**CSE 587 HW # 4**

**Name: Daniel Nazareth**
**UbitName: dnazaret**
**Email: dnazaret@buffalo.edu**

***SUMMARY:*** The project uses the Apache HBASE 0.98.11 column NoSQL database layered on Hadoop 2.6.0 to quickly and efficiently compute the most and least volatile stocks from small,medium and large sized NASDAQ stock datasets.

***RATIONALE FOR USING HBASE:*** The biggest advantage of HBase is that it offers a highly scalable way to store huge amounts of sparsely, often very loosely related data, yet still maintain the ability to retrieve information efficiently. Just as Hadoop maintains name and datanodes, so too does Hbase keep master and slave servers to facilitate this process.

**To summarise, HBase offer the following benefits:**

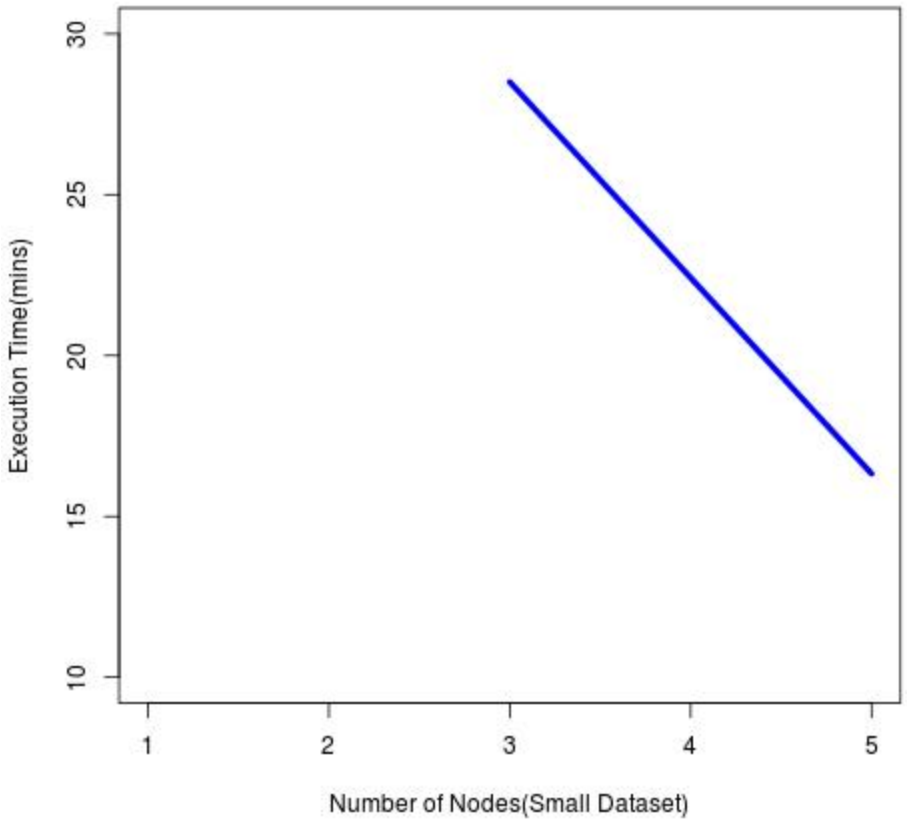
- Shorter,simpler and much more readable code.
- Shorter development and QA cycles leading to quicker releases to production.
- Highly scalable way of storing very sparse data

***RESULTS:*** The results are summarised in tabular and graphical form below for each of the small,medium and large datasets. This can be verified by running the attached code at the command line as well.

