UdeA
Analytics &
DataScience

PROGRAMACIÓN SOBRE GRANDES VOLUMENES DE DATOS

RDD

UNIVERSIDAD
DE ANTIOQUIA
1 8 0 3

Magister - Efraín Alberto Oviedo
alberto.oviedo@udea.edu.co

UNIVERSIDAD DE ANTIOQUIA
FACULTAD DE INGENIERÍA
ESPECIALIZACIÓN EN ANALÍTICA Y CIENCIA DE DATOS

AGENDA

UdeA
Analytics &
DataScience

1. **RDD**
   - ❑Acciones
   - ❑Transformaciones
   - ❑Filtros
2. Ejemplos
3. Ejercicios

Qué es un RDD

Un RDD (Dataset Distribuido Resiliente) es una colección de elementos que es tolerante a fallos y que es capaz de operar en paralelo.

Se considera como la principal abstracción de datos de Spark definido desde la Versión 1.0, y permite que los datos se puedan dividir y almacenar en los nodos del clúster.

Características

- Inmutable: Una vez creado **NO** puede ser modificados
- Tolerancia a fallas: Ya que los RDD están particionados y distribuidos en los nodos del clúster, si un nodo falla se puede recuperar los datos consultando otro nodo
- Evaluación perezosa: Las transformaciones realizada al RDD no se ejecutan de inmediato, se almacenan en un DAG (Grafo Acíclico dirigido) y se resuelven cuando sea necesario resolverlas
- Se almacena la secuencia de transformaciones con el fin de poder recuperarse si un nodo falla

Crear un RDD

Tenemos dos opciones para crear RDDs

- Paralelizar una colección existente

Crear un RDD usando el método parallelize del SparkContext

```
[3]  num = [0, 1, 2, 3, 4, 5]
     type(num)
```

```
     list
```

```
[4]  numRdd=sc.parallelize(num)
     type(numRdd)
```

```
     pyspark.rdd.RDD
```

Puede utilizarse un segundo atributo para indicar el número de particiones que se desea crear dela variable

sc.parallelize(num,10)

Crear un RDD

- Hacer referencia a un conjunto de datos en un sistema de almacenamiento externo

Crear un RDD usando el método textFile del SparkContext

```
textRdd = sc.textFile("local/data/animales.txt")
type(textRdd)
```

```
pyspark.rdd.RDD
```

Se puede crear el RDD a partir de cualquier fuente de almacenamiento compatible con Hadoop como: HDFS, Hbase, Cassandra, AWS S3

Evaluación perezosa

Vamos a crear un RDD a partir de un archivo que no existe. Esto solo generará error cuando se ejecute el grafo

```
[8]  textRdd = sc.textFile("local/data/archivo_no_existe.txt")
```

No se genera ningún error a pesar de que el archivo no existe.

Ahora apliquemos un collect() para obligar a que se ejecute el grafo

```
[9]  textRdd.collect()
```

```
---------------------------------------------------------------------------
Py4JJavaError                             Traceback (most recent call last)
<ipython-input-9-d68fda5930d1> in <module>()
----> 1 textRdd.collect()
```

AGENDA

UdeA
Analytics & DataScience

1. RDD
   - ❑ **Transformaciones**
   - ❑ Acciones
   - ❑ Persistencia
2. Ejemplos
3. Ejercicios

Transformaciones

Las transformaciones se ejecutan sobre un conjunto de datos y generan un nuevo conjunto de datos

Transformación	Descripción
Map(func)	Entrega un nuevo conjunto de datos resultado de pasar cada elemento por la función indicada
Filter(func)	Entrega un conjunto de datos con los elementos en los que la función retorne verdadero
flatMap(func)	Funciona similar al map solo que cada elemento de entrada puede asignarse a 0 o mas salidas
sample(withReplacement, fraction, seed)	Muestrea una fracción de los datos, con o sin reemplazo, usando una semilla generadora de números aleatorios
union(otherDataset)	Entrega un nuevo conjunto de datos que incluye los datos actuales y los que se pasen como argumento

