

AOS Assignment - 5

House Pricing Analysis Using PySpark

Objective

The main aim of this analysis is to glean insights from the House Pricing Dataset which contains information on all registered property sales in England and Wales sold for full market value.

Dataset

The dataset provides details about various properties, including their transaction identifier, price, and country, etc. For this assignment, we focus on three main fields: 'Transaction unique identifier', 'Price', and 'Country'.

Methodology

The analysis was performed using PySpark, a powerful framework for big data processing. We primarily utilized its DataFrame API to perform operations on the dataset.

Question 1: Second Highest Value Transaction('Price') in Selected Countries

Procedure:

1. Loaded the dataset into a DataFrame.
2. Filtered the records for the countries 'GREATER LONDON', 'CLEVELAND', and 'ESSEX'.
3. Aggregated the prices by country to find the maximum price and sorted in descending order.
4. Extracted the second record to get the second highest price and its corresponding country.

Results:

Second highest transaction is 20000000 which is transacted in GREATER LONDON.

Question 2: Country with the Second Most Transactions

Procedure:

1. Grouped the dataset by 'Country'.
2. Counted the number of records (transactions) for each country.
3. Sorted the countries based on their transaction counts in descending order.
4. Extracted the second record to determine the country with the second most transactions.

Results:

Country with the second most transactions is 'GREATER MANCHESTER' with 198338 transactions.

Question 3: Number of Transactions for each country

Procedure:

1. Grouped the dataset by 'Country'.
2. Counted the number of records (transactions) for each country.
3. Sorted the results alphabetically by country.
4. Saved the results into a CSV file named "output.csv in 2023202020_q3 folder".

Execution Time Measurement

To measure the efficiency of our PySpark operations, we monitored the execution time for each task using Python's built-in `time` module. This was done across different core configurations (2, 4, and 6 cores) to understand the scalability and performance of our operations.

Question 1

2 cores

```
cs3304.099@node06:~/assignment5/env
salloc: Granted job allocation 1059490
salloc: Waiting for resource configuration
salloc: Nodes node12 are ready for job
[cs3304.099@node12 env]$ python3 quesIV1.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 19:33:48 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
/home/itit/cs3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
  FutureWarning
Second highest transaction is 20000000 which is transacted in GREATER LONDON.
[cs3304.099@node12 env]$ exit
logout
salloc: Relinquishing job allocation 1059490
(env) [cs3304.099@abacus env]$ client_loop: send disconnect: Broken pipe
zenvls@zenvls: $ ssh cs3304.099@abacus.itit.ac.in
cs3304.099@abacus.itit.ac.in's password:
Last login: Thu Nov  9 19:31:11 2023 from 10.2.131.168
-----
Disk quotas for user cs3304.099 (uid 4591):
  Filesystem  space   quota   limit   grace   files   quota   limit   grace
  /home      1085M   2500M   2500M           2120      0      0
-----
[cs3304.099@abacus ~]$ cd assignment5/env/
[cs3304.099@abacus env]$ source ./bin/activate
(env) [cs3304.099@abacus env]$ slnt3 -c 2
salloc: Granted job allocation 1059519
salloc: Waiting for resource configuration
salloc: Nodes node06 are ready for job
[cs3304.099@node06 env]$ python3 quesIV1.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 21:21:20 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
/home/itit/cs3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
  FutureWarning
Second highest transaction is 20000000 which is transacted in GREATER LONDON.
Total time taken by the process: 39.70907211303711 seconds
[cs3304.099@node06 env]$
```

4 cores

```
cs3304.099@node06:~/assignment5/env
cs3304.099@abacus.itit.ac.in's password:
Last login: Thu Nov  9 19:31:11 2023 from 10.2.131.168
-----
Disk quotas for user cs3304.099 (uid 4591):
  Filesystem  space   quota   limit   grace   files   quota   limit   grace
  /home      1085M   2500M   2500M           2120      0      0
-----
[cs3304.099@abacus ~]$ cd assignment5/env/
[cs3304.099@abacus env]$ source ./bin/activate
(env) [cs3304.099@abacus env]$ slnt3 -c 2
salloc: Granted job allocation 1059519
salloc: Waiting for resource configuration
salloc: Nodes node06 are ready for job
[cs3304.099@node06 env]$ python3 quesIV1.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 21:21:20 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
/home/itit/cs3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
  FutureWarning
Second highest transaction is 20000000 which is transacted in GREATER LONDON.
Total time taken by the process: 39.70907211303711 seconds
[cs3304.099@node06 env]$ exit
logout
salloc: Relinquishing job allocation 1059519
(env) [cs3304.099@abacus env]$ slnt3 -c 4
salloc: Granted job allocation 1059520
salloc: Waiting for resource configuration
salloc: Nodes node06 are ready for job
[cs3304.099@node06 env]$ python3 quesIV1.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 21:23:41 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
/home/itit/cs3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
  FutureWarning
Second highest transaction is 20000000 which is transacted in GREATER LONDON.
Total time taken by the process: 24.888995885849 seconds
[cs3304.099@node06 env]$
```

