Cheatography

ggvis_18BCE2193 Cheat Sheet

by Anshumaan Singh (Anshumaan Singh) via cheatography.com/126462/cs/24498/

Installation

install.packages("ggvis")
library(ggvis)

install.packages("ggvis") will install all the required packages you need for visualization through ggvis
-library(ggvis) will call the ggvis package to be used in your visualization

Layers

Simple Layer	Multiple Layer
Here I am using the dataset mtcars and visualising it through layer points.	I have taken the mtcars dataset and visualized the multiple layers using different strokes

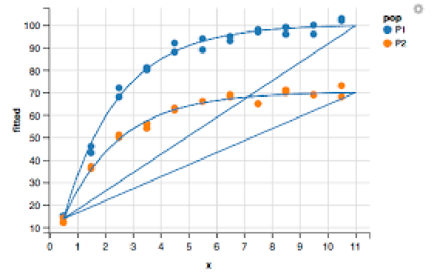
Global Vs Local properties

A property that is set inside ggvis() is applied globally. While a property set inside layer_<marks>() is applied locally. Local properties can override global properties when applicable.

Scale Types

Any visual property in the visual- ization can be adjusted with scale(). ggvis provides several different functions for creating scales:

Model Prediction



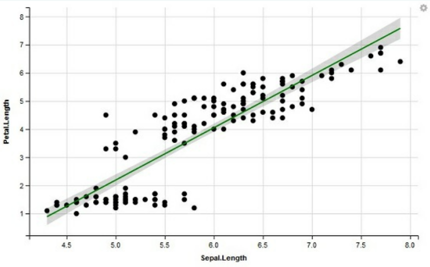
faithful %>%
ggvis(~eruptions,~waiting)
%>%
layer_points(fill := "gree- n", fillOpacity := 0.5) %>% layer_model_predictions(- model = "lm", stroke := "re- d") %>% layer_smooths(stroke :=)

Overview

Graphics

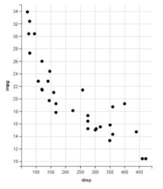
The graphics produced by ggvis are fundamentally web graphics and work very differ- ently from tradit- ional R graphics. This allows us to implement exciting new features like interactivity
The goal of ggvis is to make it easy to build interactive graphics for explor- atory data analysis. ggvis has a similar underlying theory to ggplot2 (the grammar of graphics).

Simple Layer



mtcars %>% ggvis(~mpg, ~disp,fill = ~vs) %>% layer_points()

Scale Types (cont)



scale_datetime(),
scale_lo- gical(),
scale_nominal(),
scale_numeric(),
scale_sin- gular()
Code faithful %>%ggvis(~eruptions,~wai- ting, fill = ~eruptions) %>% layer_points() %>%scale_numeric("fill", range)

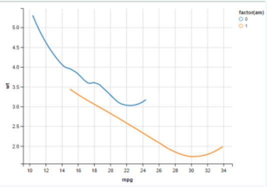
More about ggvis

1.Differences and similarities to ggplot2.
2.The relationship between ggvis and Vega

Popular In-Built plot types

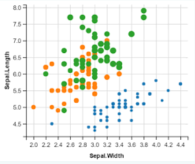
1. layer_points()
2. layer_lines()
3. layer_bars()
4. layer_smooths()
5. layer_histograms()

Multiple Layer



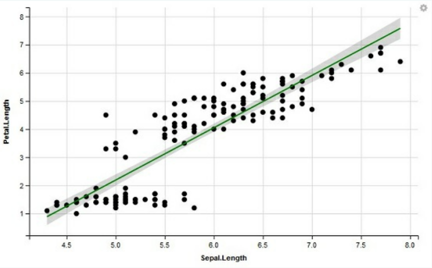
mtcars %>% ggvis(~wt,~mpg) %>% layer_smooths(span= 1) %>%layer_smooths(span = 0.3, stroke := "- red")

ggvis & interaction ()



train_tbl %>% group_by(season,holiday) %>% ggvis(~count, fill = ~inter- action(season,holiday)) %>%

Interactive Plots



ggivs comes several

input_checkbox(), input_checkboxgroup
input_numeric(), input_radiobuttons(),
input_select(), input_slider(), and inp

label = "ABCD " , cho black") -
value = "black" - Use text()
map = as.name used to return variable nam

Are the common argu these functions.

The goal is to combine the best of R (e.g. every modelling function you can imagine) and the best of the web (everyone has a web browser). Data manipulation and transformation are done in R, and the graphics are rendered in a web browser, using Vega. For RStudio users, ggvis graphics display in a viewer panel, which is possible because RStudio is a web browser.