Index

A

Anonymous function, 66
Apache Hadoop, 1
Apache HBase, 42–44
Apache Hive, 6–7, 230
Apache Kafka, 8, 178
Apache License, 7
Apache Mahout, 5
Apache Mesos, 38–42
Apache Pig, 7
Apache Spark, 9
Apache Storm, 2
Apache Tez, 2
Atomicity, Consistency, Isolation, and Durability (ACID), 12
avg() function, 209

B

bfs() function, 225
Big data
- characteristics, 2
- variety, 3
- velocity, 3
- veracity, 3
- volume, 2

Breadth-first search algorithm, 220, 225

C

CentOS operating system, 15
Cluster managers, 10–11
count() function, 140, 198, 247
Count of records, 195
createCSV() function, 152
createDataFrame() function, 191
createJSON() function, 157, 158
createOrReplaceTempView() function, 207
createStream() function, 181
CSV file
- reading, 150
  - paired RDD, 152
  - parseCSV() function, 151
- writing RDD to, 152

D

Data aggregation, 200
- filament data, 119
- mean, 121–123, 125–126
- paired RDD, 121
- RDD, 120

DataFrame, 188
- changing data type of column, 192
- compound logical expression, 194
- creation, 191, 196
- data aggregation, 200
- data joining, 210
  - full outer join, 220
  - inner join, 215
  - left outer join, 217
  - reading student data table, PostgreSQL database, 212
  - reading subject data, JSON file, 215
  - right outer join, 219
- exploratory data analysis, 195
- filament data nested list creation, 188
- filter() and count() functions, 193, 198
- RDD of row objects, creation, 190
- schema creation, 189
- schema definition, 196
- schema printing, 192

© Raju Kumar Mishra 2018
R. K. Mishra, *PySpark Recipes*, https://doi.org/10.1007/978-1-4842-3141-8

DataFrame (*cont.*)
- SQL and HiveQL queries, execution of, 207
- summary statistics, 197

DataFrame abstraction, 187
Data joining, 210
- full outer join, 220
- inner join, 215
- left outer join, 217
- reading student data table, PostgreSQL database, 212
- reading subject data, JSON file, 215
- right outer join, 219

DataNodes, 4
Dataset interface, 187
Data structure, labeled point, 242
Dense vector creation, 236
describe() function, 197
Distributed systems, 1

E

E-commerce companies, 1
Extract, transform, and load (ETL), 7

F

filter() function, 193, 198
Full outer join, 220

G

Google file system (GFS), 4
GraphFrames library, 10, 187
GraphFrames object creation, 224
groupBy() function, 200

H

Hadoop distributed file system (HDFS), 4–5, 15
- reading data from, 145
- saving RDD data to, 146

Hadoop installation
- .bashrc file, 21
- CentOS User, 16–17
- downloading, 19
- environment file, 20
- installation directory, 19–20
- Java, 17
- jps command, 23
- NameNode format, 22
- passwordless login, 18–19
- problem, 16
- properties files, 20–21
- solution, 16
- starting script, 22

HBase, 2, 12–14
Hive installation, 27–29
Hive property, 37
HiveQL and SQL queries, execution of, 207
HiveQL commands, 7
Hive query language (HQL), 6

I

Inner join, 215
I/O operations. *See* PySpark, input/output (I/O) operations
IPython
- integration, 79
- Notebook, 81–83
- pip, 80
- PySpark, 81

J

Java database connectivity (JDBC), 12
JavaScript object notation (JSON)
- reading file, 154
- reading subject data from, 215
- writing RDD to file, 156

jsonParse() function, 155–156

K

K-nearest neighbors (KNN) algorithm, PySpark, 166

L

Labeled point, 242, 245, 254
Lasso regression, 257
Left outer join, 217
Len() function, 140
Linear regression, 235, 243
Local matrix creation, 239

M

Machine learning, 235
map() function, 154, 190, 245

Map-reduce model, 5
Matrices
  local matrix creation, 239
  row matrix creation, 241
MLlib, 10
Mutable collection, 56

N, O

NameNode, 4
nc command, 175
Netcat, 174
newAPIHadoopRDD() function, 159–160
NoSQL databases, 2, 15
NumPy
  array(), 73
  dtype, 74–75
  mean, 78
  mean temperature, 77
  medians, 78
  min() and max(), 76
  ndarray, 72
  pip, 72
  shape, 75
  standard deviation, 77
  temperature readings, 71
  variance, 77–78
  vstack(), 73–74

