Bootstrapping Big Data with Spark SQL and Data Frames

Brock Palen | @brockpalen | brockp@umich.edu
http://arc-ts.umich.edu
In Memory
- Small to modest data
- Interactive or batch work
- Might have many thousands of jobs
- Excel, R, SAS, Stata, SPSS

In Server
- Small to medium data
- Interactive or batch work
- Hosted/shared and transactional data
- SQL / NoSQL
- Hosted data pipelines
- iRODS / Globus
- Document databases

Big Data
- Medium to huge data
- Batch work
- Full table scans
- Hadoop, Spark, Flink
- Presto, HBase, Impala
Coming Soon: Bigger Big Data
Spark runs on Hadoop, Mesos, standalone, or in the cloud. It can access diverse data sources including HDFS, Cassandra, HBase, and S3.

http://spark.apache.org
What we will run

SELECT
    author, subreddit_id, count(subreddit_id) AS posts
FROM
    reddit_table
GROUP BY author, subreddit_id
ORDER BY posts DESC
Spark Submit Options

Spark-submit / pyspark takes R, Python, or Scala

```
pyspark \n--master yarn-client \n--queue training \n--num-executors 12 \n--executor-memory 5g \n--executor-cores 4
```

- pyspark for interactive
- spark-submit for scripts
Reddit History: August 2016 -- 279,383,793 Records
## Data Format Matters

<table>
<thead>
<tr>
<th>Format</th>
<th>Type</th>
<th>Size</th>
<th>Size w/Snappy</th>
<th>Time Load / Query</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text / JSON / CSV</td>
<td>1.7 TB</td>
<td>2,353 s / 1,292 s</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parquet</td>
<td>Column</td>
<td>229 GB</td>
<td>117 GB</td>
<td>3.8 s / 22.1 s</td>
</tr>
</tbody>
</table>

Other types:
- Avro
- ORC
- Sequence File

Parquet Format

Use: parquet-tools
To look at parquet data and schema off Hadoop filesystems systems.

Details: https://parquet.apache.org/
Data Frames

- Popularized by R and Python Pandas
- Spark Data Frames and Pandas share most of their API
- Similar to SQL Object Relational Models
  - `le: table.column  table.function()` etc
Getting Started

```python
reddit = sqlContext.read.parquet("reddit_full.parquet")

reddit.show()

reddit.count()

reddit.printSchema()

reddit.select("over_18").show()

reddit.filter(reddit['over_18'] == 1 ).count()
```
Started Ctd.

```python
reddit.groupBy("author","subreddit_id").count().show()

from pyspark.sql.functions import *

reddit.groupBy("author","subreddit_id").agg(count("subreddit_id")).show()
```
Aggregations Ctd.

```python
reddit.groupBy("author","subreddit_id").agg(count("subreddit_id")).alias("posts")).orderBy("posts", ascending=[0]).show()

reddit.groupBy("author","subreddit_id").agg(count("subreddit_id")).alias("posts")).orderBy("posts", ascending=[0]).filter(reddit.author != '[deleted]').show()
```
Data Frames and Spark SQL

- SQL in Spark is a feature of Data Frames
  - They can be mixed,
  - Make a table with `<dataframe>.registerTempTable("tablename")`
Spark SQL

```python
reddit.registerTempTable("reddit_table")

results = sqlContext.sql("SELECT author, subreddit_id, count(subreddit_id) AS posts FROM reddit_table WHERE author NOT LIKE '[deleted]' GROUP BY author, subreddit_id ORDER BY posts DESC")

results.show()

results.write.parquet("myresults")
```
Other Handy Functions

# Get query plan
<statement>.explain()

# save data partitioned by column
results.write.parquet("myresults_p.parquet", "overwrite", "over_18")

Read: https://spark.apache.org/docs/1.6.2/api/python/pyspark.sql.html

Use Sqoop to import data from DBMS: http://sqoop.apache.org/
Thank You - Contact

- hpc-support@umich.edu
- http://arc-ts.umich.edu/
- @ARCTS_UM
- brockp@umich.edu
- http://myumi.ch/aV7kz