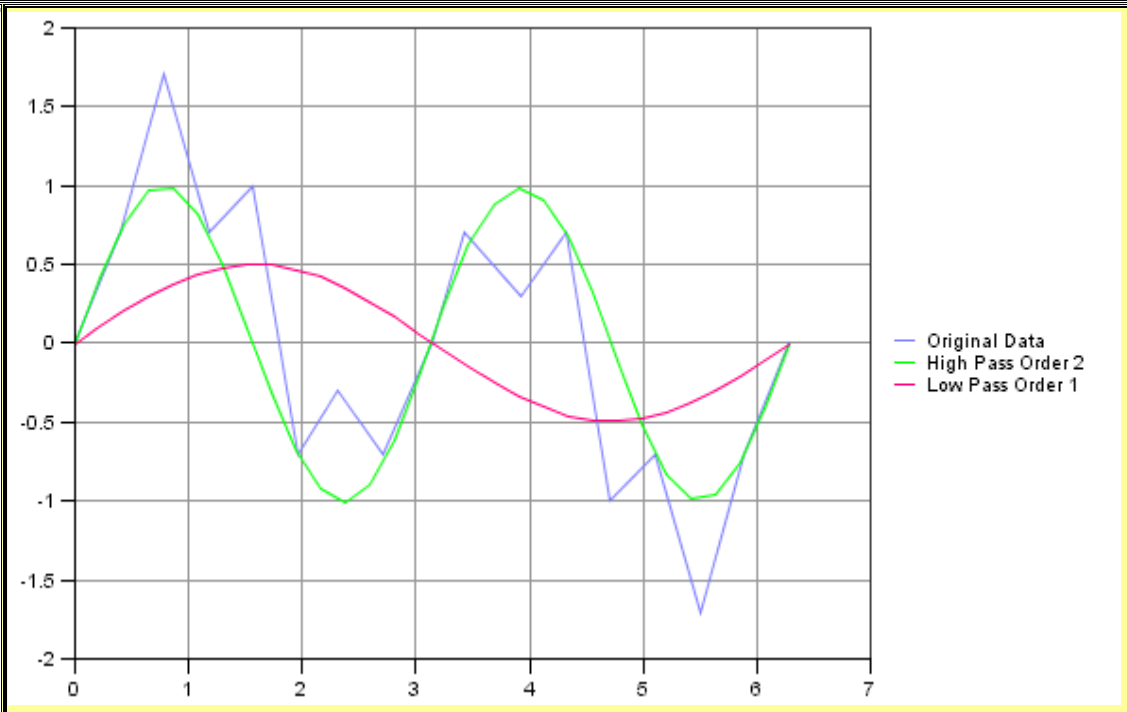
Parameters

Parameter Name	Description
Column X	X column for the filter operation. Column numbers begin with zero
Column Y	Y column for the filter regression operation. Column numbers begin with zero
Filter	Filter type options include Low Pass (remove orders above second cut-off order), High Pass (remove orders below first cut-off order), Band Pass (only keep orders between first cut-off order and second cut-off order), Band Stop (remove orders between first cut-off order and second cut-off order), Dominant (keep number of dominant orders specified and remove the rest)
First Cut-off Order	Cut-off applicable to High Pass, Band Pass and Band Stop Filter options
Second Cut-off Order	Cut-off applicable to Low Pass , Band Pass and Band Stop Filter options
Dominant Orders	Number of dominant orders to be retained beginning from the most dominant to the least. The dominant orders are the ones for which the Fourier coefficients have the greatest magnitude
Order	For a selected order of n, the Fourier equation fit to the data will extend up to the terms including cos(nx) and sin(nx)
Smoothen	Total number of (x,y) pairs to be returned in the result columns. This number should be larger than the original data set.
Result Column X Name	Name of result column X
Result Column Y Name	Name of result column Y
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input		Parameter	Result
angle	values	Column X = 0 Column Y = 1 Filter = Low Pass Second Cut-off Order = 1 Dominant Orders = Order = 5 Smoothen = 50 Result Column X Name = LPX Result Column Y Name = LPY	See plot below
0	0		
0.392699	0.707107		
0.785398	1.707107		
1.178097	0.707107		
1.570796	1		
1.963495	-0.70711		
2.316594	-0.29289		
2.709293	-0.70711		
3.141593	0		
3.433292	0.707107		
3.926991	0.29289		
4.31969	0.707107		
4.712389	-1		
5.105088	-0.70711		
5.497787	-1.70711		
5.890486	-0.70711		
6.283185	0		

In the plot shown below, the original data is displayed along with the results of low-pass (order 1) and high pass (order 2) filtering.



Module Name	Track Order	**Module Type**	Functional
Description	The Track Order module fits a set of periodic data to a Fourier series of selected order and calculates the magnitude of each order. Two columns of related data (x,y pairs) are fit to an equation of the form ($f(x) = b_0 + b_1\cos(x) + b_2\sin(x) + b_3\cos(2*x) + b_4\sin(2*x) + ...$). The coefficients of the fit are used to calculate the magnitude of each order.		
Parameters			

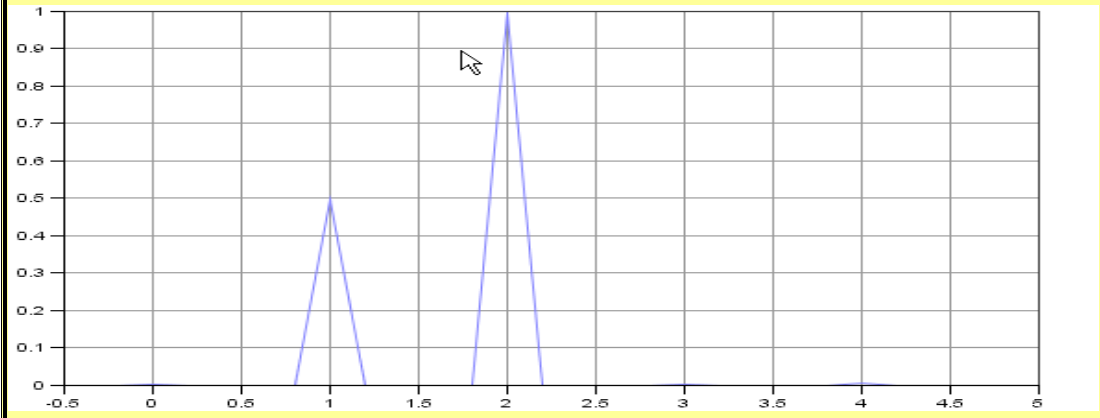
Parameter Name	Description
Column X	X column for the Track Order operation. Column numbers begin with zero
Column Y	Y column for the Track Order operation. Column numbers begin with zero
Order	For a selected order of n, the Fourier equation fit to the data will extend up to the terms including cos(nx) and sin(nx)
Result Column X Name	Name of result column X
Result Column Y Name	Name of result column Y
Include Original	If “Yes” include the original

Data	dataset in the result table

Unit Tests

Input		Parameter	Result
angle	values	Column X = 0	See plot below.
0	0		
0.392699	0.707107	Column Y = 1	
0.785398	1.707107		
1.178097	0.707107	Order = 5	
1.570796	1		
1.963495	-0.70711	Smoothen = 30	
2.316594	-0.29289		
2.709293	-0.70711		
3.141593	0		
3.433292	0.707107		
3.926991	0.29289		
4.31969	0.707107		
4.712389	-1		
5.105088	-0.70711		
5.497787	-1.70711		
5.890486	-0.70711		
6.283185	0		

The following plot shows Track Order analysis performed on the above input data. The dominant orders in this case are 2 and 1.



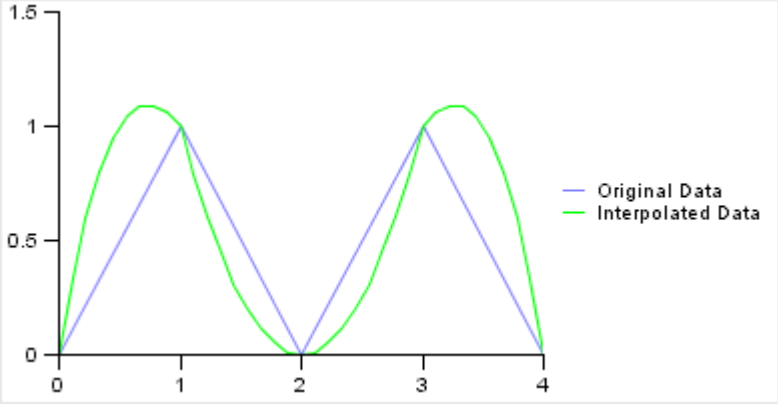
<table>
<tr><td>Module Name</td><td>Interpolation</td><td>Module Type</td><td>Functional</td></tr>
<tr><td>Description</td><td colspan="3">Interpolation on a set of scattered (x,y) data is performed in order to estimate y values at intermediate x points. A piecewise fit on subsets of data points is performed with a spline or piecewise polynomial of selected degree. The result (x,y) pairs include the original data plus (x,y) pairs evaluated at the number of intermediate points specified by the smoothen parameter.</td></tr>
<tr><td>Parameters</td><td colspan="3">
<table>
<tr><th>Parameter Name</th><th>Description</th></tr>
<tr><td>Column X</td><td>X column for the interpolation operation. Column numbers begin with zero</td></tr>
<tr><td>Column Y</td><td>Y column for the interpolation operation. Column numbers begin with zero</td></tr>
<tr><td>Operation</td><td>Spline type LINEAR, QUADRATIC or CUBIC indicating spline of degree 1, 2 or 3 respectively</td></tr>
<tr><td>Segment</td><td>Number of points per segment including both end points. Each consecutive pair of (x,y) points in the input data is treated as a segment.</td></tr>
<tr><td>Result Column X Name</td><td>Name of result column X</td></tr>
<tr><td>Result Column Y Name</td><td>Name of result column Y</td></tr>
<tr><td>Include Original Data</td><td>If “Yes” include the original dataset in the result table</td></tr>
</table>
</td></tr>
</table>

Unit Tests

In the plot

Input	Parameter	Result
X,Y 0,0 1,1 2,0 3,1 4,0	Column X = 0 Column Y = 1 Operation = CUBIC Segment = 10	See plot below

shown below, the original data is displayed along with the interpolated data.



