Ejercicio de Evaluación "Manejo de datos masivos con Hadoop"

Sistema de Bases de Datos

Diego Díaz

Contenido

Introducción

En este documento se encuentran las respuestas de la práctica de la asignatura Sistemas de Bases de Datos de la UNED del curso 2016-2017.
Se ha utilizado el entorno de Cloudera para trabajar con Hadoop.
Anexo a esta memoria, se entregan todos los ficheros en python, los ficheros de texto obtenidos en las tareas Mapreduce y las imágenes utilizadas en este documento. Todo ello se encuentra en la carpeta Archivos.

Proceso de carga de los datos en Hadoop

Para el proceso de carga del fichero GSD1001_full.soft.txt se realizan las siguientes acciones las cuales se pueden ver resumidas en la **Imagen1**:

1. En el cluster de Hadoop se crea un nuevo directorio con el entorno gráfico de HUE que llamamos Dataset.

2. Desde esta carpeta subo con la terminal el fichero GDS1001_full.soft.txt
Hadoop fs -put GDS1001_full.soft Dataset

3. Se verifica que el archivo se ha subido correctamente.

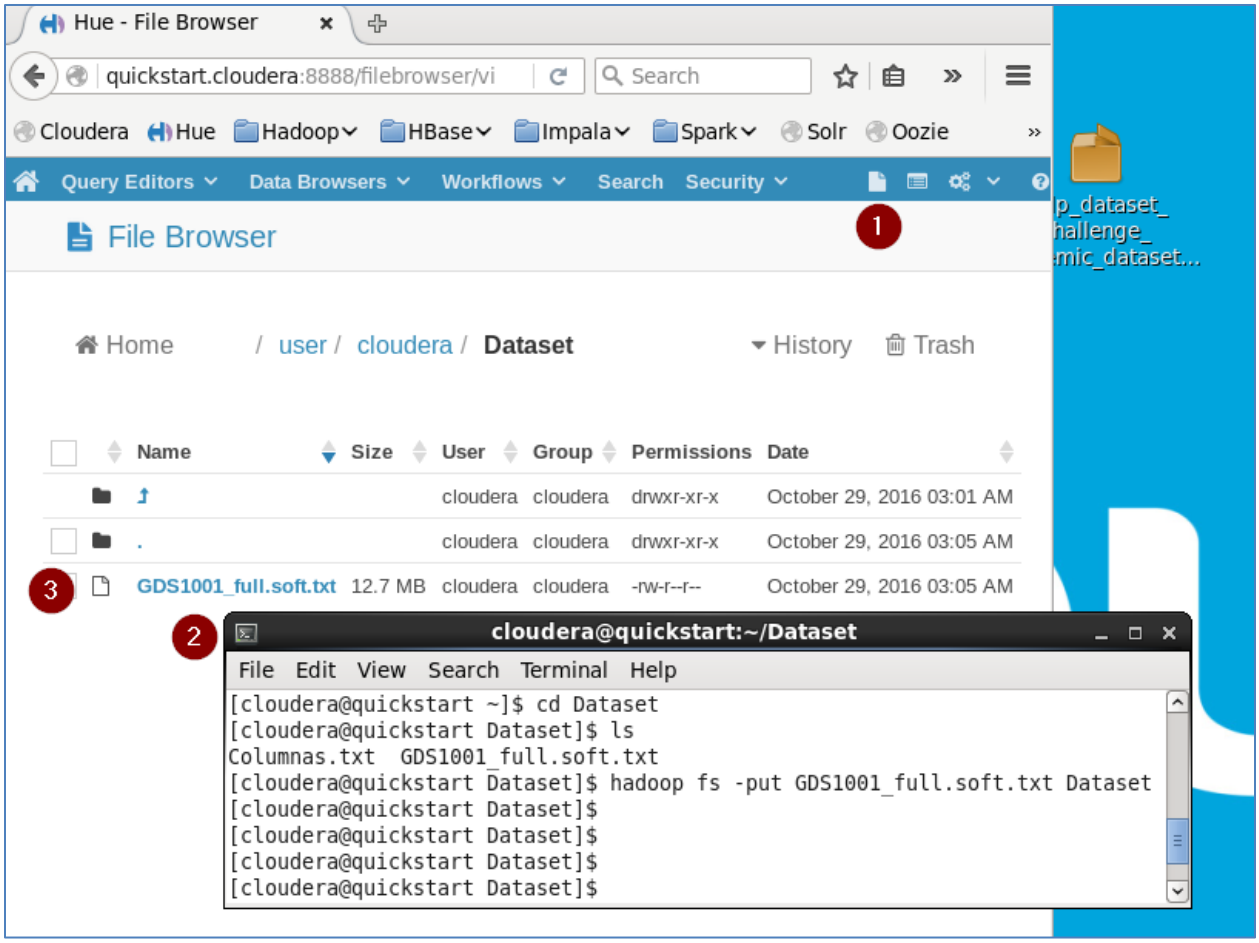

Imagen 1. Acciones para la preparación del fichero de datos.

1. TAREA 1. Trabajo con MapReduce

Partiendo del fichero *GSD1001_full.soft.txt*
Nos quedamos con los primeros 13 campos: "idref", "ident", "gsm19023", "gsd19024", "gsd19025", "gsd19026", "genetitle", "genesymbol", "geneID", "uniGenetitle", "uniGenesymbol", "uniGeneID", "NucleotideTitle". Además, tenemos que incluir tres nuevas columnas, llamadas *"max", "min" y "avg"* que contendrán datos numéricos resultado de calcular el valor máximo, mínimo y medio de las columnas *"gsm19023", "gsd19024", "gsd19025" y "gsd19026"* para cada fila, respectivamente. De esta forma, el resultado de este punto es un fichero que llamaremos *DataClean.txt* que tendrá 16 columnas.

1. En Mi carpeta local en cloudera > Dataset se guardan los dos script en Python que necesitamos para la tarea MapReduce.[1]
mapper_clean.py
reducer_clean.py

[1] El contenido de los archivos mapper_clean.py y reducer_clean.py se entregan junto a esta memoria en la carpeta Archivos/Dataset

En Hadoop hay un programa en Java que se llama hadoop-streaming.jar. Este programa lee y escribe por la salida estándar (stdin/stdout) línea por línea. De esta manera Python puede leer cada línea como una cadena de caracteres y analizar la sintaxis (parser) con funciones que separan la información (strip/split).

2. Con estos tres elementos abrimos un terminal y ejecutamos el siguiente comando. Se verifica que el proceso ha tenido éxito como se observa en la **Imagen 2.**

hadoop jar /usr/lib/hadoop-mapreduce/hadoop-streaming.jar
-file **Dataset/mapper_clean.py**
-mappe **Dataset/mapper_clean.py**

-file **Dataset/reducer_clean.py**
-reduce**Dataset/reducer_clean.py**

-input **/user/cloudera/Dataset/GDS1001_full.soft.txt**[2]
-output **/user/cloudera/Dataset-output**[3]

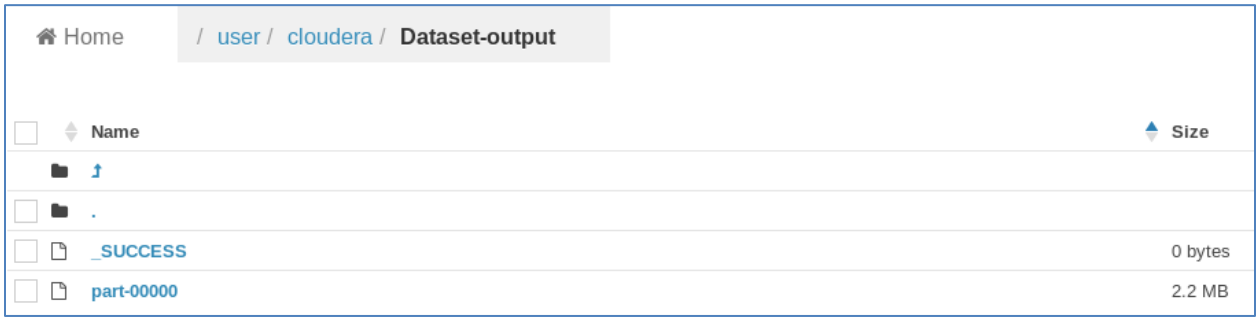

Imagen 2. Muestra éxito (success) y el fichero de salida (part-00000)

