

MÁSTER

# INFRAESTRUCTURAS COMPUTACIONALES PARA EL PROCESAMIENTO DE DATOS MASIVOS

PRÁCTICA DEL MÓDULO 1: ECOSISTEMA DE PROCESAMIENTO  
PARALELO PARA DATOS MASIVOS: APACHE HADOOP

Versión 1.0

Dr. Agustín C. Caminero Herráez — Dr. Rafael Pastor Vargas  
Dr. Salvador Ros Muñoz

MÁSTER UNIVERSITARIO EN INGENIERÍA Y CIENCIA  
DE DATOS

# Contenido

Introducción .....2

Ejercicio 1: MapReduce.....3

    Ejercicio 1.1: Contador de clientes valorados por países. ....4

    Ejercicio 1.2: País con mejores clientes. ....5

    Ejercicio 1.3: Mejorando el país con mejores clientes.....6

Ejercicio 2: Hive.....7

Ejercicio 3: Pig..... 10

## Introducción

En este documento se presenta el Trabajo Práctico (TP) del módulo 1 de la asignatura "INFRAESTRUCTURAS COMPUTACIONALES PARA EL PROCESAMIENTO DE DATOS MASIVOS", del "MÁSTER UNIVERSITARIO EN INGENIERÍA Y CIENCIA DE DATOS" de la UNED.

Este trabajo se realiza de forma individual. En las siguientes secciones se exponen los diferentes ejercicios que es necesario implementar para este trabajo.

Se proponen tres ejercicios diferentes. El primer ejercicio consiste en la implementación de un trabajo MapReduce con la librería MRJob. El segundo ejercicio consiste en la implementación de un trabajo utilizando Hive y el tercer ejercicio consiste en utilizar la herramienta Pig.

**ES NECESARIO REALIZAR EL EJERCICIO 1 DE FORMA OBLIGATORIA, Y UN EJERCICIO MÁS A ELEGIR ENTRE LOS EJERCICIOS 2 Y 3.**

El ejercicio 1 se valorará con un máximo de 8 puntos sobre 10, mientras que los ejercicios 2 y 3 se valorarán con un máximo de 2 puntos sobre 10.

La forma de evaluar el trabajo se hará en base a lo siguiente:

- **Jupyter notebook** con el código realizado. Se debe incluir no solamente el código sino también las explicaciones necesarias, imágenes, ... de forma que el notebook sea autocontenido. Al principio del notebook se debe indicar el nombre del/la estudiante. Los notebooks tienen que estar ejecutados (deben mostrar la salida de la ejecución del código realizado), y se deben incluir las órdenes necesarias para demostrar el correcto funcionamiento de los desarrollos. Por ejemplo, para demostrar que una tabla se ha creado y cargado de datos correctamente, habría que ejecutar un select que devuelva algunos de los contenidos de la tabla. Los notebooks no deben contener órdenes que ralenticen su carga de forma innecesaria, por ejemplo no se deben utilizar consultas select \* from tabla para tablas con más de una decena de filas, habría que utilizar la cláusula limit.
- De forma optativa, se puede incluir una **memoria explicativa**.

Se valorará positivamente se incluya un apartado final donde se explique la opinión del/la estudiante sobre este trabajo, puntos fuertes y/o débiles, recomendaciones para el futuro, así como una valoración general de este módulo.

## PRÁCTICA DEL MÓDULO 1: ECOSISTEMA DE PROCESAMIENTO PARALELO PARA DATOS MASIVOS: APACHE HADOOP

Este material se deberá incluir en un fichero comprimido y enviado a través del curso virtual dentro de los plazos establecidos para su entrega. El nombre de dicho fichero comprimido deberá tener la estructura *MR-ApellidosNombre.zip*, donde *Apellidos* y *Nombre* deben sustituirse por los valores correspondientes para el/la estudiante que realiza el envío. Un nombre correcto es, por ejemplo, el siguiente:

- MR-CamineroHerraezAgustin.zip

A continuación se detallan los ejercicios a completar.

