

# Interactive charting using FSharp.Plotly



# Plotly.js

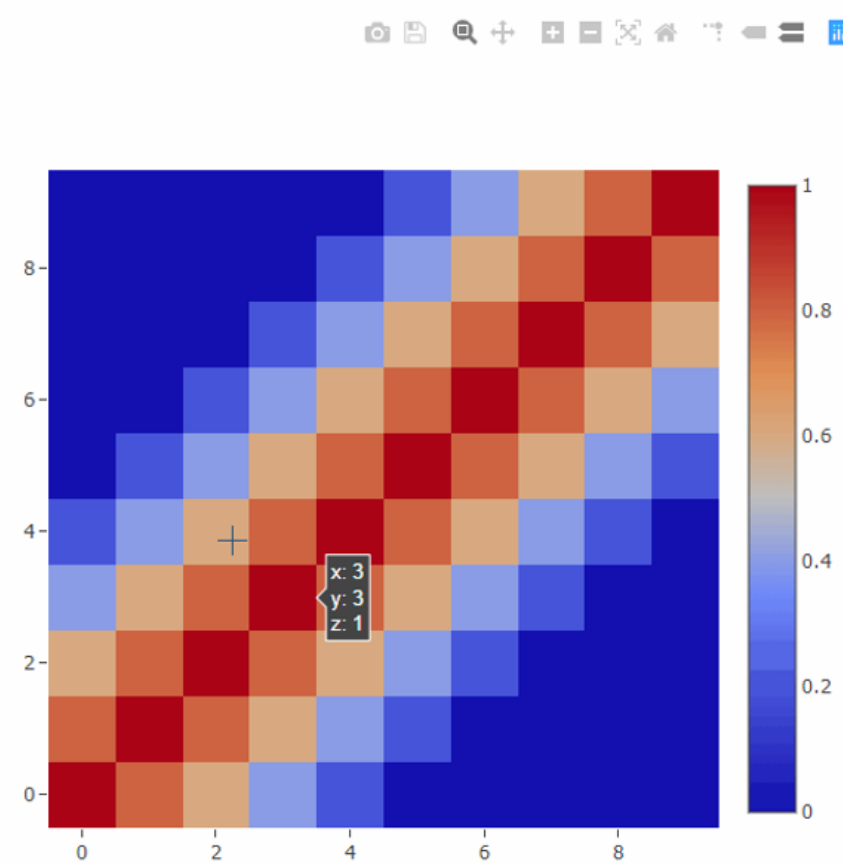
- Open source charting library
- Supports scientific charts, 3D graphs, statistical charts, etc.
- All charts are interactive





# Plotly.js

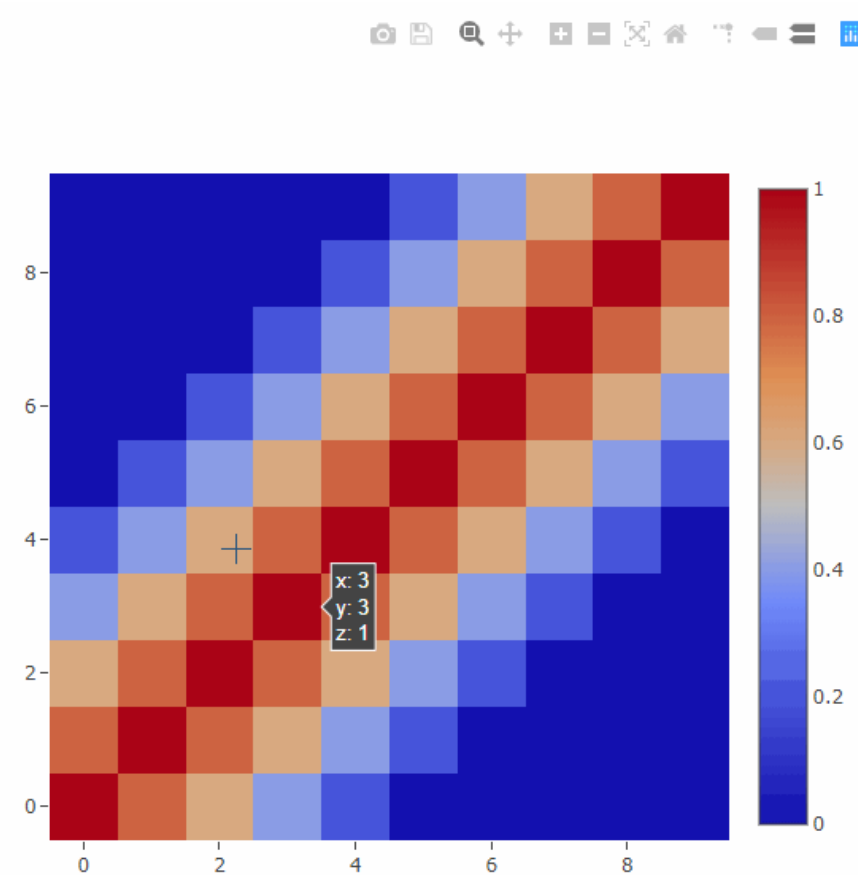
- Plotly charts are interactive:





# Plotly.js

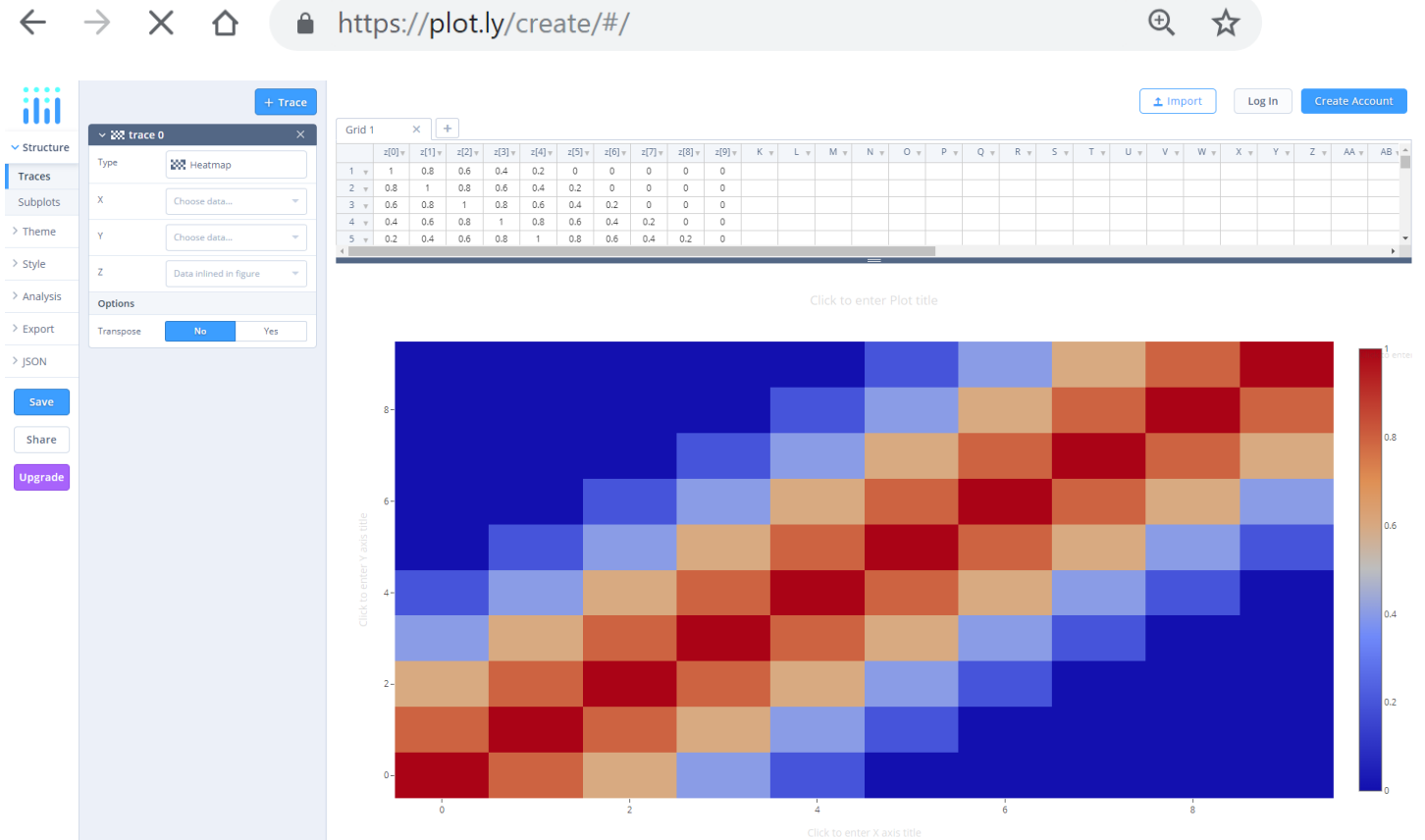
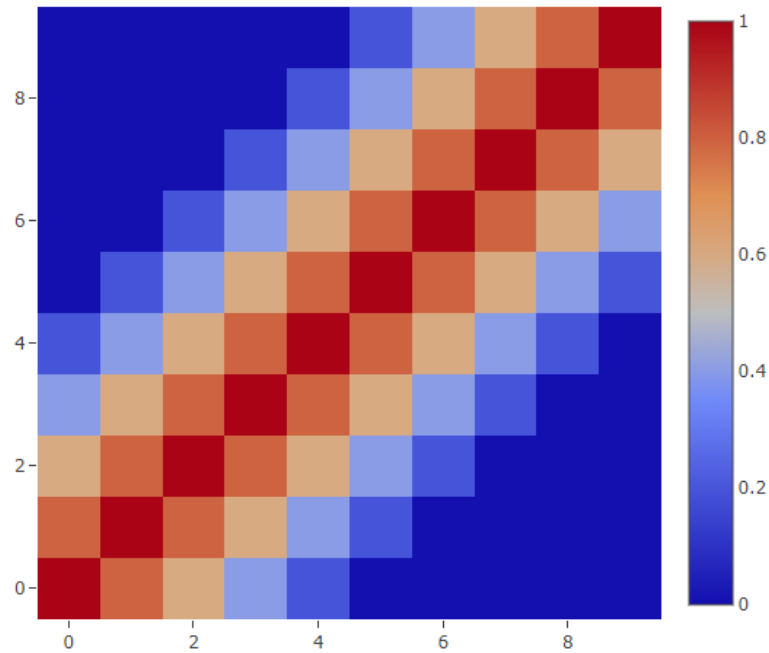
- Plotly charts are interactive:

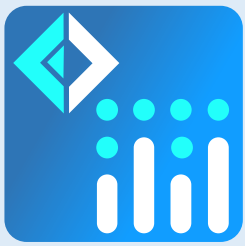




# Plotly.js

- Plotly.js is a service





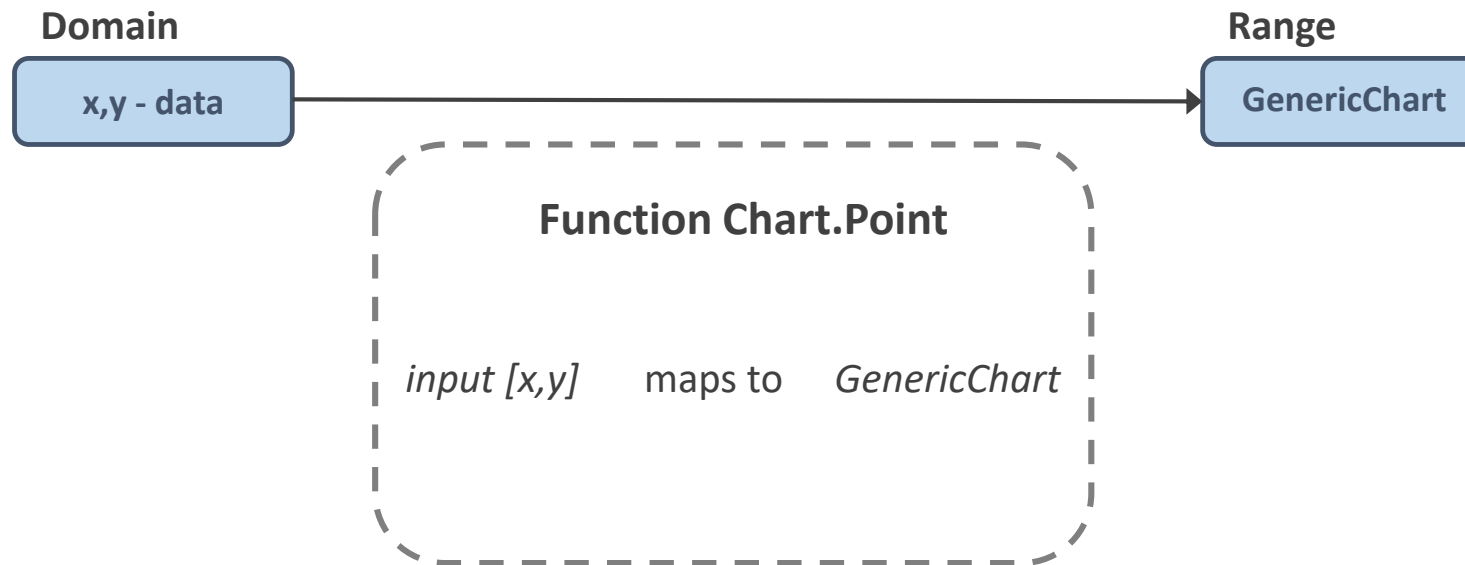
# FSharp.Plotly

- Open source F# wrapper for plotly.js
- Designed to work with many programming styles (object oriented, functional, mixtures,...)
- Documentation and tutorials can be found at:



<http://muehlhaus.github.io/FSharp.Plotly/>

# Chart functions



# Chart functions

Domain

x,y - data

Range

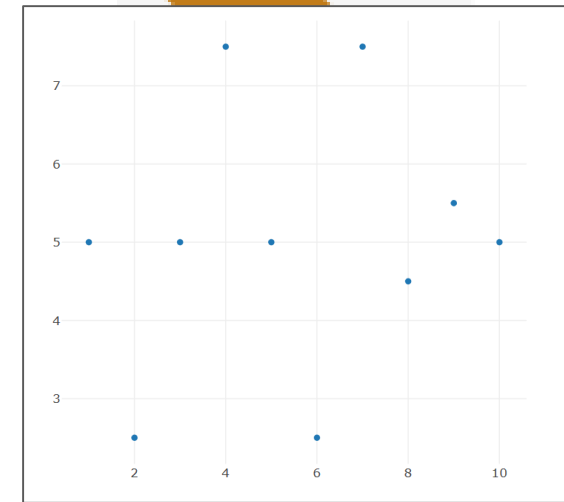
GenericChart

Function Chart.Point

[1.;2. ;3.;4. ;5.;6. ;7. ;8. ;9. ;10.; ]

[5.;2.5;5.;7.5;5.;2.5;7.5;4.5;5.5;5. ; ]

input [x,y] maps to GenericChart

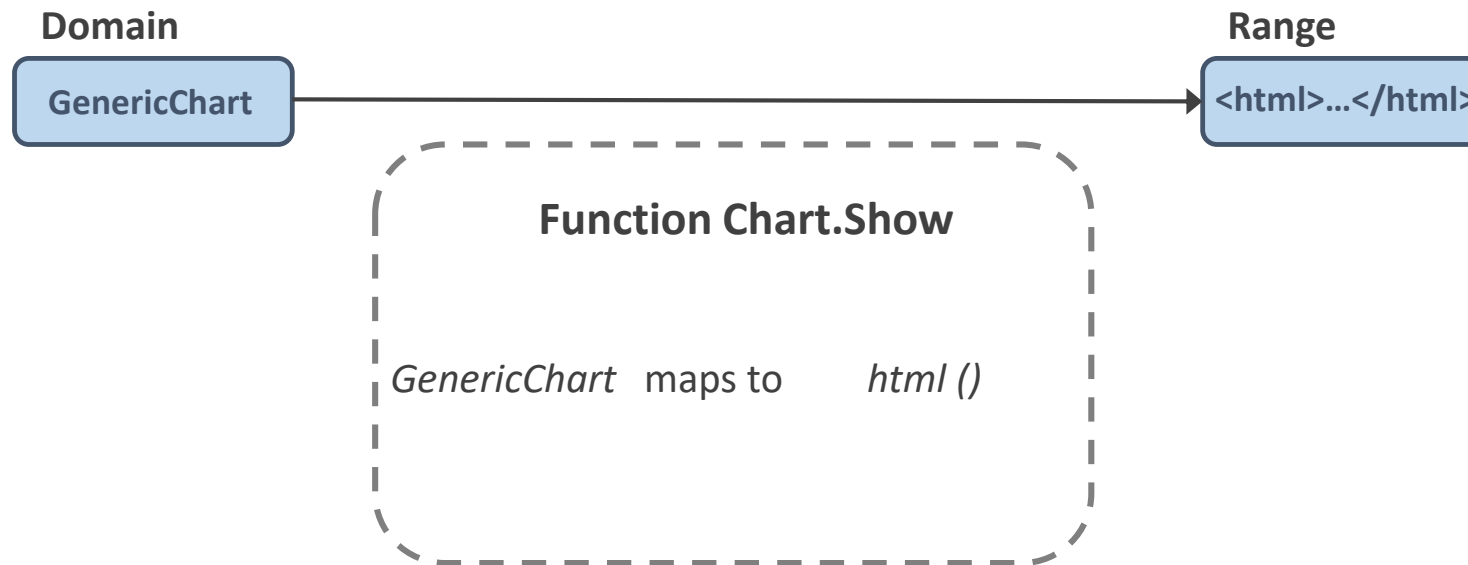




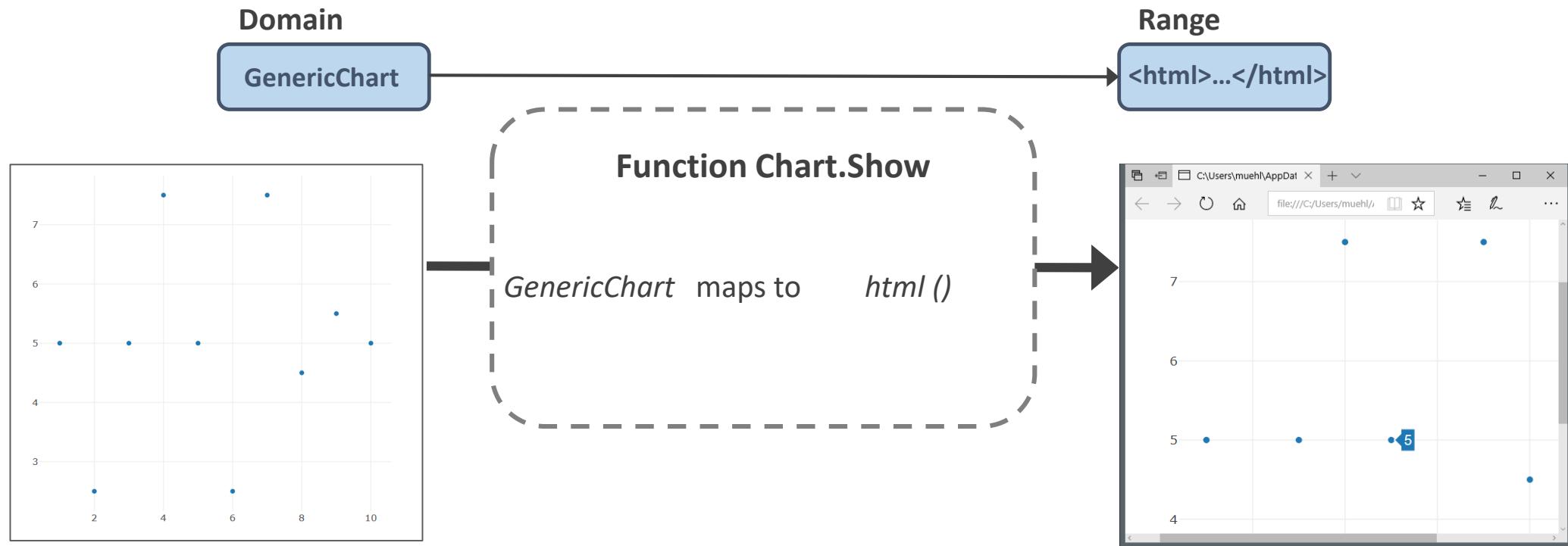
# Chart functions - Example code

```
let x = [1.; 2.; 3.; 4.; 5.; 6.; 7.; 8.; 9.; 10.; ]  
let y = [5.; 2.5; 5.; 7.5; 5.; 2.5; 7.5; 4.5; 5.5; 5.; ]  
Chart.Point(x, y)
```