3. Se renombra el fichero de salida desde el terminal con la secuencia de comandos que se detalla en la **Imagen 3**. Este fichero se utiliza para los apartados siguientes con el nombre de DataClean.txt[4]

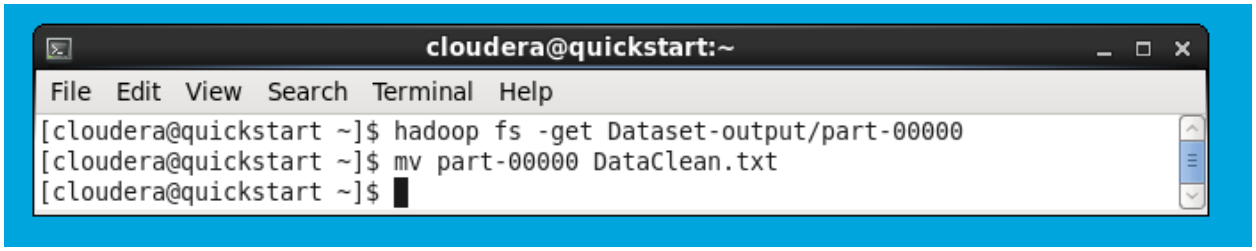

Imagen 3. Secuencia de comandos para descarga de fichero de Hadoop y renombramiento.

[2] -input **/user/cloudera/Dataset/*** (funcionaria igualmente)
[3] Este directorio se crea en el cluster de Hadoop, así que no debe estar creado inicialmente
[4] Este fichero se entrega con la documentación en la carpeta Archivos\DataFinal

2. TAREA 2. Trabajo con MapReduce

Partiendo del fichero DataClean.txt creado en el punto anterior, implementa un trabajo MapReduce que revuelva el mayor “avg” de aquellas filas cuyo valor de "gsm19023" esté entre 100 y 1000.
El proceso es el mismo que en el apartado anterior cambiando el nombre de las carpetas y archivos correspondientes. Se crean nuevos script en Python para tal propósito.[5]

El resultado obtenido de la media es = 2218.625

Se renombra el fichero de salida desde el terminal con la secuencia de comandos que se detalla en la **Imagen 4**.

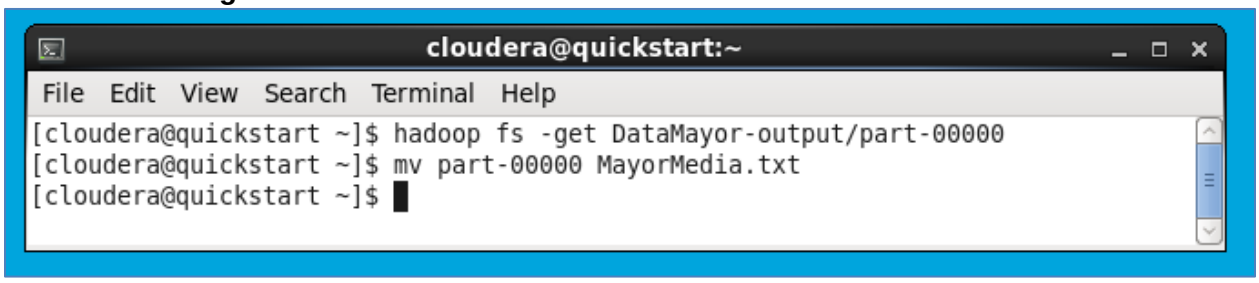

Imagen 4. Secuencia de comandos para descarga de fichero de Hadoop y renombramiento.