### Ejercicio 1: MapReduce.

Este ejercicio consta de 3 partes, que se detallan en este apartado. Para cada uno de las partes es necesario presentar lo siguiente:

- El diseño del programa MapReduce. Este diseño debe contener al menos respuestas a las siguientes preguntas:
  - ¿Cuántos pasos MapReduce son necesarios?
  - ¿Qué hace cada función de cada paso? (No es necesario código ejecutable, una descripción en texto o en pseudocódigo es suficiente).
  - ¿Qué datos se pasan de una función a la siguiente?
- Implementación de este diseño. Este código debe ejecutarse en el entorno de desarrollo propuesto para el tema de MapReduce.

El detalle de cada apartado se encuentra a continuación.

## PRÁCTICA DEL MÓDULO 1: ECOSISTEMA DE PROCESAMIENTO PARALELO PARA DATOS MASIVOS: APACHE HADOOP

### Ejercicio 1.1: Contador de clientes valorados por países.

Partiendo de los ficheros de datos de países y clientes y del código visto en el ejemplo `mrjob-join`, mejora dicho código para que el programa devuelva cuántos clientes con valoración “bueno” hay en cada país. En concreto, la salida del programa MapReduce debe ser un fichero con el contenido que se muestra en la Figura 1 . Este ejercicio tiene una puntuación de hasta 3 puntos sobre 10.

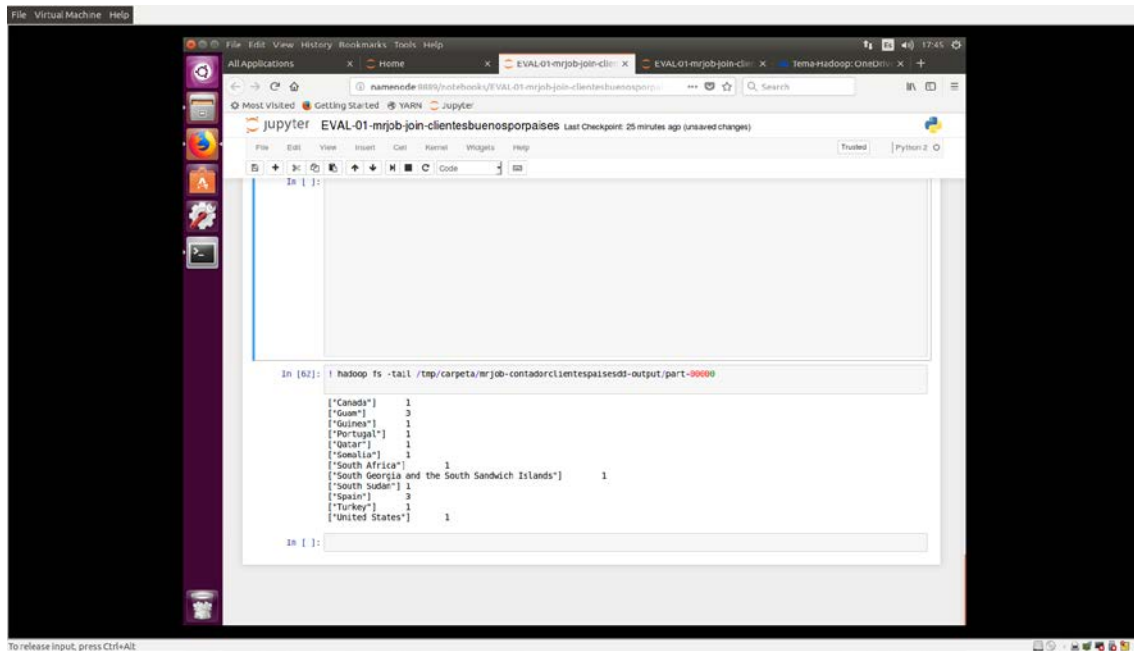


Figura 1. Fichero de salida del contador de buenos clientes.

## PRÁCTICA DEL MÓDULO 1: ECOSISTEMA DE PROCESAMIENTO PARALELO PARA DATOS MASIVOS: APACHE HADOOP

### Ejercicio 1.2: País con mejores clientes.

Partiendo del código implementado en el ejercicio anterior, extiéndelo para que devuelva el país en el que hay más clientes valorados como "bueno". En el caso de que haya más de un país con el mismo número de clientes buenos empatados en el primer lugar, se devolverá solamente uno de ellos. El resultado de este ejercicio se muestra en la Figura 2. Este ejercicio tiene una puntuación de hasta 3 puntos sobre 10.

