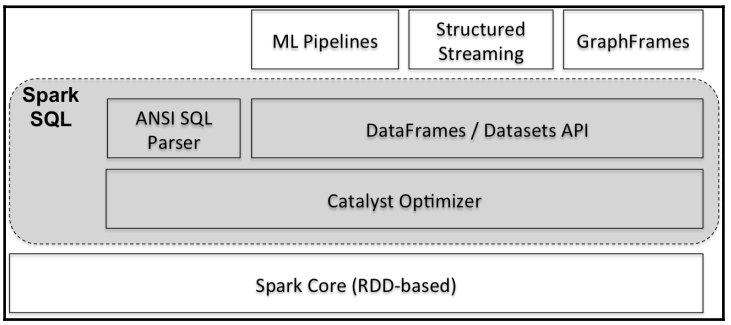
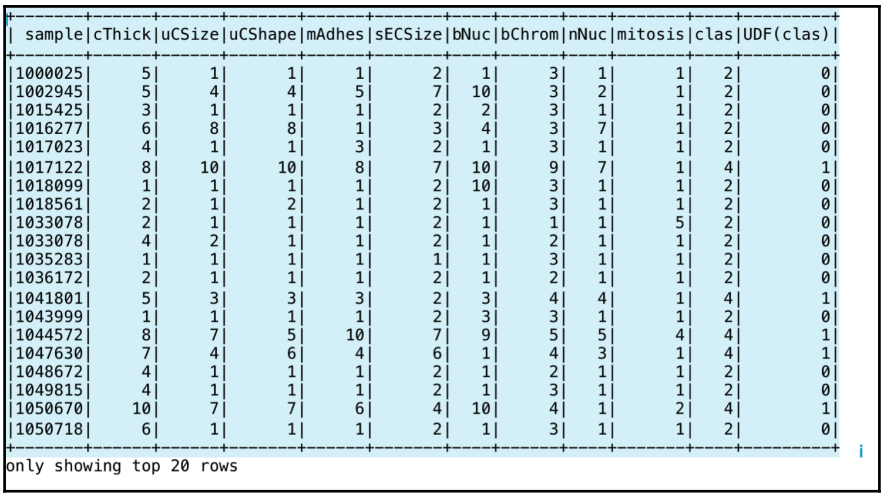
Chapter 1: Getting Started with Spark SQL



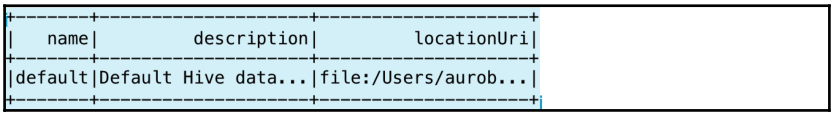

```
Aurobindos-MacBook-Pro-2:spark-2.2.0-bin-hadoop2.7 aurobindosarkar$ bin/spark-shell
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
17/08/19 19:45:38 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
17/08/19 19:45:44 WARN metastore.ObjectStore: Failed to get database global_temp, returning NoSuchObjectException
Spark context Web UI available at http://192.168.8.100:4040
Spark context available as 'sc' (master = local[*], app id = local-1503152139834).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.2.0
      /_/

Using Scala version 2.11.8 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_77)
Type in expressions to have them evaluated.
Type :help for more information.
```

```
scala> df.show()
+-------+------+------+-------+------+-------+----+------+----+-------+-----+
| sample|cThick|uCSize|uCShape|mAdhes|sECSize|bNuc|bChrom|nNuc|mitosis| clas|
+-------+------+------+-------+------+-------+----+------+----+-------+-----+
|1000025|     5|     1|      1|     1|      2|   1|     3|   1|      1|    2|
|1002945|     5|     4|      4|     5|      7|  10|     3|   2|      1|    2|
|1015425|     3|     1|      1|     1|      2|   2|     3|   1|      1|    2|
|1016277|     6|     8|      8|     1|      3|   4|     3|   7|      1|    2|
|1017023|     4|     1|      1|     3|      2|   1|     3|   1|      1|    2|
|1017122|     8|    10|     10|     8|      7|  10|     9|   7|      1|    4|
|1018099|     1|     1|      1|     1|      2|  10|     3|   1|      1|    2|
|1018561|     2|     1|      2|     1|      2|   1|     3|   1|      1|    2|
|1033078|     2|     1|      1|     1|      2|   1|     1|   1|      5|    2|
|1033078|     4|     2|      1|     1|      2|   1|     2|   1|      1|    2|
|1035283|     1|     1|      1|     1|      1|   1|     3|   1|      1|    2|
|1036172|     2|     1|      1|     1|      2|   1|     2|   1|      1|    2|
|1041801|     5|     3|      3|     3|      2|   3|     4|   4|      1|    4|
|1043999|     1|     1|      1|     1|      2|   3|     3|   1|      1|    2|
|1044572|     8|     7|      5|    10|      7|   9|     5|   5|      4|    4|
|1047630|     7|     4|      6|     4|      6|   1|     4|   3|      1|    4|
|1048672|     4|     1|      1|     1|      2|   1|     2|   1|      1|    2|
|1049815|     4|     1|      1|     1|      2|   1|     3|   1|      1|    2|
|1050670|    10|     7|      7|     6|      4|  10|     4|   1|      2|    4|
|1050718|     6|     1|      1|     1|      2|   1|     3|   1|      1|    2|
+-------+------+------+-------+------+-------+----+------+----+-------+-----+
only showing top 20 rows
```

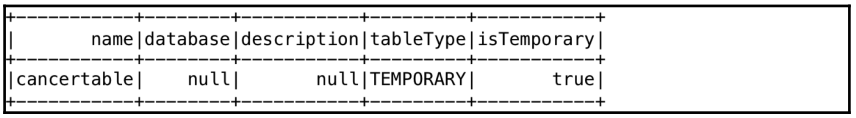
```
scala> sqlDF.show()
+-------+----+
| sample|bNuc|
+-------+----+
|1000025|   1|
|1002945|  10|
|1015425|   2|
|1016277|   4|
|1017023|   1|
|1017122|  10|
|1018099|  10|
|1018561|   1|
|1033078|   1|
|1033078|   1|
|1035283|   1|
|1036172|   1|
|1041801|   3|
|1043999|   3|
|1044572|   9|
|1047630|   1|
|1048672|   1|
|1049815|   1|
|1050670|  10|
|1050718|   1|
+-------+----+
   only showing top 20 rows
```



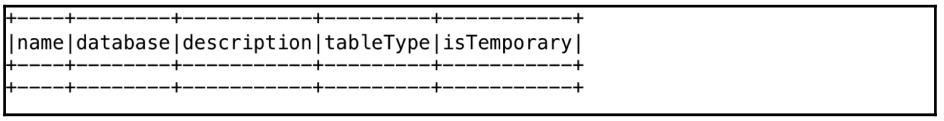
```
+-------+------+------+-------+------+-------+----+------+----+-------+----+---------+
| sample|cThick|uCSize|uCShape|mAdhes|sECSize|bNuc|bChrom|nNuc|mitosis|clas|UDF(clas)|
+-------+------+------+-------+------+-------+----+------+----+-------+----+---------+
|1000025|     5|     1|      1|     1|      2|   1|     3|   1|      1|   2|        0|
|1002945|     5|     4|      4|     5|      7|  10|     3|   2|      1|   2|        0|
|1015425|     3|     1|      1|     1|      2|   2|     3|   1|      1|   2|        0|
|1016277|     6|     8|      8|     1|      3|   4|     3|   7|      1|   2|        0|
|1017023|     4|     1|      1|     3|      2|   1|     3|   1|      1|   2|        0|
|1017122|     8|    10|     10|     8|      7|  10|     9|   7|      1|   4|        1|
|1018099|     1|     1|      1|     1|      2|  10|     3|   1|      1|   2|        0|
|1018561|     2|     1|      2|     1|      2|   1|     3|   1|      1|   2|        0|
|1033078|     2|     1|      1|     1|      2|   1|     1|   1|      5|   2|        0|
|1033078|     4|     2|      1|     1|      2|   1|     2|   1|      1|   2|        0|
|1035283|     1|     1|      1|     1|      1|   1|     3|   1|      1|   2|        0|
|1036172|     2|     1|      1|     1|      2|   1|     2|   1|      1|   2|        0|
|1041801|     5|     3|      3|     3|      2|   3|     4|   4|      1|   4|        1|
|1043999|     1|     1|      1|     1|      2|   3|     3|   1|      1|   2|        0|
|1044572|     8|     7|      5|    10|      7|   9|     5|   5|      4|   4|        1|
|1047630|     7|     4|      6|     4|      6|   1|     4|   3|      1|   4|        1|
|1048672|     4|     1|      1|     1|      2|   1|     2|   1|      1|   2|        0|
|1049815|     4|     1|      1|     1|      2|   1|     3|   1|      1|   2|        0|
|1050670|    10|     7|      7|     6|      4|  10|     4|   1|      2|   4|        1|
|1050718|     6|     1|      1|     1|      2|   1|     3|   1|      1|   2|        0|
+-------+------+------+-------+------+-------+----+------+----+-------+----+---------+
only showing top 20 rows
```



```
+-------+--------------------+--------------------+
|   name|         description|         locationUri|
+-------+--------------------+--------------------+
|default|Default Hive data...|file:/Users/aurob...|
+-------+--------------------+--------------------+
```



```
+-----------+--------+-----------+---------+-----------+
|       name|database|description|tableType|isTemporary|
+-----------+--------+-----------+---------+-----------+
|cancertable|    null|       null|TEMPORARY|       true|
+-----------+--------+-----------+---------+-----------+
```

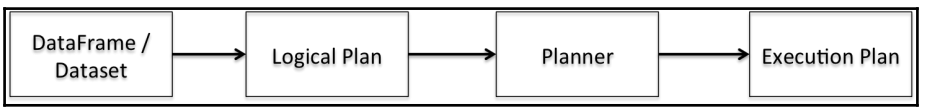


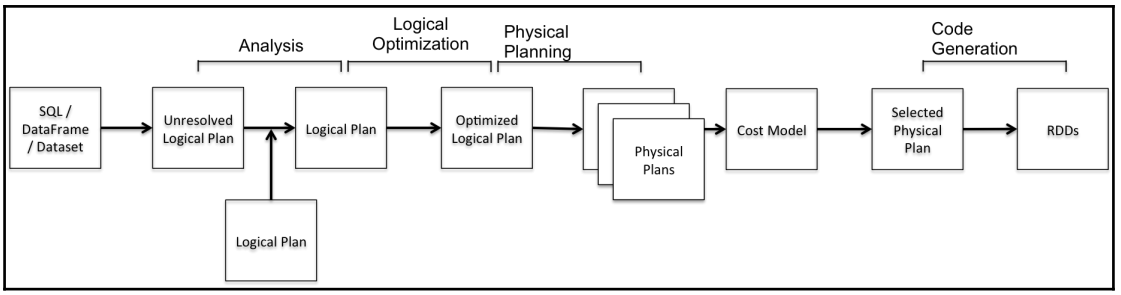
```
+----+--------+-----------+---------+-----------+
|name|database|description|tableType|isTemporary|
+----+--------+-----------+---------+-----------+
+----+--------+-----------+---------+-----------+
```

```
scala> rest2DSM1.show()
+--------------------+--------------------+-------------+------------+-------------+------------+
|                name|              street|         city|       phone|      cuisine|    stdphone|
+--------------------+--------------------+-------------+------------+-------------+------------+
|Adriano's Ristorante|2930 Beverly Glen...|  Los Angeles|310/475-9807|      Italian|310-475-9807|
|Arnie Morton's of...|435 S. La Cienega...|  Los Angeles|310/246-1501|     American|310-246-1501|
|  Art's Delicatessen| 12224 Ventura Blvd.|  Studio City|818/762-1221|     American|818-762-1221|
|   Barney Greengrass| 9570 Wilshire Blvd.|Beverly Hills|310/777-5877|     American|310-777-5877|
|         Beaurivage|26025 Pacific Coa...|       Malibu|310/456-5733|       French|310-456-5733|
|       Bistro Garden|    176 N. Canon Dr.|  Los Angeles|310/550-3900|  Californian|310-550-3900|
|        Border Grill|             4th St.|  Los Angeles|310/451-1655|      Mexican|310-451-1655|
|       Broadway Deli|   3rd St. Promenade| Santa Monica|310/451-0616|     American|310-451-0616|
|           Ca'Brea| 346 S. La Brea Ave.|  Los Angeles|213/938-2863|      Italian|213-938-2863|
|          Ca'del Sol| 4100 Cahuenga Blvd.|  Los Angeles|818/985-4669|      Italian|818-985-4669|
|          Cafe Bizou| 14016 Ventura Blvd.| Sherman Oaks|818/788-3536|       French|818-788-3536|
|          Cafe Pinot|    700 W. Fifth St.|  Los Angeles|213/239-6500|  Californian|213-239-6500|
|California Pizza ...|  207 S. Beverly Dr.|  Los Angeles|310/275-1101|  Californian|310-275-1101|
|           Campanile| 624 S. La Brea Ave.|  Los Angeles|213/938-1447|     American|213-938-1447|
|            Canter's| 419 N. Fairfax Ave.|  Los Angeles|213/651-2030|     American|213-651-2030|
|                Cava|             3rd St.|  Los Angeles|213/658-8898|Mediterranean|213-658-8898|
|         Cha Cha Cha|  656 N. Virgil Ave.|  Los Angeles|213/664-7723|    Caribbean|213-664-7723|
|          Chan Dara|310 N. Larchmont ...|  Los Angeles|213/467-1052|         Asian|213-467-1052|
|     Chinois on Main|       2709 Main St.| Santa Monica|310/392-9025|       French|310-392-9025|
|              Citrus|   6703 Melrose Ave.|  Los Angeles|213/857-0034|  Californian|213-857-0034|
+--------------------+--------------------+-------------+------------+-------------+------------+
only showing top 20 rows
```

```
+--------+
|count(1)|
+--------+
|     112|
+--------+
```

```
scala> sqlDF.show()
+--------------------+--------------------+------------+------------+
|                name|                name|       phone|    stdphone|
+--------------------+--------------------+------------+------------+
|      Buckhead Diner|      Buckhead Diner|404-262-3336|404-262-3336|
|Lespinasse (New Y...|          Lespinasse|212-339-6719|212-339-6719|
| Tavern on the Green| Tavern on the Green|212-873-3200|212-873-3200|
|   Brasserie Le Coze|   Brasserie Le Coze|404-266-1440|404-266-1440|
|         Bacchanalia|         Bacchanalia|404-365-0410|404-365-0410|
|        Pinot Bistro|        Pinot Bistro|818-990-0500|818-990-0500|
|Hedgerose Heights...|Hedgerose Heights...|404-233-7673|404-233-7673|
|               Jo Jo|               Jo Jo|212-223-5656|212-223-5656|
|Ritz-Carlton Cafe...|Restaurant  Ritz-...|404-659-0400|404-659-0400|
|Ritz-Carlton Rest...|Restaurant  Ritz-...|404-659-0400|404-659-0400|
|          Montrachet|          Montrachet|212-219-2777|212-219-2777|
|             Abruzzi|             Abruzzi|404-261-8186|404-261-8186|
|          River Cafe|   River Caf&eacute;|718-522-5200|718-522-5200|
|   Cafe des Artistes|Caf&eacute; des A...|212-877-3500|212-877-3500|
|   Bone's Restaurant|              Bone's|404-237-2663|404-237-2663|
|           Matsuhisa|           Matsuhisa|310-659-9639|310-659-9639|
|     PlumpJack Cafe|PlumpJack Caf&eac...|415-563-4755|415-563-4755|
|             Aquavit|             Aquavit|212-307-7311|212-307-7311|
|     Heera of India|      Heera of India|404-876-4408|404-876-4408|
|              Lutece|       Lut&egrave;ce|212-752-2225|212-752-2225|
+--------------------+--------------------+------------+------------+
only showing top 20 rows
```



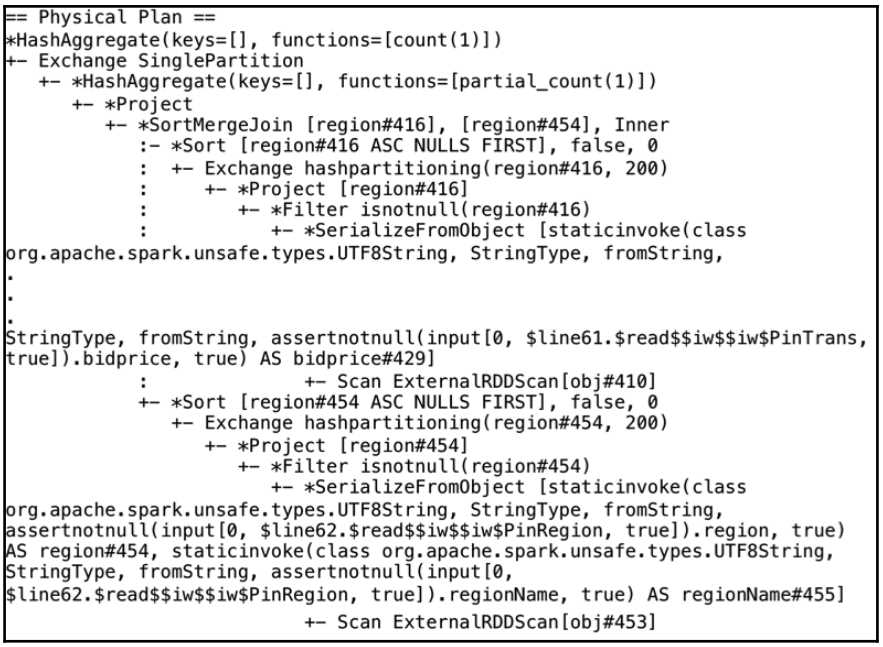


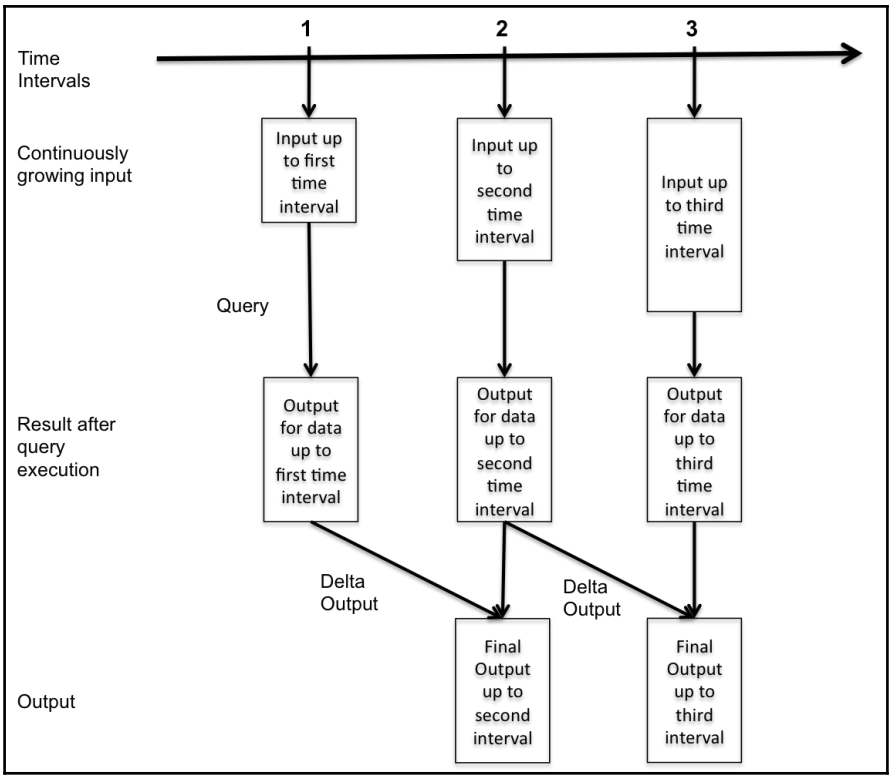

```
== Parsed Logical Plan ==
'Project [unresolvedalias('count(1), Some(<function1>))]
+- Project [region#416, bidid#411, timestamp#412, ipinyouid#413,
useragent#414, IP#415, city#417, adexchange#418, domain#419, url#420,
urlid#421, slotid#422, slotwidth#423, slotheight#424, slotvisibility#425,
slotformat#426, slotprice#427, creative#428, bidprice#429, regionName#455]
   +- Join Inner, (region#416 = region#454)
      :- SerializeFromObject [staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType,
.
.
.
$line61.$read$$iw$$iw$PinTrans, true))).bidprice, true) AS bidprice#429]
      :  +- ExternalRDD [obj#410]
      +- SerializeFromObject [staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType, fromString,
assertnotnull(assertnotnull(input[0, $line62.$read$$iw$$iw$PinRegion,
true))).region, true) AS region#454, staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType, fromString,
assertnotnull(assertnotnull(input[0, $line62.$read$$iw$$iw$PinRegion,
true))).regionName, true) AS regionName#455]
         +- ExternalRDD [obj#453]
```

```
== Analyzed Logical Plan ==
count(1): bigint
Aggregate [count(1) AS count(1)#583L]
+- Project [region#416, bidid#411, timestamp#412, ipinyouid#413,
useragent#414, IP#415, city#417, adexchange#418, domain#419, url#420,
urlid#421, slotid#422, slotwidth#423, slotheight#424, slotvisibility#425,
slotformat#426, slotprice#427, creative#428, bidprice#429, regionName#455]
   +- Join Inner, (region#416 = region#454)
      :- SerializeFromObject [staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType,
.
.
.
$line61.$read$$iw$$iw$PinTrans, true))).bidprice, true) AS bidprice#429]
      :  +- ExternalRDD [obj#410]
      +- SerializeFromObject [staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType, fromString,
assertnotnull(assertnotnull(input[0, $line62.$read$$iw$$iw$PinRegion,
true))).region, true) AS region#454, staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType, fromString,
assertnotnull(assertnotnull(input[0, $line62.$read$$iw$$iw$PinRegion,
true))).regionName, true) AS regionName#455]
         +- ExternalRDD [obj#453]
```

```
== Optimized Logical Plan ==
Aggregate [count(1) AS count(1)#583L]
+- Project
   +- Join Inner, (region#416 = region#454)
      :- Project [region#416]
      :  +- Filter isnotnull(region#416)
      :     +- SerializeFromObject [staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType,
.
.
.
$line61.$read$$iw$$iw$PinTrans, true]).bidprice, true) AS bidprice#429]
      :        +- ExternalRDD [obj#410]
      +- Project [region#454]
         +- Filter isnotnull(region#454)
            +- SerializeFromObject [staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType, fromString,
assertnotnull(input[0, $line62.$read$$iw$$iw$PinRegion, true]).region, true)
AS region#454, staticinvoke(class org.apache.spark.unsafe.types.UTF8String,
StringType, fromString, assertnotnull(input[0,
$line62.$read$$iw$$iw$PinRegion, true]).regionName, true) AS regionName#455]
               +- ExternalRDD [obj#453]
```

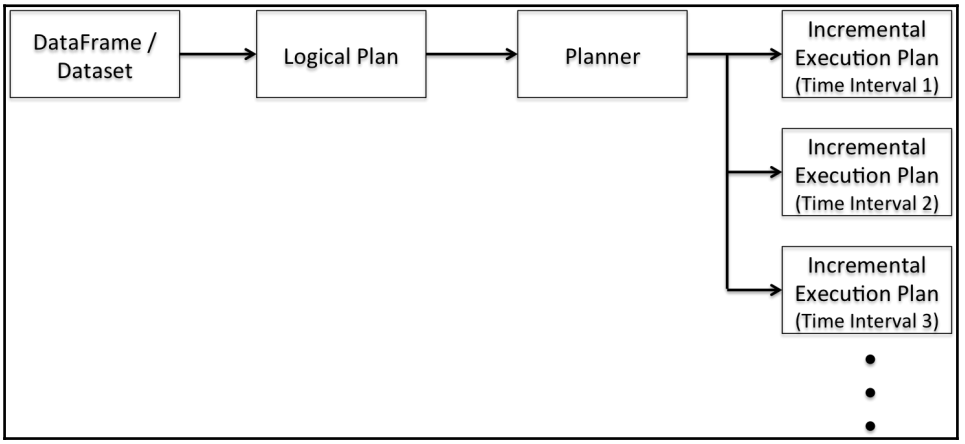
```
== Physical Plan ==
*HashAggregate(keys=[], functions=[count(1)], output=[count(1)#583L])
+- Exchange SinglePartition
   +- *HashAggregate(keys=[], functions=[partial_count(1)],
output=[count#585L])
      +- *Project
         +- *SortMergeJoin [region#416], [region#454], Inner
            :- *Sort [region#416 ASC NULLS FIRST], false, 0
            :  +- Exchange hashpartitioning(region#416, 200)
            :     +- *Project [region#416]
            :        +- *Filter isnotnull(region#416)
            :           +- *SerializeFromObject [staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType,
.
.
.
$line61.$read$$iw$$iw$PinTrans, true]).bidprice, true) AS bidprice#429]
            :              +- Scan ExternalRDDScan[obj#410]
            +- *Sort [region#454 ASC NULLS FIRST], false, 0
               +- Exchange hashpartitioning(region#454, 200)
                  +- *Project [region#454]
                     +- *Filter isnotnull(region#454)
                        +- *SerializeFromObject [staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType, fromString,
assertnotnull(input[0, $line62.$read$$iw$$iw$PinRegion, true]).region, true)
AS region#454, staticinvoke(class org.apache.spark.unsafe.types.UTF8String,
StringType, fromString, assertnotnull(input[0,
$line62.$read$$iw$$iw$PinRegion, true]).regionName, true) AS regionName#455]
                           +- Scan ExternalRDDScan[obj#453]
```


== Physical Plan ==
*HashAggregate(keys=[], functions=[count(1)])
+- Exchange SinglePartition
   +- *HashAggregate(keys=[], functions=[partial_count(1)])
      +- *Project
         +- *SortMergeJoin [region#416], [region#454], Inner
            :- *Sort [region#416 ASC NULLS FIRST], false, 0
            : +- Exchange hashpartitioning(region#416, 200)
            :   +- *Project [region#416]
            :   +- *Filter isnotnull(region#416)
            :   +- *SerializeFromObject [staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType, fromString,
.
.
.
StringType, fromString, assertnotnull(input[0, $line61.$read$$iw$$iw$PinTrans,
true]).bidprice, true) AS bidprice#429]
            :   +- Scan ExternalRDDScan[obj#410]
            +- *Sort [region#454 ASC NULLS FIRST], false, 0
               +- Exchange hashpartitioning(region#454, 200)
                  +- *Project [region#454]
                     +- *Filter isnotnull(region#454)
                        +- *SerializeFromObject [staticinvoke(class
org.apache.spark.unsafe.types.UTF8String, StringType, fromString,
assertnotnull(input[0, $line62.$read$$iw$$iw$PinRegion, true]).region, true)
AS region#454, staticinvoke(class org.apache.spark.unsafe.types.UTF8String,
StringType, fromString, assertnotnull(input[0,
$line62.$read$$iw$$iw$PinRegion, true]).regionName, true) AS regionName#455]
                           +- Scan ExternalRDDScan[obj#453]



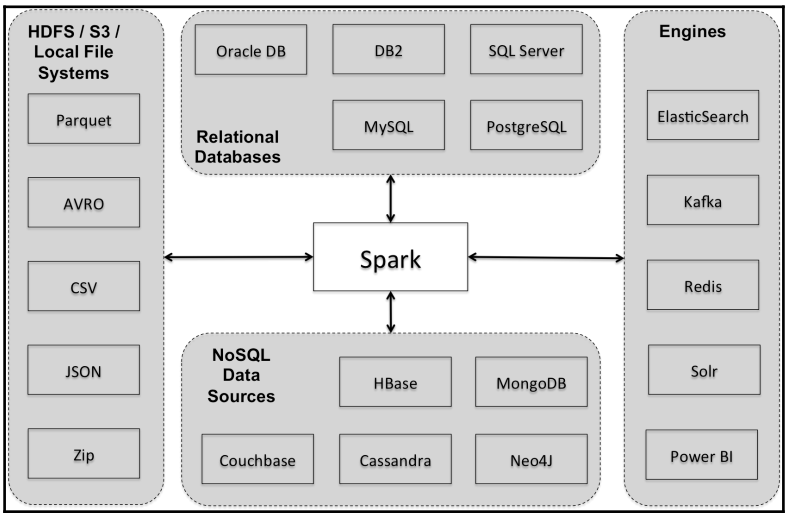

```
-------------------------------------------
Batch: 0
-------------------------------------------
+----+-----+
|city|count|
+----+-----+
| 148|21104|
| 243| 3207|
|  31| 1386|
| 137|  564|
|  85|30907|
| 251|  933|
|  65| 8247|
|  53| 1244|
| 255|  618|
| 133| 1069|
| 296|  613|
| 322|  320|
|  78|  290|
| 362|   29|
| 321|  543|
| 375| 6125|
| 108| 5937|
| 155| 2927|
| 193| 1347|
| 211| 1453|
+----+-----+
only showing top 20 rows
```

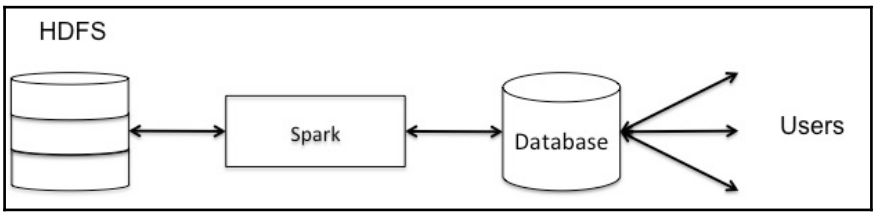
```
-------------------------------------------
Batch: 1
-------------------------------------------
+----+-----+
|city|count|
+----+-----+
| 148|41789|
| 243| 6733|
|  31| 2844|
|  85|63244|
| 251| 1968|
| 137| 1163|
|  65|17408|
| 255| 1351|
|  53| 2433|
| 296| 1209|
| 133| 2190|
| 322|  648|
|  78|  604|
| 321| 1113|
| 362|   58|
| 375|11748|
| 108|12641|
| 155| 5982|
| 193| 2913|
|  34| 2707|
+----+-----+
only showing top 20 rows
```



```
== Physical Plan ==
*HashAggregate(keys=[city#1033], functions=[count(1)])
+- StateStoreSave [city#1033],
OperatorStateId(/private/var/folders/tj/prwqrjj16jn4k5jh6g91rwtc0000gn/T/tempo
rary-570c0fdf-55ff-40cb-88eb-d00c01ed3f22/state,0,1), Complete, 0
   +- *HashAggregate(keys=[city#1033], functions=[merge_count(1)])
      +- StateStoreRestore [city#1033],
OperatorStateId(/private/var/folders/tj/prwqrjj16jn4k5jh6g91rwtc0000gn/T/tempo
rary-570c0fdf-55ff-40cb-88eb-d00c01ed3f22/state,0,1)
         +- *HashAggregate(keys=[city#1033], functions=[merge_count(1)])
            +- Exchange hashpartitioning(city#1033, 200)
               +- *HashAggregate(keys=[city#1033],
functions=[partial_count(1)])
                  +- *FileScan csv [city#1033] Batched: false, Format: CSV,
Location: InMemoryFileIndex[file:/Users/aurobindosarkar/Downloads/make-
ipinyou-data-master/original-data/ip..., PartitionFilters: [], PushedFilters:
[], ReadSchema: struct<city:int>
```

Chapter 2:
Using Spark SQL for Processing Structured and Semistructured Data





```
Aurobindos-MacBook-Pro-2:Downloadsaurobindosarkar$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.6.27 Homebrew

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

```
+---------------+-------------+------+-----+-------------------+-----------------------------+
| Field         | Type        | Null | Key | Default           | Extra                       |
+---------------+-------------+------+-----+-------------------+-----------------------------+
| transactionID | int(11)     | NO   | PRI | NULL              | auto_increment              |
| invoiceNo     | varchar(20) | YES  |     | NULL              |                             |
| stockCode     | varchar(20) | YES  |     | NULL              |                             |
| description   | varchar(255)| YES  |     | NULL              |                             |
| quantity      | int(11)     | YES  |     | NULL              |                             |
| unitPrice     | double      | YES  |     | NULL              |                             |
| customerID    | varchar(20) | YES  |     | NULL              |                             |
| country       | varchar(100)| YES  |     | NULL              |                             |
| invoiceDate   | timestamp   | NO   |     | CURRENT_TIMESTAMP | on update CURRENT_TIMESTAMP |
+---------------+-------------+------+-----+-------------------+-----------------------------+
9 rows in set (0.01 sec)
```

```
+---------+---------+--------------------+--------+---------------+---------+----------+--------------+-------------------+
|invoiceNo|stockCode|         description|quantity|    invoiceDate|unitPrice|customerID|       country|                 ts|
+---------+---------+--------------------+--------+---------------+---------+----------+--------------+-------------------+
|   536365|   85123A|WHITE HANGING HEA...|       6|01/12/10 8:26|     2.55|     17850|United Kingdom|2010-12-01 08:26:...|
|   536365|    71053| WHITE METAL LANTERN|       6|01/12/10 8:26|     3.39|     17850|United Kingdom|2010-12-01 08:26:...|
|   536365|   84406B|CREAM CUPID HEART...|       8|01/12/10 8:26|     2.75|     17850|United Kingdom|2010-12-01 08:26:...|
|   536365|   84029G|KNITTED UNION FLA...|       6|01/12/10 8:26|     3.39|     17850|United Kingdom|2010-12-01 08:26:...|
|   536365|   84029E|RED WOOLLY HOTTIE...|       6|01/12/10 8:26|     3.39|     17850|United Kingdom|2010-12-01 08:26:...|
|   536365|    22752|SET 7 BABUSHKA NE...|       2|01/12/10 8:26|     7.65|     17850|United Kingdom|2010-12-01 08:26:...|
|   536365|    21730|GLASS STAR FROSTE...|       6|01/12/10 8:26|     4.25|     17850|United Kingdom|2010-12-01 08:26:...|
|   536366|    22633|HAND WARMER UNION...|       6|01/12/10 8:28|     1.85|     17850|United Kingdom|2010-12-01 08:28:...|
|   536366|    22632|HAND WARMER RED P...|       6|01/12/10 8:28|     1.85|     17850|United Kingdom|2010-12-01 08:28:...|
|   536367|    84879|ASSORTED COLOUR B...|      32|01/12/10 8:34|     1.69|     13047|United Kingdom|2010-12-01 08:34:...|
|   536367|    22745|POPPY'S PLAYHOUSE...|       6|01/12/10 8:34|      2.1|     13047|United Kingdom|2010-12-01 08:34:...|
|   536367|    22748|POPPY'S PLAYHOUSE...|       6|01/12/10 8:34|      2.1|     13047|United Kingdom|2010-12-01 08:34:...|
|   536367|    22749|FELTCRAFT PRINCES...|       8|01/12/10 8:34|     3.75|     13047|United Kingdom|2010-12-01 08:34:...|
|   536367|    22310|IVORY KNITTED MUG...|       6|01/12/10 8:34|     1.65|     13047|United Kingdom|2010-12-01 08:34:...|
|   536367|    84969|BOX OF 6 ASSORTED...|       6|01/12/10 8:34|     4.25|     13047|United Kingdom|2010-12-01 08:34:...|
|   536367|    22623|BOX OF VINTAGE JI...|       3|01/12/10 8:34|     4.95|     13047|United Kingdom|2010-12-01 08:34:...|
|   536367|    22622|BOX OF VINTAGE AL...|       2|01/12/10 8:34|     9.95|     13047|United Kingdom|2010-12-01 08:34:...|
|   536367|    21754|HOME BUILDING BLO...|       3|01/12/10 8:34|     5.95|     13047|United Kingdom|2010-12-01 08:34:...|
|   536367|    21755|LOVE BUILDING BLO...|       3|01/12/10 8:34|     5.95|     13047|United Kingdom|2010-12-01 08:34:...|
|   536367|    21777|RECIPE BOX WITH M...|       4|01/12/10 8:34|     7.95|     13047|United Kingdom|2010-12-01 08:34:...|
+---------+---------+--------------------+--------+---------------+---------+----------+--------------+-------------------+
only showing top 20 rows
```

```
scala>writeData.show()
+---------+---------+--------------------+--------+---------+----------+--------------+--------------------+
|invoiceNo|stockCode|         description|quantity|unitPrice|customerID|       country|         invoiceDate|
+---------+---------+--------------------+--------+---------+----------+--------------+--------------------+
|  C579889|    23245|SET OF 3 REGENCY ...|      -8|     4.15|     13853|United Kingdom|2011-12-01 08:12:...|
|  C579890|    84947|ANTIQUE SILVER TE...|      -1|     1.25|     15197|United Kingdom|2011-12-01 08:14:...|
|  C579890|    23374|RED SPOT PAPER GI...|      -1|     0.82|     15197|United Kingdom|2011-12-01 08:14:...|
|  C579890|    84945|MULTI COLOUR SILV...|      -2|     0.85|     15197|United Kingdom|2011-12-01 08:14:...|
|  C579891|    23485|BOTANICAL GARDENS...|      -1|     25.0|     13644|United Kingdom|2011-12-01 08:18:...|
|  C579891|    23186|FRENCH STYLE STOR...|      -6|     0.29|     13644|United Kingdom|2011-12-01 08:18:...|
|  C579892|    23461|SWEETHEART BIRD H...|      -4|     4.15|     13310|United Kingdom|2011-12-01 08:23:...|
|  C579893|    37449|CERAMIC CAKE STAN...|      -2|     9.95|     13468|United Kingdom|2011-12-01 08:25:...|
|  C579894|    23012|GLASS APOTHECARY ...|      -1|     3.95|     13098|United Kingdom|2011-12-01 08:26:...|
|  C579894|    23568|EGG CUP HENRIETTA...|      -1|     1.25|     13098|United Kingdom|2011-12-01 08:26:...|
|  C579894|    23319|BOX OF 6 MINI 50'...|      -6|     2.08|     13098|United Kingdom|2011-12-01 08:26:...|
|  C579895|    23374|RED SPOT PAPER GI...|      -5|     0.82|     14329|United Kingdom|2011-12-01 08:27:...|
|  C579896|    23456|MEDIUM PARLOUR PI...|      -2|     4.15|     13971|United Kingdom|2011-12-01 08:27:...|
|  C579897|    23382|BOX OF 6 CHRISTMA...|      -2|     3.75|     17636|United Kingdom|2011-12-01 08:29:...|
|  C579897|    23256|CHILDRENS CUTLERY...|      -1|     4.15|     17636|United Kingdom|2011-12-01 08:29:...|
|  C579897|    23203|JUMBO BAG VINTAGE...|      -1|     2.08|     17636|United Kingdom|2011-12-01 08:29:...|
|  C579897|    22470|HEART OF WICKER L...|      -2|     2.95|     17636|United Kingdom|2011-12-01 08:29:...|
|  C579898|    22153|ANGEL DECORATION ...|      -1|     0.42|     14299|United Kingdom|2011-12-01 08:32:...|
|  C579898|    22969|HOMEMADE JAM SCEN...|      -1|     1.45|     14299|United Kingdom|2011-12-01 08:32:...|
|   579899|    23301|GARDENERS KNEELIN...|      24|     1.65|     15687|United Kingdom|2011-12-01 08:33:...|
+---------+---------+--------------------+--------+---------+----------+--------------+--------------------+

only showing top 20 rows
```

```
Aurobindos-MacBook-Pro-2:Downloadsaurobindosarkar$ mongo
MongoDB shell version: 3.2.0
connecting to: test
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
        http://docs.mongodb.org/
Questions? Try the support group
        http://groups.google.com/group/mongodb-user
Server has startup warnings:
2015-12-27T17:40:17.264+0530 I CONTROL  [initandlisten]
2015-12-27T17:40:17.264+0530 I CONTROL  [initandlisten] ** WARNING: soft rlimits too low. Number of files is
256, should be at least 1000
>
```

```
{
     "_id" : ObjectId("57f9e1598a793a2f1013dfed"),
     "dbn" : "17K548",
     "school_name" : "Brooklyn School for Music & Theatre",
     "boro" : "Brooklyn",
     "building_code" : "K440",
     "phone_number" : "718-230-6250",
     "fax_number" : "718-230-6262",
     "grade_span_min" : 9,
     "grade_span_max" : 12,
     "expgrade_span_min" : "",
     "expgrade_span_max" : "",
     "bus" : "B41, B43, B44-SBS, B45, B48, B49, B69",
     "subway" : "2, 3, 4, 5, F, S to Botanic Garden ; B, Q to Prospect Park",
     "primary_address_line_1" : "883 Classon Avenue",
     "city" : "Brooklyn",
     "state_code" : "NY",
     "zip" : 11225,
     "website" : "Bkmusicntheatre.com",
     "total_students" : 399,
     "campus_name" : "Prospect Heights Educational Campus",
       "school_type" : "",
```

```
"overview_paragraph" : "Brooklyn School for Music & Theatre (BSMT) uses our
academic program to accommodate the intellectual, social, emotional and
physical needs of creative high school students. Our vision is to provide a
model professional environment where respect is mutual, ideas are shared and
learning is not limited to the classroom. We prepare students for higher
education and professional careers in the music and theatre industries.",
     "program_highlights" : "We offer highly competitive positions in our
Drama, Chorus and Dance Company wherein students receive small group
instruction focused on sharpening and furthering their skills while developing
their professional portfolio for auditions in their chosen field.",
     "language_classes" : "Spanish",
     "advancedplacement_courses" : "English Language and Composition, United
States History",
     "online_ap_courses" : "",
     "online_language_courses" : "",
     "extracurricular_activities" : "Variety of clubs: Chess, The Step Team,
Fashion, Tech Team, Women's Group; Extensive arts after-school program: Dance
Company, Drama Company and Chorus Company, back stage crew program that trains
students in running the lights, sound, video and all back stage and pit crew
responsibilities; Saturday and After-school classes for Regents Preparation;
School Leadership Team; Student Government; at least three annual major
school-wide productions; two annual talent shows",
     "psal_sports_boys" : "Baseball, Basketball & JV Basketball, Cross
Country, Indoor Track, Outdoor Track, Soccer, Volleyball",
     "psal_sports_girls" : "Basketball, Cross Country, Indoor Track, Outdoor
Track, Soccer, Softball",
     "psal_sports_coed" : "",
     "school_sports" : "",
```

```
"partner_cbo" : "F.Y.R.EZONE (Finding Your Rhythm thru Education) is an
entertainment company built on high academic expectations and is committed to
meaningful learning. We are vested in the “whole” child. Our engaging teaching
and effective programs will be challenged and guided to academic success.
FYREZONE is committed to enhancing self-esteem, self awareness, preventing
drop-out, and most importantly instilling a confidence that can take them
through their Junior High, High School, college years.",
      "partner_hospital" : "",
      "partner_highered" : "",
      "partner_cultural" : "In 2002, Roundabout Theatre was selected by New
York City Department of Education to help design, develop, and operate
Brooklyn School for Music and Theatre. Since the school’s development,
Roundabout has provided year-long programs connecting the process of theatre
production to project-based learning objectives and curriculum standards.
Step-in-School Inc is an enrichment programs that works to teach Step, aligned
with all of its historic, artistic and physio-educational components.
Additionally, Step-in-School works to connect the developmental assets of this
art form to college preparatory services and youth professional
developmental.",
      "partner_nonprofit" : "One To World’s Global Classroom connects New York
City youth with trained, international university scholars through interactive
workshops that engage students in learning about world cultures and global
issues. Through face-to-face interactions and meaningful cross-cultural
exchange with international leaders of tomorrow, today’s New York City K-12
students develop the skills, awareness and understanding to become global
citizens in their communities, both locally and worldwide.",
```

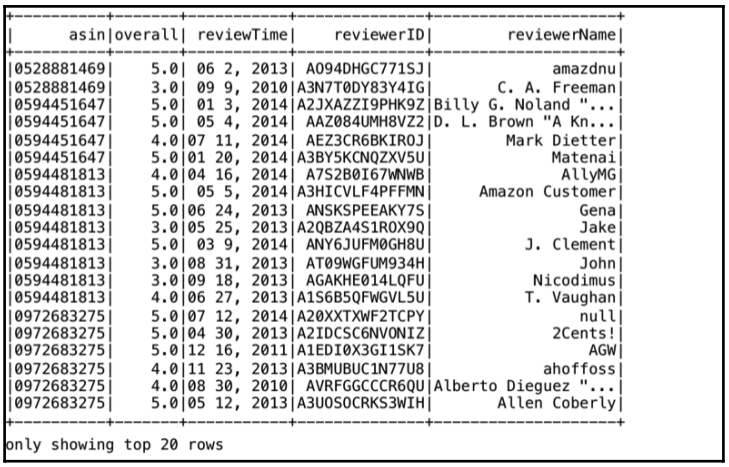
```
"partner_corporate" : "",
      "partner_financial" : "",
      "partner_other" : "",
      "addtl_info1" : "",
      "addtl_info2" : "",
      "start_time" : "8:10 AM",
      "end_time" : "3:00 PM",
      "se_services" : "This school will provide students with disabilities the
supports and services indicated on their IEPs.",
      "ell_programs" : "ESL",
      "school_accessibility_description" : "Functionally Accessible",
      "number_programs" : 1,
      "priority01" : "Priority to Brooklyn students or residents",
      "priority02" : "Then to New York City residents",
      "priority03" : "",
      "priority04" : "",
      "priority05" : "",
      "priority06" : "",
      "priority07" : "",
      "priority08" : "",
      "priority09" : "",
      "priority10" : "",
      "Location 1" : "883 Classon Avenue\nBrooklyn, NY
11225\n(40.67029890700047, -73.96164787599963)"
}
```

```
scala> case class School(dbn: String, school_name: String, boro: String,
building_code: String, phone_number: String, fax_number: String,
grade_span_min: String, grade_span_max: String, expgrade_span_min: String,
expgrade_span_max: String, bus: String, subway: String,
primary_address_line_1: String, city: String, state_code: String, zip: String,
website: String, total_students: String, campus_name: String, school_type:
String, overview_paragraph: String, program_highlights: String,
language_classes: String, advancedplacement_courses: String,
online_ap_courses: String, online_language_courses: String,
extracurricular_activities: String, psal_sports_boys: String,
psal_sports_girls: String, psal_sports_coed: String, school_sports: String,
partner_cbo: String, partner_hospital: String, partner_highered: String,
partner_cultural: String, partner_nonprofit: String, partner_corporate:
String, partner_financial: String, partner_other: String, addtl_info1: String,
addtl_info2: String, start_time: String, end_time: String, se_services:
String, ell_programs: String, school_accessibility_description: String,
number_programs: String, priority01: String, priority02: String, priority03:
String, priority04: String, priority05: String, priority06: String,
priority07: String, priority08: String, priority09: String, priority10:
String, Location_1: String)
```

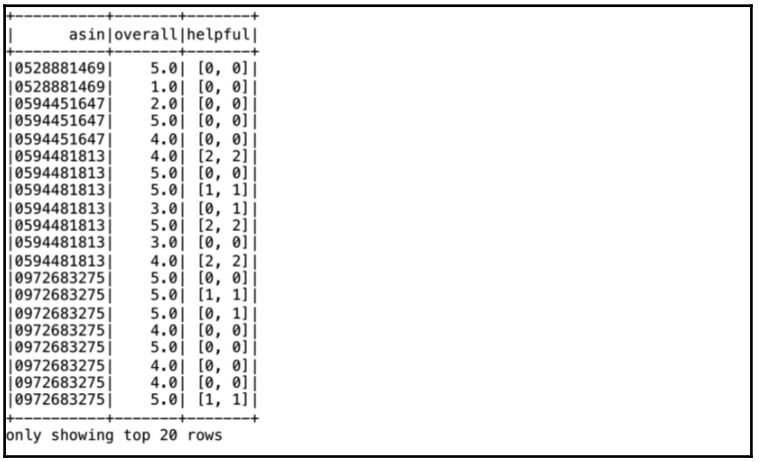
```
[17K548,Brooklyn School for Music & Theatre,Brooklyn,K440,718-230-6250,718-
230-6262,9,12,,,B41, B43, B44-SBS, B45, B48, B49, B69,2, 3, 4, 5, F, S to
Botanic Garden ; B, Q to Prospect Park,883 Classon
Avenue,Brooklyn,NY,11225,Bkmusicntheatre.com,399,Prospect Heights Educational
Campus,,Brooklyn School for Music & Theatre (BSMT) uses our academic program
to accommodate the intellectual, social, emotional and physical needs of
creative high school students. Our vision is to provide a model professional
environment where respect is mutual, ideas are shared and learning is not
limited to the classroom. We prepare students for higher education and
professional careers in the music and theatre industries.,We offer highly
competitive positions in our Drama, Chorus and Dance Company wherein students
receive small group instruction focused on sharpening and furthering their
skills while developing their professional portfolio for auditions in their
chosen field.,Spanish,English Language and Composition, United States
History,,,Variety of clubs: Chess, The Step Team, Fashion, Tech Team, Women's
Group; Extensive arts after-school program: Dance Company, Drama Company and
Chorus Company, back stage crew program that trains students in running the
lights, sound, video and all back stage and pit crew responsibilities;
Saturday and After-school classes for Regents Preparation;
```