Module Name	Moving Average	Module Type	Functional
Description	Calculate the moving average for a set of numbers. Moving averages are used in the financial sector to spot stock price trends in volatile markets. They are used as building blocks for other stock market indicators.		

Parameters

Parameter Name	Description
Column	Index of the column that needs to be used for moving average calculation Column numbers begin with zero
Moving Average Period	Period to be used in moving average computation. Generally a 10 day period is used.
Moving Average Type	Two different types of moving averages can be calculated. They are simple and exponential.
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
sun1stockprice	Column = 0	EMA
64.75		null
63.79	Moving Average	null
63.73	Period =10	null
63.73		null
63.55	Moving Average	null
63.19	Type =Exponential	null
63.91		null
63.85		null
62.95	Include	null
63.37	Original Data =	63.682
61.33	No	63.254
61.51		62.937
61.87		62.743
60.25		62.29
59.35		61.755
59.95		61.427
58.93		60.973
57.68		60.374
58.82		60.092
58.87		59.87

<table>
<tr><td>Module Name</td><td>Bollinger Bands</td><td>Module Type</td><td>Functional</td></tr>
<tr><td>Description</td><td colspan="3">Calculate the Bollinger Bands for a set of numbers. Bollinger Bands are an indicator that allows users to compare volatility and relative price levels over a period time. The indicator consists of three number designed to encompass the majority of a security's price action..
The number in the middle is a Simple Moving Average (SMA), the first number is SMA + 2 standard deviations and the last number is SMA - .2 standard deviations.</td></tr>
<tr><td>Parameters</td><td colspan="3">
<table>
<tr><th>Parameter Name</th><th>Description</th></tr>
<tr><td>Column</td><td>Index of the column that needs to be used for Bollinger bands calculation Column numbers begin with zero</td></tr>
<tr><td>Moving Average Period</td><td>Period to be used in moving average computation. Generally a 10 day period is used.</td></tr>
<tr><td>Standard Deviation Count</td><td>The number of standard deviations to be used to compute the upper and lower band values. Generally this is set to 2.</td></tr>
<tr><td>Include Original Data</td><td>If “Yes” include the original dataset in the result table</td></tr>
<tr><td></td><td></td></tr>
</table>
</td></tr>
</table>

Unit Tests

Input	Parameter	Result		
WALMART	Column = 0	Upper Band	Middle Band	Lower Band
103.13		null	null	null
109.00	Moving Average Period =20	null	null	null
103.06		null	null	null
102.75		null	null	null
108	Standard Deviation Count =2	null	null	null
107.56		null	null	null
105.25		null	null	null
107.69		null	null	null
108.63	Include Original Data = No	null	null	null
107.00		null	null	null
109		null	null	null
110.00		null	null	null
112.75		null	null	null
113.50		null	null	null
114.25		null	null	null
115.25		null	null	null
121.50		null	null	null
126.88		null	null	null
122.50		null	null	null
119		null	null	null
122.50		null	null	null
118		null	null	null
122		null	null	null
121.19		null	null	null
123.63		null	null	null
122.75		null	null	null
123.13		null	null	null
122.13		null	null	null
119		125.879	112.3	98.728
112.69		126.457	112.8	99.05
110.63		127.21	113.7	100.19
		127.52	114.6	101.73
		128.494	115.4	102.31
		129.107	116.2	103.22
		129.314	117.1	104.8
		129.43	117.8	106.13
		129.171	118.3	107.43
		128.515	118.6	108.65
		128.303	118.7	109.03

<table>
<tr><td>Module Name</td><td>Linear Regression</td><td>Module Type</td><td>Functional</td></tr>
<tr><td>Description</td><td colspan="3">Linear regression is a statistical tool used to predict future values from past values.

A Linear Regression trendline is computed using the least squares method. It plots a straight line through prices so as to minimize the distances between the prices and the resulting trendline.

In Technical analysis, a popular method of using Linear Regression trendline is to construct Linear Regression Channel lines. Developed by Gilbert Raff, the channel is constructed by plotting two parallel, equidistant lines above and below a Linear Regression trendline. The distance between the either channel line to the regression line is the greatest distance of any a value from the regression line.

In the process module output, the values in the column named “Upper” correspond to one regression channel and those in the column named “Lower” correspond to the other regression channel. The regression trend line values are in the column named “ Regression Value”</td></tr>
<tr><td>Parameters</td><td colspan="3"><table><tr><th>Parameter Name</th><th>Description</th></tr><tr><td>Column</td><td>Index of the column that needs to be used for Linear regression calculation Column numbers begin with zero</td></tr><tr><td>Include Original Data</td><td>If “Yes” include the original dataset in the result table</td></tr><tr><td></td><td></td></tr></table></td></tr>
</table>

Unit Tests

Input	Parameter	Result			
closing-price	Column = 0	closing-price	Lower	Regression Value	Upper
64.75		64.75	63.629	65.093	66.558
63.79	Include Original Data = Yes	63.79	63.279	64.743	66.208
63.73		63.73	62.929	64.393	65.858
63.73		63.73	62.579	64.044	65.508
63.55		63.55	62.229	63.694	65.158
63.19		63.19	61.879	63.344	64.808
63.91		63.91	61.529	62.994	64.458
63.85		63.85	61.179	62.644	64.108
62.95		62.95	60.829	62.294	63.758
63.37		63.37	60.479	61.944	63.409
61.33		61.33	60.129	61.594	63.059
61.51		61.51	59.78	61.244	62.709
61.87		61.87	59.43	60.894	62.359
60.25		60.25	59.08	60.544	62.009
59.35		59.35	58.73	60.194	61.659
59.95		59.95	58.38	59.844	61.309
58.93		58.93	58.03	59.494	60.959
57.68		57.68	57.68	59.145	60.609
58.82		58.82	57.33	58.795	60.259
58.87		58.87	56.98	58.445	59.909

Module Name	Stochastic Oscillator	**Module Type**	Functional

Description

Stochastic Oscillator is a momentum indicator that shows the location of the current close relative to the high/low range over a set number of periods.

Two types of Stochastic Oscillators are supported. They are Fast and Slow.

The main inputs to the Stochastic Oscillator calculation are the “K Period” and “D Period”.

The “Fast” stochastic oscillator output is composed of two values namely %K values and %D values.

%K values are computed using the following formula

$$\%K = \left(\frac{RecentClose - LowestLowinlastKperiod}{HighestHigh(inKperiod) - LowestLow(inKperiod)} \right) X100$$

%D values are the D period Simple Moving Average of the %K values

In case of Slow stochastic oscillator,
%K values are the 3 day Simple Moving Average of the %K values computed using the fast formula shown above.
%D values are the D period Simple Moving Average of the %K values

Hence,

- %K (fast) = %K formula presented above using K Period
- %D (fast) = D period SMA of %K (fast)
- %K (slow) = 3-day SMA of %K (fast)
- %D (slow) = D period -day SMA of %K (slow)

(refefence:
http://www.stockcharts.com/education/IndicatorAnalysis/indic_stochasticOscillator.html)

Parameters

Parameter Name	Description
High Column	Column containing the stock high values
Low Column	Column containing the stock low values
Close Column	Column containing the stock close values
K Period	The period to be used for %K computation.
D Period	The period to be used for %D computation.
Type	Type of Stochastic Oscillator (Fast or Slow)
Include Original Data	If “Yes” include the original dataset in the result table

Unit Tests

Input	Parameter	Result
The input is echoed in the output	High Column =1 Low Column = 2 Close Column = 3 K Period= 14 D Period=3 Type = Slow Include Original Data = Yes	

Date	High	Low	Close	%K Slow	%D Slow
26-Sep-03	20.44	19.95	19.96	null	null
29-Sep-03	20.28	20.02	20.2	null	null
30-Sep-03	20.13	19.3	19.59	null	null
1-Oct-03	20.26	19.68	20.2	null	null
2-Oct-03	20.3	19.75	19.99	null	null
3-Oct-03	21.02	20.27	20.76	null	null
6-Oct-03	20.99	20.19	20.8	null	null
7-Oct-03	21.01	20.45	21	null	null
8-Oct-03	21.13	20.69	20.79	null	null
9-Oct-03	21.39	20.71	20.95	null	null
10-Oct-03	21.21	20.71	20.79	null	null
13-Oct-03	21.09	20.86	21	null	null
14-Oct-03	21.18	20.75	21.14	null	null
15-Oct-03	21.42	20.92	21.02	27.044	9.01468
16-Oct-03	21.1	20.78	21.04	54.4025	27.1488
17-Oct-03	21.03	20.63	20.64	75.4717	52.3061
20-Oct-03	21.13	20.75	21.08	75.2476	68.3739
21-Oct-03	21.24	20.9	21.11	75.0348	75.2514
22-Oct-03	21.01	20.6	20.63	65.8898	72.0574
23-Oct-03	20.37	20.02	20.16	42.4032	61.1092
24-Oct-03	20.1	19.24	19.8	23.8201	44.0377
27-Oct-03	20.21	19.81	19.93	22.4465	29.5566
28-Oct-03	20.93	20.07	20.9	44.4954	30.254
29-Oct-03	20.94	20.64	20.82	60.0917	42.3445
30-Oct-03	21.13	20.71	20.86	74.3119	59.633

<table>
<tr><td>Module Name</td><td>Chaikin Oscillator</td><td>Module Type</td><td>Functional</td></tr>
<tr><td>Description</td><td colspan="3">Chaikin Oscillator uses the concept of Accumulation/Distribution line to indicate trends.

