



Interactive Data Visualization and Exploration

Using the Loon R package

Adrian Waddell

PhUSE 2016, Barcelona



UNIVERSITY OF
WATERLOO

Motivation for new interactive visualization tools

Carefully designed, general, and extendable framework

- simple plots
- attention to high-dimensional data
- extendable
- study and compare methodologies visually
 - analysis, research and teaching
- integrated in popular statistical environment such as R

Introduction of Loon with the Gapminder Data

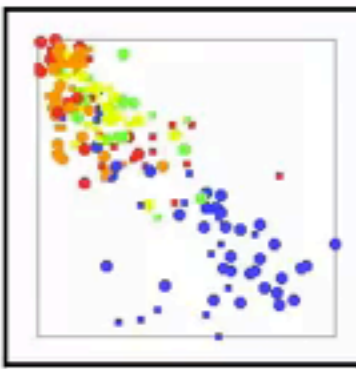
- Western World
 - Long Life & Small Family
- Third World
 - Short Life & Large Family



data from gapminder.com, for year 2009

Loon Inspector

World View



Analysis Layers Glyphs

Plot

axes: swap labels
 scales guides

glyphs: pointlabels

linking group: world [3 linked]

scale to: selected plot world

Select

static: all none invert

dynamic: select deselect invert

by: sweeping brushing

by color:

■ ■ ■ ■ ■

Modify

■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■

■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■

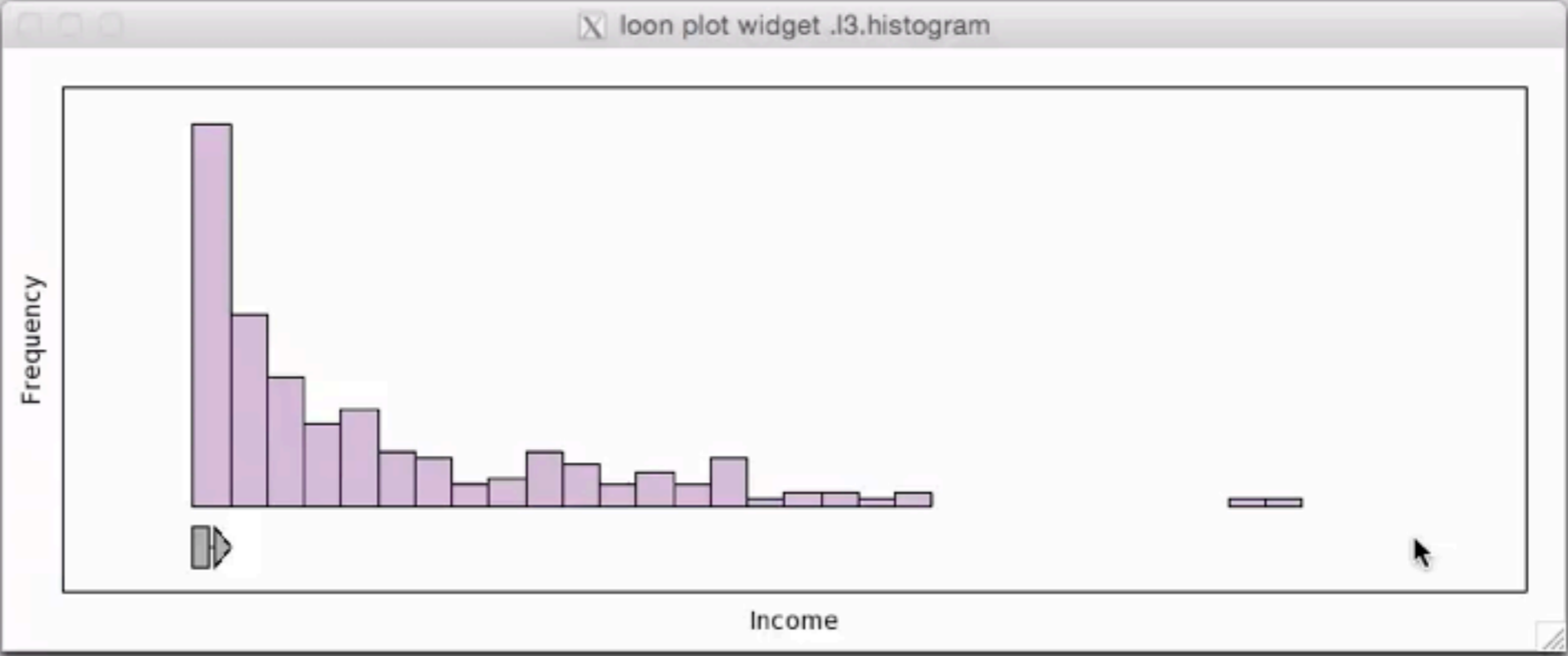
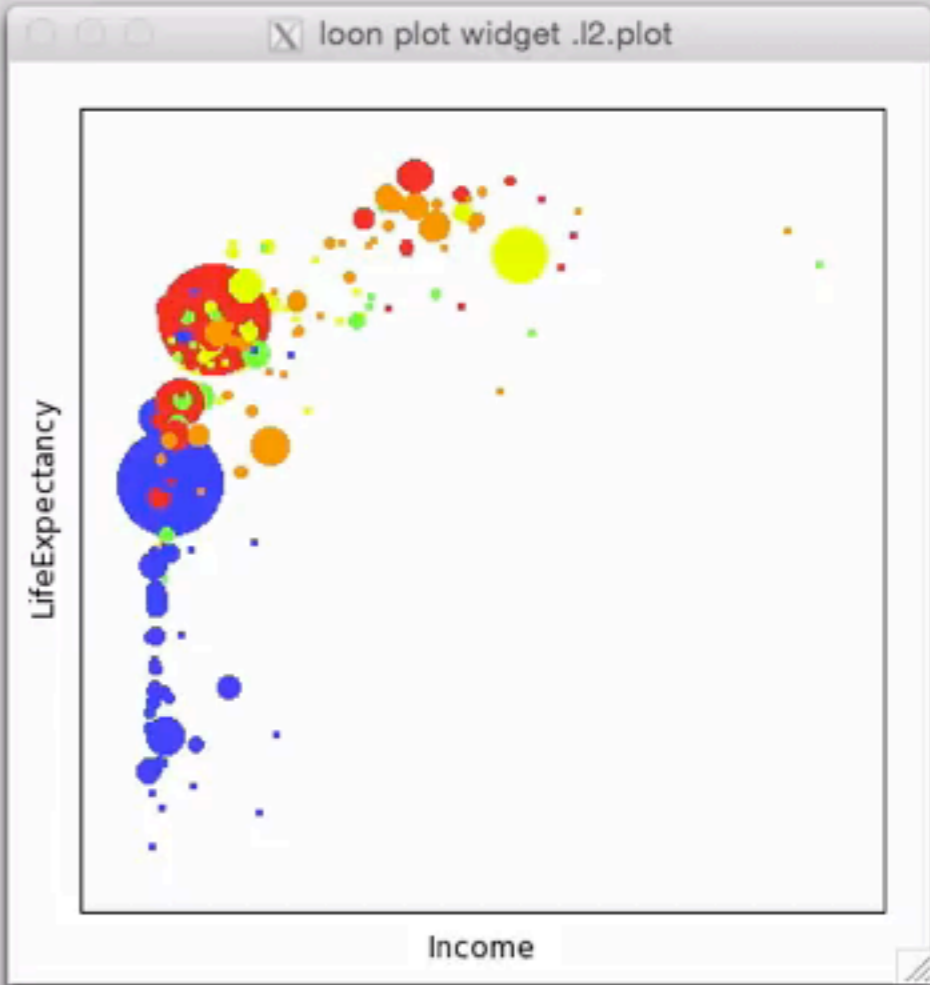
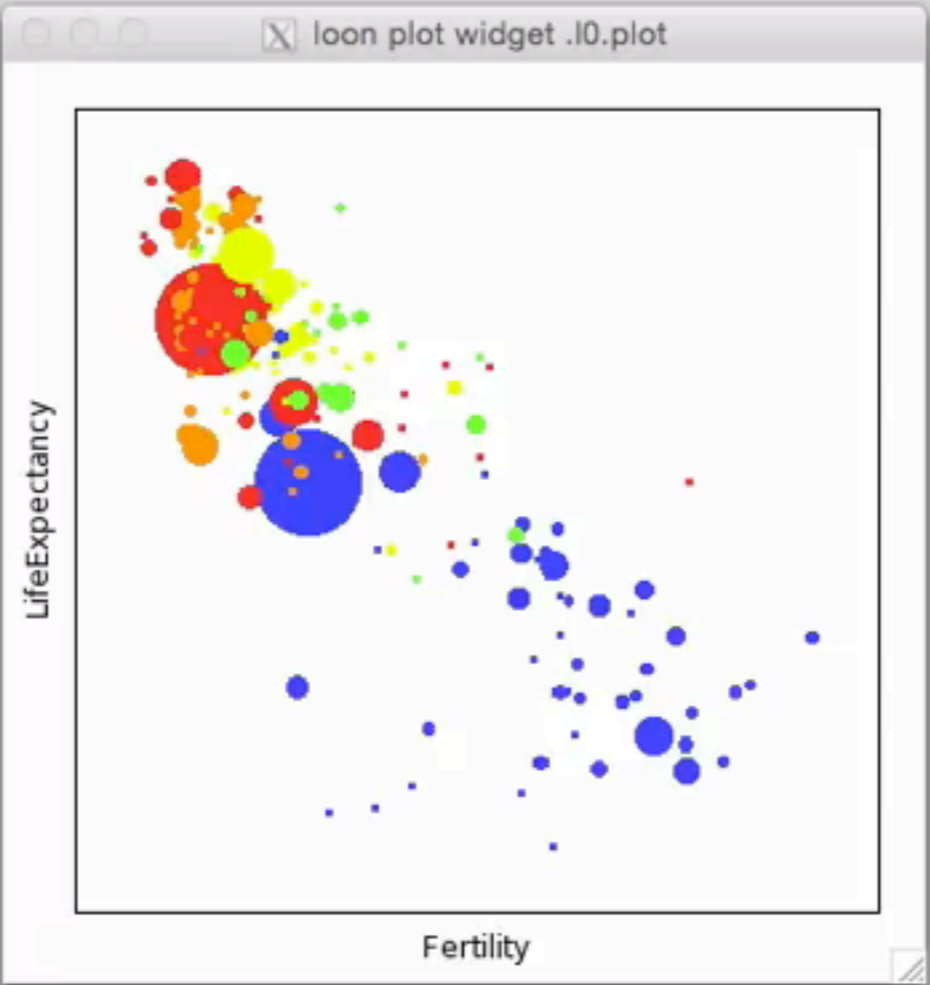
deactivate reactivate

move: ✖ ✚ ≡ ||| :: ⇄ ↻

glyph: ● ■ ▲ ◆ ○ □ △ ◇

... — set

size: abs: - + rel: - +

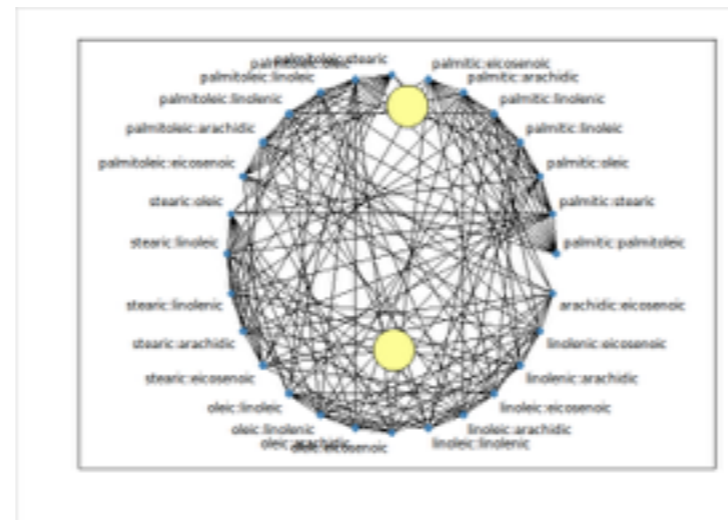
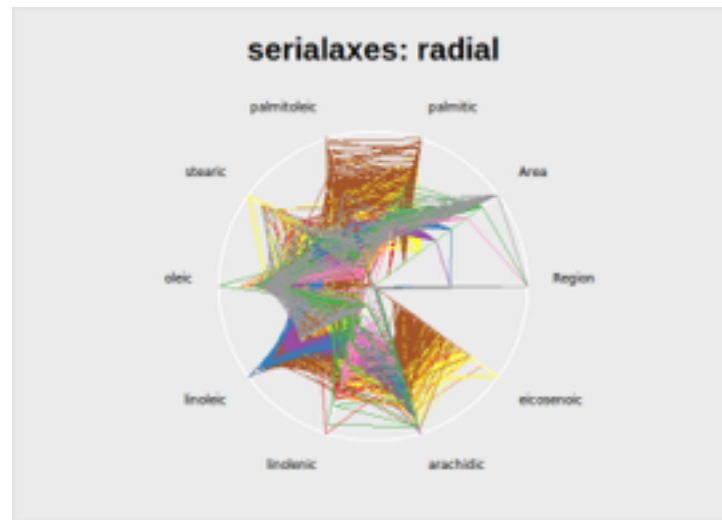
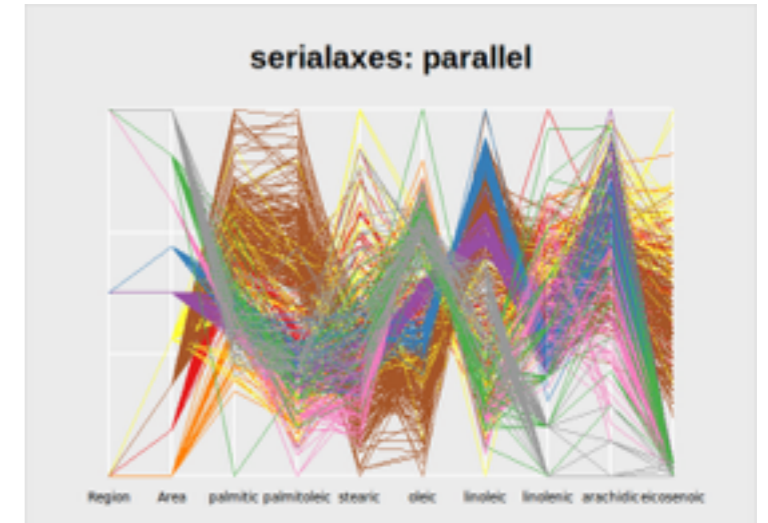
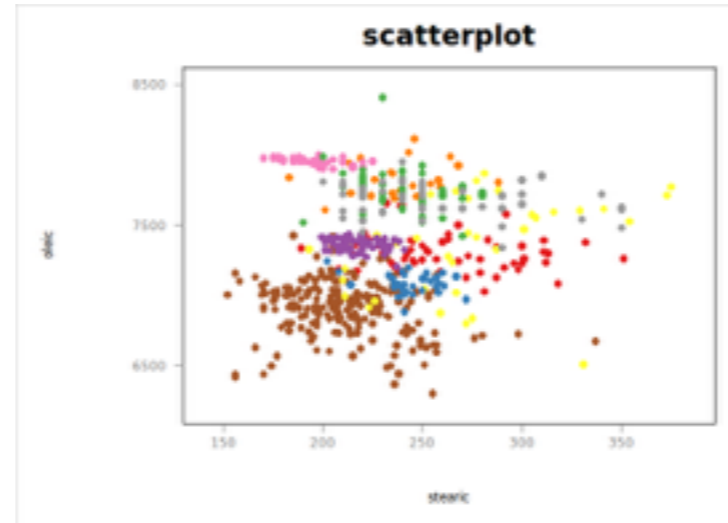
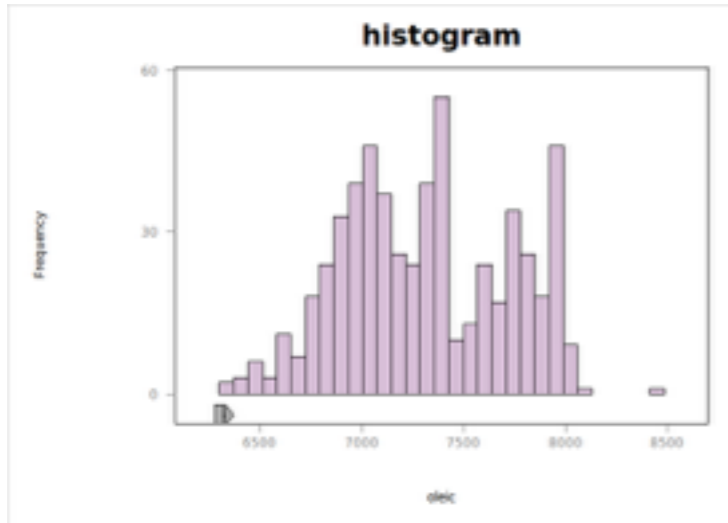


```
p1 <- l_plot(x=Fertility, y=LifeExpectancy,  
            color=region_cols, size=pop_size,  
            itemlabel=country,  
            linkingGroup="world")
```

```
p2 <- l_plot(x=Income, y=LifeExpectancy,  
            itemlabel=country,  
            linkingGroup="world")
```

```
h <- l_hist(x=Income, linkingGroup="world")
```

