

Hadoop, Spark Installation and algorithm execution

Operating System and Cloud
Computing DS203

Our Team Members :

Rishi Koushik : 21bds067

Nischay Kondai : 21bds045

Vamsi Madhav : 21bds051

Yuvraaj Bhama : 21bds071



What is Hadoop?



Hadoop is a software framework that is freely available for storing and processing big data sets across multiple computers. The aim is to allow the system to expand from one server to thousands of machines, each with local storage and computing power. Initially created by Doug Cutting and Mike Cafarella in 2005, the software was designed to manage the enormous quantities of data created by web crawlers.

What is Spark?

Apache Spark is an open-source distributed computing system that processes large data sets across clusters of computers. Developed at the University of California, Berkeley, in 2009 and made available to the public in 2010, Spark uses an in-memory computing engine to provide faster data processing than traditional disk-based systems. It offers different libraries and APIs for data processing, including SQL, machine learning, and graph processing. Spark's core engine uses RDDs to allow parallel processing across nodes in a cluster.



Steps for Installation:

Hadoop

1. Download Hadoop
2. Install Java
3. Configure SSH
4. Set up environment variables
5. Configure Hadoop
6. Format the Hadoop file system
7. Start Hadoop

Spark

1. Ensure that your system meets the minimum requirements to run Apache Spark.
2. Download the latest version of Apache Spark from the official website.
3. Install Java if it's not already installed.
4. Set up environment variables for Java and Spark.
5. Configure Apache Spark by unzipping the downloaded file and adjusting it to your system specifications.
6. Start Apache Spark using the command line or launch it via a notebook interface.

Spark Algorithm : Generalized Linear Regression

```
51 * An example demonstrating generalized linear regression.  
52 * Run with  
53 * <pre>  
54 * bin/run-example ml.JavaGeneralizedLinearRegressionExample  
55 * </pre>  
56 */  
57  
58 public class JavaGeneralizedLinearRegressionExample {  
59  
60     public static void main(String[] args) {  
61         SparkSession spark = SparkSession  
62             .builder()  
63             .appName("JavaGeneralizedLinearRegressionExample")  
64             .getOrCreate();  
65  
66         // $example on$  
67         // Load training data  
68         Dataset<Row> dataset = spark.read().format("libsvm")  
69             .load("hdfs://namenode:9000/example_algo/sample_linear_regression_data.txt");  
70  
71         GeneralizedLinearRegression glr = new GeneralizedLinearRegression()  
72             .setFamily("gaussian")  
73             .setLink("identity")  
74             .setMaxIter(10)  
75             .setRegParam(0.3);  
76  
77         // Fit the model  
78         GeneralizedLinearRegressionModel model = glr.fit(dataset);  
79  
80         // Print the coefficients and intercept for generalized linear regression model  
81         System.out.println("Coefficients: " + model.coefficients());  
82         System.out.println("Intercept: " + model.intercept());  
83  
84         // Summarize the model over the training set and print out some metrics  
85         GeneralizedLinearRegressionTrainingSummary summary = model.summary();  
86         System.out.println("Coefficient Standard Errors: "  
87             + Arrays.toString(summary.coefficientStandardErrors()));  
88         System.out.println("T Values: " + Arrays.toString(summary.tValues()));  
89         System.out.println("P Values: " + Arrays.toString(summary.pValues()));  
90         System.out.println("Dispersion: " + summary.dispersion());  
91         System.out.println("Null Deviance: " + summary.nullDeviance());  
92         System.out.println("Residual Degree Of Freedom Null: " + summary.residualDegreeOfFreedomNull());  
93         System.out.println("Deviance: " + summary.deviance());  
94         System.out.println("Residual Degree Of Freedom: " + summary.residualDegreeOfFreedom());  
95         System.out.println("AIC: " + summary.aic());  
96         System.out.println("Deviance Residuals: ");  
97         summary.residuals().show();  
98         // $example off$  
99  
100        spark.stop();
```

Algorithm Execution :

The screenshot shows the Apache Spark 3.4.0 UI interface, specifically the "Stages" tab under the "Jobs" section. The title bar indicates the application is "JavaGeneralizedLinearRe..." and the version is "3.4.0".

