Universidad Rey Juan Carlos

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

# Inferencia Estadística

## Ejercicios de la Asignatura

Grado en Ciencia e Ingeniería de Datos

AUTORES

- Víctor Aceña Gil
- Isaac Martín de Diego
- Carmen Lancho Martín

2025-2026

# Índice de Ejercicios

# Introducción

Este documento recopila una serie de ejercicios prácticos y teóricos diseñados para complementar la asignatura "Inferencia Estadística" del Grado en Ciencia e Ingeniería de datos. El objetivo de esta colección es afianzar los conocimientos adquiridos en cada tema, fomentando tanto la comprensión de los fundamentos teóricos como la habilidad para implementar y diagnosticar modelos en R.

## Estructura de los Ejercicios

Los ejercicios están organizados por temas, siguiendo la estructura del curso. Para cada tema, encontrarás una mezcla de:

- **Preguntas Conceptuales:** Diseñadas para reforzar la comprensión de la teoría subyacente.
- **Problemas Prácticos con R:** Enfocados en la aplicación de las técnicas a conjuntos de datos reales o simulados.
- **Ejercicios de Interpretación:** Centrados en la habilidad crítica de interpretar correctamente las salidas de los modelos estadísticos.

## Requisitos Previos

Para abordar estos ejercicios, se asume que el estudiante ha estudiado el contenido teórico del tema correspondiente y posee un manejo básico del entorno de programación R y RStudio.

# Introducción a la Inferencia Estadística

### Ejercicio 1

**Pregunta**: Describe en tus propias palabras qué es la Ciencia de Datos y su importancia en el análisis de grandes volúmenes de datos.

### Ejercicio 2

**Pregunta**: Enumera las herramientas estadísticas que se utilizan en la inferencia estadística y explica brevemente su propósito.

### Ejercicio 3

**Pregunta**: Define los términos "población" y "muestra" y explica la diferencia entre ambos.

### Ejercicio 4

**Pregunta**: ¿Qué es una distribución de probabilidad y cómo se relaciona con las variables cualitativas y cuantitativas?

### Ejercicio 5

**Pregunta**: Realiza un resumen descriptivo de un conjunto de datos utilizando medidas de tendencia central y dispersión.

### Ejercicio 6

**Pregunta**: Explica la diferencia entre estadística descriptiva e inferencial y proporciona un ejemplo de cómo se utiliza cada una.

## Ejercicio 7

**Pregunta**: Diseña un experimento para ilustrar cómo el muestreo aleatorio simple puede ser utilizado para estimar una característica de una población.

## Ejercicio 8

**Pregunta**: Explica el Teorema Central del Límite y su relevancia en la inferencia estadística.

## Ejercicio 9

**Pregunta**: Compara y contrasta la estadística paramétrica y no paramétrica, dando ejemplos de cuándo se utilizaría cada una.

## Ejercicio 10

**Pregunta**: Discute las diferencias entre los enfoques frecuentista y bayesiano en la inferencia estadística y da un ejemplo de aplicación para cada uno.

# Análisis Exploratorio de Datos

Utilizar el dataframe denominado *bank* con el que hemos trabajo en el tema 2 de la asignatura para responder a las siguientes cuestiones:

### Ejercicio 1

**Estructura del DataFrame**: - ¿Cuántos campos y observaciones tiene el dataframe `bank`? Utiliza las funciones `head` y `dim`.

### Ejercicio 2

**Resumen del DataFrame**: - Evalúa el dataframe con la función `summary`. - ¿Tiene observaciones con elementos faltantes (NA)? - ¿A qué categorías corresponden las observaciones en la variable `job`?

### Ejercicio 3

**Distribución de la Edad**: - ¿Cuál es la edad máxima y mínima de los clientes en el dataframe? - ¿Cuál es la media y la mediana de la edad de los clientes?

### Ejercicio 4

**Balance Promedio**: - ¿Cuál es el balance promedio anual (balance) de los clientes? - ¿Cuál es el balance promedio anual de los clientes que han suscrito un depósito a plazo fijo (`y` = "yes")?

### Ejercicio 5

**Frecuencia de Contacto**: - ¿Cuál es el número máximo y mínimo de contactos realizados durante esta campaña (`campaign`)?

