CSE 587

Data Intensive Computing

Homework 3

Stock Volatility Computation: Pig vs Hive

Vidyadhar Reddy Annapureddy

vannapur@buffalo.edu (50134358)

A. Pig Latin

Pig is a high-level platform for creating MapReduce programs used with Hadoop. The language Pig Latin abstracts the programming from the Java MapReduce idiom into a notation which makes MapReduce programming high level, similar to that of SQL for RDBMS systems. Pig Latin can be extended using UDF (User Defined Functions) which the user can write in Java, Python etc. and then call directly from the language. Pig uses lazy evaluation, uses extract, transform, load (ETL), is able to store data at any point during a pipeline, declares execution plans, and supports pipeline splits, thus allowing workflows to proceed along DAGs instead of strictly sequential pipelines.

The implementation uses Pig Latin and two User Defined Functions (written in Java) to calculate volatility index for the stocks.

a. UDF1, included in **myUDFS.jar**, calculates the as input every stock data for each day of a month and returns the stock name and x_i value as tab-separated string values.
b. UDF2, included in **Vol.jar,** computes the volatility value for each stock from x_i obtained previously for each month.

Although, the same computation could have been done without inclusion of UDFs in Pig implementation but this particular choice was made considering my comfort levels with Java and ease of understanding.

Execution Time: PIG

	Small dataset			**Medium dataset**			**Large dataset**		
Nodes	**1**	**2**	**4**	**1**	**2**	**4**	**1**	**2**	**4**
JobID	3603941	3603927	3600447	3603251	3603927	3600451	3600457	3597408	3597439
Cores	12	24	48	12	24	48	12	24	48
Time (sec)	593	491	458	942	737	688	2379	1971	1840

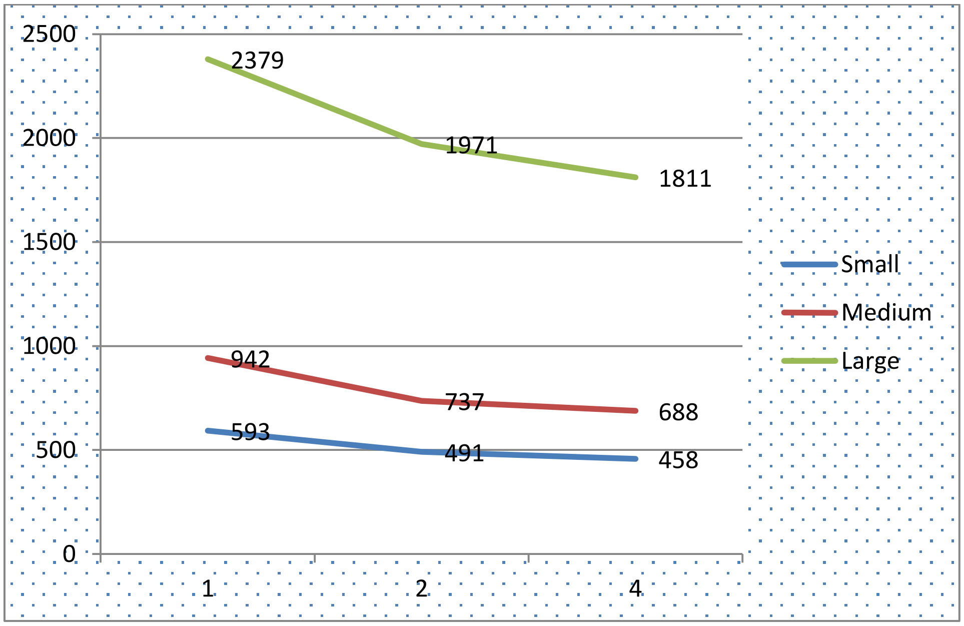

Figure1: Speed Plot describing Pig implementation time (in seconds) vs No. of nodes

Stock Volatility Analysis Results with PIG

i. Small dataset

Top 10 stocks Min Volatility	
Stock	**Volatility Index**
LDRI	0.000514933658210
GAINO	0.000565007416050
VGSH	0.001301490618956
MBSD	0.002500045910462
TRTLU	0.003478105118699
AGZD	0.003938593878697
SKOR	0.003948740216630
CADT	0.004156636196742
AXPWW	0.004438837295584
VCSH	0.004637760185632