Nota: Para comprobar el resultado obtenido con mi Script, creo una tabla test que carga el fichero de salida de mi mapper_mayor.py y hago una consulta con HIVE. Se comprueba en la **Imagen 5** que se obtiene el resultado esperado.

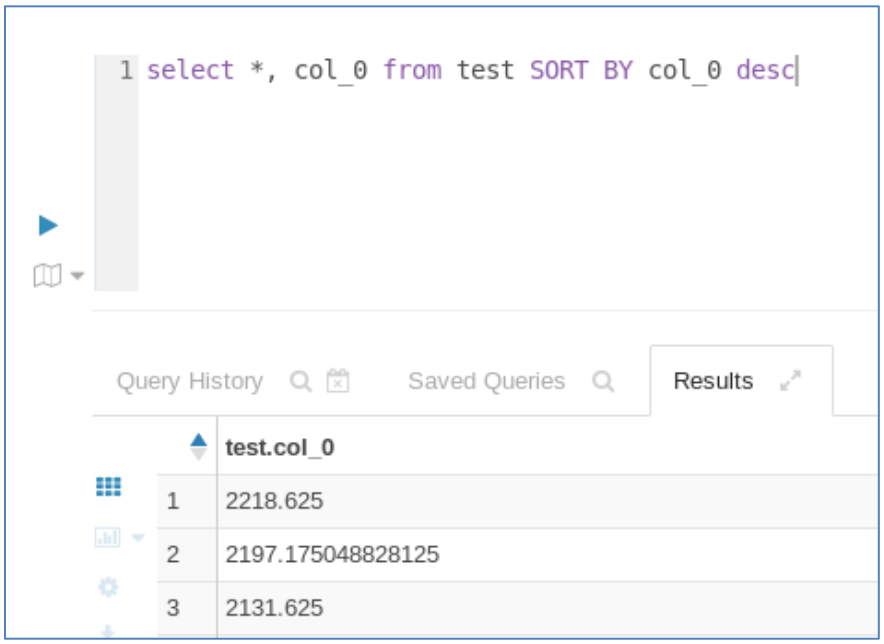

Imagen 5. Resultado con HIVE coincide con mi tarea MapReduce

[5] El contenido de los archivos mapper_mayor.py y reducer_mayor.py se entregan junto a esta memoria en la carpeta Archivos/DataMayor el resultado de la tarea MapReduce en el fichero MayorMedia.txt en la carpeta Archivos\DataFinal

3. TAREA 3. Crear una tabla interna y otra externa

3.1. Crear una tabla interna

Se elige utilizar el entorno gráfico de HUE tal como se muestra en la **Imagen 5.**

1. Se sube el fichero renombrado en la actividad 1 al cluster de Hadoop
Hadoop fs -put DataClean.txt

2. Opción en HUE: Data Browser > Metastore Tables.

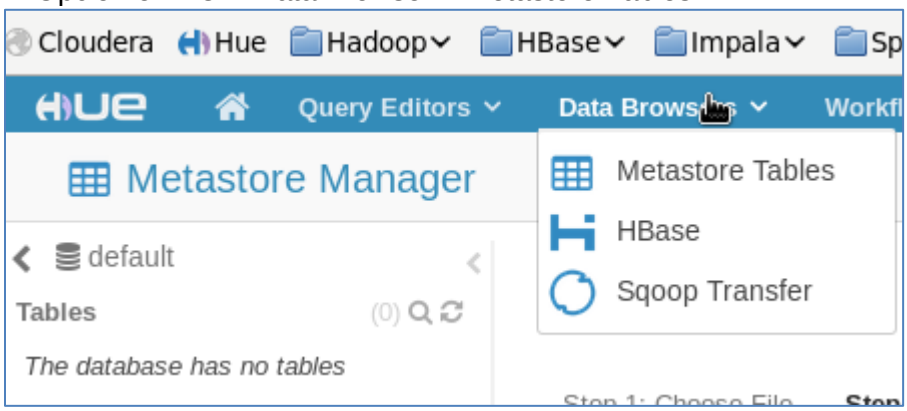

Imagen 5. Selección en el entorno gráfico para crear una tabla interna.

