



Creating MapReduce program to calculating pi



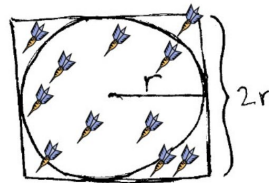
Table of Content

1. Theory
2. Setup
3. Create input files
4. Execution
5. Result
6. Conclusion
7. References

Theory

There are many ways to calculate Pi. But in this project, we are using MapReduce

- Throw N darts on the board. Each dart lands at a random position (x,y) on the board.



- Note if each dart landed inside the circle or not
 - Check if $x^2 + y^2 < r$
- Take the total number of darts that landed in the circle as S

$$4 \left(\frac{S}{N} \right) = \pi$$

Formula:

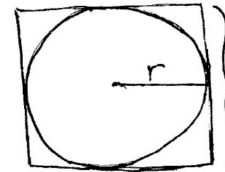
$$4 * S / N = 4 * (\pi * r * r) / (4 * r * r) = \pi$$

Note:

- S = darts inside the circle = the area of the circle
- N = darts on the board = the area of the square

Sample MapReduce Code- Estimate π

- Estimating π by random sampling
- Imagine you have a dart board like so:



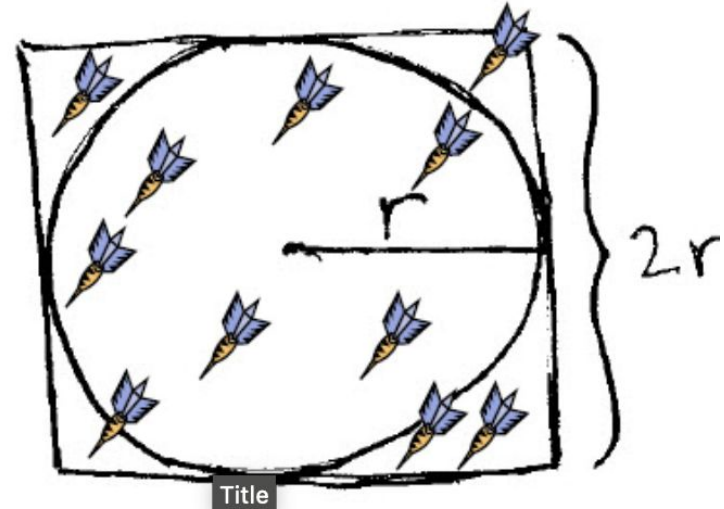
$$P(\text{dart in circle}) = \frac{\pi r^2}{4r^2} = \frac{\pi}{4}$$
$$\Rightarrow \pi = 4P(\text{dart in circle})$$

- π is simply the (ratio of darts that land inside the circle to the total number of darts thrown) times 4

How?

1. Let (x,y) be a random position of the dart inside the square. Then, we map each (x,y) pair to a result. If the pair is inside the circle, then result = 1, otherwise 0.
2. To calculate the π , we need to sum all the pair result inside the circle as S , and divide by the total number of pair N , multiply by 4, and get π .

$$\pi = 4(S/N)$$

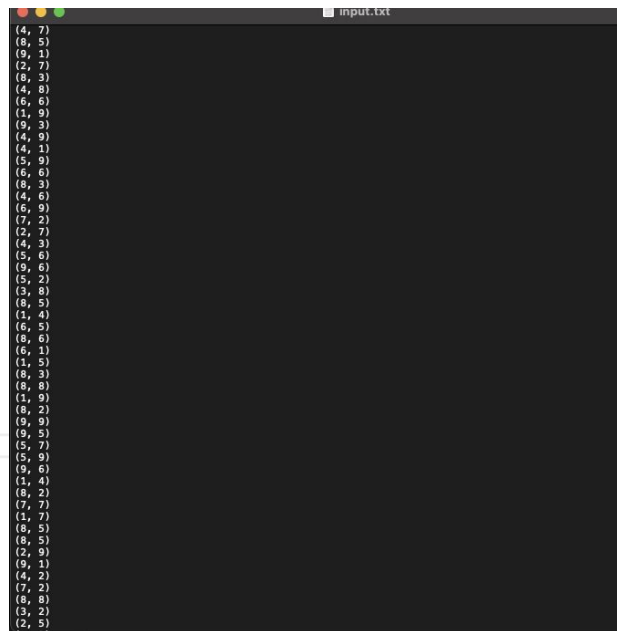


Create input files

Randomly generate 200 coordinates in (x,y) format and write them into the input.txt

