NumFocus - Julia Google Summer of Code proposal: Sputink, a modern tool for data exploration based on JuliaDB and WebIO

Pietro Vertechi, mentored by Shashi Gowda

March 21, 2018

Introduction

The recent JuliaDB package implements effective data wrangling algorithms on large datasets, potentially stored across different processors.

The package is complemented by a set of plotting recipes based on OnlineStats.jl as well as a macro from StatPlots.jl to simplify statistical visualizations of the data: `http://juliadb.org/latest/api/plotting.html`.

I plan to build a web app, tentatively called Sputnik.jl, to allow users to access algorithms from JuliaDB, OnlineStats and StatPlots (also incorporating some of my prior work - GroupedErrors.jl for analysis of population data) from a friendly user interface.

While a web app will never grant the same flexibility as coding a Julia script, I believe it has the following two advantages:

- It is more inviting for users not very comfortable with coding.
- It simplifies completely exploratory data analysis on a dataset with a large number of columns where doing all plots by hand would be too time consuming.

I intend to integrate ideas from my previous experience building a QML-based GUI for data visualization: PlugAndPlot.jl. Despite having enjoyed the flexibility and features of QML while developing PlugAndPlot, this time I'd prefer to focus on a web app (based on WebIO) as it can be easily deployed:

- On the plot pane in Juno (a popular Julia IDE)
- In a Jupyter notebook
- In an electron window
- Served in the browser

If the time allows it, I'd like to investigate whether it is feasible - for users whose data is stored on a server - to deploy such app from the server and analyze the data remotely. In that way, researchers who are willing to open-source their data could quickly set up a website where everybody can consult their data interactively (in my view, this is an excellent way to accompany a publication where only some analysis of the data are accessible).

Plan

As a first step, I inted to port PlugAndPlot.jl from QML to WebIO. This will involve adding some functionality to the WebIO, InteractNext, CSSUtils stack as not all widgets and features of QML are implemented there yet. It will also be a learning opportunity for me as, despite having some experience with traditional GUI toolkits (Gtk and QML), I'm not as familiar with the recently developed WebIO stack. I will be under @shashi mentoring who is one of the main developer of the WebIO stack and will help me familiarize myself with this software.

As a second step, I intend to optimize the analytical core of PlugAndPlot, GroupedErrors.jl, a package which accepts any table that can iterate data, in the case where the input data is a JuliaDB table. This should be possible without sacrificing the "iterator based interface", once a set of PRs on which I'm working to collect iterators as a set of columns efficiently are merged: see:
`https://github.com/JuliaComputing/IndexedTables.jl/pull/137` and
`https://github.com/JuliaComputing/IndexedTables.jl/pull/135`.

As a third step, I intend to rethink the UI design, adding features specific to JuliaDB (such as the powerful set of online statistical analysis and visualizations - or the TableView.jl package, also WebIO based, to visualize the data in a spreadsheet format), incorporating feedback from the JuliaDB developers. To maintain the flexibility of working with a script, I intend to add a textbox where users can type in calls to functions from JuliaDB or JuliaDBMeta to do some preprocessing on their data before visualizing it.

Throughout this process, I will try, as much as possible, to keep the components of the app modularized so that it would also be possible, for a user, to recombine these components to build GUIs with a different design or calling different algorithms and visualization techniques in the background.

Finally, if time permits, I will investigate whether it is feasible to deploy this app from a server where the data is stored, thus simplifying interactive visualizations of shared data.

About me

Even though my background (bachelor and master) is in mathematics, I'm currently enrolled in a PhD in neuroscience.