



This file includes examples of User Defined Functions that create sparklines, the simple, intense, word-size graphics invented by Edward Tufte.

All parameters can now be passed **directly** inside the formulas

Disclaimer : This version of Sparklines has been developped on Excel 2003 - It is not fully fonctionnal on Excel 2000 nor on Excel 2007

Barchart(Points ; Min ; Max ; Line1; Line2 ; ColorChange ; ColorPositive ; ColorNegative)

Scalable sparkline with color change threshold and horizontal target line

Points : range of cells containing all values to be displayed (row or column)

ShowLine : (Boolean) optional - if set at TRUE, the horizontal lines are displayed.

ColorPositive and ColorNegative : Long optional - you can choose the color code - if omitted, standard color is grey + red

Max (Range)

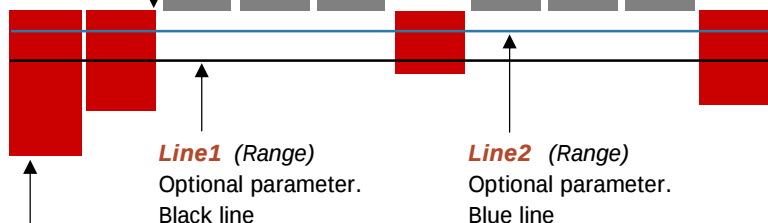
Optional parameter.

If not specified then takes the maximum value of all Points

ColorChange (Range)

Optional parameter.

If not specified then color changes at Zero.



Min (Range)

Optional parameter.

If not specified then takes the minimum value of all Points



If you already use Sparklines,
note that the parameter
ShowLine has been removed

LineChart(Points1 ;Points2 ; Min ; Max ; Line ; MaxZone; MinZone;Tags ; ColorLine1 ; ColorLine2)

Points 1 and 2: cell ranges containing all values to be displayed (row or column)
 Points1 is a compulsory parameter - Points2 is optional, displayed in blue.

Max (Range - optional)

If not specified then takes the maximum value of all Points

Tags (Boolean - Optional)

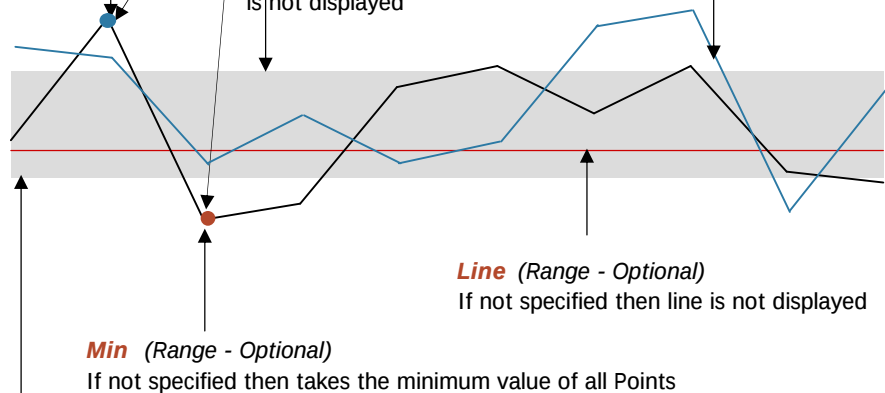
If "TRUE", displays red and blue dots for Min and Max values

MaxZone (Range - Optional)

If not specified grey zone is not displayed

Points2 (Range - Optional)

If not specified, not displayed



MinZone (Range - Optional)

Optional parameter.

If not specified grey zone is not displayed



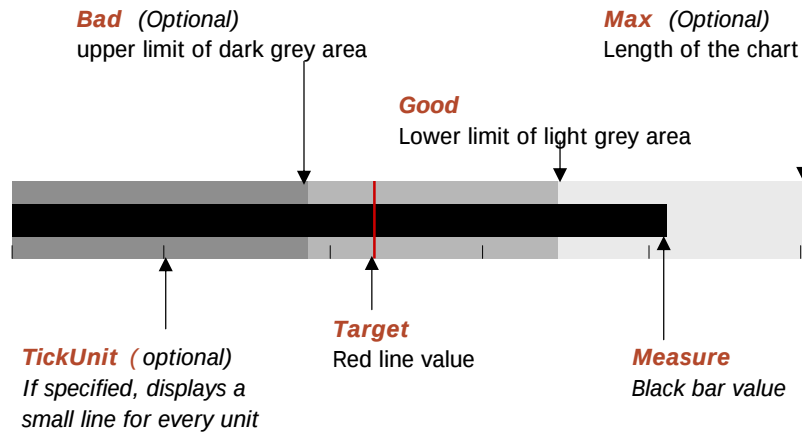
MaxZone and MinZone not specified

Line = Zero



Added parameters for
 color of
 lines 1 + 2

BulletChart(Measure ; Target ; Max ; Good ; Bad ; TickUnit ; ColorScheme)



