

Taller de R: Estadística y Programación

Taller 4

24/11/2023

En este taller se evalúan los temas vistos en las clases 11 a 15 del curso. Lea atentamente las instrucciones del taller.

Instrucciones

- Este taller representa el **33.3%** de la nota total del curso y podrá responderlo de manera individual o en grupo de hasta 3 personas. En las primeras líneas del script debe escribir su nombre, código y la versión de R sobre la que está trabajando. Además, al inicio del código debe llamar/installar las librerías que va a usar en la sesión. Por ejemplo: `pacman`, `rio`, `tidyverse`, `sf`, `leaflet`, `rvest`, `xml2`, `osmdata` y `ggsn` (a lo menos).
- Asegúrate de descargar las bases de datos del repositorio <https://github.com/taller-r-202302/problem-sets> y crear un nuevo repositorio en tu cuenta de GitHub. Si va a trabajar en grupo, solo 1 de los integrantes del grupo debe crear el repositorio y compartir el acceso a los demás integrantes. El repositorio debe ser público para que se pueda acceder desde cualquier cuenta de GitHub. Adicionalmente, este repositorio debe incluir a lo menos tres carpetas: **input** (datos originales), **output** (datos procesados) y **code** (script con la respuesta del taller).
- Todos los integrantes del grupo deberán colgar el link al repositorio de GitHub en la actividad **Problem-set-4** de Bloque Neón antes de las 23:59 horas del 8 de diciembre de 2023.
- Por favor sea lo más organizado posible y comente paso a paso cada línea de código, pero recuerden **NO** usar ningún acento o carácter especial dentro del código para evitar problemas al abrir los scripts en los diferentes sistemas operativos.
- No seguir las instrucciones tiene una penalización del **20%** de la nota final.

Problem set 4

1. Regresiones (30%)

En la carpeta **input** se encuentra el archivo `data_regresiones.rds`. Este archivo contiene información de las características físicas (baños, cuartos, área total del inmueble...) y de geolocalización (distancia al centro de negocios de la ciudad -CBD-, colegios y parques) para 54.577 inmuebles en venta en Bogotá.

- **1.1** Estimaciones

Use el conjunto de datos `input/data_regresiones.rds` para estimar tres modelos econométricos en los que la variable dependiente sea el precio de la vivienda. Cada modelo debe incluir a lo menos una variable diferente.

- **1.2** Presentar resultados

Combine en una sola tabla los coeficientes de los tres modelos estimados. Usando esta tabla, genere **un único** gráfico (`coefplot`) en el que presente los resultados de los tres modelos.

- **1.3** Exportar resultados

Exporte la tabla como `output/resultados_regresiones.xlsx` y el gráfico como `output/plot_regresiones.png`

2. Datos espaciales (35%)

- **2.1** Descargar datos.

Seleccione una ciudad de Colombia y usando la librería que se conecta a OpenStreetMaps descargue los restaurantes (puntos) y parques (polígonos) para esa ciudad.

- **2.2** Visualizaciones

Use la función `leaflet` para visualizar la información descargada en el punto anterior.

- **2.3** Geocodificar direcciones

Use la función `geocode_OSM()` para geocodificar una dirección (la que usted elija) dentro la ciudad que seleccionó en el punto **1.1**.

- **2.4** Exportar mapa

Use la función `ggplot` y pinte en un solo mapa los restaurantes, parques y la dirección que geocodificó (sí desea agregue la capa de vías usando la función `get_stamenmap`). Adicione la barra de escalas, la estrella del norte y un `theme` para mejorar la apariencia del mapa. Exporte este mapa en un archivo `output/mapa_amenities.png`

3. Web-scraping y procesamiento de texto (35%)

- **3.1** Desde la consola de Rstudio lea la siguiente url <https://es.wikipedia.org/wiki> y cree un objeto que contenga el HTML de la página como un objeto `xml_document`.
- **3.2** Use el `xpath` para extraer el título de la página (Departamentos de Colombia).
- **3.3** Extraiga la tabla que contiene los departamentos de Colombia y exporte el objeto en un archivo `output/tabla_departamento.xlsx`
- **3.4** Extraiga los parrafos del documento (elementos con etiqueta `p`) y genere una nube de palabras. Exporte el objeto en un archivo `output/nube_palabras.png`