```
School Leadership Team; Student Government; at least three annual major
school-wide productions; two annual talent shows,Baseball, Basketball & JV
Basketball, Cross Country, Indoor Track, Outdoor Track, Soccer,
Volleyball,Basketball, Cross Country, Indoor Track, Outdoor Track, Soccer,
Softball,,,F.Y.R.EZONE (Finding Your Rhythm thru Education) is an
entertainment company built on high academic expectations and is committed to
meaningful learning. We are vested in the “whole” child. Our engaging teaching
and effective programs will be challenged and guided to academic success.
FYREZONE is committed to enhancing self-esteem, self awareness, preventing
drop-out, and most importantly instilling a confidence that can take them
through their Junior High, High School, college years.,,,In 2002, Roundabout
Theatre was selected by New York City Department of Education to help design,
develop, and operate Brooklyn School for Music and Theatre. Since the school’s
development, Roundabout has provided year-long programs connecting the process
of theatre production to project-based learning objectives and curriculum
standards. Step-in-School Inc is an enrichment programs that works to teach
Step, aligned with all of its historic, artistic and physio-educational
components. Additionally, Step-in-School works to connect the developmental
assets of this art form to college preparatory services and youth professional
developmental.,One To World‘s Global Classroom connects New York City youth
with trained, international university scholars through interactive workshops
that engage students in learning about world cultures and global issues.
Through face-to-face interactions and meaningful cross-cultural exchange with
international leaders of tomorrow, today’s New York City K-12 students develop
the skills, awareness and understanding to become global citizens in their
communities, both locally and worldwide.,,,,,,8:10 AM,3:00 PM,This school will
provide students with disabilities the supports and services indicated on
their IEPs.,ESL,Functionally Accessible,1,Priority to Brooklyn students or
residents,Then to New York City residents,,,,,,,,,null]
```

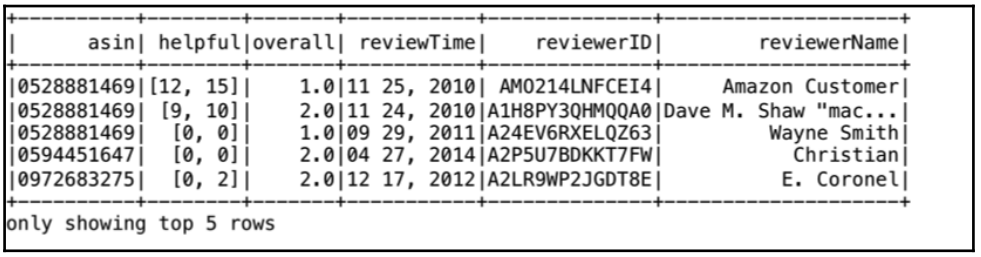
```
root
 |-- asin: string (nullable = true)
 |-- helpful: array (nullable = true)
 |    |-- element: long (containsNull = true)
 |-- overall: double (nullable = true)
 |-- reviewText: string (nullable = true)
 |-- reviewTime: string (nullable = true)
 |-- reviewerID: string (nullable = true)
 |-- reviewerName: string (nullable = true)
 |-- summary: string (nullable = true)
 |-- unixReviewTime: long (nullable = true)
```



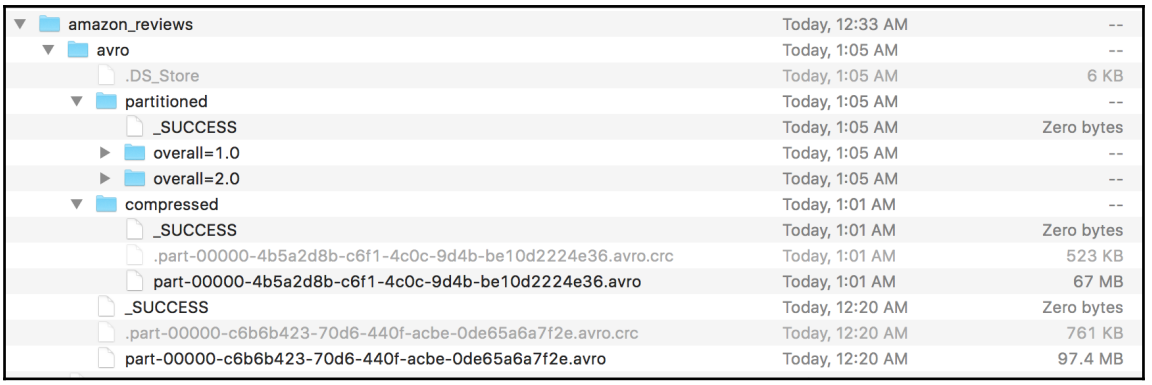
```
+----------+-------+----------+--------------+--------------------+
|      asin|overall| reviewTime|    reviewerID|        reviewerName|
+----------+-------+----------+--------------+--------------------+
|0528881469|    5.0| 06 2, 2013| AO94DHGC771SJ|             amazdnu|
|0528881469|    3.0| 09 9, 2010|A3N7T0DY83Y4IG|       C. A. Freeman|
|0594451647|    5.0| 01 3, 2014|A2JXAZZI9PHK9Z|Billy G. Noland "...|
|0594451647|    5.0| 05 4, 2014| AAZ084UMH8VZ2|D. L. Brown "A Kn...|
|0594451647|    4.0|07 11, 2014| AEZ3CR6BKIROJ|        Mark Dietter|
|0594451647|    5.0|01 20, 2014|A3BY5KCNQZXV5U|             Matenai|
|0594481813|    4.0|04 16, 2014| A7S2B0I67WNWB|              AllyMG|
|0594481813|    5.0| 05 5, 2014|A3HICVLF4PFFMN|     Amazon Customer|
|0594481813|    5.0|06 24, 2013| ANSKSPEEAKY7S|                Gena|
|0594481813|    3.0|05 25, 2013|A2QBZA4S1ROX9Q|                Jake|
|0594481813|    5.0| 03 9, 2014| ANY6JUFM0GH8U|          J. Clement|
|0594481813|    3.0|08 31, 2013| AT09WGFUM934H|                John|
|0594481813|    3.0|09 18, 2013| AGAKHE014LQFU|           Nicodimus|
|0594481813|    4.0|06 27, 2013|A1S6B5QFWGVL5U|          T. Vaughan|
|0972683275|    5.0|07 12, 2014|A20XXTXWF2TCPY|                null|
|0972683275|    5.0|04 30, 2013|A2IDCSC6NVONIZ|             2Cents!|
|0972683275|    5.0|12 16, 2011|A1EDI0X3GI1SK7|                 AGW|
|0972683275|    4.0|11 23, 2013|A3BMUBUC1N77U8|             ahoffoss|
|0972683275|    4.0|08 30, 2010| AVRFGGCCCR6QU|Alberto Dieguez "...|
|0972683275|    5.0|05 12, 2013|A3UOSOCRKS3WIH|       Allen Coberly|
+----------+-------+----------+--------------+--------------------+
only showing top 20 rows
```



```
+----------+-------+-------+
|      asin|overall|helpful|
+----------+-------+-------+
|0528881469|    5.0| [0, 0]|
|0528881469|    1.0| [0, 0]|
|0594451647|    2.0| [0, 0]|
|0594451647|    5.0| [0, 0]|
|0594451647|    4.0| [0, 0]|
|0594481813|    4.0| [2, 2]|
|0594481813|    5.0| [0, 0]|
|0594481813|    5.0| [1, 1]|
|0594481813|    3.0| [0, 1]|
|0594481813|    5.0| [2, 2]|
|0594481813|    3.0| [0, 0]|
|0594481813|    4.0| [2, 2]|
|0972683275|    5.0| [0, 0]|
|0972683275|    5.0| [1, 1]|
|0972683275|    5.0| [0, 1]|
|0972683275|    4.0| [0, 0]|
|0972683275|    5.0| [0, 0]|
|0972683275|    4.0| [0, 0]|
|0972683275|    4.0| [0, 0]|
|0972683275|    5.0| [1, 1]|
+----------+-------+-------+
only showing top 20 rows
```



```
+----------+--------+-------+----------+--------------+--------------------+
|      asin| helpful|overall| reviewTime|    reviewerID|        reviewerName|
+----------+--------+-------+----------+--------------+--------------------+
|0528881469|[12, 15]|    1.0|11 25, 2010| AMO214LNFCEI4|     Amazon Customer|
|0528881469| [9, 10]|    2.0|11 24, 2010|A1H8PY3QHMQQA0|Dave M. Shaw "mac...|
|0528881469|  [0, 0]|    1.0|09 29, 2011|A24EV6RXELQZ63|         Wayne Smith|
|0594451647|  [0, 0]|    2.0|04 27, 2014|A2P5U7BDKKT7FW|           Christian|
|0972683275|  [0, 2]|    2.0|12 17, 2012|A2LR9WP2JGDT8E|          E. Coronel|
+----------+--------+-------+----------+--------------+--------------------+
only showing top 5 rows
```




```
+----------+-------+--------------+------------------------------+
|asin      |overall|reviewerID    |reviewerName                  |
+----------+-------+--------------+------------------------------+
|0528881469|1.0    |AMO214LNFCEI4 |Amazon Customer               |
|0528881469|1.0    |A24EV6RXELQZ63|Wayne Smith                   |
|0972683275|1.0    |A2JMN2JA9LSHVL|Eric G. Gruner                |
|0972683275|1.0    |A38FGQVJM18OWV|George S. Mitchell "gsmitchell"|
|0972683275|1.0    |A15K7HV1XD6YWR|Musawir Karim                 |
+----------+-------+--------------+------------------------------+
only showing top 5 rows
```

Chapter 3:
Using Spark SQL for Data Exploration

```
scala> :paste
// Entering paste mode (ctrl-D to finish)

import org.apache.spark.sql.types._
import spark.implicits._

val age = StructField("age", DataTypes.IntegerType)
val job  = StructField("job", DataTypes.StringType)
val marital  = StructField("marital", DataTypes.StringType)
val edu  = StructField("edu", DataTypes.StringType)
val credit_default  = StructField("credit_default", DataTypes.StringType)
val housing  = StructField("housing", DataTypes.StringType)
val loan  = StructField("loan", DataTypes.StringType)
val contact  = StructField("contact", DataTypes.StringType)
val month  = StructField("month", DataTypes.StringType)
val day  = StructField("day", DataTypes.StringType)
val dur  = StructField("dur", DataTypes.DoubleType)
val campaign  = StructField("campaign", DataTypes.DoubleType)
val pdays  = StructField("pdays", DataTypes.DoubleType)
val prev  = StructField("prev", DataTypes.DoubleType)
val pout  = StructField("pout", DataTypes.StringType)
val emp_var_rate  = StructField("emp_var_rate", DataTypes.DoubleType)
val cons_price_idx  = StructField("cons_price_idx", DataTypes.DoubleType)
val cons_conf_idx  = StructField("cons_conf_idx", DataTypes.DoubleType)
val euribor3m  = StructField("euribor3m", DataTypes.DoubleType)
val nr_employed  = StructField("nr_employed", DataTypes.DoubleType)
val deposit  = StructField("deposit", DataTypes.StringType)

val fields = Array(age, job, marital, edu, credit_default, housing, loan, contact, month, day, dur, campaign, pdays, prev, pout,
emp_var_rate, cons_price_idx, cons_conf_idx, euribor3m, nr_employed, deposit)

val schema = StructType(fields)

val df = spark.read.schema(schema).option("sep", ";").option("header", true).csv("file:///Users/aurobindosarkar/Downloads/bank-
additional/bank-additional-full.csv")

// Exiting paste mode, now interpreting.
```

```
scala> df.count()
res0: Long = 41188
```

```
scala> case class Call(age: Double, job: String, marital: String, edu: String, credit_default: String, housing: String, loan: String,
contact: String, month: String, day: String, dur: Double, campaign: Double, pdays: Double, prev: Double, pout: String, emp_var_rate:
Double, cons_price_idx: Double, cons_conf_idx: Double, euribor3m: Double, nr_employed: Double, deposit: String)

scala> val ds = df.as[Call]

scala> ds.printSchema()
root
 |-- age: integer (nullable = true)
 |-- job: string (nullable = true)
 |-- marital: string (nullable = true)
 |-- edu: string (nullable = true)
 |-- credit_default: string (nullable = true)
 |-- housing: string (nullable = true)
 |-- loan: string (nullable = true)
 |-- contact: string (nullable = true)
 |-- month: string (nullable = true)
 |-- day: string (nullable = true)
 |-- dur: double (nullable = true)
 |-- campaign: double (nullable = true)
 |-- pdays: double (nullable = true)
 |-- prev: double (nullable = true)
 |-- pout: string (nullable = true)
 |-- emp_var_rate: double (nullable = true)
 |-- cons_price_idx: double (nullable = true)
 |-- cons_conf_idx: double (nullable = true)
 |-- euribor3m: double (nullable = true)
 |-- nr_employed: double (nullable = true)
 |-- deposit: string (nullable = true)
```

```
scala> val dfMissing = spark.read.schema(schema).option("sep", ";").option("header", true).csv("file:///Users/aurobindosarkar/
Downloads/bank-additional/bank-additional-full-with-missing.csv")

scala> val dfMissing = spark.read.schema(schema).option("sep", ";").option("header", true).csv("file:///Users/aurobindosarkar/
Downloads/bank-additional/bank-additional-full-with-missing.csv")

scala> val dsMissing = dfMissing.as[Call]
```

```
scala> dsMissing.groupBy("marital").count().show()
+--------+-----+
| marital|count|
+--------+-----+
|    null|   80|
|divorced| 4612|
| married|24928|
|  single|11568|
+--------+-----+


scala> dsMissing.groupBy("job").count().show()
+-------------+-----+
|          job|count|
+-------------+-----+
|   management| 2924|
|      retired| 1720|
|         null|  330|
|self-employed| 1421|
|      student|  875|
|  blue-collar| 9254|
| entrepreneur| 1456|
|       admin.|10422|
|   technician| 6743|
|     services| 3969|
|    housemaid| 1060|
|   unemployed| 1014|
+-------------+-----+
```

```
scala> case class CallStats(age: Int, dur: Double, campaign: Double, prev:
Double, deposit: String)

scala> val dsSubset = ds.select($"age", $"dur", $"campaign", $"prev",
$"deposit")

scala> dsSubset.show(5)
+---+---+--------+----+-------+
|age|dur|campaign|prev|deposit|
+---+---+--------+----+-------+
| 56|261|       1|   0|     no|
| 57|149|       1|   0|     no|
| 37|226|       1|   0|     no|
| 40|151|       1|   0|     no|
| 56|307|       1|   0|     no|
+---+---+--------+----+-------+
only showing top 5 rows

scala> val dsCallStats = dsSubset.as[CallStats]

scala> dsCallStats.cache()
```

```
scala> dsSubset.describe().show()
+-------+------------------+------------------+-----------------+-------------------+
|summary|               age|               dur|         campaign|               prev|
+-------+------------------+------------------+-----------------+-------------------+
|  count|             41188|             41188|            41188|              41188|
|   mean| 40.02406040594348| 258.2850101971448|2.567592502670681|0.17296299893172767|
| stddev|10.421249980934057|259.27924883646455|2.770013542902331|0.49490107983928927|
|    min|                17|                 0|                1|                  0|
|    max|                98|              4918|               56|                  7|
+-------+------------------+------------------+-----------------+-------------------+
```

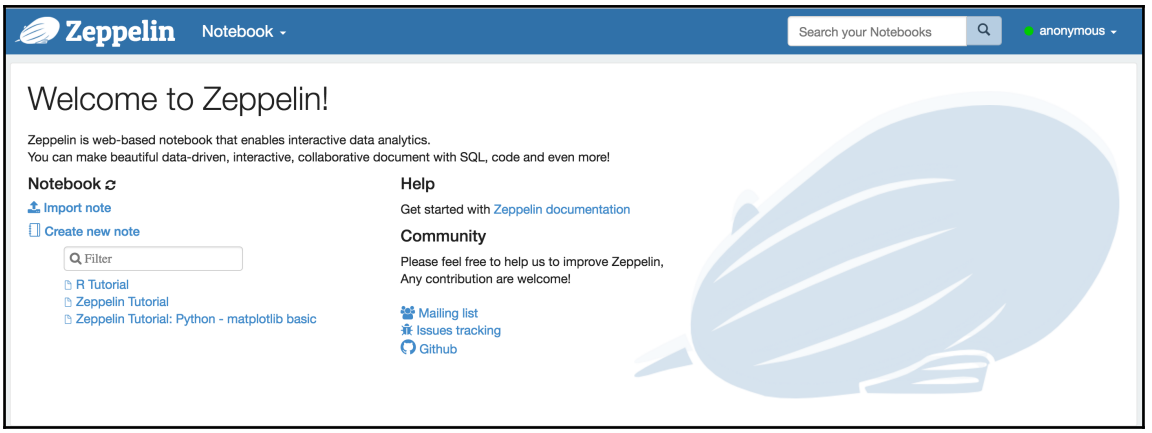
```
scala> val cov = dsSubset.stat.cov("age","dur")
cov: Double = -2.3391469421267863

scala> val corr = dsSubset.stat.corr("age","dur")
corr: Double = -8.657050101409879E-4
```

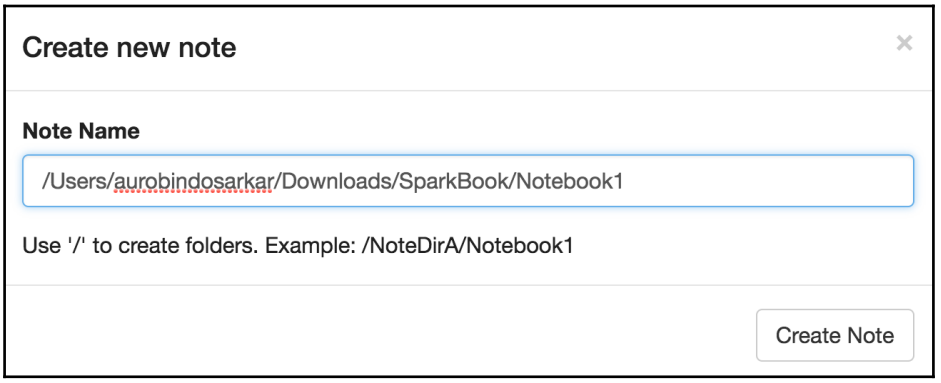
```
scala> ds.stat.crosstab("age", "marital").orderBy("age_marital").show(10)
+-----------+--------+-------+------+-------+
|age_marital|divorced|married|single|unknown|
+-----------+--------+-------+------+-------+
|         17|       0|      0|     5|      0|
|         18|       0|      0|    28|      0|
|         19|       0|      0|    42|      0|
|         20|       0|      1|    64|      0|
|         21|       0|      8|    94|      0|
|         22|       0|     16|   121|      0|
|         23|       0|     30|   196|      0|
|         24|       4|     78|   381|      0|
|         25|      17|    150|   429|      2|
|         26|      13|    196|   489|      0|
+-----------+--------+-------+------+-------+
only showing top 10 rows

scala> val freq = df.stat.freqItems(Seq("edu"), 0.3)

scala> freq.collect()(0)
res137: org.apache.spark.sql.Row = [WrappedArray(high.school,
university.degree, professional.course)]

scala> val quantiles = df.stat.approxQuantile("age", Array(0.25,0.5,0.75),0.0)
quantiles: Array[Double] = Array(32.0, 38.0, 47.0)
```

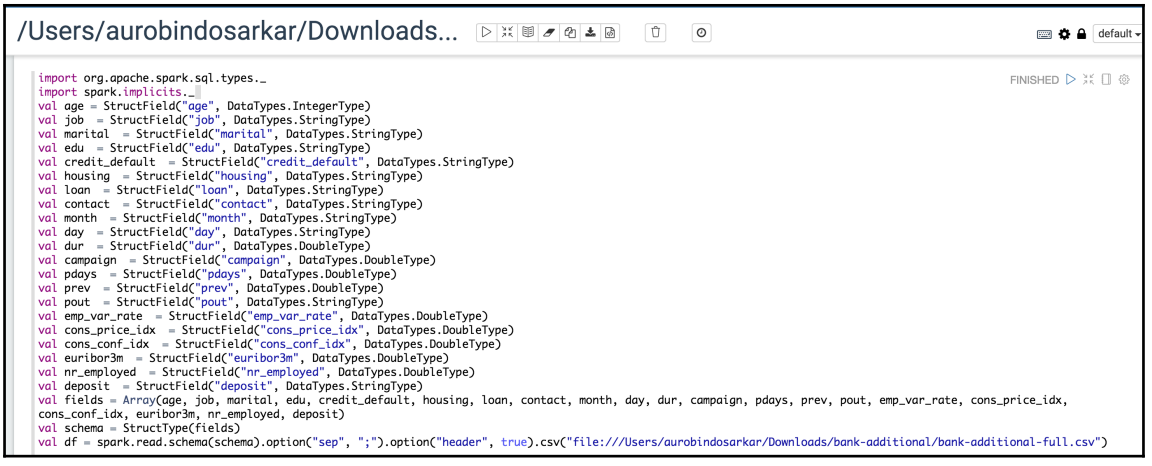
```
scala> import org.apache.spark.sql.expressions.scalalang.typed.{
     |   count => typedCount,
     |   avg => typedAvg,
     |   sum => typedSum}

scala> (dsCallStats.groupByKey(callstats =>
callstats.deposit).agg(typedCount[CallStats](_.age).name("A"),typedAvg[CallSta
ts](_.campaign).name("B"),typedAvg[CallStats](_.dur).name("C"),typedAvg[CallSt
ats](_.prev).name("D")).withColumnRenamed("value", "E")).select($"E".name("TD
Subscribed?"), $"A".name("Total Customers"), round($"B", 2).name("Avg
calls(curr)"), round($"C", 2).name("Avg dur"), round($"D", 2).name("Avg
calls(prev)")).show()
+--------------+---------------+---------------+-------+---------------+
|TD Subscribed?|Total Customers|Avg calls(curr)|Avg dur|Avg calls(prev)|
+--------------+---------------+---------------+-------+---------------+
|            no|          36548|           2.63| 220.84|           0.13|
|           yes|           4640|           2.05| 553.19|           0.49|
+--------------+---------------+---------------+-------+---------------+
```

```
scala> (dsCallStats.groupByKey(callstats =>
callstats.age).agg(typedCount[CallStats](_.age).name("A"),typedAvg[CallStats](
_.campaign).name("B"),typedAvg[CallStats](_.dur).name("C"),typedAvg[CallStats]
(_.prev).name("D")).withColumnRenamed("value", "E")).select($"E".name("Age"),
$"A".name("Total Customers"), round($"B", 2).name("Avg calls(curr)"),
round($"C", 2).name("Avg dur"), round($"D", 2).name("Avg
calls(prev)")).orderBy($"age").show(5)
+---+---------------+---------------+-------+---------------+
|Age|Total Customers|Avg calls(curr)|Avg dur|Avg calls(prev)|
+---+---------------+---------------+-------+---------------+
| 17|              5|            2.2|  420.0|            1.8|
| 18|             28|           1.32| 321.79|           0.75|
| 19|             42|           2.29|  271.5|           0.67|
| 20|             65|           2.35| 288.49|           0.63|
| 21|            102|           2.03| 264.25|           0.28|
+---+---------------+---------------+-------+---------------+
only showing top 5 rows
```

```
scala> import org.apache.spark.mllib.linalg.Vectors

scala> import org.apache.spark.mllib.clustering.KMeans

scala> val vectors = df.rdd.map(r =>
Vectors.dense(r.getDouble(10),r.getDouble(11), r.getDouble(12),
r.getDouble(13)))

scala> vectors.cache()
res13: vectors.type = MapPartitionsRDD[71] at map at <console>:80

scala> val kMeansModel = KMeans.train(vectors, 2, 20)

scala> kMeansModel.clusterCenters.foreach(println)
[182.0339819280448,2.590526082377072,965.4441765064582,0.17010366428294252]
[796.6381604696674,2.405675146771037,941.5154598825832,0.19315068493150686]
```


Zeppelin
Notebook
Search your Notebooks
anonymous
Welcome to Zeppelin!
Zeppelin is web-based notebook that enables interactive data analytics.
You can make beautiful data-driven, interactive, collaborative document with SQL, code and even more!
Notebook
Import note
Create new note
Filter
R Tutorial
Zeppelin Tutorial
Zeppelin Tutorial: Python - matplotlib basic
Help
Get started with Zeppelin documentation
Community
Please feel free to help us to improve Zeppelin,
Any contribution are welcome!
Mailing list
Issues tracking
Github


Create new note
Note Name
/Users/aurobindosarkar/Downloads/SparkBook/Notebook1
Use '/' to create folders. Example: /NoteDirA/Notebook1
Create Note


/Users/aurobindosarkar/Downloads...
default
FINISHED
import org.apache.spark.sql.types._
import spark.implicits._
val age = StructField("age", DataTypes.IntegerType)
val job = StructField("job", DataTypes.StringType)
val marital = StructField("marital", DataTypes.StringType)
val edu = StructField("edu", DataTypes.StringType)
val credit_default = StructField("credit_default", DataTypes.StringType)
val housing = StructField("housing", DataTypes.StringType)
val loan = StructField("loan", DataTypes.StringType)
val contact = StructField("contact", DataTypes.StringType)
val month = StructField("month", DataTypes.StringType)
val day = StructField("day", DataTypes.StringType)
val dur = StructField("dur", DataTypes.DoubleType)
val campaign = StructField("campaign", DataTypes.DoubleType)
val pdays = StructField("pdays", DataTypes.DoubleType)
val prev = StructField("prev", DataTypes.DoubleType)
val pout = StructField("pout", DataTypes.StringType)
val emp_var_rate = StructField("emp_var_rate", DataTypes.DoubleType)
val cons_price_idx = StructField("cons_price_idx", DataTypes.DoubleType)
val cons_conf_idx = StructField("cons_conf_idx", DataTypes.DoubleType)
val euribor3m = StructField("euribor3m", DataTypes.DoubleType)
val nr_employed = StructField("nr_employed", DataTypes.DoubleType)
val deposit = StructField("deposit", DataTypes.StringType)
val fields = Array(age, job, marital, edu, credit_default, housing, loan, contact, month, day, dur, campaign, pdays, prev, pout, emp_var_rate, cons_price_idx,
cons_conf_idx, euribor3m, nr_employed, deposit)
val schema = StructType(fields)
val df = spark.read.schema(schema).option("sep", ";").option("header", true).csv("file:///Users/aurobindosarkar/Downloads/bank-additional/bank-additional-full.csv")

```
df.select($"age", $"job", $"marital", $"edu").show()
```

```
+---+-----------+--------+------------------+
|age|        job| marital|               edu|
+---+-----------+--------+------------------+
| 56|  housemaid| married|           basic.4y|
| 57|   services| married|        high.school|
| 37|   services| married|        high.school|
| 40|     admin.| married|           basic.6y|
| 56|   services| married|        high.school|
| 45|   services| married|           basic.9y|
| 59|     admin.| married|professional.course|
| 41|blue-collar| married|            unknown|
| 24| technician|  single|professional.course|
| 25|   services|  single|        high.school|
| 41|blue-collar| married|            unknown|
| 25|   services|  single|        high.school|
| 29|blue-collar|  single|        high.school|
| 57|  housemaid|divorced|           basic.4y|
| 35|blue-collar| married|           basic.6v|
```

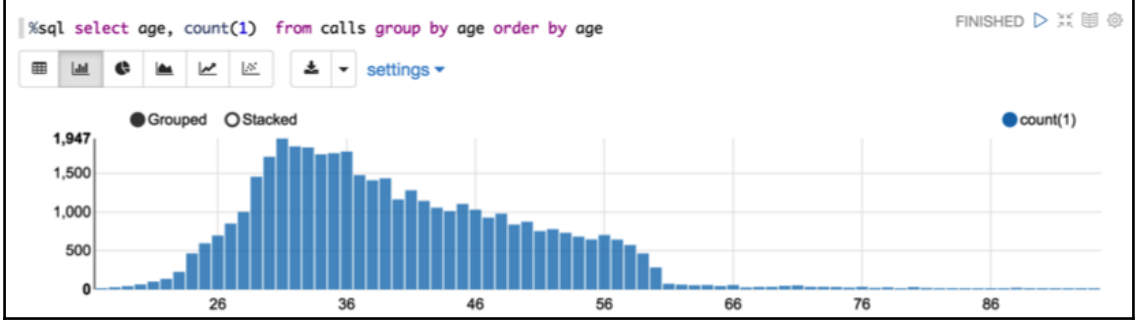
Took 1 sec. Last updated by anonymous at November 12 2016, 5:11:56 PM.

```
df.createOrReplaceTempView("calls")
```

Took 0 sec. Last updated by anonymous at November 10 2016, 5:45:08 AM.

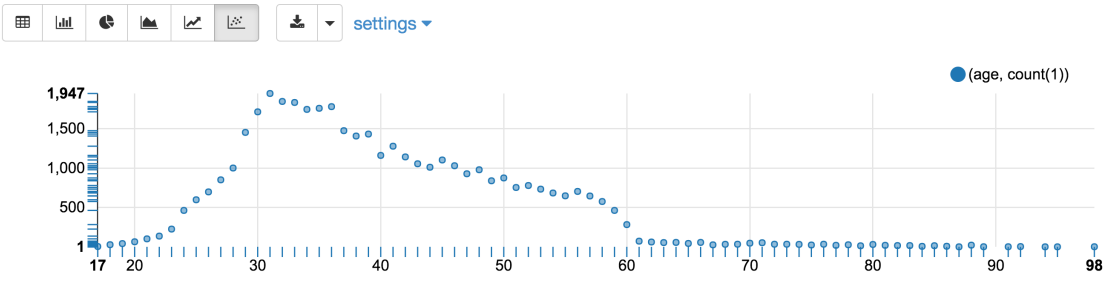
```
%sql select age, count(1) from calls group by age order by age
```

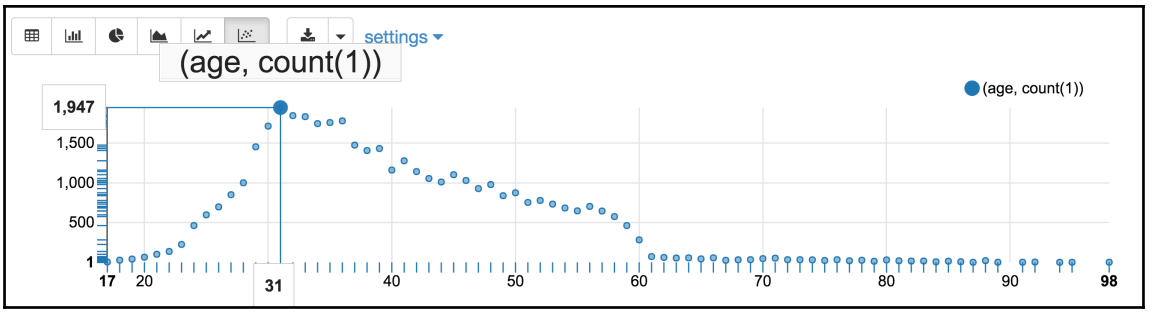
FINISHED

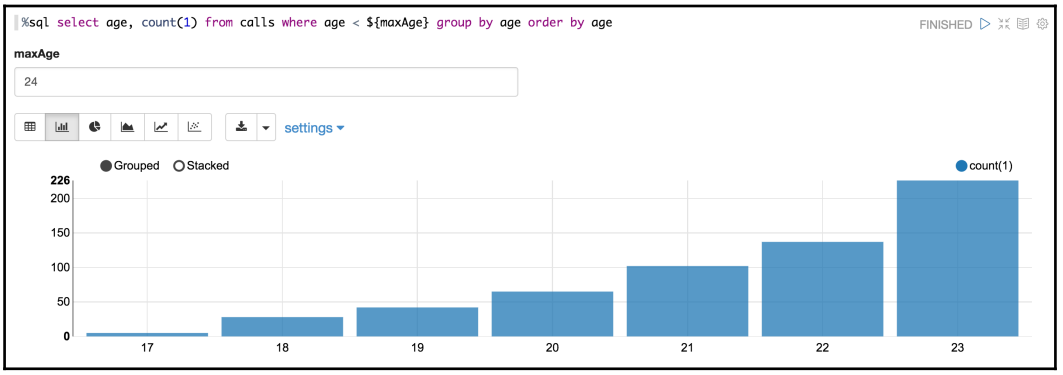


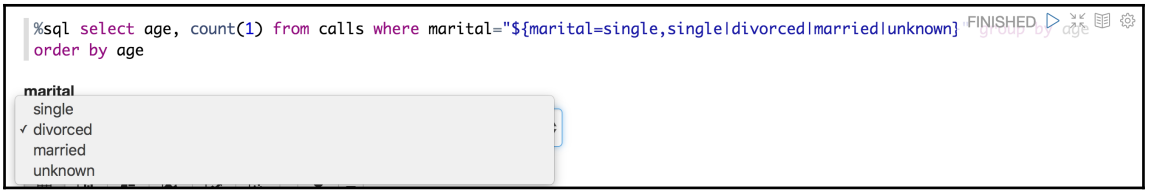

```
%sql select age, count(1) from calls group by age order by age
```

FINISHED




settings
(age, count(1))
(age, count(1))
1,947
1,500
1,000
500
1
17
20
31
40
50
60
70
80
90
98


%sql select age, count(1) from calls where age < ${maxAge} group by age order by age
FINISHED
maxAge
24
settings
Grouped
Stacked
count(1)
226
200
150
100
50
0
17
18
19
20
21
22
23


%sql select age, count(1) from calls where marital="${marital=single,single|divorced|married|unknown}
order by age
FINISHED
marital
single
divorced
married
unknown

```
%sql select age, count(1) from calls where marital="${marital=single,single|divorced|married|unknown}
order by age
```

FINISHED

marital

single

age	count(1)
17	5
18	28
19	42
20	64
21	94

```
scala> import scala.collection.immutable.Map

scala> val fractions = Map("unknown" -> .10, "divorced" -> .15, "married" ->
0.5, "single" -> .25)

scala> val dsStratifiedSample = ds.stat.sampleBy("marital", fractions, 36L)

scala> dsStratifiedSample.count()
res15: Long = 16014
```

```
scala> val dsSampleWithReplacement = ds.sample(true, .10)

scala>
dsSampleWithReplacement.groupBy("marital").count().orderBy("marital").show()
+--------+-----+
| marital|count|
+--------+-----+
|divorced|  472|
| married| 2496|
|  single| 1162|
| unknown|    8|
+--------+-----+
```

```
scala> import org.apache.spark.mllib.linalg.Vector
import org.apache.spark.mllib.linalg.Vector

scala> val rowsRDD = df.rdd.map(r => (r.getAs[String](2), List(r.getInt(0),
r.getString(1), r.getString(2), r.getString(3), r.getString(4),
r.getString(5), r.getString(6), r.getString(7), r.getString(8),
r.getString(9), r.getDouble(10), r.getDouble(11), r.getDouble(12),
r.getDouble(13), r.getString(14), r.getDouble(15), r.getDouble(16),
r.getDouble(17), r.getDouble(18), r.getDouble(19), r.getString(20))))

scala> rowsRDD.take(2).foreach(println)
(married,List(56, housemaid, married, basic.4y, no, no, no, telephone, may,
mon, 261.0, 1.0, 999.0, 0.0, nonexistent, 1.1, 93.994, -36.4, 4.857, 5191.0,
no))
(married,List(57, services, married, high.school, unknown, no, no, telephone,
may, mon, 149.0, 1.0, 999.0, 0.0, nonexistent, 1.1, 93.994, -36.4, 4.857,
5191.0, no))
```

```
scala> val fractions = Map("unknown" -> .10, "divorced" -> .15, "married" ->
0.5, "single" -> .25)
```

```
scala> val rowsSampleRDD = rowsRDD.sampleByKey(true, fractions, 1)

scala> val rowsSampleRDDExact = rowsRDD.sampleByKeyExact(true, fractions, 1)
```

```
scala> println(rowsRDD.countByKey)
Map(married -> 24928, unknown -> 80, single -> 11568, divorced -> 4612)

scala> println(rowsSampleRDD.countByKey)
Map(married -> 12500, unknown -> 11, single -> 2920, divorced -> 702)

scala> println(rowsSampleRDDExact.countByKey)
Map(married -> 12464, unknown -> 8, single -> 2892, divorced -> 692)
```

```
scala> val rowsRandomSampleRDD = rowsRDD.sample(true, .1)

scala> println(rowsRandomSampleRDD.countByKey)
Map(married -> 2524, unknown -> 6, single -> 1198, divorced -> 472)
```

```
scala> val sourceDF = df.select($"job", $"marital", $"edu", $"housing",
$"loan", $"contact", $"month", $"day", $"dur", $"campaign", $"pdays", $"prev",
$"pout", $"deposit")
```

```
scala>
sourceDF.groupBy("marital").pivot("housing").agg(count("housing")).sort("marit
al").show()
+--------+-----+-------+-----+
| marital|   no|unknown|  yes|
+--------+-----+-------+-----+
|divorced| 2092|    121| 2399|
| married|11389|    588|12951|
|  single| 5097|    280| 6191|
| unknown|   44|      1|   35|
+--------+-----+-------+-----+
```

```
scala> sourceDF.groupBy("job").pivot("marital", Seq("unknown", "divorced",
"married", "single")).agg(round(sum("campaign"), 2), round(avg("campaign"),
2)).sort("job").toDF("Job", "U-Tot", "U-Avg", "D-Tot", "D-Avg", "M-Tot", "M-
Avg", "S-Tot", "S-Avg").show()
+-------------+-----+-----+------+-----+-------+-----+-------+-----+
|          Job|U-Tot|U-Avg| D-Tot|D-Avg|  M-Tot|M-Avg|  S-Tot|S-Avg|
+-------------+-----+-----+------+-----+-------+-----+-------+-----+
|       admin.| 24.0| 1.71|3398.0| 2.65|13807.0| 2.63|10113.0| 2.61|
|  blue-collar| 54.0| 3.86|1893.0|  2.6|17051.0| 2.55| 4678.0| 2.56|
| entrepreneur| 11.0| 3.67| 485.0| 2.71| 2664.0| 2.49|  532.0| 2.62|
|    housemaid| 17.0| 5.67| 396.0| 2.46| 2089.0| 2.69|  296.0| 2.49|
|   management| 16.0| 5.33| 866.0| 2.62| 5192.0| 2.49| 1166.0| 2.33|
|      retired| 17.0|  3.4| 865.0| 2.49| 3110.0| 2.44|  268.0| 2.88|
|self-employed| 11.0|  2.2| 341.0| 2.56| 2525.0| 2.79|  904.0| 2.39|
|     services| 33.0|  5.5|1345.0| 2.53| 5903.0| 2.57| 2990.0| 2.63|
|      student|  1.0|  1.0|  19.0| 2.11|  112.0| 2.73| 1709.0| 2.07|
|   technician| 21.0| 1.75|2113.0| 2.73| 9348.0| 2.55| 5897.0| 2.58|
|   unemployed| 11.0|  2.2| 303.0| 2.44| 1701.0| 2.68|  585.0| 2.33|
|      unknown| 39.0| 4.33|  29.0| 2.23|  633.0| 2.71|  173.0| 2.34|
+-------------+-----+-----+------+-----+-------+-----+-------+-----+
```

```
scala> sourceDF.groupBy("job").pivot("marital", Seq("unknown", "divorced",
"married", "single")).agg(round(sum("dur"), 2), round(avg("dur"),
2)).sort("job").toDF("Job", "U-Tot", "U-Avg", "D-Tot", "D-Avg", "M-Tot", "M-
Avg", "S-Tot", "S-Avg").show()
+-------------+------+------+--------+------+---------+------+--------+------+
|          Job| U-Tot| U-Avg|   D-Tot| D-Avg|    M-Tot| M-Avg|   S-Tot| S-Avg|
+-------------+------+------+--------+------+---------+------+--------+------+
|       admin.|4753.0| 339.5|325183.0|254.05|1335988.0|254.33|984517.0|254.07|
|  blue-collar|4860.0|347.14|198377.0| 272.5|1750593.0|261.79|494245.0|270.82|
| entrepreneur|1146.0| 382.0| 41972.0|234.48| 287624.0|268.56| 52576.0| 259.0|
|    housemaid| 355.0|118.33| 40625.0|252.33| 193334.0|248.82| 31168.0|261.92|
|   management| 477.0| 159.0| 82252.0| 248.5| 540274.0|258.63|128635.0|256.76|
|      retired|2247.0| 449.4| 89253.0|256.47| 352288.0|276.52| 26997.0|290.29|
|self-employed|2475.0| 495.0| 38910.0|292.56| 236205.0|261.29| 97756.0|257.93|
|     services|1440.0| 240.0|137672.0|258.78| 576591.0|251.35|309879.0|272.54|
|      student| 155.0| 155.0|  3177.0| 353.0|   9279.0|226.32|235612.0|285.94|
|   technician|3988.0|332.33|179986.0|232.54| 925448.0|252.17|577894.0|252.69|
|   unemployed| 756.0| 151.2| 30969.0|249.75| 161178.0|254.22| 60041.0|239.21|
|      unknown|2366.0|262.89|  2105.0|161.92|  48628.0|207.81| 25994.0|351.27|
+-------------+------+------+--------+------+---------+------+--------+------+
```

```
scala> sourceDF.groupBy("job").pivot("marital", Seq("divorced",
"married")).agg(round(avg("dur"), 2)).sort("job").show()
+-------------+--------+-------+
|          job|divorced|married|
+-------------+--------+-------+
|       admin.|  254.05| 254.33|
|  blue-collar|   272.5| 261.79|
| entrepreneur|  234.48| 268.56|
|    housemaid|  252.33| 248.82|
|   management|   248.5| 258.63|
|      retired|  256.47| 276.52|
|self-employed|  292.56| 261.29|
|     services|  258.78| 251.35|
|      student|   353.0| 226.32|
|   technician|  232.54| 252.17|
|   unemployed|  249.75| 254.22|
|      unknown|  161.92| 207.81|
+-------------+--------+-------+
```

```
scala> sourceDF.groupBy("job", "housing").pivot("marital", Seq("divorced",
"married")).agg(round(avg("dur"), 2)).sort("job").show
+-------------+-------+--------+-------+
|          job|housing|divorced|married|
+-------------+-------+--------+-------+
|       admin.|     no|  251.43| 257.44|
|       admin.|    yes|   259.6| 251.86|
|       admin.|unknown|  193.11| 251.27|
|  blue-collar|     no|  272.92| 268.64|
|  blue-collar|unknown|  211.15|  233.1|
|  blue-collar|    yes|  275.48| 256.89|
| entrepreneur|unknown|   302.2|  189.9|
| entrepreneur|    yes|   271.8| 279.45|
| entrepreneur|     no|  191.42| 259.84|
|    housemaid|    yes|  239.06| 233.45|
|    housemaid|unknown|   166.2|  342.9|
|    housemaid|     no|  275.84| 259.95|
|   management|    yes|  253.36| 256.95|
|   management|     no|  244.96| 260.18|
|   management|unknown|   230.5| 264.98|
|      retired|unknown|  231.25| 238.11|
|      retired|    yes|  250.91| 284.82|
|      retired|     no|  265.21| 270.03|
|self-employed|    yes|  303.29| 264.08|
|self-employed|unknown|  186.67| 228.27|
+-------------+-------+--------+-------+
only showing top 20 rows
```

```
scala> import org.apache.spark.sql._

scala> val saveDF = sourceDF.groupBy("deposit").pivot("month", Seq("jan",
"feb", "mar", "apr", "may", "jun", "jul", "aug", "sep", "oct", "nov",
"dec")).agg(count("deposit")).sort("deposit").na.fill(0)

scala> val writer: DataFrameWriter[Row] = saveDF.write

scala>
writer.format("csv").mode("overwrite").save("file:///Users/aurobindosarkar/Dow
nloads/saveDF")


scala> val dataRDD =
sc.textFile("file:///Users/aurobindosarkar/Downloads/saveDF/*.csv").map(_.spli
t(","))
```

```
scala> val labels = List("deposit", "jan", "feb", "mar", "apr", "may", "jun",
"jul", "aug", "sep", "oct", "nov", "dec")

scala> val labelQ2 = List("apr", "may", "jun")
scala> val labelQ3 = List("jul", "aug", "sep")

scala> val indexQ2 = labelQ2.map(x => labels.indexOf(x))
scala> val indexQ3 = labelQ3.map(x => labels.indexOf(x))


scala> dataRDD.map(x => indexQ2.map(i => x(i).toDouble).sum).collect
res133: Array[Double] = Array(19735.0, 1984.0)

scala> dataRDD.map(x => indexQ3.map(i => x(i).toDouble).sum).collect
res134: Array[Double] = Array(12362.0, 1560.0)
```

Chapter 4:
Using Spark SQL for Data Munging

```
scala> import org.apache.spark.sql.types._
scala> import spark.implicits._
scala> import org.apache.spark.sql.functions.{from_unixtime, unix_timestamp}
scala> import org.apache.spark.sql.functions.udf
scala> import org.apache.spark.sql.{Row, Column, DataFrame}
scala> import scala.collection.mutable.WrappedArray
scala> import com.google.common.collect.ImmutableMap
scala> import org.apache.spark.rdd.RDD
```

```
scala> case class HouseholdEPC(date: String, time: String, gap: Double, grp:
Double, voltage: Double, gi: Double, sm_1: Double, sm_2: Double, sm_3: Double)
```

```
scala> val hhEPCRdd =
sc.textFile("file:///Users/aurobindosarkar/Downloads/household_power_consumpti
on.txt")
```

```
scala> hhEPCRdd.count()
res71: Long = 2075260
```

```
scala> val header = hhEPCRdd.first()

header: String =
Date;Time;Global_active_power;Global_reactive_power;Voltage;Global_intensity;S
ub_metering_1;Sub_metering_2;Sub_metering_3
```

