18-NYCTaxiCabTripDask

August 10, 2020

1 Dask dataframes on HDFS

To use Dask dataframes in parallel across an HDFS cluster to read CSV data. We can coordinate these computations with distributed and dask.dataframe.

As Spark, Dask can work in cluster mode. You can use the dask module dask_jobqueue to launch a Dask cluster with the job manager SLURM.

[1]:

```python
from dask_jobqueue import SLURMCluster

cluster = SLURMCluster(cores=16,
                       queue='test',
                       project='myproject',
                       memory="16GB",
                       walltime="01:00:00")
```

The cluster generates a traditional job script and submits that an appropriate number of times to the job queue. You can see the job script that it will generate as follows:

[2]:

```python
print(cluster.job_script())
```

```
#!/usr/bin/env bash

#SBATCH -J dask-worker
#SBATCH -p test
#SBATCH -A myproject
#SBATCH -n 1
#SBATCH --cpus-per-task=16
#SBATCH --mem=15G
#SBATCH -t 01:00:00

/usr/share/miniconda3/envs/big-data/bin/python -m distributed.cli.dask_worker
tcp://10.1.0.4:41943 --nthreads 4 --nprocs 4 --memory-limit 4.00GB --name name
--nanny --death-timeout 60
```

Access to the cluster using following lines:

```
import dask.dataframe as dd
from dask.distributed import Client
```

```
client = Client(cluster)
```

nyc2014 is a dask.dataframe objects which present a subset of the Pandas API to the user, but farm out all of the work to the many Pandas dataframes they control across the network.

```
nyc2014 = dd.read_csv('/opt/datasets/nyc-data/2014/yellow*.csv',
parse_dates=['pickup_datetime', 'dropoff_datetime'],
skipinitialspace=True)
nyc2014 = c.persist(nyc2014)
progress(nyc2014)
```

1.0.1 Exercises

- Display head of the dataframe
- Display number of rows of this dataframe.
- Compute the total number of passengers.
- Count occurrences in the payment_type column both for the full dataset, and filtered by zero tip (tip_amount == 0).
- Create a new column, tip_fraction
- Plot the average of the new column tip_fraction grouped by day of week.
- Plot the average of the new column tip_fraction grouped by hour of day.

Dask dataframe documentation

[3]:

```
# import dask.dataframe as dd
# from distributed import Client, progress
#
# c = Client('127.0.0.1:8786')
# nyc2014 = dd.read_csv('hdfs://svmass2.mass.uhb.fr:54310/user/datasets/nyc-tlc/
 ↪2014/yellow*.csv',
# parse_dates=['pickup_datetime', 'dropoff_datetime'],
# skipinitialspace=True)
#
# nyc2015 = dd.read_csv('hdfs://svmass2.mass.uhb.fr:54310/user/datasets/nyc-tlc/
 ↪2015/yellow*.csv',
# parse_dates=['tpep_pickup_datetime', 'tpep_dropoff_datetime'])
# nyc2014, nyc2015 = c.persist([nyc2014, nyc2015])
#
# progress(nyc2014, nyc2015)
```