Añadir nueva tabla. En el paso 1(Choose File) se introduce el nombre de la tabla y se selecciona el fichero necesario para importar los datos (DataClean.txt). Se elimina de la ubicación original por lo que se hace una copia de este fichero antes de este proceso.
En el paso 2 (Choose Delimiter) no se modifica nada y se pulsa siguiente. En el paso 3 (Define columns) se puede cambiar el nombre de las columnas (Bulk edit column names) tal como se indica en la **Imagen 6.** Los nombres se sacan de las primeras 13 columnas del archivo "Columnas.txt" al cual se le han añadido las tres columnas: maximo, mínimo, media.

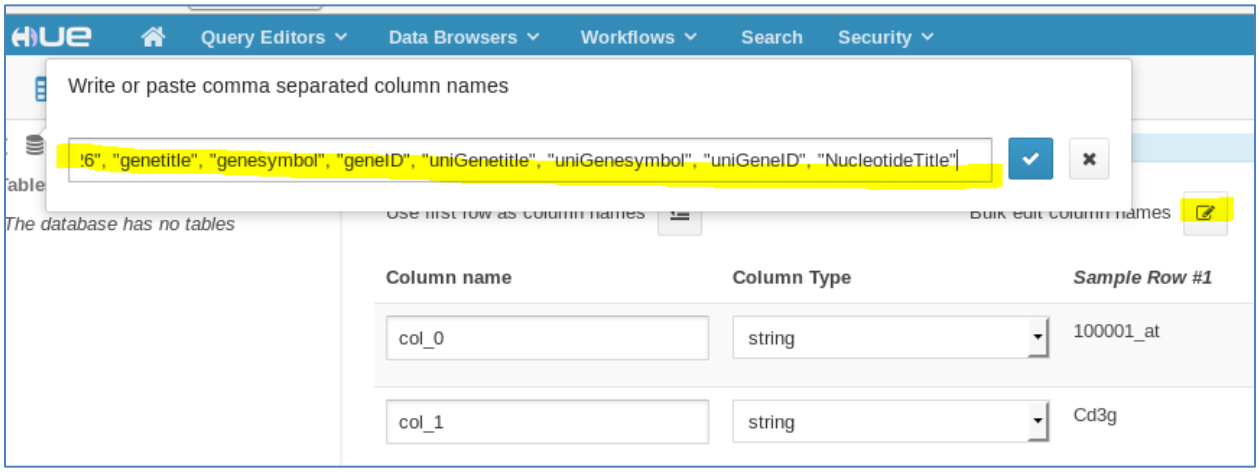

Imagen 6. Edición de los nombres de las columnas.

Se aplican los cambios tal como se ve en la **Imagen 7** y para finalizar se crea la tabla y aparecerá tal como se muestra en la **Imagen 8.**