# Rendering your charts



# Rendering your charts



# Rendering your charts - Example code

```
let x = [1.; 2.; 3.; 4.; 5.; 6.; 7.; 8.; 9.; 10.; ]  
let y = [5.; 2.5; 5.; 7.5; 5.; 2.5; 7.5; 4.5; 5.5; 5.; ]  
  
Chart.Point(x, y)  
|> Chart.Show
```

# Style your charts

## Additional parameters for chart functions:

```

... |> static member Point:
... |>   x      : seq<'a> *
... |>   y      : seq<'b> *
... |>   Name   : string option *
        Showlegend : bool option *
        MarkerSymbol: StyleParam.Symbol option *
float list Color    : 'c option *
let x · Opacity     : float option *
        Labels     : seq<'d> option *
float list TextPosition: StyleParam.TextPosition option *
let y · TextFont    : Font option
        -> GenericChart.GenericChart
  
```

Chart.Point(x,y)

## Chart extension functions:

```

Chart.with
  |> · Cha
  |> · Cha
  |> · Cha
  |> · Cha
  
```

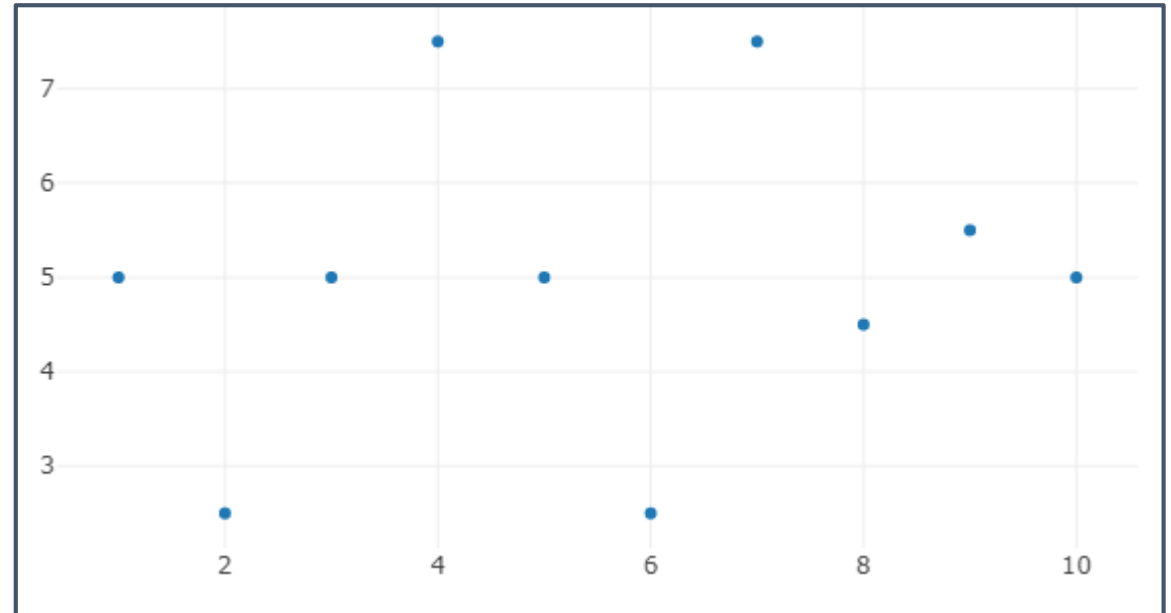
- withTraceName
- withXError
- withXErrorStyle
- withX\_Axis
- withX\_AxisStyle**
- withYError
- withYErrorStyle
- withY\_Axis
- withY\_AxisStyle

# Style your charts - Example code

```

Chart.Point(x,y)

|> Chart.Show
  
```

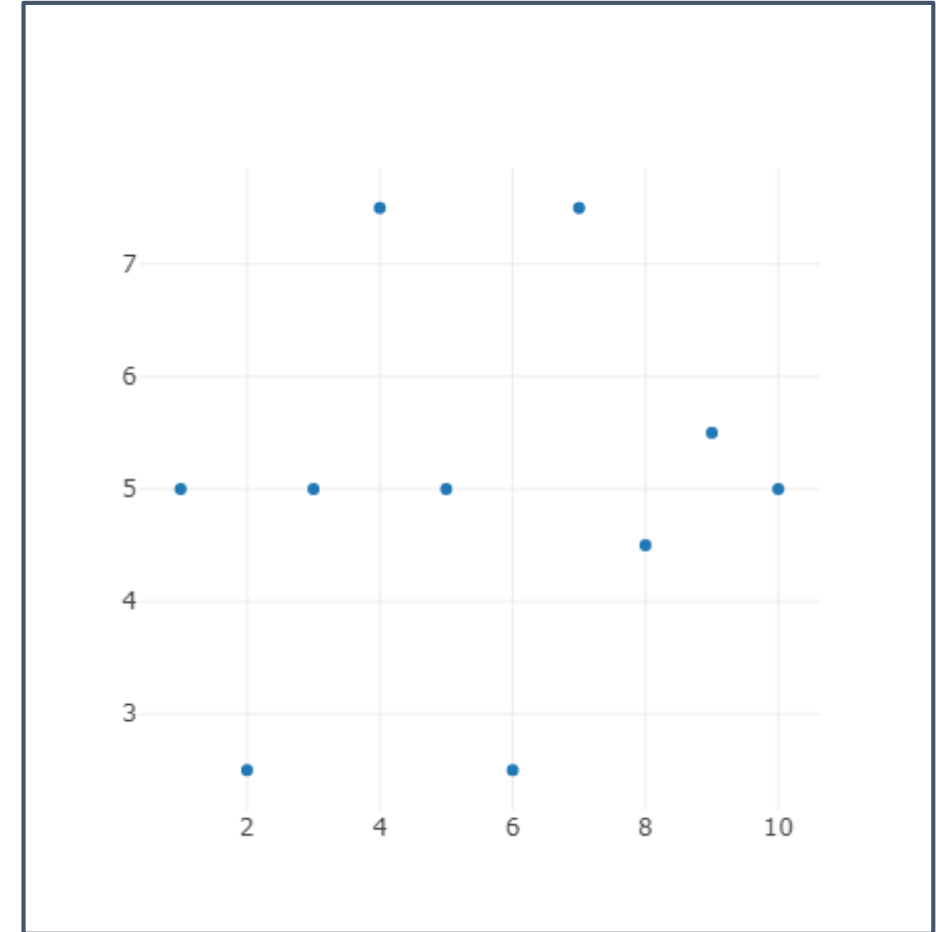


# Style your charts - Example code

```

Chart.Point(x,y)
|> Chart.withSize(500.,500.)

|> Chart.Show
  
```

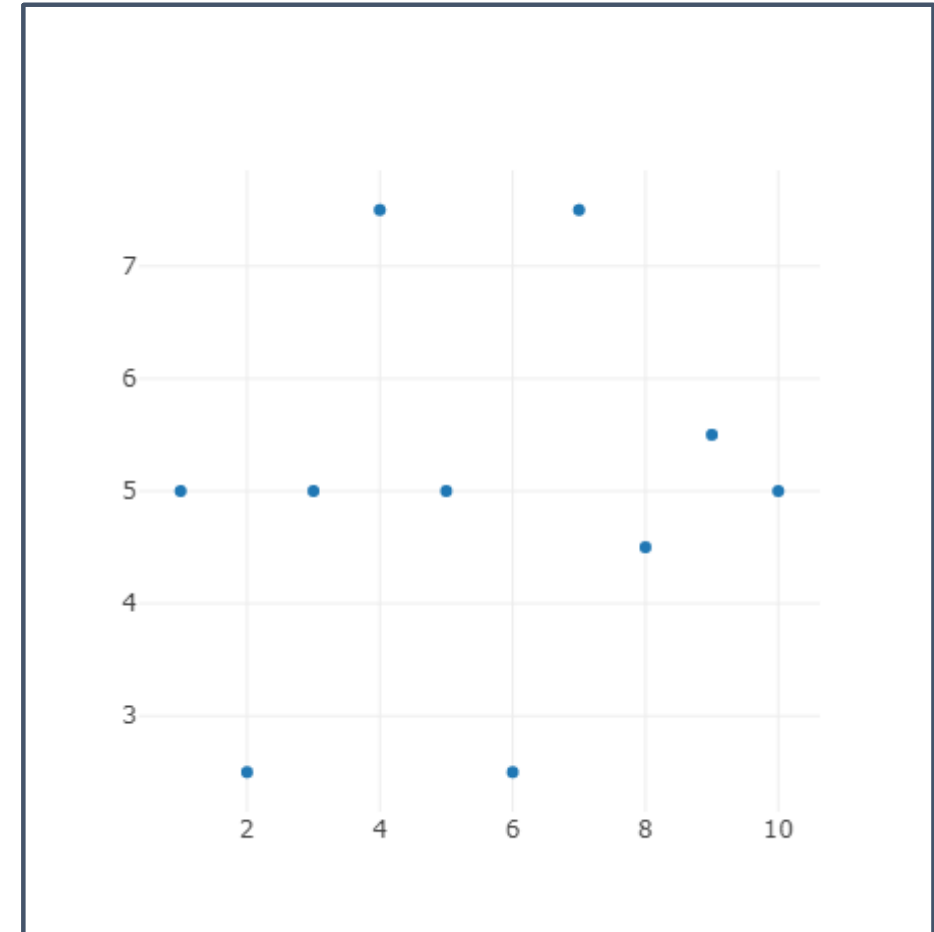


# Style your charts - Example code

```

Chart.Point(x,y)
|> Chart.withSize(500.,500.)
|> Chart.withTitle "ChartTitle"

|> Chart.Show
  