```
scala> val data = hhEPCRdd.filter(row => row != header).filter(rows =>
!rows.contains("?"))
```

```
scala> val hhEPCClassRdd = data.map(_.split(";")).map(p =>
HouseholdEPC(p(0).trim().toString,p(1).trim().toString,p(2).toDouble,p(3).toDo
uble,p(4).toDouble,p(5).toDouble,p(6).toDouble,p(7).toDouble,p(8).toDouble))

scala> val hhEPCDF = hhEPCClassRdd.toDF()
```

```
scala> hhEPCDF.show(5)
+----------+--------+-----+-----+-------+----+----+----+----+
|      date|    time|  gap|  grp|voltage|  gi|sm_1|sm_2|sm_3|
+----------+--------+-----+-----+-------+----+----+----+----+
|16/12/2006|17:24:00|4.216|0.418| 234.84|18.4| 0.0| 1.0|17.0|
|16/12/2006|17:25:00| 5.36|0.436| 233.63|23.0| 0.0| 1.0|16.0|
|16/12/2006|17:26:00|5.374|0.498| 233.29|23.0| 0.0| 2.0|17.0|
|16/12/2006|17:27:00|5.388|0.502| 233.74|23.0| 0.0| 1.0|17.0|
|16/12/2006|17:28:00|3.666|0.528| 235.68|15.8| 0.0| 1.0|17.0|
+----------+--------+-----+-----+-------+----+----+----+----+
only showing top 5 rows

scala> hhEPCDF.count()
res73: Long = 2049280
```

```
scala> hhEPCDF.describe().show()
```

```
scala> hhEPCDF.describe().select($"summary", $"gap", $"grp", $"voltage",
$"gi", $"sm_1", $"sm_2", $"sm_3", round($"gap", 4).name("rgap"), round($"grp",
4).name("rgrp"), round($"voltage", 4).name("rvoltage"), round($"gi",
4).name("rgi"), round($"sm_1", 4).name("rsm_1"), round($"sm_2",
4).name("rsm_2"), round($"sm_3", 4).name("rsm_3")).drop("gap", "grp",
"voltage", "gi", "sm_1", "sm_2", "sm_3").show()
```

```
+-------+---------+---------+---------+---------+---------+---------+---------+
|summary|     rgap|     rgrp| rvoltage|      rgi|    rsm_1|    rsm_2|    rsm_3|
+-------+---------+---------+---------+---------+---------+---------+---------+
|  count|2049280.0|2049280.0|2049280.0|2049280.0|2049280.0|2049280.0|2049280.0|
|   mean|   1.0916|   0.1237| 240.8399|   4.6278|   1.1219|   1.2985|   6.4584|
| stddev|   1.0573|   0.1127|     3.24|   4.4444|    6.153|    5.822|   8.4372|
|    min|    0.076|      0.0|    223.2|      0.2|      0.0|      0.0|      0.0|
|    max|   11.122|     1.39|   254.15|     48.4|     88.0|     80.0|     31.0|
+-------+---------+---------+---------+---------+---------+---------+---------+
```

```
scala> val numDates =
hhEPCDF.groupBy("date").agg(countDistinct("date")).count()
numDates: Long = 1433
```

```
scala> val hhEPCDatesDf = hhEPCDF.withColumn("dow",
from_unixtime(unix_timestamp($"date", "dd/MM/yyyy"),
"EEEEE")).withColumn("day", dayofmonth(to_date(unix_timestamp($"date",
"dd/MM/yyyy").cast("timestamp")))).withColumn("month",
month(to_date(unix_timestamp($"date",
"dd/MM/yyyy").cast("timestamp")))).withColumn("year",
year(to_date(unix_timestamp($"date", "dd/MM/yyyy").cast("timestamp"))))

scala> hhEPCDatesDf.show(5)
+---------+--------+-----+-----+--------+-----+----+----+----+--------+---+-----+----+
|     date|    time|  gap|  grp|voltage|   gi|sm_1|sm_2|sm_3|     dow|day|month|year|
+---------+--------+-----+-----+--------+-----+----+----+----+--------+---+-----+----+
|16/12/2006|17:24:00|4.216|0.418| 234.84|18.4| 0.0| 1.0|17.0|Saturday| 16|   12|2006|
|16/12/2006|17:25:00| 5.36|0.436| 233.63|23.0| 0.0| 1.0|16.0|Saturday| 16|   12|2006|
|16/12/2006|17:26:00|5.374|0.498| 233.29|23.0| 0.0| 2.0|17.0|Saturday| 16|   12|2006|
|16/12/2006|17:27:00|5.388|0.502| 233.74|23.0| 0.0| 1.0|17.0|Saturday| 16|   12|2006|
|16/12/2006|17:28:00|3.666|0.528| 235.68|15.8| 0.0| 1.0|17.0|Saturday| 16|   12|2006|
+---------+--------+-----+-----+--------+-----+----+----+----+--------+---+-----+----+

only showing top 5 rows
```

```
scala> val delTmDF = hhEPCDF.drop("time")


scala> val finalDayDf1 =
delTmDF.groupBy($"date").agg(sum($"gap").name("A"),sum($"grp").name("B"),avg($
"voltage").name("C"),sum($"gi").name("D"), sum($"sm_1").name("E"),
sum($"sm_2").name("F"), sum($"sm_3").name("G")).select($"date", round($"A",
2).name("dgap"), round($"B", 2).name("dgrp"), round($"C", 2).name("dvoltage"),
round($"C", 2).name("dgi"), round($"E", 2).name("dsm_1"), round($"F",
2).name("dsm_2"), round($"G", 2).name("dsm_3")).withColumn("day",
dayofmonth(to_date(unix_timestamp($"date",
"dd/MM/yyyy").cast("timestamp")))).withColumn("month",
month(to_date(unix_timestamp($"date",
"dd/MM/yyyy").cast("timestamp")))).withColumn("year",
year(to_date(unix_timestamp($"date", "dd/MM/yyyy").cast("timestamp"))))
```

```
+----------+-------+------+--------+------+------+------+-------+---+-----+----+
|      date|   dgap|  dgrp|dvoltage|   dgi| dsm_1| dsm_2|  dsm_3|day|month|year|
+----------+-------+------+--------+------+------+------+-------+---+-----+----+
| 30/1/2007| 1707.8|180.93|  241.84|241.84|1123.0|1424.0| 8149.0| 30|    1|2007|
| 13/2/2007|1414.55|134.68|  241.17|241.17|   0.0|2828.0| 6256.0| 13|    2|2007|
|19/11/2007|1723.65|112.03|  242.77|242.77|1089.0| 253.0|10750.0| 19|   11|2007|
| 12/1/2008|2871.41|162.72|  239.96|239.96|5594.0|2621.0|11116.0| 12|    1|2008|
| 26/2/2008| 507.12| 84.59|  239.53|239.53|   0.0| 295.0| 2013.0| 26|    2|2008|
+----------+-------+------+--------+------+------+------+-------+---+-----+----+
only showing top 5 rows
```

```
scala> val readingsByMonthDf = hhEPCDatesDf.groupBy($"year",
$"month").count().orderBy($"year", $"month")

scala> readingsByMonthDf.count()
res77: Long = 48

scala> readingsByMonthDf.show(5)
+----+-----+-----+
|year|month|count|
+----+-----+-----+
|2006|   12|21992|
|2007|    1|44638|
|2007|    2|40318|
|2007|    3|44639|
|2007|    4|39477|
+----+-----+-----+
only showing top 5 rows
```

```
scala> case class HouseholdEPCDTmDay(date: String, day: String, month: String,
year: String, dgap: Double, dgrp: Double, dvoltage: Double, dgi: Double,
dsm_1: Double, dsm_2: Double, dsm_3: Double)

scala> val finalDayDs1 = finalDayDf1.as[HouseholdEPCDTmDay]
```

```
scala> case class DayWeather(CET: String, Max_TemperatureC: Double,
Mean_TemperatureC: Double, Min_TemperatureC: Double, Dew_PointC: Double,
MeanDew_PointC: Double, Min_DewpointC: Double, Max_Humidity: Double,
Mean_Humidity: Double, Min_Humidity: Double, Max_Sea_Level_PressurehPa:
Double, Mean_Sea_Leve_PressurehPa: Double, Min_Sea_Level_PressurehPa: Double,
Max_VisibilityKm: Double, Mean_VisibilityKm: Double, Min_VisibilitykM: Double,
Max_Wind_SpeedKmph: Double, Mean_Wind_SpeedKmph: Double, Max_Gust_SpeedKmph:
Double, Precipitationmm: Double, CloudCover: Double, Events: String,
WindDirDegrees: Double)
```

```
scala> val dwRdd1 =
sc.textFile("file:///Users/aurobindosarkar/Downloads/Paris_Weather/Paris_Weather_Year_1.c
sv")
scala> val dwRdd2 =
sc.textFile("file:///Users/aurobindosarkar/Downloads/Paris_Weather/Paris_Weather_Year_2.c
sv")
scala> val dwRdd3 =
sc.textFile("file:///Users/aurobindosarkar/Downloads/Paris_Weather/Paris_Weather_Year_3.c
sv")
scala> val dwRdd4 =
sc.textFile("file:///Users/aurobindosarkar/Downloads/Paris_Weather/Paris_Weather_Year_4.c
sv")

scala> println("Number 0f Readings - Year 1: " + dwRdd1.count())
Number 0f Readings - Year 1: 366

scala> println("Number 0f Readings - Year 2: " + dwRdd2.count())
Number 0f Readings - Year 2: 367

scala> println("Number 0f Readings - Year 3: " + dwRdd3.count())
Number 0f Readings - Year 3: 366

scala> println("Number 0f Readings - Year 4: " + dwRdd4.count())
Number 0f Readings - Year 4: 377
```

```
scala> val header = dwRdd1.first()

header: String = CET,Max TemperatureC,Mean TemperatureC,Min TemperatureC,Dew
PointC,MeanDew PointC,Min DewpointC,Max Humidity, Mean Humidity, Min Humidity,
Max Sea Level PressurehPa, Mean Sea Level PressurehPa, Min Sea Level
PressurehPa, Max VisibilityKm, Mean VisibilityKm, Min VisibilitykM, Max Wind
SpeedKm/h, Mean Wind SpeedKm/h, Max Gust SpeedKm/h,Precipitationmm,
CloudCover, Events,WindDirDegrees

scala> val data1 = dwRdd1.filter(row => row != header)
scala> val data2 = dwRdd2.filter(row => row != header)
scala> val data3 = dwRdd3.filter(row => row != header)
scala> val data4 = dwRdd4.filter(row => row != header)
```

```
scala> val emptyFieldRowsRDD = data1.map(_.split(",")).filter(!_.contains(""))

scala> emptyFieldRowsRDD.count()
res81: Long = 139
```

```
scala> val csvDF = spark.read.format("csv").option("header",
true).option("inferSchema",
true).load("file:///Users/aurobindosarkar/Downloads/Paris_Weather/Paris_Weathe
r_Year_1.csv")

scala> csvDF.select($"CET", $" Events").show()
+--------------------+-----------------+
|                 CET|           Events|
+--------------------+-----------------+
|2006-11-16 00:00:...|             Rain|
|2006-11-17 00:00:...|             Rain|
|2006-11-18 00:00:...|             Rain|
|2006-11-19 00:00:...|             Rain|
|2006-11-20 00:00:...|             Rain|
|2006-11-21 00:00:...|Rain-Thunderstorm|
|2006-11-22 00:00:...|             Rain|
|2006-11-23 00:00:...|             Rain|
|2006-11-24 00:00:...|             Rain|
|2006-11-25 00:00:...|             Rain|
|2006-11-26 00:00:...|                 |
|2006-11-27 00:00:...|             Rain|
|2006-11-28 00:00:...|             Rain|
|2006-11-29 00:00:...|              Fog|
|2006-11-30 00:00:...|              Fog|
|2006-12-01 00:00:...|                 |
|2006-12-02 00:00:...|             Rain|
|2006-12-03 00:00:...|             Rain|
|2006-12-04 00:00:...|             Rain|
|2006-12-05 00:00:...|             Rain|
+--------------------+-----------------+
only showing top 20 rows
```

```
scala> val dropRowsWithEmptyFieldsDF = csvDF.filter($" Events" =!=
"").filter($" Max Gust SpeedKm/h" =!= "")
scala> dropRowsWithEmptyFieldsDF.count()
res80: Long = 139
```

```
scala> def processRdd(data: RDD[String]): RDD[DayWeather] = { val dwClassRdd =
data.map(_.split(",")).map(c => c.map(f => f match { case x if x.isEmpty() =>
"0"; case x => x })).map(p => DayWeather(p(0).trim().toString, p(1).toDouble,
p(2).toDouble, p(3).toDouble, p(4).toDouble, p(5).toDouble, p(6).toDouble,
p(7).toDouble, p(8).toDouble, p(9).toDouble, p(10).toDouble, p(11).toDouble,
p(12).toDouble, p(13).toDouble, p(14).toDouble, p(15).toDouble,
p(16).toDouble, p(17).toDouble, p(18).toDouble, p(19).toDouble,
p(20).toDouble, p(21).trim().toString, p(22).toDouble)); dwClassRdd; }

scala> val dwClassRdd1 = processRdd(data1)
scala> val dwClassRdd2 = processRdd(data2)
scala> val dwClassRdd3 = processRdd(data3)
scala> val dwClassRdd4 = processRdd(data4)

scala> dwClassRdd1.take(5).foreach(println)
DayWeather(2006-11-
16,17.0,14.0,12.0,13.0,10.0,8.0,100.0,77.0,63.0,1004.0,1002.0,1001.0,10.0,10.0
,4.0,40.0,23.0,60.0,0.0,6.0,Rain,185.0)
DayWeather(2006-11-
17,14.0,11.0,9.0,11.0,8.0,7.0,94.0,83.0,72.0,1009.0,1007.0,1004.0,10.0,10.0,10
.0,32.0,18.0,58.0,0.0,5.0,Rain,202.0)
DayWeather(2006-11-
18,13.0,10.0,8.0,9.0,6.0,3.0,100.0,78.0,54.0,1019.0,1014.0,1007.0,10.0,10.0,10
.0,34.0,18.0,52.0,0.0,6.0,Rain,215.0)
DayWeather(2006-11-
19,10.0,8.0,5.0,9.0,6.0,4.0,100.0,90.0,76.0,1022.0,1018.0,1013.0,10.0,9.0,5.0,
26.0,11.0,0.0,0.0,5.0,Rain,284.0)
DayWeather(2006-11-
20,12.0,8.0,5.0,12.0,7.0,4.0,100.0,92.0,81.0,1021.0,1012.0,1003.0,10.0,9.0,3.0
,34.0,21.0,52.0,0.0,6.0,Rain,196.0)
```

```
scala> val dwDS1 = dwClassRdd1.toDF().na.replace(Seq("CET", "Events"),Map("0" ->
"NA")).as[DayWeather]
scala> val dwDS2 = dwClassRdd2.toDF().na.replace(Seq("CET", "Events"),Map("0" ->
"NA")).as[DayWeather]
scala> val dwDS3 = dwClassRdd3.toDF().na.replace(Seq("CET", "Events"),Map("0" ->
"NA")).as[DayWeather]
scala> val dwDS4 = dwClassRdd4.toDF().na.replace(Seq("CET", "Events"),Map("0" ->
"NA")).as[DayWeather]
```

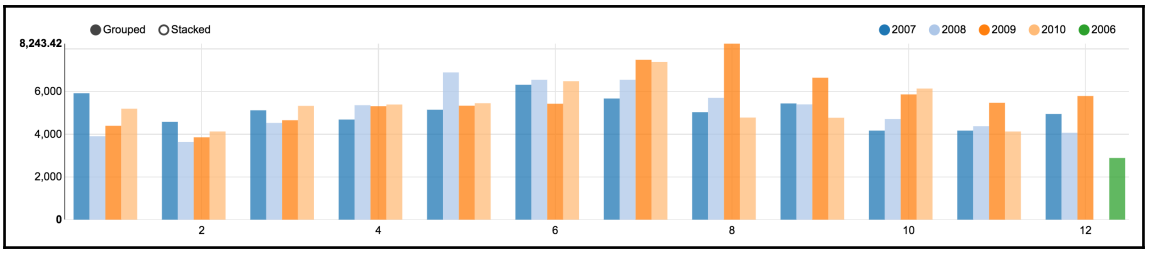
```
scala> val finalDs2 = dwDS1.union(dwDS2).union(dwDS3).union(dwDS4)

scala> finalDs2.count()
res83: Long = 1472
```

```
scala> val joinedDF =
finalDayDs1.join(finalDs2).where(unix_timestamp(finalDayDs1("date"),
"dd/MM/yyyy") === unix_timestamp(finalDs2("CET"), "yyyy-MM-dd"))
```

```
scala> joinedDF.count()
res84: Long = 1433
```

```
scala> val corr = joinedDF.stat.corr("Mean_TemperatureC","dgap")
corr: Double = -0.5407030887973712

scala> println("Mean_TemperatureC to grp : Correlation = %.4f".format(corr))
Mean_TemperatureC to grp : Correlation = -0.5407

scala> val corr = joinedDF.stat.corr("Mean_TemperatureC","dgrp")
corr: Double = 0.47091563055684876

scala> println("Mean_TemperatureC to dgrp : Correlation = %.4f".format(corr))
Mean_TemperatureC to dgrp : Correlation = 0.4709

scala> val corr = joinedDF.stat.corr("Mean_Humidity","dgap")
corr: Double = 0.3420834803263318

scala> val corr = joinedDF.stat.corr("Mean_Humidity","dgrp")
corr: Double = -0.2778028625600493

scala> val corr = joinedDF.stat.corr("Max_TemperatureC","dsm_1")
corr: Double = -0.12588773465430714

scala> val corr = joinedDF.stat.corr("Max_TemperatureC","dsm_2")
corr: Double = -0.12267913849215695

scala> val corr = joinedDF.stat.corr("Max_TemperatureC","dsm_3")
corr: Double = -0.37749055597177456
```

```
scala> val joinedMonthlyDF = joinedDF.groupBy("year",
"month").agg(sum($"dgap").name("A"),sum($"dgrp").name("B"),avg($"dvoltage").na
me("C"),sum($"dgi").name("D"), sum($"dsm_1").name("E"),
sum($"dsm_2").name("F"), sum($"dsm_3").name("G")).select($"year", $"month",
round($"A", 2).name("mgap"), round($"B", 2).name("mgrp"), round($"C",
2).name("mvoltage"), round($"C", 2).name("mgi"), round($"E", 2).name("msm_1"),
round($"F", 2).name("msm_2"), round($"G", 2).name("msm_3")).orderBy("year",
"month")
```




```
scala> val joinedDayDF =
finalDayDs1.join(finalDs2).where(unix_timestamp(finalDayDs1("date"),
"dd/MM/yyyy") === unix_timestamp(finalDs2("CET"), "yyyy-MM-dd"))

scala> joinedDayDF.count()
res87: Long = 1433

scala> val joinedDayDowDF = joinedDayDF.withColumn("dow",
from_unixtime(unix_timestamp($"date", "dd/MM/yyyy"), "EEEEE"))
```

```
scala> joinedDayDowDF.printSchema()
root
 |-- date: string (nullable = true)
 |-- dgap: double (nullable = true)
 |-- dgrp: double (nullable = true)
 |-- dvoltage: double (nullable = true)
 |-- dgi: double (nullable = true)
 |-- dsm_1: double (nullable = true)
 |-- dsm_2: double (nullable = true)
 |-- dsm_3: double (nullable = true)
 |-- day: integer (nullable = true)
 |-- month: integer (nullable = true)
 |-- year: integer (nullable = true)
 |-- CET: string (nullable = true)
 |-- Max_TemperatureC: double (nullable = false)
 |-- Mean_TemperatureC: double (nullable = false)
 |-- Min_TemperatureC: double (nullable = false)
 |-- Dew_PointC: double (nullable = false)
 |-- MeanDew_PointC: double (nullable = false)
 |-- Min_DewpointC: double (nullable = false)
 |-- Max_Humidity: double (nullable = false)
 |-- Mean_Humidity: double (nullable = false)
 |-- Min_Humidity: double (nullable = false)
 |-- Max_Sea_Level_PressurehPa: double (nullable = false)
 |-- Mean_Sea_Leve_PressurehPa: double (nullable = false)
 |-- Min_Sea_Level_PressurehPa: double (nullable = false)
 |-- Max_VisibilityKm: double (nullable = false)
 |-- Mean_VisibilityKm: double (nullable = false)
 |-- Min_VisibilitykM: double (nullable = false)
 |-- Max_Wind_SpeedKmph: double (nullable = false)
 |-- Mean_Wind_SpeedKmph: double (nullable = false)
 |-- Max_Gust_SpeedKmph: double (nullable = false)
 |-- Precipitationmm: double (nullable = false)
 |-- CloudCover: double (nullable = false)
 |-- Events: string (nullable = true)
 |-- WindDirDegrees: double (nullable = false)
 |-- dow: string (nullable = true)
```

```
scala> import org.apache.spark.sql.types.{StructType, StructField, StringType}
scala> import org.apache.spark.sql.DataFrame
scala> import spark.implicits._
```

```
scala> val path = "hdfs://localhost:9000/20_newsgroups/comp.graphics"
scala> val data = spark.sparkContext.wholeTextFiles(path)
scala> var output : org.apache.spark.rdd.RDD[(String, String, Int)] =
sc.emptyRDD
scala> val files = data.map { case (filename, content) => filename}
```

```
scala> def Process(filename: String):
org.apache.spark.rdd.RDD[(String, String, Int)]= {
     |     println("Processing:" + filename)
     |     val fpath = filename.split('/').last;
     |     val lines = spark.sparkContext.textFile(filename);
     |     val counts = lines.flatMap(line => line.split("
")).map(word => word).map(word => (word, 1)).reduceByKey(_ + _);
     |     val word_counts = counts.map( x => (fpath,x._1, x._2));
     |     word_counts
     | }

scala> val buf =
scala.collection.mutable.ArrayBuffer.empty[org.apache.spark.rdd.RDD[(
String, String, Int)]]

scala> files.collect.foreach( filename => { buf += Process(filename);
})
```

```
Processing:hdfs://localhost:9000/20_newsgroups/comp.graphics/37261
Processing:hdfs://localhost:9000/20_newsgroups/comp.graphics/37913
Processing:hdfs://localhost:9000/20_newsgroups/comp.graphics/37914
                                    .
                                    .
                                    .
Processing:hdfs://localhost:9000/20_newsgroups/comp.graphics/40008
Processing:hdfs://localhost:9000/20_newsgroups/comp.graphics/40027
Processing:hdfs://localhost:9000/20_newsgroups/comp.graphics/40062
```

```
scala> output = spark.sparkContext.union(buf.toList);
```

```
files.collect.foreach( filename => { output = output.union(Process(filename));
})
```

```
scala> output.cache()

scala> output.take(5).foreach(println)
(37261,sent,1)
(37261,sponsoring,1)
(37261,deadline,1)
(37261,seminar),1)
(37261,Short,1)
```

```
scala> val stopWords = sc.broadcast(Set("as", "able", "about",
"above", "according", "accordingly", "across", "actually", "..."))

scala> def processLine(s: String, stopWords: Set[String]):
List[String] = {
     |     s.toLowerCase()
     |       .split("\\s+")
     |       .filter(x => x.matches("[A-Za-z]+"))
     |       .filter(!stopWords.contains(_))
     |       .toList
     | }

scala> val groupedRDD = output.map{ case (x, y, z) => (x,
(processLine(y.trim(), stopWords.value)).mkString, z)}.filter{case
(x, y, z) => !y.equals("")}

scala> groupedRDD.take(20).foreach(println)
(37261,sent,1)
(37261,sponsoring,1)
(37261,deadline,1)
(37261,short,1)
(37261,materials,1)
(37261,june,2)
(37261,include,1)
(37261,reality,2)
(37261,organizations,1)
(37261,type,1)
(37261,reality,1)
(37261,voice,1)
(37261,visualization,3)
(37261,naval,2)
(37261,please,1)
(37261,visualization,1)
(37261,viewgraphs,1)
(37261,solicited,1)
(37261,proposed,1)
(37261,purpose,1)
```

```
scala> val words = groupedRDD.map{ case (x, y, z) => y}
scala> val wordsDF = words.toDF
```

```
scala> val stopwords =
sc.textFile("file:///Users/aurobindosarkar/Downloads/sparkworks/Spark
Book/data/stopwords.txt")
```

```
scala> val regex = "[,.:;'\"\\?\\-!\\(\\)]".r

scala> val stopwordsDF = stopwords.flatMap(line =>
line.split("[\\s]")).map(word =>
regex.replaceAllIn(word.trim.toLowerCase, "")).filter(word =>
!word.isEmpty).toDF()

scala> stopwordsDF.count()
res4: Long = 544
```

```
scala> words.count()
res5: Long = 88268

scala> val cleanwords = wordsDF.except(stopwordsDF)

scala> cleanwords.count()
res7: Long = 10245
```

```
bin/spark-shell --jars /Users/aurobindosarkar/Downloads/spark-timeseries-
master/target/sparkts-0.4.0-SNAPSHOT-jar-with-dependencies.jar
```

```
scala> import java.sql.Timestamp
scala> import java.time.{LocalDateTime, ZoneId, ZonedDateTime}
scala> import com.cloudera.sparkts._
scala> import com.cloudera.sparkts.stats.TimeSeriesStatisticalTests
scala> import org.apache.spark.{SparkContext, SparkConf}
scala> import org.apache.spark.sql.{DataFrame, Row, SQLContext}
scala> import org.apache.spark.sql.types._
```

```
scala> val amznRDD =
sc.textFile("file:///Users/aurobindosarkar/Downloads/yahoo/tableAMZN.csv")
scala> val orclRDD =
sc.textFile("file:///Users/aurobindosarkar/Downloads/yahoo/tableORCL.csv")
scala> val ibmRDD =
sc.textFile("file:///Users/aurobindosarkar/Downloads/yahoo/tableIBM.csv")
scala> val cscoRDD =
sc.textFile("file:///Users/aurobindosarkar/Downloads/yahoo/tableCSCO.csv")
scala> val googRDD =
sc.textFile("file:///Users/aurobindosarkar/Downloads/yahoo/tableGOOG.csv")
scala> val msftRDD =
sc.textFile("file:///Users/aurobindosarkar/Downloads/yahoo/tableMSFT.csv")

scala> case class Stock(ticker: String, datestr: String, open: Double, high:
Double, low: Double, close: Double, volume: Int, adjclose: Double)
```

```
scala> val header = amznRDD.first()

scala> val amznData = amznRDD.filter(row => row !=
header).map(_.split(",")).map(p => Stock("AMZN", p(0).trim().toString,
p(1).toDouble, p(2).toDouble, p(3).toDouble, p(4).toDouble, p(5).toInt,
p(6).toDouble)).toDF.as[Stock]

scala> val orclData = orclRDD.filter(row => row !=
header).map(_.split(",")).map(p => Stock("ORCL", p(0).trim().toString,
p(1).toDouble, p(2).toDouble, p(3).toDouble, p(4).toDouble, p(5).toInt,
p(6).toDouble)).toDF.as[Stock]

scala> val ibmData = ibmRDD.filter(row => row !=
header).map(_.split(",")).map(p => Stock("IBM", p(0).trim().toString,
p(1).toDouble, p(2).toDouble, p(3).toDouble, p(4).toDouble, p(5).toInt,
p(6).toDouble)).toDF.as[Stock]

scala> val cscoData = cscoRDD.filter(row => row !=
header).map(_.split(",")).map(p => Stock("CSCO", p(0).trim().toString,
p(1).toDouble, p(2).toDouble, p(3).toDouble, p(4).toDouble, p(5).toInt,
p(6).toDouble)).toDF.as[Stock]

scala> val googData = googRDD.filter(row => row !=
header).map(_.split(",")).map(p => Stock("GOOG", p(0).trim().toString,
p(1).toDouble, p(2).toDouble, p(3).toDouble, p(4).toDouble, p(5).toInt,
p(6).toDouble)).toDF.as[Stock]

scala> val msftData = msftRDD.filter(row => row !=
header).map(_.split(",")).map(p => Stock("MSFT", p(0).trim().toString,
p(1).toDouble, p(2).toDouble, p(3).toDouble, p(4).toDouble, p(5).toInt,
p(6).toDouble)).toDF.as[Stock]
```

```
scala> val allData =
amznData.union(orclData).union(ibmData).union(cscoData).union(googData).union(msftData)
```

```
scala> val allWithDMY = allData.withColumn("day",
dayofmonth(to_date(unix_timestamp($"datestr", "yyyy-MM-
dd").cast("timestamp")))).withColumn("month",
month(to_date(unix_timestamp($"datestr", "yyyy-MM-
dd").cast("timestamp"))))).withColumn("year",
year(to_date(unix_timestamp($"datestr", "yyyy-MM-dd").cast("timestamp"))))
```

```
scala>
allWithDMY.write.mode("overwrite").csv("file:///Users/aurobindosarkar/Downloads/dtDF")
```

```
scala> def loadObservations(sqlContext: SQLContext, path: String): DataFrame =
{
     |     val rowRdd = sqlContext.sparkContext.textFile(path).map { line =>
     |     val tokens = line.split(',');
     |     val dt = ZonedDateTime.of(tokens(10).toInt, tokens(9).toInt,
tokens(8).toInt, 0, 0, 0, 0, ZoneId.systemDefault());
     |     val ticker = tokens(0).toString;
     |     val open = tokens(2).toDouble;
     |     val high = tokens(3).toDouble;
     |     val low = tokens(4).toDouble;
     |     val close = tokens(5).toDouble;
     |     val volume = tokens(6).toInt;
     |     val adjclose = tokens(7).toDouble;
     |     Row(Timestamp.from(dt.toInstant), ticker, open, high, low, close,
volume, adjclose);
     |    }
     |    val fields = Seq(StructField("timestamp", TimestampType, true),
StructField("ticker", StringType, true), StructField("open", DoubleType,
true), StructField("high", DoubleType, true), StructField("low", DoubleType,
true), StructField("close", DoubleType, true), StructField("volume",
IntegerType, true), StructField("adjclose", DoubleType, true));
     |    val schema = StructType(fields);
     |    sqlContext.createDataFrame(rowRdd, schema);
     | }
```

```
scala> val tickerObs = loadObservations(spark.sqlContext,
"file:///Users/aurobindosarkar/Downloads/dtDF")

scala> tickerObs.show(5)
+--------------------+------+----------+----------+----------+----------+-------+----------+
|           timestamp|ticker|      open|      high|       low|     close| volume|  adjclose|
+--------------------+------+----------+----------+----------+----------+-------+----------+
|2016-12-02 00:00:...|  AMZN|743.400024| 748.48999|736.700012|740.340027|3499200|740.340027|
|2016-12-01 00:00:...|  AMZN|752.409973|753.369995|738.030029|743.650024|4626500|743.650024|
|2016-11-30 00:00:...|  AMZN|     762.0|768.090027|    750.25|750.570007|4580100|750.570007|
|2016-11-29 00:00:...|  AMZN|     768.0|769.890015|761.320007| 762.52002|3266500| 762.52002|
|2016-11-28 00:00:...|  AMZN| 776.98999|     777.0| 764.23999| 766.77002|4380900| 766.77002|
+--------------------+------+----------+----------+----------+----------+-------+----------+
only showing top 5 rows
```

```
scala> val zone = ZoneId.systemDefault()
zone: java.time.ZoneId = Asia/Kolkata

scala> val dtIndex =
DateTimeIndex.uniformFromInterval(ZonedDateTime.of(LocalDateTime.parse("2015-
12-04T00:00:00"), zone), ZonedDateTime.of(LocalDateTime.parse("2016-12-
04T00:00:00"), zone), new BusinessDayFrequency(1))
```

```
scala> tickerTsrdd.cache()

scala> println(tickerTsrdd.count())
6
```

```
scala> tickerTsrdd.take(2).foreach(println)

(GOOG,[766.809998,763.25,762.369995,751.609985,749.460022,738.869995,747.77002,743.400024,758.090027,749
.429993,739.309998,747.77002,750.0,750.309998,748.400024,NaN,762.51001,776.599976,771.0,758.880005,NaN,7
41.840027,742.580017,743.619995,726.390015,714.469971,716.030029,726.070007,700.559998,714.719971,694.45
0012,NaN,701.789978,698.450012,706.590027,725.25,711.669983,713.039978,699.98999,730.960022,742.950012,7
52.0,764.650024,726.950012,708.01001,683.570007,682.73999,678.109985,684.119995,683.109985,682.400024,Na
N,691.0,708.400024,697.349976,700.909973,706.460022,695.849976,699.559998,705.75,705.070007,697.77002,71
8.809998,718.849976,712.419983,710.890015,695.159973,693.969971,705.23999,712.820007,726.820007,730.4899
9,728.330017,736.090027,737.780029,737.599976,742.090027,740.75,738.059998,735.299988,NaN,733.530029,744
.77002,750.530029,744.950012,749.909973,745.289978,737.799988,745.690002,740.280029,739.150024,736.09997
6,743.090027,751.719971,753.200012,759.0,766.609985,753.929993,752.669983,759.140015,718.77002,723.15002
4,708.140015,705.840027,691.02002,693.01001,698.210022,692.359985,695.700012,701.429993,711.119995,712.9
00024,723.179993,715.289978,713.309998,710.830017,716.48999,706.22998,706.630005,700.320007,709.73999,70
4.23999,720.090027,725.27002,724.119995,732.659973,NaN,735.719971,734.150024,730.400024,722.340027,716.5
49988,716.650024,728.280029,728.580017,719.409973,718.359985,718.27002,718.919983,710.359985,691.719971,
693.710022,695.940002,697.460022,701.869995,675.219971,668.26001,680.039978,684.109985,692.099976,699.21
0022,NaN,694.950012,697.77002,695.359985,705.630005,715.090027,720.640015,716.97998,720.950012,719.84997
6,733.780029,736.960022,741.190002,738.630005,742.73999,739.77002,738.419983,741.77002,745.909973,768.78
9978,772.880005,771.070007,773.179993,771.609985,782.219971,781.76001,784.26001,784.679993,784.849976,78
3.219971,782.440002,777.140015,779.909973,777.5,775.419983,772.150024,772.080017,769.640015,769.409973,7
69.539978,772.150024,769.090027,767.049988,768.780029,771.460022,NaN,780.080017,780.349976,775.320007,75
9.659973,769.02002,759.690002,762.48999,771.76001,768.880005,765.700012,771.409973,776.219971,787.210022
,786.900024,774.210022,783.01001,781.559998,775.01001,777.289978,772.559998,776.429993,776.469971,776.85
9985,775.080017,785.940002,783.070007,786.140015,778.190002,778.530029,779.960022,795.26001,801.5,796.96
9971,799.369995,813.109985,807.669983,799.070007,795.349976,795.369995,784.539978,783.609985,768.700012,
762.130005,762.02002,782.52002,790.51001,785.309998,762.559998,754.02002,736.080017,758.48999,764.47998,
771.22998,760.539978,769.200012,768.27002,760.98999,NaN,761.679993,768.23999,770.840027,758.039978,747.9
19983,750.5])

(IBM,[140.429993,139.550003,138.050003,136.610001,136.779999,134.570007,135.929993,137.789993,139.289993
,136.75,134.899994,135.5,137.929993,138.539993,138.25,NaN,137.610001,139.779999,139.339996,137.619995,Na
N,135.949997,135.850006,135.169998,132.860001,131.630005,133.229996,132.899994,131.169998,132.910004,130
.029999,NaN,128.110001,121.860001,122.910004,122.5,122.080002,122.589996,120.959999,122.220001,124.79000
1,124.830002,122.940002,124.720001,127.650002,128.570007,126.980003,124.07,120.190002,117.849998,121.040
001,NaN,122.739998,126.099998,132.449997,133.080002,133.770004,132.399994,132.800003,134.5,132.029999,13
1.029999,134.369995,136.300003,137.800003,137.800003,140.149994,139.070007,140.410004,140.190002,142.360
001,142.779999,142.960007,144.789993,147.039993,147.089996,148.630005,148.100006,145.399994,147.949997,N
aN,148.399994,149.330002,148.410004,151.449997,152.520004,152.070007,150.0,150.020004,148.25,149.350006,
149.25,149.630005,151.229996,151.160004,151.720001,152.529999,144.0,146.110001,149.300003,148.5,148.8099
98,149.080002,150.470001,147.070007,145.940002,145.270004,144.130005,144.25,146.470001,147.289993,147.33
9996,149.970001,148.949997,148.839996,147.720001,149.460007,148.0,147.339996,144.929993,147.25,146.77000
4,148.309998,151.690002,152.440002,152.839996,NaN,153.740005,152.509995,153.5,152.889999,152.729996,153.
330002,154.0,153.419998,152.369995,151.279999,151.059998,150.679993,151.059998,151.990005,153.610001,154
.050003,152.919998,155.350006,146.589996,143.5,145.699997,148.460007,151.779999,152.350006,NaN,151.67999
3,152.369995,152.600006,154.460007,155.330002,157.039993,158.020004,160.279999,159.779999,159.860001,159
.580002,161.360001,160.449997,162.070007,162.649994,162.119995,161.830002,161.369995,160.619995,161.4499
97,160.580002,160.669998,161.550003,163.5,162.039993,161.770004,162.080002,163.529999,161.949997,161.880
005,160.699997,160.440002,161.360001,160.039993,160.0,160.259995,159.050003,158.630005,158.320007,159.72
0001,159.399994,158.880005,159.539993,159.550003,NaN,160.350006,161.639999,159.0,155.690002,158.289993,1
55.809998,154.050003,155.660004,153.839996,154.869995,154.449997,155.529999,156.110001,154.979996,153.97
9996,156.770004,158.289993,158.110001,158.850006,157.610001,156.460007,157.080002,156.880005,155.669998,
157.020004,154.789993,154.289993,153.720001,154.449997,154.770004,150.720001,151.259995,151.520004,149.6
30005,150.570007,150.880005,151.809998,153.350006,152.610001,153.690002,152.789993,151.949997,152.369995
,152.429993,155.720001,155.169998,154.809998,160.220001,161.270004,158.210007,158.669998,159.289993,159.
800003,160.389999,162.770004,162.669998,161.979996,NaN,163.139999,164.520004,163.529999,162.220001,159.8
20007,160.020004])
```

```
scala> val filled = tickerTsrdd.fill("linear")
```

```
scala> val stats = filled.seriesStats()

scala> stats.foreach(println)

(count: 261, mean: 28.518161, stdev: 2.328072, max: 31.870001, min: 22.510000)
(count: 261, mean: 691.321877, stdev: 90.598730, max: 844.359985, min:
482.070007)
(count: 261, mean: 740.845729, stdev: 32.360211, max: 813.109985, min:
668.260010)
(count: 261, mean: 39.054598, stdev: 1.973492, max: 41.770000, min: 33.939999)
(count: 261, mean: 148.243046, stdev: 11.410119, max: 164.520004, min:
117.849998)
(count: 261, mean: 54.719253, stdev: 3.231158, max: 61.119999, min: 48.430000)
```

```
scala> import org.apache.spark.sql.types._

scala> import spark.implicits._

scala> import org.apache.spark.sql.functions.{from_unixtime, unix_timestamp}

scala> import org.apache.spark.sql.functions.udf

scala> import org.apache.spark.sql.{Row, Column, DataFrame}

scala> import scala.collection.mutable.WrappedArray

scala> import org.apache.spark.rdd.RDD

scala> val inputRDD =
sc.textFile("file:///Users/aurobindosarkar/Downloads/ZZAlpha1/combined.txt")
```

```
scala> inputRDD.count()

res0: Long = 481261

scala> inputRDD.take(5).foreach(println)
Jan 03 2012_006 Big_100_1_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, Avg of 1 = 0.888
Jan 03 2012_006 Big_100_1_LONG_SHORT_F.pdf, S, LLY 0.956 =40.11/41.95, Avg of 1 = 0.956
Jan 03 2012_006 Big_100_2_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, SU 1.068
=31.78/29.77, Avg of 2 = 0.978
Jan 03 2012_006 Big_100_2_LONG_SHORT_F.pdf, S, LLY 0.956 =40.11/41.95, MO 1.005
=28.79/28.65, Avg of 2 = 0.981
Jan 03 2012_006 Big_100_5_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, MON 1.107
=78.97/71.32, OXY 1.021 =98.73/96.73, RY 1.001 =51.84/51.77, SU 1.068 =31.78/29.77, Avg
of 5 = 1.017
```

```
scala> def countSubstring( str:String, substr:String ) =
substr.r.findAllMatchIn(str).length

scala> def nSubs(substr: String) = udf((x: String) => countSubstring(x,
substr))

scala> val nCommas = inputRDD.toDF().withColumn("commas",
nSubs(",")($"value"))

scala> nCommas.describe().select($"summary", $"commas").where(($"summary" ===
"count") || ($"summary" === "max")).show()
+-------+------+
|summary|commas|
+-------+------+
|  count|481261|
|    max|    22|
+-------+------+
```

```
scala> nCommas.take(5).foreach(println)
[Jan 03 2012_006 Big_100_1_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, Avg of 1 =
0.888,3]
[Jan 03 2012_006 Big_100_1_LONG_SHORT_F.pdf, S, LLY 0.956 =40.11/41.95, Avg of 1 =
0.956,3]
[Jan 03 2012_006 Big_100_2_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, SU 1.068
=31.78/29.77, Avg of 2 = 0.978,4]
[Jan 03 2012_006 Big_100_2_LONG_SHORT_F.pdf, S, LLY 0.956 =40.11/41.95, MO 1.005
=28.79/28.65, Avg of 2 = 0.981,4]
[Jan 03 2012_006 Big_100_5_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, MON 1.107
=78.97/71.32, OXY 1.021 =98.73/96.73, RY 1.001 =51.84/51.77, SU 1.068 =31.78/29.77, Avg
of 5 = 1.017,7]
```

```
scala> def insertSubstring( str:String, substr:String, times: Int ): String =
{ val builder = StringBuilder.newBuilder; builder.append(str.substring(0,
str.lastIndexOf(substr)+1));builder.append(substr * (22-
times));builder.append(str.substring(str.lastIndexOf(substr)+1));
builder.toString;}

scala> def nInserts(substr: String) = udf((x: String, times: Int) =>
insertSubstring(x, substr, times))
```

```
scala> val fixedLengthDf = nCommas.withColumn("record",
nInserts(",")($"value", $"commas")).drop("value", "commas")
```

```
scala> fixedLengthDf.take(5).foreach(println)
[Jan 03 2012_006 Big_100_1_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67,,,,,,,,,,,,,,,,,,
Avg of 1 = 0.888]
[Jan 03 2012_006 Big_100_1_LONG_SHORT_F.pdf, S, LLY 0.956
=40.11/41.95,,,,,,,,,,,,,,,,,,, Avg of 1 = 0.956]
[Jan 03 2012_006 Big_100_2_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, SU 1.068
=31.78/29.77,,,,,,,,,,,,,,,,,, Avg of 2 = 0.978]
[Jan 03 2012_006 Big_100_2_LONG_SHORT_F.pdf, S, LLY 0.956 =40.11/41.95, MO 1.005
=28.79/28.65,,,,,,,,,,,,,,,,,, Avg of 2 = 0.981]
[Jan 03 2012_006 Big_100_5_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, MON 1.107
=78.97/71.32, OXY 1.021 =98.73/96.73, RY 1.001 =51.84/51.77, SU 1.068
=31.78/29.77,,,,,,,,,,,,,,,,, Avg of 5 = 1.017]
```

```
scala> fixedLengthDf.count()
res5: Long = 481261

scala> val dupRemovedDf = fixedLengthDf.dropDuplicates()

scala> dupRemovedDf.count()
res6: Long = 478057
```

```
scala> case class Portfolio(datestr: String, ls: String, stock1: String,
stock2: String, stock3: String, stock4: String, stock5: String, stock6:
String, stock7: String, stock8: String, stock9: String, stock10: String,
stock11: String, stock12: String, stock13: String, stock14: String, stock15:
String, stock16: String, stock17: String, stock18: String, stock19: String,
stock20: String, avgstr: String )
```

```
scala> val rowsRdd = dupRemovedDf.rdd.map{ row: Row =>
row.getString(0).split(",") }

scala> val dfFixed = rowsRdd.map(s => Portfolio(s(0), s(1), s(2), s(3), s(4),
s(5), s(6), s(7), s(8), s(9), s(10), s(11), s(12), s(13), s(14), s(15), s(16),
s(17), s(18), s(19), s(20), s(21), s(22))).toDF()

scala> dfFixed.count()
res7: Long = 478057
```

```
scala> dfFixed.select("datestr", "ls", "stock1", "stock2", "avgstr").show(5)
+--------------------+---+--------------------+--------------------+-----------------+
|             datestr| ls|              stock1|              stock2|           avgstr|
+--------------------+---+--------------------+--------------------+-----------------+
|Jan 03 2012_006 B...|  L| DB 0.888 =35.23/...|                    | Avg of 1 = 0.888|
|Jan 03 2012_006 B...|  S| LLY 0.956 =40.11...|                    | Avg of 1 = 0.956|
|Jan 03 2012_006 B...|  L| DB 0.888 =35.23/...| SU 1.068 =31.78/...| Avg of 2 = 0.978|
|Jan 03 2012_006 B...|  S| LLY 0.956 =40.11...| MO 1.005 =28.79/...| Avg of 2 = 0.981|
|Jan 03 2012_006 B...|  L| DB 0.888 =35.23/...| MON 1.107 =78.97...| Avg of 5 = 1.017|
+--------------------+---+--------------------+--------------------+-----------------+
only showing top 5 rows
```

```
scala> val df2 = dfFixed.na.replace("stock2",ImmutableMap.of("",
"NA")).select("datestr", "ls", "stock1", "stock2", "avgstr")

scala> df2.show(5)
+--------------------+---+--------------------+--------------------+-----------------+
|             datestr| ls|              stock1|              stock2|           avgstr|
+--------------------+---+--------------------+--------------------+-----------------+
|Jan 03 2012_006 B...|  L| DB 0.888 =35.23/...|                  NA| Avg of 1 = 0.888|
|Jan 03 2012_006 B...|  S| LLY 0.956 =40.11...|                  NA| Avg of 1 = 0.956|
|Jan 03 2012_006 B...|  L| DB 0.888 =35.23/...| SU 1.068 =31.78/...| Avg of 2 = 0.978|
|Jan 03 2012_006 B...|  S| LLY 0.956 =40.11...| MO 1.005 =28.79/...| Avg of 2 = 0.981|
|Jan 03 2012_006 B...|  L| DB 0.888 =35.23/...| MON 1.107 =78.97...| Avg of 5 = 1.017|
+--------------------+---+--------------------+--------------------+-----------------+
only showing top 5 rows
```

```
scala> val df3 = dfFixed.select("datestr", "ls", "stock1", "avgstr")

scala> df3.take(5).foreach(println)
[Jan 03 2012_006 Big_100_1_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, Avg of 1 = 0.888]
[Jan 03 2012_006 Big_100_1_LONG_SHORT_F.pdf, S, LLY 0.956 =40.11/41.95, Avg of 1 = 0.956]
[Jan 03 2012_006 Big_100_2_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, Avg of 2 = 0.978]
[Jan 03 2012_006 Big_100_2_LONG_SHORT_F.pdf, S, LLY 0.956 =40.11/41.95, Avg of 2 = 0.981]
[Jan 03 2012_006 Big_100_5_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, Avg of 5 = 1.017]
```