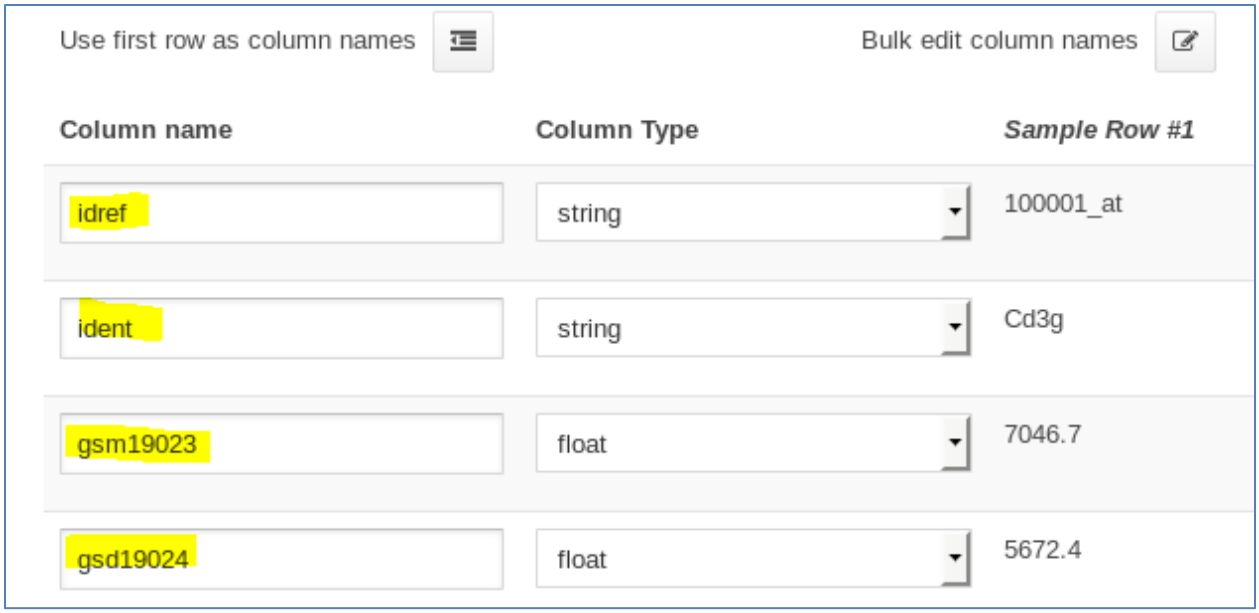

Imagen 7. Cambio de nombres de las columnas con opción "Bulk edit column names"

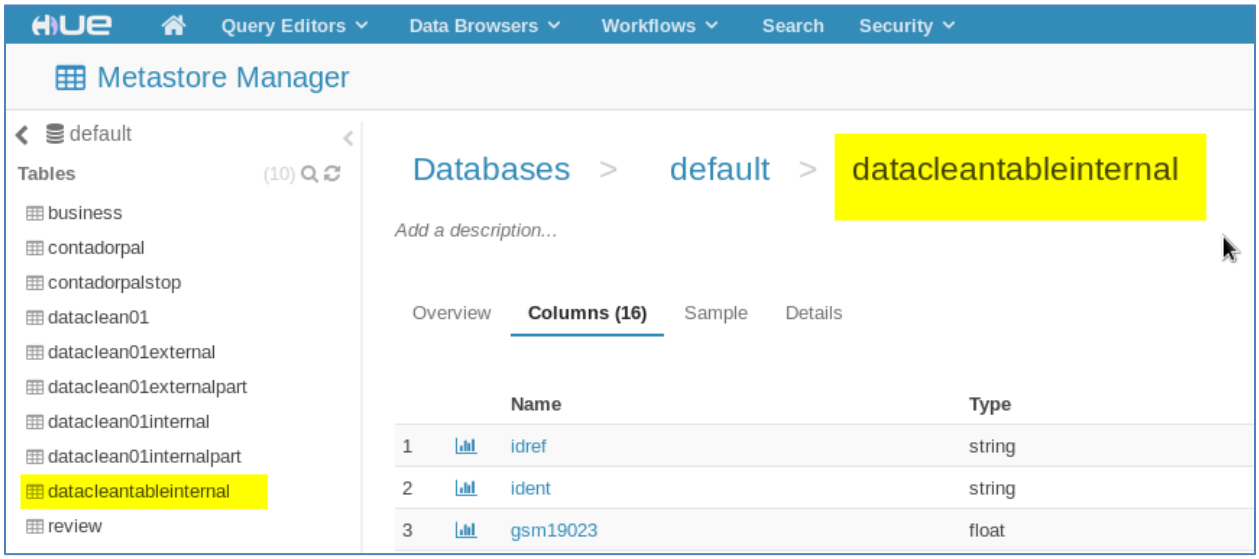

Imagen 8. La nueva tabla aparece dentro de la lista de Tablas de "Metastore Manager".