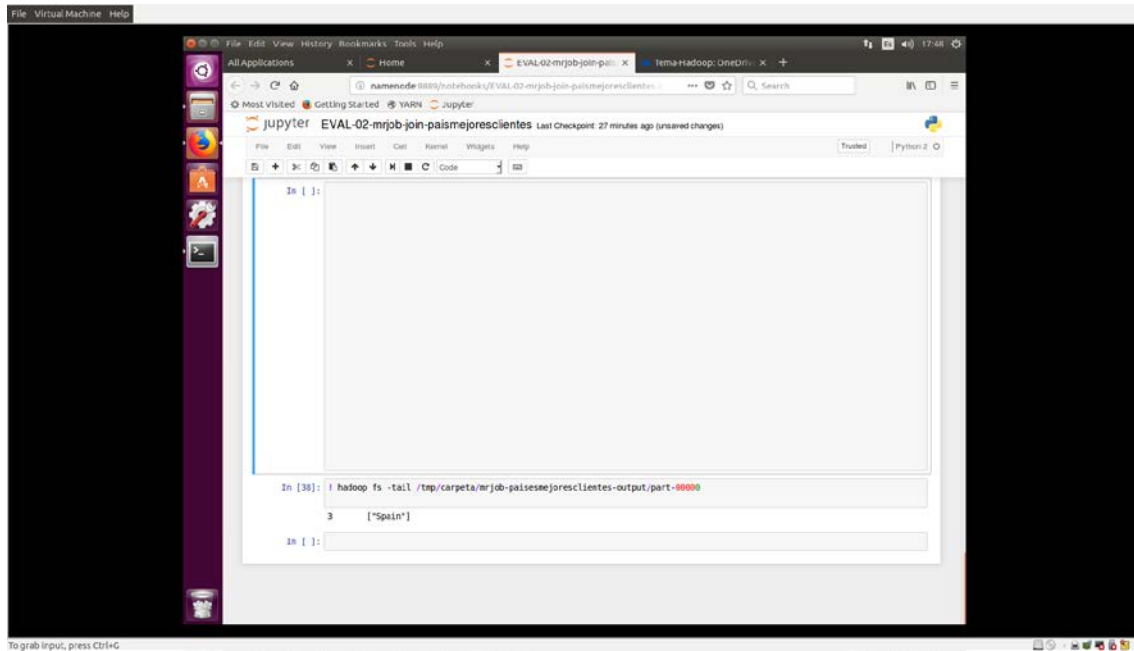


Figura 2. Fichero de salida con los mejores clientes.

## PRÁCTICA DEL MÓDULO 1: ECOSISTEMA DE PROCESAMIENTO PARALELO PARA DATOS MASIVOS: APACHE HADOOP

### Ejercicio 1.3: Mejorando el país con mejores clientes.

Partiendo del código implementado para el ejercicio anterior, mejóralo para que, en el caso de que haya más de un país empatado con el mayor número de buenos clientes, se devuelvan todos esos países. Utilizando los ficheros de datos sobre países y clientes vistos anteriormente, la salida de este programa MapReduce debería ser la que se muestra en la Figura 3. Hay que tener en cuenta que la entrada del reducer es una clave y un *generator* que contiene los valores que comparten la misma clave. Este ejercicio tiene una puntuación de hasta 4 puntos sobre 10.