```
scala> def replaceFirstSubstring( str:String, substr:String, repl: String):
String = { val result = str.replaceFirst(substr, repl); result;}

scala> def nRepls(substr: String, repl: String) = udf((x: String) =>
replaceFirstSubstring(x, substr, repl))

scala> val df4 = df3.withColumn("cleanDatestr", nRepls("_", "
")($"datestr")).drop("datestr").withColumnRenamed("cleanDatestr",
"datestr").select("datestr", "ls", "stock1", "avgstr")

scala> df4.take(5).foreach(println)
[Jan 03 2012 006 Big_100_1_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, Avg of 1 = 0.888]
[Jan 03 2012 006 Big_100_1_LONG_SHORT_F.pdf, S, LLY 0.956 =40.11/41.95, Avg of 1 = 0.956]
[Jan 03 2012 006 Big_100_2_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, Avg of 2 = 0.978]
[Jan 03 2012 006 Big_100_2_LONG_SHORT_F.pdf, S, LLY 0.956 =40.11/41.95, Avg of 2 = 0.981]
[Jan 03 2012 006 Big_100_5_LONG_SHORT_F.pdf, L, DB 0.888 =35.23/39.67, Avg of 5 = 1.017]
```

```
scala> val df4A = df4.withColumn("temp1", nRepls("=",
"")($"stock1")).withColumn("temp", nRepls("/", " ")($"temp1")).drop("temp1",
"stock1")

scala> df4A.take(5).foreach(println)
[Jan 03 2012 006 Big_100_1_LONG_SHORT_F.pdf, L, Avg of 1 = 0.888, DB 0.888  35.23 39.67]
[Jan 03 2012 006 Big_100_1_LONG_SHORT_F.pdf, S, Avg of 1 = 0.956, LLY 0.956  40.11 41.95]
[Jan 03 2012 006 Big_100_2_LONG_SHORT_F.pdf, L, Avg of 2 = 0.978, DB 0.888  35.23 39.67]
[Jan 03 2012 006 Big_100_2_LONG_SHORT_F.pdf, S, Avg of 2 = 0.981, LLY 0.956  40.11 41.95]
[Jan 03 2012 006 Big_100_5_LONG_SHORT_F.pdf, L, Avg of 5 = 1.017, DB 0.888  35.23 39.67]
```

```
scala> def splitString( str:String, sep:String): Array[String] = { val result
= str.split(sep); result;}

scala> def splitStr(sep: String) = udf((x: String) => splitString(x.trim(),
sep))

scala> val df5 = df4A.withColumn("temp1", splitStr("
")($"datestr")).withColumn("temp2", splitStr("
")($"temp")).withColumn("avgarray", splitStr(" ")($"avgstr")).drop("datestr",
"temp").select("temp1", "ls", "temp2", "avgarray")
```

```
scala> df5.show(5)
+--------------------+---+--------------------+--------------------+
|               temp1| ls|               temp2|            avgarray|
+--------------------+---+--------------------+--------------------+
|[Jan, 03, 2012, 0...|  L|[DB, 0.888, 35.23...|[Avg, of, 1, =, 0...|
|[Jan, 03, 2012, 0...|  S|[LLY, 0.956, 40.1...|[Avg, of, 1, =, 0...|
|[Jan, 03, 2012, 0...|  L|[DB, 0.888, 35.23...|[Avg, of, 2, =, 0...|
|[Jan, 03, 2012, 0...|  S|[LLY, 0.956, 40.1...|[Avg, of, 2, =, 0...|
|[Jan, 03, 2012, 0...|  L|[DB, 0.888, 35.23...|[Avg, of, 5, =, 1...|
+--------------------+---+--------------------+--------------------+
only showing top 5 rows
```

```
scala> def retArrayIndex( str:Array[String], index:Int): String = { val result
= str(index); result;}

scala> def retArrayVal(index: Int) = udf((x: WrappedArray[String]) =>
retArrayIndex(x.toArray[String], index))

scala> val df6 = df5.withColumn("month",
retArrayVal(0)($"temp1")).withColumn("dom",
retArrayVal(1)($"temp1")).withColumn("year",
retArrayVal(2)($"temp1")).withColumn("y4",
retArrayVal(3)($"temp1")).withColumn("file",
retArrayVal(4)($"temp1")).withColumn("ticker",
retArrayVal(0)($"temp2")).withColumn("S/P Ratio",
retArrayVal(1)($"temp2")).withColumn("SP",
retArrayVal(2)($"temp2")).withColumn("PP",
retArrayVal(3)($"temp2")).withColumn("nStocks",
retArrayVal(2)($"avgarray")).drop("temp1", "temp2", "avgarray")

scala> df6.show(5)
+---+-----+---+----+---+--------------------+------+---------+-----+-----+-------+
| ls|month|dom|year| y4|                file|ticker|S/P Ratio|   SP|   PP|nStocks|
+---+-----+---+----+---+--------------------+------+---------+-----+-----+-------+
|  L|  Jan| 03|2012|006|Big_100_1_LONG_SH...|    DB|    0.888|35.23|39.67|      1|
|  S|  Jan| 03|2012|006|Big_100_1_LONG_SH...|   LLY|    0.956|40.11|41.95|      1|
|  L|  Jan| 03|2012|006|Big_100_2_LONG_SH...|    DB|    0.888|35.23|39.67|      2|
|  S|  Jan| 03|2012|006|Big_100_2_LONG_SH...|   LLY|    0.956|40.11|41.95|      2|
|  L|  Jan| 03|2012|006|Big_100_5_LONG_SH...|    DB|    0.888|35.23|39.67|      5|
+---+-----+---+----+---+--------------------+------+---------+-----+-----+-------+
only showing top 5 rows
```

```
scala> def extractUptoSecondSubstring( str:String, sub:String): String = { val
result = str.substring(0, str.indexOf(sub, str.indexOf(sub) + 1)); result;}

scala> def extractStr(sub: String) = udf((x: String) =>
extractUptoSecondSubstring(x.trim(), sub))

scala> val df6A = df6.withColumn("type",
extractStr("_")($"file")).drop("file").select("month", "dom", "year", "y4",
"type", "ls", "ticker", "S/P Ratio", "SP", "PP", "nStocks")
```

```
scala> df6A.show(5)
+-----+---+----+---+-------+---+------+---------+-----+-----+-------+
|month|dom|year| y4|   type| ls|ticker|S/P Ratio|   SP|   PP|nStocks|
+-----+---+----+---+-------+---+------+---------+-----+-----+-------+
|  Jan| 03|2012|006|Big_100|  L|    DB|    0.888|35.23|39.67|      1|
|  Jan| 03|2012|006|Big_100|  S|   LLY|    0.956|40.11|41.95|      1|
|  Jan| 03|2012|006|Big_100|  L|    DB|    0.888|35.23|39.67|      2|
|  Jan| 03|2012|006|Big_100|  S|   LLY|    0.956|40.11|41.95|      2|
|  Jan| 03|2012|006|Big_100|  L|    DB|    0.888|35.23|39.67|      5|
+-----+---+----+---+-------+---+------+---------+-----+-----+-------+

only showing top 5 rows
```

```
scala> def containsSubstring( str:String, substr:String): Double = {
if (str.contains(substr)) 1 else 0}

scala> def udfContains(substr: String) = udf((x: String) =>
containsSubstring(x, substr))
scala> def udfVec() = udf[org.apache.spark.ml.linalg.Vector, String,
Int, Double, Double, Double] { (a, b, c, d, e) => val x = a match {
case "Monday" => 1; case "Tuesday" => 2; case "Wednesday" => 3; case
"Thursday" => 4; case "Friday" => 5; case "Saturday" => 6; case
"Sunday" => 7; }; Vectors.dense(x, b, c, d, e);}
```

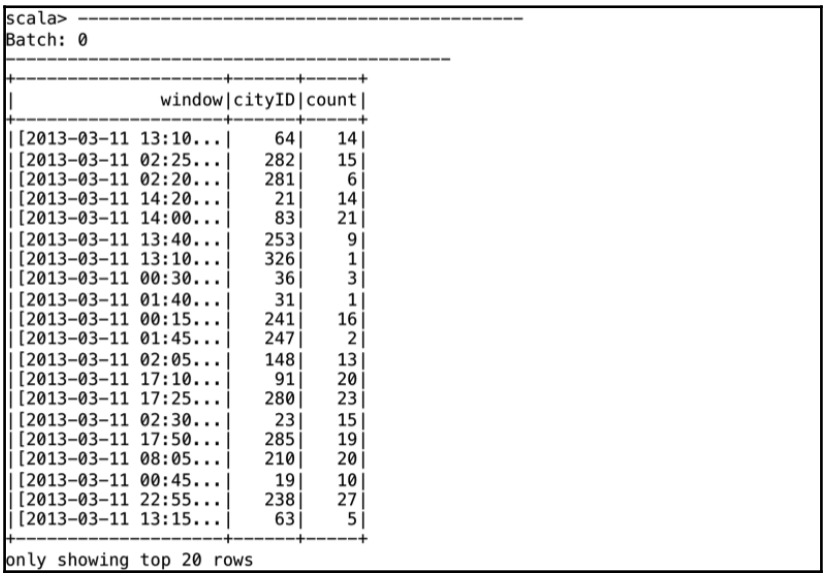
```
scala> val joinedRained = joinedDayDowDF.withColumn("label",
udfContains("Rain")($"Events")).withColumn("features",
udfVec()($"dow", $"month", $"dsm_1", $"dsm_2", $"dsm_3"))

scala> val labelIndexer = new
StringIndexer().setInputCol("label").setOutputCol("indexedLabel").fit
(joinedRained)

scala> val featureIndexer = new
VectorIndexer().setInputCol("features").setOutputCol("indexedFeatures
").setMaxCategories(7).fit(joinedRained)
```

```
scala> val Array(trainingData, testData) =
joinedRained.randomSplit(Array(0.7, 0.3))
```

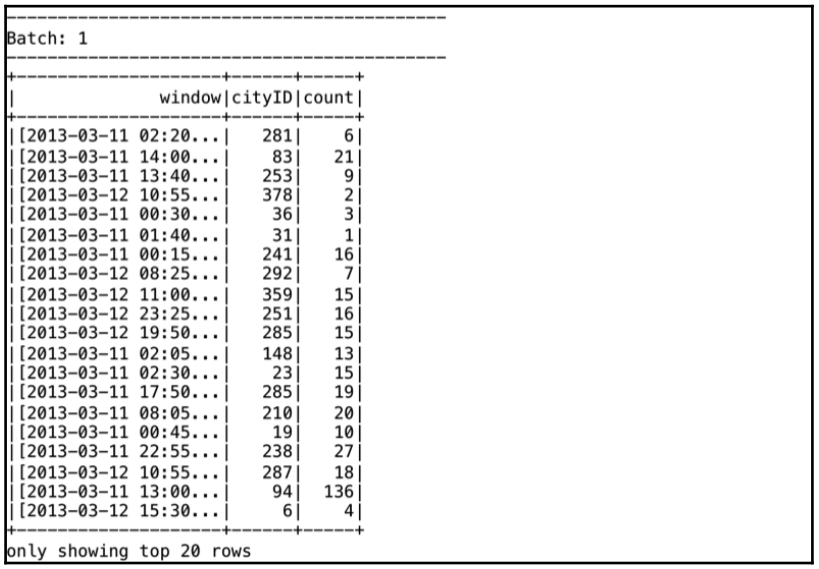
```
+--------------+-----+--------------------+
|predictedLabel|label|            features|
+--------------+-----+--------------------+
|           0.0|  1.0|[5.0,8.0,336.0,26...|
|           1.0|  0.0|[4.0,7.0,837.0,47...|
|           0.0|  1.0|[6.0,4.0,2007.0,3...|
|           0.0|  1.0|[2.0,7.0,1292.0,3...|
|           0.0|  0.0|[6.0,5.0,2278.0,2...|
+--------------+-----+--------------------+
only showing top 5 rows
```

```
Learned classification forest model:
RandomForestClassificationModel (uid=rfc_8dbd03c1c4e1) with 10 trees
  Tree 0 (weight 1.0):
    If (feature 3 <= 2492.0)
     If (feature 2 <= 4307.0)
      If (feature 3 <= 689.0)
       If (feature 0 in {1.0,2.0,4.0,5.0,6.0})
        If (feature 1 <= 10.0)
         Predict: 0.0
        Else (feature 1 > 10.0)
         Predict: 1.0
.
.
.
Tree 9 (weight 1.0):
    If (feature 3 <= 689.0)
     If (feature 3 <= 258.0)
      If (feature 1 <= 8.0)
       If (feature 4 <= 5457.0)
        If (feature 4 <= 1980.0)
         Predict: 0.0
        Else (feature 4 > 1980.0)
         Predict: 0.0
```

Chapter 5:
Using Spark SQL in Streaming Applications



```
scala> -------------------------------------------
Batch: 0
-------------------------------------------
+--------------------+------+-----+
|              window|cityID|count|
+--------------------+------+-----+
|[2013-03-11 13:10...|    64|   14|
|[2013-03-11 02:25...|   282|   15|
|[2013-03-11 02:20...|   281|    6|
|[2013-03-11 14:20...|    21|   14|
|[2013-03-11 14:00...|    83|   21|
|[2013-03-11 13:40...|   253|    9|
|[2013-03-11 13:10...|   326|    1|
|[2013-03-11 00:30...|    36|    3|
|[2013-03-11 01:40...|    31|    1|
|[2013-03-11 00:15...|   241|   16|
|[2013-03-11 01:45...|   247|    2|
|[2013-03-11 02:05...|   148|   13|
|[2013-03-11 17:10...|    91|   20|
|[2013-03-11 17:25...|   280|   23|
|[2013-03-11 02:30...|    23|   15|
|[2013-03-11 17:50...|   285|   19|
|[2013-03-11 08:05...|   210|   20|
|[2013-03-11 00:45...|    19|   10|
|[2013-03-11 22:55...|   238|   27|
|[2013-03-11 13:15...|    63|    5|
+--------------------+------+-----+
only showing top 20 rows
```



```
-------------------------------------------
Batch: 1
-------------------------------------------
+--------------------+------+-----+
|              window|cityID|count|
+--------------------+------+-----+
|[2013-03-11 02:20...|   281|    6|
|[2013-03-11 14:00...|    83|   21|
|[2013-03-11 13:40...|   253|    9|
|[2013-03-12 10:55...|   378|    2|
|[2013-03-11 00:30...|    36|    3|
|[2013-03-11 01:40...|    31|    1|
|[2013-03-11 00:15...|   241|   16|
|[2013-03-12 08:25...|   292|    7|
|[2013-03-12 11:00...|   359|   15|
|[2013-03-12 23:25...|   251|   16|
|[2013-03-12 19:50...|   285|   15|
|[2013-03-11 02:05...|   148|   13|
|[2013-03-11 02:30...|    23|   15|
|[2013-03-11 17:50...|   285|   19|
|[2013-03-11 08:05...|   210|   20|
|[2013-03-11 00:45...|    19|   10|
|[2013-03-11 22:55...|   238|   27|
|[2013-03-12 10:55...|   287|   18|
|[2013-03-11 13:00...|    94|  136|
|[2013-03-12 15:30...|     6|    4|
+--------------------+------+-----+
only showing top 20 rows
```

```
scala> -------------------------------------------
Batch: 0
-------------------------------------------
+--------------------+---------------+-----+
|              window|       cityName|count|
+--------------------+---------------+-----+
|[2013-03-11 11:05...|        bazhong|    1|
|[2013-03-11 19:30...|       shaoxing|  135|
|[2013-03-11 01:15...|        nanping|    4|
|[2013-03-11 21:40...|       xiangfan|   23|
|[2013-03-11 18:55...|         wuzhou|   18|
|[2013-03-11 03:20...|        jiuquan|    5|
|[2013-03-11 20:20...|      liangshan|   31|
|[2013-03-11 19:20...|       shaoyang|   18|
|[2013-03-11 04:40...|        nanyang|   13|
|[2013-03-11 09:25...|taizhou_jiangsu|    3|
|[2013-03-11 07:10...|           yaan|    2|
|[2013-03-11 07:30...|        suining|    8|
|[2013-03-11 22:35...|        lvliang|   36|
|[2013-03-11 14:20...|       zhongwei|    3|
|[2013-03-11 05:55...|        bazhong|    1|
|[2013-03-11 12:50...|         ziyang|   21|
|[2013-03-11 02:25...|        taizhou|   23|
|[2013-03-11 03:50...|         benxi|    3|
|[2013-03-11 08:05...|       neijiang|   25|
|[2013-03-11 14:20...|         fuyang|   33|
+--------------------+---------------+-----+
only showing top 20 rows
```

```
scala> -------------------------------------------
Batch: 0
-------------------------------------------
+-------------------+--------------------+--------+---------+---------+
|                 ts|               bidid|bidprice|slotprice| cityName|
+-------------------+--------------------+--------+---------+---------+
|2013-03-11 17:21:01|f2ce7b51f499eae08...|     300|        5| changzhi|
|2013-03-11 17:21:01|dabbf5f389089c39d...|     300|       59|   ningbo|
|2013-03-11 17:21:01|55cc617434cd07963...|     300|        5|  unknown|
|2013-03-11 17:21:01|de10b3396b222f60f...|     300|        5| shenyang|
|2013-03-11 17:21:01|375afd6a39e551874...|     300|       52| changzhi|
|2013-03-11 17:21:01|67f35f322a4936370...|     300|        5|   leshan|
|2013-03-11 17:21:01|1dbfbeafe74bbd9e0...|     300|      295|   leshan|
|2013-03-11 17:21:01|6d1f7c8008fd440b9...|     300|        5|    jinan|
|2013-03-11 17:21:01|c3872ff374277ad36...|     300|        8| shenzhen|
|2013-03-11 17:21:01|5530617cd63368116...|     300|       52|     wuxi|
|2013-03-11 17:21:01|8927586d9cbdd83a2...|     300|        5|  nanjing|
|2013-03-11 17:21:01|bec9fd1c23f5f3908...|     300|        5|  chengdu|
|2013-03-11 17:21:01|a35a52d81f0f78ed7...|     300|        5|guangzhou|
|2013-03-11 17:21:01|184bf096e7fa66a12...|     300|      148|   zhuhai|
|2013-03-11 17:21:01|cbe88409742e18f86...|     300|        5| mianyang|
|2013-03-11 17:21:01|97c5e98e080d43a5f...|     300|        5| jincheng|
|2013-03-11 17:21:01|ccd41a3c166b0c110...|     300|        5|   daqing|
|2013-03-11 17:21:01|dfadede03a58b6dfd...|     300|        5|      aba|
|2013-03-11 17:21:01|be059b6aa9289bb46...|     300|        5| mianyang|
|2013-03-11 17:21:01|d6afb0b0a9f486164...|     300|        5|  nanjing|
+-------------------+--------------------+--------+---------+---------+

only showing top 20 rows
```

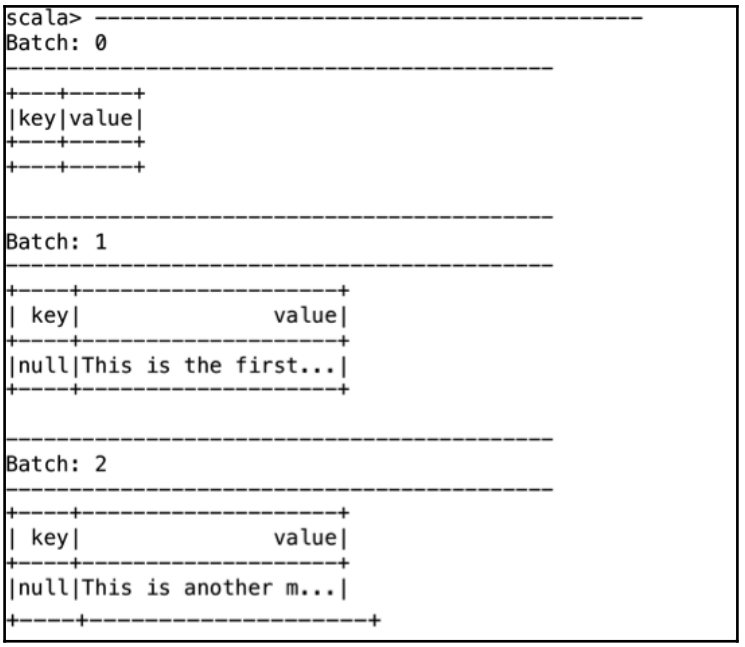
```
scala> Mozilla/5.0 (Windows NT 5.1) AppleWebKit/537.1 (KHTML, like Gecko)
Chrome/21.0.1180.89 Safari/537.1
Mozilla/5.0 (Windows NT 5.1) AppleWebKit/537.1 (KHTML, like Gecko)
Chrome/21.0.1180.89 Safari/537.1
Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; InfoPath.2)
                                        .
                                        .
                                        .
Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; Trident/4.0; .NET CLR
2.0.50727)
Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; Trident/4.0; .NET CLR
2.0.50727)
Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; Trident/4.0; FDM; .NET4.0C)
```

```
+----------+-------+
|adexchange|  count|
+----------+-------+
|         3| 839458|
|         1|1049105|
|         2|1540938|
+----------+-------+
```

```
Found 17 items
drwxr-xr-x   - aurobindosarkar supergroup          0 2017-08-25 00:03
/pout/_spark_metadata
-rw-r--r--   1 aurobindosarkar supergroup    5028115 2017-08-25 00:03
/pout/part-00000-30859feb-965e-41be-b257-bb872b3c0f44-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    5147398 2017-08-25 00:03
/pout/part-00000-6927b229-fcd0-4411-9de8-41a7b77359dd-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    4956498 2017-08-25 00:03
/pout/part-00001-403362a0-1141-4602-8e94-241442546ed8-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    5095930 2017-08-25 00:03
/pout/part-00001-a8c5de46-5e42-425f-920e-f3bad5c05fbf-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    5064670 2017-08-25 00:03
/pout/part-00002-90f4e91d-45a6-40e5-b833-ff0014e989ca-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    5110948 2017-08-25 00:03
/pout/part-00002-eafe9de5-c376-4a1d-a2bb-c9cf006b60d1-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    5179985 2017-08-25 00:03
/pout/part-00003-6a053817-36e3-44f8-9220-74bfa33d8fd2-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    4863236 2017-08-25 00:03
/pout/part-00003-facee78e-9be9-4d95-b850-8e053f1db48d-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    4991215 2017-08-25 00:03
/pout/part-00004-333d8dcb-8b63-4c46-b5e9-6b989003d9c3-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    5077441 2017-08-25 00:03
/pout/part-00004-746821ad-4657-426e-aef6-94baa672e10d-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    4971655 2017-08-25 00:03
/pout/part-00005-0cc84880-6058-4038-85d7-83089d100036-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    5170948 2017-08-25 00:03
/pout/part-00005-78629915-701e-4c2e-a6d0-aad09e1eea54-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    4973106 2017-08-25 00:03
/pout/part-00006-930e03a4-7ab4-4061-87da-f626abf6bd1c-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    5037874 2017-08-25 00:03
/pout/part-00006-f86ca876-dbe0-4a1b-80af-65a4990026db-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    4866992 2017-08-25 00:03
/pout/part-00007-1a0cecb0-53bd-4fb4-8804-47dd7e5612d5-c000.snappy.parquet
-rw-r--r--   1 aurobindosarkar supergroup    4649468 2017-08-25 00:03
/pout/part-00007-6b85aebd-80d6-4d4f-8a8e-ab605155f7b0-c000.snappy.parquet
```

```
Found 4 items
drwxr-xr-x   - aurobindosarkar supergroup          0 2017-08-25 00:03
/poutcp/commits
-rw-r--r--   1 aurobindosarkar supergroup         45 2017-08-25 00:03
/poutcp/metadata
drwxr-xr-x   - aurobindosarkar supergroup          0 2017-08-25 00:03
/poutcp/offsets
drwxr-xr-x   - aurobindosarkar supergroup          0 2017-08-25 00:03
/poutcp/sources
```

```
{
  "id" : "0ebe31f5-6b76-46ea-a328-cd0c637be49c",
  "runId" : "6f203d14-2a3a-4c9f-9ea0-8a6783d97873",
  "name" : null,
  "timestamp" : "2017-08-25T18:38:28.421Z",
  "numInputRows" : 0,
  "inputRowsPerSecond" : 0.0,
  "processedRowsPerSecond" : 0.0,
  "durationMs" : {
    "getOffset" : 2,
    "triggerExecution" : 2
  },
  "stateOperators" : [ {
    "numRowsTotal" : 156570,
    "numRowsUpdated" : 0
  } ],
  "sources" : [ {
    "description" :
"FileStreamSource[file:/Users/aurobindosarkar/Downloads/make-ipinyou-data-
master/original-data/ipinyou.contest.dataset/bidfiles]",
    "startOffset" : {
      "logOffset" : 1
    },
    "endOffset" : {
      "logOffset" : 1
    },
    "numInputRows" : 0,
    "inputRowsPerSecond" : 0.0,
  ...
```


scala> ---
Batch: 0

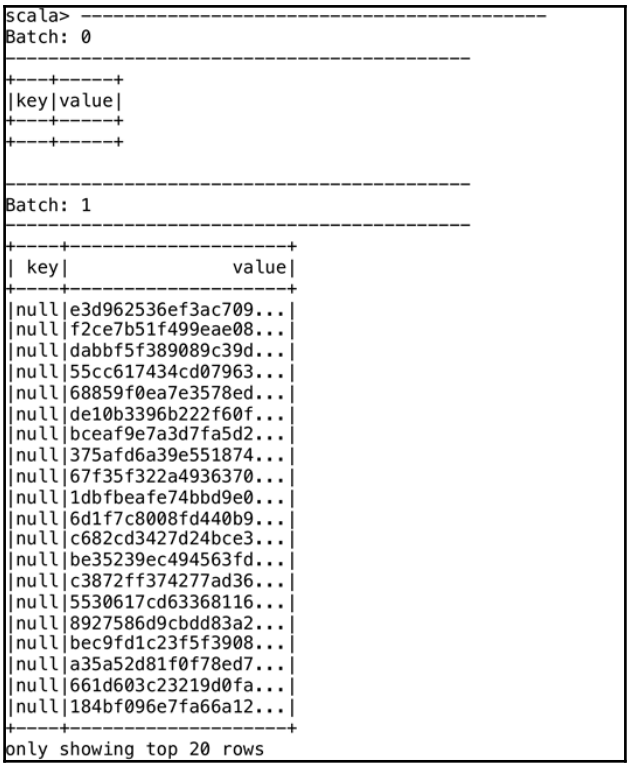
+---+-----+
|key|value|
+---+-----+
+---+-----+

Batch: 1

+----+-----------------+
| key| value|
+----+-----------------+
|null|This is the first...|
+----+-----------------+

Batch: 2

+----+-----------------+
| key| value|
+----+-----------------+
|null|This is another m...|
+----+-----------------+


scala> ---
Batch: 0

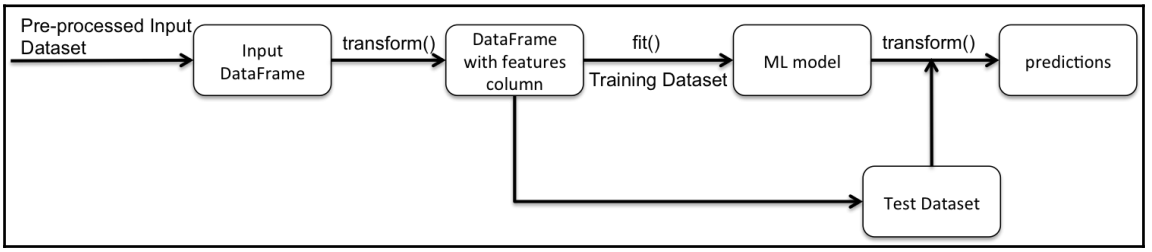
+---+-----+
|key|value|
+---+-----+
+---+-----+

Batch: 1

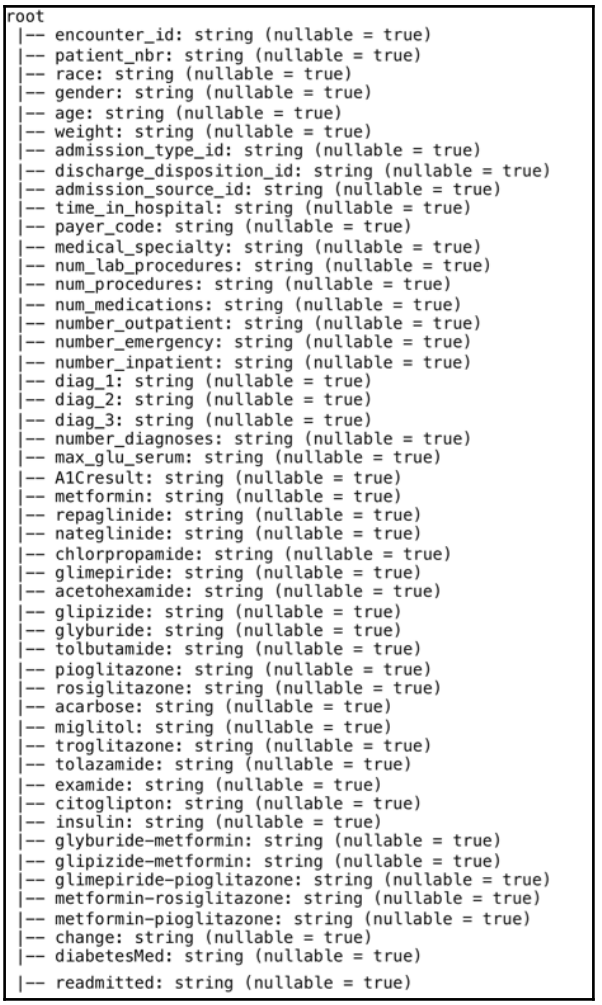
+----+--------------------+
| key| value|
+----+--------------------+
null	e3d962536ef3ac709...
null	f2ce7b51f499eae08...
null	dabbf5f389089c39d...
null	55cc617434cd07963...
null	68859f0ea7e3578ed...
null	de10b3396b222f60f...
null	bceaf9e7a3d7fa5d2...
null	375afd6a39e551874...
null	67f35f322a4936370...
null	1dbfbeafe74bbd9e0...
null	6d1f7c8008fd440b9...
null	c682cd3427d24bce3...
null	be35239ec494563fd...
null	c3872ff374277ad36...
null	5530617cd63368116...
null	8927586d9cbdd83a2...
null	bec9fd1c23f5f3908...
null	a35a52d81f0f78ed7...
null	661d603c23219d0fa...
null	184bf096e7fa66a12...
+----+--------------------+
only showing top 20 rows

```
[{"$type":"Tfl.Api.Presentation.Entities.Prediction,
Tfl.Api.Presentation.Entities","id":"-
403982650","operationType":1,"vehicleId":"201","naptanId":"940GZZLUER
C","stationName":"Edgware Road (Circle Line) Underground
Station","lineId":"circle","lineName":"Circle","platformName":"Eastbo
und - Platform
1","bearing":"","destinationNaptanId":"940GZZLUERC","destinationName"
:"Edgware Road (Circle Line) Underground Station","timestamp":"2017-
08-25T18:02:52Z","timeToStation":693,"currentLocation":"Between
Sloane Square and South Kensington","towards":"Edgware Road
(Circle)","expectedArrival":"2017-08-
25T18:14:25Z","timeToLive":"2017-08-
25T18:14:25Z","modeName":"tube","timing":{"$type":"Tfl.Api.Presentati
on.Entities.PredictionTiming,
Tfl.Api.Presentation.Entities","countdownServerAdjustment":"00:00:00"
,"source":"0001-01-01T00:00:00","insert":"0001-01-
01T00:00:00","read":"2017-08-25T18:02:31.585Z","sent":"2017-08-
25T18:02:52Z","received":"0001-01-
01T00:00:00"}},{"$type":"Tfl.Api.Presentation.Entities.Prediction,
Tfl.Api.Presentation.Entities","id":"1116945755","operationType":1,"v
ehicleId":"202","naptanId":"940GZZLUERC","stationName":"Edgware Road
(Circle Line) Underground
Station","lineId":"circle","lineName":"Circle","platformName":"Eastbo
und - Platform
1","bearing":"","destinationNaptanId":"940GZZLUERC","destinationName"
:"Edgware Road (Circle Line) Underground Station","timestamp":"2017-
08-25T18:02:52Z","timeToStation":1053,"currentLocation":"At St.
James's Park Platform 1","towards":"Edgware Road
(Circle)","expectedArrival":"2017-08-
25T18:20:25Z","timeToLive":"2017-08-
25T18:20:25Z","modeName":"tube","timing":{"$type":"Tfl.Api.Presentati
on.Entities.PredictionTiming,
Tfl.Api.Presentation.Entities","countdownServerAdjustment":"00:00:00"
,"source":"0001-01-01T00:00:00","insert":"0001-01-
01T00:00:00","read":"2017-08-25T18:02:31.6Z","sent":"2017-08-
25T18:02:52Z","received":"0001-01-
01T00:00:00"}},{"$type":"Tfl.Api.Presentation.Entities.Prediction,
Tfl.Api.Presentation.Entities","id":"557369852","operationType":1,"ve
hicleId":"203","naptanId":"940GZZLUERC","stationName":"Edgware Road
(Circle Line) Underground
Station","lineId":"circle","lineName":"Circle","platformName":"Eastbo
und - Platform
1","bearing":"","destinationNaptanId":"940GZZLUERC","destinationName"
:"Edgware Road (Circle Line) Underground Station","timestamp":"2017-
08-25T18:02:52Z","timeToStation":1713,"currentLocation":"At Monument
Platform 1","towards":"Edgware Road
(Circle)","expectedArrival":"2017-08-
25T18:31:25Z","timeToLive":"2017-08-
25T18:31:25Z","modeName":"tube","timing":{"$type":"Tfl.Api.Presentati
on.Entities.PredictionTiming,
Tfl.Api.Presentation.Entities","countdownServerAdjustment":"00:00:00"
,"source":"0001-01-01T00:00:00","insert":"0001-01-
01T00:00:00","read":"2017-08-25T18:02:31.632Z","sent":"2017-08-
25T18:02:52Z","received":"0001-01-01T00:00:00"}}]
```

Chapter 6:
Using Spark SQL in Machine Learning Appli cations




```
scala> import spark.implicits._
scala> import org.apache.spark.sql.types._
scala> import org.apache.spark.sql.{DataFrameNaFunctions, Row}
scala> import org.apache.spark.ml.feature.{StringIndexer, VectorAssembler,
IndexToString, VectorIndexer, OneHotEncoder, PCA, Binarizer, VectorSlicer,
StandardScaler, Bucketizer, ChiSqSelector, Normalizer }
scala> import org.apache.spark.ml.Pipeline
scala> import org.apache.spark.ml.evaluation.{MulticlassClassificationEvaluator,
BinaryClassificationEvaluator}
scala> import org.apache.spark.sql.functions.{ sum,when , row_number, max, broadcast}
scala> import org.apache.spark.sql.expressions.Window
scala> import org.apache.spark.ml.classification.{RandomForestClassificationModel,
RandomForestClassifier, LogisticRegression, DecisionTreeClassificationModel,
DecisionTreeClassifier, GBTClassificationModel, GBTClassifier}
scala> import org.apache.spark.ml.tuning.{CrossValidator, ParamGridBuilder}
scala> import org.apache.spark.ml.linalg.{Vector, Vectors}
```



```
root
 |-- encounter_id: string (nullable = true)
 |-- patient_nbr: string (nullable = true)
 |-- race: string (nullable = true)
 |-- gender: string (nullable = true)
 |-- age: string (nullable = true)
 |-- weight: string (nullable = true)
 |-- admission_type_id: string (nullable = true)
 |-- discharge_disposition_id: string (nullable = true)
 |-- admission_source_id: string (nullable = true)
 |-- time_in_hospital: string (nullable = true)
 |-- payer_code: string (nullable = true)
 |-- medical_specialty: string (nullable = true)
 |-- num_lab_procedures: string (nullable = true)
 |-- num_procedures: string (nullable = true)
 |-- num_medications: string (nullable = true)
 |-- number_outpatient: string (nullable = true)
 |-- number_emergency: string (nullable = true)
 |-- number_inpatient: string (nullable = true)
 |-- diag_1: string (nullable = true)
 |-- diag_2: string (nullable = true)
 |-- diag_3: string (nullable = true)
 |-- number_diagnoses: string (nullable = true)
 |-- max_glu_serum: string (nullable = true)
 |-- A1Cresult: string (nullable = true)
 |-- metformin: string (nullable = true)
 |-- repaglinide: string (nullable = true)
 |-- nateglinide: string (nullable = true)
 |-- chlorpropamide: string (nullable = true)
 |-- glimepiride: string (nullable = true)
 |-- acetohexamide: string (nullable = true)
 |-- glipizide: string (nullable = true)
 |-- glyburide: string (nullable = true)
 |-- tolbutamide: string (nullable = true)
 |-- pioglitazone: string (nullable = true)
 |-- rosiglitazone: string (nullable = true)
 |-- acarbose: string (nullable = true)
 |-- miglitol: string (nullable = true)
 |-- troglitazone: string (nullable = true)
 |-- tolazamide: string (nullable = true)
 |-- examide: string (nullable = true)
 |-- citoglipton: string (nullable = true)
 |-- insulin: string (nullable = true)
 |-- glyburide-metformin: string (nullable = true)
 |-- glipizide-metformin: string (nullable = true)
 |-- glimepiride-pioglitazone: string (nullable = true)
 |-- metformin-rosiglitazone: string (nullable = true)
 |-- metformin-pioglitazone: string (nullable = true)
 |-- change: string (nullable = true)
 |-- diabetesMed: string (nullable = true)
 |-- readmitted: string (nullable = true)
```

```
[2278392,8222157,Caucasian,Female,[0-10),?,6,25,1,1,?,Pediatrics-
Endocrinology,41,0,1,0,0,0,250.83,?,?,1,None,None,No,No,No,No,No,No,No,No,No,No,No,No,
No,No,No,No,No,No,No,No,No,No,No,No,NO]
[149190,55629189,Caucasian,Female,[10-
20),?,1,1,7,3,?,?,59,0,18,0,0,0,276,250.01,255,9,None,None,No,No,No,No,No,No,No,No,
No,No,No,No,No,No,No,No,Up,No,No,No,No,Ch,Yes,>30]
[64410,86047875,AfricanAmerican,Female,[20-
30),?,1,1,7,2,?,?,11,5,13,2,0,1,648,250,V27,6,None,None,No,No,No,No,No,No,Steady,No,No
,No,No,No,No,No,No,No,No,No,No,No,No,No,No,No,Yes,NO]
[500364,82442376,Caucasian,Male,[30-
40),?,1,1,7,2,?,?,44,1,16,0,0,0,0,8,250.43,403,7,None,None,No,No,No,No,No,No,No,No,No,No
,No,No,No,No,No,No,No,Up,No,No,No,No,No,Ch,Yes,NO]
[16680,42519267,Caucasian,Male,[40-
50),?,1,1,7,1,?,?,51,0,8,0,0,0,197,157,250,5,None,None,No,No,No,No,No,No,Steady,No,No,
No,No,No,No,No,No,No,No,No,Steady,No,No,No,No,No,Ch,Yes,NO]
```

```
scala> val joinDF = diaDataAdmttedDF.join(dchrgDispDF,
diaDataAdmttedDF("discharge_disposition_id") ===
dchrgDispDF("dchrgDispId")).withColumnRenamed("description",
"discharge_disposition").drop(dchrgDispDF("dchrgDispId")).join(admTypeDF,
diaDataAdmttedDF("admission_type_id") ===
admTypeDF("admTypeId")).withColumnRenamed("description",
"admission_type").drop(admTypeDF("admTypeId")).join(admSrcDF,
diaDataAdmttedDF("admission_source_id") ===
admSrcDF("admission_source_id")).withColumnRenamed("description",
"admission_source").drop(admSrcDF("admission_source_id"))
```

```
+------------+--------------------+---------+--------------------+
|encounter_id|           dchrgDisp|  admType|    admission_source|
+------------+--------------------+---------+--------------------+
|   392013782|  Discharged to home| Elective|  Physician Referral|
|    46598346|                NULL| Elective|Transfer from a h...|
|   138229974|  Discharged to home|Emergency|      Emergency Room|
|     7269804|Discharged/transf...|   Urgent|  Physician Referral|
|   311887940|  Discharged to home|   Urgent|      Emergency Room|
|   201558258|Discharged/transf...|Emergency|      Emergency Room|
|   230227854|  Discharged to home|   Urgent|  Physician Referral|
|    59322732|                NULL| Elective|  Physician Referral|
|   128185728|  Discharged to home|   Urgent|      Emergency Room|
|   180128832|Discharged/transf...|Emergency|      Emergency Room|
|   119373450|Discharged/transf...| Elective|  Physician Referral|
|    82723956|  Discharged to home|Emergency|      Emergency Room|
|   259028538|  Discharged to home|Emergency|  Physician Referral|
|   361461086|  Discharged to home|   Urgent|  Physician Referral|
|    59154006|  Discharged to home| Elective|  Physician Referral|
|   233433594|Discharged/transf...|Emergency|      Emergency Room|
|   423739748|  Discharged to home|   Urgent|  Physician Referral|
|   190090386|  Discharged to home|   Urgent|      Emergency Room|
|   147463902|Discharged/transf...|Emergency|      Emergency Room|
|   366137420|Discharged/transf...| Elective|  Physician Referral|
+------------+--------------------+---------+--------------------+
only showing top 20 rows
```

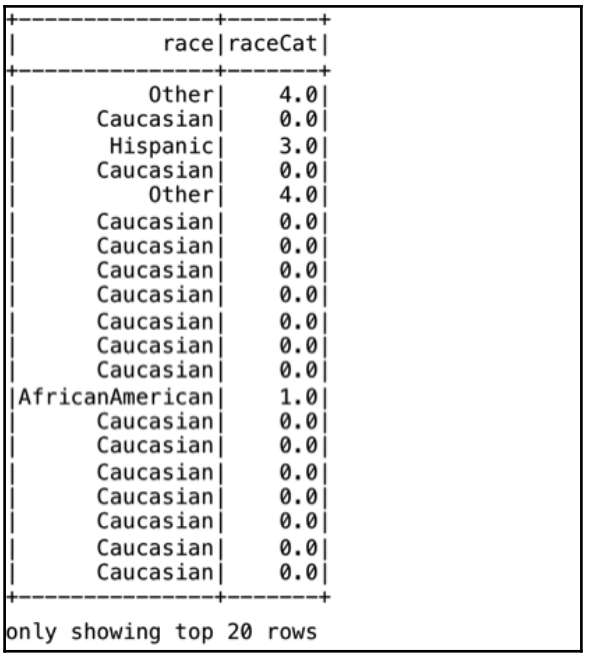
```
[Discharged to home,43473]
[Discharged/transferred to SNF,9167]
[Discharged/transferred to home with home health service,8442]
[NULL,2397]
[Discharged/transferred to another short term hospital,1463]
[Discharged/transferred to another rehab fac including rehab units of a hospital
.,1274]
[Discharged/transferred to another type of inpatient care institution,844]
[Not Mapped,675]
[Discharged/transferred to ICF,554]
[Left AMA,411]
```

```
scala> val diaDataDrpColsDF = diaDataRplcMedSplDF.drop("encounter_id", "patient_nbr",
"diag_2", "diag_3", "max_glu_serum", "metformin", "repaglinide", "nateglinide",
"chlorpropamide", "glimepiride", "acetohexamide", "glipizide", "glyburide",
"tolbutamide", "pioglitazone", "rosiglitazone", "acarbose", "miglitol",
"troglitazone", "tolazamide", "examide", "citoglipton", "insulin", "glyburide-
metformin", "glipizide-metformin", "glimepiride-pioglitazone", "metformin-
rosiglitazone", "metformin-pioglitazone")
```

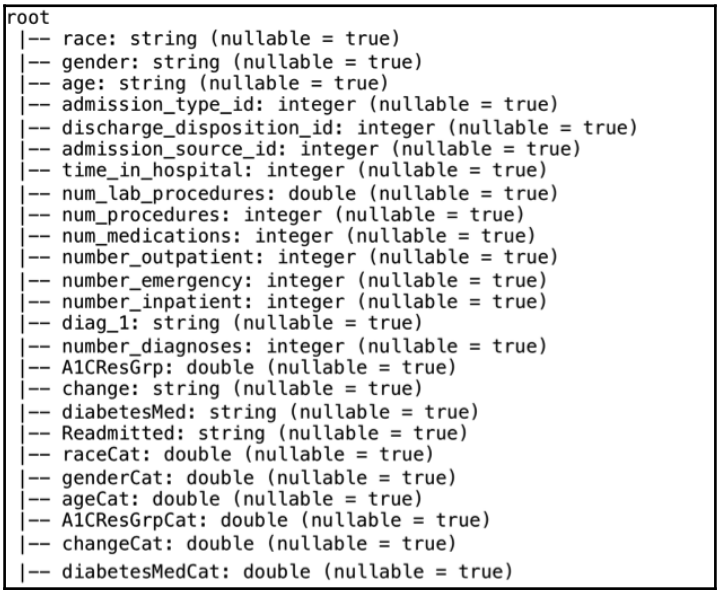
```
scala>  val diaDataFinalDF = diaDataRmvGndrDF.select($"race", $"gender",
$"age_category", $"admission_type_id".cast(IntegerType),
$"discharge_disposition_id".cast(IntegerType),
$"admission_source_id".cast(IntegerType), $"time_in_hospital".cast(IntegerType),
$"num_lab_procedures".cast(DoubleType), $"num_procedures".cast(IntegerType),
$"num_medications".cast(IntegerType), $"number_outpatient".cast(IntegerType),
$"number_emergency".cast(IntegerType), $"number_inpatient".cast(IntegerType),
$"diag_1", $"number_diagnoses".cast(IntegerType), $"A1CResGrp", $"change",
$"diabetesMed", $"Readmitted").withColumnRenamed("age_category", "age")
```

```
root
 |-- race: string (nullable = true)
 |-- gender: string (nullable = true)
 |-- age: string (nullable = true)
 |-- admission_type_id: integer (nullable = true)
 |-- discharge_disposition_id: integer (nullable = true)
 |-- admission_source_id: integer (nullable = true)
 |-- time_in_hospital: integer (nullable = true)
 |-- num_lab_procedures: double (nullable = true)
 |-- num_procedures: integer (nullable = true)
 |-- num_medications: integer (nullable = true)
 |-- number_outpatient: integer (nullable = true)
 |-- number_emergency: integer (nullable = true)
 |-- number_inpatient: integer (nullable = true)
 |-- diag_1: string (nullable = true)
 |-- number_diagnoses: integer (nullable = true)
 |-- A1CResGrp: double (nullable = true)
 |-- change: string (nullable = true)
 |-- diabetesMed: string (nullable = true)
 |-- Readmitted: string (nullable = true)
```