3.2. Crear una tabla externa

Las carpetas externas mantienen los datos en su ubicación original (no los mueven). La ventaja de esto es que varias tablas externas puedan acceder a los mismos datos (si se borra de la ubicación entonces las consultas quedan vacías porque no habrá datos)

1. Se sube el fichero renombrado en la actividad 1 al cluster de Hadoop en una carpeta.
Hadoop fs -mkdir DataExternal
Hadoop fs -put DataClean.txt DataExternal

2. Se ejecuta el código (1) desde el editor de consultas de HIVE y aparecerá (2) la nueva tabla como se muestra en la **Imagen 9.**

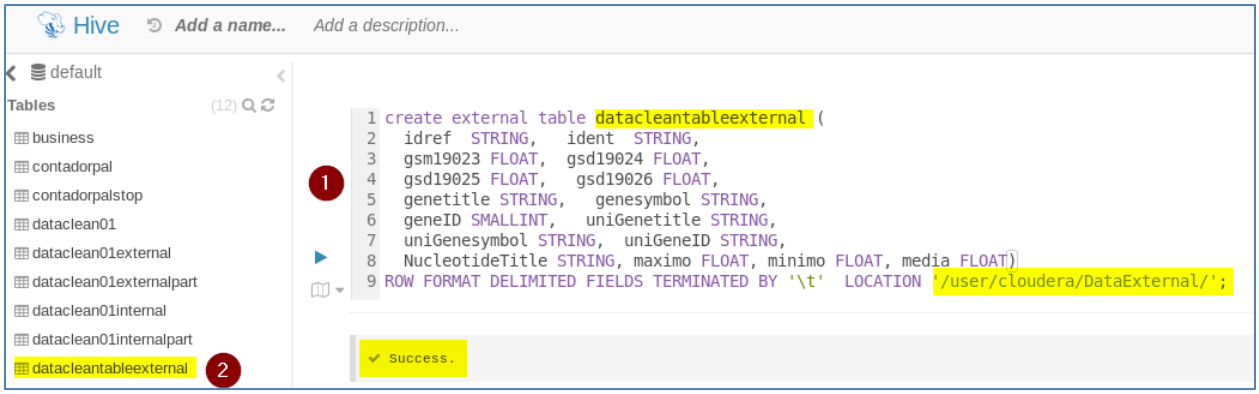

Imagen 9. Proceso de creación de una tabla externa con comando con Hive.

<u>Se pueden consultar si los valores de la tabla se han subido correctamente</u>
select * from datacleantableexternal
select * from datacleantableinternal

4. TAREA 4. Consultas

Para realizar las consultas se utiliza HIVE. El resultado de las consultas se muestra una sola vez pero sí que se comprueba que se obtiene el mismo resultado tanto con la tabla interna como con la tabla externa[6].

Se muestra en la **Imagen 10** la manera de entrar a las opciones de HIVE.

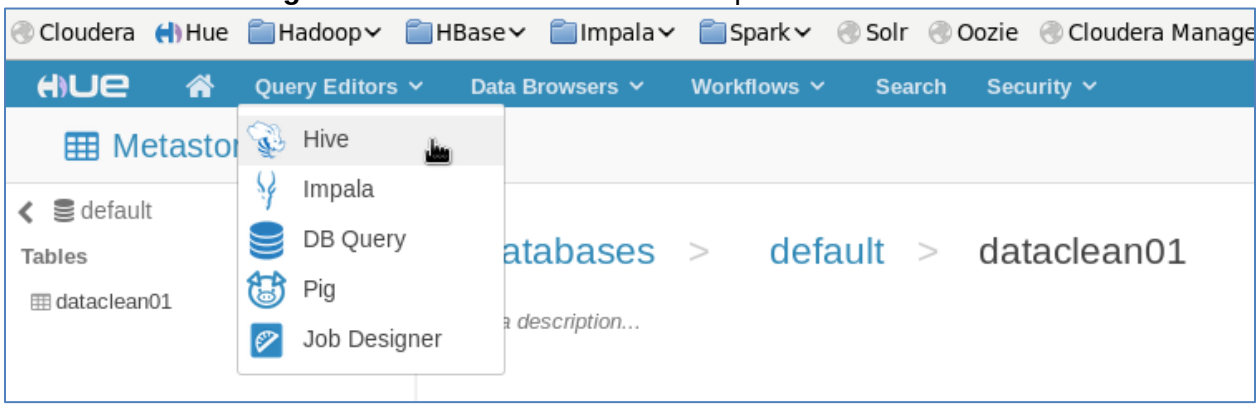

Imagen 10. Entrada en el entorno Hive para realizar las consultas

[6] El resultado de las consultas se puede ver en las imágenes en la carpeta Archivos\Imagenes

