How to Visualize Data With Highcharter

INTRODUCTION

Highcharter is a R wrapper for Highcharts javascript library and its modules. Highcharts is very mature and flexible javascript charting library and it has a great and powerful API.

The main features of this package are:

Various chart type with the same style: scatters, bubble, line, time series, heatmaps, treemap, bar charts, networks.

Chart various R object with one function. With hchart(x) you can chart: data.frames, numeric, histogram, character, density, factors, ts, mts, xts, stl, ohlc, acf, forecast, mforecast, ets, igraph, dist, dendrogram, phylo, survfit classes.

Support Highstock charts. You can create a candlestick charts in 2 lines of code. Support xts objects from the quantmod package.

Support Highmaps charts. It’s easy to create choroplethts or add information in geojson format.

Piping styling.

Themes: you configure your chart in multiples ways. There are implemented themes like economist, financial times, google, 538 among others.
Plugins: motion, drag points, fontawesome, url-pattern, annotations.

**Installation**

install.packages("highcharter")
library(highcharter)

**Basic Example**

This is a simple example using hchart function.

library("highcharter")
data(diamonds, mpg, package = "ggplot2")

hchart(mpg, "scatter", hcaes(x = displ, y = hwy, group = class))

Learn more about using different visualization packages in the online course [R: Complete Data Visualization Solutions](#). In this course, you will learn how to:

- Work extensively with the ggplot package and its functionality
- Learn what visualizations exist for your specific use case
- And much more

**The highcharts API**

highchart() %>%
hc_chart(type = "column") %>%
hc_title(text = "A highcharter chart") %>%
hc_xAxis(categories = 2012:2016) %>%
hc_add_series(data = c(3900, 4200, 5700, 8500, 11900),
name = "Downloads")

Generic Function hchart
Among its features highcharter can chart various objects depending of its class with the generic `hchart` function.

```r
hchart(diamonds$cut, colorByPoint = TRUE, name = "Cut")
```

**Zoom**

```r
hchart(diamonds$price, color = "#B71C1C", name = "Price") %>%
hc_title(text = "You can zoom me")
```

**Forecasting**

One of the nicest class which hchart can plot is the forecast class from the forecast package.

```r
library("forecast")

airforecast <- forecast(auto.arima(AirPassengers), level = 95)
hchart(airforecast)
```

**Highstock**

With highcharter you can use the highstock library which include sophisticated navigation options like a small navigator series, preset date ranges, date picker, scrolling and panning. With highcarter it’s easy make candlesticks or ohlc charts using time series data. For example data from quantmod package.

```r
library("quantmod")

x <- getSymbols("GOOG", auto.assign = FALSE)
y %>%
hc_add_series(x) %>%
hc_add_series(y, type = "ohlc")
```

**Highmaps**

You can chart maps and choropleth using the highmaps module.

```r
data(unemployment)
```
hcmap("countries/us/us-all-all", data = unemployment, 
name = "Unemployment", value = "value", joinBy = c("hc-key", 
"code"),
borderColor = "transparent") %>%
hc_colorAxis(dataClasses = color_classes(c(seq(0, 10, by = 2), 
50))) %>%
hc_legend(layout = "vertical", align = "right", 
floating = TRUE, valueDecimals = 0, valueSuffix = "%")

Now, let’s move on to the first set of real exercises on the leaflet package!