CS5b/0 > input.py > ...

```
1 import random
2
3 # Generate 200 random coordinates
4 coordinates = [(random.randint(1, 9), random.randint(1, 9)) for _ in range(200)]
5
6 # Write the coordinates to the input file
7 with open("input.txt", "w") as file:
8     for x, y in coordinates:
9         print(f"({x}, {y})")
10        file.write(f"({x}, {y})\n")
```



```
(4, 7)
(8, 5)
(9, 1)
(2, 7)
(8, 3)
(4, 8)
(6, 6)
(1, 9)
(9, 3)
(4, 9)
(4, 1)
(5, 9)
(6, 6)
(8, 3)
(4, 6)
(6, 9)
(7, 2)
(2, 7)
(4, 3)
(5, 5)
(9, 6)
(5, 2)
(3, 8)
(6, 5)
(1, 4)
(6, 5)
(8, 6)
(6, 1)
(1, 5)
(8, 3)
(8, 8)
(1, 9)
(8, 2)
(9, 9)
(9, 5)
(5, 7)
(5, 8)
(9, 6)
(1, 4)
(6, 2)
(7, 7)
(1, 7)
(8, 5)
(6, 5)
(2, 9)
(9, 1)
(4, 2)
(7, 2)
(8, 8)
(3, 2)
(2, 5)
```

Create a Bucket and dataproc Clusters on GCP

w6h1

Location: us-west1 (Oregon) Storage class: Standard Public access: Not public Protection: None

OBJECTS CONFIGURATION PERMISSIONS PROTECTION LIFECYCLE OBSERVABILITY NEW INVENTORY REPORTS NEW

Buckets > w6h1 > input

UPLOAD FILES UPLOAD FOLDER CREATE FOLDER TRANSFER DATA MANAGE HOLDS DOWNLOAD DELETE

Filter by name prefix only		Filter		Filter objects and folders		Show deleted data			
<input type="checkbox"/>	Name	Size	Type	Created	Storage class	Last modified	Public access	Version history	Encryption
<input type="checkbox"/>	input.txt	1.4 KB	text/plain	Jun 26, 2023, 9:54:33 PM	Standard	Jun 26, 2023, 9:54:33 PM	Not public	—	Google-managed

Clusters

+ CREATE CLUSTER REFRESH START STOP DELETE REGIONS + 5 RECOMMENDED ALERTS

Filter Search clusters, press Enter

<input type="checkbox"/>	Name	Status	Region	Zone	Total worker nodes	Scheduled deletion	Cloud Storage staging bucket
<input type="checkbox"/>	w6h1	Running	us-west1	us-west1-c	0	Off	dataproc-staging-us-west1-66473074362-bo3oyqoj

No clusters selected

PERMISSIONS

Please see



PySpark Pi calculation program

```
pi.py x input.py
pi.py
1 from pyspark.sql import SparkSession
2 import sys
3
4 # Create a SparkSession
5 spark = SparkSession.builder.appName("PiEstimation").getOrCreate()
6
7 if len(sys.argv) != 3:
8     raise Exception("Exactly 2 arguments are required: <inputUri> <outputUri>")
9 inputUri = sys.argv[1]
10 outputUri = sys.argv[2]
11
12 # Read the input file containing 20 coordinates (x, y)
13 coordinates = spark.read.text(inputUri)
14
15 # Define the function to calculate if a point is inside the circle
16 # Radius = 5
17 def points(row):
18     x, y = map(float, row.value[1:-1].split(','))
19     if x**2 + y**2 <= 5**2:
20         return "inside"
21     else:
22         return "outside"
23
24 # Calculate the number of points inside and outside the unit circle
25 point_counts = coordinates.rdd.map(points).countByValue()
26
27 # Get the count of points inside the circle
28 inside_circle_count = point_counts.get("inside", 0)
29
30 # Get the count of points outside the circle
31 outside_circle_count = point_counts.get("outside", 0)
32
33 # Calculate the total number of points
34 total_count = coordinates.count()
35
36 # Estimate the value of pi
37 pi_estimate = 4.0 * inside_circle_count / total_count
38
39 # Print
40 print("Points inside the circle:", inside_circle_count)
41 print("Points outside the circle:", outside_circle_count)
42 print("Pi is approximately:", pi_estimate)
43
44 # Stop the SparkSession
45 spark.stop()
--
```