Top 10 stocks Max Volatility	
Stock	**Volatility Index**
ACST	9.271589761859980
NETE	5.396253961502240
XGTI	4.542344311472950
TNXP	3.248332196781860
EGLE	3.022206537901310
PTCT	1.846253701581770
GOGO	1.779342175186130
MEILW	1.718813406576530
ROIQW	1.396532083959140
CFRXZ	1.079268244968940

ii. Medium dataset

Top 10 stocks Min Volatility	
Stock	**Volatility Index**
LDRI-1	0.000514933658210
LDRI-2	0.000514933658210
LDRI-3	0.000514933658210
GAINO-2	0.000565007416050
GAINO-3	0.000565007416050
GAINO-1	0.000565007416050
VGSH-3	0.001301490618956
VGSH-2	0.001301490618956
VGSH-1	0.001301490618956
MBSD-1	0.002500045910462

Top 10 stocks Max Volatility	
Stock	**Volatility Index**
ACST-3	9.271589761859980
ACST-1	9.271589761859980
ACST-2	9.271589761859980
NETE-3	5.396253961502240
NETE-2	5.396253961502240
NETE-1	5.396253961502240
XGTI-3	4.542344311472950
XGTI-2	4.542344311472950
XGTI-1	4.542344311472950
TNXP-2	3.248332196781860

iii. Large dataset

Top 10 stocks Min Volatility	
Stock	**Volatility Index**
LDRI-9	0.000514933658210
LDRI-3	0.000514933658210
LDRI-1	0.000514933658210
LDRI-10	0.000514933658210
LDRI-8	0.000514933658210
LDRI-2	0.000514933658210
LDRI-7	0.000514933658210
LDRI-5	0.000514933658210
LDRI-6	0.000514933658210
LDRI-4	0.000514933658210

Top 10 stocks Max Volatility	
Stock	**Volatility Index**
ACST-7	9.271589761859980
ACST-6	9.271589761859980
ACST-1	9.271589761859980
ACST-2	9.271589761859980
ACST-4	9.271589761859980
ACST-5	9.271589761859980
ACST-9	9.271589761859980
ACST-8	9.271589761859980
ACST-3	9.271589761859980
ACST-10	9.271589761859980

B. HIVE

Apache Hive is a data warehouse infrastructure built on top of Hadoop for providing data summarization, ad-hoc queries, and analysis of large datasets stored in Hadoop compatible file systems. Hive is a data warehouse system for Hadoop that. Hive enables developers not familiar with MapReduce to write data queries that are translated into MapReduce jobs in Hadoop. Although there are quite a few limitations of HIVE, like no support for *Update*, *Insert* single rows, *Delete* commands, and limited built-in functions, it serves the purpose of computing stock volatility index just fine with execution for large dataset completing in just over 30 minutes.

Execution Time: HIVE

	Small			**Medium**			**Large**		
Nodes	**1**	**2**	**4**	**1**	**2**	**4**	**1**	**2**	**4**
JobID	3603917	3603735	36033580	3603062	3603815	3603736	3603345	3595827	3603062
Cores	12	24	48	12	24	48	12	24	48
Time (sec)	725	614	499	1170	887	721	2352	2010	1840

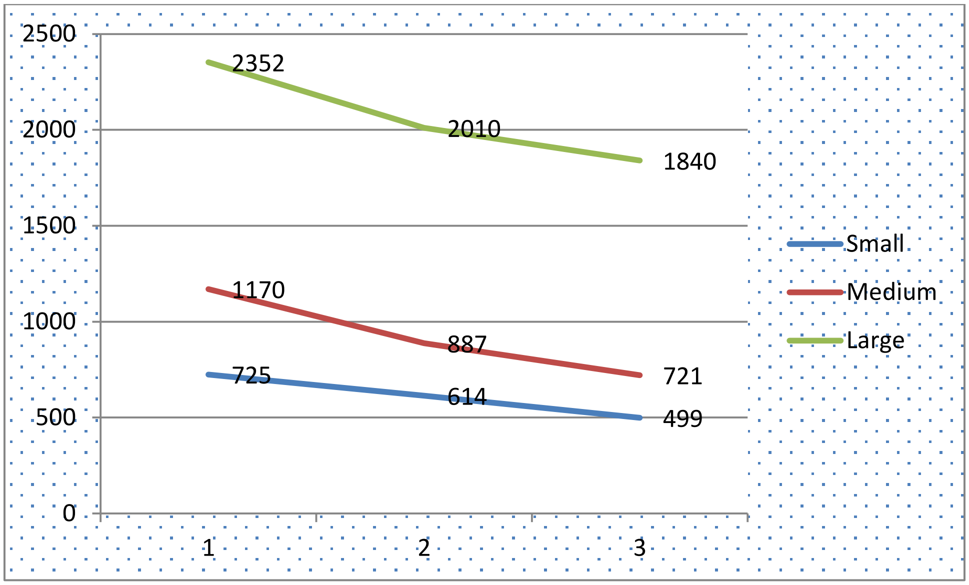

Figure2: Speed Plot describing Hive implementation time (in seconds) vs No. of nodes