Stages for All Jobs

Completed Stages: 6
Skipped Stages: 1

Completed Stages (6)

| Stage Id | Description | Submitted | Duration | Tasks: Succeeded/Total | Input | Output | Shuffle Read | Shuffle Write |
|----------|---|------------------------------|----------|------------------------|-----------|--------|--------------|---------------|
| 6 | sum at GeneralizedLinearRegression.scala:712 | +details 2023/05/10 12:22:19 | 0.6 s | 1/1 | 116.3 KiB | | | |
| 5 | sum at GeneralizedLinearRegression.scala:1384 | +details 2023/05/10 12:22:18 | 0.6 s | 1/1 | 116.3 KiB | | | |
| 4 | head at GeneralizedLinearRegression.scala:1251 | +details 2023/05/10 12:22:16 | 0.9 s | 1/1 | | 80.0 B | | |
| 2 | head at GeneralizedLinearRegression.scala:1251 | +details 2023/05/10 12:22:14 | 2 s | 1/1 | 116.3 KiB | | 80.0 B | |
| 1 | treeAggregate at WeightedLeastSquares.scala:107 | +details 2023/05/10 12:21:59 | 12 s | 1/1 | 116.3 KiB | | | |
| 0 | reduce at MLUtils.scala:94 | +details 2023/05/10 12:21:50 | 5 s | 1/1 | 116.3 KiB | | | |

Page: 1 1 Pages. Jump to . Show items in a page.

Skipped Stages (1)

| Stage Id | Description | Submitted | Duration | Tasks: Succeeded/Total | Input | Output | Shuffle Read | Shuffle Write |
|----------|--|------------------|----------|------------------------|-------|--------|--------------|---------------|
| 3 | head at GeneralizedLinearRegression.scala:1251 | +details Unknown | Unknown | 0/1 | | | | |

Page: 1 1 Pages. Jump to . Show items in a page.

Benefits of HDFS cluster for Generalized Linear Regression

- Capability to handle large data volumes, which is critical for GLM algorithms that require substantial data to train and develop models accurately. HDFS clusters store and process large datasets across many nodes in a distributed manner, making them ideal for large-scale machine learning.
- Fault tolerance and high availability features : HDFS stores data redundantly across numerous nodes, ensuring data remains accessible even if some nodes fail.
- Scalable storage solution for GLM, which enables organizations to add more nodes as their data requirements expand. A scalable storage solution allows organizations to store and analyze vast amounts of data efficiently, which is crucial for machine learning that demands training models using large datasets.

Output Achieved

```
+-----+
| devianceResiduals|
+-----+
| -10.974359174246889|
|  6.8872328138420559|
|  -4.596541837478908|
| -20.411667435819638|
| -10.270419345342642|
| -6.0156058956799905|
| -10.663939415849267|
|  2.1153968525024713|
|  3.9807132379137675|
| -17.225218272069533|
| -4.611647633532147|
|  6.4176669407698546|
|  11.407137945380537|
| -20.70176540467664|
| -2.683748540510967|
| -16.755494794232536|
|  8.154668342638725|
| -1.4355057987358848|
| -0.6435058688185704|
|  -1.13802589316832|
+-----+
only showing top 20 rows
```

