An R Case Study: From Research to Production

Putting a statistical algorithm in a production environment

Aidan Boland

Clavis Insight, Dublin, Ireland
Email: aidan.boland@clavisinsight.com

Abstract: In industry, bringing an algorithm from it's research phase into a usable product can be an awkward hurdle to cross. During research the algorithm may be coded in one language, but in practice it may need to be used within a different language environment. It's often not worthwhile re-writing the function.
This case study discusses the methods used to bring a statistical algorithm written in R from it's research phase, into daily use within a production environment. The algorithm in question is a supervised classification method and is used to automatically categorise e-commerce products from online stores.
R's ecosystem makes it very simple to provide access to code without the end user needing any knowledge of R. These same methods can be used in academia to open up quick and easy access to new statistical methods.

Introduction

R is one of the most popular languages in statistics. However, in industry R is not often used within a production environment. Bringing research from locally run R scripts to production code can be a long and arduous process, especially if the code must be ported into a different language to match the companies standard.
Alternatively, if the R code is already robust, it can be made accessible through Graphical User Interfaces (GUI) and Application Programming Interfaces (API). The ecosystem of R libraries makes its extremely simple and straightforward to create these accessible methods.

Case Study

Clavis Insight are a leading firm in e-commerce analytics. Large amounts of data is processed from online stores across the globe every day; this allows clients to monitor their performance in the online marketplace. One issue faced by the company is to categorise products into groups which reflect a clients individual specification.
An algorithm based on multinomial logistic regression was created to automatically classify new products into their relevant categories. The research was completed

and coded using R. Bringing the algorithm from research into production was done in 2 steps.
First, during a testing period a GUI was created which allowed users to load and classify products. The user could then download the categorised data.
After testing, an API was created which allowed the algorithm to be run directly from within the production code. This allowed the classification of new data without any manual intervention.

Methods

GUI (shiny)

A GUI gives users simple point and click access to algorithms. The shiny library in R makes it very easy to create a GUI. Users can easily change parameters and results can be displayed graphically on screen.

API (plumber)

An API is an endpoint which can be called from almost any computer language. Data and parameters can be passed to the API, and results can be returned. Creating an API for code is the most versatile way of allowing access to methods. The plumber library in R is a simple way to create API's for R functions.

Discussion

There are many easy ways in R to provide users with access to algorithms without the user needing any knowledge of R. These methods provide a quick solution for bringing an algorithm from research into production use.
A related problem in academia is providing simple access to new and novel algorithms. While publishing raw code is one method of sharing work, this assumes that the end user will be able to run the code on their own machine. Making code accessible through a GUI allows users simple access to the methods. While creating an API allows users to build the methods into their own code without needing to fully understand the original code.

References

Trestle Technology, LLC (2017). Plumber: An API Generator for R.

Chang W., Cheng J., Allaire JJ., Xie Y. and McPherson J. (2017). Shiny: Web Application Framework for R.