## Ejercicio 6

**Análisis de Duración**: - ¿Cuál es la duración media y mediana del último contacto en segundos (`duration`)? - ¿Cuál es la duración media del último contacto en segundos para los clientes que suscribieron un depósito a plazo fijo (`y` = "yes")?

## Ejercicio 7

**Relación entre Variables**: - ¿Existe alguna relación entre el balance promedio anual y la duración del último contacto? Utiliza una visualización adecuada para responder a esta pregunta.

## Ejercicio 8

**Segmentación por Trabajo**: - ¿Cuál es la media y mediana del balance anual de los clientes agrupados por tipo de trabajo (`job`)?

## Ejercicio 9

**Análisis de Contactos Anteriores**: - ¿Cuál es el número máximo y mínimo de días que pasaron desde el último contacto de una campaña anterior (`pdays`)? - ¿Cuál es la media de `pdays` para los clientes que suscribieron un depósito a plazo fijo (`y` = "yes")?

## Ejercicio 10

**Estudio General**: - Haciendo un estudio general de los datos, ¿qué puedes concluir? ¿Existe alguna relación significativa entre las variables `balance`, `duration`, y `campaign`? Se recomienda hacer un análisis visual y estadístico de estas variables.

# Procesado y Limpieza de Datos

### Ejercicio 1

Crear el siguiente *dataframe* mediante estas instrucciones

```
set.seed(1234)

stocks = data.frame(time = as.Date("2009-01-01") + 0:9,
                    Wallmart = rnorm(10,20,1),
                    Target = rnorm(10,20,2),
                    Walgreens = rnorm (10,20,4)
                    )
stocks
```

```
         time Wallmart   Target Walgreens
1  2009-01-01 18.79293 19.04561  20.53635
2  2009-01-02 20.27743 18.00323  18.03726
3  2009-01-03 21.08444 18.44749  18.23781
4  2009-01-04 17.65430 20.12892  21.83836
5  2009-01-05 20.42912 21.91899  17.22512
6  2009-01-06 20.50606 19.77943  14.20718
7  2009-01-07 19.42526 18.97798  22.29902
8  2009-01-08 19.45337 18.17761  15.90538
9  2009-01-09 19.43555 18.32566  19.93945
10 2009-01-10 19.10996 24.83167  16.25621
```

A continuación, realizar las siguientes operaciones de limpieza de datos:

- Como se puede observar, hay un problema de clave-valor en las compañías con sus observaciones. Por lo tanto, se pide transformar los datos para que tengan una clave "*stock*" y un valor "*precio*". Utilizar la instrucción "*gather*".
- Devolver el dataframe al estado original empleando la instrucción *spread*.
- Utilizando el operador tubería %>% se desea realizar las siguientes operaciones anidadas:
  - Transformar los datos para que tengan una clave "*stock*" y el calor sea el "*precio*". Utilizar la instrucción "*gather*".

- Agrupar los datos por la clave "*stock*" mediante la instrucción "*group_by*".
- Obtener el precio mínimo y el máximo utilizando la instrucción "*summarise*".

## Ejercicio 2

En este ejercicio vamos a manejar datos contenidos en distintos *dataframes* y operar sobre ellos con *dplyr*.

1. Descargar el paquete *nycflights13*.
2. Evaluar el contenido de los dataframes proporcionados por el paquete. Utilizar *head* y *summary*.
3. Simplificar los dataframes originales a 100 observaciones mediante el comando *head*. Asignarlos a una variable que indique el tipo de dataframe añadiendo la coletilla "*_simple". *Ejemplo:"*flights_simple*".
4. Selecciona los tipos de aerolínea ("*carrier*") mediante la instrucción *select* y el operador *unique* concatenados con el operador tubería %>%. (Utilizar "*airlines_simple*").
5. Obtener la media y el número máximo de asientos ("*seats*") que tienen los aviones. Utilizar el operador tubería %>% y la instrucción *summarise*.
6. Ordenar los aviones por su número de motores ("*engines*") y número de asientos ("*seats*"). Utilizar la instrucción *arrange*.
7. Averigua qué número de cola ("*tailnum*") comparten los dataframes "*flights_simple*" y "*planes_simple*" que has creado anteriormente. Obten su aerolínea ("*carrier*"). Utilizar la instrucción *inner_join*.
8. Cruzar los datos de vuelos ("*flights*") con los aviones ("*planes*") por el número de cola ("*tailnum*") que no coincidan (usar la instrucción *anti_join*). De esos obtener aquellos con 2 o más motores(usar la instrucción *filter*). Finalmente obtener los distintos modelos de avión que satisfacen las premisas anteriores (usar la instrucción *unique*).
9. Crea una nueva variable ("*total_delay*") que calcule el retraso total sumando los delays acumulados ("*dep_delay*") y ("*arr_delay*"). Utilizar la instrucción *mutate*. Almacena el dataframe resultante en "*flights_total*".
10. En base a la variable anteriormente obtenida ("*total_delay*"), devuelve los aviones que han llegado con antelación a su destino, es decir aquellos tal que la variable *total_delay* tiene valores negativos.