Execution

gcloud dataproc jobs submit pyspark pi.py
--cluster=w6h1 --region=us-west1 --
gs://w6h1/input/input.txt gs://w6h1/output

```
tfang@78cloudshell:~$ (cd $PWD) gcloud dataproc jobs submit pyspark pi.py --cluster=w6h1 --region=us-west1 -- gs://w6h1/input/input.txt gs://w6h1/output
Job [c408d371e38740a7a7e9a98fcb985812] submitted.
Waiting for job output...
23/06/27 04:54:54 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
23/06/27 04:54:54 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
23/06/27 04:54:55 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartbeat
23/06/27 04:54:55 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
23/06/27 04:54:55 INFO org.sparkproject.jetty.util.log: Logging initialized @2825ms to org.sparkproject.jetty.util.log.Slf4jLog
23/06/27 04:54:55 INFO org.sparkproject.jetty.server.Server: jetty-9.4.40.v20210413; built: 2021-04-13T20:42:42.666Z; gtc: d881a572662e1943a14ae12e7e1207989f218b74; jvm 1.8.0_372-b07
23/06/27 04:54:55 INFO org.sparkproject.jetty.server.Server: Started @2919ms
23/06/27 04:54:55 INFO org.apache.hadoop.yarn.client.RMPProxy: Connecting to ResourceManager at w6h1-m/10.138.0.9:8032
23/06/27 04:54:56 INFO org.apache.hadoop.yarn.client.AHSProxy: Connecting to Application History server at w6h1-m/10.138.0.9:10200
23/06/27 04:54:57 INFO org.apache.hadoop.conf.Configuration: resource-types.xml not found
23/06/27 04:54:57 INFO org.apache.hadoop.yarn.util.resource.ResourceUtils: Unable to find 'resource-types.xml'.
23/06/27 04:54:57 INFO org.apache.hadoop.yarn.client.api.impl.YarnClientImpl: Submitted application application_1687838987543_0007
23/06/27 04:54:58 INFO org.apache.hadoop.yarn.client.RMPProxy: Connecting to ResourceManager at w6h1-m/10.138.0.9:8030
23/06/27 04:55:01 INFO com.google.cloud.hadoop.repackaged.gcs.com.google.cloud.hadoop.gcsio.GoogleCloudStorageImpl: ignoring exception of type GoogleJsonResponseException: verified object already exists
with desired state.
Points inside the circle: 37
Points outside the circle: 163
Pi is approximately: 0.74
23/06/27 04:55:12 INFO org.sparkproject.jetty.server.AbstractConnector: Stopped Spark$2f588b80(HTTP/1.1, (http://1.1)){}{0.0.0.0:0}
Job [c408d371e38740a7a7e9a98fcb985812] finished successfully.
done: true
driverControlFilesUri: gs://dataproc-staging-us-west1-66473074362-bo3oqoq/google-cloud-dataproc-metainfo/3bb07439-774d-4fcl-a754-c1d2bf261087/jobs/c408d371e38740a7a7e9a98fcb985812/
driverOutputResourceUri: gs://dataproc-staging-us-west1-66473074362-bo3oqoq/google-cloud-dataproc-metainfo/3bb07439-774d-4fcl-a754-c1d2bf261087/jobs/c408d371e38740a7a7e9a98fcb985812/driveroutput
placement:
  clusterName: w6h1
  clusterUuid: 3bb07439-774d-4fcl-a754-c1d2bf261087
pysparkJob:
  args:
  - gs://w6h1/input/input.txt
  - gs://w6h1/output
  mainPythonFileUri: gs://dataproc-staging-us-west1-66473074362-bo3oqoq/google-cloud-dataproc-metainfo/3bb07439-774d-4fcl-a754-c1d2bf261087/jobs/c408d371e38740a7a7e9a98fcb985812/staging/pi.py
reference:
  jobId: c408d371e38740a7a7e9a98fcb985812
  projectId: ca3703f
status:
  state: DONE
  stateStartTime: '2023-06-27T04:55:16.662221Z'
statusHistory:
- state: PENDING
  stateStartTime: '2023-06-27T04:54:51.461208Z'
- state: SETUP_DONE
  stateStartTime: '2023-06-27T04:54:51.490776Z'
  details: Agent reported job success
  state: RUNNING
  stateStartTime: '2023-06-27T04:54:51.673991Z'
yarnApplications:
- name: PiEstimation
  progress: 1.0
  state: FINISHED
  trackingUri: http://w6h1-m:8088/proxy/application_1687838987543_0007/
tfang@78cloudshell:~$ (cd $PWD) g
```




Result

I use 200 pairs with a radius of 5 for this project, and the results is

Inside: 37

Outside: 163

Pi: 0.74

```
Points inside the circle: 37  
Points outside the circle: 163  
Pi is approximately: 0.74
```



Conclusion

The result 0.74 is far off π , but I only use 200 pairs of numbers. If we increase the number of pairs to 2000 or more, the result will be much closer to the π .



References

Exercises for Pi: https://hc.labnet.sfbu.edu/~henry/npu/classes/mapreduce/pi/slide/exercise_pi.html

MapRedcue Pi concept:

https://hc.labnet.sfbu.edu/~henry/npu/classes/mapreduce/pi/slide/mapreduce_pi.html