Chapter 3. A Tour of Spark's Toolset

Spark-submit: sends your application code to cluster and launches it to execute there.

Datasets API - for writing statically typed code in Java and Scala. The Dataset API is not available in Python and R, because those languages are dynamically typed.

Structured Streaming - take the same operations that you perform in batch mode using Spark's structured APIs,and run them in a streaming fashion

Window function

- includes all data from each day in aggregation
- Window over the time–series column in our data
- Helpful tool for manipulating date and timestamps
- Can specify our requirements in a more human form (via intervals) -Spark will group all of them together for us

Streaming:

- readStream
- maxFilesPerTrigger
- action outputs to an in-memory table that will update after each trigger
- trigger is based on an individual file (the readoption that we set)

Shouldn't use either of these streaming methods in production, but they do make for convenient demonstration of Structured Streaming's power. Notice how this window is built onevent time, as well, not the time at which Spark processes the data. This was one of the shortcomings of Spark Streaming that Structured Streaming has resolved

MLlib

- allows for preprocessing, munging, training of models, and making predictions at scale on data. You can even use models trained in MLlib to make predictions in Structured Streaming.
- Spark's machine learning API performs tasks, from classification to regression, and clustering to deep learning
- All machine learning algorithms in Spark take as input a Vector type, which must be a set of numerical values

K-means

- clustering algorithm in which “k” centers are randomly assigned within the data.
- Points closest to that point are then “assigned” to a class and the center of the assigned points is computed.
- Center point is the centroid.
- Label the points closest to that centroid, to the centroid’s class, and shift the centroid to the new center of that cluster of points.
- Repeat this process for a finite set of iterations or until convergence (our center points stop changing)

Spark, training machine learning models is a two-phase process.

1. initialize an untrained model
2. Train it.

There are always two types for every algorithm in MLlib’s DataFrame API.

- They follow the naming pattern of Algorithm, for the untrained version
- Algorithm Model for the trained version.

In our example, this is KMeans and then KMeansModel.Estimators in MLlib’s DataFrame API share roughly the same interface that we saw earlier with our preprocessing transformers like the StringIndexer. Makes training an entire pipeline (including model) simple.

Python object manipulation via Resilient Distributed Datasets (RDDs). DataFrame operations are built on RDDs and compile down to these lower-level tools for convenient and efficient distributed execution. There are some things that you might use RDDs for, especially when you’re reading or manipulating raw data, but for the most part you should stick to the Structured APIs.