

Ejercicio 2

Vamos a continuar el learning path propuesto repasando los conceptos que hemos aprendido de Azure Stream Analytics. Como fuente de datos, junto con este enunciado habéis recibido una solución de Visual Studio que contiene un sencillo proyecto llamado LegendsAPIProducer. Este proyecto genera una aplicación de consola que llama a la API de League of Legends para obtener información de las partidas de un jugador, y las envía a un EventHub desde el que las consumiremos con Stream Analytics.

Lo primero que necesitamos es configurar el entorno, consistente en:

- Un Azure Event Hub, desde el que consumiremos datos en streaming
 - La cadena de conexión a este EventHub tiene que ser configurada en el fichero AppConfig del proyecto LegendsAPIProducer
 - Los valores de esa cadena de conexión los utilizaremos también a la hora de configurarlo como stream input en nuestro job de Stream Analytics
- Una cuenta de Azure Storage, donde almacenaremos los datos de salida
 - El nombre y la clave de conexión tendremos que proporcionárselos como output al job de Stream Analytics
- Una clave para la API de League of Legends
 - Es completamente gratuita y podemos conseguirla desde <https://developer.riotgames.com/sign-in>

El ejercicio consiste en predecir si el jugador cuya partida estamos analizando va a ganar o perder la partida a medida que esta se está jugando

- Crearemos el previamente mencionado job de Stream Analytics y configuraremos el input (desde Azure EventHub) y el output (a Blob Storage)
- Arrancaremos la aplicación LegendsAPIProducer para ir generando datos
- Crearemos una query cuyo resultado será el id y el timestamp de un momento concreto de la partida, junto con una predicción del resultado
 - Esta predicción tomará un valor de -2 a +2, siendo -2 derrota casi asegurada y +2 victoria casi asegurada
 - En aquellos casos en los que la diferencia entre el oro del equipo del jugador y el rival sea menor que +-4000, ignoraremos el evento: la situación está equilibrada
 - Una diferencia de entre +-4000 y +-7500 será un +-1
 - Una diferencia superior a +- 7500 será un +-2
- Volcaremos los resultados a un fichero .csv en el Blob Storage
- Opcionalmente, podemos unir los datos del stream con los datos de referencia contenidos en el fichero referenceData.csv
 - Este fichero contiene los nombres de los campeones disponibles, y puede unirse en base al SummonerChampionID con los datos del stream
 - Podemos tener entonces en cuenta un modificador de +-1 dependiendo del personaje
 - El jugador suele ganar las partidas jugadas con Lux, así que sumaremos 1 a la predicción

- El jugador suele perder las partidas jugadas con Ekko, así que restaremos uno a la predicción
- La aplicación LegendsAPIProducer genera al final del streaming de cada partida la información de si se ganó o se perdió, lo que podemos utilizar para ver si estábamos en lo cierto en nuestra predicción

Todos los ficheros necesarios están disponibles en <https://github.com/martinezmiranda/hadooptraining> dentro de la carpeta Ejercicio 2

Veremos una posible solución a este ejercicio en la segunda sesión online, Machine Learning sobre Hadoop y Azure ML, para la que podéis registraros aquí <https://goo.gl/IOQtIC>. Además, después de la sesión propondremos una aplicación de los nuevos conceptos vistos para ampliar este ejercicio.