```

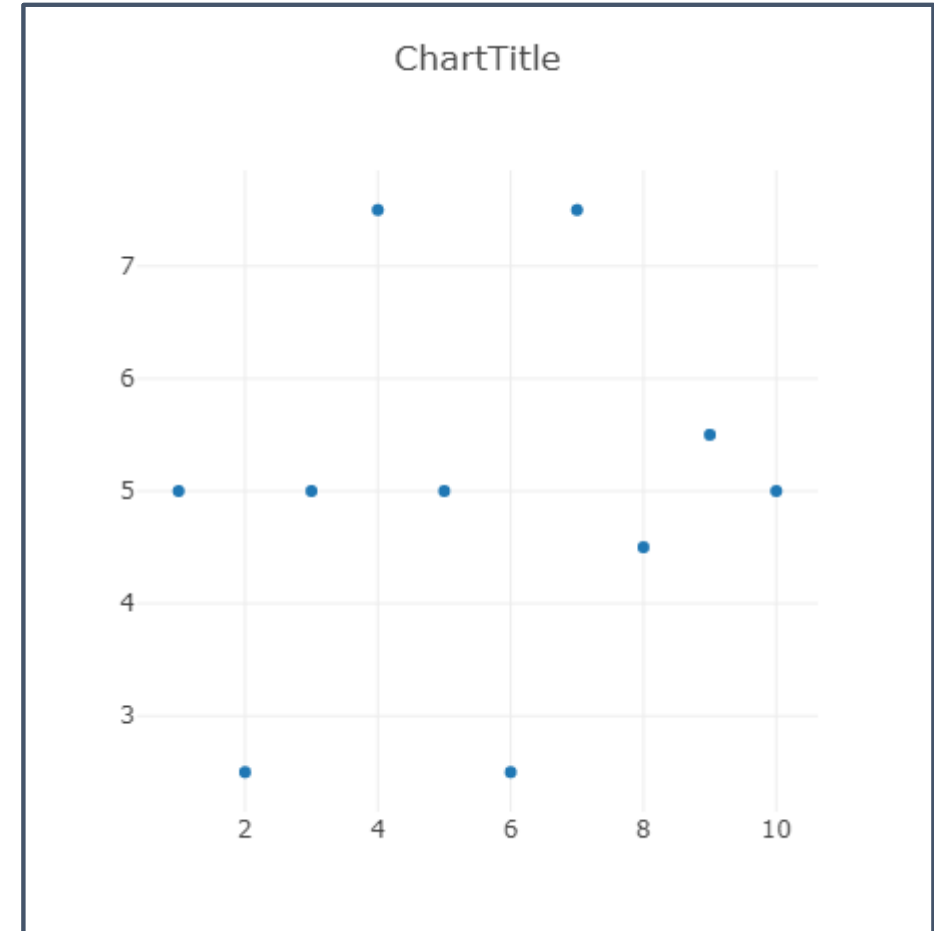




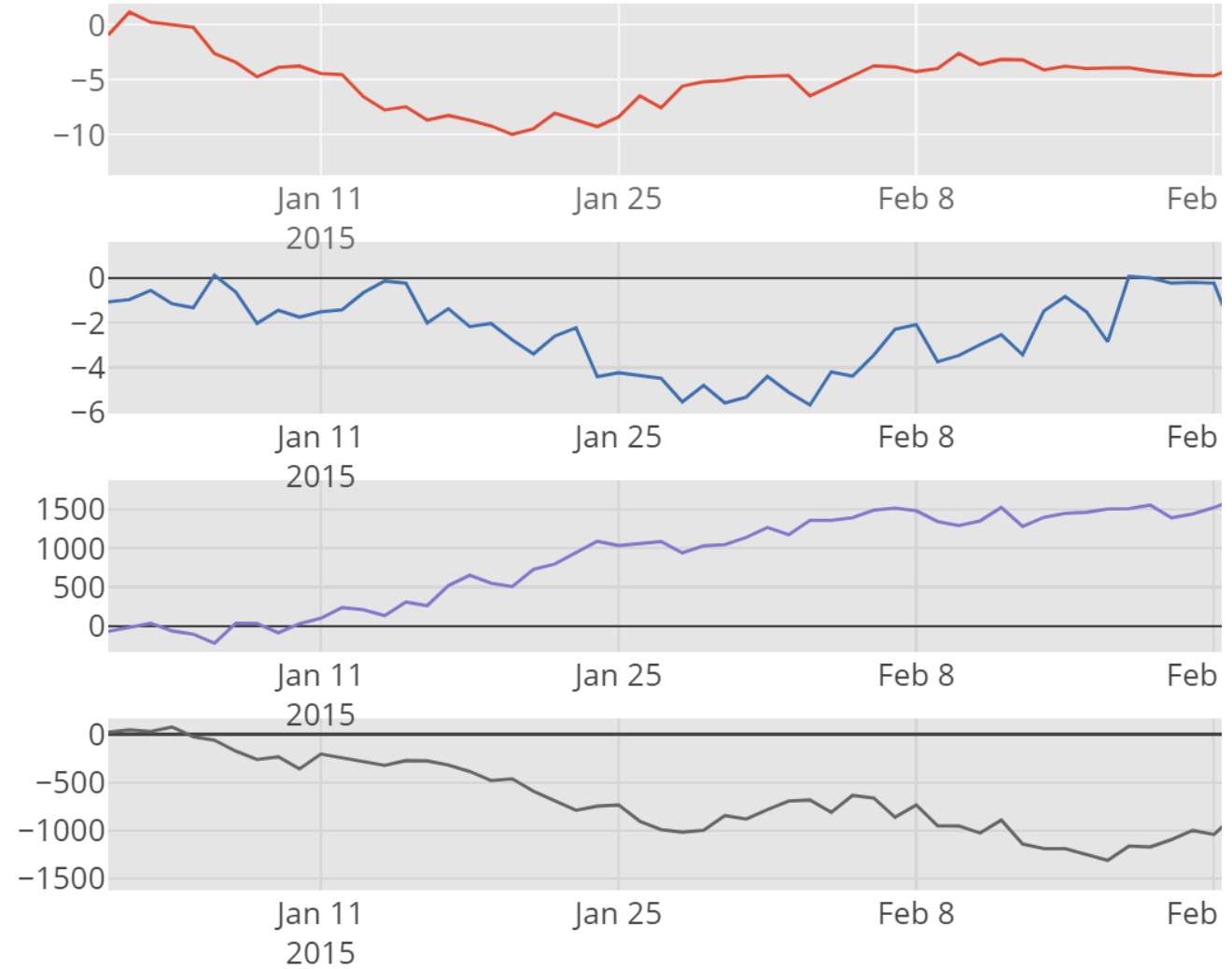
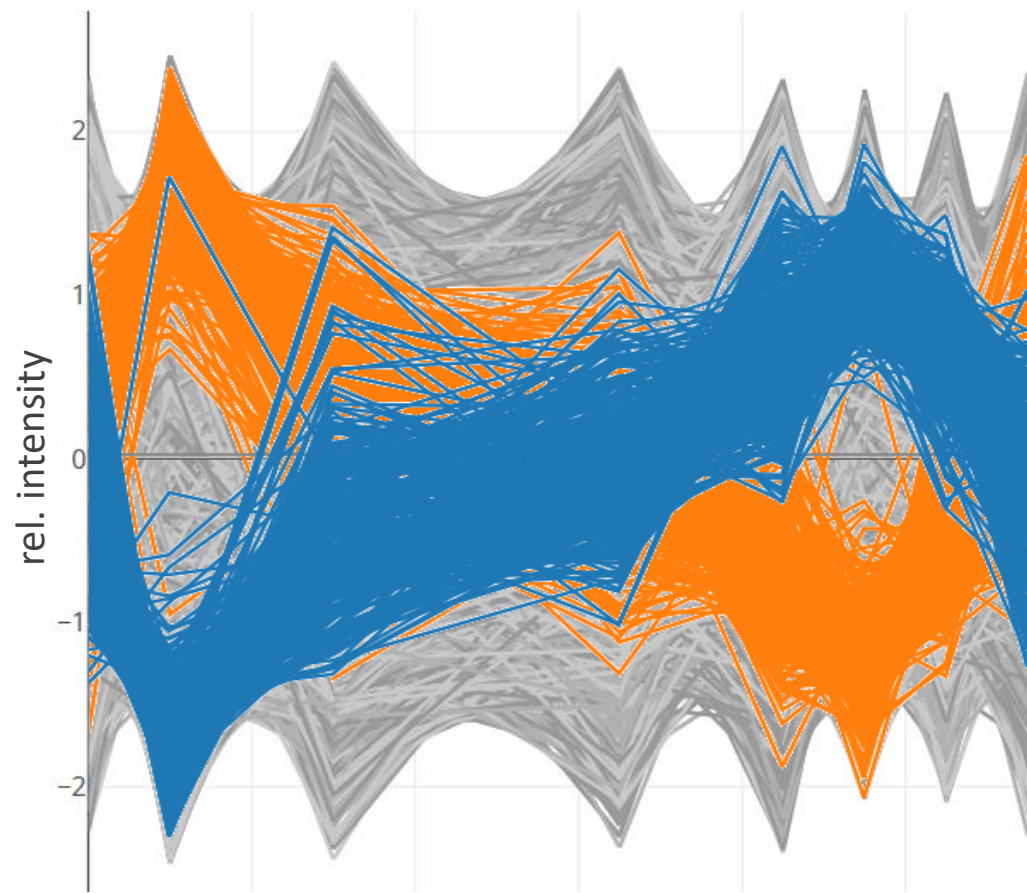
# Style your charts - Example code

```

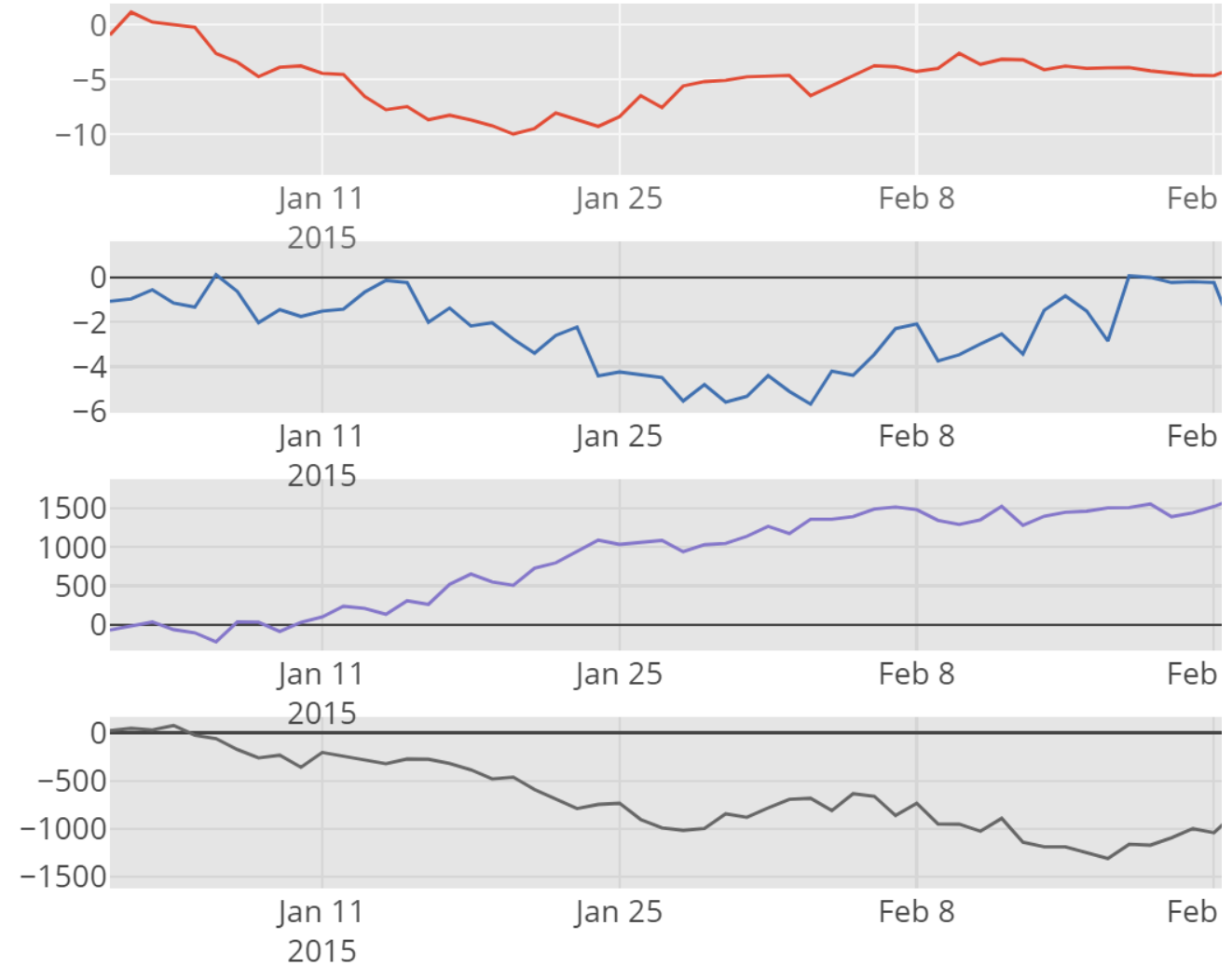
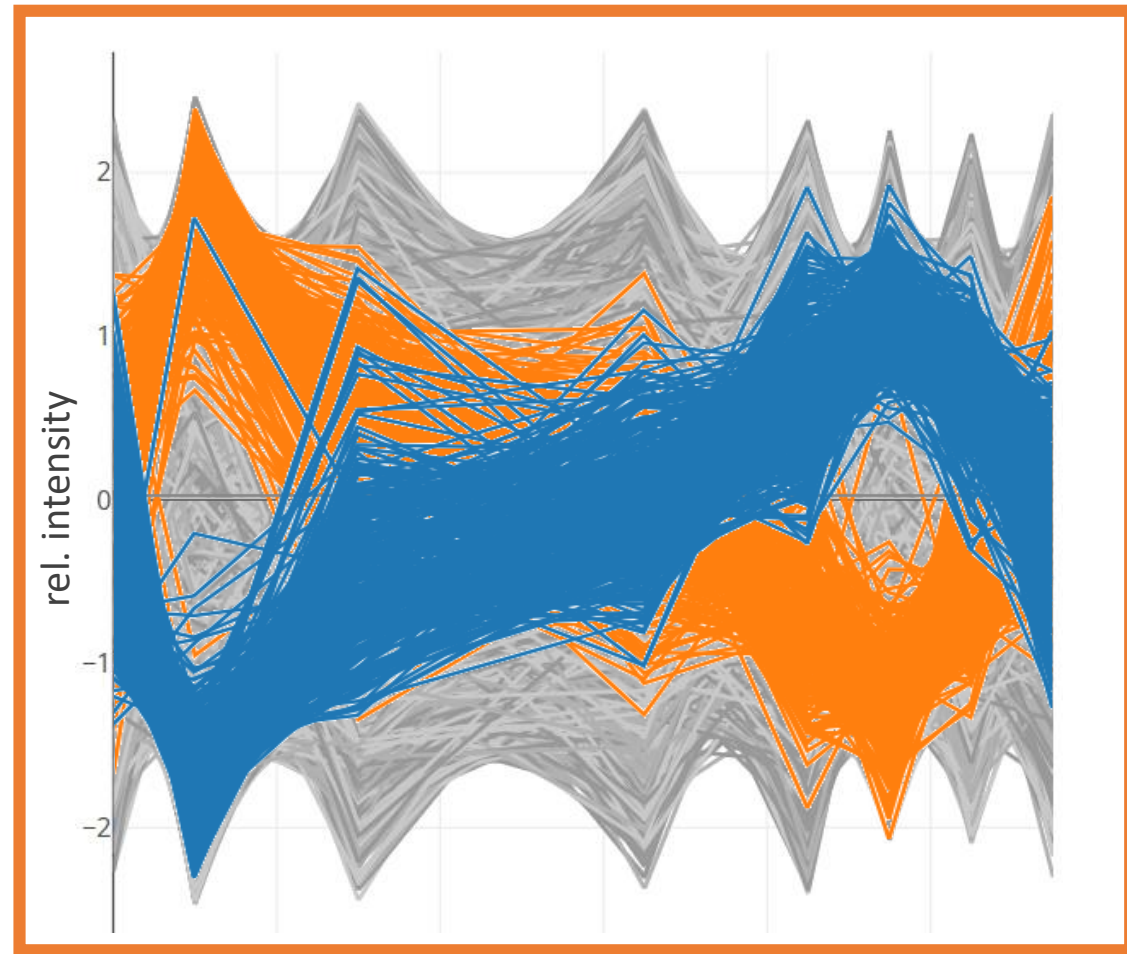
Chart.Point(x,y)
|> Chart.withSize(500.,500.)
|> Chart.withTitle("ChartTitle")
|> Chart.withX_AxisStyle("XAxisTitle",Showgrid=false,Showline=true)
|> Chart.withY_AxisStyle("YAxisTitle",Showgrid=false,Showline=true)
|> Chart.Show
  