Chaikin Oscillator is a Moving Average Convergence Divergence indicator (MACD) applied to the Accumulation/Distribution Line. It is the difference between the specified short period exponential moving average and the long period exponential moving average of the Accumulation/Distribution Line.

The Chaikin oscillator output is composed of four values namely
• Close Location Value (CLV)
• Product of CLV and Volume
• Running total of product of CLV and Volume
• Oscillator values
(refefence: http://www.stockcharts.com/education/IndicatorAnalysis/indic_ChaikinOscillator.html)</td></tr>
<tr><td>Parameters</td><td colspan="3"><table><tr><th>Parameter Name</th><th>Description</th></tr><tr><td>High Column</td><td>Column containing the stock high values</td></tr><tr><td>Low Column</td><td>Column containing the stock low values</td></tr><tr><td>Close Column</td><td>Column containing the stock close values</td></tr><tr><td>Volume Column</td><td>Column containing the stock volume values</td></tr><tr><td>Short Period</td><td>Short period to be used for computation.</td></tr><tr><td>Long Period</td><td>Long period to be used for computation.</td></tr><tr><td>Include Original Data</td><td>If “Yes” include the original dataset in the result table</td></tr></table></td></tr>
</table>

Unit Tests

Input	Parameter
The input is echoed in the result	High Column =0 Low Column = 1 Close Column = 2 Volume Column=3 Short Period= 3 Long Period=10 Type = Slow Include Original Data = Yes

Result

High	Low	Close	Volume	CLV	CLV*V	SUM CLV*V	OSC
59.44	58.44	58.44	1987200	-1	-1987200	-1987200	null
59.88	58.63	58.94	1832200	-0.504	-923428.8	-2910628.8	Null
59.63	58.69	59.13	2369500	-0.0638298	-151244.68	-3061873.481	Null
59.94	58.56	59.5	3091900	0.3623188	1120253.6	-1941619.858	Null
60.25	58.81	59.25	3375200	-0.3888889	-1312577.8	-3254197.635	null
60.75	58.5	58.5	4530500	-1	-4530500	-7784697.635	null
60	58.13	58.44	3175000	-0.6684492	-2122326.2	-9907023.839	Null
60.38	58.5	59.56	3792800	0.1276596	484187.23	-9422836.605	Null
59.88	57.38	59.75	8017800	0.896	7183948.8	-2238887.805	null
60.38	57.88	58.5	5304100	-0.504	-2673266.4	-4912154.205	-400746.57
61.19	59.38	59.88	4531100	-0.4475138	-2027729.8	-6939884.039	-899664.39
61.94	61.06	61.25	3560900	-0.5681818	-2023238.6	-8963122.675	-1665737.2
63	61.38	62	2794800	-0.2345679	-655570.37	-9618693.046	-2036290.6
63.38	62.13	63.25	5864800	0.792	4644921.6	-4973771.446	-524833.64
62.19	60.06	62.13	6864200	0.943662	6477484.5	1503713.061	2202219.6
60.88	59.38	60	10012200	-0.1733333	-1735448	-231734.9386	2565442.5
62.75	58.75	59	18074200	-0.875	-1.58E+07	-1.60E+07	-2551210
68.19	62.63	64.61	13265900	-0.2877698	-3817525.2	-1.99E+07	-5627124.9
69	67.81	68.31	3836400	-0.1596639	-612534.45	-2.05E+07	-6568794.2
67.94	67	67.44	2380600	-0.0638298	-151953.19	-2.06E+07	-6405208.2
69	66.69	68.19	5431300	0.2987013	1622336.4	-1.90E+07	-5239797.1
67.94	66.5	67.31	3778600	0.125	472325	-1.85E+07	-4136407.6
67.19	65.75	67.06	4260200	0.8194444	3490997.2	-1.50E+07	-2198212.1
67.87	66.56	66.56	1655200	-1	-1655200	-1.67E+07	-1732131
68.44	66.19	67.94	5828800	0.5555556	3238222.2	-1.35E+07	-353651.56
66.94	65.81	66.69	9634500	0.5575221	5371446.9	-8088545.053	1951518.7

Module Name	Ultimate Oscillator	Module Type	Functional
Description	"Ultimate" Oscillator combines a stock's price action during three different time frames into one bounded oscillator. Values range from 0 to 100 with 50 as the center line. Oversold territory exists below 30 and overbought territory extends from 70 to 100. The time frames used by the Ultimate Oscillator are user defined. The typical value for the small period is 7, medium period is 14 and long period is 28.		
Parameters			

Parameter Name	Description
High Column	Column containing the stock high values
Low Column	Column containing the stock low values
Close Column	Column containing the stock close values
Small Period	Small period to be used for computation.
Medium Period	Medium period to be used for computation
Long Period	Long period to be used for computation.
Include Original Data	If “Yes” include the original dataset in the result table

Unit Test

Example for Ultimate Oscillator

Input Parameters	Parameter	Values
	High Column	1
	Low Column	2
	Close Column	3
	Small Period	7
	Medium Period	14
	Long Period	28
	Include Original Data	Yes
	Input data	Input data is echoed in the output

Results

Date	High	Low	Close	ULT
26-Sep-03	20.44	19.95	19.96	null
29-Sep-03	20.28	20.02	20.2	null
30-Sep-03	20.13	19.3	19.59	null
1-Oct-03	20.26	19.68	20.2	null
2-Oct-03	20.3	19.75	19.99	null
3-Oct-03	21.02	20.27	20.76	null
6-Oct-03	20.99	20.19	20.8	null
7-Oct-03	21.01	20.45	21	null
8-Oct-03	21.13	20.69	20.79	null
9-Oct-03	21.39	20.71	20.95	null

10-Oct-03	21.21	20.71	20.79	null
13-Oct-03	21.09	20.86	21	null
14-Oct-03	21.18	20.75	21.14	null
15-Oct-03	21.42	20.92	21.02	null
16-Oct-03	21.1	20.78	21.04	null
17-Oct-03	21.03	20.63	20.64	null
20-Oct-03	21.13	20.75	21.08	null
21-Oct-03	21.24	20.9	21.11	null
22-Oct-03	21.01	20.6	20.63	null
23-Oct-03	20.37	20.02	20.16	null
24-Oct-03	20.1	19.24	19.8	null
27-Oct-03	20.21	19.81	19.93	null
28-Oct-03	20.93	20.07	20.9	null
29-Oct-03	20.94	20.64	20.82	null
30-Oct-03	21.13	20.71	20.86	null
31-Oct-03	21.12	20.83	20.93	null
3-Nov-03	21.77	21.18	21.71	null
4-Nov-03	21.85	21.4	21.58	
5-Nov-03	23.06	21.35	21.8	58.074
6-Nov-03	23.14	22.61	22.9	66.639
7-Nov-03	23.02	22.23	22.34	59.73
10-Nov-03	22.67	22.1	22.19	56.715
11-Nov-03	22.45	22.05	22.35	59.476
12-Nov-03	23	22.37	22.97	58.844
13-Nov-03	22.75	22	22.7	61.511
14-Nov-03	22.99	22.22	22.26	61.827
17-Nov-03	22.16	21.8	22.05	50.406
18-Nov-03	22.4	21.69	21.73	48.644
19-Nov-03	22.17	21.6	22.05	54.169
20-Nov-03	22.37	21.77	21.94	50.79
21-Nov-03	22.28	21.55	22.17	49.593
24-Nov-03	22.98	22.36	22.8	52.234
25-Nov-03	22.95	22.6	22.62	55.724
26-Nov-03	22.87	22.43	22.81	55.811
28-Nov-03	22.86	22.61	22.7	61.534
1-Dec-03	23.25	22.83	23.18	65.595
2-Dec-03	23.24	23.06	23.11	68.676
3-Dec-03	23.8	23.13	23.3	59.516
4-Dec-03	24.11	23.68	23.98	64.469
5-Dec-03	23.85	23.62	23.75	67.993

Module Name	Price Oscillator	Module Type	Functional
Description	The Price Oscillator is an indicator based on the difference between two moving averages over a short and a long period. It is expressed as either a percentage or in absolute terms. The short and long periods values can vary depending on user preference. **Absolute Price Oscillator (APO)** is calculated by subtracting the longer moving average from the shorter moving average. The **Percentage Price Oscillator (PPO)** is found by subtracting the longer moving average from the shorter moving average and then dividing the result by the longer moving average.		
Parameters			

Parameter Name	Description
Column	Column containing the stock high values
Short Period	Column containing the stock low values
Long Period	Column containing the stock close values
Oscillator Type	Type of oscillator. The two different types supported Absolute and Percentage
Include Original Data	If “Yes” include the original dataset in the result table

Unit Test Example for Price Oscillator

Input Parameters	Parameter	Values
	Column	4
	Short Period	2
	Long Column	3
	Oscillator Type	Absolute
	Long Period	10
	Include Original Data	Yes
	Input data	Input data is echoed in the output