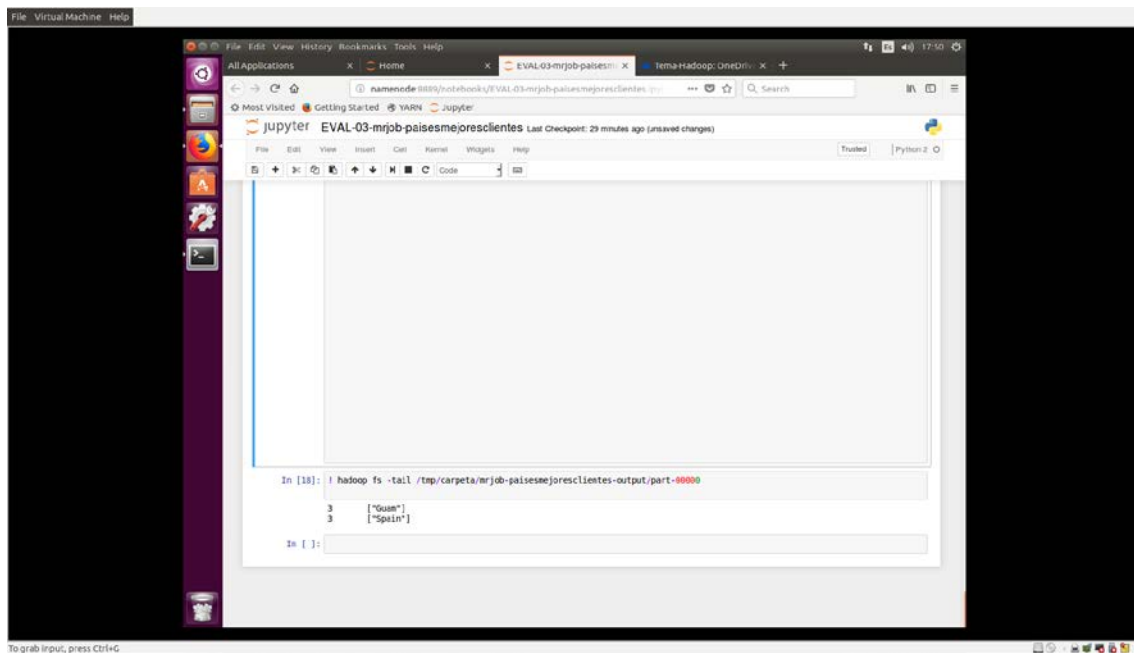


Figura 3. Fichero de salida con los mejores clientes, mejorado.

## Ejercicio 2: Hive

En este trabajo se van a utilizar ficheros (**disponibles en el curso virtual**) descargados de la Kaggle (<https://www.kaggle.com/sp1thas/book-depository-dataset>) y contienen información sobre libros y sus autores. Son los archivos authors.csv y dataset.csv, con una serie de campos separados por ; donde su estructura es la siguiente:

- authors.csv: author\_id; author\_name.
- datasets.csv: title; author\_id; bestsellers-rank; imprint; publication-date; rating-avg; rating-count

El detalle de los campos de estos ficheros se puede encontrar en [11]. Estos dos ficheros se relacionan mediante el campo author\_id. Por cuestiones de tamaño se ha reducido el total de campos para adaptarlos al ejercicio, también se han eliminado algunos registros para eliminar inconsistencias.

En el curso virtual se han publicado datos consistentes en un fichero CSV en el que cada línea del fichero contiene la siguiente información:

- Author\_id: Identificador del autor.
- Author\_name: Nombre completo del autor.
- Title: Título del libro.
- Bestsellers-rank: Posición de este libro en el ranking de bestsellers.
- Imprint: Este campo no se encuentra documentado.
- Publication-date: Fecha de publicación.
- Rating-avg: Valoración media.
- Rating-count: Contador de valoraciones.

En la siguiente tabla se muestran las primeras filas del fichero authors.csv:

author_id	author_name
3	Cecelia Mecca
4	Ana Maria Spagna



## PRÁCTICA DEL MÓDULO 1: ECOSISTEMA DE PROCESAMIENTO PARALELO PARA DATOS MASIVOS: APACHE HADOOP

Y en la siguiente tabla se muestran las primeras filas del dataset.csv:

title	author_id	bestsellers-rank	imprint	publication-date	rating-avg	rating-count
The Mercenary : Order of the Broken Blade	3	51517		17/09/2019	4.49	211
100 Skills You'll Need for the End of the World (as We Know It)	4	162985		05/05/2015	2.96	139

Partiendo de este dataset, desarrolla las órdenes de HiveQL que implementen las siguientes tareas.

1. (2 puntos) Crea las tablas necesarias para almacenar datos. Pueden ser internas o externas en función de los datos que se desee. La decisión de interna o externa debe estar razonada.
2. (1 puntos) Importa los datos en las tablas creadas.
3. (1 punto) Crea una vista sobre las tablas creadas. Esta vista tendrá para cada título, el nombre del autor, fecha de publicación, y valoración media.
4. (3 puntos) Crea las consultas de Hive necesarias para responder las siguientes cuestiones:
  - o ¿Cuál es el título del libro con mejor valoración media del autor Rand McNally?
  - o ¿Cuáles son los cinco autores cuyos libros tienen más valoraciones?
  - o ¿Cuáles son los dos libros más recientes del autor Rand McNally?
5. (3 puntos) Selecciona un dataset, impórtalo en las tablas de Hive necesarias e implementa al menos dos consultas sobre dicho dataset. Para realizar este ejercicio, es necesario seguir los siguientes pasos:
  - Selecciona una fuente de datos públicamente disponible en Internet. Puede ser un dataset o alguna otra fuente de datos como por ejemplo una red social. Si es una fuente de datos, deberás utilizar alguna de las herramientas de inyección de datos y serdes vistos en la asignatura para capturar datos para realizar este ejercicio. Si se decide utilizar un dataset, en los siguientes enlaces se pueden encontrar algunos de libre acceso (existen más repositorios de datasets):
    - o <https://www.kaggle.com/datasets>
    - o <https://data.worldbank.org/>

