Ejercicio 3

Tras haber trabajado con nuestros datos utilizando Hive y ejecutando consultas en HiveQL, o realizando un procesado de streams sobre los mismos, vamos a ir un paso más allá y tratar de analizarlos mediante machine learning, para realizar predicciones sobre ellos.

En la sesión de streaming, generamos una "predicción" de que equipo iba a ganar una partida en base a los eventos que iban ocurriendo, centrándonos únicamente en la cantidad de oro de cada equipo. ¿Es esto real? ¿Es suficiente? Lo que intentaremos en este tercer ejercicio es aplicar un algoritmo de machine learning sobre un set de datos de partidas para ver si podemos predecir con más fiabilidad que equipo gano una determinada partida, en base a una serie de eventos que consideraremos claves.

Junto con este enunciado tenéis disponibles dos ficheros, training.csv y test.csv, que utilizaremos respectivamente como entradas de entrenamiento y de test para nuestro algoritmo de machine learning. Utilizaremos machine learning sobre Spark, SparkML, sobre la topología de Cloudera vista en la primera sesión presencial.

Lo primero que necesitamos es configurar el entorno. Para ello, con las máquinas virtuales desplegadas, subiremos los ficheros test.csv y training.csv a HDFS utilizando Hue desde el master node. Las versiones test_headers.csv y training_headers.csv son los mismos ficheros pero con las cabeceras, para poder saber que columna contiene cada dato y utilizarlo como referencia.

El ejercicio consiste en predecir, dados unos datos sobre una partida, si el equipo gano o perdió. Los datos que nos interesan son los siguientes:

- Winner. 1 si el equipo fue el ganador, 0 en caso contrario
- FirstBlood. 1 si el equipo fue el primero en dañar al rival, 0 en caso contrario
- FirstTower. 1 si el equipo fue el primero en destruir una torre del rival, 0 en caso contrario
- FirstInhibitor. 1 si el equipo fue el primero en destruir un inhibidor del rival, 0 en caso contrario
- FirstBaron. 1 si el equipo fue el primero en derrotar a este monstruo, 0 en caso contrario
- FirstDragon. 1 si el equipo fue el primero en derrotar a este monstruo, 0 en caso contrario
- TowerKills. Número de torres destruidas por el equipo (11 como máximo)
- InhibitorKills. Número de inhibidores destruidas por el equipo (9 como máximo)
- BaronKills. Número de veces que el equipo ha derrotado a este monstruo (5 como máximo)
- DragonKills. Número de veces que el equipo ha derrotado a este monstruo (9 como máximo)

Todos estos valores nos dan una idea del rendimiento del equipo durante la partida, y de si van por delante de su rival o no. En base a esto, construiremos nuestro experimento de machine learning.

- Podemos utilizar Python, Scala o Java. En el caso de utilizar Python no es necesario compilar el experimento, lo que simplifica el proceso
- Una de las maneras más sencillas de obtener un resultado de forma rápida es utilizando un árbol de decisión (https://spark.apache.org/docs/1.2.1/mllib-decision-tree.html)
    - Como nosotros estamos utilizando un dataset en CSV, necesitamos convertirlo a un RDD de LabeledPoint, con la label 1/0 dependiendo del valor del campo Winner

- Una vez que tengamos listo el experimento, para ejecutarlo simplemente necesitamos conectarnos por SSH al master node de nuestro cluster y ejecutar spark-submit --master yarn --deploy-mode cluster [nombreDeMiEjecutable]
- Podemos observar el resultado del experimento utilizando la función print del lenguaje elegido y consultando ese output mediante los logs del job de Spark, que obtenemos al ejecutar spark-submit y que será similar a esta http://clusterName.northeurope.cloudapp.azure.com:8088/cluster/app/application_1462528957014_0002. También podemos acceder desde la url que muestra las apps del cluster, similar a http://clusterName.northeurope.cloudapp.azure.com:8088/cluster/apps
    - También podemos consultar de este modo los logs en caso de que haya algún error durante la ejecución

Todos los ficheros necesarios están disponibles en https://github.com/martinezmiranda/hadooptraining dentro de la carpeta Ejercicio 3

Veremos una posible solución a este ejercicio (utilizando Python) en la tercera sesión online, Visualización en Hadoop IaaS y Power BI, para la que podéis registraros aquí https://goo.gl/dOOUAi. Además, después de la sesión propondremos una aplicación de los nuevos conceptos vistos para ampliar este ejercicio.