```
[Other,Male,Elder,3,1,1,3,33,2,10,1,0,1,428,9,No test was performed,No,Yes,Not
Readmitted]
[Caucasian,Female,Middle,3,18,4,13,44,0,11,0,0,2,V57,5,No test was
performed,No,Yes,Not Readmitted]
[Hispanic,Female,Elder,1,1,7,3,73,4,14,0,0,0,401,9,No test was performed,No,Yes,Not
Readmitted]
[Caucasian,Male,Elder,2,2,1,4,48,3,11,0,0,0,562,9,No test was performed,No,Yes,Not
Readmitted]
[Other,Female,Middle,2,1,7,1,13,1,6,0,0,0,427,6,No test was performed,No,No,Not
Readmitted]
```

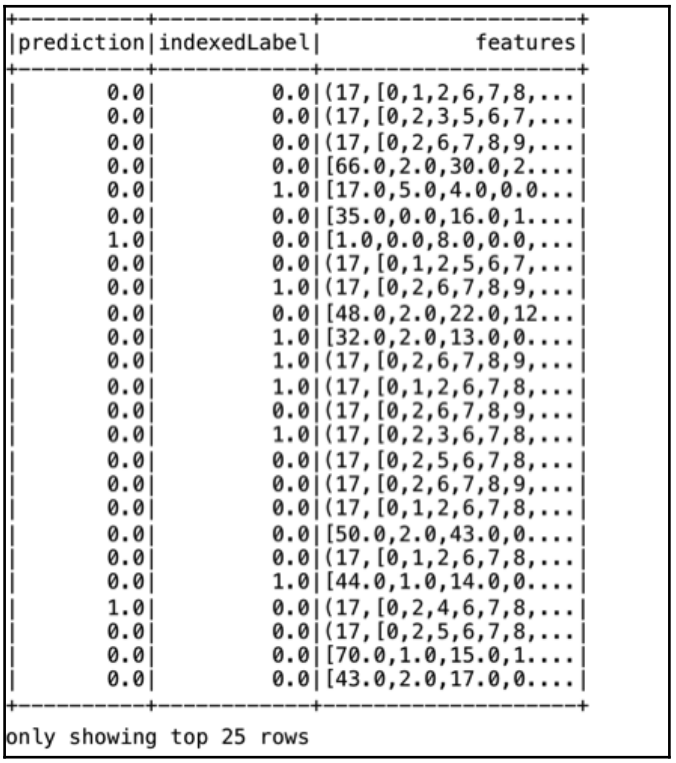


```
+---------------+-------+
|           race|raceCat|
+---------------+-------+
|          Other|    4.0|
|      Caucasian|    0.0|
|       Hispanic|    3.0|
|      Caucasian|    0.0|
|          Other|    4.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|AfricanAmerican|    1.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
|      Caucasian|    0.0|
+---------------+-------+
only showing top 20 rows
```



```
root
 |-- race: string (nullable = true)
 |-- gender: string (nullable = true)
 |-- age: string (nullable = true)
 |-- admission_type_id: integer (nullable = true)
 |-- discharge_disposition_id: integer (nullable = true)
 |-- admission_source_id: integer (nullable = true)
 |-- time_in_hospital: integer (nullable = true)
 |-- num_lab_procedures: double (nullable = true)
 |-- num_procedures: integer (nullable = true)
 |-- num_medications: integer (nullable = true)
 |-- number_outpatient: integer (nullable = true)
 |-- number_emergency: integer (nullable = true)
 |-- number_inpatient: integer (nullable = true)
 |-- diag_1: string (nullable = true)
 |-- number_diagnoses: integer (nullable = true)
 |-- A1CResGrp: double (nullable = true)
 |-- change: string (nullable = true)
 |-- diabetesMed: string (nullable = true)
 |-- Readmitted: string (nullable = true)
 |-- raceCat: double (nullable = true)
 |-- genderCat: double (nullable = true)
 |-- ageCat: double (nullable = true)
 |-- A1CResGrpCat: double (nullable = true)
 |-- changeCat: double (nullable = true)
 |-- diabetesMedCat: double (nullable = true)
```

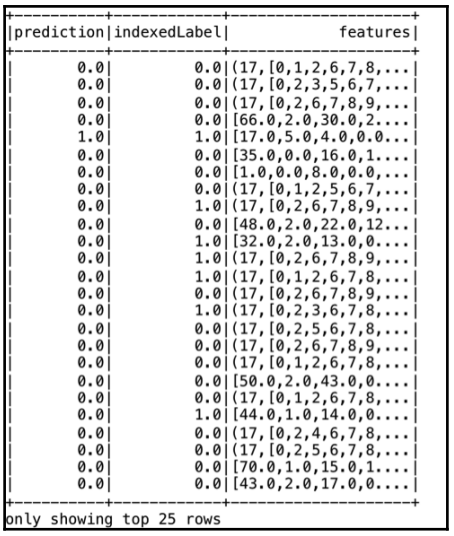
```
[Readmitted,(17,[0,2,6,7,8,9,10,12,15],[50.0,20.0,6.0,3.0,1.0,1.0,6.0,1.0,1.0])]
[Not Readmitted,[17.0,5.0,4.0,0.0,1.0,0.0,5.0,3.0,1.0,7.0,2.0,0.0,1.0,0.0,0.0,1.0,0.0]]
[Not Readmitted,(17,[0,2,3,6,7,8,9,10],[41.0,13.0,1.0,6.0,1.0,6.0,7.0,1.0])]
[Not Readmitted,(17,[0,2,5,6,7,8,9,10,12,15],[35.0,12.0,1.0,2.0,1.0,1.0,7.0,3.0,1.0,1.0])]

[Not Readmitted,[50.0,0.0,9.0,0.0,0.0,0.0,5.0,1.0,1.0,7.0,4.0,1.0,1.0,1.0,0.0,1.0,0.0]]
```



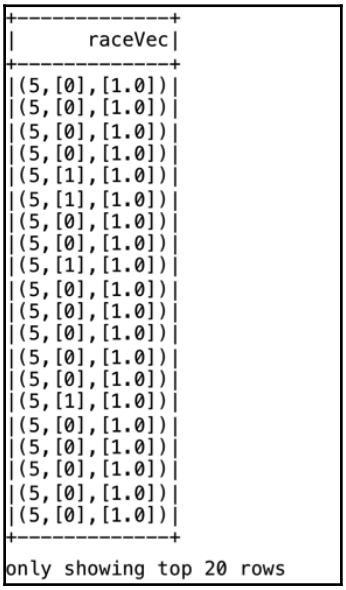
```
+----------+------------+--------------------+
|prediction|indexedLabel|            features|
+----------+------------+--------------------+
|       0.0|         0.0|(17,[0,1,2,6,7,8,...|
|       0.0|         0.0|(17,[0,2,3,5,6,7,...|
|       0.0|         0.0|(17,[0,2,6,7,8,9,...|
|       0.0|         0.0|[66.0,2.0,30.0,2....|
|       0.0|         1.0|[17.0,5.0,4.0,0.0...|
|       0.0|         0.0|[35.0,0.0,16.0,1....|
|       1.0|         0.0|[1.0,0.0,8.0,0.0,...|
|       0.0|         0.0|(17,[0,1,2,5,6,7,...|
|       0.0|         1.0|(17,[0,2,6,7,8,9,...|
|       0.0|         0.0|[48.0,2.0,22.0,12...|
|       0.0|         1.0|[32.0,2.0,13.0,0....|
|       0.0|         1.0|(17,[0,2,6,7,8,9,...|
|       0.0|         1.0|(17,[0,1,2,6,7,8,...|
|       0.0|         0.0|(17,[0,2,6,7,8,9,...|
|       0.0|         1.0|(17,[0,2,3,6,7,8,...|
|       0.0|         0.0|(17,[0,2,5,6,7,8,...|
|       0.0|         0.0|(17,[0,2,6,7,8,9,...|
|       0.0|         0.0|(17,[0,1,2,6,7,8,...|
|       0.0|         0.0|[50.0,2.0,43.0,0....|
|       0.0|         0.0|(17,[0,1,2,6,7,8,...|
|       0.0|         1.0|[44.0,1.0,14.0,0....|
|       1.0|         0.0|(17,[0,2,4,6,7,8,...|
|       0.0|         0.0|(17,[0,2,5,6,7,8,...|
|       0.0|         0.0|[70.0,1.0,15.0,1....|
|       0.0|         0.0|[43.0,2.0,17.0,0....|
+----------+------------+--------------------+
only showing top 25 rows
```

```
Learned classification forest model:
RandomForestClassificationModel (uid=rfc_d921395cbedc) with 10 trees
  Tree 0 (weight 1.0):
    If (feature 6 <= 5.0)
     If (feature 5 <= 2.0)
      If (feature 2 <= 12.0)
       If (feature 11 in {1.0,2.0,3.0,4.0,5.0})
        If (feature 1 <= 1.0)
         Predict: 1.0
        Else (feature 1 > 1.0)
         Predict: 0.0
       Else (feature 11 not in {1.0,2.0,3.0,4.0,5.0})
        If (feature 1 <= 3.0)
         Predict: 1.0
        Else (feature 1 > 3.0)
         Predict: 1.0
.
.
.
Else (feature 5 > 2.0)
       If (feature 10 <= 2.0)
        If (feature 14 in {2.0})
         Predict: 0.0
        Else (feature 14 not in {2.0})
         Predict: 1.0
       Else (feature 10 > 2.0)
        If (feature 5 <= 3.0)
         Predict: 0.0
        Else (feature 5 > 3.0)
         Predict: 0.0
```

```
paramGrid: Array[org.apache.spark.ml.param.ParamMap] =
Array({
      rfc_63209c8eb2c9-impurity: entropy,
      rfc_63209c8eb2c9-maxBins: 25,
      rfc_63209c8eb2c9-maxDepth: 4
}, {
      rfc_63209c8eb2c9-impurity: entropy,
      rfc_63209c8eb2c9-maxBins: 28,
      rfc_63209c8eb2c9-maxDepth: 4
}, {
      rfc_63209c8eb2c9-impurity: entropy,
      rfc_63209c8eb2c9-maxBins: 31,
      rfc_63209c8eb2c9-maxDepth: 4
}, {
      rfc_63209c8eb2c9-impurity: gini,
      rfc_63209c8eb2c9-maxBins: 25,
      rfc_63209c8eb2c9-maxDepth: 4
}, {
      rfc_63209c8eb2c9-impurity: gini,
      rfc_63209c8eb2c9-maxBins: 28,
      rfc_63209c8eb2c9-maxDepth: 4
}, {
      rfc_63209c8eb2c9-impurity: gini,
      rfc_63209c8eb2c9-maxBins: 31,
      rfc_63209c8eb2c9-maxDepth: 4
}, {
      rfc_63209c8eb2c9-impurity: entropy,
      rfc_63209c8eb2c9-maxBins: 25,
      rfc_63209c8eb2c9-maxDepth: 6
}, {
       rfc_63209c8eb2c9-impu...
```



```
+----------+------------+--------------------+
|prediction|indexedLabel|            features|
+----------+------------+--------------------+
|       0.0|         0.0|(17,[0,1,2,6,7,8,...|
|       0.0|         0.0|(17,[0,2,3,5,6,7,...|
|       0.0|         0.0|(17,[0,2,6,7,8,9,...|
|       0.0|         0.0|[66.0,2.0,30.0,2....|
|       1.0|         1.0|[17.0,5.0,4.0,0.0...|
|       0.0|         0.0|[35.0,0.0,16.0,1....|
|       0.0|         0.0|[1.0,0.0,8.0,0.0,...|
|       0.0|         0.0|(17,[0,1,2,5,6,7,...|
|       0.0|         1.0|(17,[0,2,6,7,8,9,...|
|       0.0|         0.0|[48.0,2.0,22.0,12...|
|       0.0|         1.0|[32.0,2.0,13.0,0....|
|       0.0|         1.0|(17,[0,2,6,7,8,9,...|
|       0.0|         1.0|(17,[0,1,2,6,7,8,...|
|       0.0|         0.0|(17,[0,2,6,7,8,9,...|
|       0.0|         1.0|(17,[0,2,3,6,7,8,...|
|       0.0|         0.0|(17,[0,2,5,6,7,8,...|
|       0.0|         0.0|(17,[0,2,6,7,8,9,...|
|       0.0|         0.0|(17,[0,1,2,6,7,8,...|
|       0.0|         0.0|[50.0,2.0,43.0,0....|
|       0.0|         0.0|(17,[0,1,2,6,7,8,...|
|       0.0|         1.0|[44.0,1.0,14.0,0....|
|       0.0|         0.0|(17,[0,2,4,6,7,8,...|
|       0.0|         0.0|(17,[0,2,5,6,7,8,...|
|       0.0|         0.0|[70.0,1.0,15.0,1....|
|       0.0|         0.0|[43.0,2.0,17.0,0....|
+----------+------------+--------------------+
only showing top 25 rows
```



```
+-------------+
|      raceVec|
+-------------+
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[1],[1.0])|
|(5,[1],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[1],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[1],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
|(5,[0],[1.0])|
+-------------+
only showing top 20 rows
```

```
scala> bucketedData.select("num_lab_procedures", "bucketedLabProcs").show()
+------------------+----------------+
|num_lab_procedures|bucketedLabProcs|
+------------------+----------------+
|              50.0|             2.0|
|              17.0|             0.0|
|              41.0|             2.0|
|              35.0|             1.0|
|              50.0|             2.0|
|              31.0|             1.0|
|              55.0|             2.0|
|              63.0|             3.0|
|              29.0|             1.0|
|              38.0|             1.0|
|               2.0|             0.0|
|              66.0|             3.0|
|              35.0|             1.0|
|              59.0|             2.0|
|              60.0|             3.0|
|              41.0|             2.0|
|              41.0|             2.0|
|              66.0|             3.0|
|              41.0|             2.0|
|               6.0|             0.0|
+------------------+----------------+
only showing top 20 rows
```

```
+----------------+
|selectedFeatures|
+----------------+
|  (1,[0],[50.0])|
|           [17.0]|
|  (1,[0],[41.0])|
|  (1,[0],[35.0])|
|           [50.0]|
|  (1,[0],[31.0])|
|  (1,[0],[55.0])|
|  (1,[0],[63.0])|
|           [29.0]|
|  (1,[0],[38.0])|
|   (1,[0],[2.0])|
|           [66.0]|
|           [35.0]|
|  (1,[0],[59.0])|
|  (1,[0],[60.0])|
|  (1,[0],[41.0])|
|  (1,[0],[41.0])|
|  (1,[0],[66.0])|
|           [41.0]|
|            [6.0]|
+----------------+
only showing top 20 rows
```

STG	SCG	STR	LPR	PEG	UNS	label	features	prediction
0.0	0.0	0.0	0.0	0.0	very_low	0	(5,[],[])	0
0.06	0.06	0.05	0.25	0.33	Low	0	[0.06,0.06,0.05,0...	0
0.1	0.1	0.15	0.65	0.3	Middle	1	[0.1,0.1,0.15,0.6...	0
0.08	0.08	0.08	0.98	0.24	Low	0	[0.08,0.08,0.08,0...	0
0.1	0.1	0.43	0.29	0.56	Middle	1	[0.1,0.1,0.43,0.2...	0
0.15	0.02	0.34	0.4	0.01	very_low	0	[0.15,0.02,0.34,0...	0
0.2	0.14	0.35	0.72	0.25	Low	0	[0.2,0.14,0.35,0....	0
0.06	0.06	0.51	0.41	0.3	Low	0	[0.06,0.06,0.51,0...	0
0.1	0.1	0.52	0.78	0.34	Middle	1	[0.1,0.1,0.52,0.7...	0
0.05	0.07	0.7	0.01	0.05	very_low	0	[0.05,0.07,0.7,0....	0
0.1	0.25	0.1	0.08	0.33	Low	0	[0.1,0.25,0.1,0.0...	0
0.15	0.32	0.05	0.27	0.29	Low	0	[0.15,0.32,0.05,0...	0
0.12	0.28	0.2	0.78	0.2	Low	0	[0.12,0.28,0.2,0....	0
0.18	0.31	0.32	0.42	0.28	Low	0	[0.18,0.31,0.32,0...	0
0.06	0.29	0.35	0.76	0.25	Low	0	[0.06,0.29,0.35,0...	0
0.04	0.28	0.55	0.25	0.1	very_low	0	[0.04,0.28,0.55,0...	0
0.09	0.255	0.6	0.45	0.25	Low	0	[0.09,0.255,0.6,0...	0
0.15	0.295	0.75	0.65	0.24	Low	0	[0.15,0.295,0.75,...	0
0.1	0.256	0.7	0.76	0.16	Low	0	[0.1,0.256,0.7,0....	0
0.06	0.35	0.12	0.43	0.29	Low	0	[0.06,0.35,0.12,0...	0

only showing top 20 rows

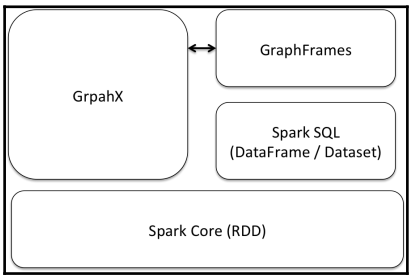
STG	SCG	STR	LPR	PEG	UNS	label	features	prediction
0.08	0.08	0.1	0.24	0.9	High	1	[0.08,0.08,0.1,0....	1
0.09	0.15	0.4	0.1	0.66	Middle	1	[0.09,0.15,0.4,0....	1
0.0	0.0	0.5	0.2	0.85	High	1	[0.0,0.0,0.5,0.2,...	1
0.18	0.18	0.55	0.3	0.81	High	1	[0.18,0.18,0.55,0...	1
0.1	0.1	0.7	0.15	0.9	High	1	[0.1,0.1,0.7,0.15...	1
0.2	0.2	0.7	0.3	0.6	Middle	1	[0.2,0.2,0.7,0.3,...	1
0.12	0.12	0.75	0.35	0.8	High	1	[0.12,0.12,0.75,0...	1
0.2	0.29	0.25	0.49	0.56	Middle	1	[0.2,0.29,0.25,0....	1
0.18	0.3	0.37	0.12	0.66	Middle	1	[0.18,0.3,0.37,0....	1
0.1	0.27	0.31	0.29	0.65	Middle	1	[0.1,0.27,0.31,0....	1
0.09	0.3	0.68	0.18	0.85	High	1	[0.09,0.3,0.68,0....	1
0.08	0.325	0.62	0.94	0.56	High	1	[0.08,0.325,0.62,...	1
0.15	0.275	0.8	0.21	0.81	High	1	[0.15,0.275,0.8,0...	1
0.12	0.245	0.75	0.31	0.59	Middle	1	[0.12,0.245,0.75,...	1
0.18	0.32	0.04	0.19	0.82	High	1	[0.18,0.32,0.04,0...	1
0.2	0.45	0.28	0.31	0.78	High	1	[0.2,0.45,0.28,0....	1
0.2	0.49	0.6	0.2	0.78	High	1	[0.2,0.49,0.6,0.2...	1
0.14	0.49	0.55	0.29	0.6	Middle	1	[0.14,0.49,0.55,0...	1
0.17	0.36	0.8	0.14	0.66	Middle	1	[0.17,0.36,0.8,0....	1
0.1	0.39	0.75	0.31	0.62	Middle	1	[0.1,0.39,0.75,0....	1

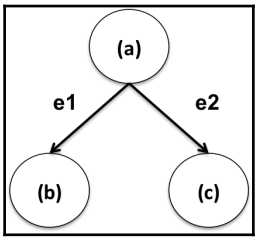
only showing top 20 rows

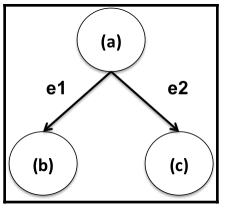
summary	STG	SCG	STR	LPR	PEG
count	211	211	211	211	211
mean	0.310042654028436	0.3123033175355451	0.414597156398104	0.45081042654028425	0.23138862559241719
stddev	0.18270580849465212	0.19447754121739496	0.24365901973174997	0.25108964835256975	0.11058937632395373
min	0.0	0.0	0.0	0.0	0.0
max	0.77	0.85	0.88	0.99	0.56

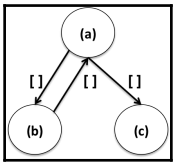
summary	STG	SCG	STR	LPR	PEG
count	192	192	192	192	192
mean	0.4005052083333334	0.40389583333333334	0.5049739583333334	0.4099479166666666	0.7035937499999997
stddev	0.2315183888390661	0.2275363375463664	0.24187395434292872	0.26345369740481017	0.13779528621524523
min	0.0	0.0	0.02	0.01	0.32
max	0.99	0.9	0.95	0.99	0.99

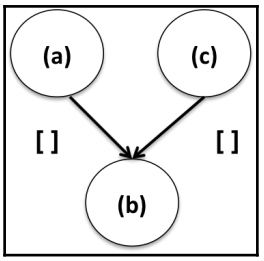
Chapter 7:
Using Spark SQL in Graph Applications

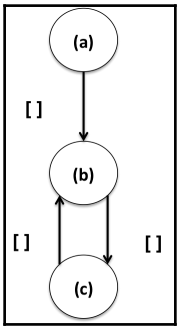


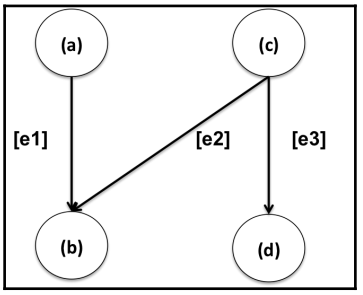













```
+--------+-------------+------+--------+-------------+---------------+--------+
|       a|           e1|     b|       c|           e2|             e3|       d|
+--------+-------------+------+--------+-------------+---------------+--------+
|[365079]|[365079,8742]|[8742]|[109254]|[109254,8742]|[109254,100010]|[100010]|
|[241393]|[241393,8742]|[8742]|[109254]|[109254,8742]|[109254,100010]|[100010]|
| [33284]| [33284,8742]|[8742]|[109254]|[109254,8742]|[109254,100010]|[100010]|
|[198072]|[198072,8742]|[8742]|[109254]|[109254,8742]|[109254,100010]|[100010]|
|[203728]|[203728,8742]|[8742]|[109254]|[109254,8742]|[109254,100010]|[100010]|
+--------+-------------+------+--------+-------------+---------------+--------+
only showing top 5 rows
```

[0771044445,0,null,null,null,null,null,null,null,null,null]

[0827229534,1,[5.0,2,2],2,WrappedArray([|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Clergy[12360]|Preaching[12368]],
[|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Clergy[12360]|Sermons[12370]]),Book,WrappedArray(
[[A2JW67OY8U6HHK,2000-7-28,9,5,10]], [[A2VE83MZF98ITY,2003-12-
14,5,5,6]]),396585,WrappedArray([0804215715], [156101074X], [0687023955],
[0687074231], [082721619X]),5,Patterns of Preaching: A Sermon Sampler]

[0738700797,2,[4.5,12,12],2,WrappedArray([|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Earth-Based Religions[12472]|Wicca[12484]],
[|Books[283155]|Subjects[1000]|Religion & Spirituality[22]|Earth-Based
Religions[12472]|Witchcraft[12486]]),Book,WrappedArray([[A11NCO6YTE4BTJ,2001-12-
16,4,5,5]], [[A9CQ3PLRNIR83,2002-1-7,5,4,5]], [[A13SG9ACZ9O5IM,2002-1-24,8,5,8]],
[[A1BDAI6VEYMAZA,2002-1-28,4,5,4]], [[A2P6KAWXJ16234,2002-2-6,16,4,16]],
[[AMACWC3M7PQFR,2002-2-14,5,4,5]], [[A3GO7UV9XX14D8,2002-3-23,6,4,6]],
[[A1GIL64QK68WKL,2002-5-23,8,5,8]], [[AEOBOF2ONQJWV,2003-2-25,5,5,8]],
[[A3IGHTES8ME05L,2003-11-25,5,5,5]], [[A1CP26N8RHYVVO,2004-2-11,9,1,13]],
[[ANEIANH0WAT9D,2005-2-7,1,5,1]]),168596,WrappedArray([0738700827], [1567184960],
[1567182836], [0738700525], [0738700940]),5,Candlemas: Feast of Flames]

[0486287785,3,[5.0,1,1],1,WrappedArray([|Books[283155]|Subjects[1000]|Home &
Garden[48]|Crafts &
Hobbies[5126]|General[5144]]),Book,WrappedArray([[A3IDGASRQAW8B2,2003-7-
10,2,5,2]]),1270652,null,0,World War II Allied Fighter Planes Trading Cards]

[0842328327,4,[4.0,1,1],5,WrappedArray([|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Reference[172810]|Commentaries[12155]|New
Testament[12159]], [|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Christian Living[12333]|Discipleship[12335]],
[|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Bibles[12059]|Translations[764432]|Life
Application[572080]], [|Books[283155]|Subjects[1000]|Religion & Spirituality[22]|Bible
& Other Sacred Texts[12056]|Bible[764430]|New Testament[572082]],
[|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Bibles[12059]|Study Guides, History &
Reference[764438]|General[572094]]),Book,WrappedArray([[A2591BUPXCS705,2004-8-
19,1,4,1]]),631289,WrappedArray([0842328130], [0830818138], [0842330313],
[0842328610], [0842328572]),5,Life Application Bible Commentary: 1 and 2 Timothy and
Titus]

a	e	b	e2
[6301798643,93678...	[6301798643,63017...	[6301797973,51638...	[6301797973,63017...
[B00000J2I0,38585...	[B00000J2I0,63052...	[6305242143,34489...	[6305242143,B0000...
[0451524934,44922...	[0451524934,03167...	[0316769487,98756...	[0316769487,04515...
[6300185788,11281...	[6300185788,15588...	[1558807225,20377...	[1558807225,63001...
[0156027321,62424...	[0156027321,01420...	[0142001740,33797...	[0142001740,01560...
[6303182135,46688...	[6303182135,63031...	[6303182232,34106...	[6303182232,63031...
[6304424841,53266...	[6304424841,63034...	[6303499988,66050...	[6303499988,63044...
[6302728657,24467...	[6302728657,63016...	[6301627024,28339...	[6301627024,63027...
[0316769487,98756...	[0316769487,04515...	[0451524934,44922...	[0451524934,03167...
[6304401744,46640...	[6304401744,63044...	[6304424841,53266...	[6304424841,63044...
[0345342968,29543...	[0345342968,04515...	[0451524934,44922...	[0451524934,03453...
[6303182232,34106...	[6303182232,63031...	[6303182135,46688...	[6303182135,63031...
[B00004S36U,37060...	[B00004S36U,B0000...	[B00004S36S,37060...	[B00004S36S,B0000...
[0385504209,296,T...	[0385504209,06710...	[0671027360,37685...	[0671027360,03855...
[0385504209,296,T...	[0385504209,06710...	[0671027387,42470...	[0671027387,03855...
[1880685000,47585...	[1880685000,15804...	[1580420826,45175...	[1580420826,18806...
[0671027387,42470...	[0671027387,03855...	[0385504209,296,T...	[0385504209,06710...
[6301627024,28339...	[6301627024,63027...	[6302728657,24467...	[6302728657,63016...
[6304424841,53266...	[6304424841,63044...	[6304401744,46640...	[6304401744,63044...
[0142001740,33797...	[0142001740,01560...	[0156027321,62424...	[0156027321,01420...

only showing top 20 rows

a	e	b	e2
[6301798643,93678...	[6301798643,63017...	[6301797973,51638...	[6301797973,63017...
[B00000J2I0,38585...	[B00000J2I0,63052...	[6305242143,34489...	[6305242143,B0000...
[6300185788,11281...	[6300185788,15588...	[1558807225,20377...	[1558807225,63001...
[6303182135,46688...	[6303182135,63031...	[6303182232,34106...	[6303182232,63031...
[6304424841,53266...	[6304424841,63034...	[6303499988,66050...	[6303499988,63044...
[6302728657,24467...	[6302728657,63016...	[6301627024,28339...	[6301627024,63027...
[0316769487,98756...	[0316769487,04515...	[0451524934,44922...	[0451524934,03167...
[0345342968,29543...	[0345342968,04515...	[0451524934,44922...	[0451524934,03453...
[6303182232,34106...	[6303182232,63031...	[6303182135,46688...	[6303182135,63031...
[B00004S36U,37060...	[B00004S36U,B0000...	[B00004S36S,37060...	[B00004S36S,B0000...
[6301627024,28339...	[6301627024,63027...	[6302728657,24467...	[6302728657,63016...
[6304424841,53266...	[6304424841,63044...	[6304401744,46640...	[6304401744,63044...
[B00004S36S,37060...	[B00004S36S,B0000...	[B00004S36U,37060...	[B00004S36U,B0000...
[1580420826,45175...	[1580420826,18806...	[1880685000,47585...	[1880685000,15804...
[6305242143,34489...	[6305242143,B0000...	[B00000J2I0,38585...	[B00000J2I0,63052...
[B0000508U6,50505...	[B0000508U6,B0000...	[B000069AUI,53688...	[B000069AUI,B0000...
[B00008NG5V,27054...	[B00008NG5V,B0000...	[B00005V8PZ,14861...	[B00005V8PZ,B0000...
[0316346624,89000...	[0316346624,00666...	[0066620996,15485...	[0066620996,03163...
[B00005V8PZ,14861...	[B00005V8PZ,B0000...	[B00008NG5V,27054...	[B00008NG5V,B0000...
[B000069AUI,53688...	[B000069AUI,B0000...	[B0000508U6,50505...	[B0000508U6,B0000...

[0771044445,0,null,null,null,null,null,null,null,null,null]

[0827229534,1,Patterns of Preaching: A Sermon
Sampler,[5.0,2,2],2,WrappedArray([|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Clergy[12360]|Preaching[12368]],
[|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Clergy[12360]|Sermons[12370]]),Book,WrappedArray(
[[A2JW67OY8U6HHK,2000-7-28,9,5,10]], [[A2VE83MZF98ITY,2003-12-
14,5,5,6]]),396585,WrappedArray([0804215715], [156101074X], [0687023955],
[0687074231], [082721619X]),5]

[0738700797,2,Candlemas: Feast of
Flames,[4.5,12,12],2,WrappedArray([|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Earth-Based Religions[12472]|Wicca[12484]],
[|Books[283155]|Subjects[1000]|Religion & Spirituality[22]|Earth-Based
Religions[12472]|Witchcraft[12486]]),Book,WrappedArray([[A11NCO6YTE4BTJ,2001-12-
16,4,5,5]], [[A9CQ3PLRNIR83,2002-1-7,5,4,5]], [[A13SG9ACZ9O5IM,2002-1-24,8,5,8]],
[[A1BDAI6VEYMAZA,2002-1-28,4,5,4]], [[A2P6KAWXJ16234,2002-2-6,16,4,16]],
[[AMACWC3M7PQFR,2002-2-14,5,4,5]], [[A3GO7UV9XX14D8,2002-3-23,6,4,6]],
[[A1GIL64QK68WKL,2002-5-23,8,5,8]], [[AEOBOF2ONQJWV,2003-2-25,5,5,8]],
[[A3IGHTES8ME05L,2003-11-25,5,5,5]], [[A1CP26N8RHYVVO,2004-2-11,9,1,13]],
[[ANEIANH0WAT9D,2005-2-7,1,5,1]]),168596,WrappedArray([0738700827], [1567184960],
[1567182836], [0738700525], [0738700940]),5]

[0486287785,3,World War II Allied Fighter Planes Trading
Cards,[5.0,1,1],1,WrappedArray([|Books[283155]|Subjects[1000]|Home & Garden[48]|Crafts
& Hobbies[5126]|General[5144]]),Book,WrappedArray([[A3IDGASRQAW8B2,2003-7-
10,2,5,2]]),1270652,null,0]

[0842328327,4,Life Application Bible Commentary: 1 and 2 Timothy and
Titus,[4.0,1,1],5,WrappedArray([|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Reference[172810]|Commentaries[12155]|New
Testament[12159]], [|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Christian Living[12333]|Discipleship[12335]],
[|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Bibles[12059]|Translations[764432]|Life
Application[572080]], [|Books[283155]|Subjects[1000]|Religion & Spirituality[22]|Bible
& Other Sacred Texts[12056]|Bible[764430]|New Testament[572082]],
[|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Bibles[12059]|Study Guides, History &
Reference[764438]|General[572094]]),Book,WrappedArray([[A2591BUPXCS705,2004-8-
19,1,4,1]]),631289,WrappedArray([0842328130], [0830818138], [0842330313],
[0842328610], [0842328572]),5]

```
[[0827229534,1,Patterns of Preaching: A Sermon
Sampler,[5.0,2,2],2,WrappedArray([|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Clergy[12360]|Preaching[12368]],
[|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Clergy[12360]|Sermons[12370]]),Book,WrappedArray(
[[A2JW67OY8U6HHK,2000-7-28,9,5,10]], [[A2VE83MZF98ITY,2003-12-
14,5,5,6]]),396585,WrappedArray([0804215715], [156101074X], [0687023955],
[0687074231], [082721619X]),5],[0827229534,1,Patterns of Preaching: A Sermon
Sampler,[5.0,2,2],2,WrappedArray([|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Clergy[12360]|Preaching[12368]],
[|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Christianity[12290]|Clergy[12360]|Sermons[12370]]),Book,WrappedArray(
[[A2JW67OY8U6HHK,2000-7-28,9,5,10]], [[A2VE83MZF98ITY,2003-12-
14,5,5,6]]),396585,WrappedArray([0804215715], [156101074X], [0687023955],
[0687074231], [082721619X]),5]]

[[0738700797,2,Candlemas: Feast of
Flames,[4.5,12,12],2,WrappedArray([|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Earth-Based Religions[12472]|Wicca[12484]],
[|Books[283155]|Subjects[1000]|Religion & Spirituality[22]|Earth-Based
Religions[12472]|Witchcraft[12486]]),Book,WrappedArray([[A11NCO6YTE4BTJ,2001-12-
16,4,5,5]], [[A9CQ3PLRNIR83,2002-1-7,5,4,5]], [[A13SG9ACZ9O5IM,2002-1-24,8,5,8]],
[[A1BDAI6VEYMAZA,2002-1-28,4,5,4]], [[A2P6KAWXJ16234,2002-2-6,16,4,16]],
[[AMACWC3M7PQFR,2002-2-14,5,4,5]], [[A3GO7UV9XX14D8,2002-3-23,6,4,6]],
[[A1GIL64QK68WKL,2002-5-23,8,5,8]], [[AEOBOF2ONQJWV,2003-2-25,5,5,8]],
[[A3IGHTES8ME05L,2003-11-25,5,5,5]], [[A1CP26N8RHYVVO,2004-2-11,9,1,13]],
[[ANEIANH0WAT9D,2005-2-7,1,5,1]]),168596,WrappedArray([0738700827], [1567184960],
[1567182836], [0738700525], [0738700940]),5],[0738700797,2,Candlemas: Feast of
Flames,[4.5,12,12],2,WrappedArray([|Books[283155]|Subjects[1000]|Religion &
Spirituality[22]|Earth-Based Religions[12472]|Wicca[12484]],
[|Books[283155]|Subjects[1000]|Religion & Spirituality[22]|Earth-Based
Religions[12472]|Witchcraft[12486]]),Book,WrappedArray([[A11NCO6YTE4BTJ,2001-12-
16,4,5,5]], [[A9CQ3PLRNIR83,2002-1-7,5,4,5]], [[A13SG9ACZ9O5IM,2002-1-24,8,5,8]],
[[A1BDAI6VEYMAZA,2002-1-28,4,5,4]], [[A2P6KAWXJ16234,2002-2-6,16,4,16]],
[[AMACWC3M7PQFR,2002-2-14,5,4,5]], [[A3GO7UV9XX14D8,2002-3-23,6,4,6]],
[[A1GIL64QK68WKL,2002-5-23,8,5,8]], [[AEOBOF2ONQJWV,2003-2-25,5,5,8]],
[[A3IGHTES8ME05L,2003-11-25,5,5,5]], [[A1CP26N8RHYVVO,2004-2-11,9,1,13]],
[[ANEIANH0WAT9D,2005-2-7,1,5,1]]),168596,WrappedArray([0738700827], [1567184960],
[1567182836], [0738700525], [0738700940]),5]]
```



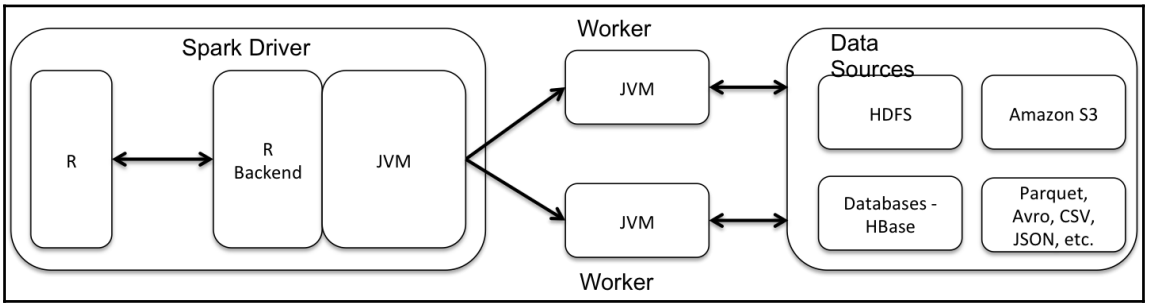
```
[1857444000,Book,Art of Attack in Chess,19.15963251427946]
[1880673851,Book,My System: 21st Century Edition,18.07328127614439]
[0812917561,Book,Ideas Behind the Chess Openings : Algebraic Edition
(Chess),17.77792169452039]
[096290497X,Book,Discerning of Spirits,17.64175373421938]
[0345457684,Book,Altered Carbon,16.11861824590565]
[0793521610,Book,Building a Jazz Vocabulary,14.999654656171334]
[0812579844,Book,The Golden Age (The Golden Age, Book 1),14.91395417693351]
[0553382136,Book,Spin State,14.641071672237276]
[0684135051,Book,Knitting Without Tears : Basic Techniques and Easy-to-Follow
Directions for Garments to Fit All Sizes (Knitting Without Tears SL
466),14.397031132425692]
[0884893723,Book,Understanding Catholic Christianity,13.957694196742478]
```

```
== Physical Plan ==
*Project [ASIN#0 AS src#37, dst#40]
+- Generate explode(similarLines#8.similar), true, false, [dst#40]
   +- *Project [ASIN#0, similarLines#8]
      +- *Filter (isnotnull(salerank#7L) && (salerank#7L < 100))
         +- *FileScan json [ASIN#0,salerank#7L,similarLines#8] Batched: false, Format:
JSON, Location: InMemoryFileIndex[file:/Users/aurobindosarkar/Downloads/input.json],
PartitionFilters: [], PushedFilters: [IsNotNull(salerank), LessThan(salerank,100)],
ReadSchema:
struct<ASIN:string,salerank:bigint,similarLines:array<struct<similar:string>>>
```

```
0771044445,unknown
0827229534,Book
0738700797,Book
0486287785,Book
0842328327,Book
1577943082,Book
0486220125,Book
B00000AU3R,Music
0231118597,Book
1859677800,Book
0375709363,Book
0871318237,Book
1590770218,Book
0313230269,Book
B00004W1W1,Music
.
.
.
B000053SPI,Music
1552126463,Book
1885588437,Book
1898323496,Book
0764224115,Book
0451409221,Book
1585674338,Book
B00004YR7L,Music
087793486X,Book
0793554098,Book
B00004YR7I,Music
0253214882,Book
```

0827229534,Book
0738700797,Book
0486287785,Book
0842328327,Book
1577943082,Book
0486220125,Book
0231118597,Book
1859677800,Book
0375709363,Book
0871318237,Book
1590770218,Book
0313230269,Book
1559362022,Book
0195110382,Book
0849311012,Book
078510870X,Book
3895780812,Book
0393049388,Book
0553525476,Book
.
.
.
0310208084,Book
0006176909,Book
0786632550,Book
0752808990,Book
4770023286,Book
0838409342,Book
0552139432,Book
B00006RGI2,Book
B000067JZT,Book
B00008MNUJ,Book
0060537612,Book
9700507734,Book
9627762644,Book
0970020503,Book
1930519206,Book
0879736836,Book

Chapter 8: Using Spark SQL with SparkR




```
root
 |-- age: integer (nullable = true)
 |-- job: string (nullable = true)
 |-- marital: string (nullable = true)
 |-- education: string (nullable = true)
 |-- default: string (nullable = true)
 |-- housing: string (nullable = true)
 |-- loan: string (nullable = true)
 |-- contact: string (nullable = true)
 |-- month: string (nullable = true)
 |-- day_of_week: string (nullable = true)
 |-- duration: integer (nullable = true)
 |-- campaign: integer (nullable = true)
 |-- pdays: integer (nullable = true)
 |-- previous: integer (nullable = true)
 |-- poutcome: string (nullable = true)
 |-- emp.var.rate: double (nullable = true)
 |-- cons.price.idx: double (nullable = true)
 |-- cons.conf.idx: double (nullable = true)
 |-- euribor3m: double (nullable = true)
 |-- nr.employed: double (nullable = true)
 |-- y: string (nullable = true)
```

```
 [1] "age"            "job"            "marital"        "education"
 [5] "default"        "housing"        "loan"           "contact"
 [9] "month"          "day_of_week"    "duration"       "campaign"
[13] "pdays"          "previous"       "poutcome"       "emp.var.rate"
[17] "cons.price.idx" "cons.conf.idx"  "euribor3m"      "nr.employed"
[21] "y"              "durationMins"
```

```
'SparkDataFrame': 21 variables:
 $ age           : int 56 57 37 40 56 45
 $ job           : chr "housemaid" "services" "services" "admin." "services" "services"
 $ marital       : chr "married" "married" "married" "married" "married" "married"
 $ education     : chr "basic.4y" "high.school" "high.school" "basic.6y" "high.school" "basic.9y"
 $ default       : chr "no" "unknown" "no" "no" "no" "unknown"
 $ housing       : chr "no" "no" "yes" "no" "no" "no"
 $ loan          : chr "no" "no" "no" "no" "yes" "no"
 $ contact       : chr "telephone" "telephone" "telephone" "telephone" "telephone" "telephone"
 $ month         : chr "may" "may" "may" "may" "may" "may"
 $ day_of_week   : chr "mon" "mon" "mon" "mon" "mon" "mon"
 $ duration      : int 261 149 226 151 307 198
 $ campaign      : int 1 1 1 1 1 1
 $ pdays         : int 999 999 999 999 999 999
 $ previous      : int 0 0 0 0 0 0
 $ poutcome      : chr "nonexistent" "nonexistent" "nonexistent" "nonexistent" "nonexistent"
"nonexistent"
 $ emp.var.rate  : num 1.1 1.1 1.1 1.1 1.1 1.1
 $ cons.price.idx: num 93.994 93.994 93.994 93.994 93.994 93.994
 $ cons.conf.idx : num -36.4 -36.4 -36.4 -36.4 -36.4 -36.4
 $ euribor3m     : num 4.857 4.857 4.857 4.857 4.857 4.857
 $ nr.employed   : num 5191 5191 5191 5191 5191 5191
 $ y             : chr "no" "no" "no" "no" "no" "no"
```

```
age       job marital   education default housing loan   contact month
1  56 housemaid married    basic.4y      no      no   no telephone   may
2  57  services married high.school unknown      no   no telephone   may
  day_of_week duration campaign pdays previous    poutcome emp.var.rate
1         mon      261        1   999        0 nonexistent          1.1
2         mon      149        1   999        0 nonexistent          1.1
  cons.price.idx cons.conf.idx euribor3m nr.employed  y
1         93.994         -36.4     4.857        5191 no
2         93.994         -36.4     4.857        5191 no
```


```
age       job marital   education default housing loan   contact month
1  56 housemaid married    basic.4y      no      no   no telephone   may
2  57  services married high.school unknown      no   no telephone   may
  day_of_week duration campaign pdays previous    poutcome emp.var.rate
1         mon      261        1   999        0 nonexistent          1.1
2         mon      149        1   999        0 nonexistent          1.1
  cons.price.idx cons.conf.idx euribor3m nr.employed  y
1         93.994         -36.4     4.857        5191 no
2         93.994         -36.4     4.857        5191 no
```


```
age       job  marital education default housing loan   contact month
1  56 housemaid  married  basic.4y      no      no   no telephone   may
2  57 housemaid divorced  basic.4y      no     yes   no telephone   may
  day_of_week duration campaign pdays previous    poutcome emp.var.rate
1         mon      261        1   999        0 nonexistent          1.1
2         mon      293        1   999        0 nonexistent          1.1
  cons.price.idx cons.conf.idx euribor3m nr.employed  y
1         93.994         -36.4     4.857        5191 no
2         93.994         -36.4     4.857        5191 no
```

```
   marital count
1  unknown     6
2 divorced   489
3  married  3228
4   single   453
```

```
age       job marital   education default housing loan   contact month
1  56 housemaid married    basic.4y      no      no   no telephone   may
2  57  services married high.school unknown      no   no telephone   may
  day_of_week duration campaign pdays previous    poutcome emp.var.rate
1         mon      261        1   999        0 nonexistent          1.1
2         mon      149        1   999        0 nonexistent          1.1
  cons.price.idx cons.conf.idx euribor3m nr.employed  y durationMins
1         93.994         -36.4     4.857        5191 no            4
2         93.994         -36.4     4.857        5191 no            2
```

```
    education age marital housing loan
1    basic.9y  19  single     yes   no
2 high.school  18  single      no   no
3 high.school  18  single     yes  yes
4    basic.9y  19  single     yes   no
5    basic.6y  19  single      no   no
6    basic.9y  19  single      no   no
```

```
> crimesStatesdf <- withColumnRenamed(crimesStatesSubset, "_c0", "comm") %>%
withColumnRenamed("_c1", "code") %>% withColumnRenamed("_c1", "st")  %>%
withColumnRenamed("_c129", "nmurders")  %>% withColumnRenamed("_c131", "nrapes")  %>%
withColumnRenamed("_c133", "nrobberies")  %>% withColumnRenamed("_c135", "nassaults")
%>% withColumnRenamed("_c137", "nburglaries")  %>% withColumnRenamed("_c139",
"nlarcenies")  %>% withColumnRenamed("_c141", "nautothefts")  %>%
withColumnRenamed("_c143", "narsons")
```

```
                    comm code nmurders nrapes nrobberies nassaults nburglaries
1 BerkeleyHeightstownship   NJ        0      0          1         4          14
2          Marpletownship   PA        0      1          5        24          57
  nlarcenies nautothefts narsons
1        138          16       2
2        376          26       1
```

```
          comm code_x nmurders nrapes nrobberies nassaults nburglaries
1 Anchoragecity     AK       23    212        568      1410        1880
2    Juneaucity     AK        0     12          3        18         128
  nlarcenies nautothefts narsons st   name code_y
1      10660        1387     105  2 Alaska     AK
2        857          59       7  2 Alaska     AK
```

```
          comm code_x nmurders nrapes nrobberies nassaults nburglaries
1 Anchoragecity     AK       23    212        568      1410        1880
2    Juneaucity     AK        0     12          3        18         128
  nlarcenies nautothefts narsons st   name code_y
1      10660        1387     105  2 Alaska     AK
2        857          59       7  2 Alaska     AK
```

```
> library(magrittr)
> usPath <- "file:///Users/aurobindosarkar/Downloads/Tennis-Major-Tournaments-Match-Statistics/USOpen-women-2013.csv"
> usdf <- read.df(usPath, "csv", header = "true", inferSchema = "true", na.strings = "NA", delimiter= ",")
> ussubdf <- select(usdf, "Player 1", "Player 2", "ROUND", "Result")%>% withColumnRenamed("Player 1", "p1") %>%
withColumnRenamed("Player 2", "p2")
> showDF(ussubdf, 2)
+----------+----------+-----+------+
|       p1|       p2|ROUND|Result|
+----------+----------+-----+------+
|S Williams|V Azarenka|    7|     1|
|F Pennetta|V Azarenka|    6|     0|
+----------+----------+-----+------+
only showing top 2 rows
> wimPath <- "file:///Users/aurobindosarkar/Downloads/Tennis-Major-Tournaments-Match-Statistics/Wimbledon-women-2013.csv"
> wimdf <- read.df(wimPath, "csv", header = "true", inferSchema = "true", na.strings = "NA", delimiter= ",")
> wimsubdf <- select(usdf, "Player 1", "Player 2", "ROUND", "Result")%>% withColumnRenamed("Player 1", "p1") %>%
withColumnRenamed("Player 2", "p2")
> showDF(wimsubdf, 2)
+----------+----------+-----+------+
|       p1|       p2|ROUND|Result|
+----------+----------+-----+------+
|S Williams|V Azarenka|    7|     1|
|F Pennetta|V Azarenka|    6|     0|
+----------+----------+-----+------+
only showing top 2 rows
```

We can append the rows of wimsubdf to ussubdf using the rbind operation as shown below.