## PRÁCTICA DEL MÓDULO 1: ECOSISTEMA DE PROCESAMIENTO PARALELO PARA DATOS MASIVOS: APACHE HADOOP

- o <https://datasetsearch.research.google.com/>
  - o [https://console.cloud.google.com/marketplace/browse?filter=solution-type:dataset&\\_ga=2.67976598.645382453.1601640482-195064274.1601640482](https://console.cloud.google.com/marketplace/browse?filter=solution-type:dataset&_ga=2.67976598.645382453.1601640482-195064274.1601640482)
- Una vez seleccionada la fuente de datos, el primer paso será publicar un mensaje en el foro del módulo indicando el enlace a dicha fuente de datos. Además, revisa si hay otro/a estudiante que haya publicado previamente dicha fuente de datos, en cuyo caso deberás elegir otra distinta. El objetivo es evitar que dos o más estudiantes utilicen los mismos datos en sus trabajos.
- Utilizando la fuente de datos seleccionada, importa los datos en Hive e implementa al menos dos consultas sobre dichos datos.
- Esta parte se evaluará de la siguiente forma:
  - o (1.5 puntos) Selección de dataset, creación de las tablas e importación de los datos.
  - o (1.5 puntos) Creación de consultas.
- Se valorará positivamente la complejidad de los datos utilizados así como la de las consultas implementadas. En el caso de que no sea necesario utilizar herramientas de inyección de datos y serdes, se podrá conseguir la máxima puntuación en este apartado si las consultas tienen complejidad suficiente.

## Ejercicio 3: Pig

Partiendo del notebook pig-indice-invertido-estudiantes.ipynb que se proporciona, implementa un índice invertido sobre el dataset de los fórum posts utilizado en el curso virtual. Este índice debe contener para cada palabra, un listado de los identificadores de los posts en los que aparece, así como un contador que indique en cuántos posts aparece.

En detalle, los pasos a seguir serían los siguientes:

1. (1 punto) Carga el fichero de los posts forum\_node.tsv, recuerda que está separado por tabuladores.
2. (2 puntos) Limpia el fichero: elimina del body caracteres que no sean letras o números, pásalo a minúsculas, confirma que el identificador es numérico, entre otras opciones. En el identificador de post, elimina caracteres que no sean numéricos.
3. (2 puntos) Separa el body en palabras y júntalas con el identificador.
4. (2 puntos) Elimina duplicados (palabras que aparecen más de una vez en un post).
5. (1 puntos) Agrupa las palabras iguales.
6. (1 puntos) Prepara los resultados. Para cada palabra, hay que mostrar en orden ascendente el identificador de los posts donde aparece y la cuenta de los posts.
7. (1 punto) Almacena los resultados en HDFS.

Una parte de la salida de este programa es el siguiente:

```
(7001187)},13)
(zycster's,{(9247)},1)
(zyrcter,{(11610)},1)
(zytrax,{(6028725),(7002663)},2)
(zyx,{(8004310)},1)
(zz,{(1007745),(10011348)},2)
(zzz,{(8385)},1)
(zzzz,{(14790),(30278)},2)
(zzzzz,{(1007093),(5006080)},2)
(zzzzzzzzzzzzzzzzzzzz,{(8353)},1)
```

Es interesante saber lo siguiente:

- La siguiente función comprueba si un dato X es numero: `org.apache.pig.piggybank.evaluation.IsNumeric(X)`. Recuerda que para utilizar una función de Piggybank es necesario registrar la librería.
- Para limpiar el body se puede utilizar esta expresión regular: `'[^a-zA-Z0-9\\']+'`