```



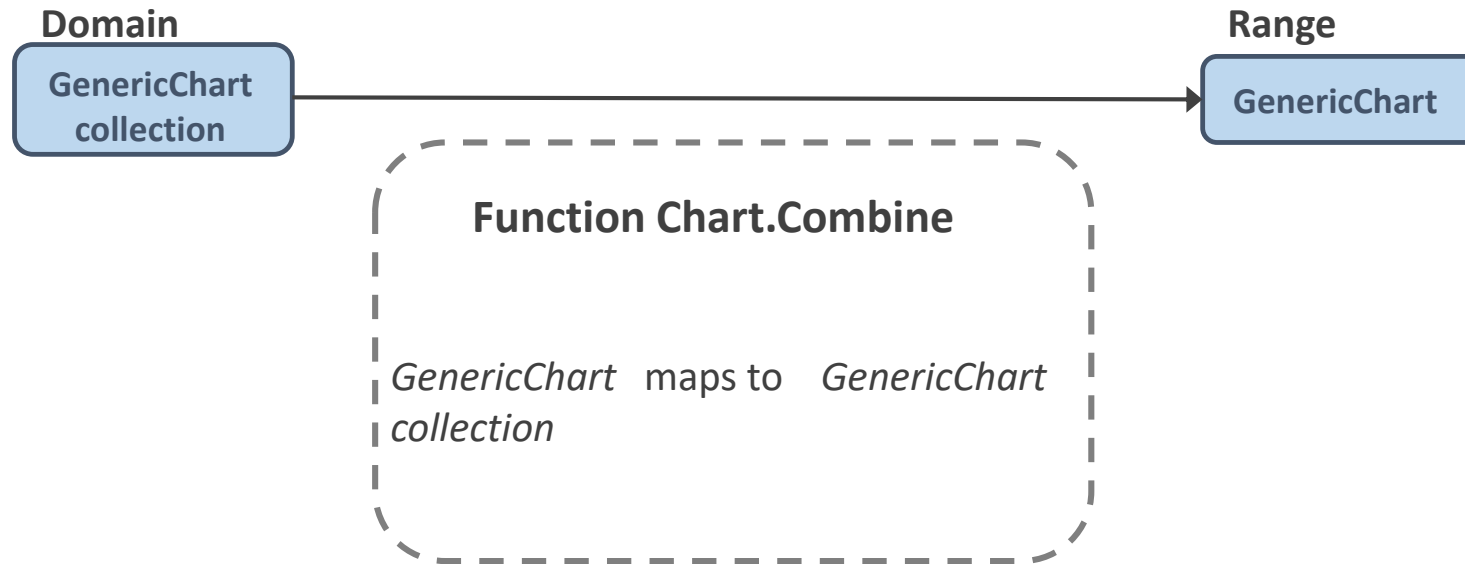
# Multicharts



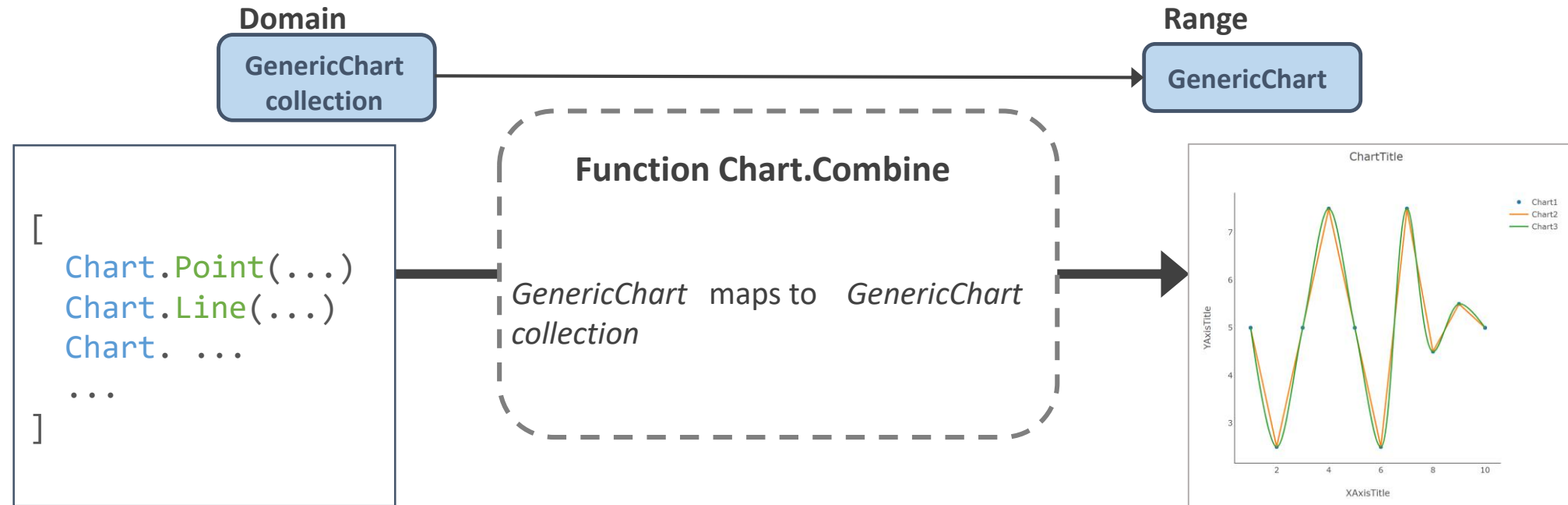
# Multicharts – Combine Charts



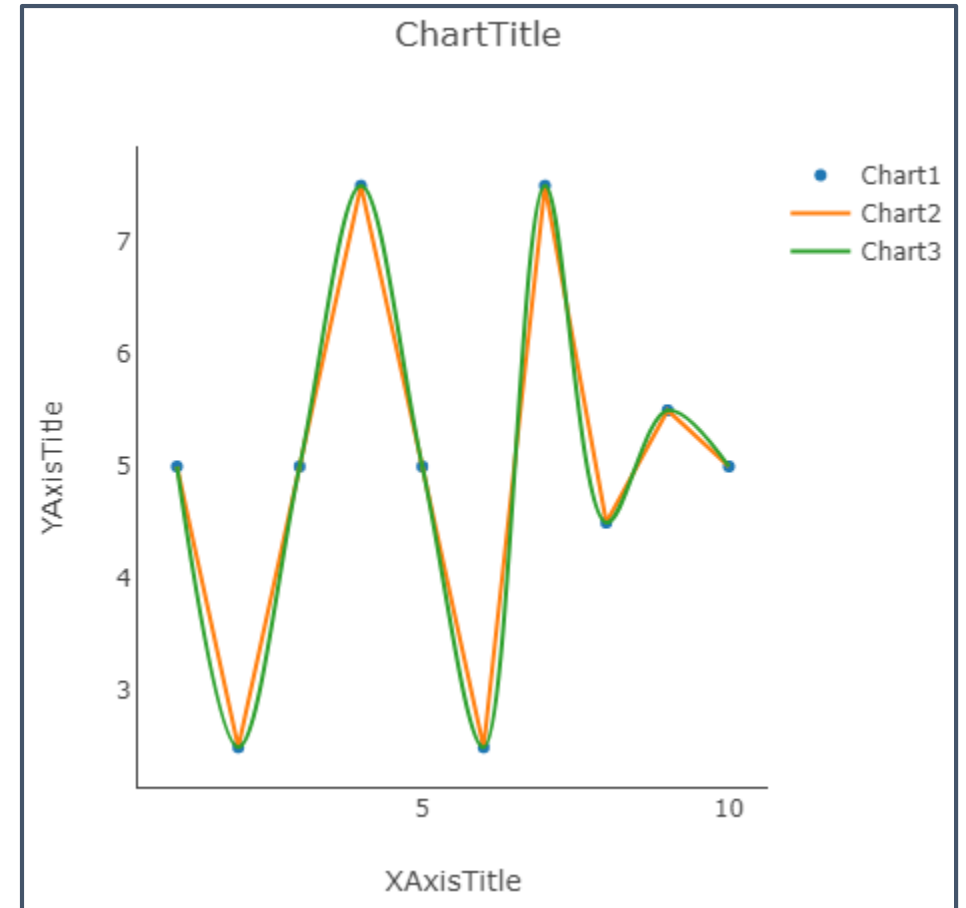
# Combine Charts – Chart.Combine



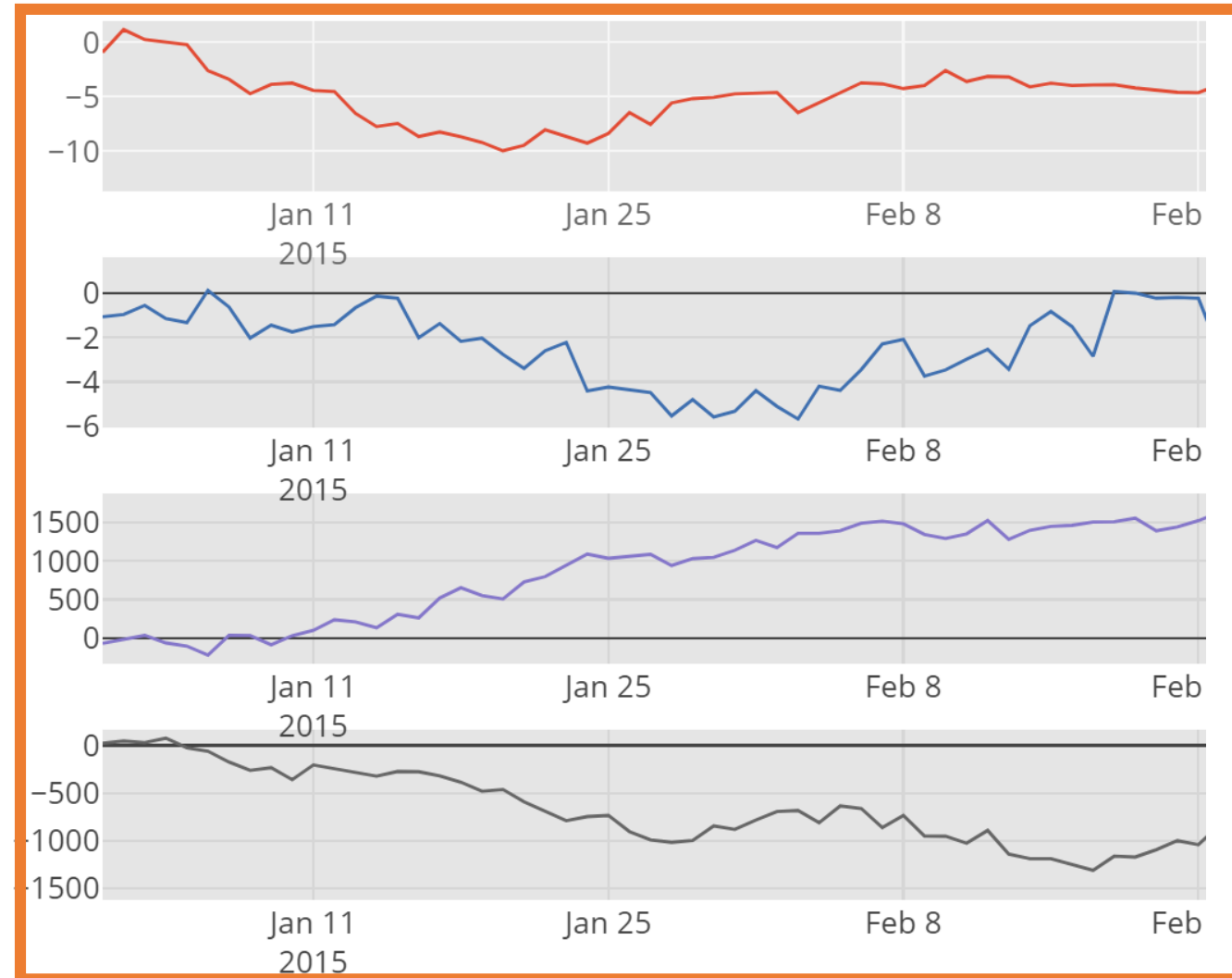
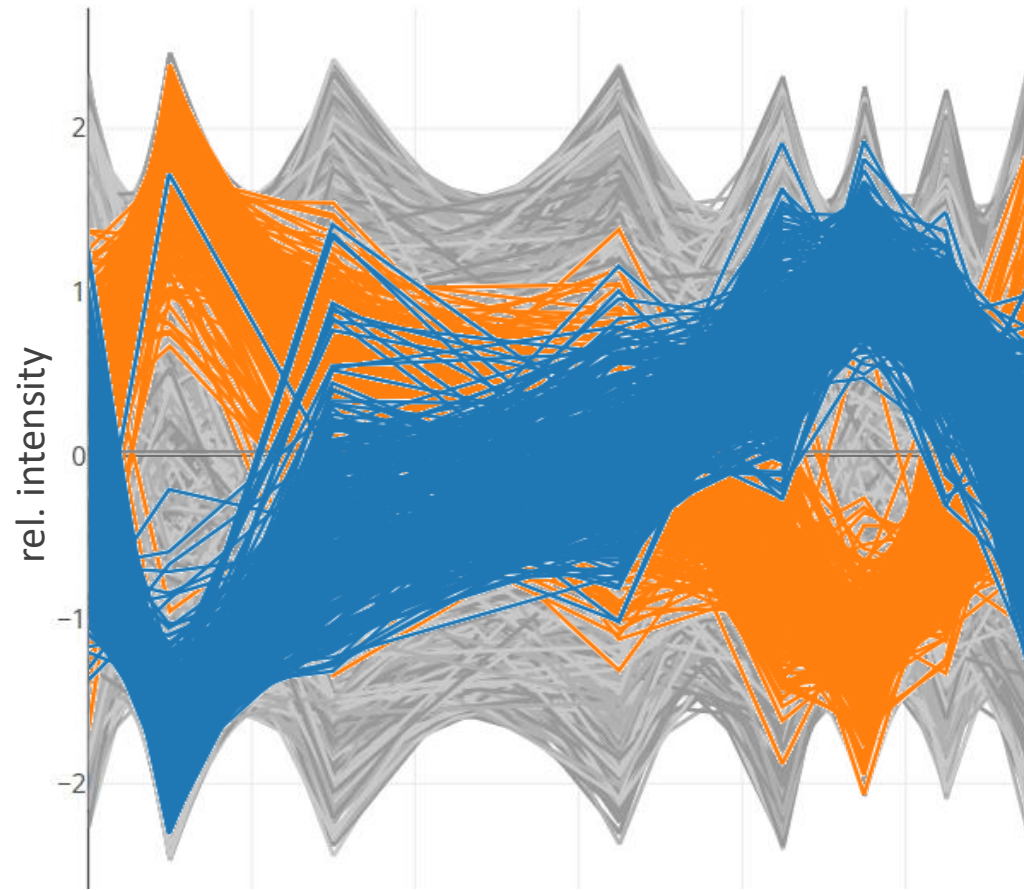
# Combine Charts – Chart.Combine



