UNIVERSIDAD DE GRANADA

MÁSTER DE CIENCIA DE DATOS E INGENIERÍA DE COMPUTADORES

CURSO ACADÉMICO 2019-2020

BIG DATA 1

PROCESAMIENTO Y ALMACENAMIENTO DE DATOS APACHE IMPALA

Procesamiento de datos con *Impala*

Documentación de los procesos ejecutados con Impala para el procesamiento de un dataset masivo.

Nicolás Cubero

12 de Abril de 2020

Índice

1. Descripción del *dataset*

En este proyecto, se trabará con *Impala* para llevar a cabo procesamientos sobre el *dataset* ***COVID-19 open line list*** [1], el cual constituye una de las fuentes de datos de **Novel Corona Virus 2019 Dataset**.

Este *dataset* recoge multitud de datos de distintos pacientes, mayoritariamente de China, que padecen la enfermedad provocada por el reciente virus *SARS-CoV-2*, o *Covid-19*, denominado de forma común como *Coronavirus*.

El *dataset* tratado en este proyecto recoge únicamente 15 de las 44 características originales recogidas en el *dataset* original, para un total de 14151 pacientes:

- **ID**: Identificador unívoco que se asigna a cada paciente.
- **age**: Edad del paciente.
- **sex**: Sexo del paciente.
- **City**: Ciudad en la que reside el paciente.
- **Province**: Provincia en la que reside el paciente.
- **Country**: País en el que reside el paciente.
- **wuhan(0)_not_wuhan(1)**: valor binario que indica si el paciente se contagió en Wuhan (1) o no (0).
- **latitude**: Latitud de la ciudad de residencia del paciente.
- **longitude**: Longitud de la ciudad de residencia del paciente.
- **date_onset_symptoms**: Fecha en la que el paciente empezó a presentar síntomas de la enfermedad.
- **date_admission_hospital**: Fecha en la que el paciente fue ingresado en el hospital.
- **date_confirmation**: Fecha en la que se confirmó el positivo en la enfermedad en el paciente.

[1] Enlace a la página web de Kaggle con el la fuente de datos COVID-19 open line list https://www.kaggle.com/sudalairajkumar/novel-corona-virus-2019-dataset

- **symptoms**: Texto que describe brevemente los síntomas que presenta el paciente.
- **outcome**: Indica el final de la enfermedad si la hubo: el paciente fue dado de alta (*discharge*) o murió (*death*).
- **date_death_or_discharge**: Fecha en la que se produjo la muerte o el alta del paciente.

2. Carga de la base de datos

En primer lugar, se procede a copiar el fichero *csv* del *dataset* descargado en el directorio en el sistema de ficheros *hdfs* desde el cual vamos a cargar los datos en *Impala* (`/user/impala/input`):

```
$ hdfs dfs -put COVID19_open_line_list_reduced.csv /user/impala/input
```

Accedemos a la *shell* de *Impala* y creamos la tabla `Covid19_Patients` a la que importamos los datos cargados. Para evitar interferir con otras bases de datos, se establecerá como punto de almacenamiento de esta tabla el fichero `/user/impala/impala_covid19_store.db`:

```
CREATE TABLE IF NOT EXISTS Covid19_Patients (ID INT, age TINYINT, sex STRING,
city STRING, province STRING, country STRING, wuhan_or_not_wuhan TINYINT,
latitude FLOAT, longitude FLOAT, date_onset_symptoms STRING,
date_admission_hospital STRING, date_confirmation STRING, symptoms STRING,
outcome STRING, date_death_or_discharge STRING)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LINES TERMINATED BY '\n'
LOCATION '/user/impala/impala_covid19_store.db'
tblproperties("skip.header.line.count"="1");
```

Y se cargan los datos en esta tabla del fichero *csv*:

```
LOAD DATA INPATH '/user/impala/input/COVID19_open_line_list_reduced.csv'
OVERWRITE INTO TABLE Covid19_Patients;
```

3. Consultas de prueba

Una vez cargada la base de datos, realizaremos dos consultas que impliquen **proyecciones** y **selecciones**:

1. **Contar el número de infectados registrados en cada ciudad**: La sentencia que permitiría obtener dicha información es la siguiente:

   ```
   SELECT  city, province, COUNT(ID) AS n_infectados FROM Covid19_Patients
   GROUP BY province, city ORDER BY n_infectados DESC;
   ```

   Esta consulta realiza una agrupación de los pacientes según la provincia y la ciudad en la que residen (aunque puesto que cada ciudad pertenece a una sóla provincia sería equivalente a agrupar únicamente por ciudad) y ejecuta una **proyección** de la tabla resultante de esta agrupación en las columnas `city`, `province` y la columna generada al contar el número de tuplas recogidas en cada par provincia, ciudad.

   Por último, devolvería la tabla resultante ordenando las tuplas en orden descenciente según el valor de la columna que recoge el número de pacientes por provincia y ciudad.

2. **Contar el número de pacientes que aún no han conseguido recuperarse o han fallecido y realizar el agrupamiento por edad y por sexo**: La sentencia que permitiría extraer esta información es la siguiente:

   ```
   SELECT sex, age, COUNT(ID) AS not_recovered FROM Covid19_Patients
   WHERE outcome!='discharged' OR outcome IS NULL
   GROUP BY sex, age ORDER BY not_recovered DESC;
   ```

   La anterior sentencia aplica una **selección** de aquellos pacientes cuyo campo en `outcome` sea diferentes de *discharged* (alta hospitalaria) y con las instancias seleccionadas efctuaría el agrupamiento primero por sexo y después por edad y realiza, al igual también una **proyección** sobre las columna sexo, edad y la columna generada al contar el número de instancias agrupadas en cada par sexo, edad.

   Por último, devolvería la tabla resultante ordenando las tuplas en orden descenciente según el valor de la columna que recoge el número de pacientes seleccionados por sexo y edad.