ColorScheme

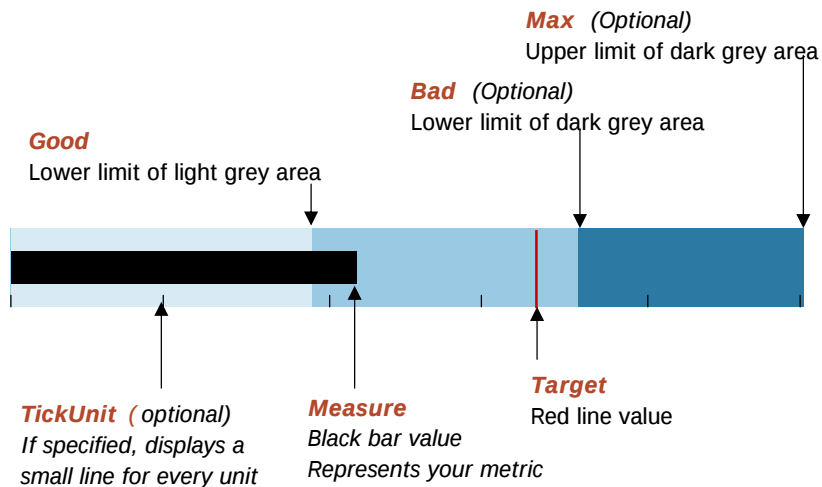
Long value of the color
Based on RGB code :

$$= 65536 * \text{Blue} + 256 * \text{Green} + \text{Red}$$

Make sure that :

Bad, Good, Target and Measure are smaller than Max (or the function will adjust it !)

RevBulletChart(Measure ; Target ; Max ; Good ; Bad ; TickUnit ; ColorScheme)



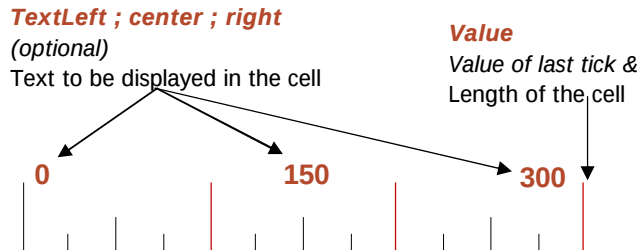
ColorScheme

Long value of the color
Based on RGB code :

$$= 65536 * \text{Blue} + 256 * \text{Green} + \text{Red}$$

ScaleLine(Value ; TextLeft ; TextCenter ; TextRight ; SizeText) **Inv**ScaleLine(Value ; TextLeft ; TextCenter ; TextRight ; SizeText)

InvScaleLine displays the grid upside down → 
Interval between **Red** lines = 100
Interval between **Black** lines = 25



Example : Scaleline(300;0;150;300)

SizeFont = well no real need to explain this one ... integer values only (no half-sizes allowed)

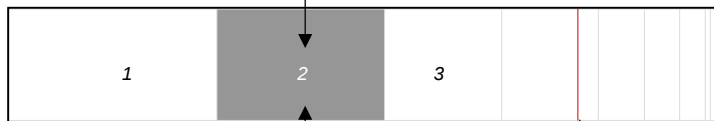
Pareto(Points ; Position ; Target ; PosColor)

Pareto displays a horizontal stacked chart of a series of values and colors a specific one.
Additionally, a red line is drawn to highlight a limit

Points : Range containing all values to be displayed (row or column)

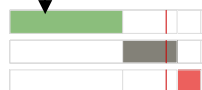
Position

Position of the rectangle you want to highlight
in the Points serie (in this example Position=2)



PosColor (Optional)