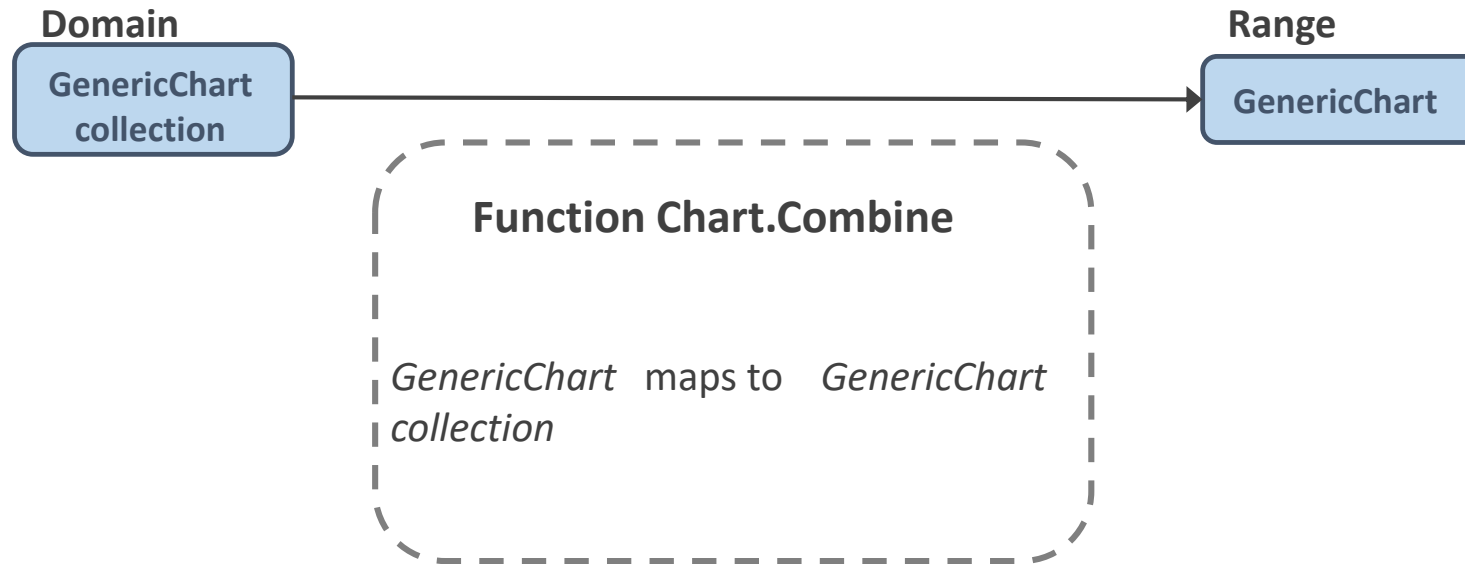
```
[
...Chart.Point(x,y,"Chart1")
...Chart.Line(x,y,"Chart2")
...Chart.Spline(x,y,"Chart3")
]>Chart.Combine
```



# Multicharts – Stack Charts

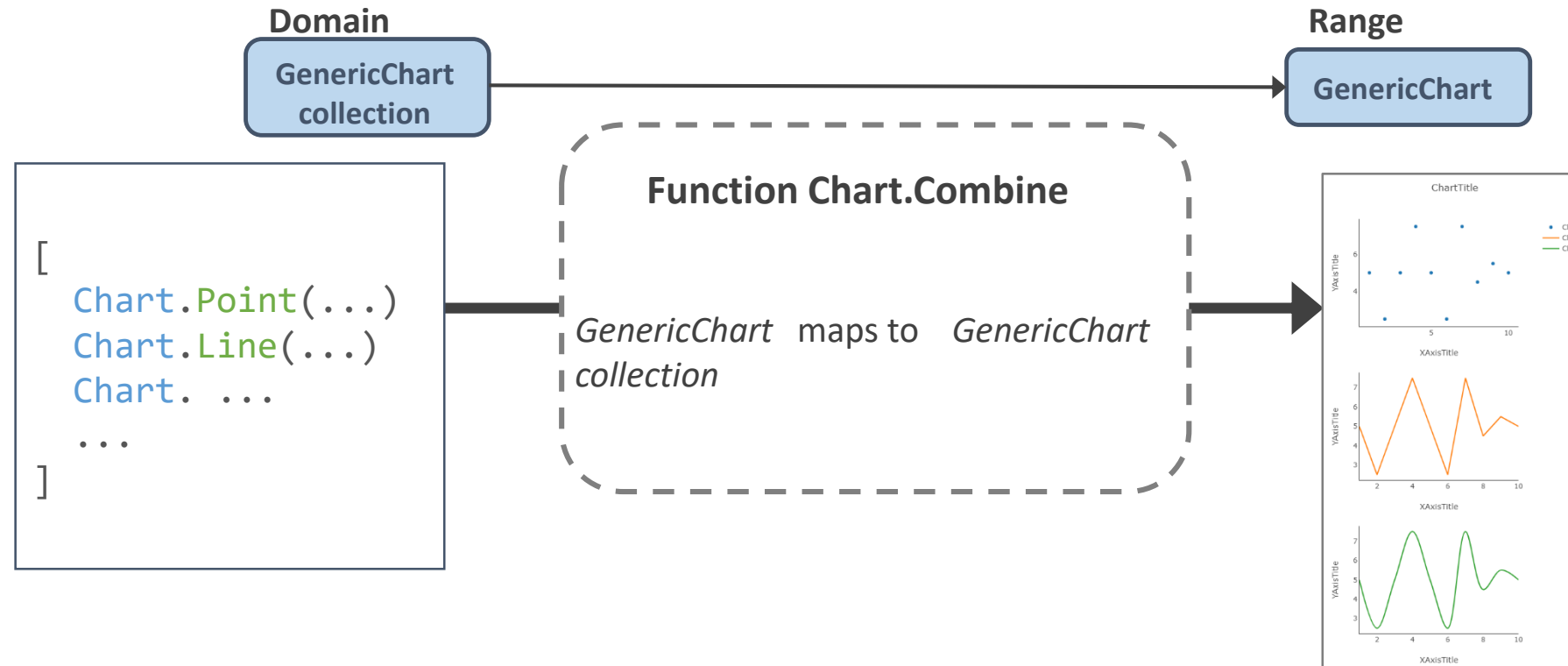


# Combine Charts – Chart.Stack





# Combine Charts – Chart.Stack



# Combine Charts – Chart.Stack

```
[  
  ...Chart.Point(x,y,"Chart1")  
  ...Chart.Line(x,y,"Chart2")  
  ...Chart.Spline(x,y,"Chart3")  
]  
|> .List.map( fun chart -> chart |> .Chart.withX_AxisStyle("XAxisTitle", Showgrid=false, Showline=true))  
|> .List.map( fun chart -> chart |> .Chart.withY_AxisStyle("YAxisTitle", Showgrid=false, Showline=true))
```