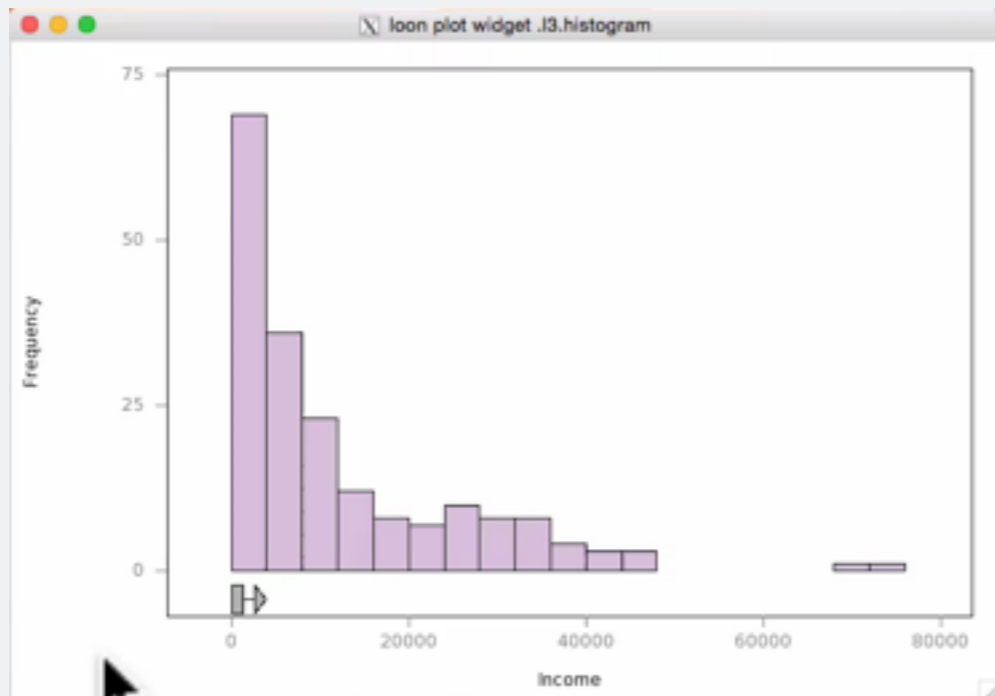
Displays



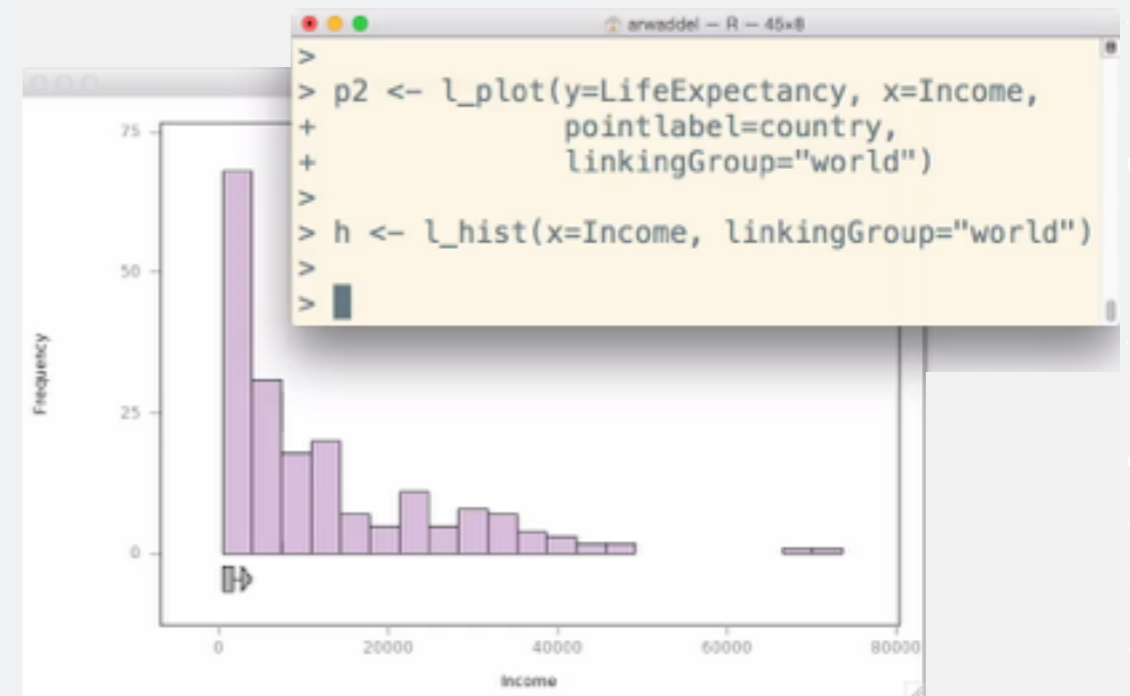
Others in Development

- barplot
- pairs, currently compound view
- ...

Interactive Statistical Visualization Toolkit



Direct Manipulation



Command Line Control

Widgets

```
::tk::button
```

```
::tk::scale
```

```
::loon::plot
```

```
::loon::plot_inspector_analysis
```



Toolkit

```
::oo::class create ::loon::classes::Histogram_Model {
  superclass ::loon::classes::Plot_Model\
  ::loon::classes::withLayers

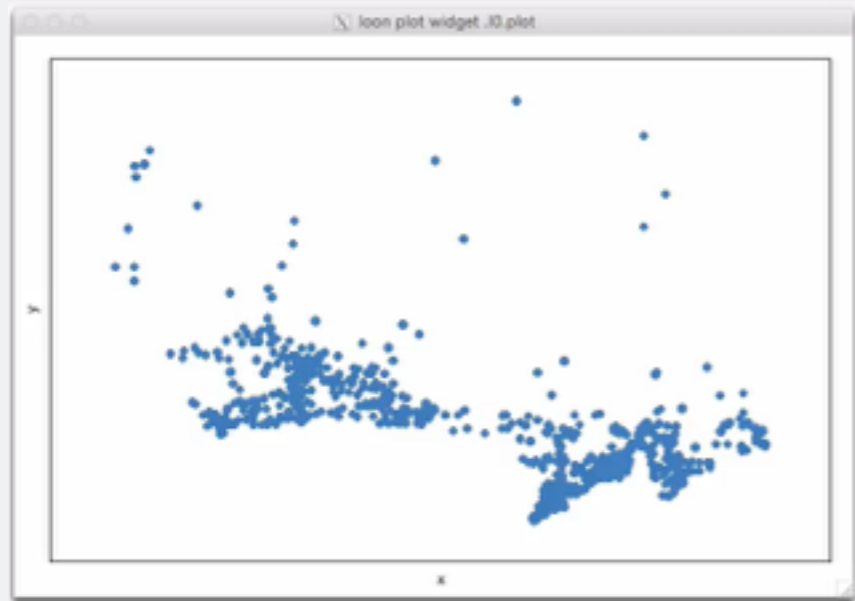
  variable minX minY maxX maxY\
  bins new_bins has_bins\
  colorStackingOrder

  constructor {} {

    set bins [dict create bin [dict create] binid {}]
```

Extendable

Statistical Graphics

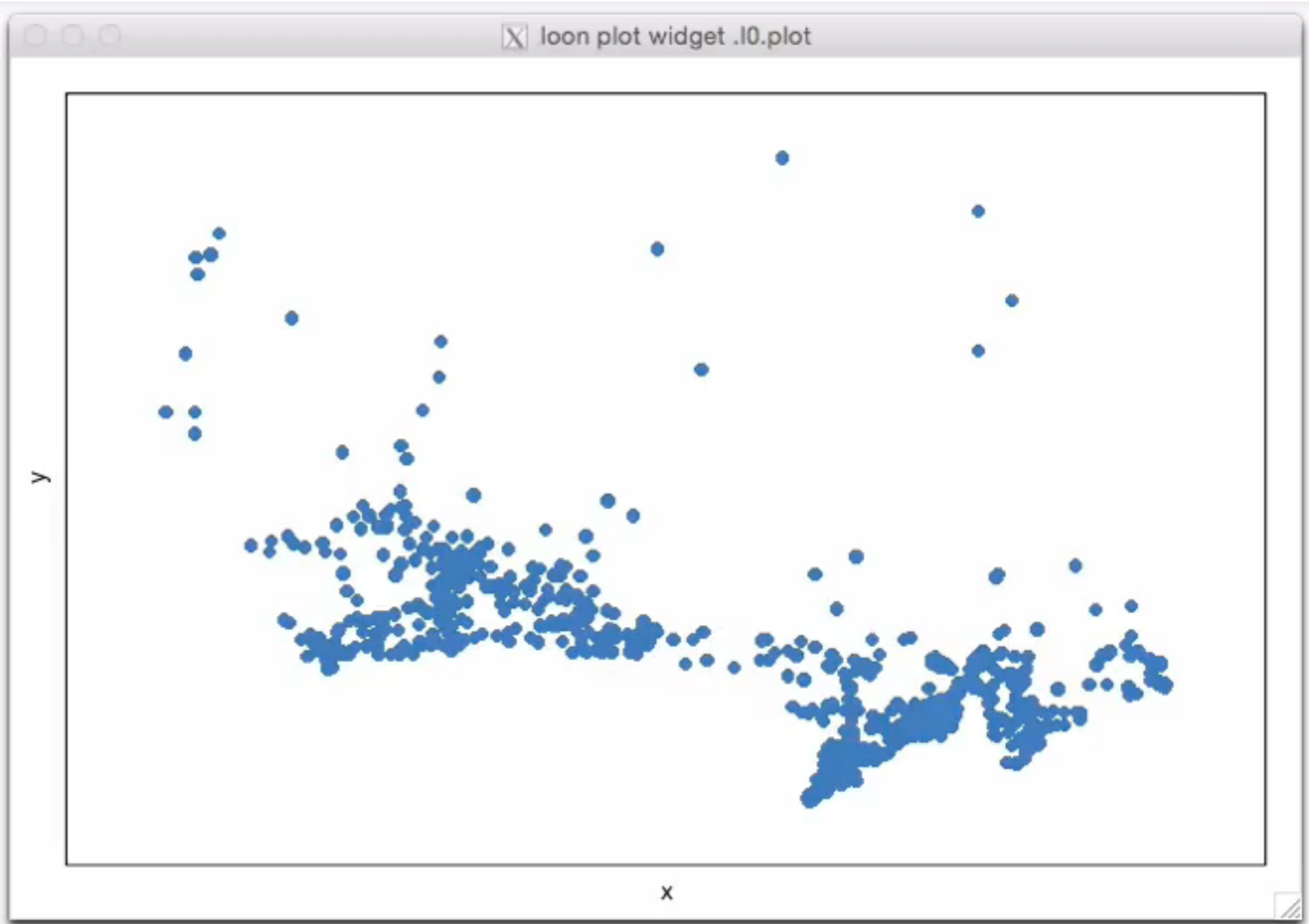


Layers

Point Glyphs

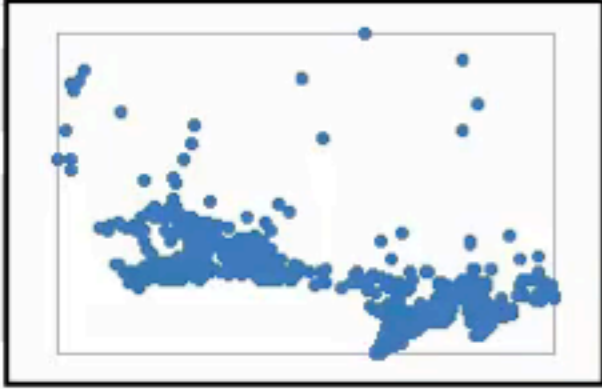
Dynamic Linking

Inspectors



Loon Inspector

World View



Analysis Layers Glyphs

Plot

axes: swap labels
 scales guides

glyphs: pointlabels

linking group: none

scale to: selected plot world

Select

static: all none invert

dynamic: select deselect invert

by: sweeping brushing

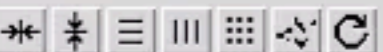
by color: ■

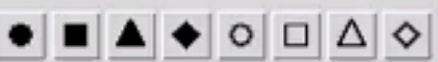
Modify

■ ■ ■ ■ ■ ■ ■ ■ ■ ■ ■

■ ■ ■ ■ ■ ■ ■ ■ ■ +

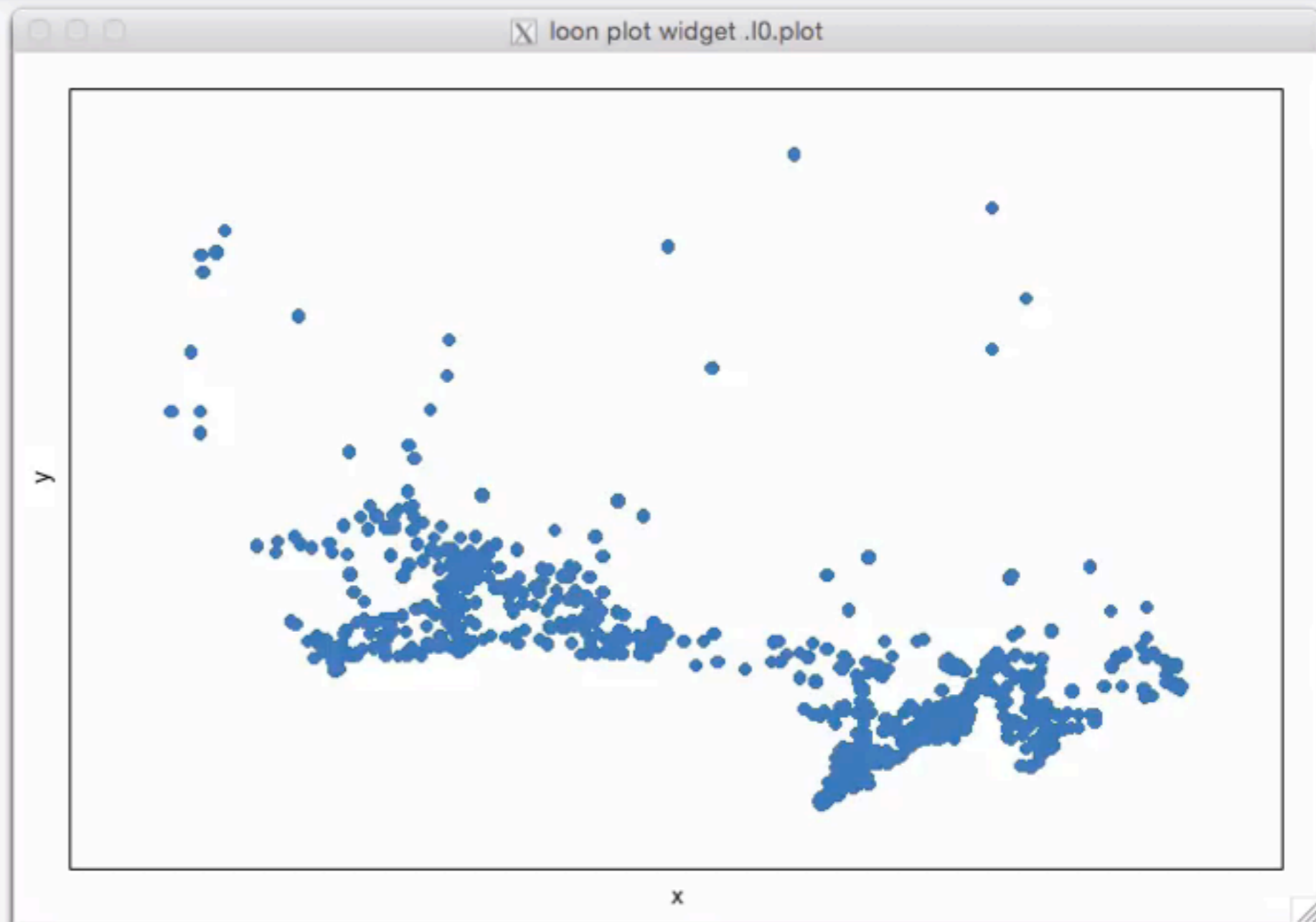
deactivate reactivate

move: 

glyph: 