6 cores

```
cs3304.099@node11:~/assignment5/env

To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 21:21:20 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
/home/iit/c3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
FutureWarning
Second highest transaction is 20000000 which is transacted in GREATER LONDON.
Total time taken by the process: 39.70907211303711 seconds
[cs3304.099@node06 env]$ exit
logout
salloc: Relinquishing job allocation 1059519
(env) [cs3304.099@abacus env]$ sint3 -c 4
salloc: Granted job allocation 1059520
salloc: Waiting for resource configuration
salloc: Nodes node06 are ready for job
[cs3304.099@node06 env]$ python3 quesIV1.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 21:23:41 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
/home/iit/c3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
FutureWarning
Second highest transaction is 20000000 which is transacted in GREATER LONDON.
Total time taken by the process: 24.8889585849 seconds
[cs3304.099@node06 env]$ exit
logout
salloc: Relinquishing job allocation 1059520
(env) [cs3304.099@abacus env]$ sint3 -c 6
salloc: Granted job allocation 1059522
salloc: Waiting for resource configuration
salloc: Nodes node11 are ready for job
[cs3304.099@node11 env]$ python3 quesIV1.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 21:24:37 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
/home/iit/c3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
FutureWarning
Second highest transaction is 20000000 which is transacted in GREATER LONDON.
Total time taken by the process: 22.963464498519897 seconds
[cs3304.099@node11 env]$
```

Question 2

2 cores

```
cs3304.099@node06:~/assignment5/env

(env) [cs3304.099@abacus env]$ sint3 -c 2
salloc: Granted job allocation 1059527
salloc: Waiting for resource configuration
salloc: Nodes node06 are ready for job
[cs3304.099@node06 env]$ python3 2023202020_q2.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 22:08:43 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
/home/iit/c3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
FutureWarning
23/11/09 22:09:06 WARN WindowExec: No Partition Defined for Window operation! Moving all data to a single partition, this can cause serious performance degradation.
23/11/09 22:09:06 WARN WindowExec: No Partition Defined for Window operation! Moving all data to a single partition, this can cause serious performance degradation.
23/11/09 22:09:16 WARN WindowExec: No Partition Defined for Window operation! Moving all data to a single partition, this can cause serious performance degradation.
23/11/09 22:09:17 WARN WindowExec: No Partition Defined for Window operation! Moving all data to a single partition, this can cause serious performance degradation.
Country with the second most transactions is 'GREATER MANCHESTER' with 198338 transactions.
Total time taken by the process: 30.68170690536499 seconds
[cs3304.099@node06 env]$
```

4 cores

```
cs3304.099@node12:~/assignment5/env
(env) [cs3304.099@abacus env]$ sint3 -c 4
salloc: Granted job allocation 1059530
salloc: Waiting for resource configuration
salloc: Nodes node12 are ready for job
[cs3304.099@node12 env]$ python3 2023202020_q2.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 22:10:01 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform
... using builtin-java classes where applicable
/home/iit/cs3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
FutureWarning
23/11/09 22:10:18 WARN WindowExec: No Partition Defined for Window operation! Moving all data t
o a single partition, this can cause serious performance degradation.
23/11/09 22:10:18 WARN WindowExec: No Partition Defined for Window operation! Moving all data t
o a single partition, this can cause serious performance degradation.
23/11/09 22:10:25 WARN WindowExec: No Partition Defined for Window operation! Moving all data t
o a single partition, this can cause serious performance degradation.
23/11/09 22:10:25 WARN WindowExec: No Partition Defined for Window operation! Moving all data t
o a single partition, this can cause serious performance degradation.
Country with the second most transactions is 'GREATER MANCHESTER' with 198338 transactions.
Total time taken by the process: 20.69897961616516 seconds
[cs3304.099@node12 env]$
```