https://spark.apache.org/docs/latest/rdd-programming-guide.html

Transformaciones

Transformación	Descripción
intersection(otherDataset)	Entrega un nuevo conjunto de datos formado por la intersección entre el conjunto de datos actual y el que se pasa como argumento
distinct([numTasks])	Entrega un nuevo conjunto de datos formado por los elementos diferentes del conjunto de datos actual
groupByKey([numTasks])	Recibe un conjunto tuplas (clave valor), y entrega un nuevo conjunto de tuplas (calve, secuencia valores)
reduceByKey(func)	Recibe un conjunto de tuplas (clave, valor) y devuelve un nuevo conjunto de tuplas con la clave y la reducción de los valores por clave según la función que se pasa como argumento
sortByKey(type)	Entrega un conjunto de tuplas clave valor ordenados ascendente o descendentemente según se solicite
join(otherDataset, [numTasks])	A partir de tuplas (K,V) y (K,W) entrega como resultado tuplas (K,(V,W))

AGENDA

UdeA
Analytics &
DataScience

1. RDD
   - ❑Transformaciones
   - ❑**Acciones**
   - ❑Persistencia
2. Ejemplos
3. Ejercicios

Acciones

Las acciones se aplican sobre un conjunto de datos y devuelven un valor o un nuevo conjunto de datos

Acción	Descripción
Reduce(func)	Agrega los elementos de un conjunto de datos aplicando sobre ellos una función
Collect()	Devuelve todos los elementos del conjunto de datos como un array
Count()	Entrega el número de elementos disponibles en el conjunto de datos
First()	Entrega el primer elemento del conjunto de datos
Take(n)	Entrega los primeros n elementos del conjunto de datos
saveAsTextFile(path)	Almacena el conjunto de datos en un archivo de texto en la ruta path
countByKey()	Devuelve un conjunto de datos representado por tuplas clave, valor. Donde el valor entregado será la suma de los elemento de la misma clave
Foreach(func)	Ejecuta la función indicada en cada elemento del conjunto de datos

https://spark.apache.org/docs/latest/api/python/pyspark.html#pyspark.RDD

AGENDA

UdeA
Analytics &
DataScience

1. RDD
   - ❑ Transformaciones
   - ❑ Acciones
   - ❑ **Persistencia**
2. Ejemplos
3. Ejercicios

Persistencia

- Teniendo en cuenta que la operación de los RDD es perezosa, la persistencia permite almacenar los RDD después de la primera vez que se calculan las operaciones

- Esto permite que las operaciones futuras se realice mucho mas rápido

- Almacenar los datos en cache es clave para algoritmos iterativos

- Para persistir un RDD se utilizan los métodos: cache (persistir en memoria), persist (permite seleccionar el tipo de persistencia)

Persistencia

Tipos de persistencia de datos

Persistencia	Descripción
MEMORY_ONLY	Es el nivel por defecto. Se almacenan en memoria como objetos Java
MEMORY_AND_DISK	Almacena en memoria y en caso de que requiera mas espacio almacenará las particiones restantes en disco
DISK_ONLY	Almacena solo en disco

AGENDA

UdeA
Analytics &
DataScience

1. RDD
   - ❑Transformaciones
   - ❑Acciones
   - ❑Persistencia
2. **Ejemplos**
3. Ejercicios

Ejemplos

- Palabras mas frecuentes en una obra de literatura

- Titanic

AGENDA

UdeA
Analytics &
DataScience

1. RDD
   - ❑Transformaciones
   - ❑Acciones
   - ❑Persistencia
2. Ejemplos
3. **Ejercicios**

Ejemplos

Se dispone de un dataset que contiene información relacionada con el hurto a personas en Colombia, son mas de 100.000 casos de hurtos cometidos en el país en la última época. Cada registro presenta la siguiente información

- Departamento
- Municipio
- Día
- Hora
- Zona (urbana, rural)
- Arma empleada
- Movil agresor
- Movil víctima
- Edad víctima
- Sexo víctima

Ejemplos

Utilizando RDDs, resuelva a las siguientes inquietudes:

1. Top 10 de los municipios de Antioquia que presentan mayor y menor número de hurtos
2. Tipos de armas más utilizadas en zona urbana y rural
3. Promedio de edad de las víctimas por departamento
4. Tipo de vehículo más utilizado para los hurtos en los fines de semana
5. Promedio de casos de hurto por sexo para cada día de la semana
6. Municipio de Colombia que presenta mayor número de hurtos a mujeres mayores de 40 años