Ejercicios sesión 1

Alberto Torres Barrán

2020-01-29

mtcars

Con el data frame mtcars (viene cargado en R):

1. Previsualizar el contenido con la función head().
2. Mirar el número de filas y columnas con nrow() y ncol().
3. Crear un nuevo data frame con los modelos de coche que consumen menos de 15 millas/galón.
4. Ordenar el data frame anterior por disp.
5. Calcular la media de las marchas (gear) de los modelos del data frame anterior.
6. Hacer una gráfica del peso (wt) con respecto al consumo (mpg).

Lahman

1. Instalar el paquete Lahman y ver que data.frames contiene (en la documentación).
2. Usando los datos contenidos en el data.frame Batting, calcular el número de partidos totales de cada uno de los jugadores.
3. Ordenar a los jugadores de mayor a menor de acuerdo a la cantidad de partidos disputados.
4. Mostrar el top 5.

diamonds (1)

Con el conjunto de datos de diamantes, realizar las siguientes operaciones:

1. Explorar el contenido de los datos: número de filas, columnas y tipo de las columnas (numéricas, texto, etc.)
2. Filtrar los diamantes con corte “Ideal”.
3. Seleccionar las columnas carat, cut, color, price y clarity.
4. Crear una nueva columna precio/quilate.
5. Agrupar los diamantes por color.
6. Calcular la media del precio/quilate para cada uno de los grupos anteriores.
7. Ordenar por precio/quilate de forma descendente.

diamonds (2)

1. Ver cuantas filas (diamantes) y columnas (variables) tiene el conjunto de datos.
2. Hacer un gráfico de barras con la cantidad de diamantes que hay para cada corte (variable cut).
3. Escoger aleatoriamente 10000 diamantes (función `sample_n()`)

La correlación mide la fuerza de una relación lineal entre dos variables. Toma valores entre 0 y ± 1, donde 0 es poca dependencia y 1 máxima dependencia (el signo indica la dirección). En R se calcula con la función `cor()`. Sabiendo lo anterior y sobre la muestra reducida de 10000 diamantes:

4. Hacer un histograma para visualizar la distribución de los precios.
5. Calcular la correlación entre las variables precio y quilates (carat)
6. Visualizar dicha correlación haciendo un gráfico de dispersión del precio sobre los quilates.
7. Repetir el gráfico anterior pero para cada uno de los valores del corte

nycflights13 (1)

Con el data frame flights (paquete nycflights13) vamos a intentar ver si el retraso medio depende o no de la hora del día:

1. Crear una nueva variable time a partir de las variables hour y minute que represente la hora y minutos como un número con un decimal.
2. Calcular el retraso medio a la llegada (arr delay) y el número de vuelos para cada uno de los valores de la variable time.
3. Guardar las tres variables anteriores en un nuevo data.frame con nombre delay.per.time.
4. Representar el retraso medio con respecto a la variable time, escalando además el tama˜ no de los puntos de acuerdo con el número de vuelos.

nycflights13 (2)

Con el data frame flights (paquete nycflights13), vamos a intentar ver si hay grandes diferencias en cuanto a retrasos de la llegada dependiendo del aeropuerto destino:

1. Calcular el retraso medio a la llegada (arr delay) y el número de vuelos para cada uno de los destinos (variable dest).
2. Hacer un merge del data.frame anterior con airports (contenido en el mismo paquete) para añadir las coordenadas de cada aeropuerto. Pista: función left join.
3. Representar la latitud con respecto a la longitud, escalando además el tama˜ no de los puntos de acuerdo con el número de vuelos.