

EVALUACIÓN DSBI

(SHINY)

Raúl Varela Ferrando

07/07/2023

Este trabajo está dividido en tres partes, la primera, que cuenta con hasta 7 puntos del total y consiste en el desarrollo de una aplicación shiny a partir del modelo que consta en la enseñanza virtual (5 puntos), además de rediseñarla (2 puntos adicionales en función de la complejidad). En mi caso, a parte de diseñar todos los apartados de la app de ejemplo, he rediseñado algunos aspectos, como cambiar el color de la página, añadir un apartado de regresión múltiple dentro del estudio descriptivo, y, por último, también he añadido otro apartado en la página principal, que a partir del dataset *mpg* del paquete *ggplot2*, podemos consultar los modelos de coches y sus características dentro de ciertos intervalos descritos a través de los distintos *inputs* definidos en la interfaz.

Por otra parte, los 3 puntos restantes, a su vez, se dividen en otras dos partes de 1.5 puntos cada una. La primera es la entrega de una aplicación shiny obtenida de internet comentando sus características principales. En mi caso, he elegido una aplicación que he obtenido a través del siguiente enlace https://github.com/flor14/rladies_shiny_meetup_2020. No es una aplicación demasiado compleja, en ella se nos muestra una nube de puntos bidimensional del mismo dataset que utilicé anteriormente, *mpg*, donde podemos elegir qué magnitudes se representan en los ejes (sólo numéricas), y a su vez, los puntos se representan con distintos colores dependiendo de la clase a la que pertenezca cada caso, teniendo como posibilidades las distintas variables categóricas del dataset.

En la parte *ui* podemos observar que hace uso de la función *fluidPage()* donde se crea toda la interfaz de la aplicación. Primero nos encontramos el título junto al subtítulo, seguido de un panel lateral creado con las funciones *sidebarLayout()* y *sidebarPanel()* anidadas. En este panel tenemos las distintas entradas, el eje X y el eje Y, en los cuáles podemos elegir entre las magnitudes *hwy*, *cty* o *displ*, y la representación por clases (Class), cuyas opciones son *class*, *cyl*, *model* y *manufacturer*. Por último, encontramos una imagen seguida del gráfico en cuestión del que hablé anteriormente.

En la parte *server* solo se detalla el gráfico de nube de puntos haciendo uso de la función *renderPlot()* y del paquete *ggplot2* donde se le asigna un color diferente a cada punto dependiendo de la clase a la que pertenece.

Por último, nos piden que hablemos de un paquete R que extienda las posibilidades de Shiny, además de añadir un ejemplo de aplicación que haga uso de este. En mi caso, voy a hablar del paquete *shinycssloaders*, cuya función principal es añadir animaciones a las salidas de nuestra aplicación mientras estas se están recalculando, como sería el caso de las regresiones o los gráficos que hemos realizado. Una aplicación que hace uso de este paquete es la que se encuentra en el siguiente enlace <https://daattali.com/shiny/shinycssloaders-demo/> , en caso de que no funcione puede visitarse este otro enlace <https://deanattali.com/blog/shinycssloaders-v1.0/> y clicar en el botón *Check out a demo* que aparece más abajo. Como se puede observar, en esta aplicación podemos elegir tanto el tipo de *Spinner* (animación elegida), el tamaño y el tiempo que se muestra en pantalla antes de actualizar la gráfica, la cuál representa una nube de puntos donde el eje X es el índice de los valores, y el eje Y el valor asociado siguiendo una distribución uniforme.