Results

Date	Open	High	Low	Close	Volume	Adj. Close*	APO
29-Dec-00	81.5	82.44	79.94	79.94	1993300	75.7	null
28-Dec-00	79.81	81.87	79.31	81.44	2234600	77.12	null
27-Dec-00	80.94	81.94	78.69	80.25	2551600	75.99	null
26-Dec-00	76.81	81.75	76.81	80.94	2807100	76.65	null
22-Dec-00	75.5	76.69	75.12	76.5	3202200	72.44	null
21-Dec-00	75.37	75.69	74.19	75	4342100	71.02	null
20-Dec-00	79.31	79.62	75.19	75.37	4568600	71.19	null

19-Dec-00	76.5	80.94	76.5	79.75	5048300	75.33	null
18-Dec-00	72.94	77.19	72.87	77	4082600	72.73	null
15-Dec-00	70.31	72.87	70.06	71.94	3149700	67.95	null
14-Dec-00	72.31	73.81	70.44	70.5	2568000	66.59	-2.451
13-Dec-00	73.56	74.81	71.75	72	4557100	68.01	-2.313
12-Dec-00	73.75	74.37	72.37	73.37	3443000	69.3	-1.89
11-Dec-00	69.87	72.56	69.69	72.31	2529000	68.3	-1.706
8-Dec-00	69.81	70.25	68.25	69.37	5158800	65.53	-1.947
7-Dec-00	69.12	70.37	68.69	69.81	3680000	65.94	-1.894
6-Dec-00	69.44	70.31	67.25	68.44	5557800	64.65	-1.958
5-Dec-00	69.31	69.69	67.62	68.25	3927900	64.47	-1.903
4-Dec-00	66.81	68.62	66.62	68.31	4152200	64.52	-1.748
1-Dec-00	62.94	67.81	61.13	66.75	7788700	63.05	-1.795
30-Nov-00	66.31	66.81	61.19	62	7637200	58.56	-2.431
29-Nov-00	69.37	69.56	64.75	66.06	6767200	62.4	-2.015
28-Nov-00	71.31	72.31	69.25	69.31	4128500	65.47	-1.174
27-Nov-00	73	73.31	70.81	71.37	3277300	67.41	-0.332
24-Nov-00	72.31	73.94	71.75	73.44	1004900	69.37	0.461
22-Nov-00	73.5	74.19	71.12	71.69	2684500	67.72	0.601
21-Nov-00	74.81	75.87	74.31	74.87	2572900	70.72	1.122
20-Nov-00	77	77.44	74.25	74.75	2828300	70.61	1.32
17-Nov-00	78	78.37	75.37	76.37	3201800	72.14	1.594
16-Nov-00	79.5	80.31	78.06	79.12	2100000	74.73	2.063
15-Nov-00	77.37	79.94	76.94	79.69	2483700	75.27	2.281
14-Nov-00	74.25	76.87	73.75	75.94	2823800	71.73	1.693
13-Nov-00	78.12	78.75	74.37	74.37	3485100	70.25	1.032
10-Nov-00	79.31	79.81	77.12	77.31	1916300	73.03	1.054
9-Nov-00	80.37	80.87	77.25	78.87	2702600	74.5	1.239
8-Nov-00	79.31	80.56	78.94	80	2332300	75.57	1.435
7-Nov-00	78.25	79.44	78.06	79.12	2669200	74.73	1.322
6-Nov-00	79.44	79.44	76.12	77.56	3175100	73.26	0.944
3-Nov-00	78.5	80.75	78.31	78.94	1856800	74.56	0.89
2-Nov-00	79.87	79.94	78.19	78.69	3110600	74.33	0.768
1-Nov-00	76.94	80.31	76.94	80.12	3492700	75.68	0.872
31-Oct-00	76.37	76.75	74.56	76.12	2960900	71.9	0.27
30-Oct-00	75.06	76.19	73	75.87	3501300	71.66	-0.113
27-Oct-00	76.56	76.56	72.56	75.12	2938500	70.96	-0.428
26-Oct-00	76.62	77.37	74	76.19	3234300	71.97	-0.412
25-Oct-00	77	79	75.25	75.31	3858600	71.14	-0.512
24-Oct-00	80.94	81.44	78.69	78.81	2557700	74.44	-0.005
23-Oct-00	83.94	83.94	80	80.5	3738900	76.04	0.528
20-Oct-00	81.37	84.44	81.25	84.12	1947800	79.46	1.335
19-Oct-00	81.44	83.87	80.69	81.56	2018300	77.04	1.307

Statistical Process Control (SPC)

Shewharts SPC is a methodology for charting the process and quickly determining when a process is “Out Of Control”. The process is then investigated to determine the root cause of the “Out Of Control” condition. When the root cause of the problem is determined, a strategy is identified to correct it.

Charts are classified as a Variables Control Chart or as an Attribute Control Chart.

1. Variables Control Charts

   As a true preventive tool, control charts for variable data provide the measure of process improvement.

   a. Run Chart
   b. XBar – R Control Charts
   c. XBar – s Control Charts
   d. Median – R Control Charts
   e. Median – s Control Charts
   f. Individuals Chart or Moving Range (MR) Chart
   g. Exponentially Weighted Moving Average (EWMA) Charts
   h. Histogram
   i. Probability Chart

2. Attribute Control Charts

   Attribute Control Charts are used when items are compared with some standard and then classified as to whether they meet the standard or not. As an example Attribute Control Charts are used to determine if the rate of nonconforming product is stable and detect when a deviation from stability has occurred.

   a. p Charts
   b. np Charts
   c. c Charts
   d. u Charts

Definition of a Sample: One or more measurements are taken as part of a sample. Samples are taken at regular intervals. Let us consider a machine shop where everyday 1000 pieces of a metal product is manufactured. As part of quality management 20 pieces are checked every day for inconsistence. The set of 20 pieces are tagged with an id that represents the day on which it was produced. In this context the 20 pieces are the sample. The tag for this sample is called sample identity. Number of items in the sample is called the sample size.

Number of nonconformities exceeding items inspected: Normally during inspection a subset of items inspected will be nonconforming to desired result. In this situation the number of nonconforming items is always less than the number of items inspected. In certain other situations number of

nonconformities for a given set of items could exceed the number of items in the set. For example when you are inspecting number of scratches on a product, your result could be 25 scratches for 10 items inspected.

Following are general guide lines for Control Chart selection:

1. If the data is of type measurable then use Variables Control Charts.
    a. To view raw data and make general observations use Run Chart.
    b. To derive statistical analysis:
        i. If the sample size is 1 then use MR Chart
        ii. If the sample size is greater than 1 and number of samples is less than 10 then use Range Charts – Xbar R or Median R.
        iii. If the sample size is greater than 1 and number of samples is greater than 10 then use s (Standard Deviation) Charts – XBar s or Median s.
        iv. EWMA chart is an alternative to XBar charts when sample size is greater than 1. EWMA chart is an alternative to MR charts when sample size is equal to 1. This chart has a build in mechanism for incorporating information from all previous samples, weighting the information from the closest sample with a higher weight.
        v. Histogram can be used to check if a process is normally distributed and centered.
        vi. Probability chart is used to determine the normality of a distribution.
2. If the data is of type countable then use Attribute Control Charts.
    a. If the number of nonconformities is less than the items checked use p Chart, when the sample sizes are not the same, or np Chart, when the sample sizes are the same for each sample.
    b. If the number of nonconformities is greater than the items checked use u Chart, when the sample sizes are not the same, or c Chart, when the sample sizes are the same for each sample.

SPC Process Module Implementation

Module Name: **SPC Variables Chart** Module Type: **Functional**

Description: SPC Variables Chart module can be used to perform statistical analysis, on data set of type measurable, and determine if a process is out of control.

Parameters:

Name	Description
Column Index Sample Identity	Index of the column whose data represents the sample identity.
Column Index Sub-group Item Value	Index of the column whose data represents individual measurements.
Chart Type	Several chart types from which one can be selected depending on: 1. Sample size. 2. Number of samples. 3. Whether sample size of each sample is the same.
LSL	Lower Specification Limit – It is the value below which a measurement is considered out of specification. This value is normally the target value minus the tolerance allowed.
USL	Upper Specification Limit – It is the value above which a measurement is considered out of specification. This value is normally the target value plus the tolerance allowed.
Target	It is the value which every measurement is expected to be. For the product to be within specification the amount of deviation from this value should be less than the tolerance.
Result Type	Chart Data or Metrics During an SPC analysis two types of data are generated. Calculations that are sample specific (Chart Data) and those that are dataset specific (Metrics). For example in a XBar-R chart XBar and Range is calculated for each sample while XDBBar, RBar, UCL and LCL are values calculated at the dataset level. Chart Data is always required for construction and display of a control chart. Metrics is used for display in a table as textual information or can be plotted in the chart.
Include Original Data	Yes/No. Determines if original data has to be included in the output. Default is Yes.

Unit Test

Dataset 1: Data set with sample size greater than 1 and constant across all samples.

SAMPLE_ID	MEASUREMENT						
1	2	6	2.0004	11	2.0002	16	1.9999
1	1.9998	6	2.0003	11	1.9999	16	2.0003
1	2.0002	6	2	11	2.0001	16	1.9993
2	1.9998	7	1.9998	12	2.0002	17	2.0002
2	2.0003	7	1.9998	12	1.9998	17	1.9998
2	2.0002	7	1.9998	12	2.0005	17	2.0004
3	1.9998	8	2	13	2	18	2
3	2.0001	8	2.0001	13	2.0001	18	2.0001
3	2.0005	8	2.0001	13	1.9998	18	2.0001
4	1.9997	9	2.0005	14	2	19	1.9997
4	2	9	2	14	2.0002	19	1.9994
4	2.0004	9	1.9999	14	2.0004	19	1.9998
5	2.0003	10	1.9995	15	1.9994	20	2.0003
5	2.0003	10	1.9998	15	2.0001	20	2.0007
5	2.0002	10	2.0001	15	1.9996	20	1.9999

Dataset 2: Data set with sample size equal to 1.