Sets the color of the
highlighted cell.
Values are 1, 2 and 3



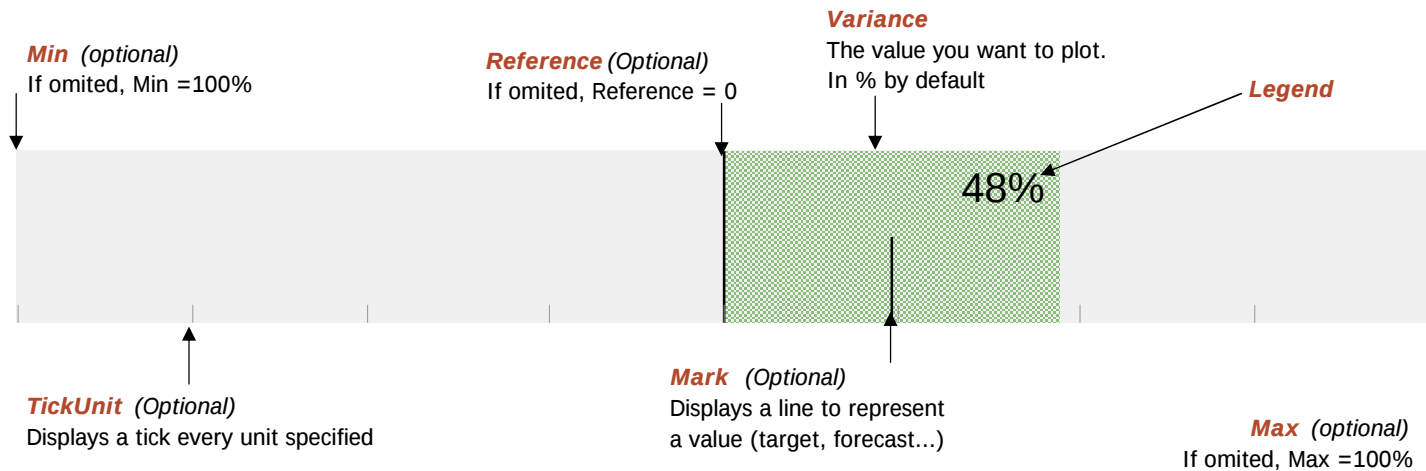
Target (Optional)

Red line value
value = 0 to 100%

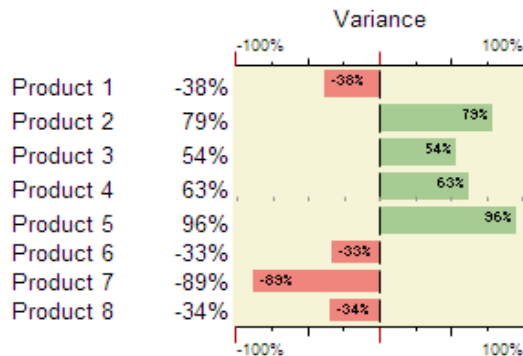
Poscolor values (hardcoded)

- 1 - Green
- 2 - Grey
- 3 - Red

VariChart(Variance ; Reference ; Min ; Max ; Mark ; TickUnit ; Legend ; ColorPositive ; ColorNegative)



Legend : Boolean parameter. If **Legend** = True, then the value of **Variance** is displayed, in % only
Position of the text changes according to the sign (+/-) of Variance



ColorPositive and ColorNegative

Long value of the color

Based on RGB code :

