

Ejercicio 1 de Shiny

Recomendaciones

Se recomienda que a medida que se desarrolle el ejercicio se hagan copias del proyecto como hemos hecho con los pasos del ejercicio realizado en clase. De esta manera si hay algún error catastrófico que impide que continuéis con el ejercicio podéis retomarlo desde la última copia. Recordad que si los ficheros los llamáis ui2 o server_copia o cualquier otra cosa que no sea ui.R o server.R no funcionará.

A pesar de esta recomendación **sólo se debe entregar la última copia con todo el trabajo hecho no todos los pasos anteriores**. Las copias de seguridad son para facilitar la tarea NO se deben adjuntar.

Todo lo que no esté especificado en este documento como textos, presentación visual, valores por defecto de los inputs, nombres, etc no tendrá impacto en la evaluación y será decisión del propio alumno qué hacer en esos casos.

Criterios de evaluación

- Funcionalidad (70%): el ejercicio cumple con todas las características descritas
- Limpieza del código (10%): el código está correctamente sangrado y ordenado además de que sea fácilmente legible (usar nombres de los parámetros, separar las líneas donde facilite la legibilidad, nombres entendibles...)
- Comentarios (20%): el código está comentado explicando lo que se ha realizado y las distintas partes del código con tus propias palabras.

Entrega el sábado 14 a las 23:59

Descripción del ejercicio

Vamos a realizar una aplicación que permita la visualización del dataset mpg. El usuario podrá elegir un par de variables para analizar su relación bivariada (diagrama de dispersión y correlación de Pearson) y sus distribuciones univariadas (histograma y summary). Además, existirá un campo numérico que permita al usuario seleccionar una submuestra aleatoria del dataset.

Realizar una aplicación shiny que usa sidebarLayout en la que aparezca:

- Dos selectores que permitan elegir cada uno una variable del dataset mpg (ver ayuda de mpg para saber qué variables contiene).
 - o Estos selectores deben mostrar al usuario un nombre de variable inteligible que describa la variable y no el nombre de la variable del dataset pero internamente su valor sí debe ser el nombre de la variable.

Por ejemplo "cyl" no debería mostrarse al usuario sino "number of cylinders" o "número de cilindros". Pero "cyl" si debería ser el valor del input para luego utilizarlo en los outputs correctamente.
- Campo numérico para elegir el tamaño de la submuestra a realizar. Su mínimo es 2 y su valor máximo es el número de registros del dataset. El valor por defecto debe ser 10.
- Un botón para accionar el análisis y que los resultados se actualicen

La aplicación tiene como objetivo:

- Realizar una muestra con la función “sample” de R sobre mpg. Es decir, no vamos a coger todos los registros sino una submuestra de ellos de tamaño elegido por el usuario (campo numérico)
- Esta muestra será analizada de varias maneras:
 - Dos histogramas (función hist) y dos summaries de cada variable por separado
 - Un diagrama de dispersión (con ggplot) con las dos variables, en el eje X e Y (indistintamente)
 - Mostrar la correlación de pearson entre ambas variables (función cor)
- Dado que los análisis requieren variables numéricas eliminaremos de las opciones para elegir las variables del dataset que no son numéricas como manufacturer.

Tened en cuenta que:

- Sólo el botón actualiza los cálculos, ninguna otra interacción debe activar la actualización
- Todos los análisis (distribuciones, summary, correlación, etc) deben realizarse exactamente con la misma muestra. Es decir, todos los outputs deben tomar la misma fuente para que todos los cálculos sean consistentes.

Recordar que al hacer un sample el proceso es aleatorio, es decir, cada muestra será distinta cada vez que la calculemos y por ello debemos garantizar la consistencia entre todos los cálculos.

Ayudas

Hasta ahora las variables que utilizábamos eran fijas ahora son programáticas. Es decir, las eliges desde el programa (el usuario puede cambiar las variables de análisis).

Para realizar esto os facilito dos pequeños comentarios de cómo hacer esto con éxito:

Gráficas de ggplot2 programático

Cuando no sabes qué variables vas a usar no puedes usar el `aes()` normal como solíamos hacer sino `aes_string()` que permite introducir las variables como cadenas de texto (nuestros inputs 😊) Por ejemplo:

```
library(ggplot2)
# A veces no conozco las variables a priori sino que un tercero
# (p.e.: usuario del shiny) me dice cuales deben ser

variableX <- "cyl"
variableY <- "cty"

ggplot(mpg, aes_string(x=variableX, y=variableY)) +
  geom_point()

# Fijaos en la diferencia:
ggplot(mpg, aes(x=cyl, y=cty)) +
  geom_point()

# VS aes_string que ahora necesita que sean cadenas de texto
# (por eso las comillas)
ggplot(mpg, aes_string(x="cyl", y="cty")) +
  geom_point()

# Estas cadenas de texto pueden ser variables:
# (por eso ya no tienen comillas)
ggplot(mpg, aes_string(x=variableX, y=variableY)) +
  geom_point()
```

Cálculos (p.e. correlación de Pearson) programáticos

Esto es más fácil, simplemente en vez de usar el `$` hay que usar el doble corchete `[[]]`

```
# Son lo mismo. Para acceder a una variable programáticamente
# en vez de usar $ hay que usar el doble corchete:
mpg$cyl
mpg[[variableX]]
```

Tened en cuenta que `variableX` es una variable de texto que contiene el nombre de otra variable. Es decir, `cyl` es una variable del dataset `mpg` pero `variableX` es una variable que contiene "cyl". No es lo mismo.

El matiz es que `cyl` lo puedes usar en `aes(x=cyl)` pero `variableX` no funcionaría con `aes(x=variableX)` porque `variableX` NO existe en el dataset. De igual manera `mpg$variableX` fallaría.

Por eso hay que usar `aes_string` y `[[]]` para que R “lea” el texto en vez de usar literalmente el nombre de la variable que escribes.

Referencias para el ejercicio

- Cómo realizar una submuestra con la función `sample`
 - o <https://stackoverflow.com/questions/18385099/random-subsampling-in-r>
 - o <https://www.rdocumentation.org/packages/base/versions/3.6.2/topics/sample>
- GGplot con variables programáticas
 - o https://www.rdocumentation.org/packages/ggplot2/versions/1.0.0/topics/aes_string