SAMPLE_ID	MEASUREMENT
1	2
2	1.9998
3	2.0002
4	1.9998
5	2.0003
6	2.0002
7	1.9998
8	2.0001
9	2.0005
10	1.9997
11	2
12	2.0004

Note:

1. For all the charts the parameter “Include Original Data” has been set to “No”.
2. Result has been shown separately for the parameter “Result Type” options.

Chart Type: Run

Input: Data Set 2

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sub-group Item Value	1
Chart Type	Run
LSL	1.9990
USL	2.0010
Target	2.0000

Result Type: Chart Data		Result Type: Metrics	
SAMPLE_ID	MEASUREMENT		
1	2	Run Chart Limits	
2	1.9998		
3	2.0002	XBar	2.000066667
4	1.9998	LSL	1.999
5	2.0003	Target	2
6	2.0002	USL	2.001
7	1.9998		
8	2.0001		
9	2.0005		
10	1.9997		
11	2		
12	2.0004		

Chart Type: XBar-R

Input: Data Set 1

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sub-group Item Value	1
Chart Type	XBar-R
LSL	1.9990
USL	2.0010
Target	2.0000

Result Type: Chart Data	Result Type: Metrics

SAMPLE_ID	RANGE	XBAR
1	4.00E-04	2
2	5.00E-04	2.0001
3	7.00E-04	2.000133333
4	7.00E-04	2.000033333
5	1.00E-04	2.000266667
6	4.00E-04	2.000233333
7	0.00E+00	1.9998
8	1.00E-04	2.000066667
9	6.00E-04	2.000133333
10	6.00E-04	1.9998
11	3.00E-04	2.000066667
12	7.00E-04	2.000166667
13	3.00E-04	1.999966667
14	4.00E-04	2.0002
15	7.00E-04	1.9997
16	1.00E-03	1.999833333
17	6.00E-04	2.000133333
18	1.00E-04	2.000066667
19	4.00E-04	1.999633333
20	8.00E-04	2.0003

RBar Chart Limits	
RBar	4.70E-04
UCL	1.21E-03
LCL	0.00E+00
Process Std Dev	2.78E-04
XBar Chart Limits	
XDBLBar	2.00E+00
UCL	2.00E+00
LCL	2.00E+00
LSL	2.00E+00
Target	2.00E+00
USL	2.00E+00

Chart Type: XBar-s

Input: Data Set 1

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sub-group Item Value	1
Chart Type	XBar-s
LSL	1.9990
USL	2.0010
Target	2.0000

Result Type: Chart Data

SAMPLE_ID	ST_DEV	XBAR
1	1.63E-04	2
2	2.16E-04	2.0001
3	2.87E-04	2.000133333
4	2.87E-04	2.000033333
5	4.71E-05	2.000266667
6	1.70E-04	2.000233333
7	2.22E-16	1.9998

Result Type: Metrics

SBar Chart Limits	
SBar	0.00019612
UCL	0.00050364
LCL	0
Process Std Dev	0.00010253
XBar Chart Limits	

8	4.71E-05	2.000066667	XDBLBar	2.00003167
9	2.62E-04	2.000133333	UCL	2.00041489
10	2.45E-04	1.9998	LCL	1.99964844
11	1.25E-04	2.000066667	LSL	1.999
12	2.87E-04	2.000166667	Target	2
13	1.25E-04	1.999966667	USL	2.001
14	1.63E-04	2.0002		
15	2.94E-04	1.9997		
16	4.11E-04	1.999833333		
17	2.49E-04	2.000133333		
18	4.71E-05	2.000066667		
19	1.70E-04	1.999633333		
20	3.27E-04	2.0003		

Chart Type: Median-R

Input: Data Set 1

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sub-group Item Value	1
Chart Type	Median-R
LSL	1.9990
USL	2.0010
Target	2.0000

Result Type: Chart Data

SAMPLE_ID	RANGE	XBAR	MEDIAN
1	4.00E-04	2	2
2	5.00E-04	2.0001	2.0002
3	7.00E-04	2.000133333	2.0001
4	7.00E-04	2.000033333	2
5	1.00E-04	2.000266667	2.0003
6	4.00E-04	2.000233333	2.0003
7	0	1.9998	1.9998

Result Type: Metrics

RBar Chart Limits	
RBar	0.00047
UCL	0.00120978
LCL	0
Process Std Dev	0.000277614
XBar Chart Limits	

8	1.00E-04	2.000066667	2.0001	XDBLBar	2.000031667
9	6.00E-04	2.000133333	2	UCL	2.000589557
10	6.00E-04	1.9998	1.9998	LCL	1.999473777
11	3.00E-04	2.000066667	2.0001	LSL	1.999
12	7.00E-04	2.000166667	2.0002	Target	2
13	3.00E-04	1.999966667	2	USL	2.001
14	4.00E-04	2.0002	2.0002		
15	7.00E-04	1.9997	1.9996		
16	1.00E-03	1.999833333	1.9999		
17	6.00E-04	2.000133333	2.0002		
18	1.00E-04	2.000066667	2.0001		
19	4.00E-04	1.999633333	1.9997		
20	8.00E-04	2.0003	2.0003		

Chart Type: Median-s

Input: Data Set 1

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sub-group Item Value	1
Chart Type	Median-s
LSL	1.9990
USL	2.0010
Target	2.0000

Result Type: Chart Data

SAMPLE_ID	ST_DEV	XBAR	MEDIAN
1	0.0001633	2	2
2	0.00021602	2.0001	2.0002
3	0.00028674	2.000133333	2.0001
4	0.00028674	2.000033333	2
5	4.714E-05	2.000266667	2.0003
6	0.00016997	2.000233333	2.0003
7	2.2204E-16	1.9998	1.9998

Result Type: Metrics

SBar Chart Limits	
SBar	1.96E-04
UCL	5.04E-04
LCL	0
Process Std Dev	1.03E-04
XBar Chart Limits	

8	4.714E-05	2.000066667	2.0001	XDBLBar	2.000031667
9	0.00026247	2.000133333	2	UCL	2.00024093
10	0.00024495	1.9998	1.9998	LCL	1.999822403
11	0.00012472	2.000066667	2.0001	LSL	1.999
12	0.00028674	2.000166667	2.0002	Target	2
13	0.00012472	1.999966667	2	USL	2.001
14	0.0001633	2.0002	2.0002		
15	0.00029439	1.9997	1.9996		
16	0.00041096	1.999833333	1.9999		
17	0.00024944	2.000133333	2.0002		
18	4.714E-05	2.000066667	2.0001		
19	0.00016997	1.999633333	1.9997		
20	0.0003266	2.0003	2.0003		

Chart Type: MR

Input: Data Set 2

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sub-group Item Value	1
Chart Type	MR
LSL	1.9990
USL	2.0010
Target	2.0000

Result Type: Chart Data

SAMPLE_ID	MEASUREMENT	MOVING_RANGE
1	2	0.0002
2	1.9998	0.0004
3	2.0002	0.0004
4	1.9998	0.0005
5	2.0003	0.0001
6	2.0002	0.0004
7	1.9998	0.0003

Result Type: Metrics

MR Chart Limits	
MRBar	3.82E-04
UCL	2.001082303
LCL	1.99905103
MR Std Dev	3.38E-04
XBar Chart Limits	
XBar	2.000066667

8	2.0001	0.0004	LSL	1.999
9	2.0005	0.0008	Target	2
10	1.9997	0.0003	USL	2.001
11	2	0.0004		
12	2.0004	null		

Chart Type: EWMA

Input: Data Set 1

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sub-group Item Value	1
Chart Type	EWMA
LSL	1.9990
USL	2.0010
Target	2.0000

Result Type: Chart Data

SAMPLE ID	XDBBAR	LCL	UCL	XHAT
1	2.00E+00	2.00E+00	2.00E+00	2.00E+00
2	2.000031667	1.999818355	2.00024497	2.00004026
3	2.000031667	1.999793201	2.00027013	2.00005888
4	2.00E+00	2.00E+00	2.00E+00	2.00E+00
5	2.000031667	1.99976938	2.00029395	2.00009635
6	2.000031667	1.999763761	2.00029957	2.00012374
7	2.000031667	1.999760226	2.00030310	2.00005899

Result Type: Metrics

EWMA Chart Limits	
XRBar	4.70E-04
Process St Dev	2.78E-04
XDBLBar	2.000031667
LSL	2.00E+00
Target	2
USL	2.001

8	2.000031667	1.999757988	2.00030534	2.00006053
9	2.000031667	1.999756565	2.00030676	2.00007509
10	2.000031667	1.999755658	2.00030767	2.00002007
11	2.000031667	1.999755079	2.00030825	2.00002939
12	2.000031667	1.999754709	2.00030862	2.00005684
13	2.000031667	1.999754473	2.00030886	2.00003881
14	2.000031667	1.999754322	2.00030901	2.00007104
15	2.000031667	1.999754225	2.00030910	1.99999683
16	2.000031667	1.999754163	2.00030917	1.99996413
17	2.000031667	1.999754123	2.00030921	1.99999797
18	2.000031667	1.999754098	2.00030923	2.00001171
19	2.000031667	1.999754082	2.00030925	1.99993603
20	2.000031667	1.999754071	2.00030926	2.00000883