--- set

size: abs: - + rel: - +

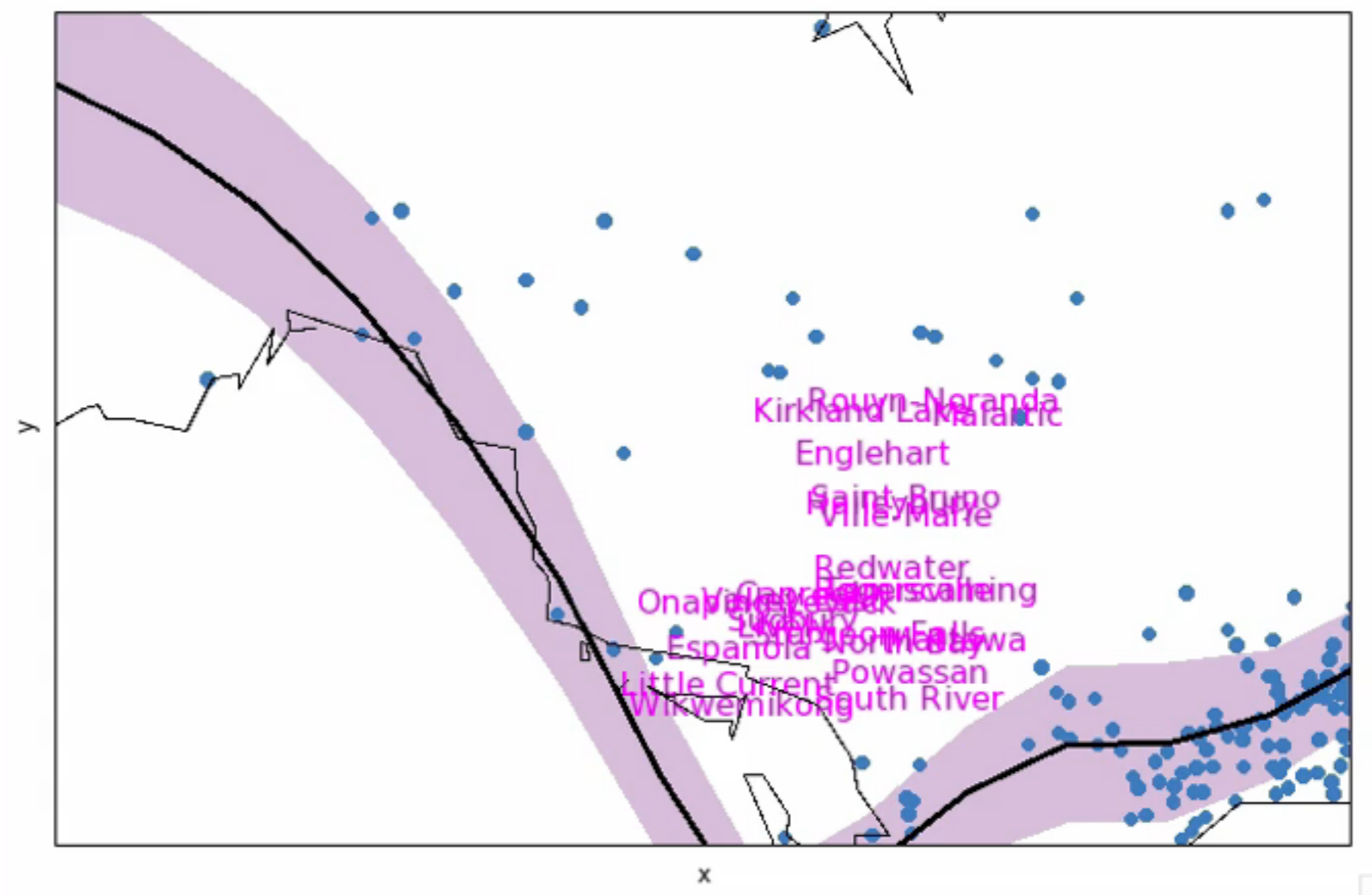


arwaddel - R - 61x10

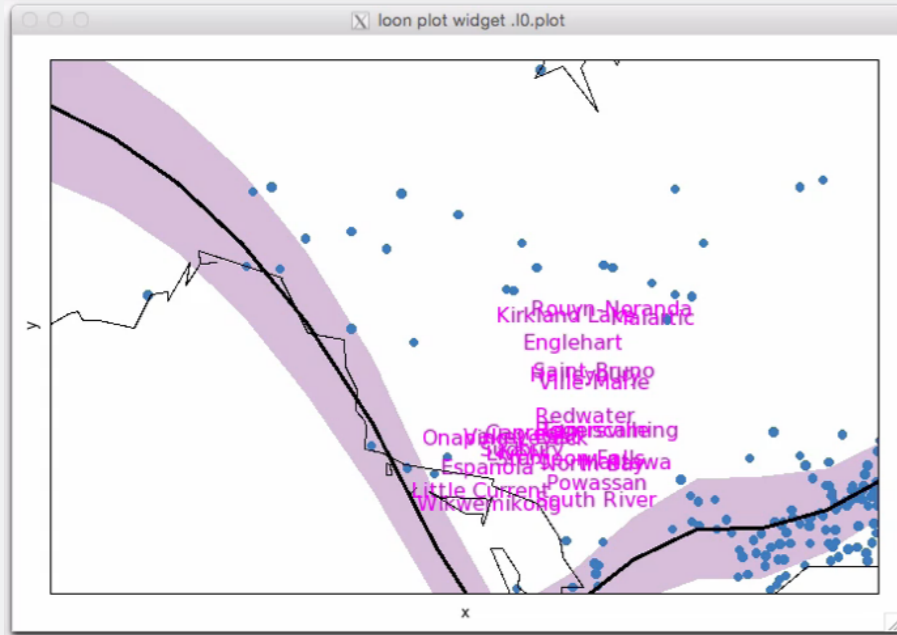
```
> l_resize(p, 693, 464)

> g <- l_glyph_add_text(p, text=canada.cities$name, label="ci
ty names")
::oo::0bj169::objects

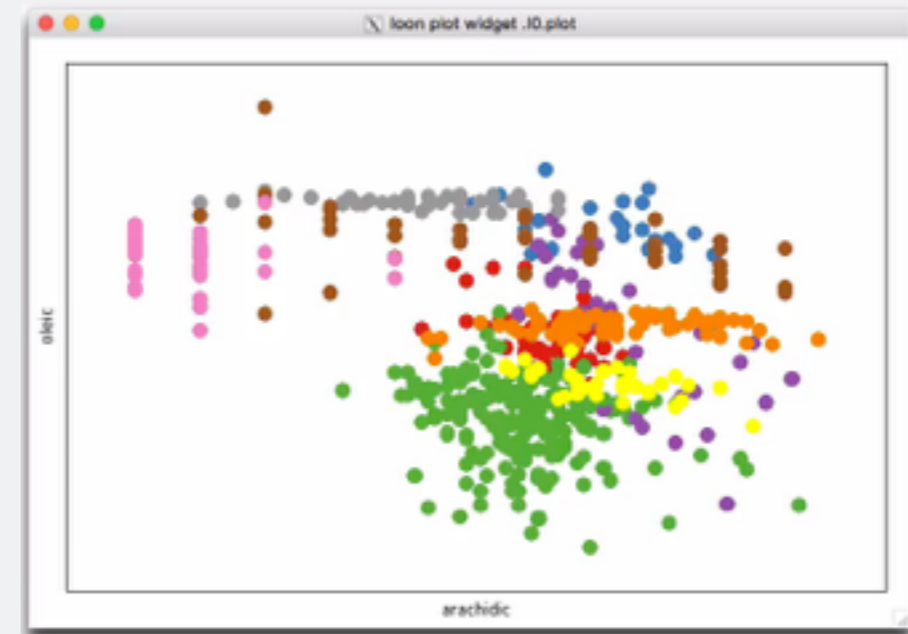
> readline("press the return key to continue: next add loess"
)
press the return key to continue: next add loess
```



Statistical Graphics



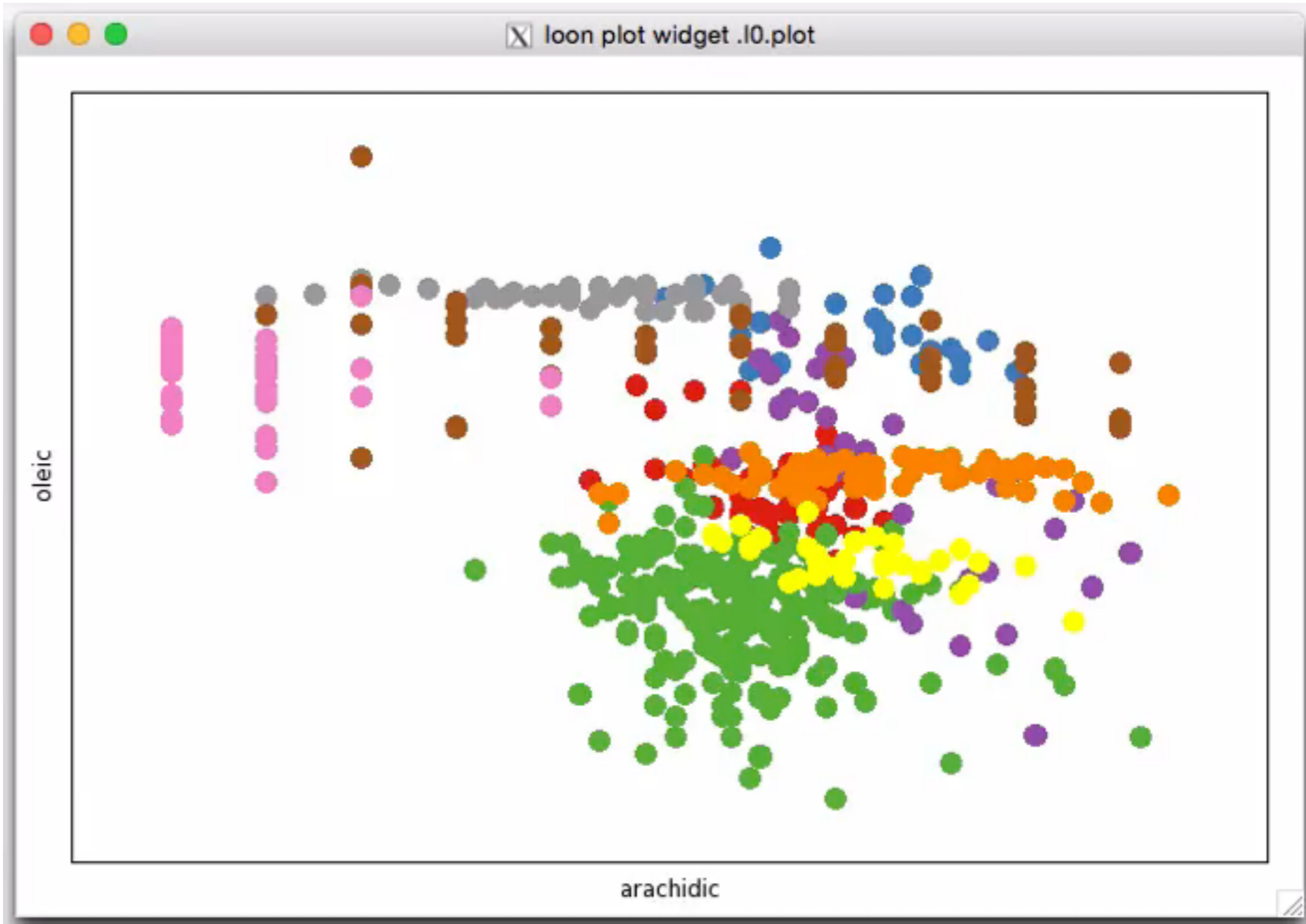
Layers



Point Glyphs

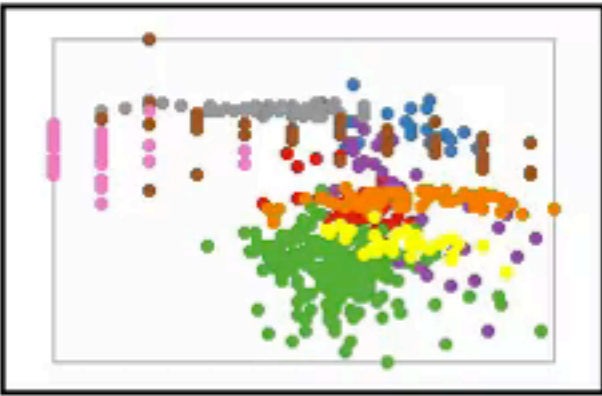
Dynamic Linking

Inspectors



Loon Inspector

World View



Analysis Layers Glyphs

Plot

axes: swap labels
 scales guides

glyphs: pointlabels

linking group: none

scale to: selected plot world


Select

static: all none invert


dynamic: select deselect invert

by: sweeping brushing

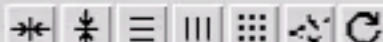
by color:




Modify



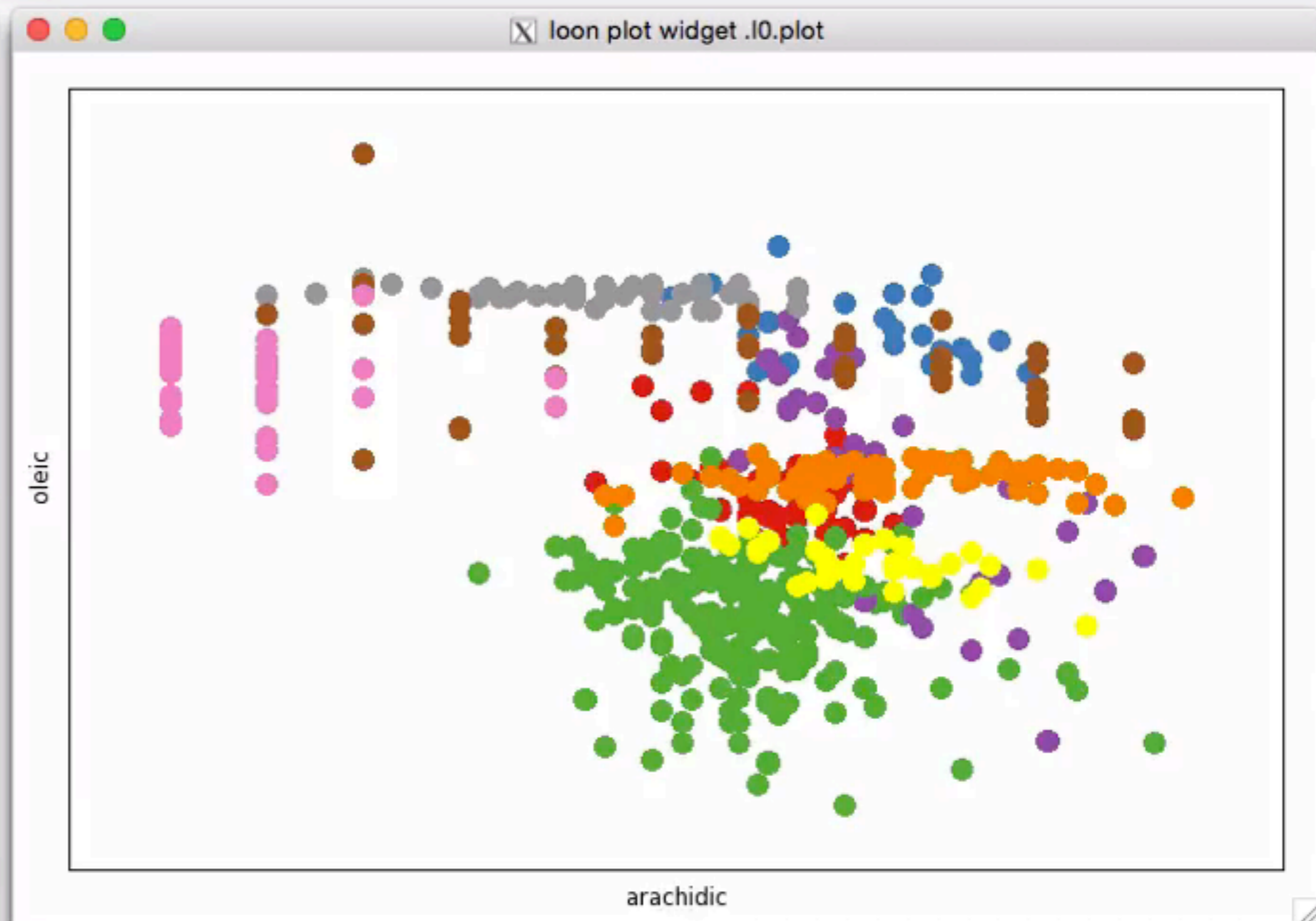
deactivate reactivate

move: 

glyph: 

--- set

size: abs: - + rel: - +



arwaddel - R - 61x10

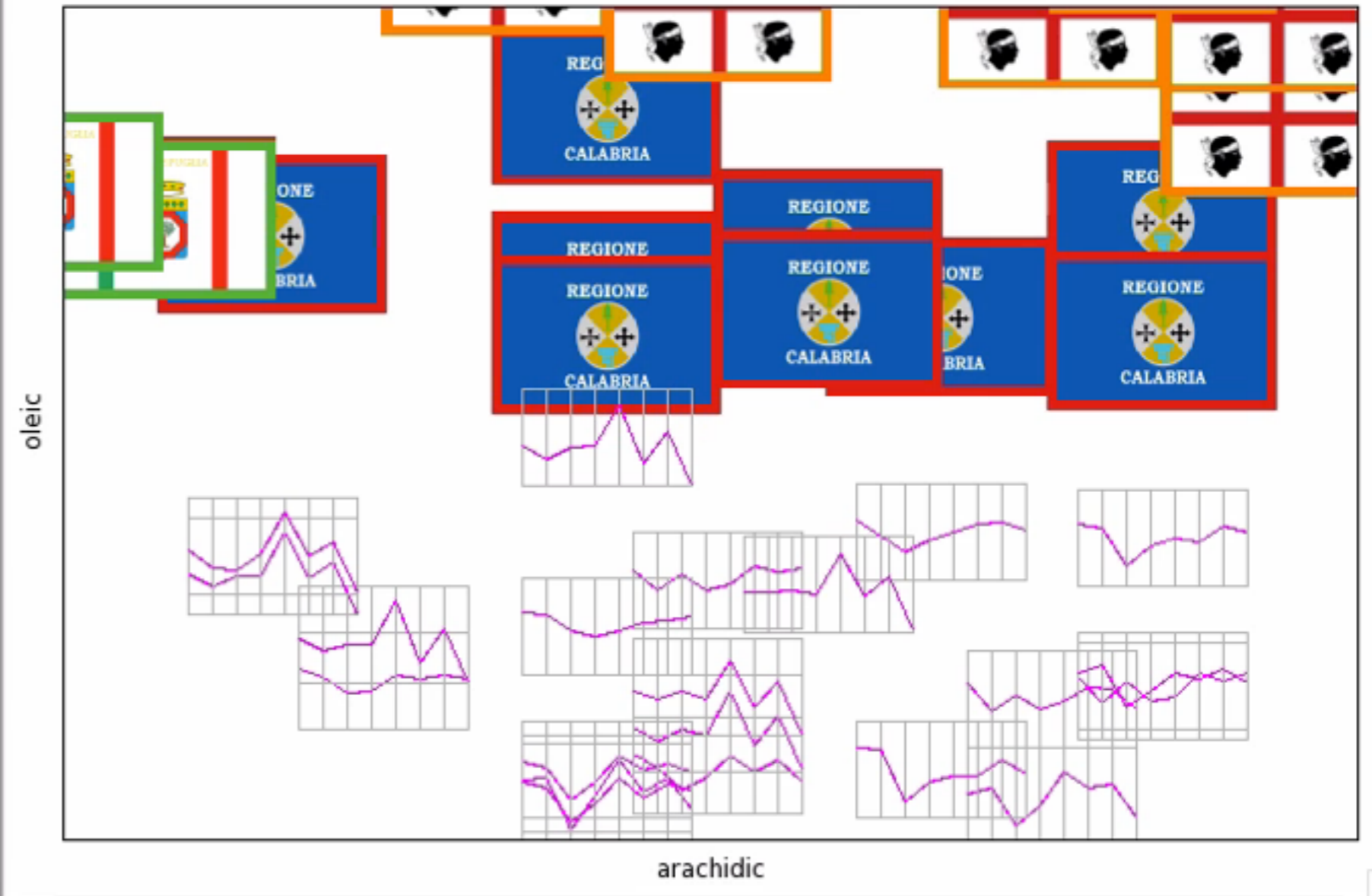
```

)
::oo::0bj169::objects

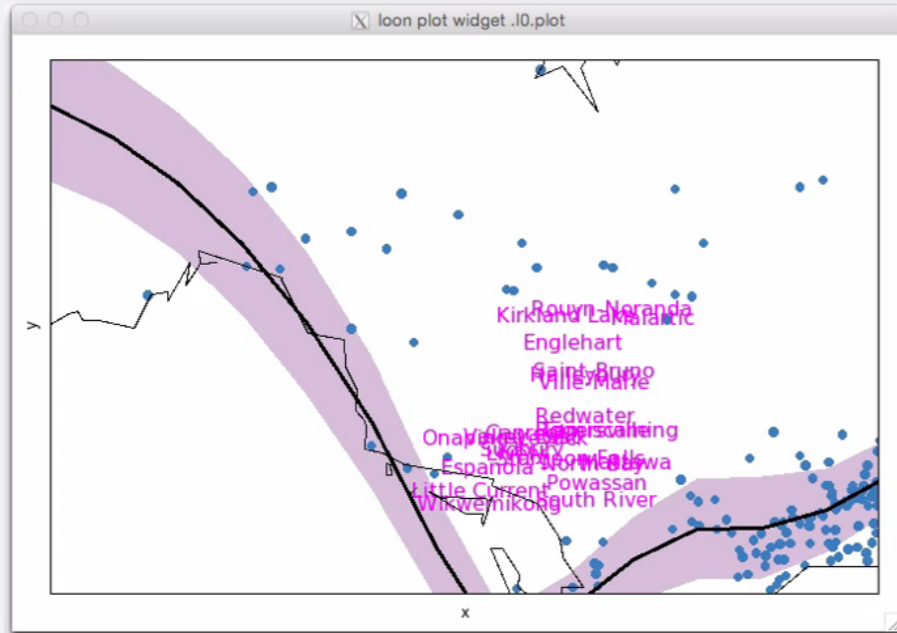
> ## Serialaxes
> sa <- l_glyph_add_serialaxes(p, data=olive[,-c(1,2)], label
="all variables")
::oo::0bj169::objects

> l_resize(p, 693, 464)
>

```



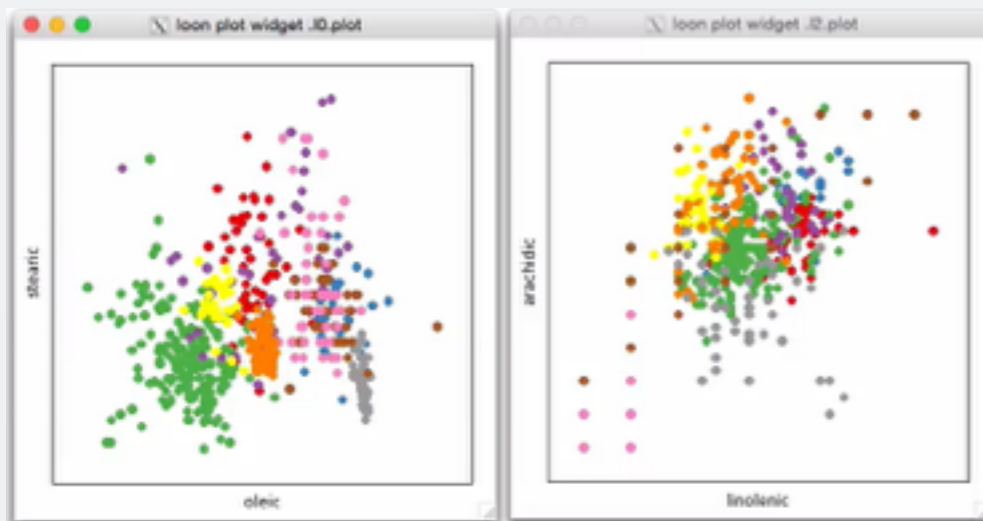
Statistical Graphics



Layers

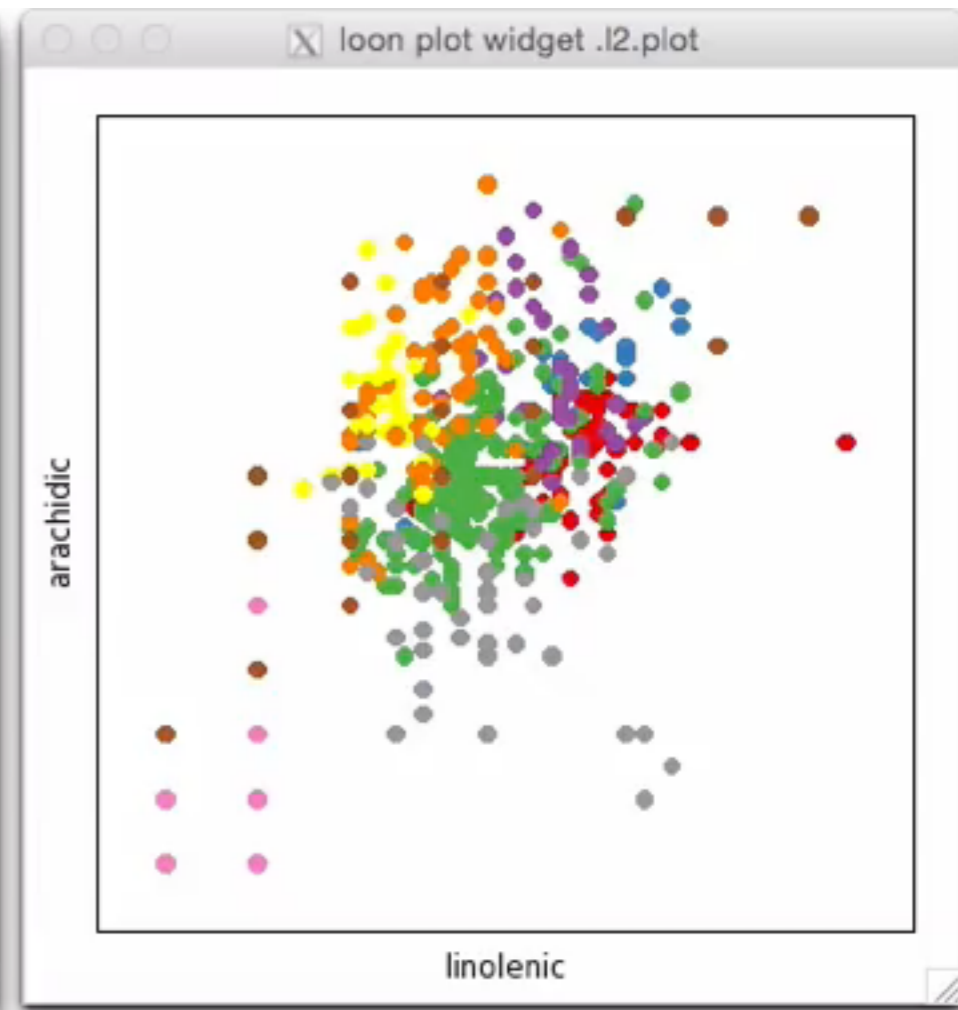
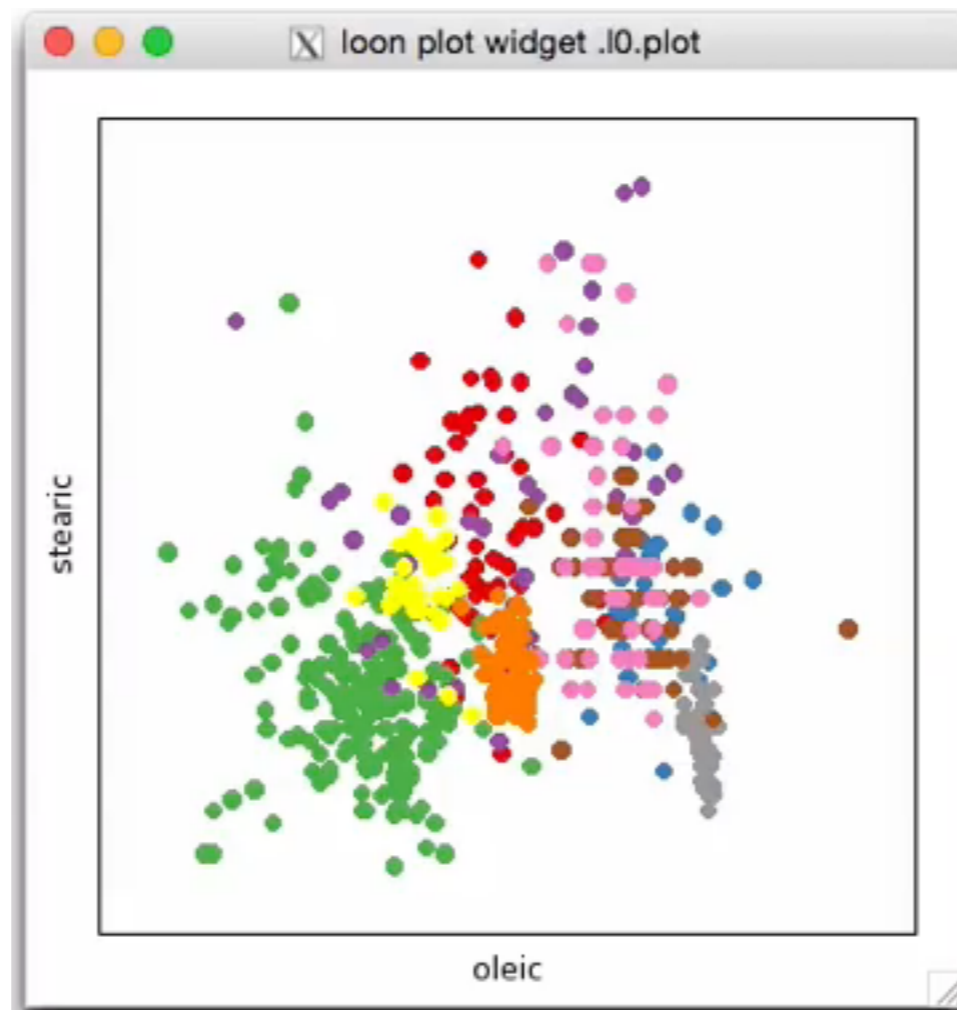


