Procesamiento de Big Data con SparkR y Hadoop

Cloud Computing: Servicios y aplicaciones

Pedro Manuel Gómez-Portillo López

gomezportillo@correo.ugr.es

https://github.com/gomezportillo/sparkR-hadoop

71722388

1. Parte 1. Hadoop

En esta parte se ha utilizado MapReduce para implementar una función en Java que lee un archivo enorme y calcula la desviación estándar del valor de sus 9 primeras columnas.

2.1. Desarrollo

Aunque en el guion se pide que se generen 3 archivos Java diferentes ha sido imposible hacerlo funcionar de esta forma, por lo que finalmente se ha optado por escribir todas las clases en un solo archivo.

2.2. Resultados

Tras ejecutar la función éste es el fichero que se genera con la desviación estándar de cada columna.

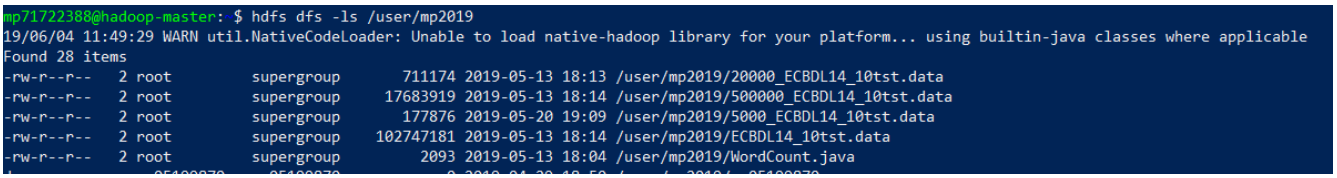
Fila	Desviación estándar
1	0.08882828241387472
2	0.021703475887332218
3	3.1757230250112003
4	2.9413769183762857
5	3.2141933769548197
6	2.6516913721701636
7	3.342770092127666
8	2.945761534602087
9	3.024082895617862

2. Parte 2. Algoritmos de clasificación con SparkR

En esta parte se usará SparkR para realizar en R el procesamiento de un dataset en el servidor Hadoop y se guardarán los resultados también en él, para luego ejecutar varios algoritmos de predicción y comparar sus resultados, siguiendo el guion de prácticas.

2.1. Desarrollo

El primer problema que se encontró al empezar a trabajar en la práctica fue no saber qué dataset utilizar ya que, aunque en el guión pone que utilicemos */user/mp2019/ECDB-2012.training* para el entrenamiento y */user/mp2019/ECDB-2012.test* para las pruebas solo pudieron encontrarse los que se ven a continuación.



Por ello, se ha decidido trabajar con */user/mp2019/ECBDL14_10tst.data* y dividirlo en dos conjuntos, entrenamiento y prueba, como se verá a continuación.

Tras configurar el entorno de trabajo como se indica en el guion y como puede verse en el código de la práctica se balancearon los datos con la clase *class*, y se obtuvieron 48637 filas de cada clase, o un 50% del total cada una, tras lo que se guardaron los datos. Cabe destacar que fue necesario convertir los datos del formato *data.frame* a *tbl_frame* para que SparkR pudiera trabajar con ellos con la función *copy_to()*.

Una vez los datos estaban preprocesados y preparados se pudo comenzar a ejecutar los algoritmos de predicción. Por último, para crear una tabla con los resultados se han utilizado las funciones *print()* y *paste()* de R.

2.2. Resultados

2.2.1. Regresión lineal

Fue el primer algoritmo utilizado, pero la ejecución falló con el error

```
# Error: java.lang.IllegalArgumentException: Field "label" does not exist
```

y aunque se consultó su API[1] y varios foros no se consiguió solucionar, por lo que se pasó al siguiente algoritmo.

1 https://www.rdocumentation.org/packages/sparklyr/versions/0.6.4/topics/ml_linear_regression

2.2.2. Regresión logística

Este algoritmo pudo ejecutarse correctamente, para lo que tardó **~4.72 segundos** y se obtuvo un *accuracy* del **~53,73%**.

2.2.3. Random forest

Los resultados de este algoritmo fueron un tiempo de ejecución de **~6,35 segundos** y un *accuracy* del **~57,66%**.

2.2.4. K-means

Por último también se intentó ejecutar el algoritmo k-means, aunque con resultados similares al de regresión lineal; el siguiente error no permitió terminar su ejecución, y aunque se intentó no ha podido solucionarse.

```
# Error: java.lang.IllegalArgumentException: requirement failed: Column prediction
must be of type DoubleType but was actually IntegerType.
```

2.2.5. Mejor algoritmo

Por tanto, teniendo en cuenta tanto el tiempo de ejecución como el *accuracy* obtenido, el algoritmo con los mejores resultados ha sido el **Random Forest** ya que, aunque ha ha invertido un 50% más de tiempo de ejecución, ha obtenido resultados un **~5% mejores**, lo que en estos algoritmos es preferible, ya que es su principal objetivo.

3. Conclusiones

Esta práctica me ha servido como primera toma de contacto a Hadoop y Spark en un lenguaje con el que no me siento muy cómodo, R, por lo que me ha requerido un esfuerzo extra por aprenderlo.

En conclusión, me alegro mucho de haberla realizado, ya que he adquirido conocimientos que pueden serme necesarios en mi futura vida laboral.