Chart Type: Histogram

Input: Data Set 1

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sub-group Item Value	1
Chart Type	Histogram
LSL	1.9990
USL	2.0010
Target	2.0000

Result Type: Chart Data

BIN_NO	BIN_X_START	BIN_X_END	BIN_X_MID	X_FREQ
1	1.999	1.9992	1.9991	0
2	1.9992	1.9994	1.9993	1
3	1.9994	1.9996	1.9995	3
4	1.9996	1.9998	1.9997	3
5	1.9998	2	1.9999	15
6	2	2.0002	2.0001	17
7	2.0002	2.0004	2.0003	13
8	2.0004	2.0006	2.0005	7

Result Type: Metrics

Histogram Limits	
XBar	2.000031667
UCL	8.67E-04
LCL	-8.67E-04
Process Std Dev	2.89E-04
LSL	1.999
Target	2
USL	2.001

9	2.0006	2.0008	2.0007	1	
10	2.0008	2.001	2.0009	0	

Chart Type: Probability

Input: Data Set 1

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sub-group Item Value	1
Chart Type	Probability
LSL	1.9990
USL	2.0010
Target	2.0000

Result Type: Chart Data

SAMPLE_ID	XBAR	PROBABILITY
1	2	0.433538527
2	2.0001	0.641014559
3	2.000133333	0.704488252
4	2.000033333	0.503514266
5	2.000266667	0.892896342
6	2.000233333	0.856765081
7	1.9998	0.110389162
8	2.000066667	0.573381768
9	2.000133333	0.704488252

Result Type: Metrics

SBar Chart Limits	
SBar	0.000196123
UCL	0.000503645
LCL	0
Process Std Dev	0.000102533
XBar Chart Limits	
XDBLBar	2.000031667

10	1.9998	0.110389162	UCL	2.000414892
11	2.000066667	0.573381768	LCL	1.999648442
12	2.000166667	0.762242587	LSL	1.999
13	1.999966667	0.365590922	Target	2
14	2.0002	0.813190895	USL	2.001
15	1.9997	0.039800532		
16	1.999833333	0.147254888		
17	2.000133333	0.704488252		
18	2.000066667	0.573381768		
19	1.999633333	0.017629626		
20	2.0003	0.921942671		

Module Name: **SPC Attribute Chart** Module Type: **Functional**

Description: SPC Attribute Chart module can be used to perform statistical analysis, on data set of type countable, and determine if a process is out of control.

Parameters:

Name	Description
Column Index Sample Identity	Index of the column whose data represents the sample identity.
Column Index Sample Size	Index of the column whose data represents the sample size.
Column Index Number of Nonconformance	Index of the column whose data represents number of nonconformance among the corresponding sample size.
Chart Type	Several chart types from which one can be selected depending on: 1. Sample size. 2. Whether sample size of each sample is the same.
Result Type	Chart Data or Metrics During an SPC analysis two types of data are generated. Calculations that are sample specific (Chart Data) and those that are dataset specific (Metrics). For example in a p chart PBar is calculated at the dataset level. Chart Data is always required for construction and display of a control chart. Metrics is used for display in a table as textual information or can be plotted in the chart.
Include Original Data	Yes/No. Determines if original data has to be included in the output. Default is Yes.

Unit Test

Dataset:

Dataset 1	Dataset 2	Dataset 3	Dataset 4
Varying sample size	Constant sample size	Failed per 1000	Varying sample size
SAMPLE TESTED FAILED	SAMPLE TESTED FAILED	SAMPLE FAILED	SAMPL TESTED FAILED

ID			ID			ID		E ID		
1	286	14	1	100	10	1	6	1	110	120
2	281	22	2	100	12	2	7	2	82	94
3	310	9	3	100	10	3	7	3	96	89
4	313	19	4	100	11	4	6	4	115	162
5	293	21	5	100	6	5	8	5	108	150
6	305	18	6	100	7	6	6	6	56	82
7	322	16	7	100	12	7	5	7	120	143
8	316	16	8	100	10	8	6	8	98	134
9	293	21	9	100	6	9	1	9	102	97
10	287	14	10	100	11	10	6	10	115	145
11	307	15	11	100	9	11	2	11	88	128
12	328	16	12	100	14	12	5	12	71	83
13	296	21	13	100	16	13	5	13	95	120
14	296	9	14	100	21	14	4	14	103	116
15	317	25	15	100	20	15	3	15	113	127
16	297	15	16	100	12	16	3	16	85	92
17	283	14	17	100	11	17	2	17	101	140
18	321	13	18	100	6	18	0	18	42	60
19	317	10	19	100	10	19	0	19	97	121
20	307	21	20	100	10	20	1	20	92	108
21	317	19	21	100	11	21	2	21	100	131
22	323	23	22	100	11	22	5	22	115	119
23	304	15	23	100	11	23	1	23	99	93
24	304	12	24	100	6	24	7	24	57	88
25	324	19	25	100	9	25	7	25	89	107
26	289	17				26	1	26	101	105
27	299	15				27	5	27	122	143
28	318	13				28	5	28	105	132
29	313	19				29	8	29	98	100
30	289	12				30	8	30	48	60

Note:

1. For all the charts the parameter “Include Original Data” has been set to “No”.
2. Result has been shown separately for the parameter “Result Type” options.

Chart Type: **p**

Input: Data Set 1

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sample Size	1
Column Index Nonconformance	2
Chart Type	p

Result Type: Chart Data

SAMPLE_ID	SAMPLE_SIZE	FAILED	PBAR	LCL	UCL	PHAT
1	286	14	0.05391399	0.01384999	0.09397799	0.04895104
2	281	22	0.05391399	0.01349512	0.09433286	0.07829181
3	310	9	0.05391399	0.01543209	0.09239589	0.02903225
4	313	19	0.05391399	0.01561696	0.09221103	0.06070287
5	293	21	0.05391399	0.01433147	0.09349652	0.07167235
6	305	18	0.05391399	0.01511795	0.09271003	0.05901639
7	322	16	0.05391399	0.01615596	0.09167203	0.04968944
8	316	16	0.05391399	0.01579918	0.09202880	0.05063291
9	293	21	0.05391399	0.01433147	0.09349652	0.07167235
10	287	14	0.05391399	0.01391985	0.09390813	0.04878048
11	307	15	0.05391399	0.01524453	0.09258346	0.04885993
12	328	16	0.05391397	0.01650290	0.09132501	0.04878048
13	296	21	0.05391399	0.01453256	0.09329542	0.07094594
14	296	9	0.05391399	0.01453256	0.09329542	0.03040540
15	317	25	0.05391399	0.01585935	0.09196864	0.07886435
16	297	15	0.05391399	0.01459892	0.09322907	0.05050505
17	283	14	0.05391399	0.01363820	0.09418979	0.04946996
18	321	13	0.05391399	0.01609719	0.0917308	0.04049844
19	317	10	0.05391399	0.01585935	0.09196864	0.03154574
20	307	21	0.05391399	0.01524453	0.09258346	0.06840390
21	317	19	0.05391399	0.01585935	0.09196864	0.05993690
22	323	23	0.05391399	0.01621445	0.09161358	0.07120743
23	304	15	0.05391399	0.01505419	0.09277379	0.04934215
24	304	12	0.05391399	0.01505419	0.09277379	0.03947368
25	324	19	0.05391399	0.01627267	0.09155531	0.05864197
26	289	17	0.05391399	0.01405848	0.09376951	0.05882352
27	299	15	0.05391399	0.01473063	0.09309736	0.05016722
28	318	13	0.05391399	0.01591923	0.09190876	0.04088050
29	313	19	0.05391399	0.01561696	0.09221103	0.060702875
30	289	12	0.053913997	0.014058484	0.09376951	0.041522491

Result Type: Metrics

P Chart Limits	
PBar	0.053913997

Chart Type: **np**

Input: Data Set 2

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sample Size	1
Column Index Nonconformance	2
Chart Type	np

Result Type: Chart Data

SAMPLE_ID	SAMPLE_SIZE	FAILED	CL	LCL	UCL
1	100	10	10.88	1.538356462	20.22164354
2	100	12	10.88	1.538356462	20.22164354
3	100	10	10.88	1.538356462	20.22164354
4	100	11	10.88	1.538356462	20.22164354
5	100	6	10.88	1.538356462	20.22164354
6	100	7	10.88	1.538356462	20.22164354
7	100	12	10.88	1.538356462	20.22164354
8	100	10	10.88	1.538356462	20.22164354
9	100	6	10.88	1.538356462	20.22164354
10	100	11	10.88	1.538356462	20.22164354
11	100	9	10.88	1.538356462	20.22164354
12	100	14	10.88	1.538356462	20.22164354
13	100	16	10.88	1.538356462	20.22164354
14	100	21	10.88	1.538356462	20.22164354
15	100	20	10.88	1.538356462	20.22164354
16	100	12	10.88	1.538356462	20.22164354
17	100	11	10.88	1.538356462	20.22164354
18	100	6	10.88	1.538356462	20.22164354
19	100	10	10.88	1.538356462	20.22164354
20	100	10	10.88	1.538356462	20.22164354
21	100	11	10.88	1.538356462	20.22164354
22	100	11	10.88	1.538356462	20.22164354
23	100	11	10.88	1.538356462	20.22164354
24	100	6	10.88	1.538356462	20.22164354
25	100	9	10.88	1.538356462	20.22164354