Point Glyphs



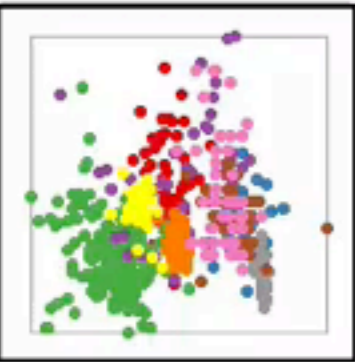
Dynamic Linking

Inspectors



Loon Inspector

World View



Analysis Layers Glyphs

Plot

axes: swap labels
 scales guides

glyphs: pointlabels

linking group: none

scale to: selected plot world


Select

static: all none invert


dynamic: select deselect invert

by: sweeping brushing


by color:

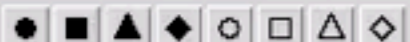


Modify

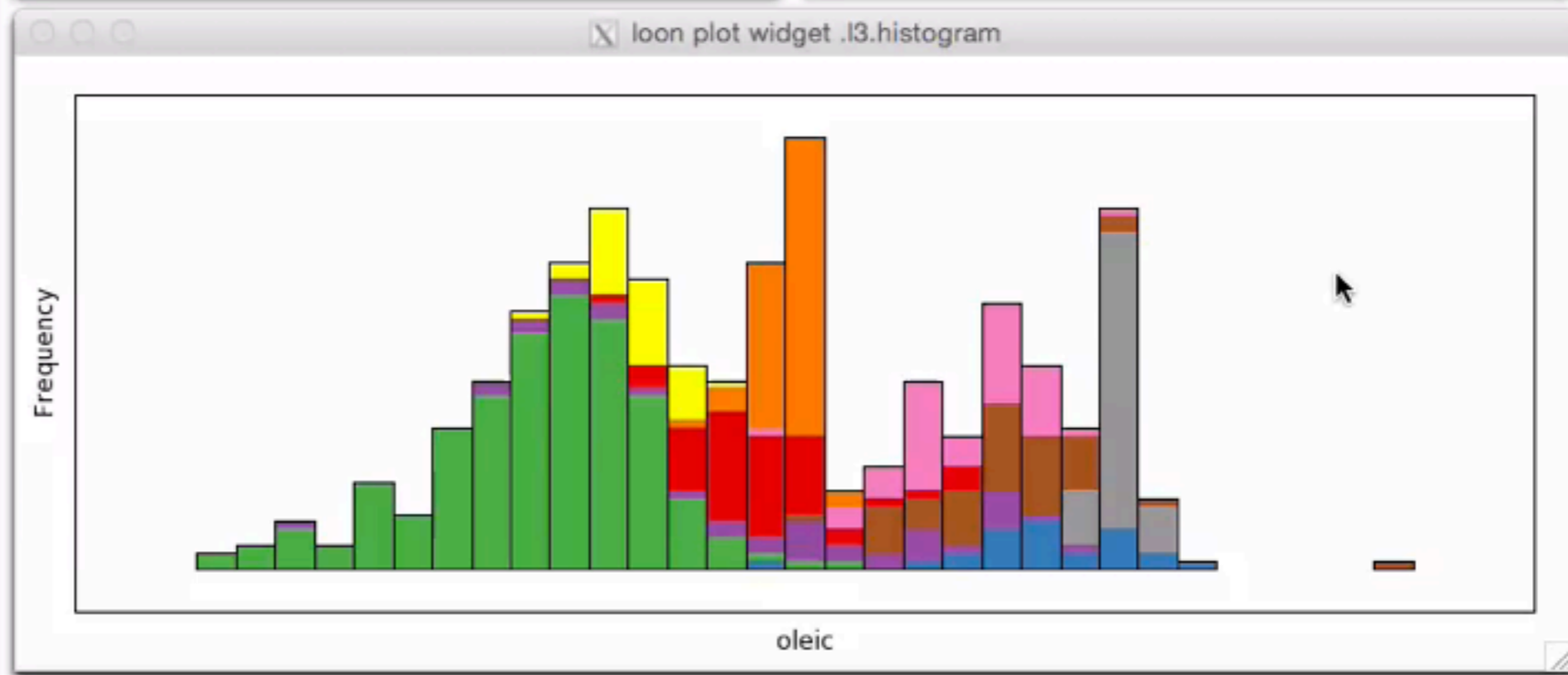
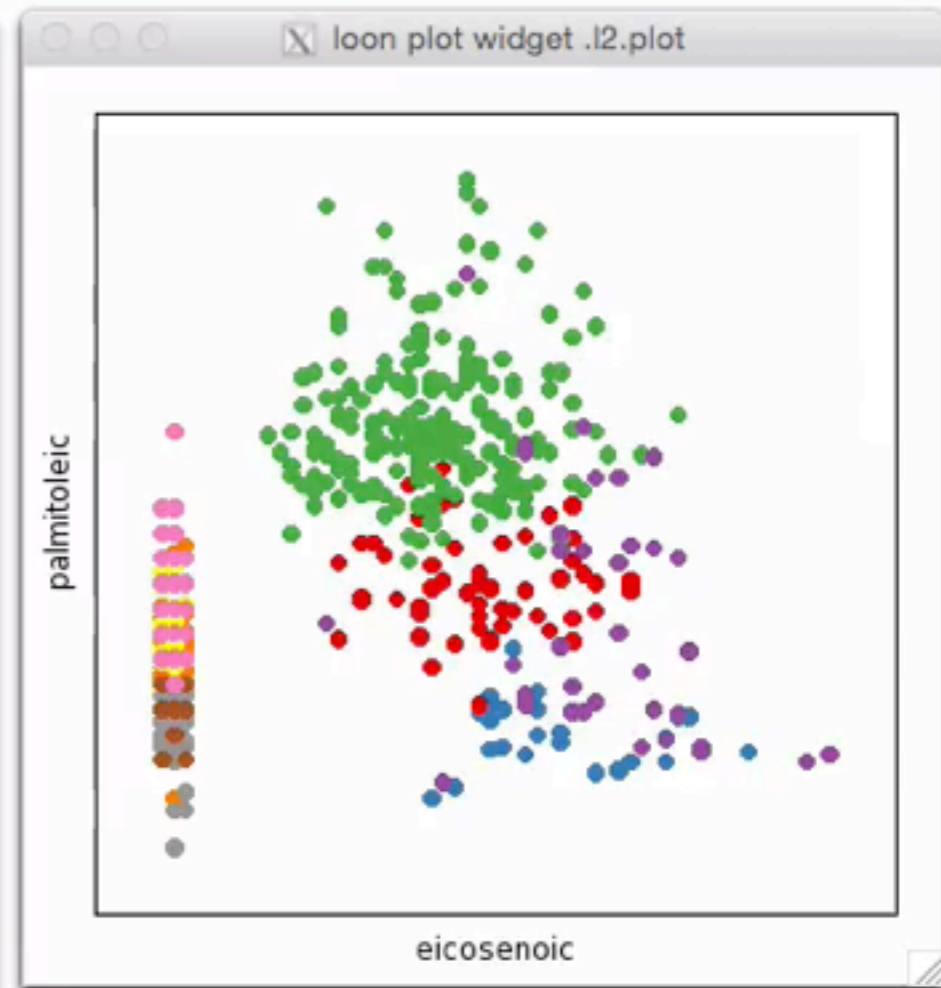
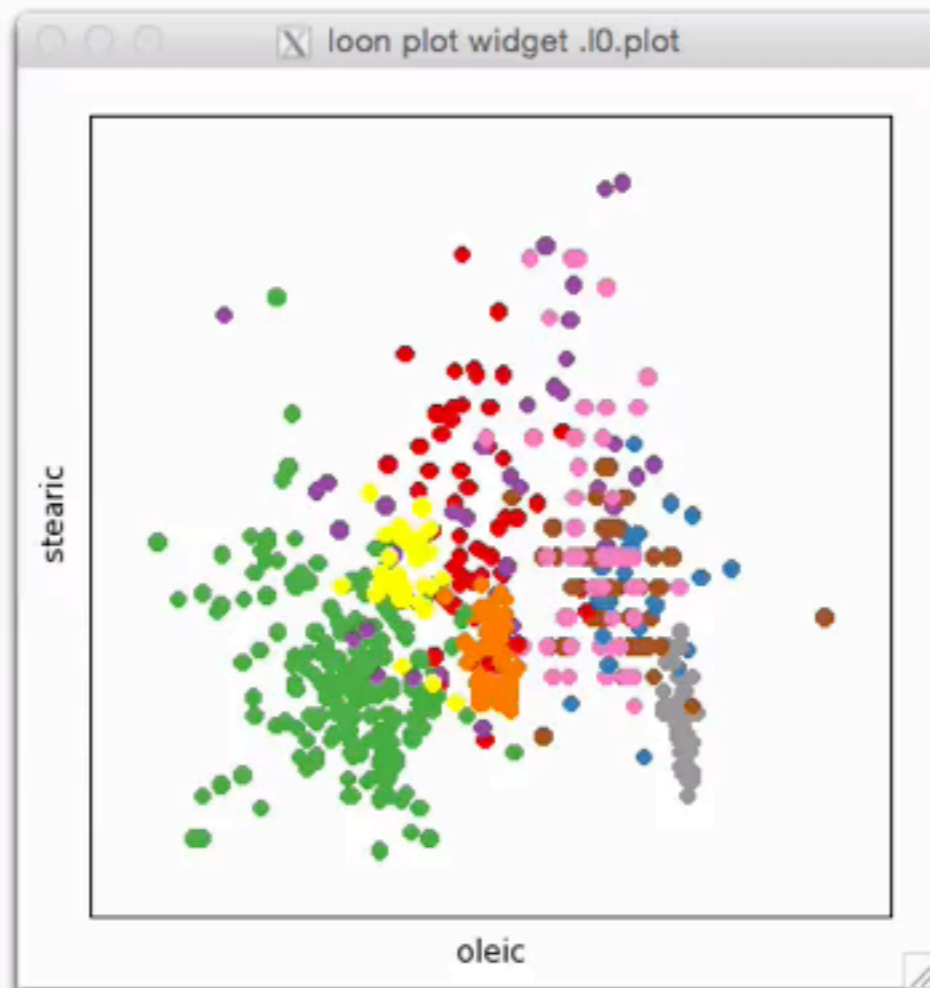


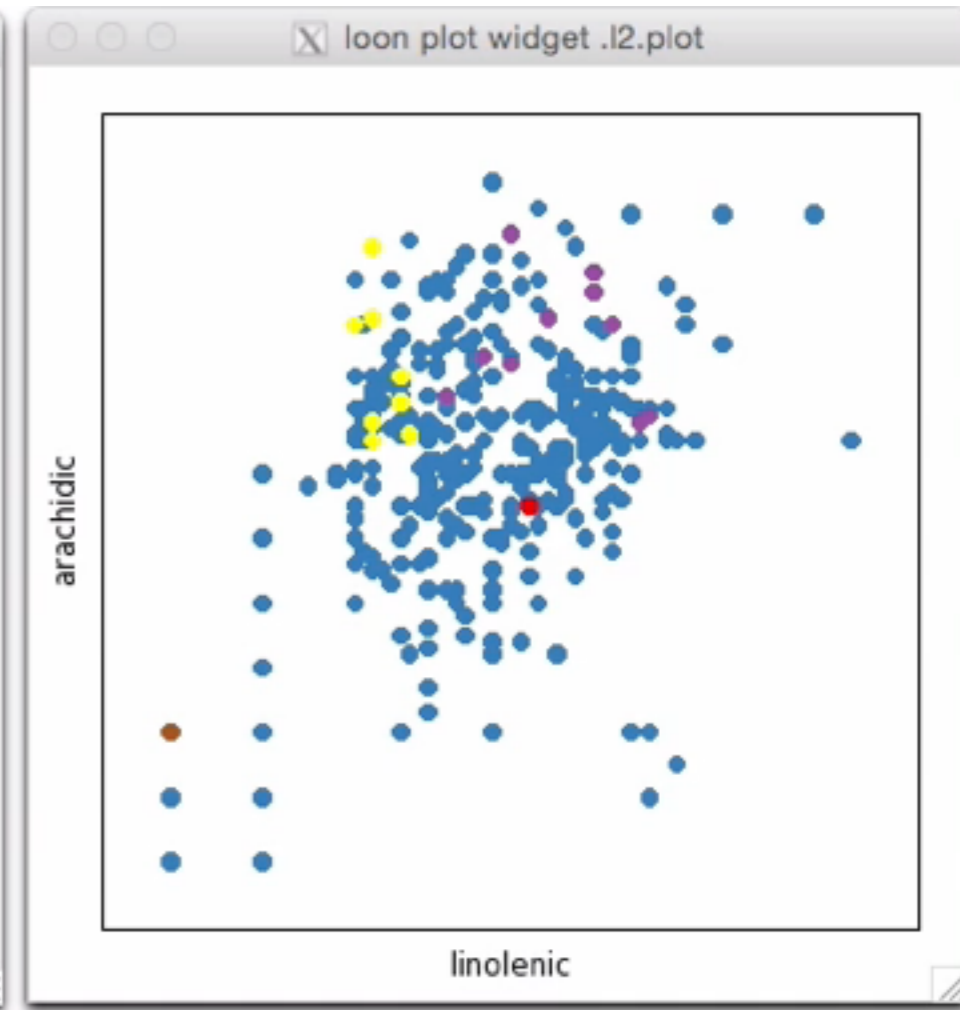
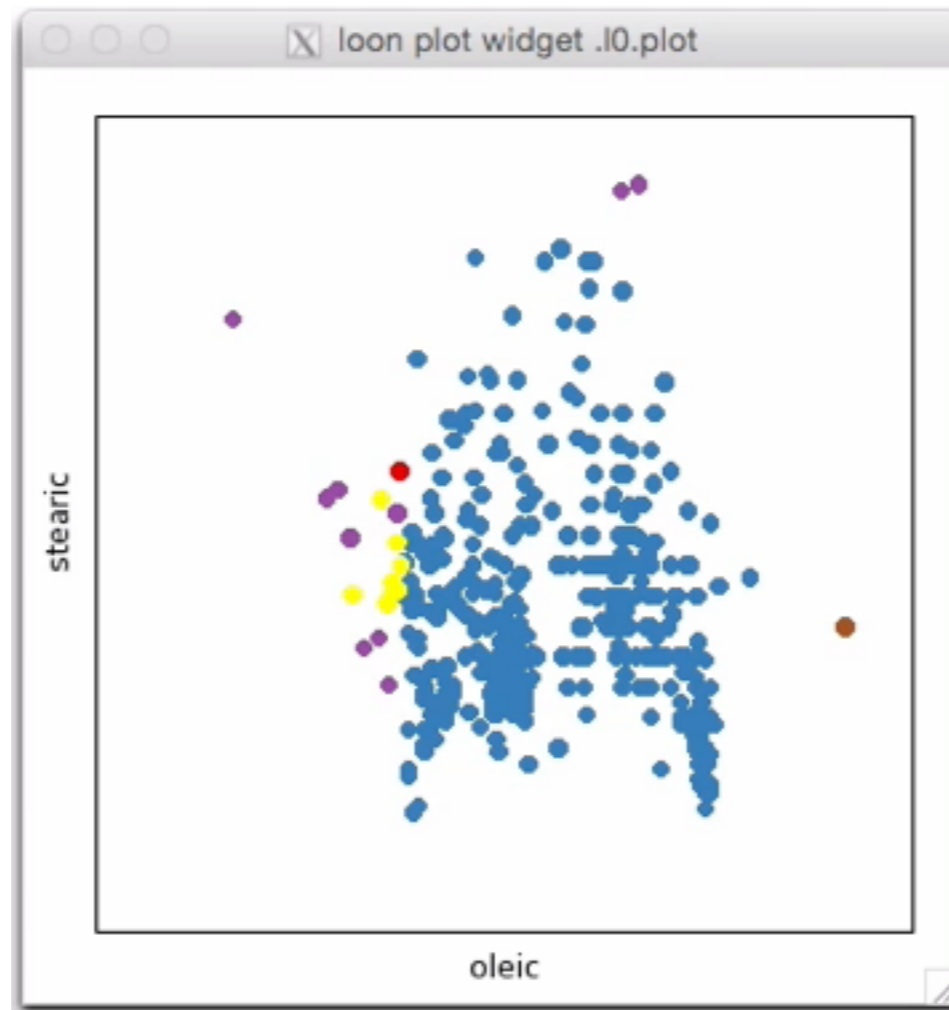
deactivate reactivate

