

Spyre: Exploratory Data Analysis in the Browser

Erik Iverson^{1,*}

1. MEI Research

*Contact author: erik@sigmafield.org

Keywords: EDA, browser, JSON, websockets, data analysis

Many exploratory data analysis (EDA) tools involve the use of a GUI that exposes the analyst to a very limited set of options for viewing and summarizing data. Alternatively, interactive languages such as *R* offer an incredibly flexible environment to facilitate data exploration, at the expense of having to repeat basic operations such as bivariate plots and tabular summaries repeatedly to gain deeper insight into the data.

Two popular *R* IDEs, Emacs ESS and RStudio, have incorporated useful shortcuts to help analysts with the mechanics of EDA. For example, the user can view which *R* objects have been created in the current workspace, create plots of those objects where appropriate, and inspect the structure and contents of objects. While useful, these features are IDE-specific, and neither flexible nor extensible.

Increasingly, the web browser is a platform not just for information retrieval and display, but for interactive applications. External processes such as *R* can communicate with modern web browsers through standards like websockets. The **spyre** package is a hybrid of the two EDA approaches above. It allows an *R* user to inspect data through the familiar interactive command-line environment, while *simultaneously* offering browser-based GUI features to facilitate common tasks when exploring data. Spyre brings R-based EDA to the browser.

Spyre communicates with a running *R* process through the use of the **websockets** package, and does *not* block the *R* process while running. In other words, the user is still able to use *R* interactively through the command line while Spyre is running. As new objects are created in the *R* process, Spyre becomes aware of them and allows the user to view basic information about the objects, plot them, and view customized summaries. Because the underlying communications are done through websockets, Spyre can be used with any *R* IDE.

In the figure below, a basic Spyre session is shown. A list of currently available objects is displayed. When clicked, a summary function based on that object's class, in this case a factor, is called. Summary data is sent to the browser via JSON. Spyre's modular design allows an analyst to easily override or augment any of the information sent between the *R* process and Spyre, using the **jsonlite** package. This allows the summaries to be displayed in simple text form, or more advanced actions to be taken, such as using the d3.js library to draw graphical summaries of the data. The wide availability of JavaScript libraries opens up many possibilities for interesting applications using Spyre.

The next extensions planned for **spyre** include a data.frame explorer and an interactive regression tool.