Generalized Linear Regression on Sample_Linear_Regression_data:

```
1 -9.490009878824548 1:0.4551273600657362 2:0.36644694351969087 3:-0.38256108933468047 4:-0.4458430198517267 5:0.33109790358914726 6:0.8067445293443565 7:-0.2624341731773887 8:-0.44850386111659524  
9:-0.07269284838169332 10:0.5658035575800715  
2 0.2577820163584905 1:0.8386555657374337 2:-0.1270180511534269 3:0.499812362510895 4:-0.22686625128130267 5:-0.6452430441812433 6:0.18869982177936828 7:-0.5804648622673358 8:0.651931743775642  
9:-0.6555641246242951 10:0.17485476357259122  
3 -4.438869807456516 1:0.5025608135349202 2:0.14208069682973434 3:0.16004976900412138 4:0.505019897181302 5:-0.9371635223468384 6:-0.2841601610457427 7:0.6355938616712786 8:-0.1646249064941625  
9:0.9480713629917628 10:0.42681251564645817  
4 -19.782762789614537 1:-0.0388509668871313 2:-0.4166870051763918 3:0.8997202693189332 4:0.6409836467726933 5:0.273289095712564 6:-0.26175701211620517 7:-0.2794902492677298 8:-0.1306778297187794  
9:-0.0853658111046115 10:-0.05462315824828923  
5 -7.966593841555266 1:-0.06195495876886281 2:0.6546448480299902 3:-0.6979368909424835 4:0.6677324708883314 5:-0.07938725467767771 6:-0.43885601665437957 7:-0.608071585153688 8:-0.6414531182501653|  
9:0.7313735926547045 10:-0.026818676347611925  
6 -7.896274316726144 1:-0.15805658673794265 2:0.26573958270655806 3:0.3997172901343442 4:-0.3693430998846541 5:0.14324061105995334 6:-0.25797542063247825 7:0.7436291919296774 8:0.6114618853239959  
9:0.2324273700703574 10:-0.25128128782199144  
7 -8.464803554195287 1:0.39449745853945895 2:0.817229160415142 3:-0.6077058562362969 4:0.6182496334554788 5:0.2558665508269453 6:-0.07320145794330979 7:-0.38884168866510227 8:0.07981886851873865  
9:0.27022202891277614 10:-0.7474843534024693  
8 2.1214592666251364 1:-0.005346215048158909 2:-0.9453716674280683 3:-0.9270309666195007 4:-0.032312290091389695 5:0.31010676221964206 6:-0.20846743965751569 7:0.8803449313707621 8:-0.23077831216541722  
9:0.29246395759528565 10:0.5409312755478819  
9 1.0720117616524107 1:0.7880855916368177 2:0.19767407429003536 3:0.9520689432368168 4:-0.845829774129496 5:0.5502413918543512 6:-0.44235539500246457 7:0.7984106594591154 8:-0.2523277127589152  
9:-0.1373808897290778 10:-0.3353514432305029  
10 -13.772441561702871 1:-0.3697050572653644 2:-0.11452811582755928 3:-0.807098168238352 4:0.4903066124307711 5:-0.6582805242342049 6:0.6107814398427647 7:-0.7204208094262783 8:-0.8141063661170889  
9:-0.9459402662357332 10:0.09666938346350307  
11 -5.082010756207233 1:-0.4356034277380735 2:0.9349906440170221 3:0.8090021580031235 4:-0.3121157071110545 5:-0.9718883630945336 6:0.6191882496201251 7:0.0429886073795116 8:0.67031110015402  
9:0.16692329718223786 10:0.37649213869502973  
12 7.887786536531237 1:0.11276440263810383 2:-0.7684997525607482 3:0.1770172737885798 4:0.7902845707138706 5:0.2529503304079441 6:-0.23483801763662826 7:0.8072501895004851 8:0.667399201927047  
9:-0.4796127376677324 10:0.9244724404994455  
13 14.323146365332388 1:-0.2049276879687938 2:0.1470694373531216 3:-0.48366999792166787 4:0.643491115907358 5:0.3183669486383729 6:0.22821350958477082 7:-0.023605251086149304 8:-0.2770587742156372  