6 cores

```
cs3304.099@node11:~/assignment5/env
(env) [cs3304.099@abacus env]$ sint3 -c 6
salloc: Granted job allocation 1059531
salloc: Waiting for resource configuration
salloc: Nodes node11 are ready for job
[cs3304.099@node11 env]$ python3 2023202020_q2.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 22:11:06 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform
... using builtin-java classes where applicable
/home/iit/cs3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
FutureWarning
23/11/09 22:11:22 WARN WindowExec: No Partition Defined for Window operation! Moving all data t
o a single partition, this can cause serious performance degradation.
23/11/09 22:11:22 WARN WindowExec: No Partition Defined for Window operation! Moving all data t
o a single partition, this can cause serious performance degradation.
23/11/09 22:11:28 WARN WindowExec: No Partition Defined for Window operation! Moving all data t
o a single partition, this can cause serious performance degradation.
23/11/09 22:11:28 WARN WindowExec: No Partition Defined for Window operation! Moving all data t
o a single partition, this can cause serious performance degradation.
Country with the second most transactions is 'GREATER MANCHESTER' with 198338 transactions.
Total time taken by the process: 19.05935788154602 seconds
[cs3304.099@node11 env]$
```

Question 3

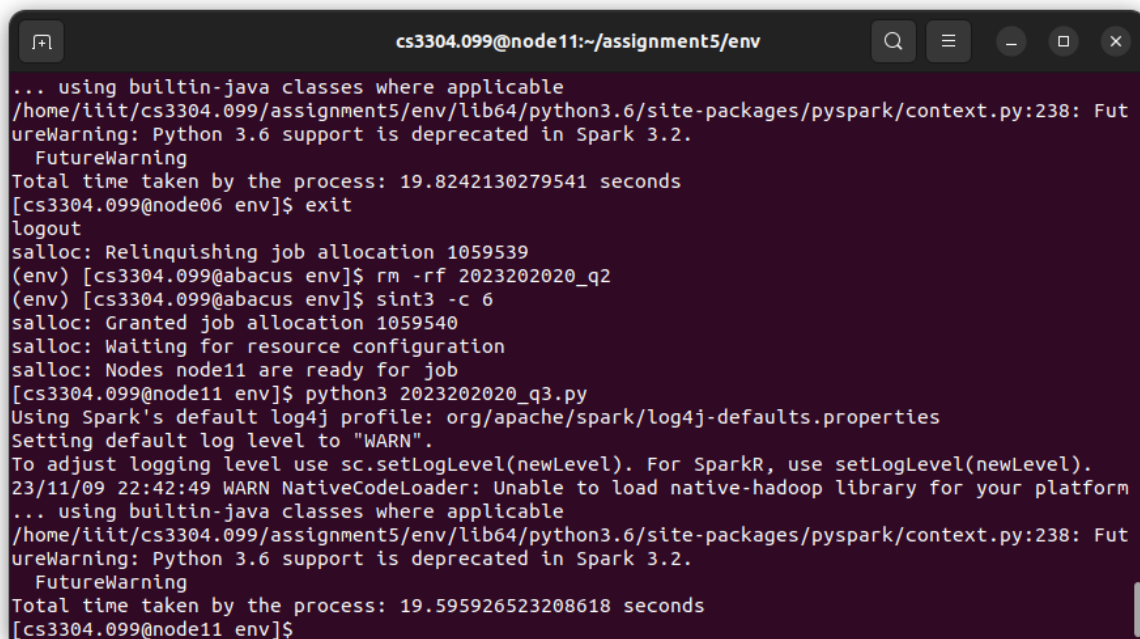
2 cores

```
cs3304.099@node06:~/assignment5/env
2023202020_q2.py bin lib pyvenv.cfg
2023202020_q3 House_Pricing.csv lib64 ques1V1.py
[cs3304.099@node10 env]$ rsrcun: error: node10: task 0: Killed
(env) [cs3304.099@abacus env]$ rm -rf directoryname
(env) [cs3304.099@abacus env]$ rm -rf 2023202020_q3
(env) [cs3304.099@abacus env]$ ls
2023202020_q1.py 2023202020_q3.py House_Pricing.csv lib pip-selfcheck.json ques1V1.py
2023202020_q2.py bin include lib64 pyvenv.cfg share
(env) [cs3304.099@abacus env]$ rm ques1V1.py
(env) [cs3304.099@abacus env]$ sint3 -c 2
salloc: Granted job allocation 1059538
salloc: Waiting for resource configuration
salloc: Nodes node06 are ready for job
[cs3304.099@node06 env]$ python3 2023202020_q3.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 22:38:50 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform
... using builtin-java classes where applicable
/home/iit/cs3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
FutureWarning
Total time taken by the process: 33.124144554138184 seconds
[cs3304.099@node06 env]$
```

