

# Table of Contents

Plot Recipes .....	1
Processing Charts .....	1
Creating Bar Charts .....	4
Creating Pie Charts .....	6
Creating Box Charts .....	7
Creating Curve Charts .....	7
Creating Regression Charts .....	9
Creating Scatter Plot Charts .....	10

# Plot Recipes

The Plot classes are in the [geoscript.plot](#) package.

## Processing Charts

*Show a chart in a GUI*

```
List data = [  
    [1,10],[45,12],[23,3],[5,20]  
]  
Chart chart = Bar.xy(data)
```



*Get an image from a chart*

```
Map data = [  
    "A":20,"B":45,"C":2,"D":14  
]  
Chart chart = Pie.pie(data)  
BufferedImage image = chart.image
```



*Save a chart to a file*

```
Map data = [  
    "A": [1, 10, 20],  
    "B": [45, 39, 10],  
    "C": [40, 30, 20],  
    "D": [14, 25, 19]  
]  
Chart chart = Box.box(data)  
File file = new File("chart.png")  
chart.save(file)
```



*Overlay multiple charts*

```
List data = [  
    [1,10],[45,12],[23,3],[5,20]  
]  
Chart chart1 = Bar.xy(data)  
Chart chart2 = Curve.curve(data)  
Chart chart3 = Regression.linear(data)  
chart1.overlay([chart2,chart3])
```



## Creating Bar Charts

Create a basic bar chart

```
List data = [
  [1,10],[45,12],[23,3],[5,20]
]
Chart chart = Bar.xy(data)
```



Create a bar chart with categories

```
Map data = [
  "A":20,"B":45,"C":2,"D":14
]
Chart chart = Bar.category(data)
```



Create a stacked bar chart with two series of data

```
Map data = [
  "A": ["B":50, "C":25, "D":25],
  "F": ["G":75, "H":10, "I":15]
]
Chart chart = Bar.category(data, stacked: true)
```



Create a 3D bar chart with categories

```
Map data = [
  "A":20, "B":45, "C":2, "D":14
]
Chart chart = Bar.category(data, trid: true)
```



## Creating Pie Charts

*Create a pie chart*

```
Map data = [  
  "A":20,"B":45,"C":2,"D":14  
]  
Chart chart = Pie.pie(data)
```



*Create a 3D pie chart*

```
Map data = [  
  "A":20,"B":45,"C":2,"D":14  
]  
Chart chart = Pie.pie(data, trid: true)
```



## Creating Box Charts

Create a box chart

```
Map data = [
  "A": [1, 10, 20],
  "B": [45, 39, 10],
  "C": [40, 30, 20],
  "D": [14, 25, 19]
]
Chart chart = Box.box(data)
```



## Creating Curve Charts



### Create a curve chart

```
List data = [  
    [1,10],[45,12],[23,3],[5,20]  
]  
Chart chart = Curve.curve(data)
```



### Create a smooth curve chart

```
List data = [  
    [1,10],[45,12],[23,3],[5,20]  
]  
Chart chart = Curve.curve(data, smooth: true)
```



### Create a 3D curve chart

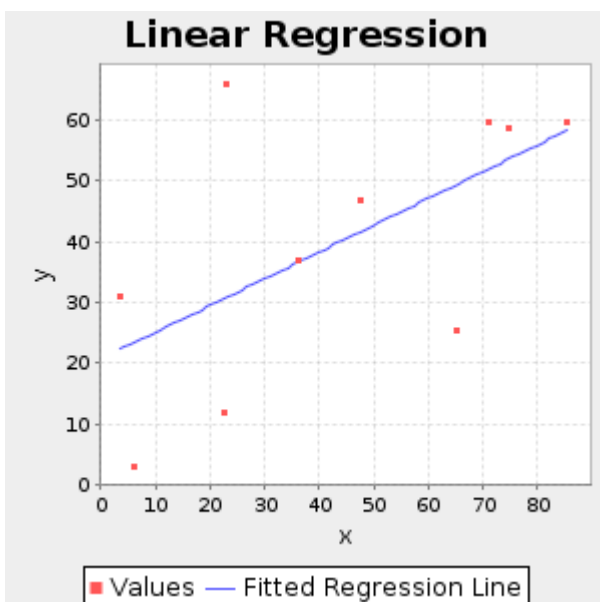
```
List data = [  
    [1,10],[45,12],[23,3],[5,20]  
]  
Chart chart = Curve.curve(data, trid: true)
```