9:0.47596326458377436 10:0.7107229819632654  
14 -20.057482615789212 1:-0.3205057828114841 2:0.51605972926996 3:0.45215640988181516 4:0.01712446974606241 5:0.5508198371849293 6:-0.2478254241316491 7:0.7256483175955235 8:0.39418662792516 9:-0.6797384914236382  
10:0.6001217520150142  
15 -0.8995693247765151 1:0.4508991072414843 2:0.589749448443134 3:0.6464818311502738 4:0.7005669004769028 5:0.9699584106930381 6:-0.7417466269908464 7:0.22818964839784495 8:0.08574936236270037  
9:-0.6945765138377225 10:0.06915201979238828  
16 -19.16829262296376 1:0.09798746565879424 2:-0.34288007110901964 3:0.440249350802451 4:-0.22440768392359534 5:-0.9695067570891225 6:-0.7942032659310758 7:-0.792286205517398 8:-0.6535487038528798  
9:0.79526764706168951 10:-0.1622831617066689  
17 5.601801561245534 1:0.6949189734965766 2:-0.32697929564739403 3:-0.15359663581829275 4:-0.89518650905020432 5:0.2057889391931318 6:-0.6676656789571533 7:-0.03553655732400762 8:0.14550349954571096  
9:0.034600542078191854 10:0.4223352065067103  
18 -3.2256352187273354 1:0.35278245969741096 2:0.7022211035026023 3:0.5686638754605697 4:-0.4202155290448111 5:-0.26102723928249216 6:0.010688215941416779 7:-0.4311544807877927 8:0.9500151672991208  
9:0.14380635780710693 10:-0.7549354840975826  
19 1.5299675726687754 1:-0.13079299081883855 2:0.0983382230287082 3:0.15347083875928424 4:0.45507300685816965 5:0.1921083467305864 6:0.6361110540492223 7:0.7675261182370992 8:-0.2543488202081907  
9:0.2927051050236915 10:0.680182444769418  
20 -0.250102447941961 1:-0.8062832278617296 2:0.8266289890474885 3:0.22684501241708888 4:0.1726291966578266 5:-0.6778773666126594 6:0.9993906921393696 7:0.1789490173139363 8:0.5584053824232391  
9:0.03495894704368174 10:-0.8505720014852347  
21 12.792267926563595 1:-0.00846120064588818 2:-0.648273596036564 3:-0.005334477339629995 4:0.3781469006858833 5:0.30565234666790686 6:-0.2822867492866177 7:0.10175120738413801 8:0.5342432888482425  
9:0.05146513075475534 10:-0.6459729964194652  
22 6.082192787194888 1:0.42519013450094767 2:0.09441503345243984 3:-0.07898439043103522 4:-0.32207498048636474 5:-0.9180071861219266 6:0.5951317320731633 7:0.41000814588717693 8:-0.3926260640533046  
9:0.2789036768568971 10:0.13163692286014528  
23 -7.481405271455238 1:0.03324842612749346 2:0.07055844751995122 3:-0.47199515597021113 4:-0.682690342465275 5:0.3983414713797069 6:-0.2136729393256811 7:-0.09066563475481249 8:-0.4640338194317184  
9:-0.03513782089224482 10:-0.1711809802758364  
24 6.739533816100517 1:0.1774546460228057 2:-0.6783644553523549 3:-0.47871398278230504 4:0.02272121490463097 5:-0.5047649289302389 6:0.26479596144873896 7:-0.32045436544054096 8:0.313047940487379  
9:0.6269418147567556 10:0.9710114516962312  
25 3.780807062175497 1:0.01715676997104909 2:0.8975962429865936 3:-0.46594560920034134 4:0.2873623499953055 5:0.8894362304584083 6:0.17973981232418468 7:0.49105791400707743 8:-0.7359842740294882  
9:0.38941133808001127 10:-0.715188477228046
```

Thank You

SLIDE PRESENTATIONS DESIGN