Summarizing Data Along Dimensions

overview :

- Representing records as Pair RDDS
- Summarizing Pair RDDS using reduceByKey and CombineByKey
- Merging data from Separate RDDS

Modelling Traffic Patterns

How many cars travel this road on a given day?

which were the days that saw the highest traffic?

were there Dodgers games on the days with high traffic?

what's the average traffic like on a game day vs a non-game day?

In all of these basically we have a metric on one side and the dimensions along which you are summarizing on the other side.

you can think of each record in your dataset as a key-value pair where the key is the dimension and the value is the metric.

So therefore if there was a way to work with records by inherently treating them as pairs, that would be very helpful.

That's exactly where the different types of RDDs come in. Till now we have been dealing just with basic RDDS where we treat each record in the RDD as a single object. Any operation that we perform on the RDD treats the entire object or record as a single entity.

on the other hand, you have another type of RDD called pair RDD where each element is treated as a key-value pair.

To treat an RDD as a pair RDD, all you need to have is records which are tuples with two objects.

Pair RDD

Each record is a tuple with 2 objects

(Airline , Delay)

(City , Sales)

(Word , Count)

In Python to create a pair RDD you just need to make sure that each record is a tuple.

Then you can apply any of the pair RDD special functions on that RDD.

we can summarize pair RDDs by the Keys.

You can use the reduceByKey or CombineByKey functions to do this. These functions basically take an RDD and Combine Values, which have the Same Key in a specified way.

Pair RDD

Summarize by Keys

→ reduceByKey

→ CombineByKey

Merge by Keys

- Join
- left outer Join
- right outer Join