= 65536 *Blue + 256*Green+ Red

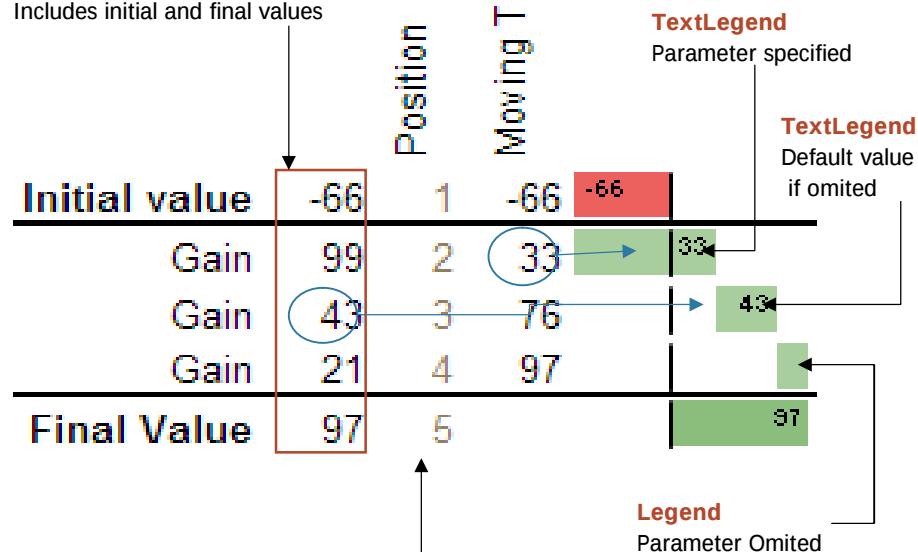
CascadeChart(Points ; Position ; Legend ; TextLegend ; ColorPositive ; ColorNegative)

Legend : Boolean parameter. If **Legend** = True, the value of the Point is displayed by default

TextLegend : optional parameter. Use it if you want to display a different value than the value of Point (Running total in the example)

POINTS

Range of values to be displayed
Includes initial and final values



POSITION

Rank of the value to be displayed
inside the range of points

Default Legend = Gain or Loss (Column M)

Initial value	-67	Σ	-67
Loss	-26	-93	-26
Gain	58	-35	58
Gain	35	0	35
Gain	43	43	43
Final Value	43		43

ColorPositive and ColorNegative

Long value of the color

Based on RGB code :

$$= 65536 * \text{Blue} + 256 * \text{Green} + \text{Red}$$

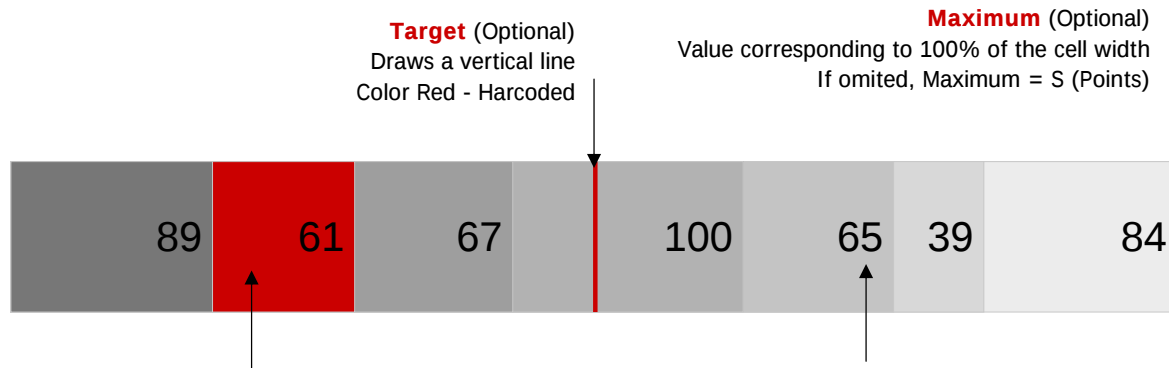
Tip : you can calculate the parameter Position, inside the formula
=cascadeChart(\$M\$10:\$M\$23;COUNT(\$M\$10:M10);TRUE)

StackedChart(Points ; Maximum ; Target ; Color ; HighlightPosition ; Legend)

Points : Range containing all values to be displayed (row or column)

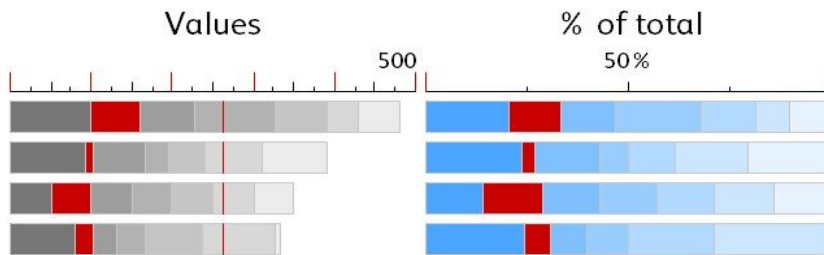
Color (Optional) : calculated with the RGB value of a color. **Color** = **Red** + **256 x Green** + **65536 x Blue**