```
> df1 <- rbind(ussubdf, wimsubdf)
> showDF(df1, 2)
+----------+----------+-----+------+
|       p1|       p2|ROUND|Result|
+----------+----------+-----+------+
|S Williams|V Azarenka|    7|     1|
|F Pennetta|V Azarenka|    6|     0|
+----------+----------+-----+------+
only showing top 2 rows
```

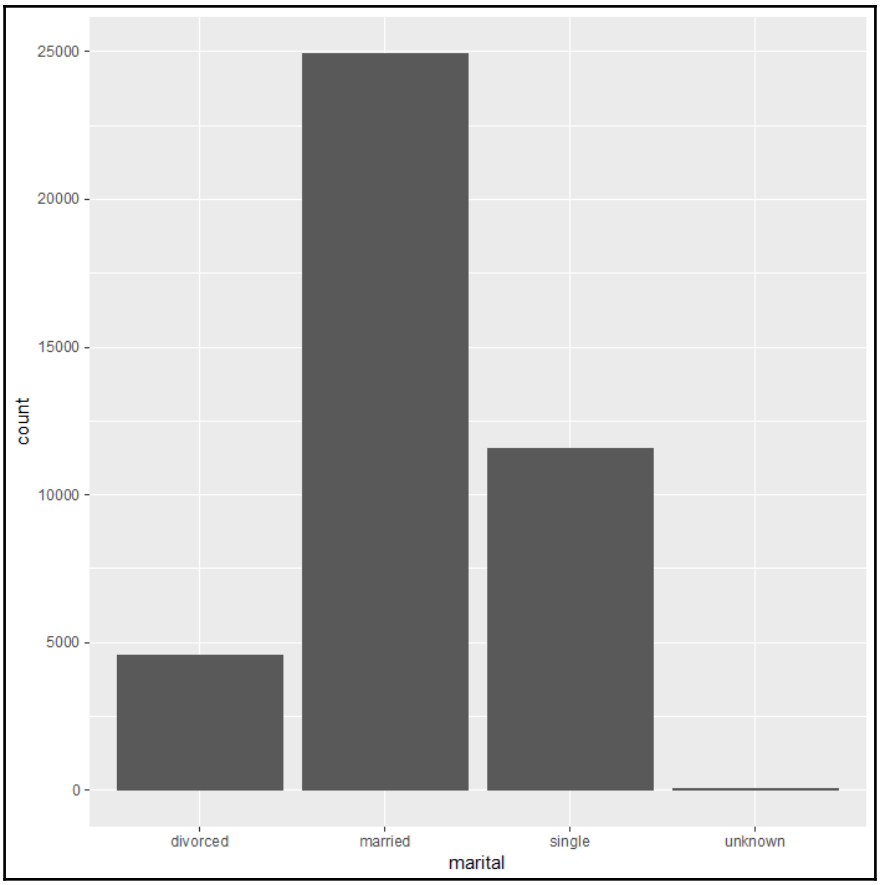
```
  duration  durMins
1      261 4.350000
2      149 2.483333
3      226 3.766667
4      151 2.516667
5      307 5.116667
6      198 3.300000
```

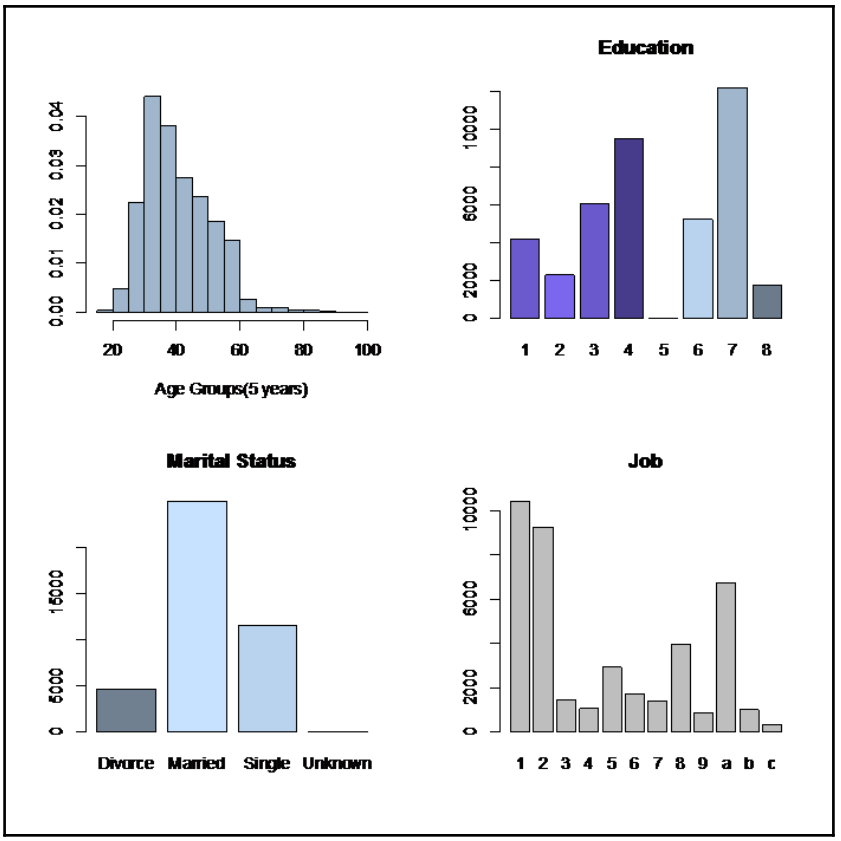
```
  age maxDuration
1  33        4918
2  52        4199
3  27        3785
4  31        3643
5  37        3631
6  28        3509
```

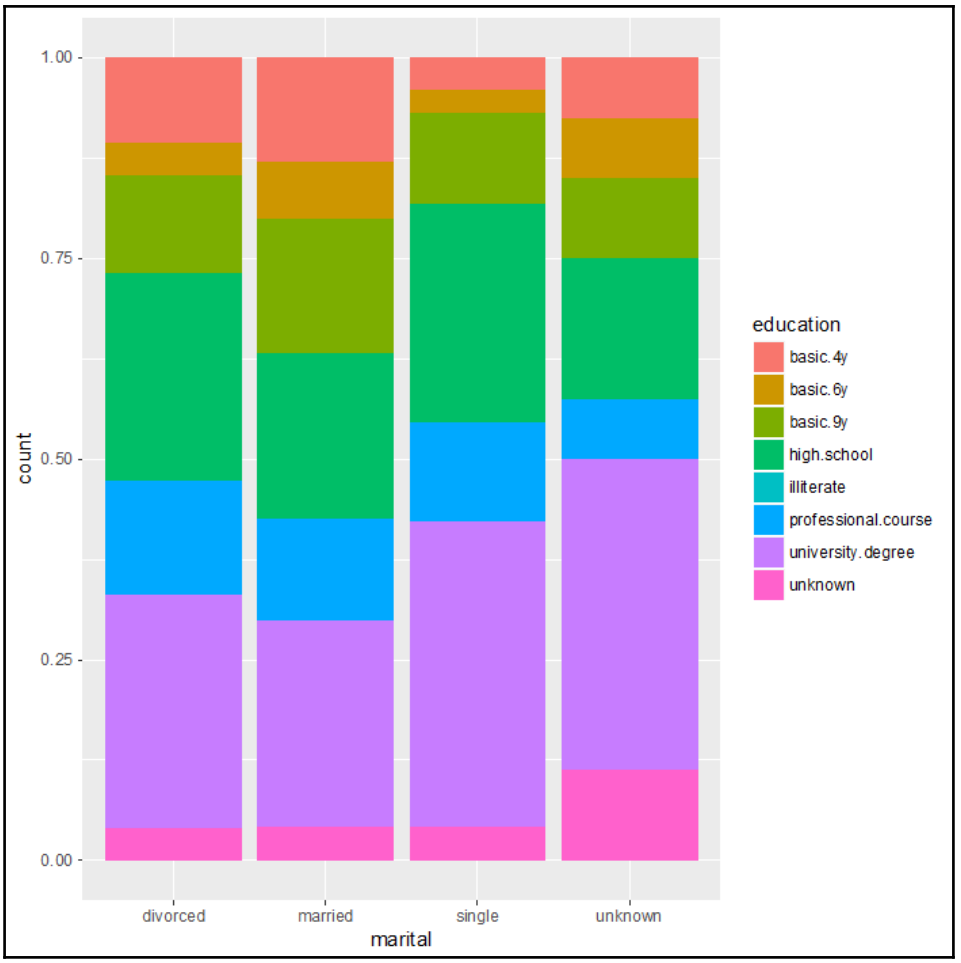
summary	duration	campaign	previous	age
count	41188	41188	41188	41188
mean	258.2850101971448	2.567592502670681	0.17296299893172767	40.02406040594348
stddev	259.27924883646455	2.770013542902331	0.49490107983928927	10.421249980934057
min	0	1	0	17
max	4918	56	7	98

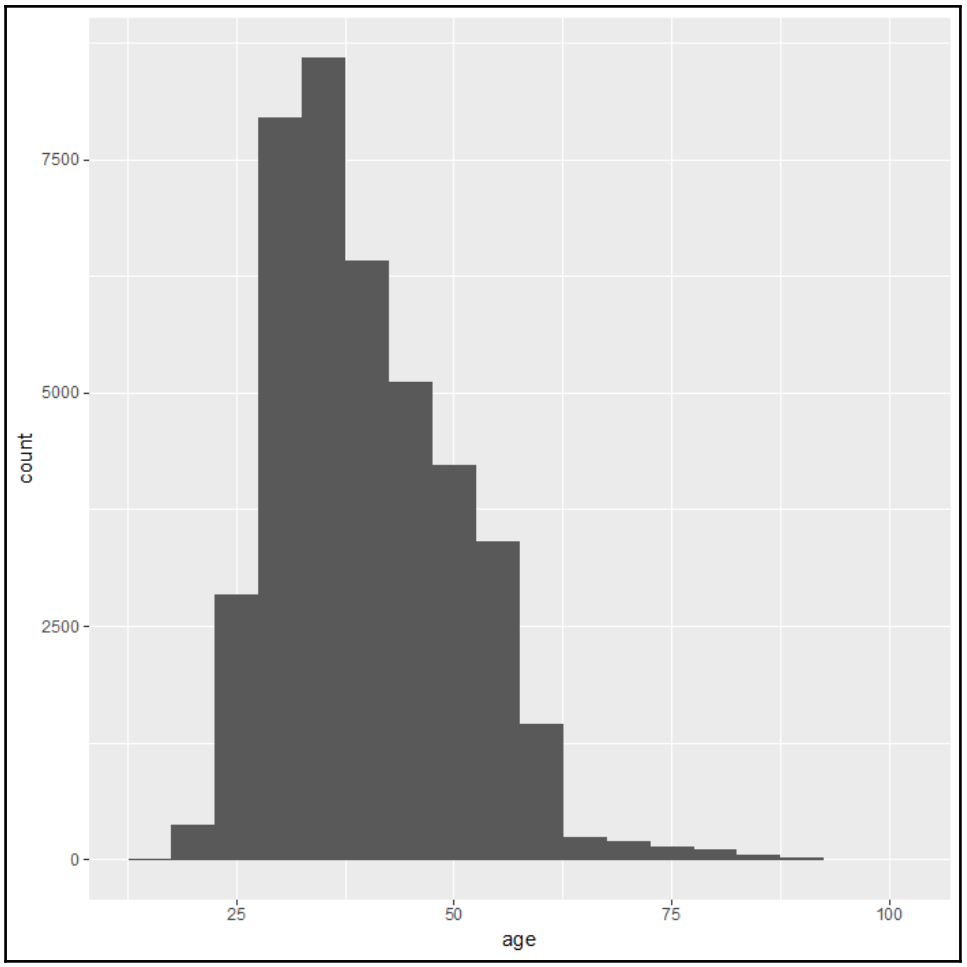
job	Count	Percentage
management	2924	7.099155093716616
retired	1720	4.175973584539186
unknown	330	0.8012042342429834
self-employed	1421	3.4500339904826647
student	875	2.1244051665533648
blue-collar	9254	22.46770904146839
entrepreneur	1456	3.5350101971447994
admin.	10422	25.303486452364766
technician	6743	16.37127318636496
services	3969	9.636301835486064
housemaid	1060	2.5735651160532194
unemployed	1014	2.4618821015829853

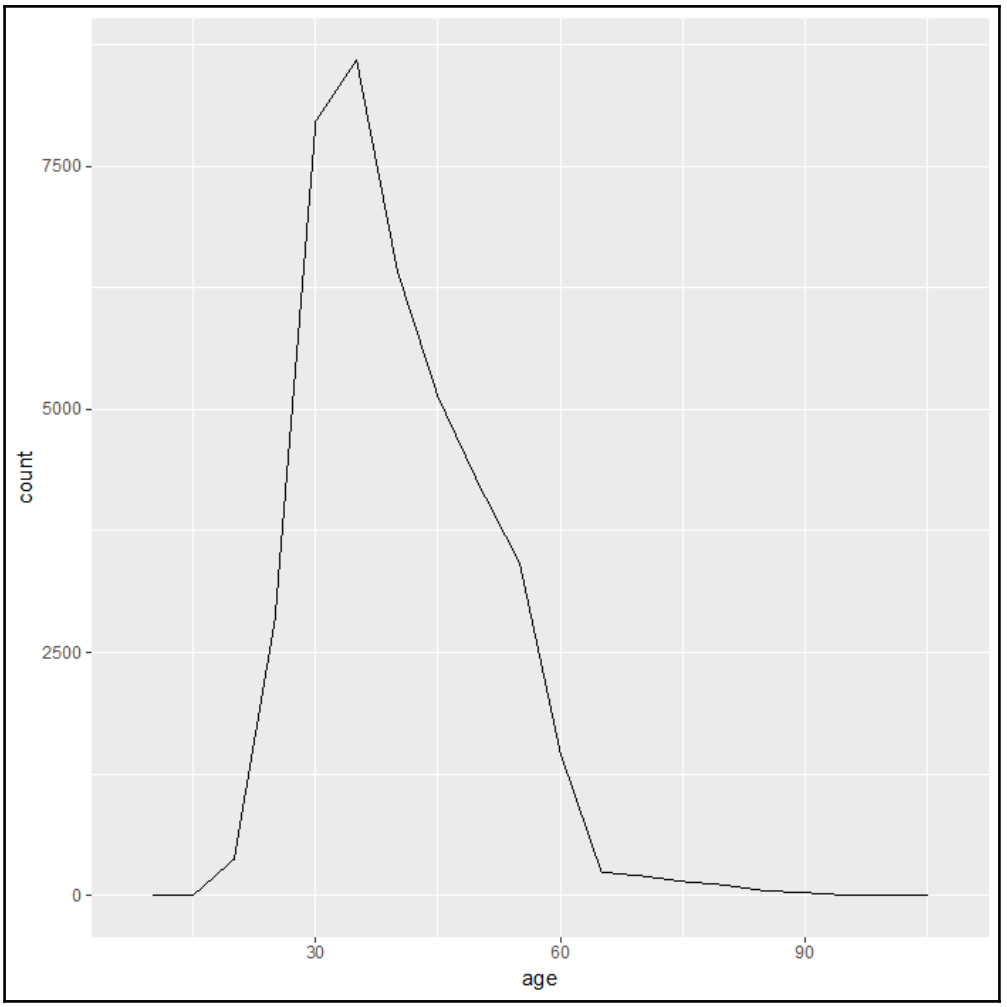
	job_marital	divorced	married	single	unknown
1	housemaid	161	777	119	3
2	services	532	2294	1137	6
3	self-employed	133	904	379	5
4	student	9	41	824	1
5	retired	348	1274	93	5
6	unknown	13	234	74	9
7	admin.	1280	5253	3875	14
8	blue-collar	728	6687	1825	14
9	technician	774	3670	2287	12
10	entrepreneur	179	1071	203	3
11	management	331	2089	501	3
12	unemployed	124	634	251	5

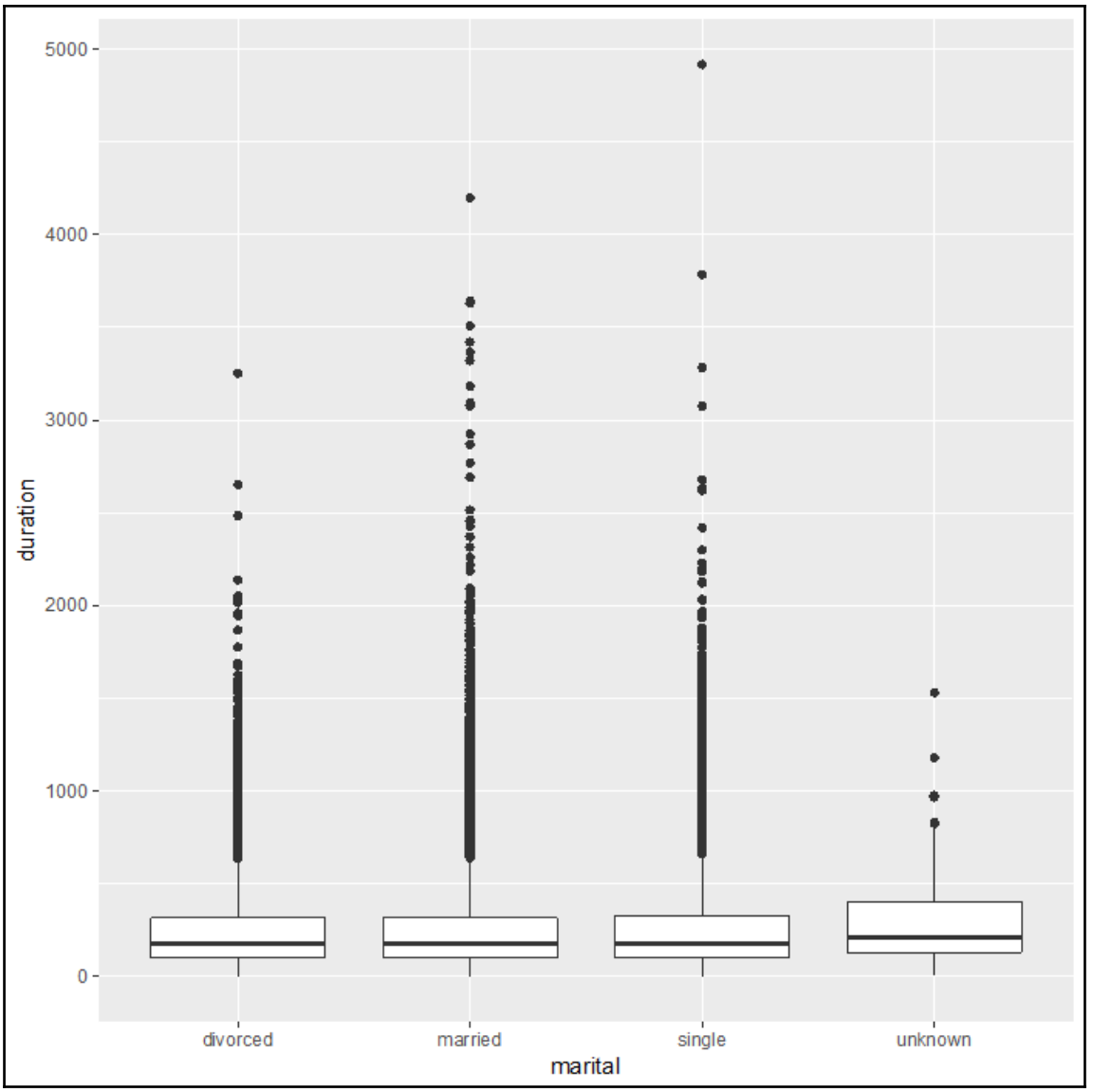

25000
20000
15000
10000
5000
0
count
divorced
married
single
unknown
marital

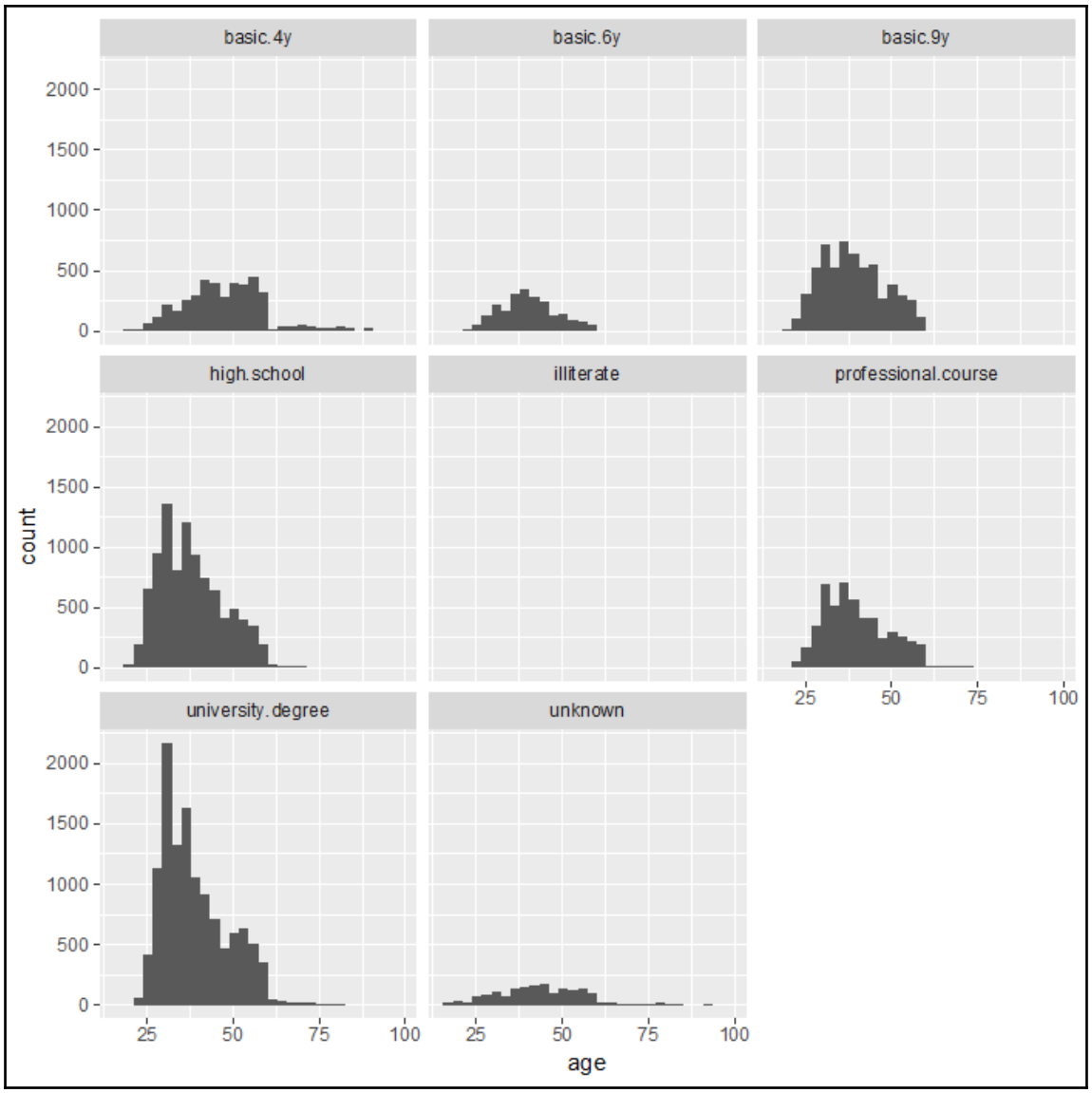

0.00
0.01
0.02
0.03
0.04
20
40
60
80
100
Age Groups(5 years)
Education
0
2000
6000
10000
1
2
3
4
5
6
7
8
Marital Status
0
5000
15000
Divorce
Married
Single
Unknown
Job
0
2000
6000
10000
1
2
3
4
5
6
7
8
9
a
b
c

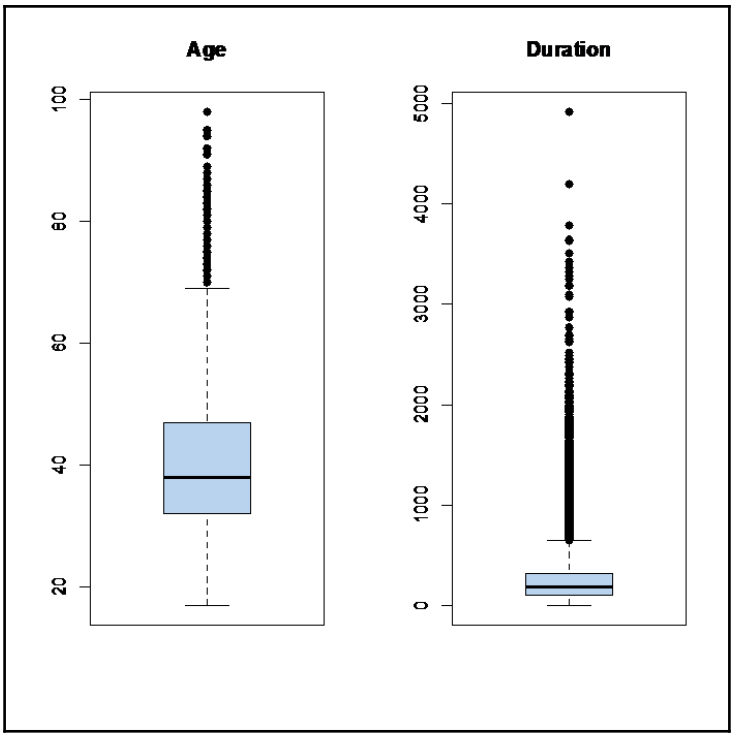

1.00
0.75
0.50
0.25
0.00
count
divorced
married
single
unknown
marital
education
basic.4y
basic.6y
basic.9y
high.school
illiterate
professional.course
university.degree
unknown

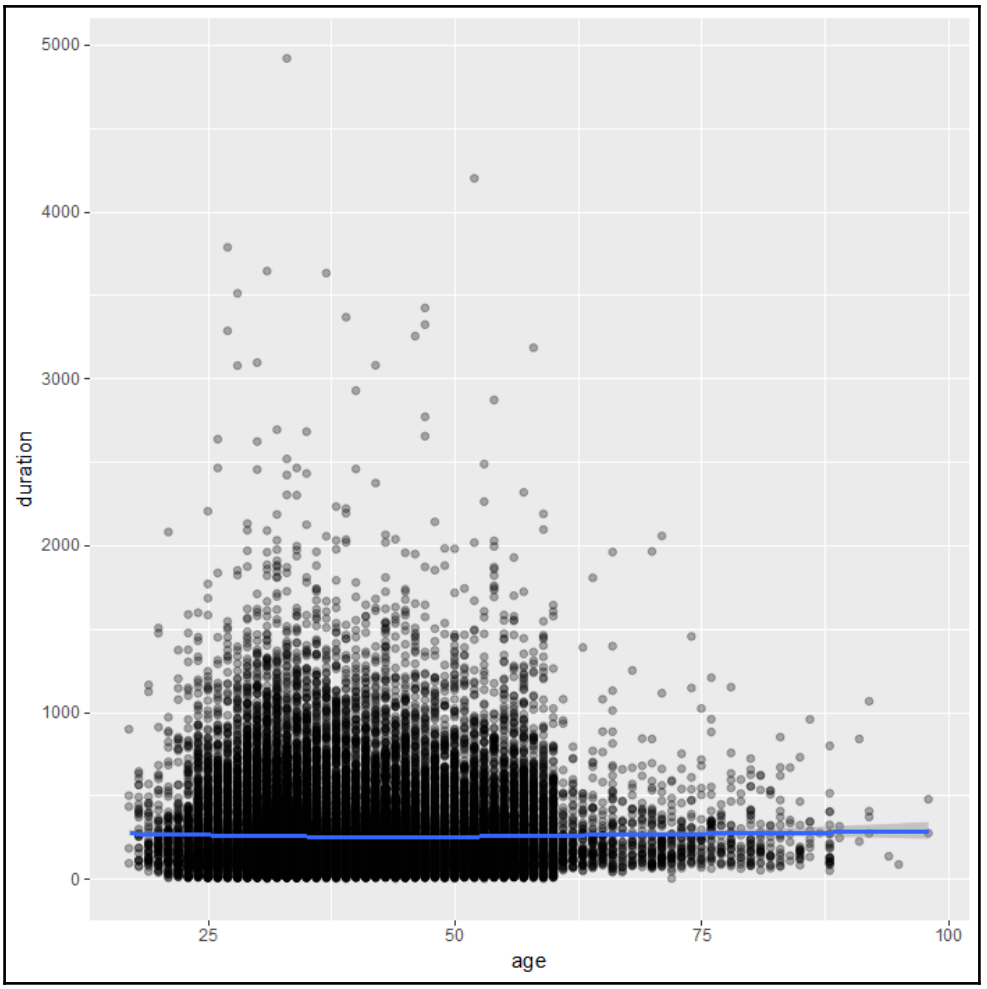

7500
5000
2500
0
count
25
50
75
100
age

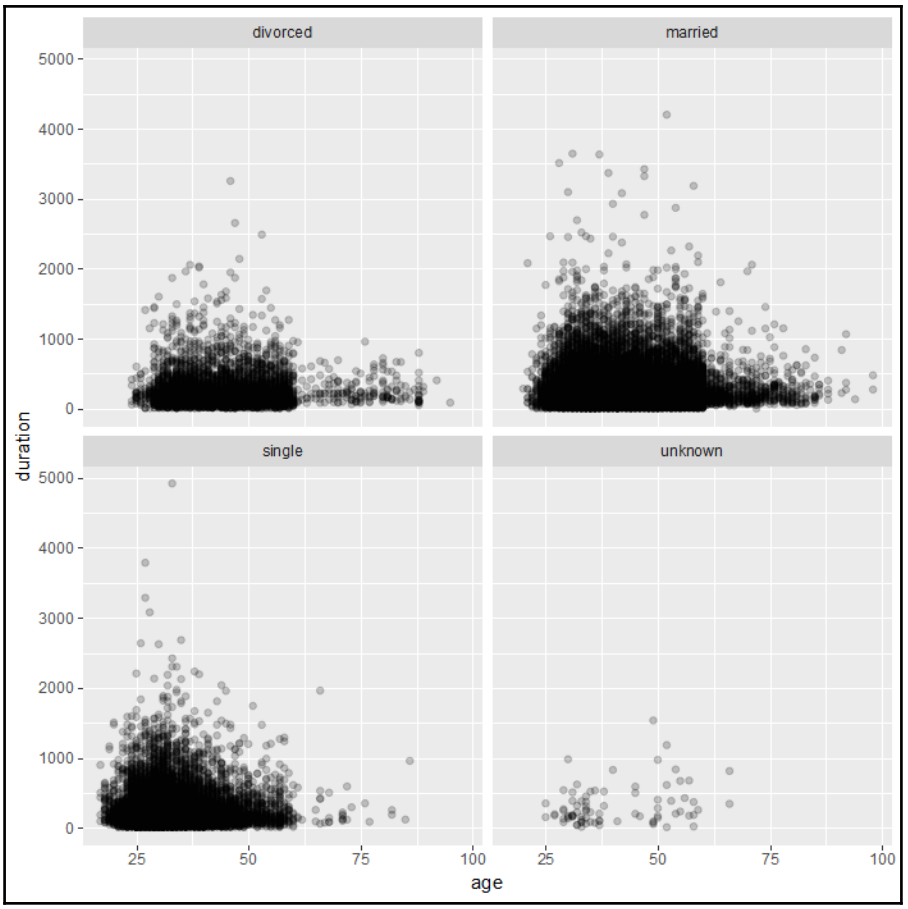

7500
5000
2500
0
count
30
60
90
age

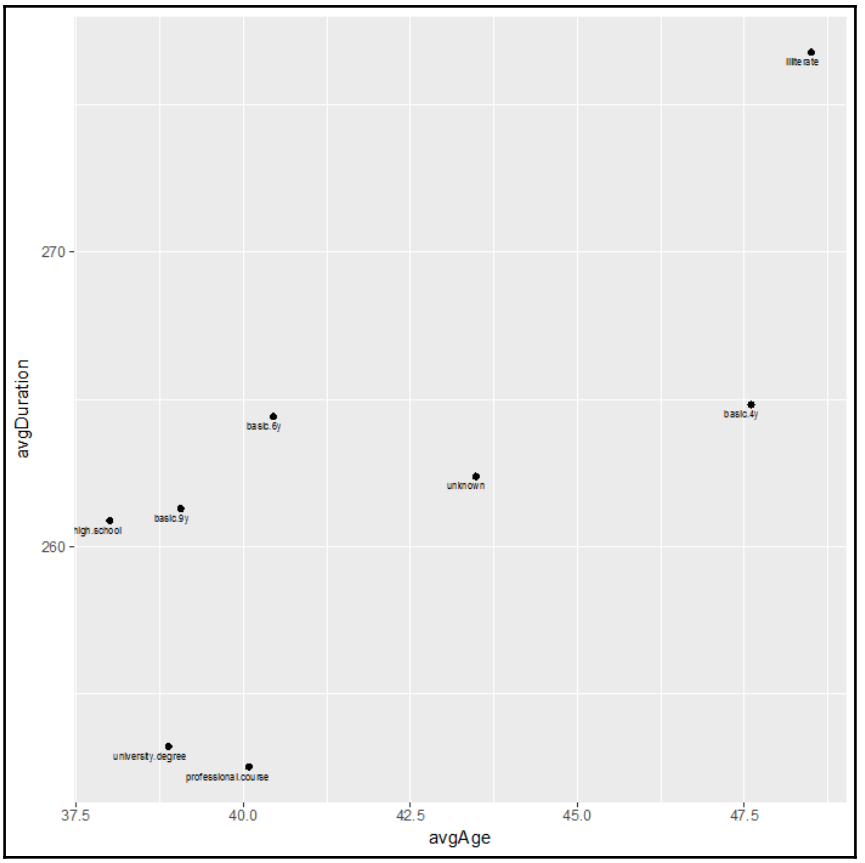

5000
4000
3000
2000
1000
0
duration
divorced
married
single
unknown
marital

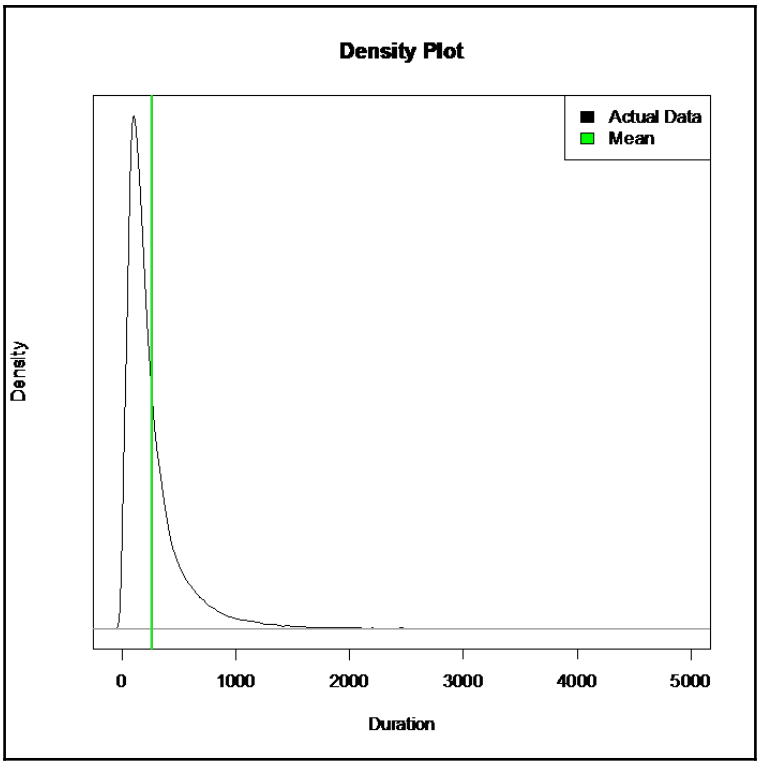

basic.4y
basic.6y
basic.9y
high.school
illiterate
professional.course
university.degree
unknown
count
age
0
500
1000
1500
2000
25
50
75
100

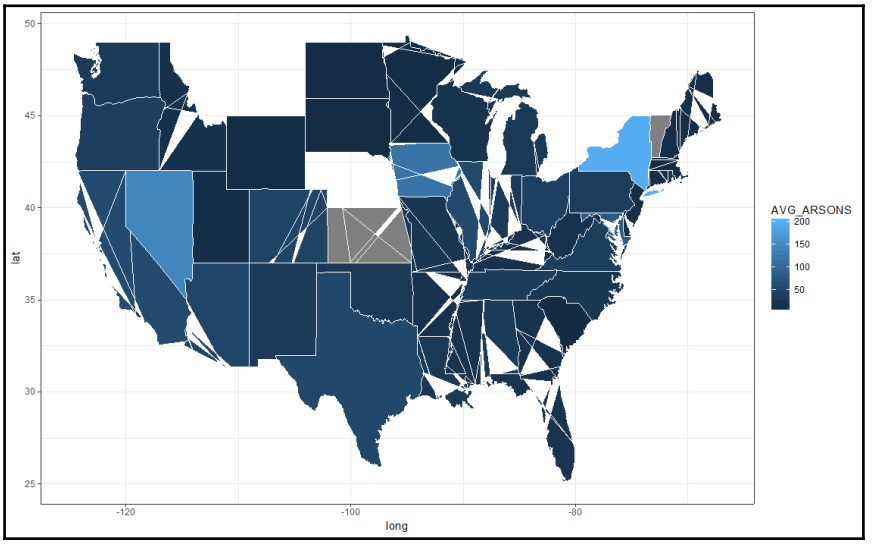

Age
Duration
100
80
60
40
20
5000
4000
3000
2000
1000
0

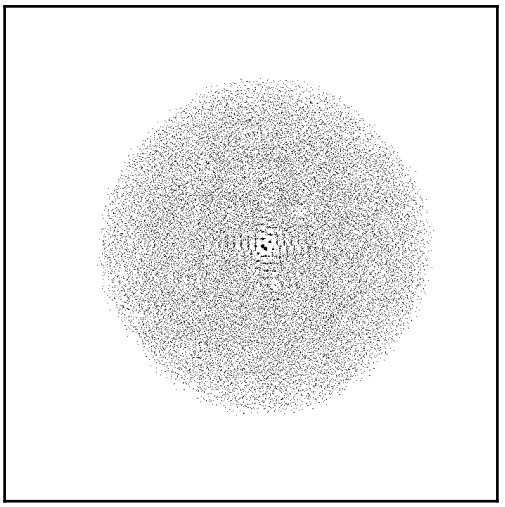

5000
4000
3000
2000
1000
0
duration
25
50
75
100
age

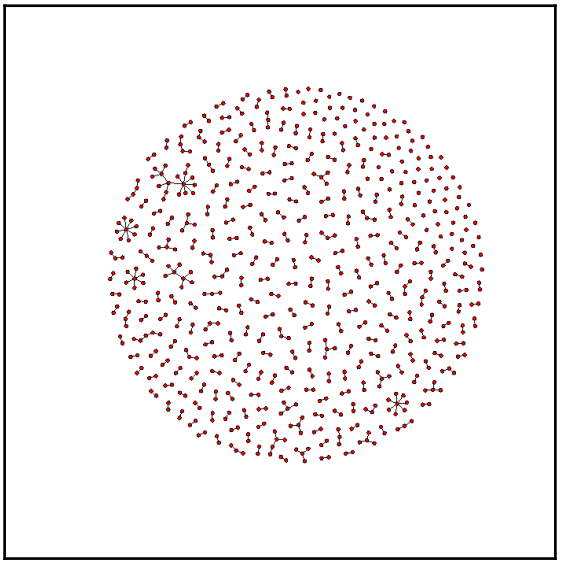

divorced
married
single
unknown
duration
age
5000
4000
3000
2000
1000
0
25
50
75
100


illiterate
basic.4y
basic.6y
unknown
basic.9y
high.school
university.degree
professional.course
270
260
avgDuration
37.5
40.0
42.5
45.0
47.5
avgAge


Density Plot
Actual Data
Mean
Density
0
1000
2000
3000
4000
5000
Duration


50
45
40
35
30
25
lat
-120
-100
-80
long
AVG_ARSONS
200
150
100
50





```
> indf <- read.df(csvPath, "csv", header = "true", inferSchema = "true", na.strings =
"NA", delimiter= ";") %>% withColumnRenamed("fixed acidity", "fixed_acidity") %>%
withColumnRenamed("volatile acidity", "volatile_acidity") %>%
withColumnRenamed("citric acid", "citric_acid") %>% withColumnRenamed("residual
sugar", "residual_sugar") %>% withColumnRenamed("free sulfur dioxide",
"free_sulfur_dioxide") %>% withColumnRenamed("total sulfur dioxide",
"total_sulfur_dioxide")
```

```
$coefficients
                      Estimate
(Intercept)          -0.6862963
fixed_acidity         0
volatile_acidity      0.3257158
citric_acid           0
residual_sugar        0
chlorides             0.3837135
free_sulfur_dioxide   0
total_sulfur_dioxide 3.210866e-08
density               0.5516459
pH                    0
sulphates             0
alcohol              -0.05705867
```

```
+-----+--------------------+--------------------+----------+
|label|       rawPrediction|         probability|prediction|
+-----+--------------------+--------------------+----------+
|  0.0|[0.54081405225716...|[0.63200176671046...|       1.0|
|  1.0|[0.63103229228006...|[0.65272349447245...|       1.0|
|  1.0|[0.60251318150008...|[0.64623107184477...|       1.0|
|  1.0|[0.68397029023517...|[0.66462424819744...|       1.0|
|  1.0|[0.66050917673984...|[0.65937475866608...|       1.0|
+-----+--------------------+--------------------+----------+
```

```
Formula:  label ~ .
Number of features:  11
Features:  fixed_acidity volatile_acidity citric_acid residual_sugar chlorides
free_sulfur_dioxide total_sulfur_dioxide density pH sulphates alcohol
Feature importances:
(11,[0,1,2,3,4,5,6,7,8,9,10],[0.016116203679387833,0.2000841279602036,0.06203377364164
843,0.04659246031382119,0.053967074218643885,0.09524761178321549,0.019576446849627522,
0.1974904528512302,0.02585073108815522,0.014956220347921618,0.26808489726614504])
Number of trees:  10
Tree weights:  1 1 1 1 1 1 1 1 1 1
 RandomForestClassificationModel (uid=rfc_caed094ec80d) with 10 trees
  Tree 0 (weight 1.0):
    If (feature 10 <= 10.5666666666667)
     If (feature 5 <= 14.0)
      If (feature 1 <= 0.305)
       If (feature 2 <= 0.26)
        If (feature 8 <= 2.96)
         Predict: 0.0
        Else (feature 8 > 2.96)
         Predict: 1.0
            .
            .
            .