# Proyecto City Bike NYC

## Introducción City Bike NYC

El sistema de uso compartido de bicicletas en la ciudad de Nueva York (EE.UU.) publica diariamente gran cantidad de datos de actividad sobre su uso.

Estos datos han dado lugar, como no, a algunos análisis sobre la evolución de este servicio y posibles factores que puedan influenciar su uso. En esta práctica vamos a proponer el análisis de datos resumen diarios sobre la utilización de este servicio entre julio de 2013 y noviembre de 2015.

La filosofía de esta práctica es fomentar que consultéis la documentación en línea tanto de Pandas como de Seaborn, para así familiarizaros más con los diferentes métodos disponibles para resolver los ejercicios propuestos. En cada pregunta, se ofrecen consejos sobre partes relevantes de esta documentación relacionadas con las tareas que se piden.

### Descripción de variables

El archivo de datos que vamos a utilizar puede obtenerse de esta url. Se trata de un fichero en formato CSV, que se ha creado mezclando datos del City Bike System con datos de la National Oceanic and Atmospheric Administration (NOAA), sobre NYC. El fichero cuenta con las siguientes columnas:

- `date`: fecha del dato, en formato YYYY-MM-DD.
- `trips`: entero positivo, número total de viajes acumulados ese día.
- `precipitation`: entero positivo, cantidad de lluvia total registrada ese día (pulgadas).
- `snow_depth`: entero positivo, altura de nieve (pulgadas).
- `snowfall`: entero positivo, registro de precipitación en forma de nieve (pulgadas).
- `max_temperature`: entero, temperatura máxima registrada (ºF).
- `min_temperature`: entero, temperatura mínima registrada (ºF).
- `average_wind_speed`: entero, velocidad promedio del viento (MPH, millas por hora).
- `dow`: [0, 7]; código de día de la semana, 0 corresponde al domingo.
- `year`: Año del registro.
- `month`: Mes del registro.
- `holiday`: Valor lógico, indica si esa fecha es festivo (TRUE) o no (FALSE).

- `stations_in_service`: Número de estaciones para tomar o dejar bicicletas que estaban en servicio ese día.
- `weekday`: Valor lógico, indica si esa fecha corresponde a un día entre semana (de lunes a viernes, ambos inclusive).
- `weekday_non_holiday`: Valor lógico, indica si la fecha corresponde a un día entre semana festivo.

Los datos están tomados con frecuencia diaria (filas del archivo).

## Ejercicio 1

Genera una tabla con valores estadísticos resumen para las variables cuantitativas de este conjunto de datos.

## Ejercicio 2

Crea un gráfico que represente la evolución del número total de viajes en bicicleta registrados en el sistema cada mes.

A continuación, genera otro gráfico con la evolución de la media mensual de temperaturas máximas y mínimas.

¿Se pueden observar patrones estacionales o algún tipo de relación entre ambas variables?

## Ejercicio 3

Representa un gráfico con dos paneles, en el que cada panel muestre el histograma y función de densidad de probabilidad del número total de viajes diarios realizados. El panel izquierdo mostrará la distribución del total de viajes diarios en dias no festivos y el panel derecho mostrará la misma distribución pero para días festivos.

## Ejercicio 4

Calcula cual es, en promedio el día de la semana en el que más viajes en bicicleta se realizan y el día que menos viajes registra, usando toda la serie de valores. Si es posible, intenta visualizar estos datos por paneles para mostrar tus conclusiones.