If color is omitted, standard color is Grey RGB(120,120,120)



HighlightPosition (Optional)
Position of the rectangle you want to highlight
in the Points serie (in this example HighlightPosition=2)
Default color is Red - hardcoded

Legend : Boolean parameter.
If **Legend = True**,
the value of the Point is displayed



Maximum
Specified = 500

Color
Omitted

Target
Specified = 263

Maximum
Omitted. Maximum = Σ (Points)
The entire cell width is used
Cell width = 100%

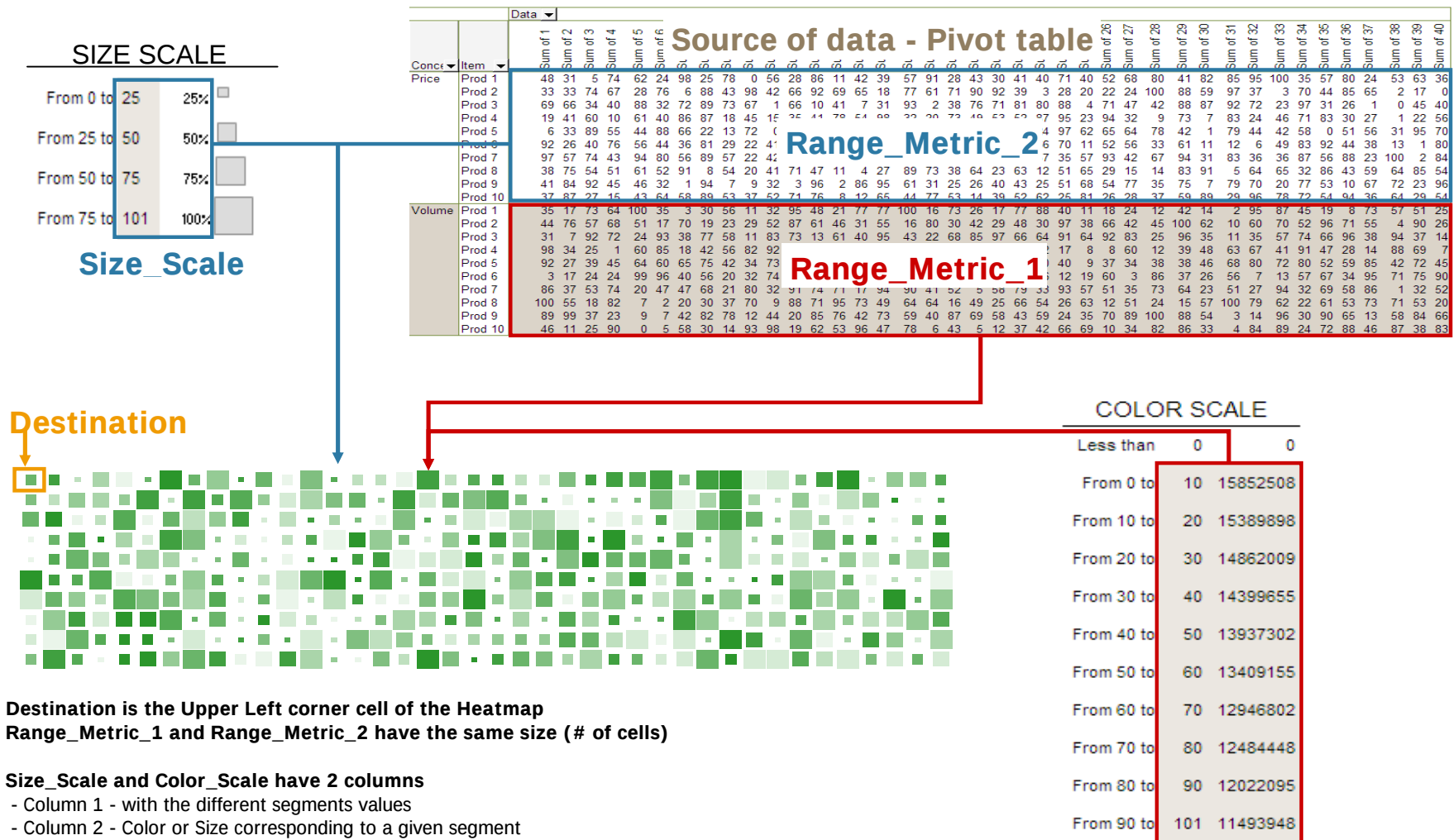
Color
Specified = 16750899

Target
Omitted

Suggested color codes

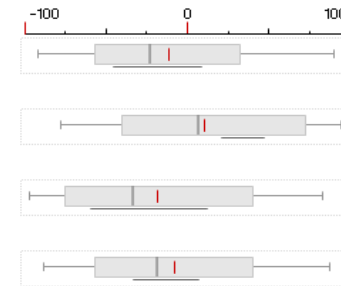
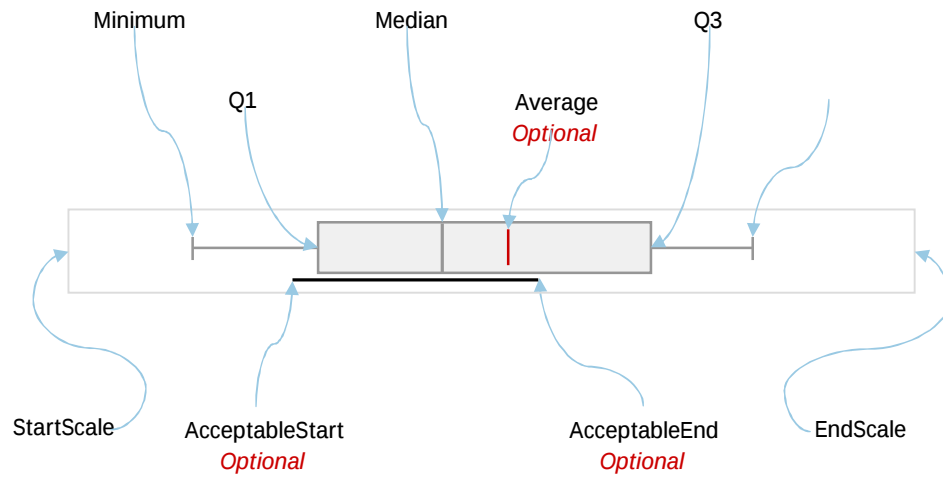
Green	4 953 610		Brown	3 368 601	