Else (feature 5 > 16.0)
       If (feature 2 <= 0.25)
        If (feature 4 <= 0.049)
         Predict: 0.0
        Else (feature 4 > 0.049)
         Predict: 1.0
       Else (feature 2 > 0.25)
        If (feature 1 <= 0.2)
         Predict: 0.0
        Else (feature 1 > 0.2)
         Predict: 1.0
```

```
+-----+--------------------+--------------------+----------+
|label|       rawPrediction|         probability|prediction|
+-----+--------------------+--------------------+----------+
|  0.0|[5.20659697980061...|[0.52065969798006...|       1.0|
|  1.0|[6.75105122414708...|[0.67510512241470...|       1.0|
|  1.0|[7.69196575734326...|[0.76919657573432...|       1.0|
|  1.0|[7.86345675093300...|[0.78634567509330...|       1.0|
|  1.0|[8.95741684971412...|[0.89574168497141...|       1.0|
+-----+--------------------+--------------------+----------+
only showing top 5 rows
```

```
> indf <- read.df(csvPath, "csv", header = "true", inferSchema = "true", na.strings =
"NA", delimiter= ";") %>% withColumnRenamed("fixed acidity", "fixed_acidity") %>%
withColumnRenamed("volatile acidity", "volatile_acidity") %>%
withColumnRenamed("citric acid", "citric_acid") %>% withColumnRenamed("residual
sugar", "residual_sugar") %>% withColumnRenamed("free sulfur dioxide",
"free_sulfur_dioxide") %>% withColumnRenamed("total sulfur dioxide",
"total_sulfur_dioxide")
```

```
Deviance Residuals:
(Note: These are approximate quantiles with relative error <= 0.01)
    Min       1Q   Median       3Q      Max
-3.8070  -0.5037  -0.0498   0.4456   3.1916

Coefficients:
                      Estimate     Std. Error  t value  Pr(>|t|)
(Intercept)           48.828       7.6391      6.3919   1.7914e-10
fixed_acidity         -0.021401    0.014567    -1.4691  0.14186
volatile_acidity      -1.6505      0.10606     -15.562  0
citric_acid           0.043129     0.089881    0.47984  0.63136
residual_sugar        0.033293     0.0034613   9.6187   0
chlorides             -1.5754      0.50526     -3.118   0.0018317
free_sulfur_dioxide   0.0041696    0.00074331  5.6095   2.1407e-08
total_sulfur_dioxide  -0.00075066  0.00032633  -2.3004  0.02147
density               -46.604      7.7231      -6.0344  1.7133e-09
pH                    0.27827      0.078437    3.5477   0.00039231
sulphates             0.43647      0.092228    4.7325   2.2805e-06
alcohol               0.2551       0.012896    19.781   0

(Dispersion parameter for gaussian family taken to be 0.571833)

    Null deviance: 3841  on 4897  degrees of freedom
Residual deviance: 2794  on 4886  degrees of freedom
AIC: 11176

Number of Fisher Scoring iterations: 1
```

```
+-------+------------------+
|quality|        prediction|
+-------+------------------+
|      7| 6.555843712558648|
|      8|6.2422874502179795|
|      6| 5.664827458475145|
|      5| 5.263011944877043|
|      6| 6.050373346784674|
+-------+------------------+
only showing top 5 rows
```

```
+-----+----------+
|label|prediction|
+-----+----------+
|  0.0|         0|
|  1.0|         1|
|  1.0|         1|
|  1.0|         1|
|  1.0|         1|
+-----+----------+
only showing top 5 rows
```

```
[[1]]

Call:
glm(formula = quality ~ ., family = family, data = lindf)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.8348  -0.4934  -0.0379   0.4637   3.1143

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)           1.502e+02  1.880e+01   7.987 1.71e-15 ***
fixed_acidity         6.552e-02  2.087e-02   3.139  0.00171 **
volatile_acidity     -1.863e+00  1.138e-01 -16.373  < 2e-16 ***
citric_acid           2.209e-02  9.577e-02   0.231  0.81759
residual_sugar        8.148e-02  7.527e-03  10.825  < 2e-16 ***
chlorides            -2.473e-01  5.465e-01  -0.452  0.65097
free_sulfur_dioxide   3.733e-03  8.441e-04   4.422 9.99e-06 ***
total_sulfur_dioxide -2.857e-04  3.781e-04  -0.756  0.44979
density              -1.503e+02  1.907e+01  -7.879 4.04e-15 ***
pH                    6.863e-01  1.054e-01   6.513 8.10e-11 ***
sulphates             6.315e-01  1.004e-01   6.291 3.44e-10 ***
alcohol               1.935e-01  2.422e-02   7.988 1.70e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 0.5645372)

    Null deviance: 3841.0  on 4897  degrees of freedom
Residual deviance: 2758.3  on 4886  degrees of freedom
AIC: 11113

Number of Fisher Scoring iterations: 2
```

```
[[2]]

Call:
glm(formula = quality ~ ., family = family, data = lindf)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.67362  -0.20825  -0.01416   0.18914   1.21919

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)           2.809e+01  1.114e+01   2.521 0.011698 *
fixed_acidity         1.281e-02  1.188e-02   1.078 0.281003
volatile_acidity     -3.346e-01  6.423e-02  -5.208  1.9e-07 ***
citric_acid           2.529e-03  5.328e-02   0.047 0.962138
residual_sugar        1.456e-02  4.365e-03   3.335 0.000854 ***
chlorides            -6.267e-02  3.128e-01  -0.200 0.841191
free_sulfur_dioxide   6.224e-04  4.631e-04   1.344 0.178939
total_sulfur_dioxide -3.694e-05  2.104e-04  -0.176 0.860626
density              -2.736e+01  1.130e+01  -2.421 0.015457 *
pH                    1.235e-01  5.903e-02   2.092 0.036417 *
sulphates             1.087e-01  5.450e-02   1.995 0.046011 *
alcohol               3.036e-02  1.421e-02   2.137 0.032594 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 656.56  on 4897  degrees of freedom
Residual deviance: 471.74  on 4886  degrees of freedom
AIC: 18258

Number of Fisher Scoring iterations: 4
```

Chapter 9:
Developing Applications with Spark SQL

```
lineWords: String = " ACCESSION NUMBER CONFORMED SUBMISSION TYPE KPUBLIC DOCUMENT
COUNT CONFORMED PERIOD OF REPORT FILED AS OF DATE DATE AS OF CHANGE FILER COMPANY DATA
COMPANY CONFORMED NAME APPLE INC CENTRAL INDEX KEY STANDARD INDUSTRIAL CLASSIFICATION
ELECTRONIC COMPUTERS IRS NUMBER STATE OF INCORPORATION CA FISCAL YEAR END FILING
VALUES FORM TYPE K SEC ACT Act SEC FILE NUMBER FILM NUMBER BUSINESS ADDRESS STREET ONE
INFINITE LOOP CITY CUPERTINO STATE CA ZIP BUSINESS PHONE MAIL ADDRESS STREET ONE
INFINITE LOOP CITY CUPERTINO STATE CA ZIP FORMER COMPANY FORMER CONFORMED NAME APPLE
COMPUTER INC DATE OF NAME CHANGE K Table of Contents UNITED STATES SECURITIES AND
EXCHANGE COMMISSION Washington Form K Mark One For the fiscal year ended September or
For the transition period from to Commis...
```

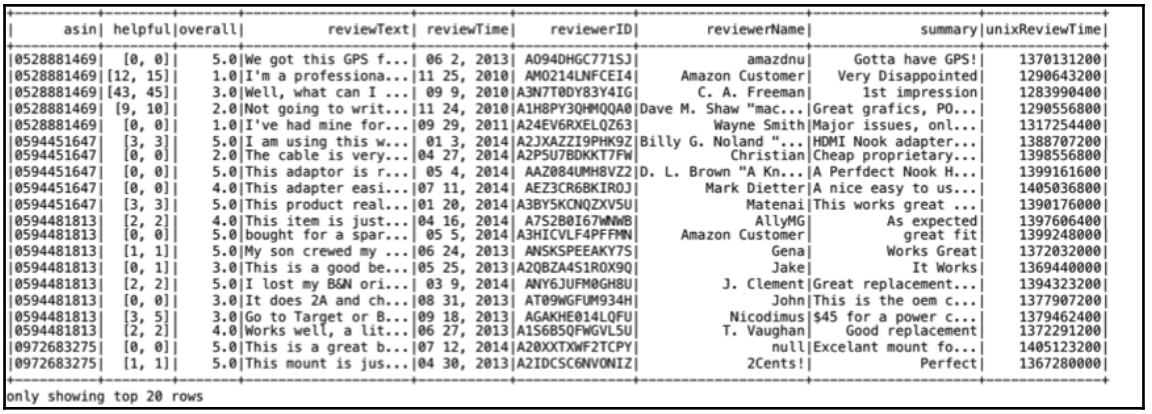
```
stopwords: Array[String] = Array(a, a's, able, about, above, according, accordingly,
across, actually, after, afterwards, again, against, ain't, all, allow, allows,
almost, alone, along, already, also, although, always, am, among, amongst, an, and,
another, any, anybody, anyhow, anyone, anything, anyway, anyways, anywhere, apart,
appear, appreciate, appropriate, are, aren't, around, as, aside, ask, asking,
associated, at, available, away, awfully, b, be, became, because, become, becomes,
becoming, been, before, beforehand, behind, being, believe, below, beside, besides,
best, better, between, beyond, both, brief, but, by, c, c'mon, c's, came, can, can't,
cannot, cant, cause, causes, certain, certainly, changes, clearly, co, com, come,
comes, concerning, consequently, consider, consideri...
```

```
+------------+
|wordsInStory|
+------------+
|       money|
|       japan|
|          IF|
|      DOLLAR|
|     FOLLOWS|
|          IF|
|      DOLLAR|
|     FOLLOWS|
|        WALL|
|      STREET|
|        WILL|
|      DIVEST|
|          By|
|          If|
|         the|
|      dollar|
|        goes|
|         the|
|         way|
|          of|
+------------+
only showing top 20 rows
```

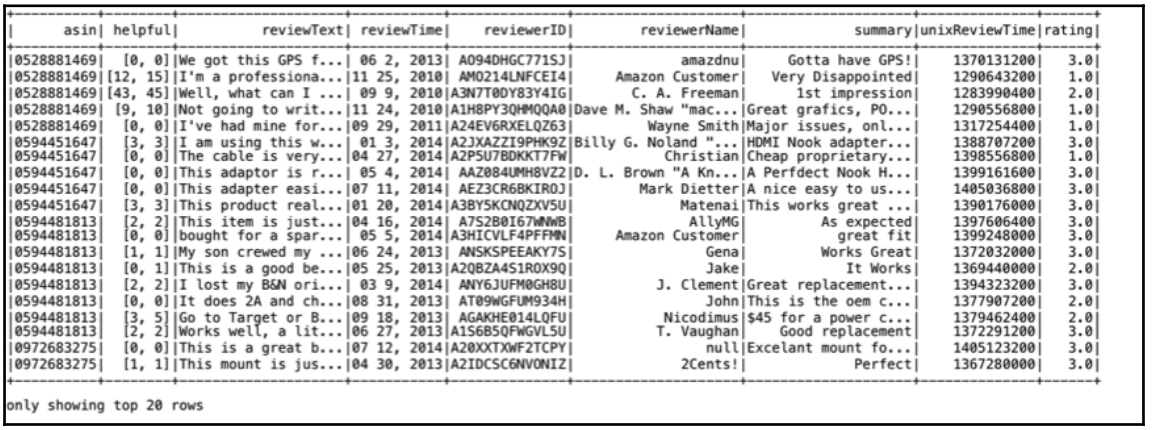
```
+------------+
|wordsInStory|
+------------+
|         MAR|
|       sugar|
|       grain|
|        corn|
|       SUGAR|
|     PROGRAM|
|       SUGAR|
|     PROGRAM|
|         CUT|
|        SENT|
|          TO|
|    CONGRESS|
|          BY|
|       MARCH|
|         The|
| Agriculture|
|  Department|
|    formally|
| transmitted|
|          to|
+------------+
only showing top 20 rows
```

```
+-----+--------------+-----------------------------------------------------------------------+
|topic|termIndices   |termWeights                                                            |
+-----+--------------+-----------------------------------------------------------------------+
|0    |[40, 168, 0]  |[0.013479828514522157, 0.007988498264811112, 0.007834556972004216]|
|1    |[3, 231, 292] |[0.011157720765464901, 0.010950386549468765, 0.009933881657369987]|
|2    |[0, 32, 3]    |[0.01699803302924405, 0.00897362312135043, 0.005552939490101682]  |
|3    |[3, 44, 211]  |[0.010640383419263965, 0.009589519391260229, 0.008719954776309678]|
|4    |[559, 130, 678]|[0.014699094408320094, 0.013142729264265737, 0.006943073670536392]|
|5    |[4, 12, 1]    |[0.03450414256020877, 0.02081996234868235, 0.016553531009112363]  |
|6    |[0, 6, 13]    |[0.1008048266175278, 0.012606198050741528, 0.012490689284082269]  |
|7    |[124, 198, 218]|[0.024444393452602815, 0.02039110399678152, 0.019951675877435307] |
|8    |[197, 6, 447] |[0.008833939029528915, 0.007881170922130781, 0.006090595819592994]|
|9    |[2, 1, 9]     |[0.017878616381976048, 0.012342706034341355, 0.00995938921623339] |
+-----+--------------+-----------------------------------------------------------------------+
```

```
vocab: Array[String] = Array(mr, applicant, court, act, tribunal, made, evidence, may,
application, respondent, v, decision, appellant, said, case, claim, order, also,
whether, first, time, j, one, ltd, b, person, reasons, australia, costs, relevant,
notice, upon, agreement, relation, appeal, proceedings, federal, hearing, pty,
information, company, part, matter, judgment, conduct, respect, circumstances,
question, however, ms, section, respondents, date, must, given, make, second,
minister, issue, review, within, law, applicants, counsel, fact, proceeding, parties,
orders, two, documents, c, provided, terms, particular, basis, see, referred,
statement, view, letter, claims, effect, australian, business, party, set, fca,
following, period, cth, group, subject, present, matters, reason, ...
```



```
+----------+-------+-------+--------------------+-----------+--------------+--------------------+--------------------+--------------+
|      asin|helpful|overall|          reviewText| reviewTime|    reviewerID|        reviewerName|             summary|unixReviewTime|
+----------+-------+-------+--------------------+-----------+--------------+--------------------+--------------------+--------------+
|0528881469|  [0, 0]|    5.0|We got this GPS f...| 06 2, 2013| AO94DHGC771SJ|             amazdnu|     Gotta have GPS!|    1370131200|
|0528881469|[12, 15]|    1.0|I'm a professiona...|11 25, 2010| AMO214LNFCEI4|     Amazon Customer|   Very Disappointed|    1290643200|
|0528881469|[43, 45]|    3.0|Well, what can I ...| 09 9, 2010|A3N7T0DY83Y4IG|       C. A. Freeman|      1st impression|    1283990400|
|0528881469| [9, 10]|    2.0|Not going to writ...|11 24, 2010|A1H8PY3QHMQQA0|Dave M. Shaw "mac...|Great grafics, PO...|    1290556800|
|0528881469|  [0, 0]|    1.0|I've had mine for...|09 29, 2011|A24EV6RXELQZ63|         Wayne Smith|Major issues, onl...|    1317254400|
|0594451647|  [3, 3]|    5.0|I am using this w...| 01 3, 2014|A2JXAZZI9PHK9Z|Billy G. Noland "...|HDMI Nook adapter...|    1388707200|
|0594451647|  [0, 0]|    2.0|The cable is very...|04 27, 2014|A2P5U7BDKKT7FW|            Christian|Cheap proprietary...|    1398556800|
|0594451647|  [0, 0]|    5.0|This adaptor is r...| 05 4, 2014| AAZ084UMH8VZ2|D. L. Brown "A Kn...|A Perfdect Nook H...|    1399161600|
|0594451647|  [0, 0]|    4.0|This adapter easi...|07 11, 2014| AEZ3CR6BKIROJ|        Mark Dietter|A nice easy to us...|    1405036800|
|0594451647|  [3, 3]|    5.0|This product real...|01 20, 2014|A3BY5KCNQZXV5U|             Matenai|This works great ...|    1390176000|
|0594481813|  [2, 2]|    4.0|This item is just...|04 16, 2014| A7S2B0I67WNWB|             AllyMG|          As expected|    1397606400|
|0594481813|  [0, 0]|    5.0|bought for a spar...| 05 5, 2014|A3HICVLF4PFFMN|     Amazon Customer|           great fit|    1399248000|
|0594481813|  [1, 1]|    5.0|My son crewed my ...|06 24, 2013| ANSKSPEEAKY7S|                Gena|          Works Great|    1372032000|
|0594481813|  [0, 1]|    3.0|This is a good be...|05 25, 2013|A2QBZA4S1ROX9Q|                Jake|             It Works|    1369440000|
|0594481813|  [2, 2]|    5.0|I lost my B&N ori...| 03 9, 2014| ANY6JUFM0GH8U|          J. Clement|Great replacement...|    1394323200|
|0594481813|  [0, 0]|    3.0|It does 2A and ch...|08 31, 2013| AT09WGFUM934H|                John|This is the oem c...|    1377907200|
|0594481813|  [3, 5]|    3.0|Go to Target or B...|09 18, 2013| AGAKHE014LQFU|           Nicodimus|$45 for a power c...|    1379462400|
|0594481813|  [2, 2]|    4.0|Works well, a lit...|06 27, 2013|A1S6B5QFWGVL5U|          T. Vaughan|    Good replacement|    1372291200|
|0972683275|  [0, 0]|    5.0|This is a great b...|07 12, 2014|A20XXTXWF2TCPY|                null|Excelant mount fo...|    1405123200|
|0972683275|  [1, 1]|    5.0|This mount is jus...|04 30, 2013|A2IDCSC6NVONIZ|             2Cents!|             Perfect|    1367280000|
+----------+-------+-------+--------------------+-----------+--------------+--------------------+--------------------+--------------+
only showing top 20 rows
```



```
+----------+-------+--------------------+-----------+--------------+--------------------+--------------------+--------------+------+
|      asin|helpful|          reviewText| reviewTime|    reviewerID|        reviewerName|             summary|unixReviewTime|rating|
+----------+-------+--------------------+-----------+--------------+--------------------+--------------------+--------------+------+
|0528881469|  [0, 0]|We got this GPS f...| 06 2, 2013| AO94DHGC771SJ|             amazdnu|     Gotta have GPS!|    1370131200|   3.0|
|0528881469|[12, 15]|I'm a professiona...|11 25, 2010| AMO214LNFCEI4|     Amazon Customer|   Very Disappointed|    1290643200|   1.0|
|0528881469|[43, 45]|Well, what can I ...| 09 9, 2010|A3N7T0DY83Y4IG|       C. A. Freeman|      1st impression|    1283990400|   2.0|
|0528881469| [9, 10]|Not going to writ...|11 24, 2010|A1H8PY3QHMQQA0|Dave M. Shaw "mac...|Great grafics, PO...|    1290556800|   1.0|
|0528881469|  [0, 0]|I've had mine for...|09 29, 2011|A24EV6RXELQZ63|         Wayne Smith|Major issues, onl...|    1317254400|   1.0|
|0594451647|  [3, 3]|I am using this w...| 01 3, 2014|A2JXAZZI9PHK9Z|Billy G. Noland "...|HDMI Nook adapter...|    1388707200|   3.0|
|0594451647|  [0, 0]|The cable is very...|04 27, 2014|A2P5U7BDKKT7FW|            Christian|Cheap proprietary...|    1398556800|   1.0|
|0594451647|  [0, 0]|This adaptor is r...| 05 4, 2014| AAZ084UMH8VZ2|D. L. Brown "A Kn...|A Perfdect Nook H...|    1399161600|   3.0|
|0594451647|  [0, 0]|This adapter easi...|07 11, 2014| AEZ3CR6BKIROJ|        Mark Dietter|A nice easy to us...|    1405036800|   3.0|
|0594451647|  [3, 3]|This product real...|01 20, 2014|A3BY5KCNQZXV5U|             Matenai|This works great ...|    1390176000|   3.0|
|0594481813|  [2, 2]|This item is just...|04 16, 2014| A7S2B0I67WNWB|             AllyMG|          As expected|    1397606400|   3.0|
|0594481813|  [0, 0]|bought for a spar...| 05 5, 2014|A3HICVLF4PFFMN|     Amazon Customer|           great fit|    1399248000|   3.0|
|0594481813|  [1, 1]|My son crewed my ...|06 24, 2013| ANSKSPEEAKY7S|                Gena|          Works Great|    1372032000|   3.0|
|0594481813|  [0, 1]|This is a good be...|05 25, 2013|A2QBZA4S1ROX9Q|                Jake|             It Works|    1369440000|   2.0|
|0594481813|  [2, 2]|I lost my B&N ori...| 03 9, 2014| ANY6JUFM0GH8U|          J. Clement|Great replacement...|    1394323200|   3.0|
|0594481813|  [0, 0]|It does 2A and ch...|08 31, 2013| AT09WGFUM934H|                John|This is the oem c...|    1377907200|   2.0|
|0594481813|  [3, 5]|Go to Target or B...|09 18, 2013| AGAKHE014LQFU|           Nicodimus|$45 for a power c...|    1379462400|   2.0|
|0594481813|  [2, 2]|Works well, a lit...|06 27, 2013|A1S6B5QFWGVL5U|          T. Vaughan|    Good replacement|    1372291200|   3.0|
|0972683275|  [0, 0]|This is a great b...|07 12, 2014|A20XXTXWF2TCPY|                null|Excelant mount fo...|    1405123200|   3.0|
|0972683275|  [1, 1]|This mount is jus...|04 30, 2013|A2IDCSC6NVONIZ|             2Cents!|             Perfect|    1367280000|   3.0|
+----------+-------+--------------------+-----------+--------------+--------------------+--------------------+--------------+------+
only showing top 20 rows
```

```
[1.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0
,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0
.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0]
```

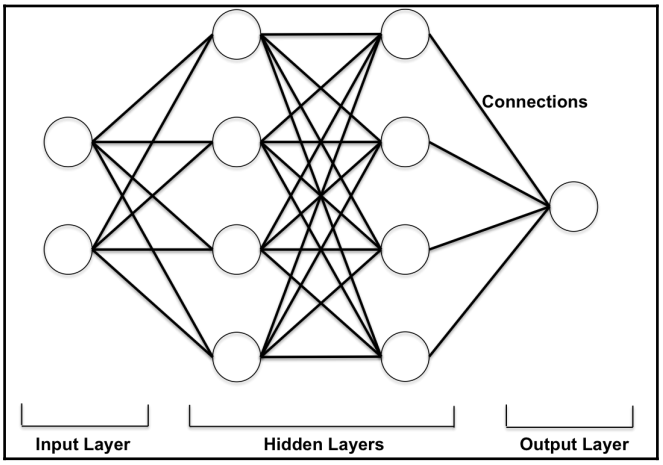
```
+-----+---+
|label| id|
+-----+---+
|  1.0|  0|
|  2.0|  1|
|  3.0|  2|
|  4.0|  3|
|  5.0|  4|
|  6.0|  5|
|  7.0|  6|
|  8.0|  7|
|  9.0|  8|
|  1.0|  9|
|  2.0| 10|
|  3.0| 11|
|  4.0| 12|
|  5.0| 13|
|  6.0| 14|
|  7.0| 15|
|  8.0| 16|
|  9.0| 17|
|  1.0| 18|
|  2.0| 19|
+-----+---+
only showing top 20 rows
```

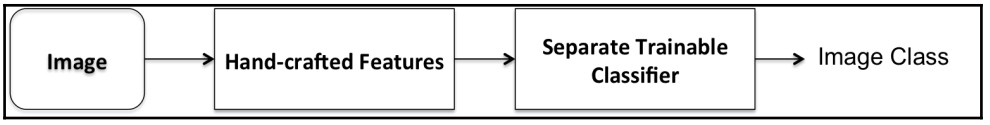
```
+----------+-----+--------------------+
|prediction|label|            features|
+----------+-----+--------------------+
|       3.0|  1.0|(857,[331,493,704...|
|       1.0|  1.0|(857,[206,493,704...|
|       1.0|  1.0|(857,[70,206,510,...|
|       3.0|  2.0|(857,[402,707,844...|
|       3.0|  2.0|(857,[332,370,613...|
+----------+-----+--------------------+
only showing top 5 rows
```

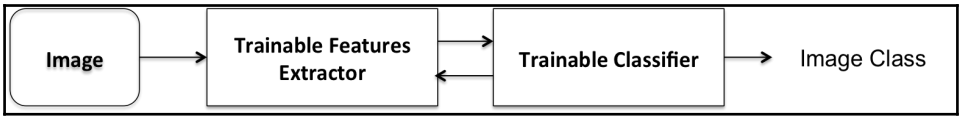
```
(99, 1.0) --> prob=[2.6027854456883833E-
7,0.9314238151102939,0.008305853987875035,0.009684546310917754,0.008694654921083497,0.
007912106138523606,0.008594082752851902,0.00851969033111665,0.00915496702922429,0.0077
100231395686196], prediction=1.0
(81, 1.0) --> prob=[7.498134657859313E-
7,0.7689461362309903,0.0276681467730525,0.03305074069318609,0.028689441789579495,0.026
109985274150933,0.02903619052527176,0.028159628722867137,0.03154087462238654,0.0267981
0555504927], prediction=1.0
.
.
.
(8589934797, 9.0) --> prob=[1.7113801070622645E-
6,0.05306638932278816,0.0587392018096767,0.04692529890827765,0.12399075223480296,0.074
38456473981786,0.12186164275694751,0.11653564109284925,0.10754704437383655,0.296947753
38089624], prediction=9.0
(8589934653, 9.0) --> prob=[1.2546665900143864E-
6,0.04134664431466302,0.04388117754243587,0.04929496870409657,0.08984018414356594,0.05
9337783485833365,0.06842089455713343,0.05013171592689752,0.04112335342906031,0.5566220
23229724], prediction=9.0
```

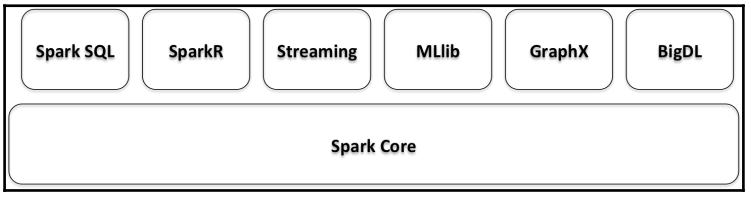
```
+----------+-----+--------------------+
|prediction|label|            features|
+----------+-----+--------------------+
|       1.0|  1.0|(857,[331,493,704...|
|       1.0|  1.0|(857,[206,493,704...|
|       1.0|  1.0|(857,[70,206,510,...|
|       2.0|  2.0|(857,[402,707,844...|
|       2.0|  2.0|(857,[332,370,613...|
+----------+-----+--------------------+
```

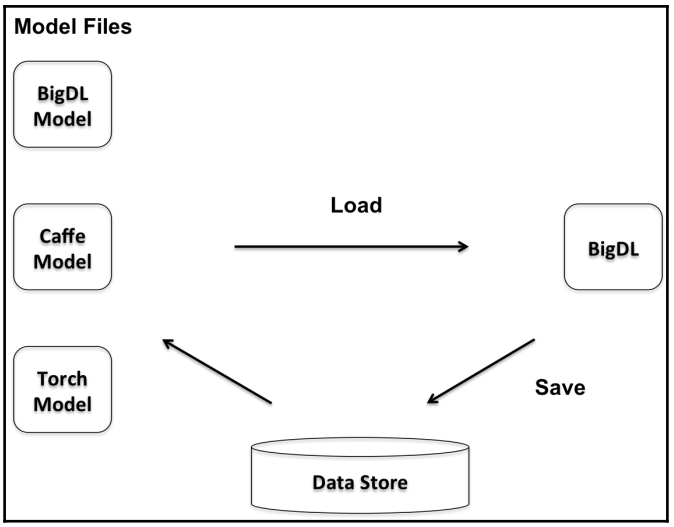
Chapter 10:
Using Spark SQL in Deep Learning Applicati ons

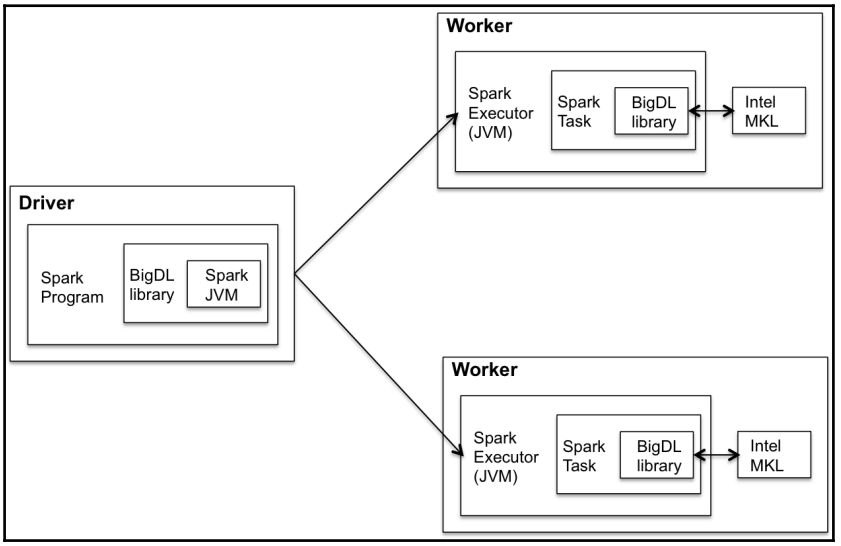


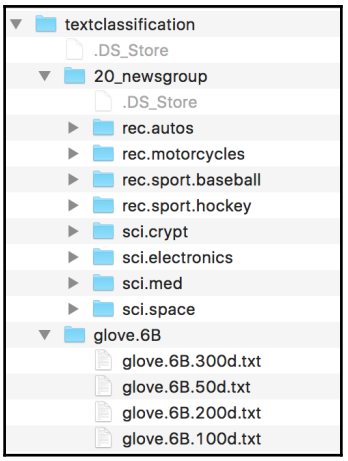


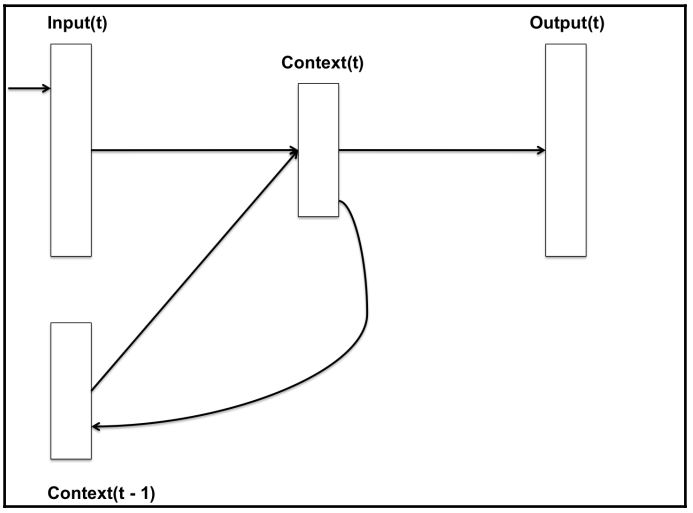


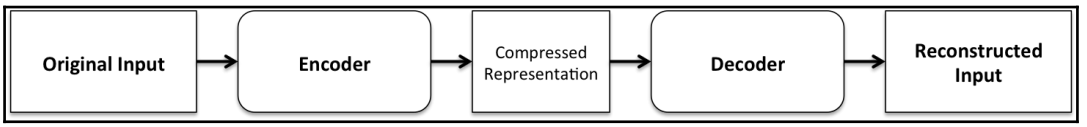


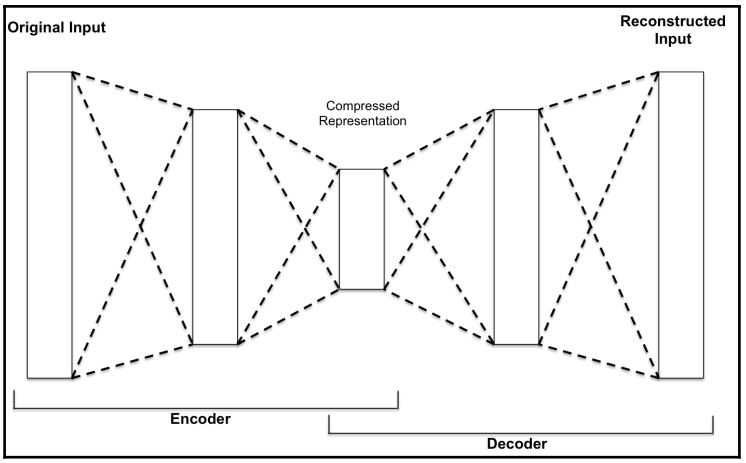

Model Files
BigDL Model
Caffe Model
Torch Model
Load
BigDL
Save
Data Store


Worker
Spark Executor (JVM)
Spark Task
BigDL library
Intel MKL
Driver
Spark Program
BigDL library
Spark JVM
Worker
Spark Executor (JVM)
Spark Task
BigDL library
Intel MKL

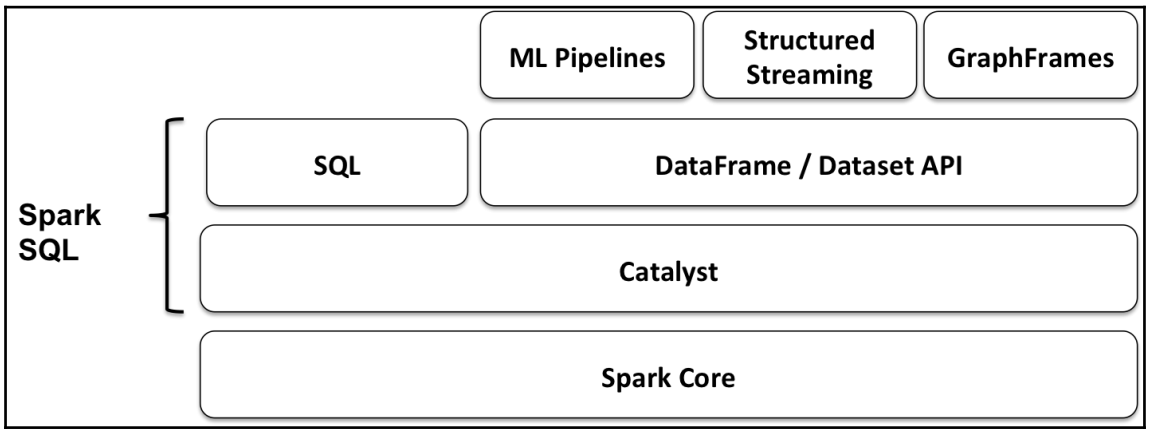

textclassification
.DS_Store
20_newsgroup
.DS_Store
rec.autos
rec.motorcycles
rec.sport.baseball
rec.sport.hockey
sci.crypt
sci.electronics
sci.med
sci.space
glove.6B
glove.6B.300d.txt
glove.6B.50d.txt
glove.6B.200d.txt
glove.6B.100d.txt

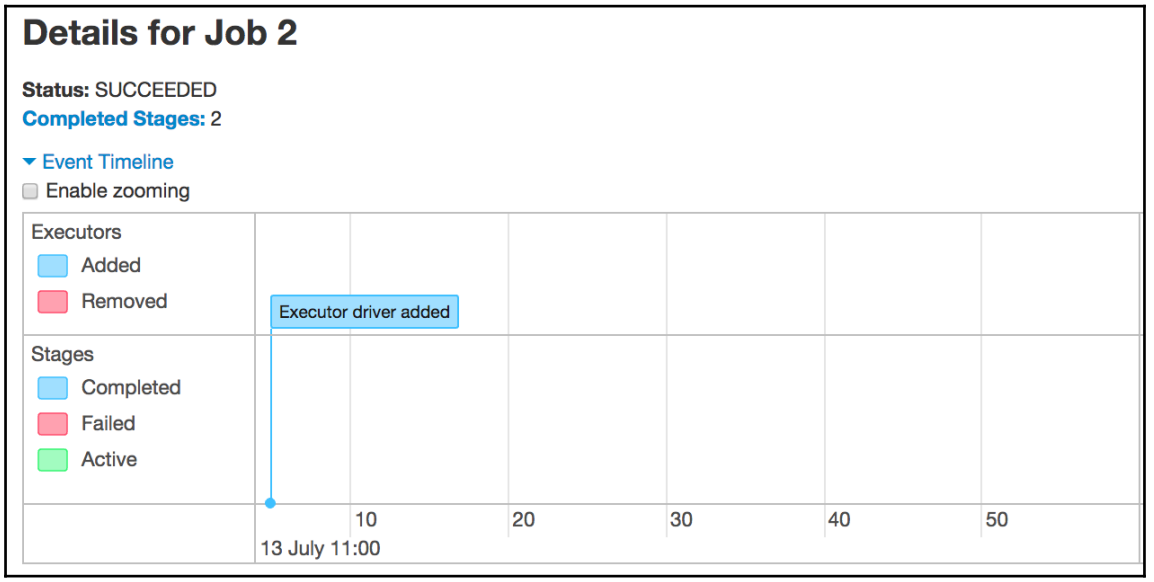

Input(t)
Context(t)
Output(t)
Context(t - 1)

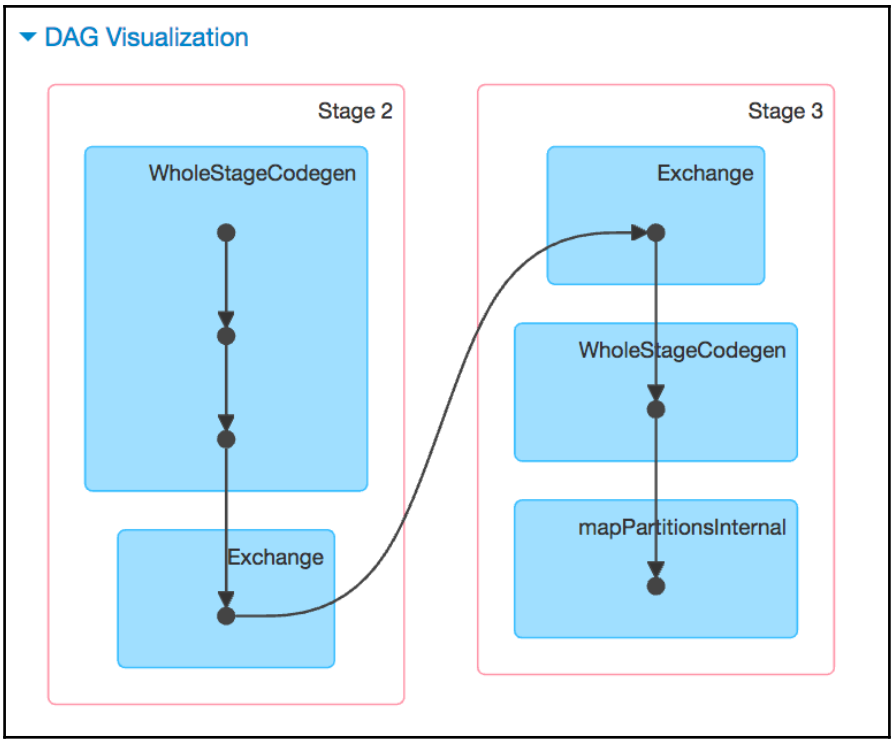

Original Input
Encoder
Compressed Representation
Decoder
Reconstructed Input


Original Input
Reconstructed Input
Compressed Representation
Encoder
Decoder

Chapter 11:
Tuning Spark SQL Components for Performance

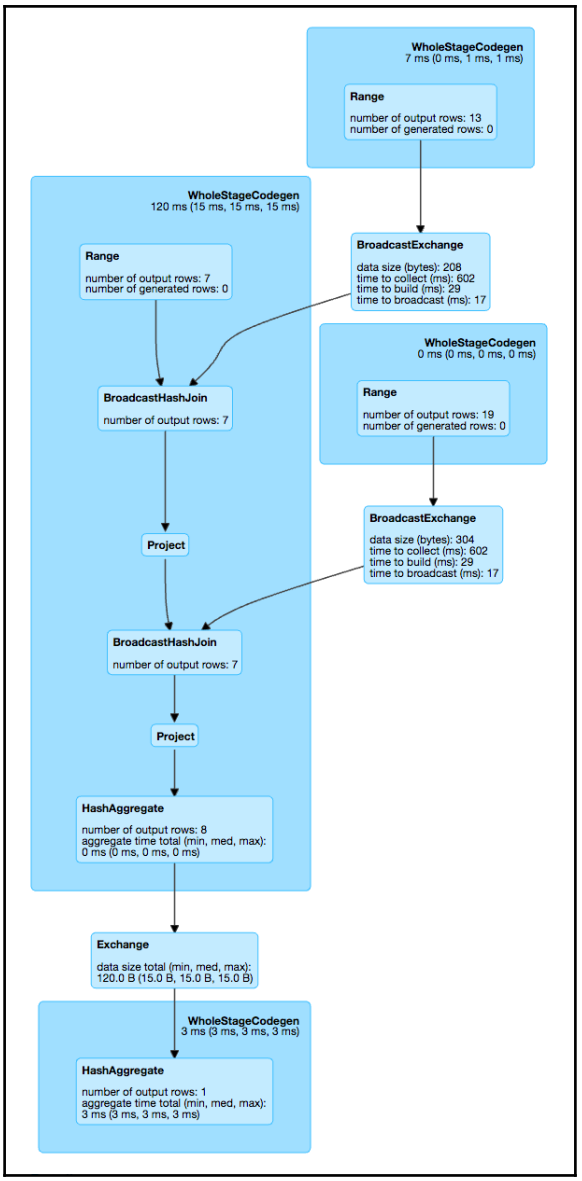






Completed Stages (2)

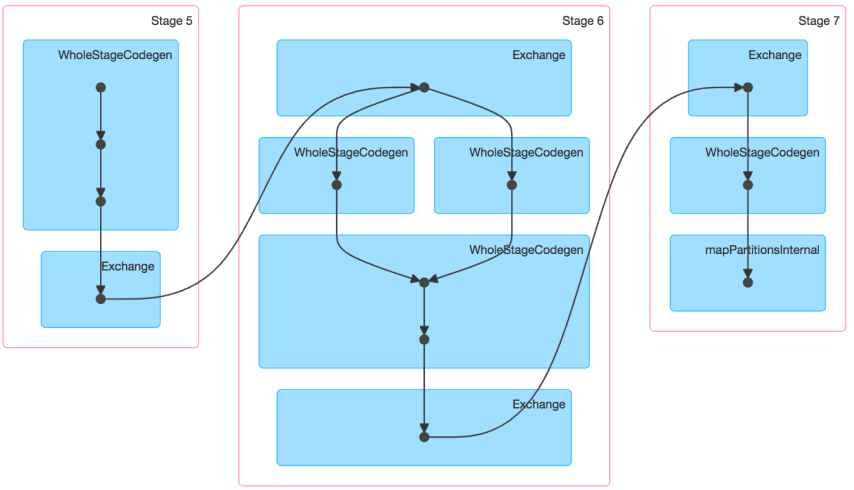
Stage Id	Description	Submitted	Duration	Tasks: Succeeded/Total	Input	Output	Shuffle Read	Shuffle Write
3	count at <console>:48 +details	2017/07/13 11:02:22	27 ms	1/1			469.0 B	
2	count at <console>:48 +details	2017/07/13 11:02:22	61 ms	8/8				469.0 B


WholeStageCodegen
7 ms (0 ms, 1 ms, 1 ms)
Range
number of output rows: 13
number of generated rows: 0
WholeStageCodegen
120 ms (15 ms, 15 ms, 15 ms)
Range
number of output rows: 7
number of generated rows: 0
BroadcastExchange
data size (bytes): 208
time to collect (ms): 602
time to build (ms): 29
time to broadcast (ms): 17
WholeStageCodegen
0 ms (0 ms, 0 ms, 0 ms)
Range
number of output rows: 19
number of generated rows: 0
BroadcastHashJoin
number of output rows: 7
Project
BroadcastExchange
data size (bytes): 304
time to collect (ms): 602
time to build (ms): 29
time to broadcast (ms): 17
BroadcastHashJoin
number of output rows: 7
Project
HashAggregate
number of output rows: 8
aggregate time total (min, med, max):
0 ms (0 ms, 0 ms, 0 ms)
Exchange
data size total (min, med, max):
120.0 B (15.0 B, 15.0 B, 15.0 B)
WholeStageCodegen
3 ms (3 ms, 3 ms, 3 ms)
HashAggregate
number of output rows: 1
aggregate time total (min, med, max):
3 ms (3 ms, 3 ms, 3 ms)

Details for Job 4

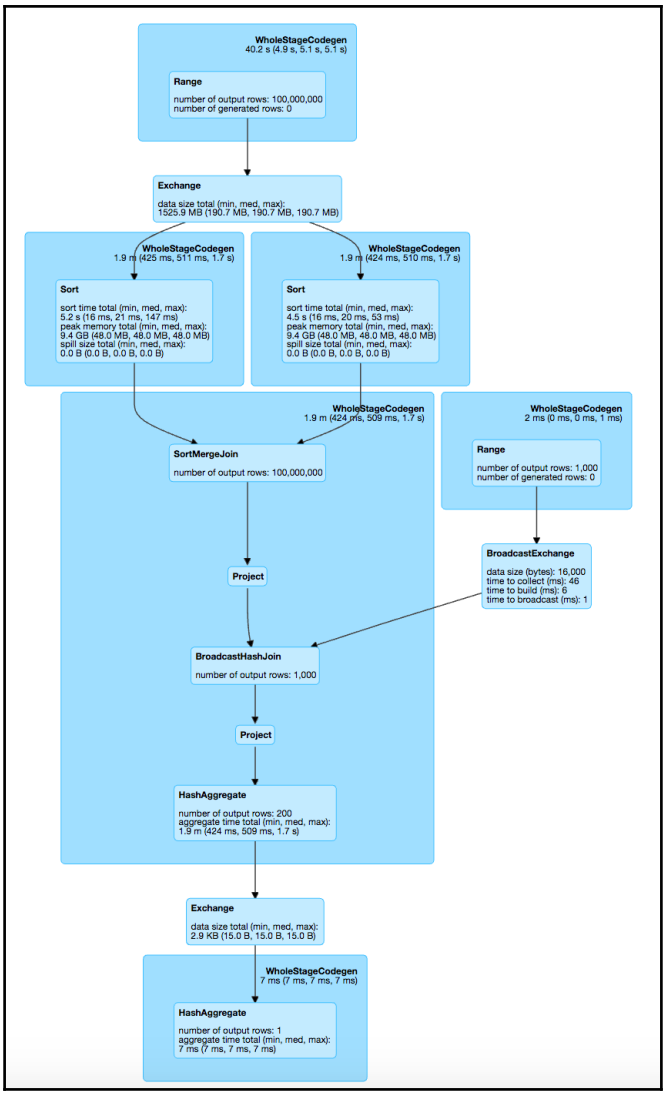
Status: SUCCEEDED
Completed Stages: 3

▸ Event Timeline
▾ DAG Visualization



Completed Stages (3)

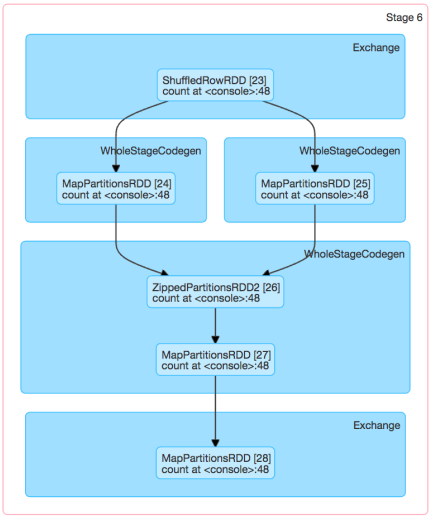
Stage Id ▾	Description		Submitted	Duration	Tasks: Succeeded/Total	Input	Output	Shuffle Read	Shuffle Write
7	count at <console>:48	+details	2017/07/13 11:11:52	14 ms	1/1			11.5 KB	
6	count at <console>:48	+details	2017/07/13 11:11:37	15 s	200/200			959.8 MB	11.5 KB
5	count at <console>:48	+details	2017/07/13 11:11:30	6 s	8/8				479.9 MB


WholeStageCodegen
40.2 s (4.9 s, 5.1 s, 5.1 s)
Range
number of output rows: 100,000,000
number of generated rows: 0
Exchange
data size total (min, med, max):
1525.9 MB (190.7 MB, 190.7 MB, 190.7 MB)
WholeStageCodegen
1.9 m (425 ms, 511 ms, 1.7 s)
Sort
sort time total (min, med, max):
5.2 s (16 ms, 21 ms, 147 ms)
peak memory total (min, med, max):
9.4 GB (48.0 MB, 48.0 MB, 48.0 MB)
spill size total (min, med, max):
0.0 B (0.0 B, 0.0 B, 0.0 B)
WholeStageCodegen
1.9 m (424 ms, 510 ms, 1.7 s)
Sort
sort time total (min, med, max):
4.5 s (16 ms, 20 ms, 53 ms)
peak memory total (min, med, max):
9.4 GB (48.0 MB, 48.0 MB, 48.0 MB)
spill size total (min, med, max):
0.0 B (0.0 B, 0.0 B, 0.0 B)
WholeStageCodegen
1.9 m (424 ms, 509 ms, 1.7 s)
SortMergeJoin
number of output rows: 100,000,000
Project
BroadcastHashJoin
number of output rows: 1,000
Project
HashAggregate
number of output rows: 200
aggregate time total (min, med, max):
1.9 m (424 ms, 509 ms, 1.7 s)
WholeStageCodegen
2 ms (0 ms, 0 ms, 1 ms)
Range
number of output rows: 1,000
number of generated rows: 0
BroadcastExchange
data size (bytes): 16,000
time to collect (ms): 46
time to build (ms): 6
time to broadcast (ms): 1
Exchange
data size total (min, med, max):
2.9 KB (15.0 B, 15.0 B, 15.0 B)
WholeStageCodegen
7 ms (7 ms, 7 ms, 7 ms)
HashAggregate
number of output rows: 1
aggregate time total (min, med, max):
7 ms (7 ms, 7 ms, 7 ms)

Details for Stage 6 (Attempt 0)

Total Time Across All Tasks: 1.9 min
Locality Level Summary: Any: 200
Shuffle Read: 959.8 MB / 200000000
Shuffle Write: 11.5 KB / 200

▾ DAG Visualization



Summary Metrics for 200 Completed Tasks

Metric	Min	25th percentile	Median	75th percentile	Max
Duration	0.4 s	0.5 s	0.5 s	0.6 s	2 s
GC Time	0 ms	9 ms	12 ms	22 ms	0.5 s
Shuffle Read Size / Records	4.8 MB / 996224	4.8 MB / 999172	4.8 MB / 1000060	4.8 MB / 1000926	4.8 MB / 1003388
Shuffle Write Size / Records	56.0 B / 1	59.0 B / 1	59.0 B / 1	59.0 B / 1	59.0 B / 1

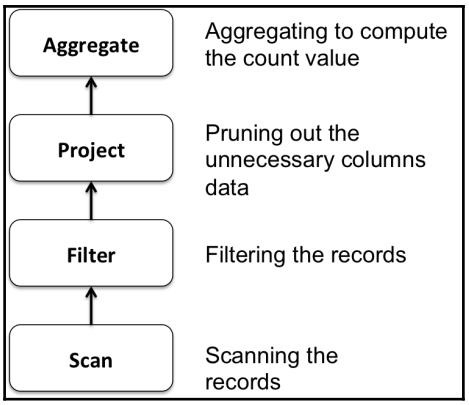
▾ Aggregated Metrics by Executor

Executor ID ▲	Address	Task Time	Total Tasks	Failed Tasks	Killed Tasks	Succeeded Tasks	Shuffle Read Size / Records	Shuffle Write Size / Records	Blacklisted
driver	192.168.1.110:54752	2.0 min	200	0	0	200	959.8 MB / 200000000	11.5 KB / 200	0

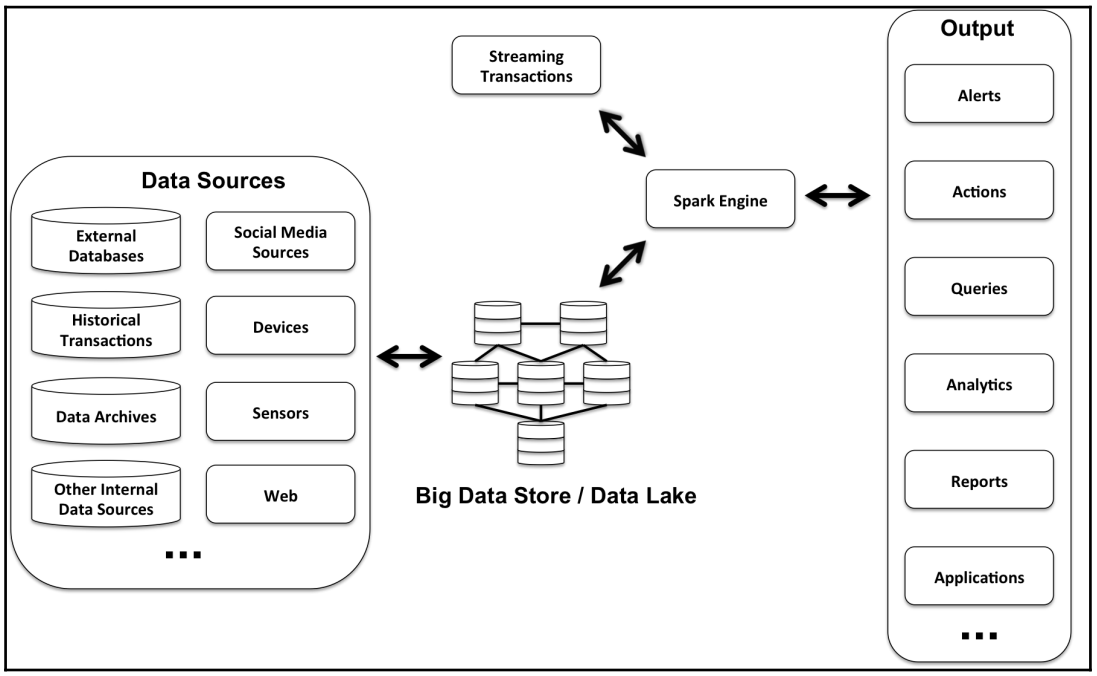
```
+---------+
|     name|
+---------+
| 33270267|
|123904911|
|174393361|
|161741766|
|159902261|
|  2386203|
+---------+