Stock Volatility Analysis Results with HIVE

i. Small Dataset

Top 10 stocks Min Volatility	
Stock	**Volatility Index**
LDRI	0.000514916
GAINO	0.00056502
VGSH	0.001301492
MBSD	0.002500073
TRTLU	0.003478115
AGZD	0.003938591
SKOR	0.003948777
CADT	0.004156634
AXPWW	0.004438757
VCSH	0.004637763

Top 10 stocks Max Volatility	
Stock	**Volatility Index**
ACST	9.271588841
NETE	5.396253906
XGTI	4.542344103
TNXP	3.24833191
EGLE	3.022206484
PTCT	1.846253782
GOGO	1.779342131
MEILW	1.718813347
ROIQW	1.396532052
CFRXZ	1.079268227

ii. Medium Dataset

Top 10 stocks Min Volatility	
Stock	**Volatility Index**
LDRI-1	0.00051493365821
LDRI-2	0.00051493365821
LDRI-3	0.00051493365821
GAINO-2	0.00056500741605
GAINO-3	0.00056500741605
GAINO-1	0.00056500741605
VGSH-3	0.001301490618956
VGSH-2	0.001301490618956
VGSH-1	0.001301490618956
MBSD-1	0.002500045910462

Top 10 stocks Max Volatility	
Stock	**Volatility Index**
ACST-3	9.27158976185998
ACST-1	9.27158976185998
ACST-2	9.27158976185998
NETE-3	5.39625396150224
NETE-2	5.39625396150224
NETE-1	5.39625396150224
XGTI-3	4.54234431147295
XGTI-2	4.54234431147295
XGTI-1	4.54234431147295
TNXP-2	3.24833219678186

iii. Large Dataset

Top 10 stocks Min Volatility	
Stock	**Volatility Index**
LDRI-9	0.000514916
LDRI-1	0.000514916
LDRI-10	0.000514916
LDRI-2	0.000514916
LDRI-3	0.000514916
LDRI-4	0.000514916
LDRI-5	0.000514916
LDRI-6	0.000514916
LDRI-7	0.000514916
LDRI-8	0.000514916

Top 10 stocks Max Volatility	
Stock	**Volatility Index**
ACST-1	9.271588841
ACST-9	9.271588841
ACST-8	9.271588841
ACST-7	9.271588841
ACST-6	9.271588841
ACST-5	9.271588841
ACST-4	9.271588841
ACST-3	9.271588841
ACST-2	9.271588841
ACST-10	9.271588841

Pig vs HIVE vs MR

Hive and Pig are frameworks which utilize Hadoop underneath. Both ultimately result into single/multiple MapReduce jobs creations to get the required output. Hive is recommended for people how are familiar to SQL and Pig is recommended for People are familiar with scripting languages. We need MapReduce when we need very deep level and fine grained control on the way we want to process our data. Sometimes, it is not very convenient to express what we need exactly in terms of Pig and Hive queries.

PIG commands are submitted as MapReduce jobs internally. An advantage PIG has over MapReduce is that the former is more concise. A 200 lines Java code written for MapReduce can be reduced to 10 lines of PIG code. A disadvantage PIG has: it is bit slower as compared to MapReduce as PIG commands are translated into MapReduce prior to execution.

Hive, on the other hand, is very convenient for those who are good at SQL. It has good support for structured data. Currently support database schema and views like structure Support concurrent multi users, multi session scenarios. Among the biggest cons: Performance degrades as data grows bigger not much to do, memory over flow issues; can't do much with it. Hierarchical data is a challenge. Un-structured data requires UDF like component Combination of multiple techniques could be a nightmare dynamic portions in case of big data.

Conclusion

Based on the results hereby generated by the implementation of stock volatility index for NASDAQ stocks for over 3 years, Pig implementation proves to be the fastest among Pig and HIVE. Having said that, it is to be noted that Pig implementation uses Java based UDFs for minor calculations and the results may vary w.r.t pure pig implementation. Both Pig and HIVE internally invoke the native MR job for computation of tasks on distributed nodes on the cluster.