Desafío - Spark

- Para realizar este desafío debes haber estudiado previamente todo el material disponibilizado correspondiente a la unidad.
- Una vez terminado el desafío, comprime la carpeta que contiene el desarrollo de los requerimientos solicitados y sube el `.zip` en el LMS.
- Desarrollo desafío:
    - El desafío se debe desarrollar de manera Individual/Grupal.

Sobre los datos del ejercicio

- El archivo `household_power_consumption.txt` recolectó 2075259 mediciones en una casa entre los meses de Diciembre 2006 y Noviembre 2010. Este archivo se encuentra en el bucket del curso con la siguiente dirección : `s3://bigdata-desafio/challenges/u4lec1/household_power_consumption.txt`.

- Atributos del archivo:

Posición	Dato	Formato
0	Fecha	Corresponde a un string con el formato de fecha dd/mm/yyyy
1	Hora	Corresponde a un string con el formato de hora hh:mm:ss
2	Global Activity Power	Corresponde a un número flotante que mide el poder activo en kilowatt, promediado por minutos
3	Global Reactive Power	Corresponde a un número flotante que mide el poder reactivo en kilowatt, promediado por minutos
4	Voltage	Corresponden a un número flotante que mide el voltaje, promediado por minutos
5	Global Intensity	Corresponde a un número flotante que mide la intensidad de la corriente, promediado por minutos
6	Sub Metering 1	Corresponde a un número flotante que mide el consumo energético de la cocina

7	Sub Metering 2	Corresponde a un número flotante que mide el consumo energético de la logia
8	Sub Metering 3	Corresponde a un número flotante que mide consumo energético de caldera y aire acondicionado

- El conjunto de datos presente algunos valores perdidos en la medición (aproximadamente el 1.25\% de las filas). Todas las fechas y timestamps están presentes, pero para algunos timestamps, los valores de medición están perdidos. Un dato perdido estará identificado con el signo `?`. Un ejemplo de observaciones con datos perdidos tendría la siguiente estructura:

```
28/4/2007;23:58:00;?;?;?;?;?;?;?;
28/4/2007;23:59:00;?;?;?;?;?;?;?;
```

Ejercicio 1: Preliminares y datos perdidos

- Genere una instancia en AWS EMR y habilite un puerto dinámico para utilizar un notebook desde JupyterHub.
- Genere los objetos `SparkConf` y `SparkContext`.
- Utilizando `Spark`, importe el archivo `household_power_consumption.txt` a un objeto RDD.
- El primer registro del archivo corresponde a las etiquetas asociadas a cada columna. Elimínalo de su RDD.
- Cada campo dentro del registro está separado por un `;`. Separe cada campo del registro por este carácter.
- Reporte las siguientes métricas:
  - La cantidad total de registros antes de eliminar los datos perdidos.
  - La cantidad total de registros nulos.
  - La cantidad de registros nulos por año. ¿En qué año hubo más registros nulos?
  - La cantidad de registros nulos por mes para el año con una mayor cantidad de registros nulos. ¿En qué mes hubo más registros nulos?

Ejercicio 2: Preparación de los datos

- Genere un objeto donde se encuentren sólo aquellos datos sin registros perdidos.
- Con el objeto sin datos perdidos, genere un objeto RDD donde los registro tengan la siguiente estructura:

```
[(2006, 12, 18, 4.464, 0.136, 234.66, 19.0, 0.0, 37.0),
(2006, 12, 18, 1.944, 0.084, 236.56, 8.2, 0.0, 2.0),
(2006, 12, 9, 3.706, 0.062, 237.81, 15.6, 1.0, 38.0),
(2006, 12, 1, 0.204, 0.0, 244.48, 0.8, 0.0, 0.0),
(2006, 12, 10, 1.34, 0.092, 238.72, 5.6, 0.0, 0.0)]
```

Donde cada registro tendrá la siguiente composición: (`año, mes, dia, global_activity_power, global_reactive_power, voltage, global_intensity, submetering_1, submetering_2, submetering_3`).

- Por defecto todos los datos dentro de este objeto serán considerados como strings. Conviértalos al tipo de dato pertinente (`int` o `float`).

- Devuelva con collect las primeras 5 observaciones.

Ejercicio 3: Patrones globales

- Extraiga la media, desviación estándar e intervalos de confianza (a ± .5 desviaciones estándar) para las columnas `global_activity_power`, `global_reactive_power`, `voltage` y `global_intensity`.

- Reporte sus resultados.

Ejercicio 4: Patrón temporal

- Reporte el promedio de `global_activity_power` a nivel mensual y anual. Genere un objeto con la acción `collectAsMap()` del RDD procesado.

- Reporte todas aquellos registros que presenten un promedio de `global_activity_power` superior a .5 desviaciones estándares. ¿A qué meses y años corresponden?

- Reporte todos aquellos registros que presenten un promedio de `global_activity_power` inferior a .5 desviaciones estándares. ¿A qué meses y años corresponden?

Ejercicio 5: Consumo energético general

- El siguiente objetivo es calcular el consumo energético general para cada registro en el RDD. La fórmula se detalla a continuación:

$$ConsumoEnergeticoGeneral = Global\ Activity\ Power\ \frac{1000}{60} - SubMetering1 - SubMetering2 - SubMetering3$$

- Implemente la función para todos los registros.

- Genere el RDD con el consumo energético general promedio por Año, Mes y Día.

- Reporte el consumo energético general para todos los días de Octubre del 2008.

(Opcional) Ejercicio 6: Submetering

Tip: *Para este ejercicio, utilice* `join` *para unir dos RDD.*

- Identifique el consumo promedio por mes y año para cada medición de submetering (1, 2 y 3).
- Identifique aquellos registros donde el consumo promedio del submetering 3 sea mayor al submetering 1.
- Identifique aquellos registros donde el consumo promedio del submetering 2 sea mayor al submetering 1.
- Identifique aquellos registros donde el consumo promedio del submetering 3 sea mayor al submetering 2.
- Identifique aquellos registros donde el consumo promedio del submetering 2 sea mayor al submetering 3.

Ejercicio 7: Descarga de resultados y eliminación de la instancia

- Saque un screenshot del término de la instancia AWS EMR.
- Descargue el notebook desde JupyterHub.
- Comprima los archivos en un `.zip`.