Gadfly Reference Card   v0.9
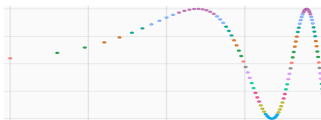
1 Introduction

Gadfly was developed by Daniel Jones to provide a system for plotting and visualization based on ggplot2 and the book "The Grammar of Graphics"
Details and examples can be found in the manual.   http://dcjones.github.io/Gadfly.jl

Grammar represents an abstraction of objects aiming to shorten the distance from mind to page by mapping data to aesthetic attributes and geometric objects. Think of looking at how y changes with x across levels of z.

```
Pkg.add("Gadfly") to install
Load with   using Gadfly
```

2 Invoking Plot .

This form is the standard "grammar of graphics" method of plotting. Data is supplied in the form of a dataframe, columns of the data are bound to *aesthetics*, and plot elements including *scales, coordinates, statistics, guides, and geometries* are added to the plot.

Conventional with dataframes

```
plot(data::AbstractDataFrame, elements::Element...; mapping...)
plot(dfname, x="c1", y="c2")              generates plot of y vs x with points
plot(dfname, x="c1", y="c2", Geom.lines)  generates plot of y vs x with lines
plot(df ..., color="c3", ...point)        add colors based on categories in c3
```

Traditional with vectors (columns)

```
plot(elements::Element...; mapping...)
plot(x=collect(1:20),y=fn(same length))   using arrays
```

Functions and Expressions

```
plot(f::Function,a,b, elements::Element...)
@plot(expr, a, b)
plot([sin, cos], 0, 10pi)                 Plot sin and cos of 0 to 10pi radians
```

```
set_default_plot_size(1cm,8cm)            Set the default plot size
```

3 Modifying Aesthetics

```
plot(..., color ="c3", ...)              color based on categories in c3
middle, lower_hinge, upper_hinge,        aesthetics for box plots
lower_fence, upper_fence, outliers
x_min, x_max, y_min, y_max               aesthetics for rectbin
x, x_min, x_max, y, y_min, y_max         aesthetics for error bars
```

```
Themes are covered in section 9
```

4 Geometries do the actual drawing

```
plot(..., Geom.point, ...)               use points (default)
plot(..., Geom.lines, ...)               use lines
plot(..., Geom.histogram(bincount=10))   use histogram
plot(..., Geom.bar, ...)                 bar chart
plot(..., Geom.bar, Geom.errorbar)         if colored, position=:stack or :dodge
plot(...,  Geom.rectbin, color="c3")     use colored 2d rectangles
plot(x=1:length(sds), y=ys, ymin=ymins,  plot error bars
ymax=ymaxs, Geom.point, Geom.errorbar)
```

```
plot(...color="c3", Geom.point, Geom.smooth)   add loess smoothing of chart points
plot(..., yintercept=[1.0, 2.0],               add horizontal line(s) to chart
     Geom.hline(color="red", size=3mm))
plot(..., xintercept=[1.0, 2.0], Geom.vline)   add vertical line(s) to chart
other variations                               see statistics
```

4 Statistics transform one or more aesthetics

```
Stat.boxplot                             outputs the middle, and upper and
plot(dfname, x="c1", y="c2", Geom.boxplot)   lower hinge, and upper and lower
                                         fence aesthetics
Stat.density                             output line with kernel density
plot(df, x="price", Geom.density)        estimate from data
Geom.rectbin with Stat.histogram2d       heatmap style with density as colour
plot(dfname, x="c1", Geom.histogram2d)
Geom.line with Stat.smooth               smoothing of data; method (loess);
plot(...color="c3", Geom.point, Geom.smooth)   smoothing=blahblah
```

5 Guides draw graphics to support the visualization such as axis ticks, labels and keys

```
plot(..., Guide.xlabel="Time")           and ylabel; override auto-labelling
plot(..., Guide.colorkey("Key Title"))
```

6 Scales transform the data to aid visualization

```
Scale.x_discrete                         categorical - equally spaced coords
Scale.x_continuous( ... )                continuous variables (x and y)
  minvalue, maxvalue,                    aesthetics
  format=:scientific (or :plain or :auto)
Scale.x_log10   Scale.y_log10            apply logarithmic scaling to axis
Scale.x_log2, Scale.x_log
Scale.x_asinh, Scale.x_sqrt
```

7 Showing Facets and Layers

```
Facets are supported with subplot_grid
set_default_plot_size(20cm, 30cm)        Set plot size for multiple facets
plot(df,xgroup="c4", ygroup="c5", x="c1",   Plot grouped subplots
  y="c2", Geom.subplot_grid(Geom.point))
```

```
layer(x="c1", y="c2", Geom.point, color="c3")   for each of many layers on one chart
```

8 Output to other Formats

```
draw(PNG("myplot.svg", 6inch, 3inch), plot(...))      Also PNG, PDF, SVG and PS
draw(D3("myplot.js", 15cm, 20cm), plot(...))
```

9 Themes

Many parameters controlling the appearance of plots can be overridden by passing a Theme object to the plot function.

`default_point_size`	Size of points in the point and boxplot geometry. (Measure)
`line_width`	Width of lines in the line geometry. (Measure)
`panel_fill`	Background color used in the main plot panel. (ColorValue or Nothing)
`panel_stroke`	Border color of the main plot panel. (ColorValue or Nothing)
`default_color`	When the color aesthetic is not bound, geometry uses this color for drawing. (ColorValue)
`grid_color`	Color of grid lines. (ColorValue or Nothing)
`grid_color_focused`	In the D3 backend, mousing over the plot makes the grid lines emphasised by transitioning to this color. (ColorValue or Nothing)
`grid_line_width`	Width of grid lines. (Measure)
`minor_label_font`	Font used for minor labels such as guide entries and labels. (String)
`minor_label_font_size`	Font size used for minor labels. (Measure)
`minor_label_color`	Color used for minor labels. (ColorValue)
`major_label_font`	Font used for major labels usch as guide titles and axis labels. (String)
`major_label_font_size`	Font size used for major labels. (Measure)
`major_label_color`	Color used for major labels. (ColorValue
`guide_label_position`	Justification (default or :left)
`bar_spacing`	Spacing between bars in Geom.bar. (Measure)
`boxplot_spacing`	Spacing between boxplots in Geom.boxplot. (Measure)
`errorbar_cap_length`	Length of caps on error bars. (Measure)
`highlight_width`	Width of lines drawn around plot geometry like points, and boxplot rectangles. (Measure)
`highlight_color`	Color used to outline plot geometry. This is a function that alters (e.g. darkens) the fill color of the geometry. (Function)
`middle_color`	Color altering function used to draw the midline in boxplots. (Function)
`middle_width`	Width of the middle line in boxplots. (Measure)
`colorkey_swatch_shape`	Shape of color key (default or :circle)

Example: `Theme(guide_label_position=:left, default_point_size=1mm)`