A. Devuelve todas las columnas de las muestras. Se escogen las 10 de mayor valor de GSD19026 como se observa en la **Imagen 11.**[7]

```
select idref, ident, gsm19023, gsd19024, gsd19025, gsd19026, genetitle, genesymbol, geneID, uniGenetitle,
uniGenesymbol, uniGeneID, NucleotideTitle, maximo, minimo, media from datacleantableexternal order by GSD19026 desc limit 10
```

Query History | Saved Queries | Results

	idref	ident	gsm19023	gsd19024	gsd19025	gsd19026	genetitle
1	101869_s_at	Hbb-bs	132601	114032	157629	154780	hemoglobin, beta adult s chain///hemoglobin, beta adult minor chain///hemoglo
2	94781_at	Hba-a1	130873	117516	138931	153184	hemoglobin alpha, adult chain 1
3	94781_at	Hba-a1	130873	117516	138931	153184	hemoglobin alpha, adult chain 1
4	94781_at	Hba-a1	130873	117516	138931	153184	hemoglobin alpha, adult chain 1
5	101071_at	Myh6	97751.8984375	87078.203125	122362	120639	myosin, heavy polypeptide 6, cardiac muscle, alpha
6	93050_at	Myl2	89821.8984375	82139.296875	108902	116616	myosin, light polypeptide 2, regulatory, cardiac, slow
7	93514_at	Myl3	74782.203125	63824.6015625	107175	112905	myosin, light polypeptide 3
8	99660_f_at	Cox7c	89623.1015625	79781.6015625	115008	109968	cytochrome c oxidase subunit VIIc
9	100921_at	Tnni3	74669	65014.69921875	109347	109793	troponin I, cardiac 3
10	102599_at	Tpt1	91097.1015625	85752.5	103993	106022	tumor protein, translationally-controlled 1

Imagen 11. Primera consulta del apartado 4.

B. Media de GSD19026 para los genes relacionados con el retraso mental. Se observa el resultado de la consulta en la **Imagen 12.**[8]

```
SELECT avg(gsd19026) from datacleantableinternal where genetitle like '%retard%'
```

Query History | Saved Queries | Results

	_c0
1	263.62222544352215

Imagen 12. Segunda consulta del apartado 4.

[7] select * from datacleantableinternal order by GSD19026 desc limit 10
[8] select avg(gsd19026) from datacleantableinternal where genetitle like '%retard%'

Opinión

La práctica me ha resultado muy interesante y me hubiese gustado que la asignatura tuviese más relación con el contenido de la misma.

Dentro de la documentación que proporciona el equipo docente se podría haber incluido un documento con sintaxis básica de Python para realizar de forma más eficiente los Script. Creo que el objetivo docente de la práctica es utilizar Hadoop tal como se muestra en los vídeos y no dicho lenguaje. El tiempo que he perdido en buscar la sintaxis lo podría haber utilizado para aprender más consultas con Hive o incluso haber programado otros Script MapReduce más completos.

Los vídeos con chroma key me han gustado mucho y el sonido es bastante aceptable. Este es el tipo de cosas que me parecen un acierto total ya que es más agradable poder seguir vídeos que están hechos con buena calidad. Las explicaciones del vídeo son claras y el proporcionar los ficheros que se necesitan para elaborar los ejercicios es desde luego de agradecer, ya que de esa manera he podido seguir los ejemplos de forma bastante fluida (también he realizado las actividades que se muestran en los vídeos).