P, Q

Page-rank algorithm, 226
  damping factor, 133
  function, 134
  loop, 135
  nested lists, 134
  optimization, 164
  paired RDDs, 135
  web-page system, 132
Paired RDD
  aggregate data (*see* Data aggregation)
  creation
    consonants, 117
    elements, 116–117
    keys(), 118
    map(), 116, 118
    values, 118
  join data
    creation, 128–129
    full outer, 131
    inner, 129
    left outer, 130
    nested list, 128
    right outer, 131
  key/value-pair architecture, 115
  page rank (*see* Page-rank algorithm)
playDataLineLength RDD, 142
PostgreSQL database, 12, 30–35, 37, 212
predict() function, 256
printSchema() function, 192
Procedural language/PostgreSQL (PL/pgSQL), 12
PySpark, 15, 37
  k-nearest neighbors (KNN) algorithm, 166
  page-rank algorithm optimization, 164
  script execution
    in local mode, 182
    Standalone and Mesos cluster managers, 184
PySpark, input/output (I/O) operations
  reading CSV file, 150
    paired RDD, 152
    parseCSV() function, 151
  reading data
    HDFS, 145
    sequential file, 147
  reading directory, 143
    textFile() function, 144
    wholeTextFiles() function, 144
  reading JSON file, 154
  reading table data, HBase, 159
  reading text file
    count() function, 140
    Len() function, 140
    textFile() function, 138
    wholeTextFiles() function, 139
  saving RDD data to HDFS, 146
  writing data to sequential file, 148
  writing RDD
    CSV file, 152
    JSON file, 156
    text file, 141
PySpark MLlib, 235
  dense vector creation, 236
  labeled point creation, 242
  local matrix creation, 239
  row matrix creation, 241
  sparse vector creation, 237

PySparkSQL, 7, 9
- breadth-first search algorithm, 220, 225
- DataFrame, 188
  - changing data type of column, 192
  - compound logical expression, 194
  - creation, 191, 196
  - data aggregation, 200
  - data joining, 210
  - exploratory data analysis, 195
  - filament data nested list creation, 188
  - filter() and count() functions, 193, 198
  - schema creation, 189
  - schema definition, 196
  - schema printing, 192
  - SQL and HiveQL queries, execution of, 207
  - summary statistics, 197
  - RDD of row objects, creation, 190
- GraphFrames object creation, 224
- page-rank algorithm, 226
- reading table data, Apache Hive, 230

PySpark shell
- problem, 25
- Python programmers, 26
- solution, 25

PySpark streaming, 163
- integration, Apache Kafka, 178
- reading data, console, 174

Python
- conditionals, 67–68
- data and data type, 46–48
- dictionary, 62–64
- for and while loops, 69–70
- functions, 65
- lambda function, 66–67
- list, 54–58
- NumPy (*see* NumPy)
- set, 60–61
- string, 48–51
- tuple, 58–60
- typecasting, 51–53

R

randomSplit() function, 246
registerTempTable() function, 207
Regression
- lasso, 257
- linear, 243
- ridge, 251

Relational database management system (RDBMS), 2, 6, 15
Resilient distributed dataset (RDD)
- action, 87–88
- creation
  - first(), 90
  - getNumPartitions(), 91
  - list, 89
  - parallelized(), 89
  - take(), 90
- data manipulation
  - collect(), 98
  - filter(), 98
  - list, 95
  - map(), 95–96
  - sortBy(), 97
  - take(), 96
- Mesos cluster manager, 113–114
- run set operations, 99–103
- SparkContext, 86
- Standalone Cluster Manager, 109–113
- summary statistics, 103–108
- temperature data, 91–94
- transformation, 87–88

Ridge regression, 251
Right outer join, 219
round() function, 209
Row matrix creation, 241

S

save() method, 248
saveAsTextFile() function, 141
select command, 208
sequenceFile() function, 148
sequenceFile() method, 148
Sequential file
- reading data from, 147
- writing data to, 148

show() function, 191, 209, 215
Shuffling, 163
socketTextStream() function, 175–176
Software libraries, 235
Spark, 163
Spark architecture
- driver, 86
- executors, 86

Spark installation
  allPySpark location, 24
  .bashrc File, 24
  downloading, 23
  environment file, 24
  problem, 23
  PySpark, 25
  solution, 23
  .tgz file, 23
spark.read.csv() function, 244
spark.sql() function, 208
Sparse vector creation, 237
split() function, 176
SQL and HiveQL queries, execution of, 207
Stochastic gradient descent (SGD), 247
stringToNumberSum() function, 176
strip() function, 176
StructField(), 189
StructType() function, 223
Structured query language (SQL), 6
summary() function, 195
Supervised machine-learning algorithm, 243

T

Table joining, 210
take() function, 245
textFile() function, 138, 143–144
train() method, 247
type() function, 208

U

Unix, 4
User-defined functions (UDFs), 7

V

Vectors
  dense vector, 236
  sparse vector, 237

W, X, Y, Z

wholeTextFiles() function, 139, 143–144