## Creating Regression Charts

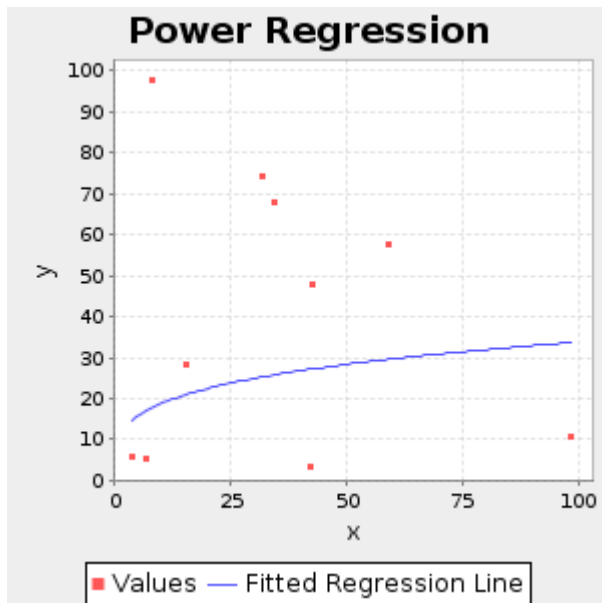
### Create a linear regression chart

```
MultiPoint multPoint = Geometry.createRandomPoints(new Bounds(0,0,100,100).geometry,  
10)  
List data = multPoint.geometries.collect{ Point pt ->  
    [pt.x, pt.y]  
}  
Chart chart = Regression.linear(data)
```



### Create a power regression chart

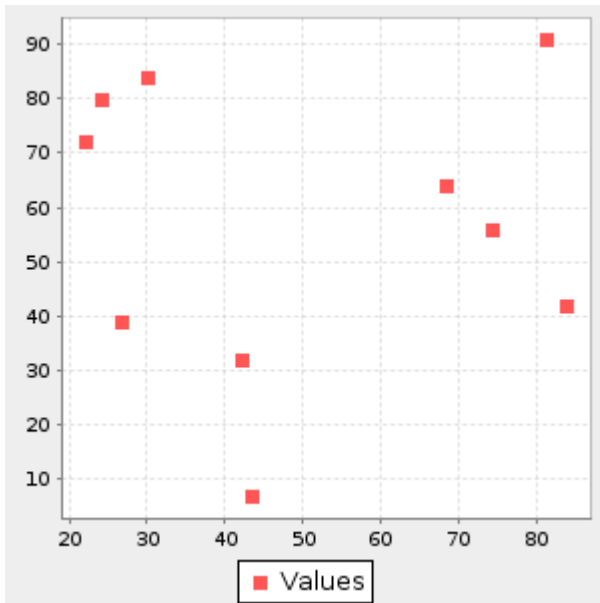
```
MultiPoint multPoint = Geometry.createRandomPoints(new Bounds(0,0,100,100).geometry,
10)
List data = multPoint.geometries.collect{ Point pt ->
    [pt.x, pt.y]
}
Chart chart = Regression.power(data)
```



## Creating Scatter Plot Charts

### Create a scatter plot chart

```
MultiPoint multPoint = Geometry.createRandomPoints(new Bounds(0,0,100,100).geometry,
10)
List data = multPoint.geometries.collect{ Point pt ->
    [pt.x, pt.y]
}
Chart chart = Scatter.scatterplot(data)
```



Create a scatter plot chart with options

```
MultiPoint mulitPoint = Geometry.createRandomPoints(new Bounds(0,0,100,100).geometry,
10)
List data = mulitPoint.geometries.collect{ Point pt ->
    [pt.x, pt.y]
}
Chart chart = Scatter.scatterplot(data, legend: false, xLabel: "X Coordinates",
yLabel: "Y Coordinates")
```