# Combine Charts – Chart.Stack

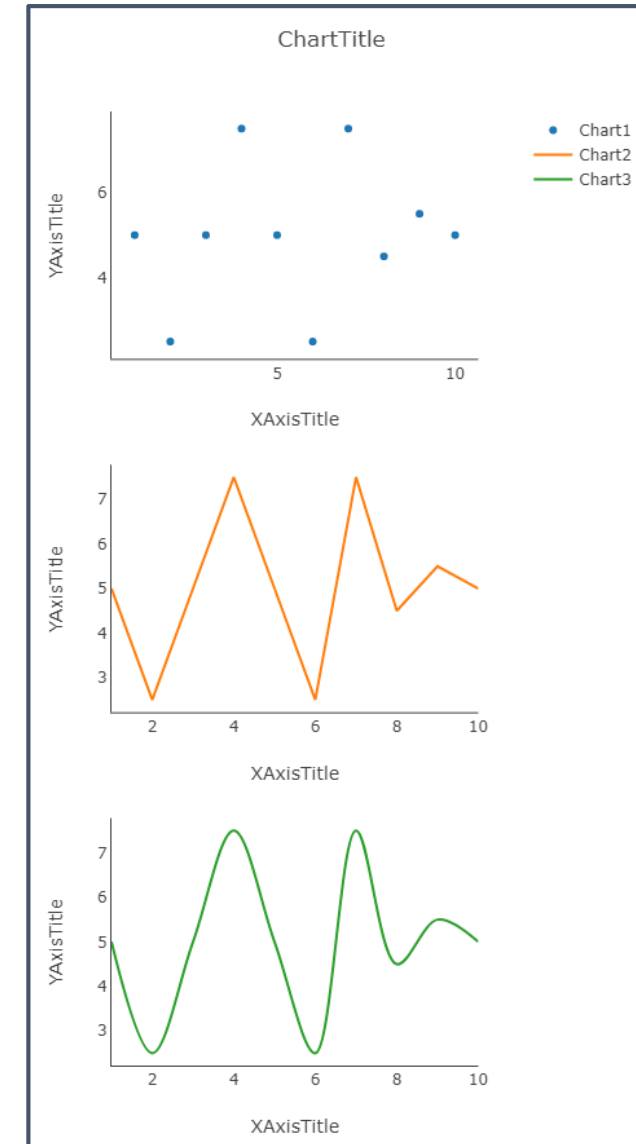
```
[
  ...Chart.Point(x,y,"Chart1")
  ...Chart.Line(x,y,"Chart2")
  ...Chart.Spline(x,y,"Chart3")
]
|>.List.map(fun chart -> chart |>.Chart.withX_AxisStyle("XAxisTitle",Showgrid=false,Showline=true))
|>.List.map(fun chart -> chart |>.Chart.withY_AxisStyle("YAxisTitle",Showgrid=false,Showline=true))
|>.Chart.Stack(1,0.1)
```

Amount of  
columns

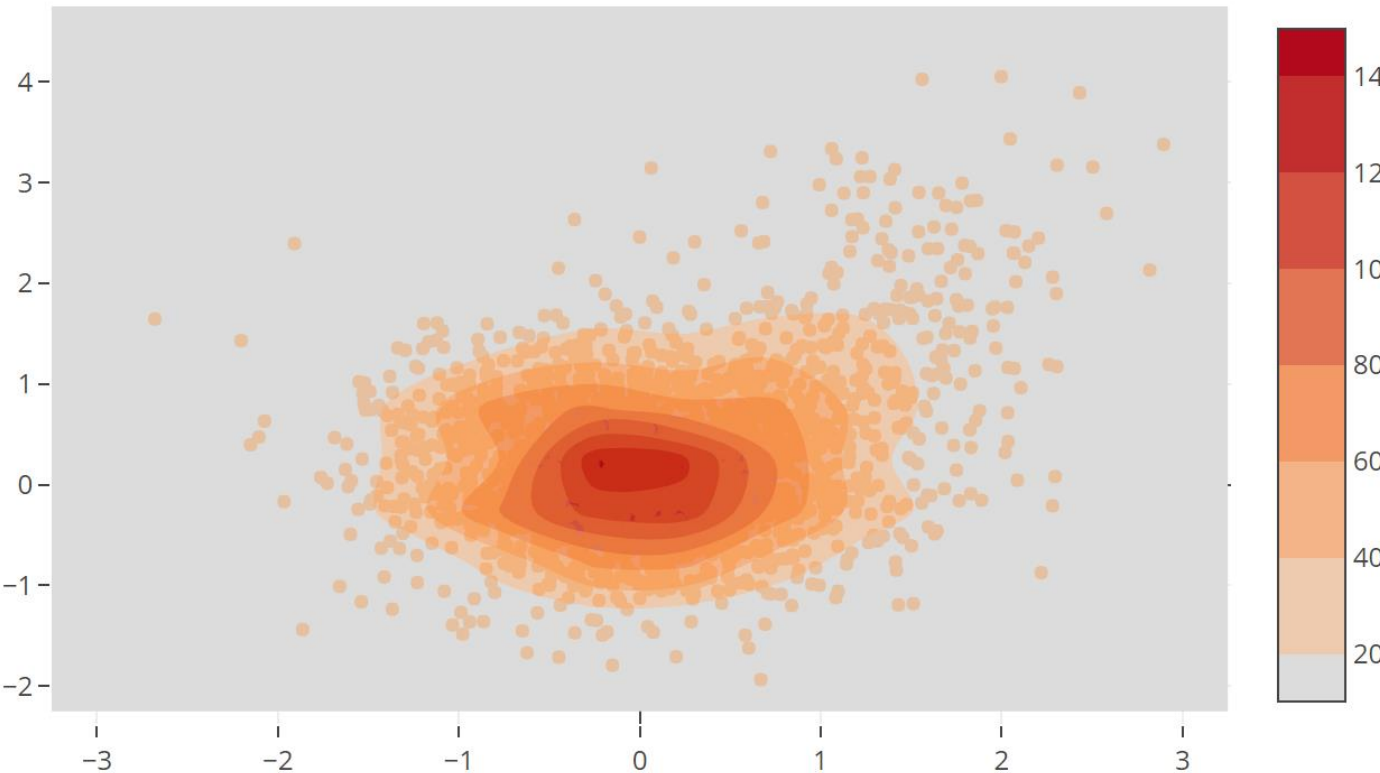
Spacing  
between  
subplots

# Combine Charts – Chart.Stack

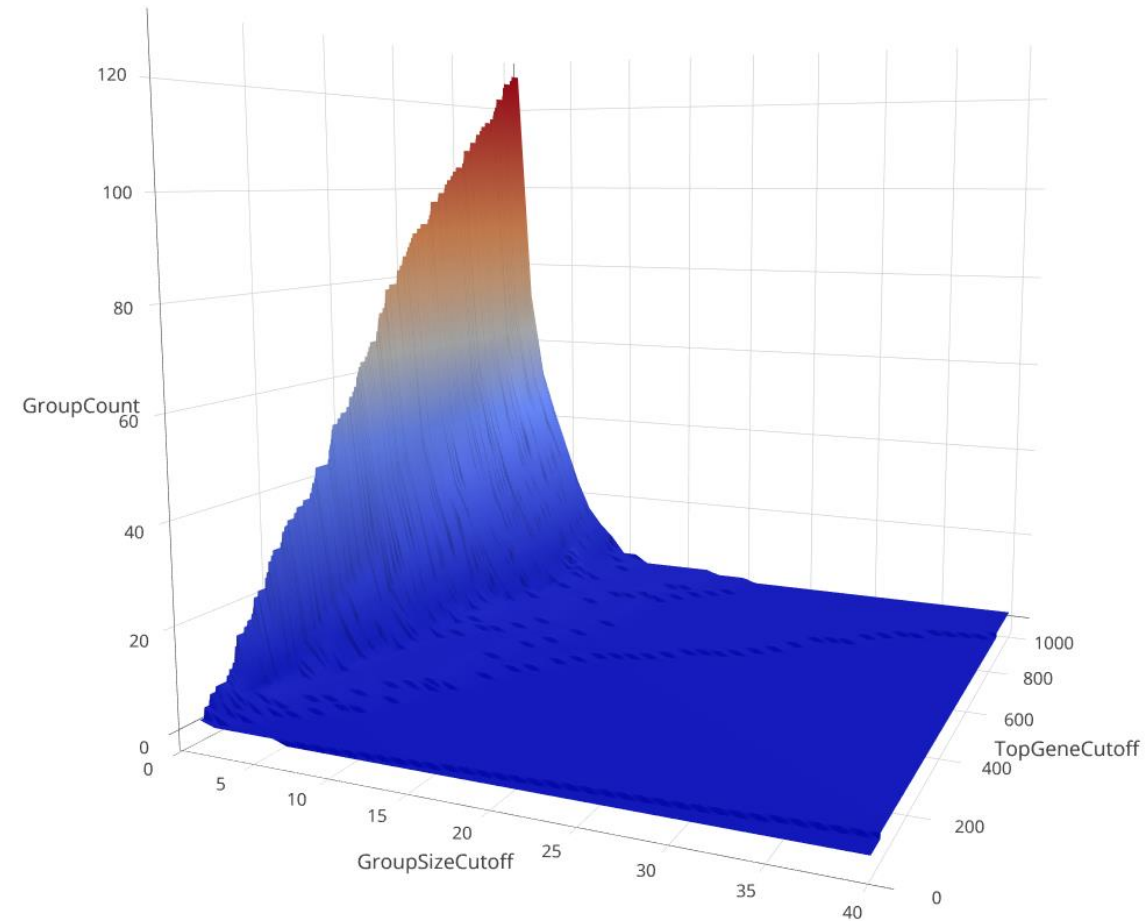
```
[
...Chart.Point(x,y,"Chart1")
...Chart.Line(x,y,"Chart2")
...Chart.Spline(x,y,"Chart3")
]
|>.List.map(fun chart->chart|>.Chart.withX_AxisStyle("XAxisTitle",Showgrid=false,Showline=true))
|>.List.map(fun chart->chart|>.Chart.withY_AxisStyle("YAxisTitle",Showgrid=false,Showline=true))
|>.Chart.Stack(1,0.1)
|>.Chart.withSize(500.,1000.)
|>.Chart.withTitle("ChartTitle")
|>.Chart.Show
```



2D Histogram + Scatter:



3D Surface:



# Thank you for your attention