move: 

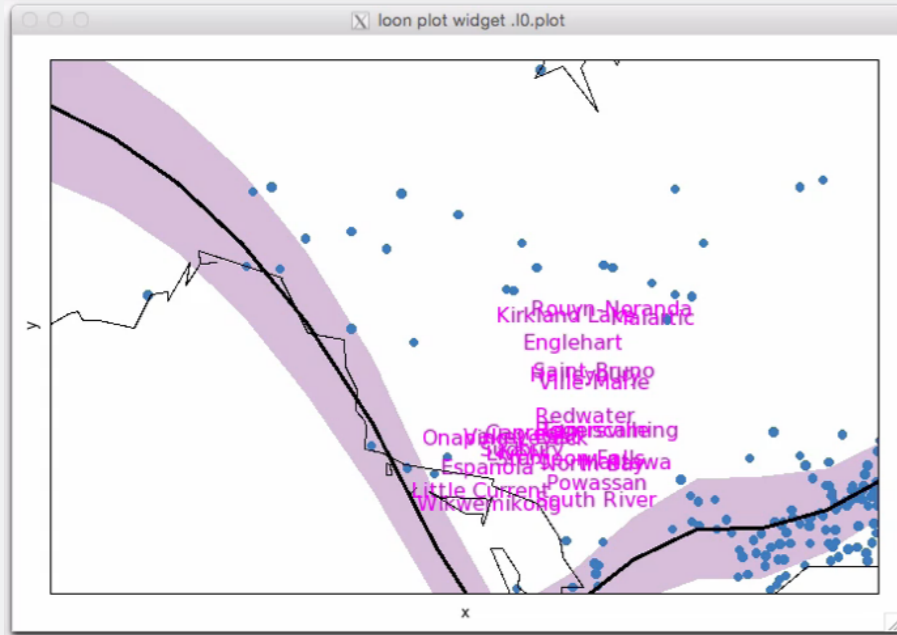
glyph: 

size: abs: - + rel: - +





Statistical Graphics



Layers



Point Glyphs



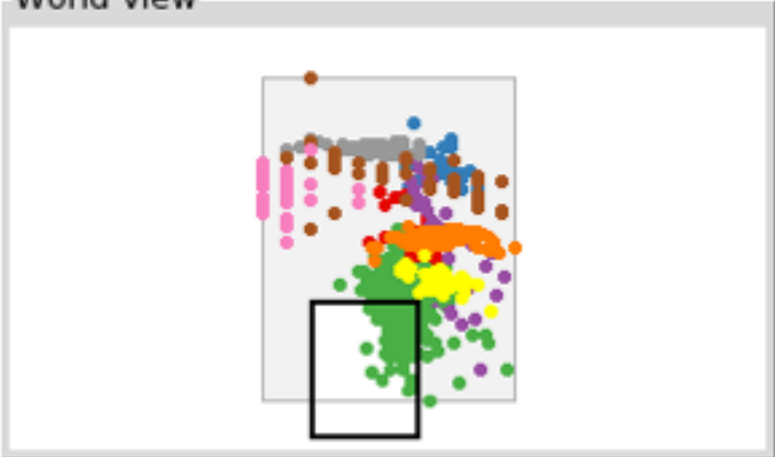
Dynamic Linking



Inspectors

Loon Inspector

World View



Analysis Layers Glyphs

Plot

axes: swap labels
 scales guides

glyphs: pointlabels

linking group: none

scale to: selected plot world

Select

static: all none invert

dynamic: select deselect invert

by: sweeping brushing

by color:

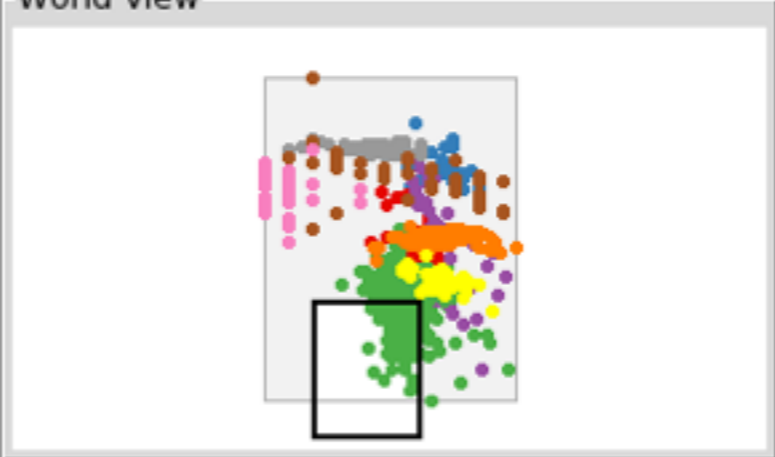
Modify

glyph0 set

size: abs: - + rel: - +

Loon Inspector

World View



Analysis Layers Glyphs

label	type	id
	text	glyph0
	image	glyph1
	serialaxes	glyph2

Serialaxes

showEnclosing
 showAxes
 transparent

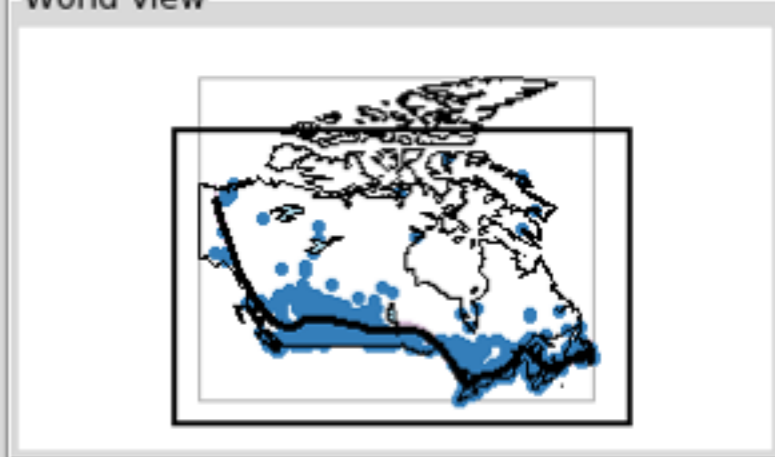
axes layout: radial parallel

scaling:

variable
 observation
 data
 none

Loon Inspector

World View



Analysis Layers Glyphs

label	type	id
▶ <input checked="" type="checkbox"/> canada	group	layer2
<input checked="" type="checkbox"/> loess fit	line	layer0
<input checked="" type="checkbox"/> Scatterplot	scatterpl	model
<input checked="" type="checkbox"/> Confidence Interval	polygon	layer1

set

Loon Inspector

World View

Analysis Layers Glyphs

Plot

axes: swap labels
 scales guides

glyphs: pointlabels

linking group: none

scale to: selected plot world

Select

static: all none invert

dynamic: select deselect invert

by: sweeping brushing

by color:

Modify

deactivate reactivate

move:

glyph:

glyph0 set

size: abs: - + rel: - +

Loon Inspector

World View

Analysis Layers

Plot

axes: swap scales
 guides labels

show: stacked colors bin handle
 outlines

yshows: frequency density spinogram

scale to: plot world

linking group: none

color stack order:

Select

static: all none invert

dynamic: select deselect invert

by: sweeping brushing

by color:

Modify

deactivate reactivate

Loon Inspector

Plot

showGuides
 showAxes
 showAxesLabels
 showLabels
 transparent

axes layout: radial parallel

scaling:
 variable
 observation
 data
 none

linking group: none

Select

all none invert

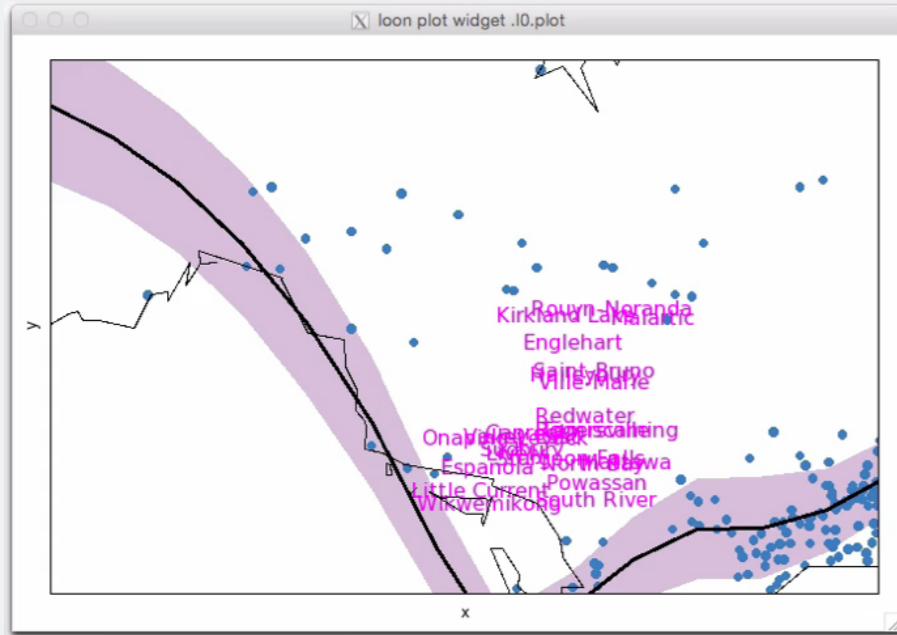
by color:

Modify

deactivate reactivate

linewidth: abs: - + rel: - +

Statistical Graphics



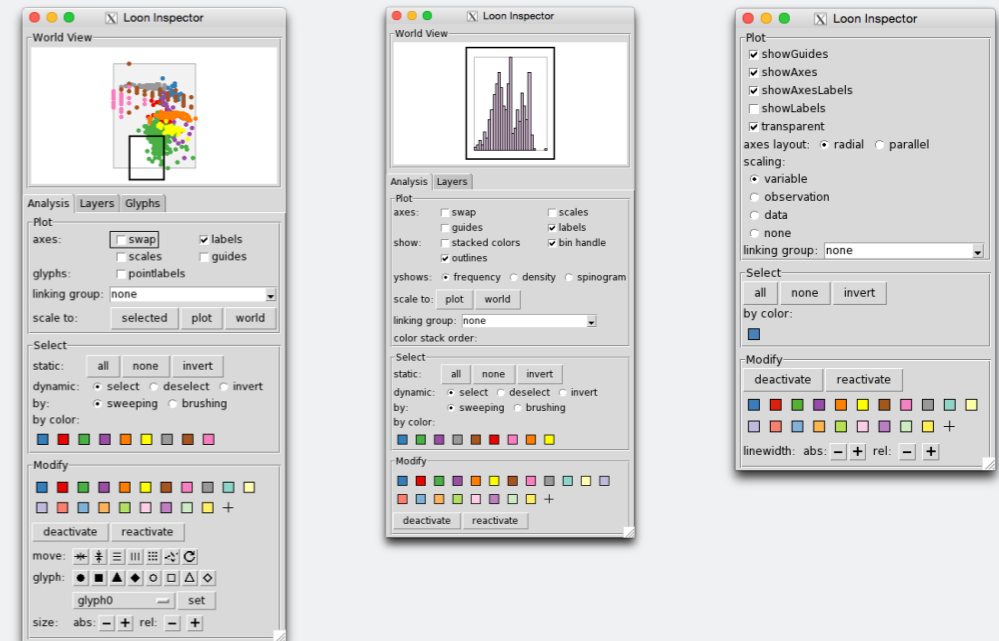
Layers



Point Glyphs



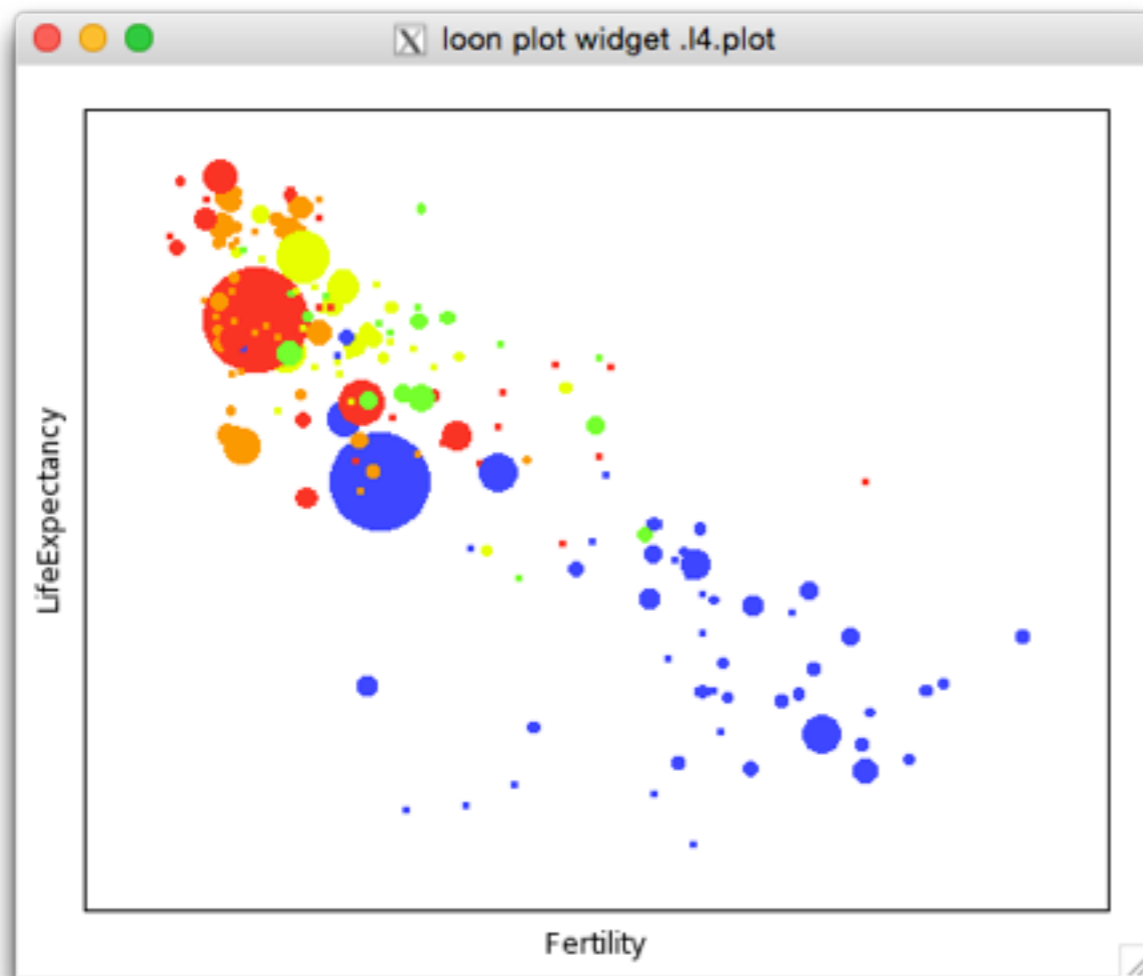
Dynamic Linking



Inspectors

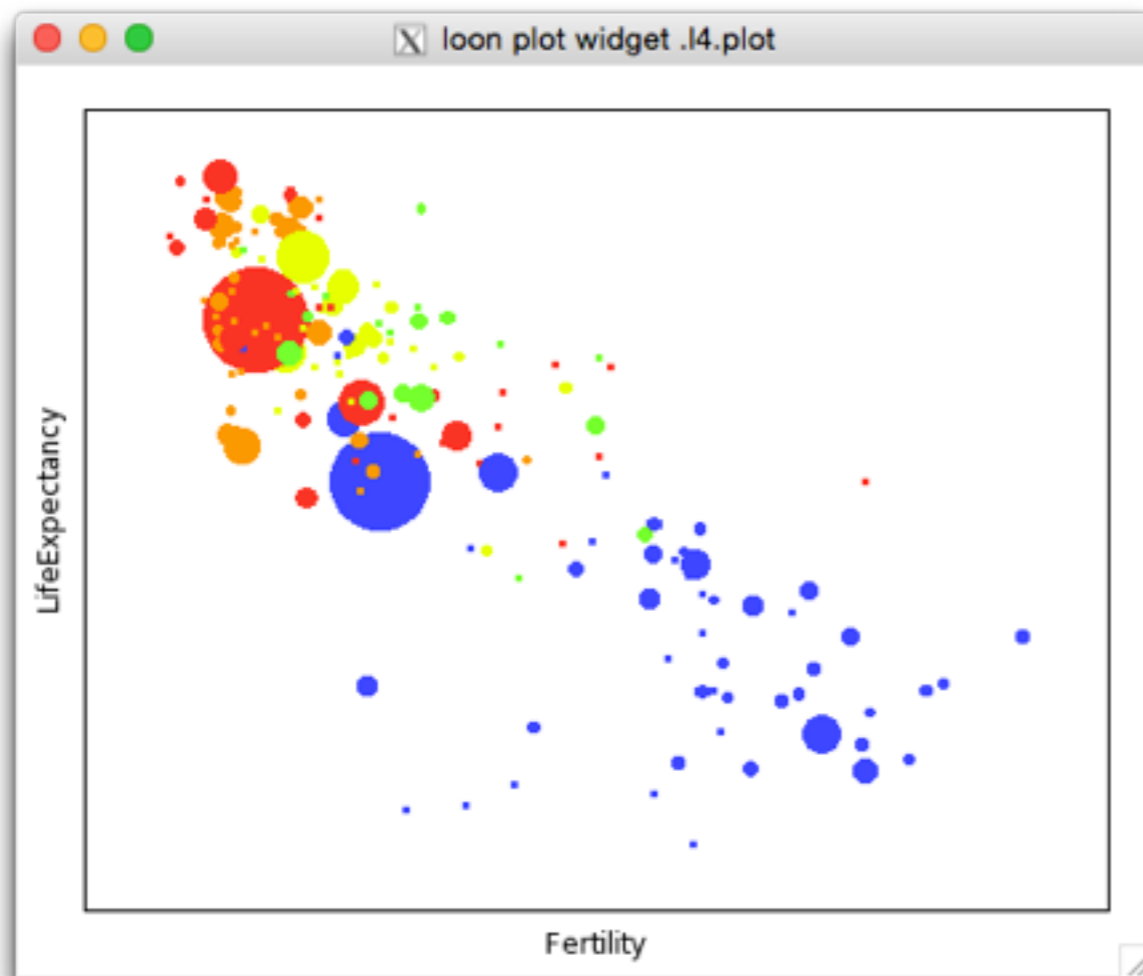
Framework

```
p <- l_plot(x=Fertility, y=LifeExpectancy,  
           color=region_cols, size=pop_size)
```



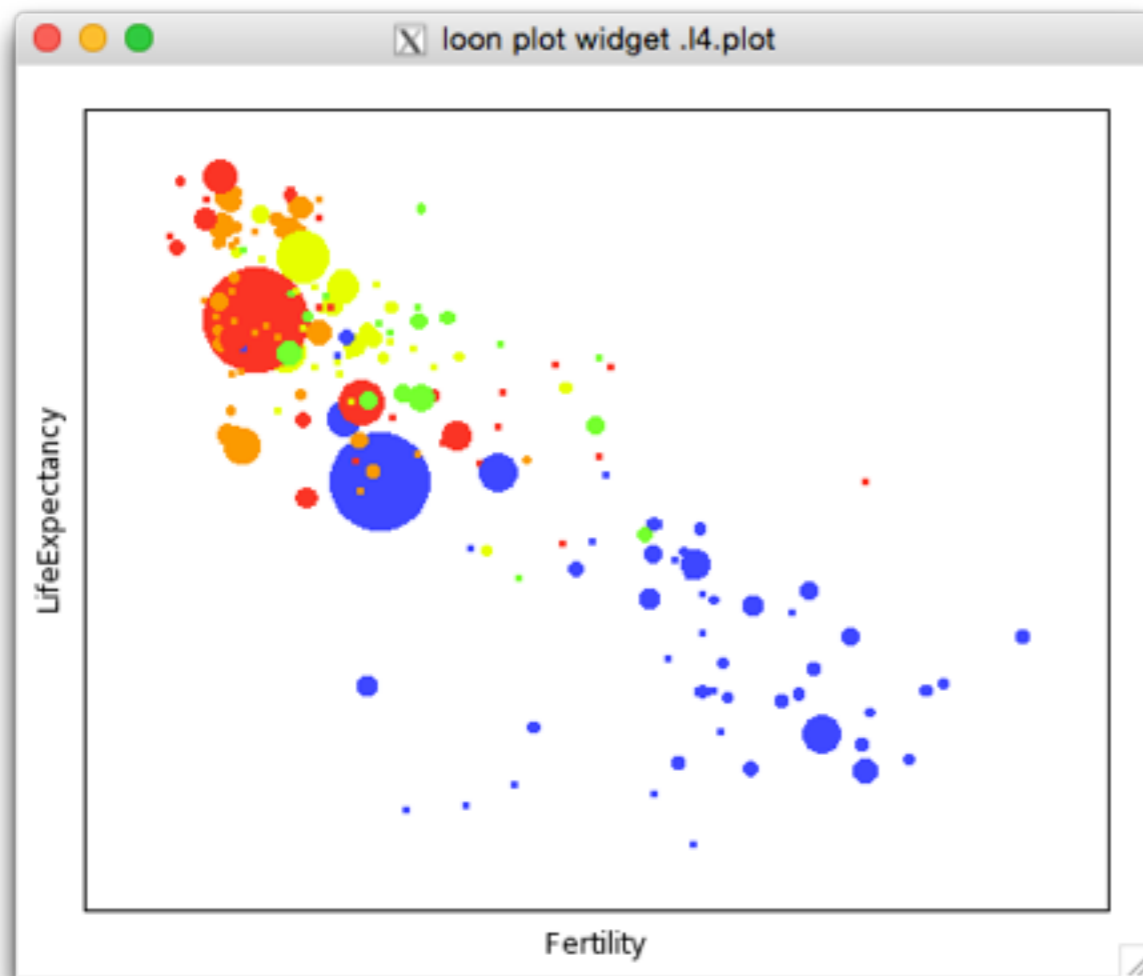
Framework

```
p <- l_plot(x=Fertility, y=LifeExpectancy,  
           color=region_cols, size=pop_size)
```



Framework

```
p <- l_plot(x=Fertility, y=LifeExpectancy,  
           color=region_cols, size=pop_size)
```



Plot States

x

y

color

size

selected

active

showScales

showLabels

...

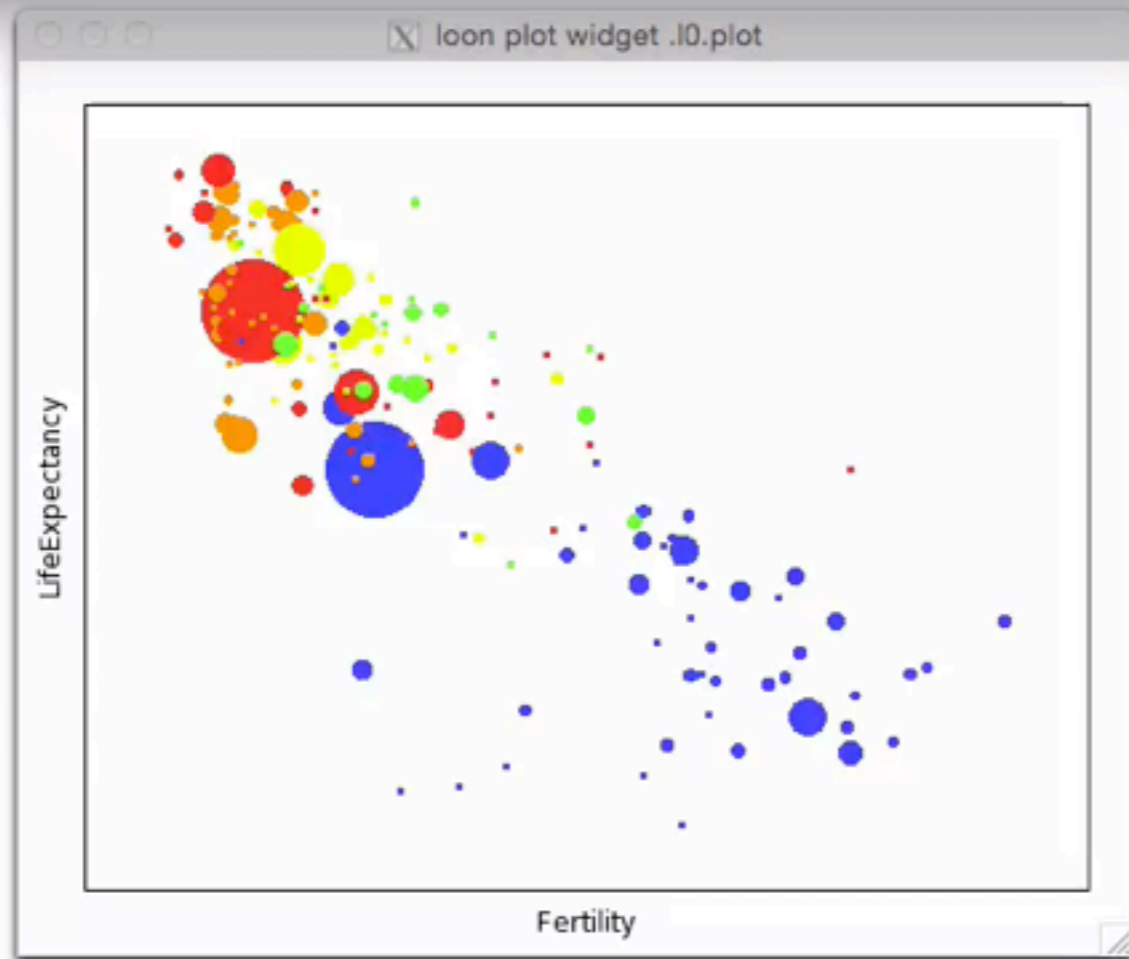
n dimensional

1 dimensional
boolean

~ 35 other states

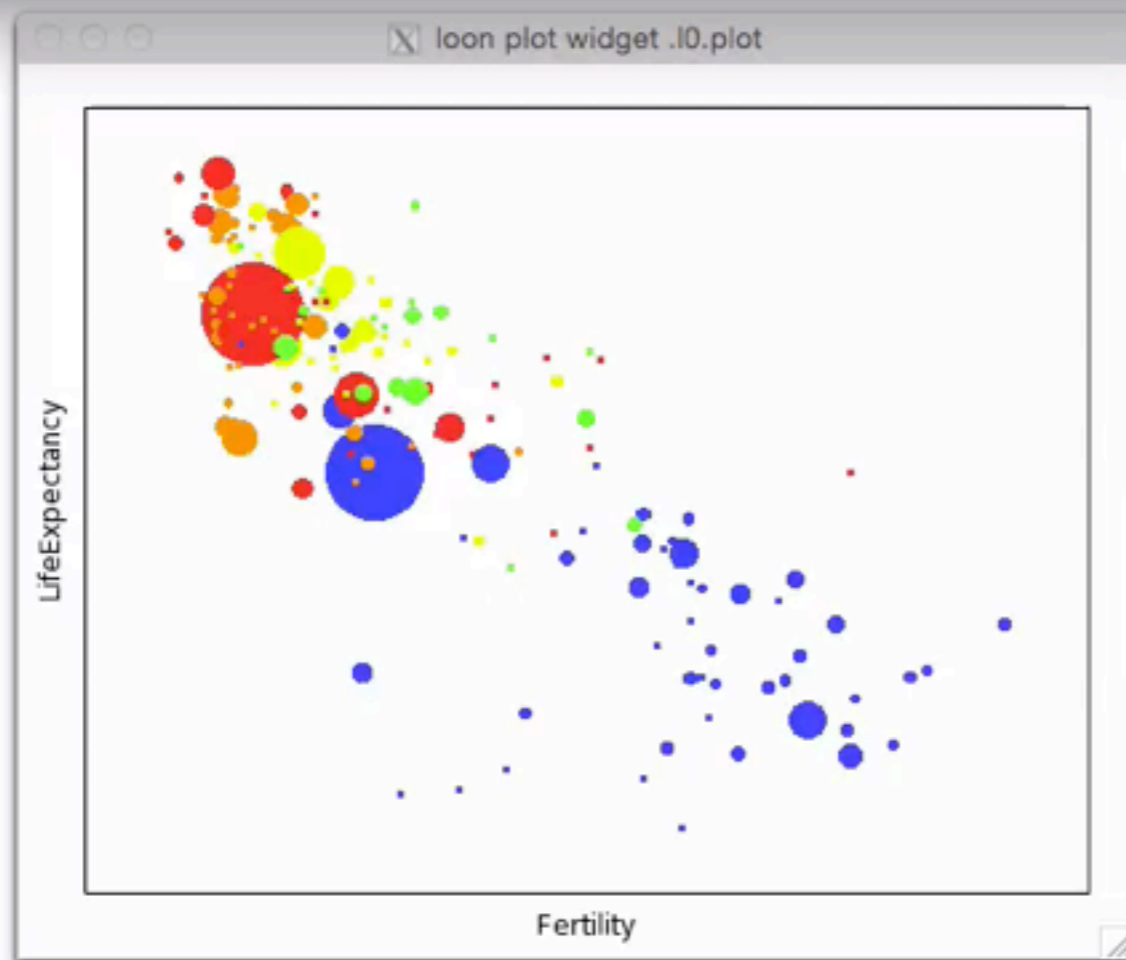
```
loon - R - 80x9
Loading required package: tcltk
Loading required package: maps

loon Version 0.7.8.9
> data(UsAndThem)
>
> p <- l_plot(x=Fertility, y=LifeExpectancy, color=region_cols, size=pop_size)
>
> █
```