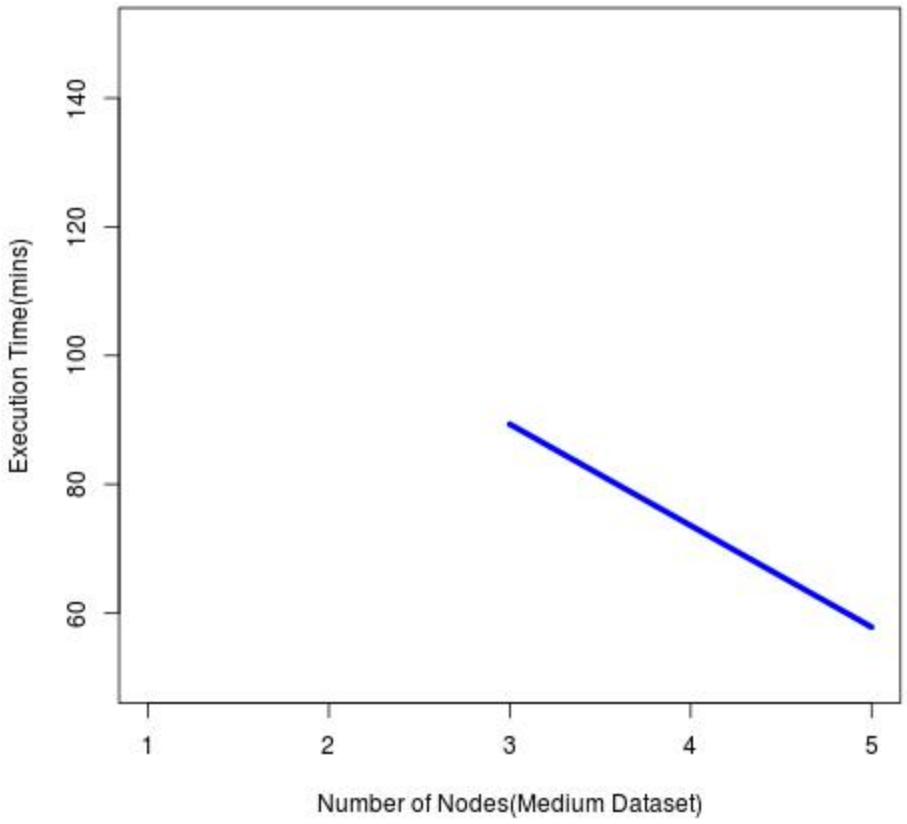
***HBASE RESULTS***

| PROBLEM SIZE | EXECUTION TIME(3 NODES,36 CORES) | EXECUTION TIME(5 NODES,60 CORES) |
|---|---|---|
| ***Small*** | 28 mins,36 seconds | 16 mins, 19 seconds |
| ***Medium*** | 89 mins,21 seconds | 57 mins,51 seconds |
| ***Large*** | 271 mins, 14 seconds | 201 mins,11 seconds |

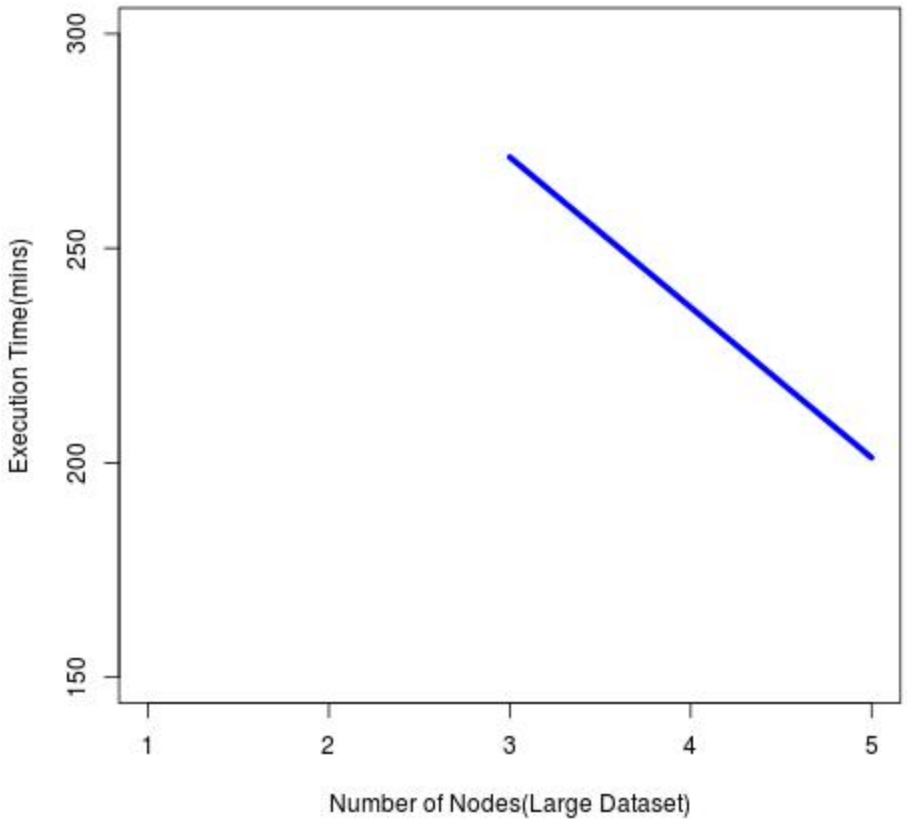
**SMALL DATASET:**




**MEDIUM DATASET:**




**LARGE DATASET:**




**COMPARISON OF MAP REDUCE/PIG/HIVE:** We see from the below tabular and graphical comparison that for small datasets, all 4 methods are closely matched with Hive being the runaway winner. For medium and large datasets Pig and Hive in particular offer significantly superior performance. However Map Reduce performance can be significantly improved by assigning more computing nodes, same as Hbase-this cannot be done for Pig/Hive.

| PROBLEM SIZE | AVERAGE EXECUTION TIME(MAP REDUCE) | AVERAGE EXECUTION TIME(PIG) | AVERAGE EXECUTION TIME(HIVE) | AVERAGE EXECUTION TIME(HBASE) |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| ***Small*** | 21.66 minutes | 27.8 mins | 13.76 minutes | 22.41 mins |
| ***Medium*** | 65 mins | 29.6 mins | 24.4 minutes | 73.55 mins |
| ***Large*** | 120 mins | 31.2 mins | Not Available | 236.22 mins |

***NOTES/ISSUES FACED:***

- **We observe that additional nodes makes little or no difference to execution times for both Pig and Hive but a huge difference for Map Reduce and Hbase which scale well for large sets over more nodes. Times scale by small factors over dataset sizes but barely at all over number of nodes allocated. This would seem to indicate that for a given Hive or Pig script, computational efficiency cannot be reduced beyond a certain point, no matter how many resources are allocated whereas this is configurable with Hbase**

***REFERENCES***


*http://hbase.apache.org/0.94/book/mapreduce.example.html*

*https://hbase.apache.org/apidocs/org/apache/hadoop/hbase/client/Result.html*

*http://sujee.net/2011/04/10/hbase-map-reduce-example/#.VTGP3XVdUVw*