Result Type: Metrics

NP Chart Limits	
NPBar	0.1088

Chart Type: **c**

Input: Data Set 3

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sample Size	
Column Index Nonconformance	1
Chart Type	c

Result Type: Chart Data

SAMPLE_ID	FAILED	CBAR	LCL	UCL
1	6	4.4	0	10.69285309
2	7	4.4	0	10.69285309
3	7	4.4	0	10.69285309
4	6	4.4	0	10.69285309
5	8	4.4	0	10.69285309
6	6	4.4	0	10.69285309
7	5	4.4	0	10.69285309
8	6	4.4	0	10.69285309
9	1	4.4	0	10.69285309
10	6	4.4	0	10.69285309
11	2	4.4	0	10.69285309
12	5	4.4	0	10.69285309
13	5	4.4	0	10.69285309
14	4	4.4	0	10.69285309
15	3	4.4	0	10.69285309
16	3	4.4	0	10.69285309
17	2	4.4	0	10.69285309
18	0	4.4	0	10.69285309
19	0	4.4	0	10.69285309
20	1	4.4	0	10.69285309
21	2	4.4	0	10.69285309
22	5	4.4	0	10.69285309
23	1	4.4	0	10.69285309
24	7	4.4	0	10.69285309
25	7	4.4	0	10.69285309
26	1	4.4	0	10.69285309
27	5	4.4	0	10.69285309
28	5	4.4	0	10.69285309
29	8	4.4	0	10.69285309
30	8	4.4	0	10.69285309

Result Type: Metrics

C Chart Limits	
CBar	4.4

Chart Type: **u**

Input: Data Set 4

Parameter:

Name	Value
Column Index Sample Identity	0
Column Index Sample Size	1
Column Index Nonconformance	2
Chart Type	u

Result Type: Chart Data

SAMPLE_ID	SAMPLE_SIZE	FAILED	UBAR	LCL	UCL	Non-Conf/Unit
1	110	120	1.200495926	0.887091405	1.513900448	1.090909091
2	82	94	1.200495926	0.837505915	1.563485937	1.146341463
3	96	89	1.200495926	0.865016429	1.535975424	0.927083333
4	115	162	1.200495926	0.893980258	1.507011595	1.408695652
5	108	150	1.200495926	0.884202823	1.51678903	1.388888889
6	56	82	1.200495926	0.761250158	1.639741695	1.464285714
7	120	143	1.200495926	0.900433942	1.500557911	1.191666667
8	98	134	1.200495926	0.868457335	1.532534517	1.367346939
9	102	97	1.200495926	0.875033008	1.525958845	0.950980392
10	115	145	1.200495926	0.893980258	1.507011595	1.260869565
11	88	128	1.200495926	0.850099019	1.550892833	1.454545455
12	71	83	1.200495926	0.810399092	1.59059276	1.169014085
13	95	120	1.200495926	0.86325537	1.537736483	1.263157895
14	103	116	1.200495926	0.876616779	1.524375074	1.126213592
15	113	127	1.200495926	0.891279627	1.509712226	1.123893805
16	85	92	1.200495926	0.843969162	1.55702269	1.082352941
17	101	140	1.200495926	0.873425774	1.527566079	1.386138614
18	42	60	1.200495926	0.693298601	1.707693252	1.428571429
19	97	121	1.200495926	0.866750185	1.534241668	1.24742268
20	92	108	1.200495926	0.857800991	1.543190862	1.173913043
21	100	131	1.200495926	0.871794491	1.529197361	1.31
22	115	119	1.200495926	0.893980258	1.507011595	1.034782609
23	99	93	1.200495926	0.870138554	1.530853298	0.939393939
24	57	88	1.200495926	0.76512024	1.635871613	1.543859649
25	89	107	1.200495926	0.852073102	1.548918751	1.202247191
26	101	105	1.200495926	0.873425774	1.527566079	1.03960396
27	122	143	1.200495926	0.90290363	1.498088223	1.172131148
28	105	132	1.200495926	0.879716172	1.521275681	1.257142857
29	98	100	1.200495926	0.868457335	1.532534517	1.020408163
30	48	60	1.200495926	0.726056271	1.674935581	1.25

Result Type: Metrics

U Chart Limits	
UBar	1.200495926

<table>
<tr><td>Module Name</td><td>Quantile</td><td>Module Type</td><td>Functional</td></tr>
<tr><td>Description</td><td colspan="3">Quantiles are essentially points taken at regular vertical intervals from the cumulative distribution function of a random variable. Dividing ordered data into equal sized data subsets is the motivation for quantiles. The quantiles are the data values marking the boundaries between consecutive subsets. This module provides for determining the percentiles, deciles, quintiles, quartiles and the median.</td></tr>
<tr><td>Parameters</td><td colspan="3">
<table>
<tr><th>Parameter Name</th><th>Description</th></tr>
<tr><td>Column Index Of Item Values</td><td>Column on which to perform the Quantile analysis. Column numbers are zero indexed.</td></tr>
<tr><td>Quantile Type</td><td>Percentiles/Deciles/Quintiles/Quartiles/Median</td></tr>
<tr><td>Include Original Data</td><td>If “Yes” include the original dataset in the result table</td></tr>
</table>
</td></tr>
</table>

Unit Tests

Quantile Type: Percentile

Input	Parameter	Result		
ITEM_VALUE 106 109 114 116 121 122 125 129	Column Index Of Item Values = 0 Quantile Type = Percentile Include Original Data = Yes	ITEM VALUE	PERCENTILES	PERCENTILE POINT
		106	1	106
		109	2	106
		114	…	106
		116	12	106
		121	13	109
		122	...	109
		125	24	109
		129	25	111.5
			26	114
			…	114
			37	114
			38	116
			…	116
			49	116
			50	118.5
			51	121
			…	121
			62	121
			63	122
			…	122
			74	122
			75	123.5
			76	125
			…	125
			87	125
			88	129
			…	129
			99	129

Unit Tests Contd.

Quantile Type: Deciles

Input	Parameter	Result		
ITEM_VALUE 106 109 114 116 121 122 125 129	Column Index Of Item Values = 0 Quantile Type = Deciles Include Original Data = Yes	ITEM VALUE	DECILES	DECILE POINT
		106	1	106
		109	2	109
		114	3	114
		116	4	116
		121	5	118.5
		122	6	121
		125	7	122
		129	8	125
			9	129

Quantile Type: Quintiles

Input	Parameter	Result		
ITEM_VALUE 106 109 114 116 121 122 125 129	Column Index Of Item Values = 0 Quantile Type = Quintiles Include Original Data = Yes	ITEM VALUE	QUINTILES	QUINTILE POINT
		106	1	109
		109	2	116
		114	3	121
		116	4	125
		121		
		122		
		125		
		129		

Unit Tests Contd.

Quantile Type: Quartiles

Input	Parameter	Result
ITEM_VALUE 106 109 114 116 121 122 125 129	Column Index Of Item Values = 0 Quantile Type = Quartiles Include Original Data = Yes	ITEM_VALUE / QUARTILE / QUARTILE POINT 106 / 1 / 111.5 109 / 2 / 118.5 114 / 3 / 123.5 116 121 122 125 129

Quantile Type: Median

Input	Parameter	Result
ITEM_VALUE 106 109 114 116 121 122 125 129	Column Index Of Item Values = 0 Quantile Type = Median Include Original Data = Yes	ITEM_VALUE / MEDIAN / MEDIAN POINT 106 / 1 / 118.5 109 114 116 121 122 125 129

<table>
<tr><td>Module Name</td><td>Mode</td><td>Module Type</td><td>Functional</td></tr>
<tr><td>Description</td><td colspan="3">In a set of data, the mode is the most frequently observed data value. There may be no mode if no value appears more than any other. There may also be two modes (bimodal), three modes (trimodal) or more modes (multimodal). In case of grouped frequency distributions, the modal class is the class with the largest frequency.</td></tr>
<tr><td>Parameters</td><td colspan="3">
<table>
<tr><th>Parameter Name</th><th>Description</th></tr>
<tr><td>Column Index Of Item Values</td><td>Column on which to perform the Mode analysis. Column numbers are zero indexed.</td></tr>
<tr><td>Mode Type</td><td>Ungrouped/Grouped

If Mode Type is Ungrouped the result will be the most frequently observed data value/s.