4 cores

```
cs3304.099@node06:~/assignment5/env
salloc: Nodes node06 are ready for job
[cs3304.099@node06 env]$ rm 2023202020_q3
rm: cannot remove '2023202020_q3': No such file or directory
[cs3304.099@node06 env]$ ls
2023202020_q1.py 2023202020_q2.py bin include lib64 pyvenv.cfg
2023202020_q2 2023202020_q3.py House_Pricing.csv lib pip-selfcheck.json share
[cs3304.099@node06 env]$ rm 2023202020_q2
rm: cannot remove '2023202020_q2': Is a directory
[cs3304.099@node06 env]$ rm -rf 2023202020_q2
[cs3304.099@node06 env]$ sint3 -c 4
bash: sint3: command not found
[cs3304.099@node06 env]$ sint3 -c 4
bash: sint3: command not found
[cs3304.099@node06 env]$ python3 2023202020_q3.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 22:41:50 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform
... using builtin-java classes where applicable
/home/iit/cs3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
FutureWarning
Total time taken by the process: 19.8242130279541 seconds
[cs3304.099@node06 env]$
```

6 cores

A terminal window titled 'cs3304.099@node11:~/assignment5/env' showing the execution of a Spark job. The output includes warnings about Python 3.6 deprecation, job allocation details from 'salloc', and the execution of 'python3 2023202020_q3.py'. It shows the job running on node11 and completing with a total time of 19.595926523208618 seconds.

```
... using builtin-java classes where applicable
/home/iit/cs3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
  FutureWarning
Total time taken by the process: 19.8242130279541 seconds
[cs3304.099@node06 env]$ exit
logout
salloc: Relinquishing job allocation 1059539
(env) [cs3304.099@abacus env]$ rm -rf 2023202020_q2
(env) [cs3304.099@abacus env]$ sint3 -c 6
salloc: Granted job allocation 1059540
salloc: Waiting for resource configuration
salloc: Nodes node11 are ready for job
[cs3304.099@node11 env]$ python3 2023202020_q3.py
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
23/11/09 22:42:49 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform
... using builtin-java classes where applicable
/home/iit/cs3304.099/assignment5/env/lib64/python3.6/site-packages/pyspark/context.py:238: FutureWarning: Python 3.6 support is deprecated in Spark 3.2.
  FutureWarning
Total time taken by the process: 19.595926523208618 seconds
[cs3304.099@node11 env]$
```

Time Comparisons:

- 1) **2 cores:**
 - a) Question 1: 39.709 seconds
 - b) Question 2: 30.681 seconds
 - c) Question 3: 33.124 seconds
- 2) **4 cores:**
 - a) Question 1: 24.888 seconds
 - b) Question 2: 20.698 seconds
 - c) Question 3: 19.834 seconds
- 3) **6 cores:**
 - a) Question 1: 22.963 seconds
 - b) Question 2: 19.059 seconds
 - c) Question 3: 19.595 seconds

Observations:

1) Performance Gains with More Cores:

Across all questions, there is a noticeable decrease in the time taken as the number of cores increases from 2 to 4. This improvement indicates that the tasks are parallelizable and benefit from the additional computational resources.

2) Diminishing Returns Beyond 4 Cores:

The performance gains from increasing the core count from 4 to 6 are less substantial than the gains from 2 to 4 cores. While there is still an improvement, it suggests that there are

diminishing returns as more cores are added, which is a common characteristic of parallel processing systems due to overheads such as communication between threads or processes.

3) Anomaly in Question 3 Trend:

Unlike the other questions, Question 3 shows a decrease in execution time when moving from 2 to 4 cores and remains nearly constant when increasing from 4 to 6 cores. This pattern indicates that the task for Question 3 is able to efficiently utilize 4 cores, but does not benefit as much from further increases, possibly due to I/O constraints or the nature of the task not requiring additional computational resources.