Blue	9 843 200		Plum	3 342 438	
Yellow	369 401		Turquoise	10 079 232	
Orange	26 367		Light green	3 394 611	
Red	203		Light blue	16 750 899	

=HeatMap(**Destination** ; **Range_Metric_1** ; **Color_Scale** ; **Range_Metric_2** ; **Size_Scale** ; **Circles**)

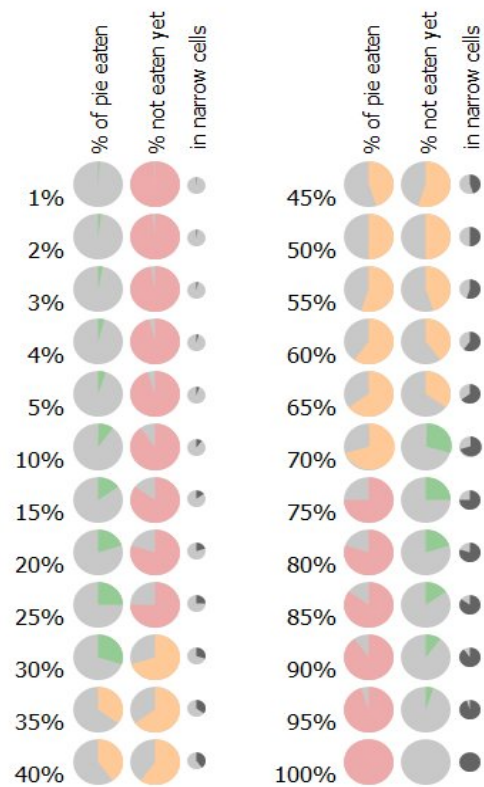


=BoxPlotChart(Minimum; Q1; Median; Q3; Maximum; Average; StartScale; EndScale; AcceptableStart; AcceptableEnd; ColorScheme)

Code contribution : NixNut



=Piechart(Percentage ; ColorBackground ; ColorSlice)



Percentage is a value between 0 and 1 (0% and 100%)

Colors = RGB() values in "long" format

30%	9816981
70%	10013439
100%	11184878

Standard colors :