If Mode Type is Grouped, based on bin size, the result will be the class/s that has the most number of observed data value.</td></tr>
<tr><td>Bin Size</td><td>This is the width of each bin. Number of bins will be determined by dividing the difference between highest observed value and lowest observed value by bin size.</td></tr>
<tr><td>Include Original Data</td><td>If “Yes” include the original dataset in the result table</td></tr>
</table>
</td></tr>
</table>

Unit Tests

Mode Type: Ungrouped

Input	Parameter	Result
FAILED 14 14 15 16 14 16 16 18 14 16 16 14	Column Index Of Item Values = 0 Mode Type = Ungrouped Bin Size = <blank> Include Original Data = Yes	FAILED ITEM COUNT 14 14 5 14 16 5 15 16 14 16 16 18 14 16 16 14

Mode Type: Grouped

Input	Parameter	Result
FAILED 1 15 23 35 50 43 32 45 23 34 2 19 23 45 72 -12 0 34 2	Column Index Of Item Values = 0 Mode Type = Grouped Bin Size = 2 Include Original Data = Yes	FAILED CLASS MIN MAX MID COUNT 1 8 0 2 1 3 15 19 22 24 23 3 23 35 50 43 32 45 23 34 2 19 23 45 72 -12 0 34 2

Appendix

Regression

Regression Type	Regression Equation
POLYNOMIAL	$y = b_0 + b_1x^1 + b_2x^2 + b_3x^3 + b_4x^4 + \ldots b_nx^n$
EXPONENTIAL	y=a*e^(b*x)
HYPERBOLA	y=x/(a*x + b)
LOGARITHMIC	y=a + b*ln(x) + c*(ln(x))^2 + …
INVERSE	y=a + b1*(1/x) + b2*(1/x)^2 + b3*(1/x)^3 + …
INVERSE_POWER	y=a*e^(b/x)
POLYEXP	y=a + b1*e^x + b2*e^2x + …
POWER	y=a*x^b

Fourier, Filter and Track Order

Any set of periodic data (ie. a paired (x,y) data set where the dependent variable y repeats itself periodically with respect to the independent variable x) can be approximated by a superposition of sine and cosine curves with varying time period. The Fourier, Filter and Track Order Process Modules are based on the decomposition of a periodic data set into components corresponding to different orders of periodicity by performing a fit to the following Fourier series equation:

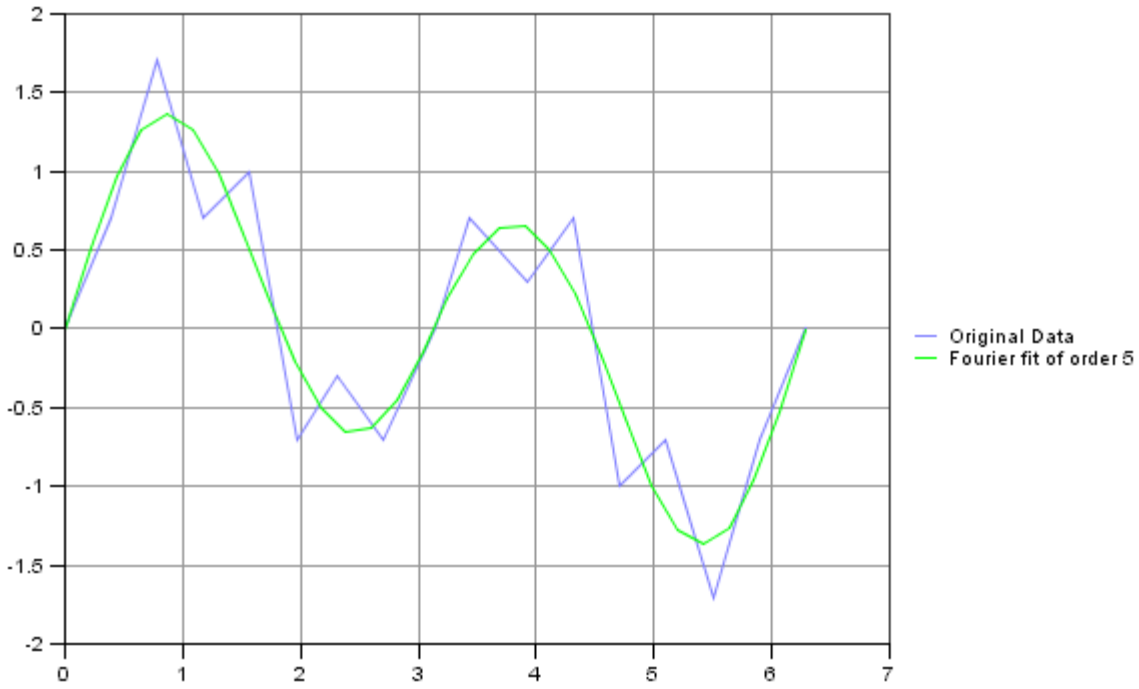
$y = b_0 + b_1\cos(x) + b_2\sin(x) + b_3\cos(2*x) + b_4\sin(2*x) + \ldots\ldots + a\cos(n*x) + c\sin(n*x)$
where n is the selected order for the Fourier fit.

The Fourier module calculates the best approximation to the data that can be obtained by fitting it to a Fourier series of the selected order.

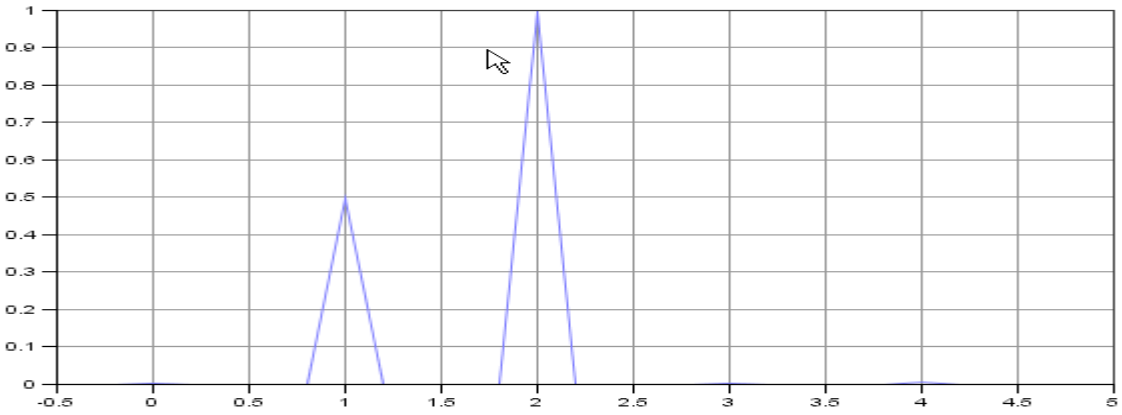
The Filter module allows the user to select which orders they wish to retain within the Fourier fit. This could be done to eliminate noise from a signal for instance.

The track order module allows the user to see which orders exist in the data and the corresponding magnitudes.

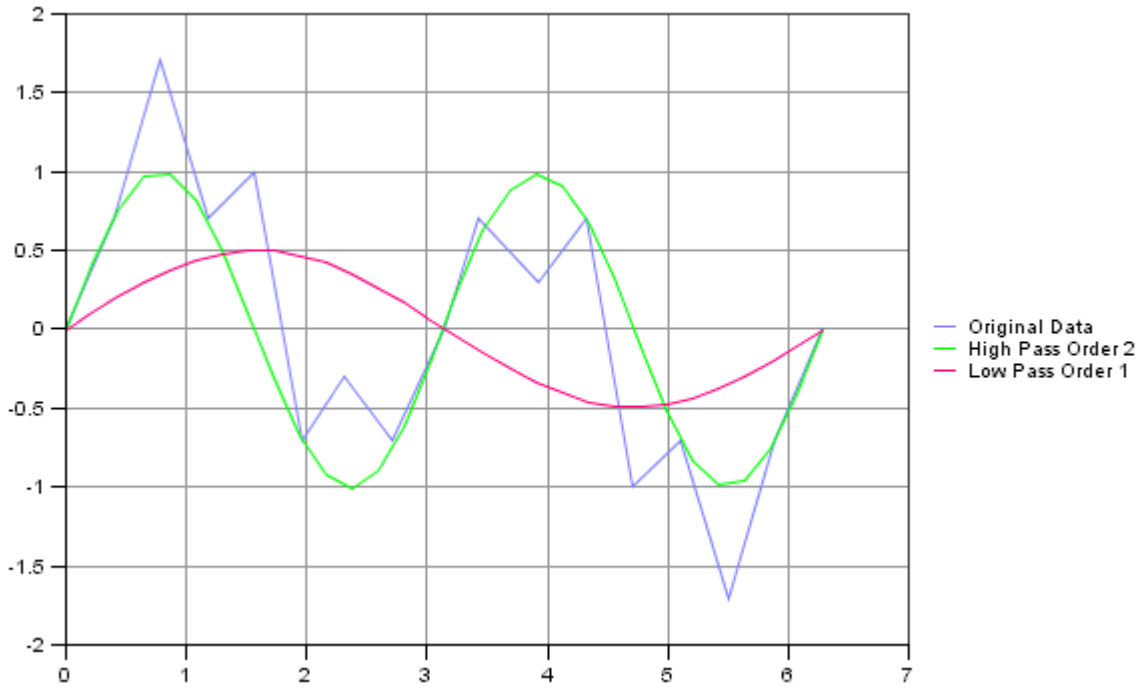
The plot shown below displays a set of data along with a Fourier fit of order 5 performed on it.



Track order analysis of the same data (shown below) reveals that the two dominant orders in the above Fourier fit are 2 and 1 with approximate magnitudes of 1 and 0.5 respectively.



In the following plot, a high pass filter of the same data with first cut-off order set to 2 retains only the 2nd order component of the Fourier series and results in the green line. A low pass filter of the data with second cut-off order set to 1 retains only the 1st order component of the Fourier series and results in the red line.



Univariate Metrics

The formulae used to calculate univariate metrics for a data set Y of length N is given in the table below.

Metric	Equation
Mean or average	$\bar{Y} = \sum_{i=1}^{N} Y_i / N$
Median	The median is the value of the point which has half the data smaller than that point and half the data larger than that point. $\tilde{Y} = Y_{(N+1)/2}$ if N is odd $\tilde{Y} = (Y_{N/2} + Y_{(N/2)+1})/2$ if N is even

Variance	$s^2 = \sum_{i=1}^{N}(Y_i - \bar{Y})^2/(N-1)$
Standard deviation	$s = \sqrt{\sum_{i=1}^{N}(Y_i - \bar{Y})^2/(N-1)}$
Standard error of mean	standard deviation/square root(n)
Skewness	$skewness = \frac{\sum_{i=1}^{N}(Y_i - \bar{Y})^3}{(N-1)s^3}$
Kurtosis	$kurtosis = \frac{\sum_{i=1}^{N}(Y_i - \bar{Y})^4}{(N-1)s^4}$