Project 2 Hot Spot Analysis

Hot Zone Analysis:

Reflection:

I followed the instruction for installing all the required software's for the project and then I checked the requirements for the project what needs to be done and then modified the required functions and followed the instructions for compiling the code.

Lessons Learned:

I learned how to filter the points in rectangles using Scala and learnt how to use group by in Scala.

I got the output in multiple files and then by using the coalesce method I got it in one file.

Implementation:

For this project we need to install jdk, sbt, Apache spark.

It is used to calculate the hotness of the rectangles (more points means hotter) by finding the points in rectangles.

In this we need to modify the ST_Contains function and this function only returns when the points are in the rectangles. Then by using the GROUP BY and sort we can count and sort the data

After modifying the function, first create a jar file using sbt assembly and then test the code using spark-submit.

Output:

1	-73.789411,40.666459,-73.756364,40.680494	1
2	-73.793638,40.710719,-73.752336,40.730202	1
3	-73.795658,40.743334,-73.753772,40.779114	1
4	-73.796512,40.722355,-73.756699,40.745784	1
5	-73.797297,40.738291,-73.775740,40.770411	1
6	-73.802033,40.652546,-73.738566,40.668036	8
7	-73.805770,40.666526,-73.772204,40.690003	3
8	-73.815233,40.715862,-73.790295,40.738951	2
9	-73.816380,40.690882,-73.768447,40.715693	1
10	-73.819131,40.582343,-73.761289,40.609861	1
11	-73.825921,40.702281,-73.790734,40.719217	2
12	-73.826577,40.757744,-73.790317,40.779587	1
13	-73.832707,40.620010,-73.746541,40.665414	200
14	-73.839460,40.746988,-73.815375,40.781725	3
15	-73.840130,40.662481,-73.801134,40.691956	4
16	-73.840817,40.775618,-73.790584,40.801020	1
17	-73.842332,40.804005,-73.790064,40.845347	2
18	-73.843148,40.701398,-73.816380,40.715998	2
19	-73.849479,40.681155,-73.810634,40.704221	2
20	-73.861099,40.714345,-73.817260,40.764083	21
21	-73.862040,40.706406,-73.829798,40.739016	16
22	-73.864482,40.833000,-73.851686,40.842478	1
23	-73.867911,40.734291,-73.847510,40.748860	1
24	-73.869236,40.765468,-73.825139,40.798517	136
25	-73.873659,40.744942,-73.853128,40.758349	1
26	-73.875093,40.711476,-73.856258,40.735947	3
27	-73.876389,40.756642,-73.854497,40.771776	67
28	-73.878249,40.581702,-73.766710,40.639962	10
29	-73.884652,40.822974,-73.856153,40.833622	1
30	-73.891752,40.727801,-73.864472,40.749165	8
31	-73.891834,40.763673,-73.855049,40.786046	334
32	-73.897174,40.822518,-73.872773,40.840171	2
33	-73.899862,40.746820,-73.869426,40.771545	12
34	-73.901227,40.834354,-73.886599,40.843533	1

Hot Cell Analysis:

Reflection:

I followed the instruction for installing all the required software's for the project and then I checked the requirements for the project what needs to be done and then modified the required functions and followed the instructions for compiling the code.

Lessons Learned:

I learned how to get the pickups and the cell coordinates and how to calculate mean, standard deviation and z-score using spark SQL.

Implementation:

For Hot Cell Analysis we need to calculate the Z-score by selecting the cell coordinates and counting the pickups for getting the how many pickups are in the coordinates of the cell and then including the cell coordinates and sum of counts of the first query in the range of 27 neighbor cells.

We need to modify the user defined function and calculate the mean and standard deviation. Then using the withColumn method we can calculate z-score.

After modifying the function, first create a jar file using sbt assembly and then test the code using spark-submit.

Output:

1	-7399	4075	15
2	-7399	4075	22
3	-7399	4075	14
4	-7399	4075	29
5	-7398	4075	15
6	-7399	4075	16
7	-7399	4075	21
8	-7399	4075	28
9	-7399	4075	23
10	-7399	4075	30
11	-7398	4075	29
12	-7398	4075	28
13	-7398	4075	14
14	-7399	4074	15
15	-7398	4075	22
16	-7399	4075	27
17	-7399	4074	23
18	-7398	4075	30
19	-7398	4075	16
20	-7399	4074	22
21	-7399	4074	16
22	-7398	4076	15
23	-7398	4075	21
24	-7399	4075	13
25	-7399	4075	9
26	-7398	4075	23
27	-7399	4074	30
28	-7398	4076	14
29	-7398	4076	28