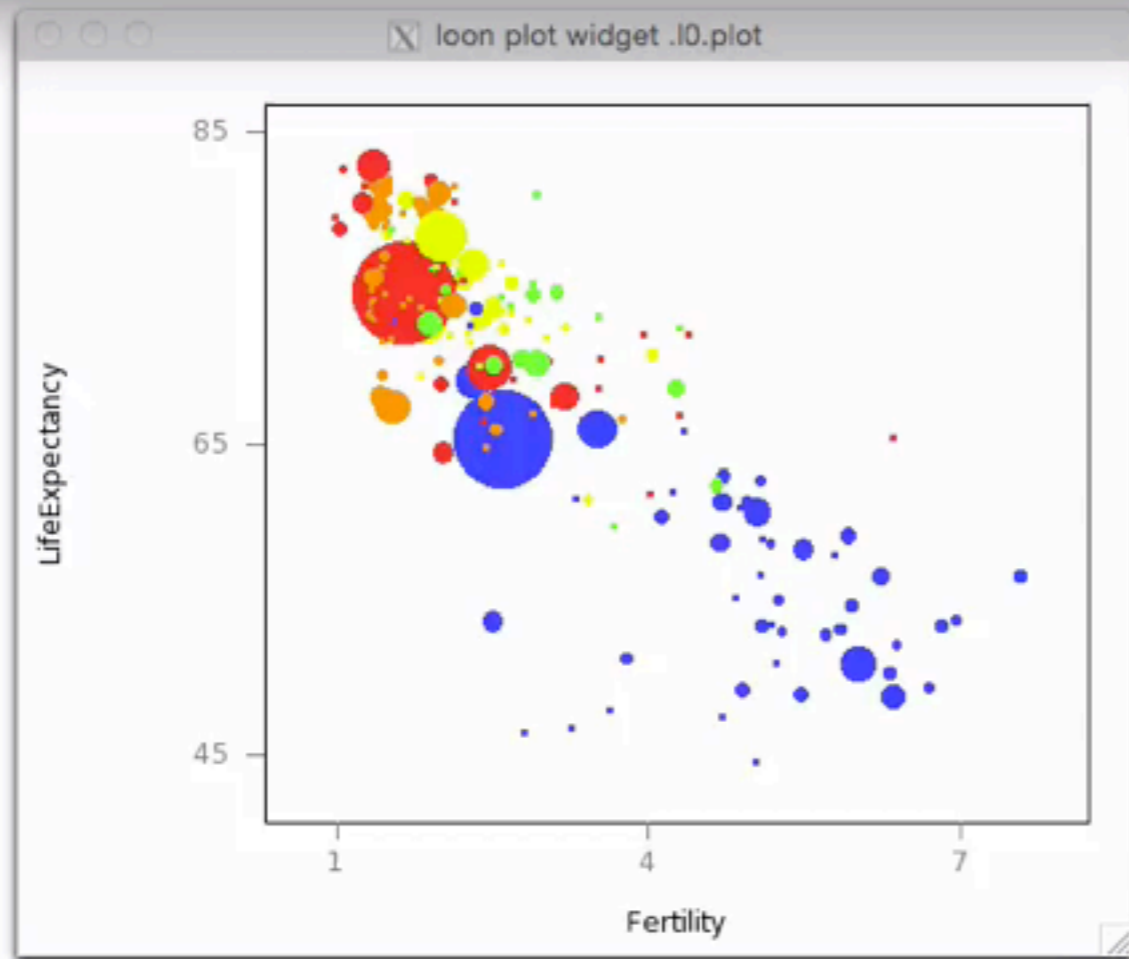


```
loon -- R -- 80x9

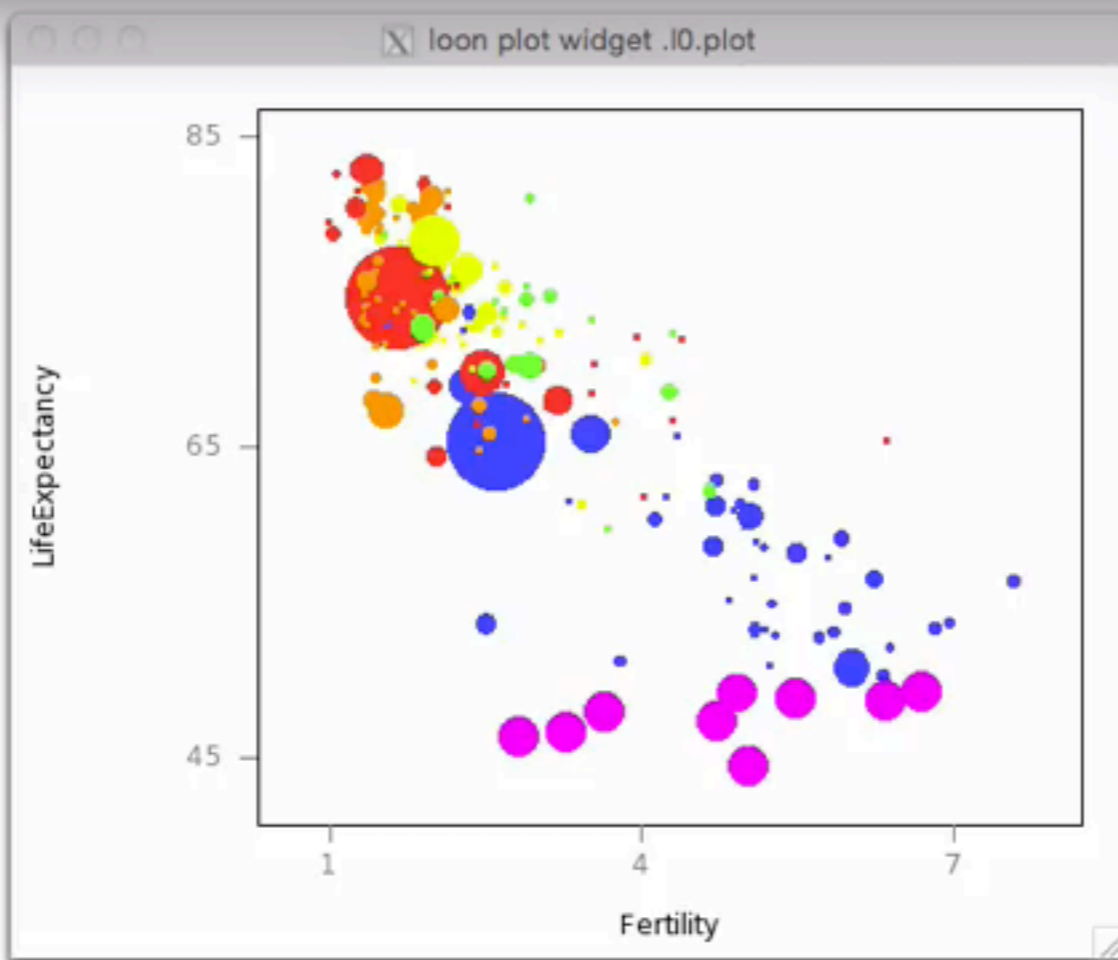
loon Version 0.7.8.9
> data(UsAndThem)
>
> p <- l_plot(x=Fertility, y=LifeExpectancy, color=region_cols, size=pop_size)
>
> p['showScales']
[1] FALSE
>
```



```
loon Version 0.7.8.9
> data(UsAndThem)
>
> p <- l_plot(x=Fertility, y=LifeExpectancy, color=region_cols, size=pop_size)
>
> p['showScales']
[1] FALSE
> p['showScales'] <- TRUE
>
```



```
loon - R - 80x9
> p['showScales']
[1] FALSE
> p['showScales'] <- TRUE
> l_configure(p, selected=TRUE, size=28, which = LifeExpectancy < 50)
> l_bind_change(p, 'selected', function() {
+   cat(paste("# selected points", sum(p['selected']), "\n"))
+ })
[1] "chgb0"
>
```



Example Visualizing Adverse Events

- **Generate Adverse Events Data**
 - Analysis Data Model (ADaM) Data Structure for Adverse Event Analysis

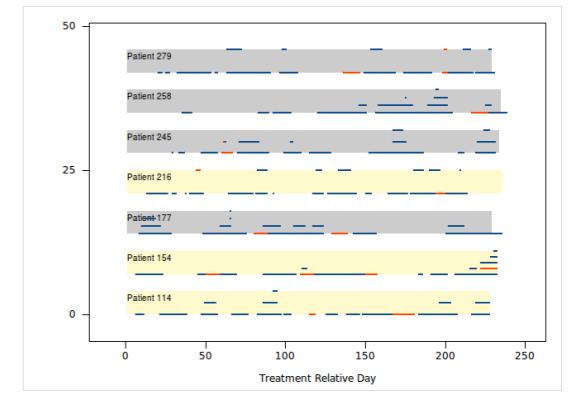
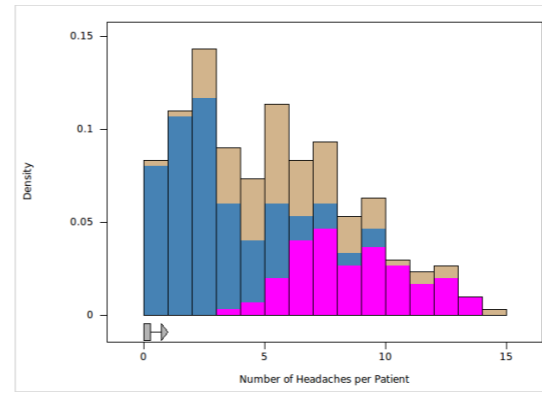
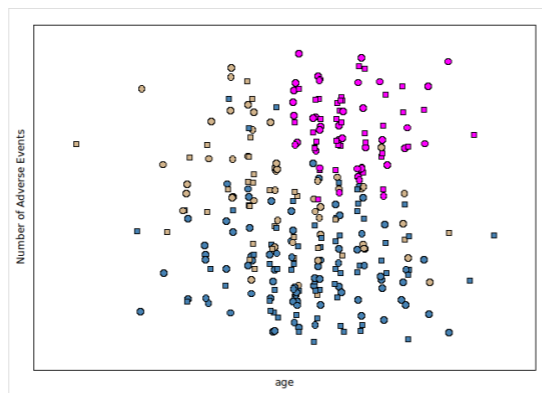
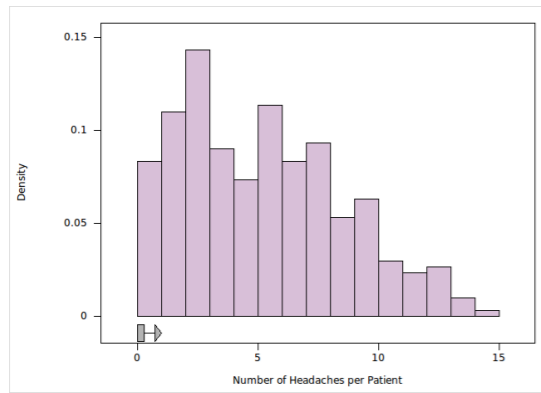
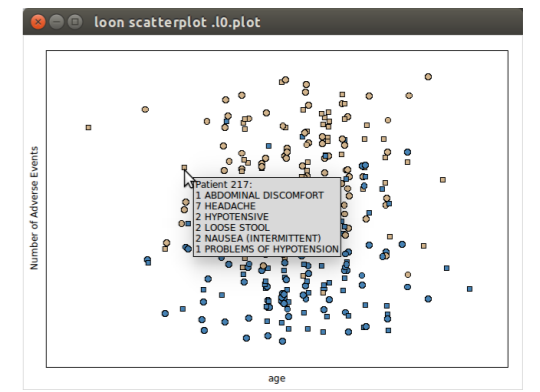
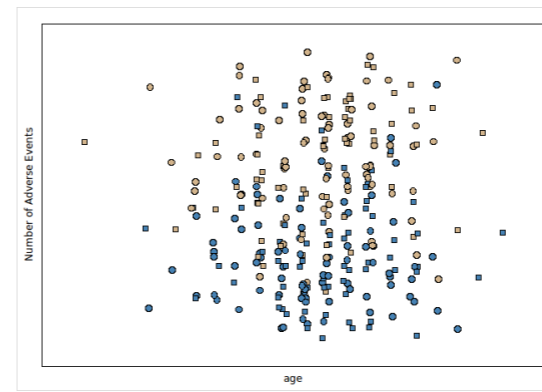
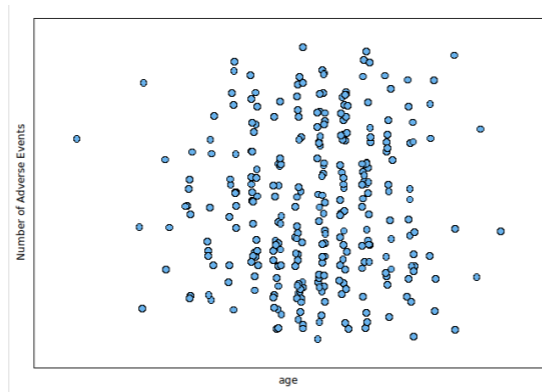
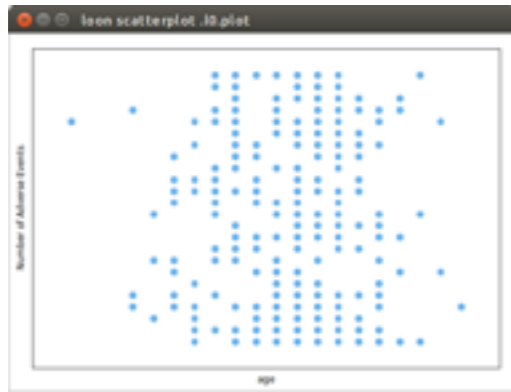
Name	Description
USUBJID	Unique Subject ID
SEX	Gender
AGE	Age
ARM	Study Arm
DISCDEAT	Discontinued Study due to Death
TRTSDT	Treatment Start
TRTEDT	Treatment End
AESEQ	Sequence Number
AETERM	Reported Term for the Adverse Event
AESEVN	Analysis Severity/Intensity (N)
ASTDT	Analysis Start Date
AENDT	Analysis End Date
ADURN	Duration of Adverse Event



Example Visualizing Adverse Events

live demo

Example Visualizing Adverse Events





waddella.github.io/loon