Time taken with CBO OFF & JOIN REORDER DISABLED: 51.440248589 seconds
```

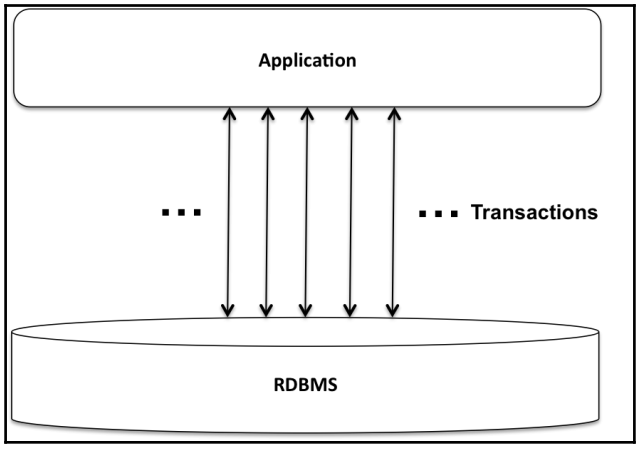
```
+---------+
|     name|
+---------+
|174393361|
| 33270267|
|123904911|
|  2386203|
|161741766|
|159902261|
+---------+

Time taken with CBO ON & JOIN REORDER DISABLED: 42.191017053 seconds
```

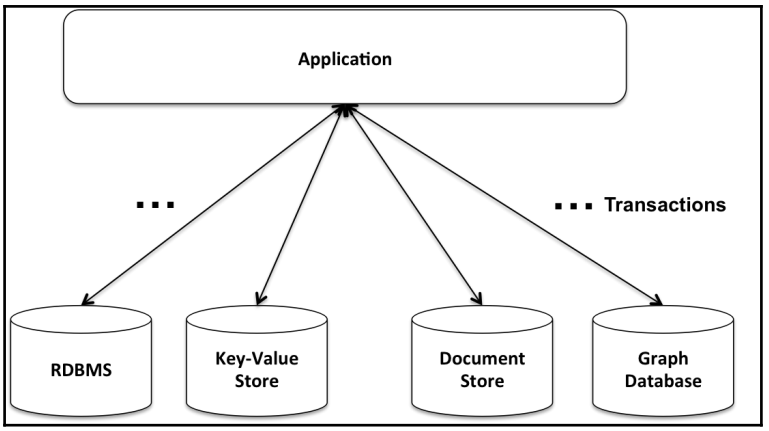
```
+---------+
|     name|
+---------+
|  2386203|
| 33270267|
|159902261|
|161741766|
|174393361|
|123904911|
+---------+

Time taken with CBO ON & JOIN REORDER ENABLED: 7.099971757 seconds
```

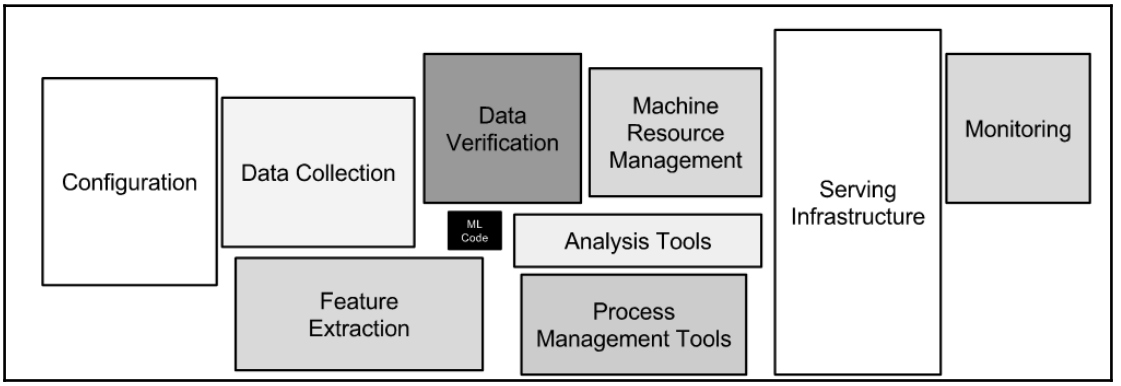

Aggregate
Aggregating to compute the count value
Project
Pruning out the unnecessary columns data
Filter
Filtering the records
Scan
Scanning the records

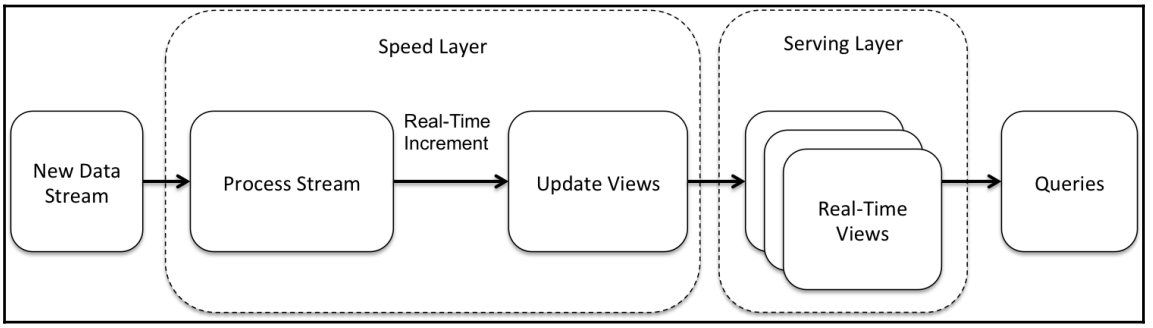
Chapter 12: Spark SQL in Large-Scale Application Architectures

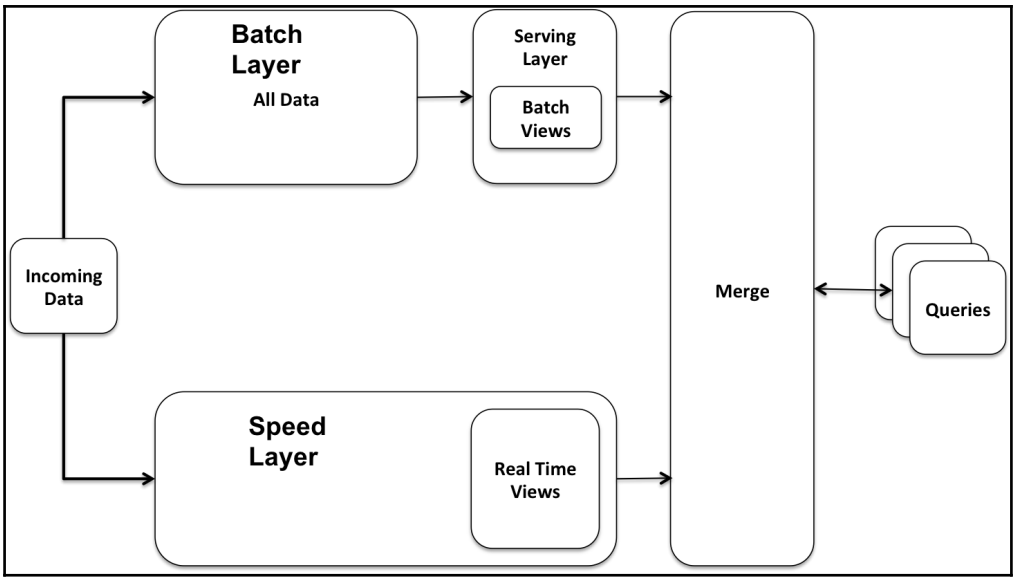


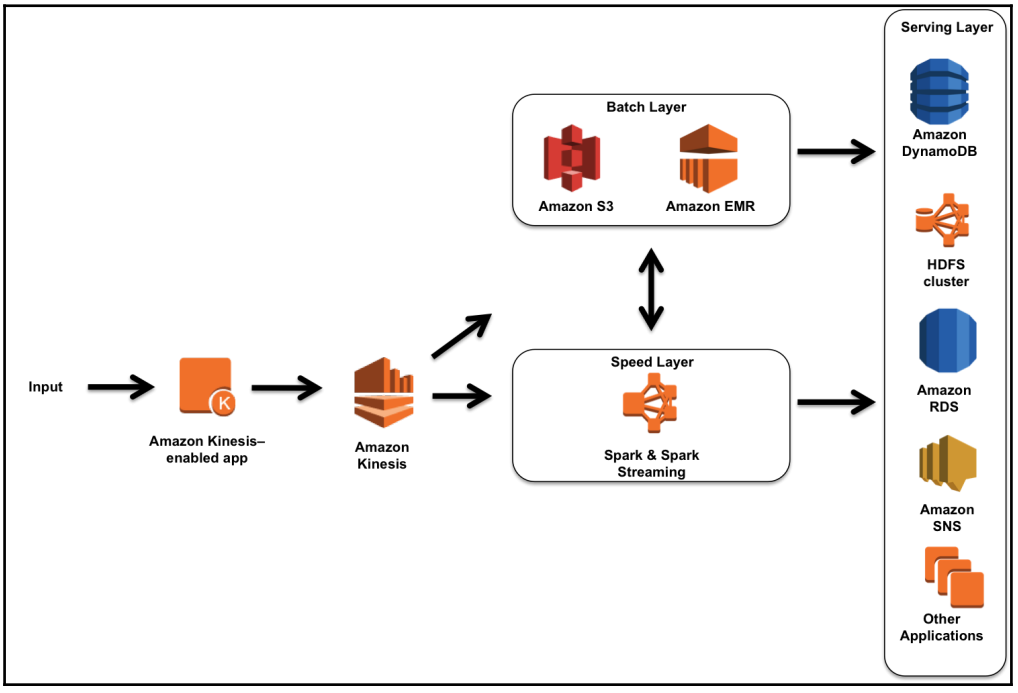


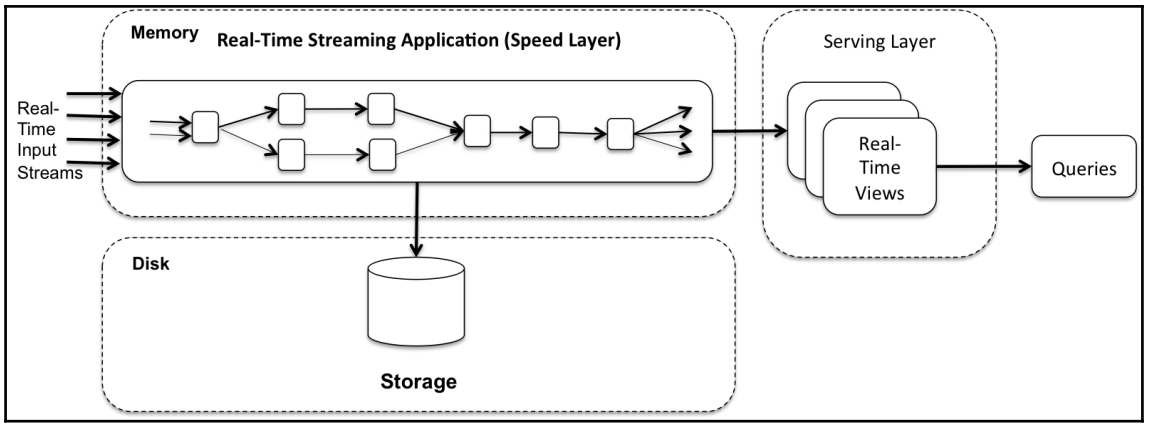

Application
...
... Transactions
RDBMS
Key-Value Store
Document Store
Graph Database

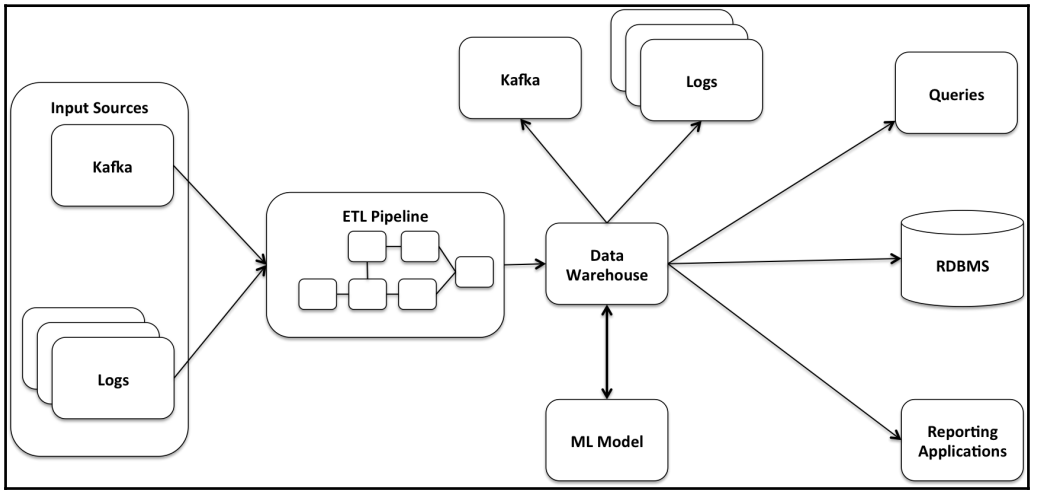

Configuration
Data Collection
Data Verification
Machine Resource Management
Serving Infrastructure
Monitoring
ML Code
Analysis Tools
Feature Extraction
Process Management Tools


Speed Layer
Serving Layer
New Data Stream
Process Stream
Real-Time Increment
Update Views
Real-Time Views
Queries


Batch Layer
All Data
Serving Layer
Batch Views
Incoming Data
Merge
Queries
Speed Layer
Real Time Views


Serving Layer
Batch Layer
Amazon S3
Amazon EMR
Amazon DynamoDB
HDFS cluster
Input
Amazon Kinesis–enabled app
Amazon Kinesis
Speed Layer
Spark & Spark Streaming
Amazon RDS
Amazon SNS
Other Applications






```
root
 |-- contributors: array (nullable = true)
 |    |-- element: long (containsNull = true)
...
 |-- place: struct (nullable = true)
...
 |    |-- country: string (nullable = true)
 |    |-- country_code: string (nullable = true)
...
```

```
-------------------------------------------
Batch: 2
-------------------------------------------
+--------------+
|       country|
+--------------+
| United States|
|United Kingdom|
|        Brasil|
|        Brasil|
| United States|
| United States|
|        Brasil|
| United States|
| United States|
|United Kingdom|
|United Kingdom|
|     Argentina|
| United States|
| United States|
|        Brasil|
|United Kingdom|
| United States|
| United States|
| United States|
|  South Africa|
+--------------+
only showing top 20 rows
```

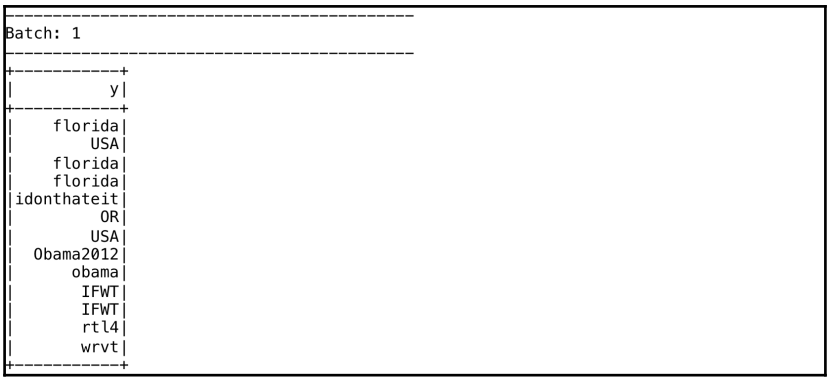
```
-------------------------------------------
Batch: 2
-------------------------------------------
+--------------------+--------------------+--------------+------------+--------------------+--------------------+--------------+----------+--------------------+
|          attributes|        bounding_box|       country|country_code|           full_name|                id|          name|place_type|                 url|
+--------------------+--------------------+--------------+------------+--------------------+--------------------+--------------+----------+--------------------+
|         [null,null]|[WrappedArray(Wra...| United States|          US|         Seaford, DE|b719350492e3ff2f|       Seaford|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...|United Kingdom|          GB|         Wigan, Wigan|e7e4fe93a6cfb48e|         Wigan|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...|        Brasil|          BR|      Bauru, São Paulo|77af3e46c7a19d54|         Bauru|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...|        Brasil|          BR|São Paulo, São Paulo|68e019afec7d0ba5|     São Paulo|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...| United States|          US|          Boston, MA|67b98f17fdcf20be|        Boston|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...| United States|          US|          New York, NY|27485069891a7938|      New York|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...|        Brasil|          BR|Foz do Iguaçu, Pa...|91f818a4abfb1d4d|  Foz do Iguaçu|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...| United States|          US| West Palm Beach, FL|03ee378d6ed2a73f|West Palm Beach|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...| United States|          US|           Queens, NY|b6ea2e341ba4356f|        Queens|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...|United Kingdom|          GB|West Lothian, Wes...|5ef1506ee13806ae|   West Lothian|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...|United Kingdom|          GB| Westminster, London|548c2b8e3921a85a|    Westminster|      city|http://api.twitte...|
|         [null,null]|                null|     Argentina|          AR|           Argentina|4d3b316fe2e52b29|     Argentina|   country|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...| United States|          US|  Citrus Heights, CA|a89175c4c91f45a3| Citrus Heights|      city|http://api.twitte...|
|[null,8144 Indian...|[WrappedArray(Wra...| United States|          US| Coach USA, Highland|5fd07ec0b51fd138|      Coach USA|       poi|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...|        Brasil|          BR|Rio de Janeiro, R...|97bcdfca1a2dca59| Rio de Janeiro|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...|United Kingdom|          GB|  Lewes, East Sussex|016d3e5601d32304|         Lewes|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...| United States|          US|       Manhattan, NY|086752cb03de1d5d|      Manhattan|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...| United States|          US|       Kalamazoo, MI|413ef5a0d23bfe4f|     Kalamazoo|      city|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...| United States|          US|            Ohio, US|de599025180e2ee7|          Ohio|     admin|http://api.twitte...|
|         [null,null]|[WrappedArray(Wra...|  South Africa|          ZA|   Pretoria, Gauteng|0e587c59401d0a27|      Pretoria|      city|http://api.twitte...|
+--------------------+--------------------+--------------+------------+--------------------+--------------------+--------------+----------+--------------------+
only showing top 20 rows
```

```
…
|     |-- id: long (nullable = true)
|     |-- id_str: string (nullable = true)
|     |-- in_reply_to_screen_name: string (nullable = true)
…
```

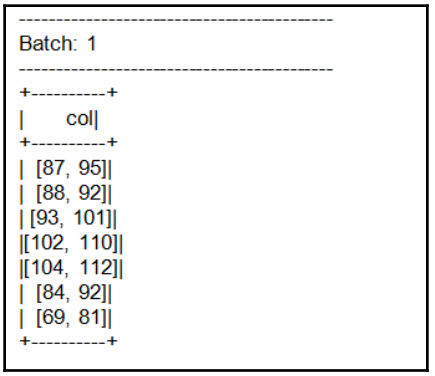
```
-------------------------------------------
Batch: 2
-------------------------------------------
+--------------------+
|        locationInfo|
+--------------------+
|        [US,Seaford]|
|          [GB,Wigan]|
|          [BR,Bauru]|
|      [BR,São Paulo]|
|         [US,Boston]|
|       [US,New York]|
|  [BR,Foz do Iguaçu]|
|[US,West Palm Beach]|
|          [US,Queens]|
|   [GB,West Lothian]|
|    [GB,Westminster]|
|      [AR,Argentina]|
| [US,Citrus Heights]|
|      [US,Coach USA]|
| [BR,Rio de Janeiro]|
|          [GB,Lewes]|
|      [US,Manhattan]|
|      [US,Kalamazoo]|
|           [US,Ohio]|
|       [ZA,Pretoria]|
+--------------------+
only showing top 20 rows
```

```
…
 |    |-- entities: struct (nullable = true)
 |    |    |-- hashtags: array (nullable = true)
 |    |    |    |-- element: struct (containsNull = true)
 |    |    |    |    |-- indices: array (nullable = true)
 |    |    |    |    |    |-- element: long (containsNull = true)
 |    |    |    |    |-- text: string (nullable = true)
 |    |    |-- media: array (nullable = true)
 |    |    |    |-- element: struct (containsNull = true)
…
 |    |    |    |    |-- type: string (nullable = true)
 |    |    |    |    |-- url: string (nullable = true)
…
```

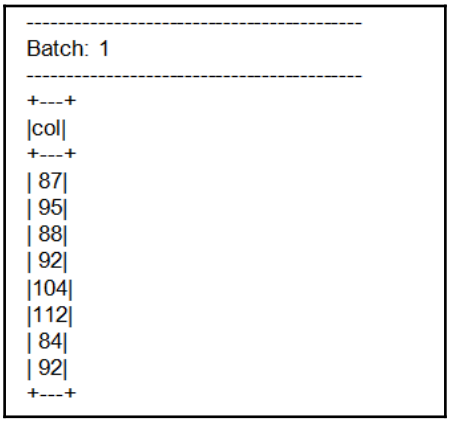
```
-------------------------------------------
Batch: 1
-------------------------------------------
+---+
|  z|
+---+
| 87|
| 88|
|104|
| 84|
+---+
```



```
-------------------------------------------
Batch: 1
-------------------------------------------
+-----------+
|          y|
+-----------+
|    florida|
|        USA|
|    florida|
|    florida|
|idonthateit|
|         OR|
|        USA|
|  Obama2012|
|      obama|
|       IFWT|
|       IFWT|
|       rtl4|
|       wrvt|
+-----------+
```



```
-------------------------------------------
Batch: 1
-------------------------------------------
+----------+
|       col|
+----------+
|  [87, 95]|
|  [88, 92]|
| [93, 101]|
|[102, 110]|
|[104, 112]|
|  [84, 92]|
|  [69, 81]|
+----------+
```



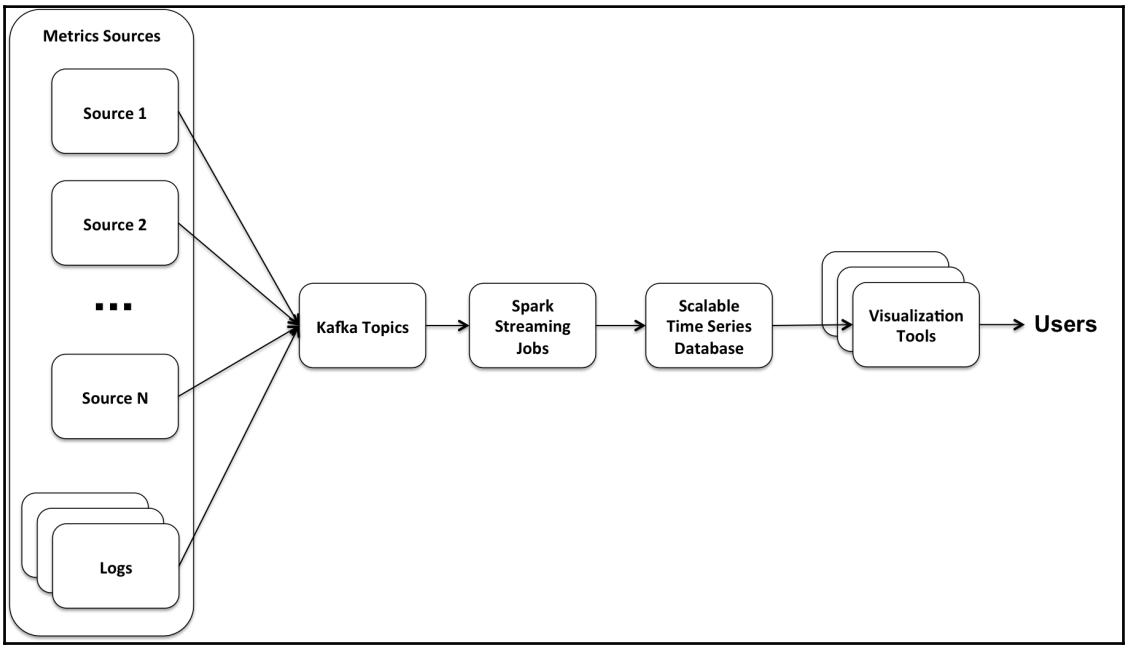
```
-------------------------------------------
Batch: 1
-------------------------------------------
+---+
|col|
+---+
| 87|
| 95|
| 88|
| 92|
|104|
|112|
| 84|
| 92|
+---+
```

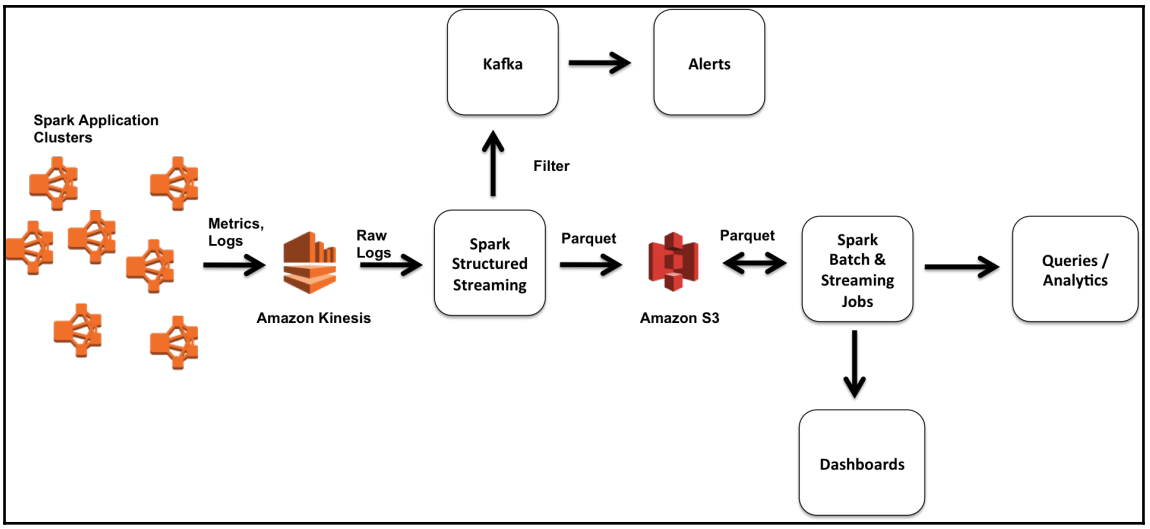
```
-------------------------------------------
Batch: 2
-------------------------------------------
+--------------------+
|                   c|
+--------------------+
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
|{"x":["photo"],"y...|
+--------------------+
only showing top 20 rows
```

```
+----+----+
|   a|   b|
+----+----+
|   1| 2.0|
|   2|null|
|null|null|
|   7| 8.0|
|null|null|
|null|null|
|   1| 2.1|
+----+----+
```

```
+---------------+----+----+----+----+----+----+
|_corrupt_record|   a|   b|   c|   d|   e|   f|
+---------------+----+----+----+----+----+----+
|           null|   1| 2.0|   3|null|null|null|
|           null|   2|null|null|   5| 3.0|null|
|           null|null|null|   4|   1|null|   6|
| {"a":7, "b":{}|null|null|null|null|null|null|
|           null|null|null|   5|   3| 4.5|null|
|           null|null|null|null|   4| 3.0|   3|
|           null|   1| 2.1|   3|   4| 3.0|   3|
+---------------+----+----+----+----+----+----+
```

```
+----+----+----+----+----+----+
|   a|   b|   c|   d|   e|   f|
+----+----+----+----+----+----+
|   1| 2.0|   3|null|null|null|
|   2|null|null|   5| 3.0|null|
|null|null|   4|   1|null|   6|
|null|null|   5|   3| 4.5|null|
|null|null|null|   4| 3.0|   3|
|   1| 2.1|   3|   4| 3.0|   3|
+----+----+----+----+----+----+
```






```
-------------------------------------------
Batch: 0
-------------------------------------------
+--------------------+----------+--------------------+--------------------+--------------------+--------------+---------+
|                  ts|      date|     clientIpAddress|   original_dateTime|             request|httpStatusCode|bytesSent|
+--------------------+----------+--------------------+--------------------+--------------------+--------------+---------+
|1995-07-01 09:30:01|1995-07-01|        199.72.81.55|[01/Jul/1995:00:0...|GET /history/apol...|           200|     6245|
|1995-07-01 09:30:06|1995-07-01|unicomp6.unicomp.net|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|     3985|
|1995-07-01 09:30:09|1995-07-01|      199.120.110.21|[01/Jul/1995:00:0...|GET /shuttle/miss...|           200|     4085|
|1995-07-01 09:30:11|1995-07-01|  burger.letters.com|[01/Jul/1995:00:0...|GET /shuttle/coun...|           304|        0|
|1995-07-01 09:30:11|1995-07-01|      199.120.110.21|[01/Jul/1995:00:0...|GET /shuttle/miss...|           200|     4179|
|1995-07-01 09:30:12|1995-07-01|  burger.letters.com|[01/Jul/1995:00:0...|GET /images/NASA-...|           304|        0|
|1995-07-01 09:30:12|1995-07-01|  burger.letters.com|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|        0|
|1995-07-01 09:30:12|1995-07-01|     205.212.115.106|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|     3985|
|1995-07-01 09:30:13|1995-07-01|         d104.aa.net|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|     3985|
|1995-07-01 09:30:13|1995-07-01|      129.94.144.152|[01/Jul/1995:00:0...|      GET / HTTP/1.0|           200|     7074|
|1995-07-01 09:30:14|1995-07-01|unicomp6.unicomp.net|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|    40310|
|1995-07-01 09:30:14|1995-07-01|unicomp6.unicomp.net|[01/Jul/1995:00:0...|GET /images/NASA-...|           200|      786|
|1995-07-01 09:30:14|1995-07-01|unicomp6.unicomp.net|[01/Jul/1995:00:0...|GET /images/KSC-l...|           200|     1204|
|1995-07-01 09:30:15|1995-07-01|         d104.aa.net|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|    40310|
|1995-07-01 09:30:15|1995-07-01|         d104.aa.net|[01/Jul/1995:00:0...|GET /images/NASA-...|           200|      786|
|1995-07-01 09:30:15|1995-07-01|         d104.aa.net|[01/Jul/1995:00:0...|GET /images/KSC-l...|           200|     1204|
|1995-07-01 09:30:17|1995-07-01|      129.94.144.152|[01/Jul/1995:00:0...|GET /images/ksclo...|           304|        0|
|1995-07-01 09:30:17|1995-07-01|      199.120.110.21|[01/Jul/1995:00:0...|GET /images/launc...|           200|     1713|
|1995-07-01 09:30:18|1995-07-01|ppptky391.asahi-n...|[01/Jul/1995:00:0...|GET /facts/about_...|           200|     3977|
|1995-07-01 09:30:19|1995-07-01|  net-1-141.eden.com|[01/Jul/1995:00:0...|GET /shuttle/miss...|           200|    34029|
+--------------------+----------+--------------------+--------------------+--------------------+--------------+---------+
only showing top 20 rows
```

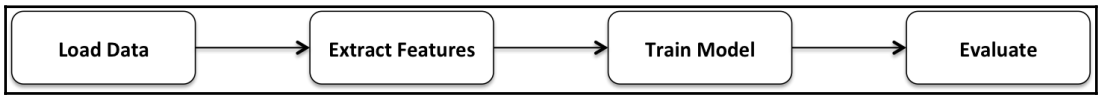
```
-------------------------------------------
Batch: 0
-------------------------------------------
+--------------------+----------+--------------------+--------------------+--------------------+--------------+----------+
|                  ts|      date|     clientIpAddress|   original_dateTime|             request|httpStatusCode|bytesSent|
+--------------------+----------+--------------------+--------------------+--------------------+--------------+----------+
|1995-07-01 09:30:01|1995-07-01|        199.72.81.55|[01/Jul/1995:00:0...|GET /history/apol...|           200|      6245|
|1995-07-01 09:30:06|1995-07-01|unicomp6.unicomp.net|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|      3985|
|1995-07-01 09:30:09|1995-07-01|      199.120.110.21|[01/Jul/1995:00:0...|GET /shuttle/miss...|           200|      4085|
|1995-07-01 09:30:11|1995-07-01|  burger.letters.com|[01/Jul/1995:00:0...|GET /shuttle/coun...|           304|         0|
|1995-07-01 09:30:11|1995-07-01|      199.120.110.21|[01/Jul/1995:00:0...|GET /shuttle/miss...|           200|      4179|
|1995-07-01 09:30:12|1995-07-01|  burger.letters.com|[01/Jul/1995:00:0...|GET /images/NASA-...|           304|         0|
|1995-07-01 09:30:12|1995-07-01|  burger.letters.com|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|         0|
|1995-07-01 09:30:12|1995-07-01|     205.212.115.106|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|      3985|
|1995-07-01 09:30:13|1995-07-01|         d104.aa.net|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|      3985|
|1995-07-01 09:30:13|1995-07-01|      129.94.144.152|[01/Jul/1995:00:0...|      GET / HTTP/1.0|           200|      7074|
|1995-07-01 09:30:14|1995-07-01|unicomp6.unicomp.net|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|     40310|
|1995-07-01 09:30:14|1995-07-01|unicomp6.unicomp.net|[01/Jul/1995:00:0...|GET /images/NASA-...|           200|       786|
|1995-07-01 09:30:14|1995-07-01|unicomp6.unicomp.net|[01/Jul/1995:00:0...|GET /images/KSC-l...|           200|      1204|
|1995-07-01 09:30:15|1995-07-01|         d104.aa.net|[01/Jul/1995:00:0...|GET /shuttle/coun...|           200|     40310|
|1995-07-01 09:30:15|1995-07-01|         d104.aa.net|[01/Jul/1995:00:0...|GET /images/NASA-...|           200|       786|
|1995-07-01 09:30:15|1995-07-01|         d104.aa.net|[01/Jul/1995:00:0...|GET /images/KSC-l...|           200|      1204|
|1995-07-01 09:30:17|1995-07-01|      129.94.144.152|[01/Jul/1995:00:0...|GET /images/ksclo...|           304|         0|
|1995-07-01 09:30:17|1995-07-01|      199.120.110.21|[01/Jul/1995:00:0...|GET /images/launc...|           200|      1713|
|1995-07-01 09:30:18|1995-07-01|ppptky391.asahi-n...|[01/Jul/1995:00:0...|GET /facts/about_...|           200|      3977|
|1995-07-01 09:30:19|1995-07-01|  net-1-141.eden.com|[01/Jul/1995:00:0...|GET /shuttle/miss...|           200|     34029|
+--------------------+----------+--------------------+--------------------+--------------------+--------------+----------+
only showing top 20 rows
```

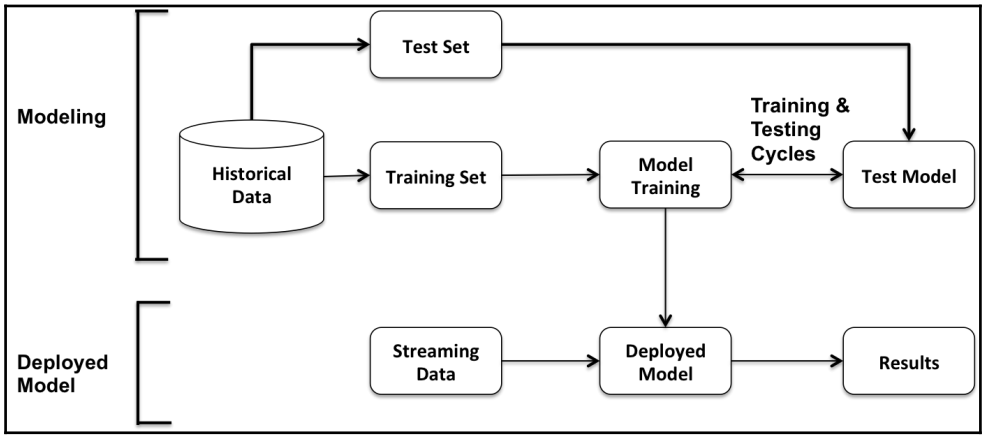

```
[{"ts":"1995-07-01T09:30:01.000+05:30","date":"1995-07-
01","clientIpAddress":"199.72.81.55","rfc1413ClientIdentity":"-","remoteUser":"-
","original_dateTime":"[01/Jul/1995:00:00:01 -0400]","request":"GET /history/apollo/
HTTP/1.0","httpStatusCode":"200","bytesSent":"6245"}]
[{"ts":"1995-07-01T09:30:06.000+05:30","date":"1995-07-
01","clientIpAddress":"unicomp6.unicomp.net","rfc1413ClientIdentity":"-
","remoteUser":"-","original_dateTime":"[01/Jul/1995:00:00:06 -0400]","request":"GET
/shuttle/countdown/ HTTP/1.0","httpStatusCode":"200","bytesSent":"3985"}]
[{"ts":"1995-07-01T09:30:09.000+05:30","date":"1995-07-
01","clientIpAddress":"199.120.110.21","rfc1413ClientIdentity":"-","remoteUser":"-
","original_dateTime":"[01/Jul/1995:00:00:09 -0400]","request":"GET
/shuttle/missions/sts-73/mission-sts-73.html
HTTP/1.0","httpStatusCode":"200","bytesSent":"4085"}]
```

```
-------------------------------------------
Batch: 0
-------------------------------------------
+--------------------+
|        parsed_value|
+--------------------+
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
|[1995-07-01 09:30...|
+--------------------+
only showing top 20 rows
```

```
-------------------------------------------
Batch: 0
-------------------------------------------
+--------------------+--------------+-----+
|              window|httpStatusCode|count|
+--------------------+--------------+-----+
|[1995-07-19 14:20...|           200|  220|
|[1995-07-21 02:35...|           404|    4|
|[1995-07-17 17:00...|           200|  318|
|[1995-07-23 09:00...|           304|   22|
|[1995-07-27 06:10...|           404|    2|
|[1995-07-15 16:35...|           404|    2|
|[1995-07-27 00:15...|           200|  574|
|[1995-07-14 20:55...|           302|   28|
|[1995-07-05 08:00...|           404|    1|
|[1995-07-18 20:10...|           404|    2|
|[1995-07-27 06:05...|           302|    8|
|[1995-07-18 09:00...|           404|    2|
|[1995-07-19 05:40...|           200|  367|
|[1995-07-10 09:30...|           200|  334|
|[1995-07-26 18:55...|           302|    3|
|[1995-07-13 06:25...|           304|   71|
|[1995-07-21 02:30...|           302|   18|
|[1995-07-14 17:40...|           404|    1|
|[1995-07-15 06:00...|           302|   18|
|[1995-07-06 13:30...|           200|  270|
+--------------------+--------------+-----+
only showing top 20 rows
```

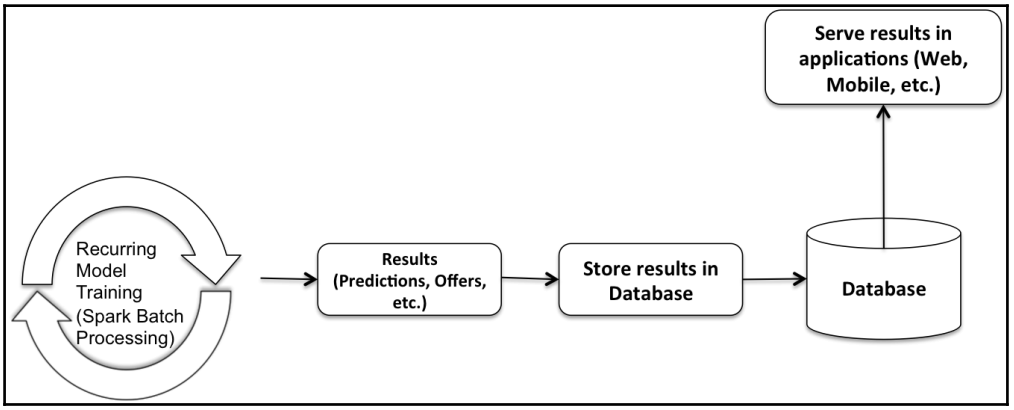
```
-------------------------------------------
Batch: 0
-------------------------------------------
+--------------------+--------------------+-----+
|              window|             request|count|
+--------------------+--------------------+-----+
|[1995-07-04 01:20...|GET /software/win...|    1|
|[1995-07-12 09:30...|GET /shuttle/miss...|    5|
|[1995-07-13 00:30...|GET /images/launc...|    9|
|[1995-07-06 02:50...|GET /shuttle/tech...|    1|
|[1995-07-17 04:50...|GET /shuttle/miss...|    4|
|[1995-07-03 19:50...|GET /shuttle/miss...|    1|
|[1995-07-11 02:35...|GET /software/win...|    2|
|[1995-07-11 07:40...|GET /elv/DELTA/de...|    1|
|[1995-07-26 19:25...|GET /shuttle/coun...|    1|
|[1995-07-17 20:55...|GET /history/apol...|    1|
|[1995-07-24 02:20...|GET /history/gemi...|    1|
|[1995-07-01 23:30...|GET /shuttle/miss...|    3|
|[1995-07-18 00:50...|GET /history/apol...|    1|
|[1995-07-09 22:25...|GET /shuttle/miss...|    1|
|[1995-07-20 10:25...|GET /history/apol...|    2|
|[1995-07-15 01:25...|GET /shuttle/miss...|    1|
|[1995-07-07 12:55...|GET /cgi-bin/imag...|    1|
|[1995-07-13 12:05...|GET /shuttle/miss...|    1|
|[1995-07-13 19:15...|GET /shuttle/miss...|    1|
|[1995-07-19 20:20...|GET /shuttle/miss...|    2|
+--------------------+--------------------+-----+
only showing top 20 rows
```

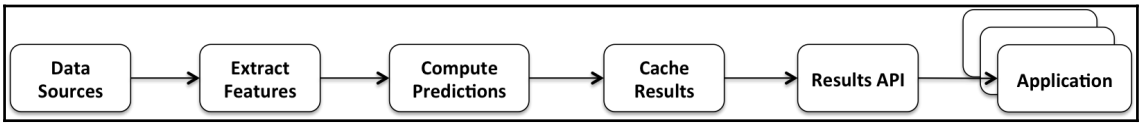


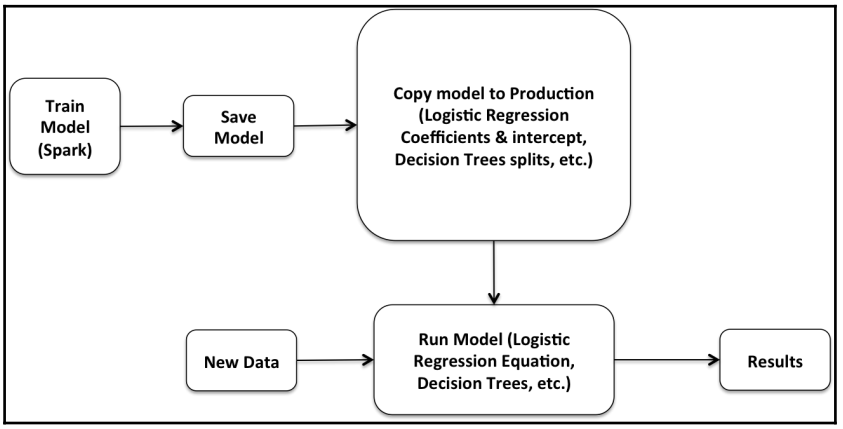



```
+----------+-----------+--------------------+--------------------+-------------+
|numClasses|numFeatures|     interceptVector|  coefficientMatrix|isMultinomial|
+----------+-----------+--------------------+--------------------+-------------+
|         2|     262144|[-1.5906514317596...|1 x 262144 CSCMat...|        false|
+----------+-----------+--------------------+--------------------+-------------+
```

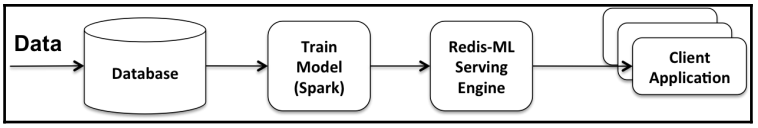
```
[2,262144,[-1.5906514317596054],1  x 262144  CSCMatrix
(0,15554)  1.6097382089819763
(0,17222)  2.7668185824796496
(0,24152)  1.6097382089819763
(0,27526)  -1.9454856869974295
(0,28698)  2.7668185824796496
(0,30913)  -1.9454856869974295
(0,42633)  -2.381670949064051
(0,51505)  1.6097382089819763
(0,155117)  -2.381670949064051
(0,227410)  2.7668185824796496
(0,234657)  3.28241759359622,false]
```

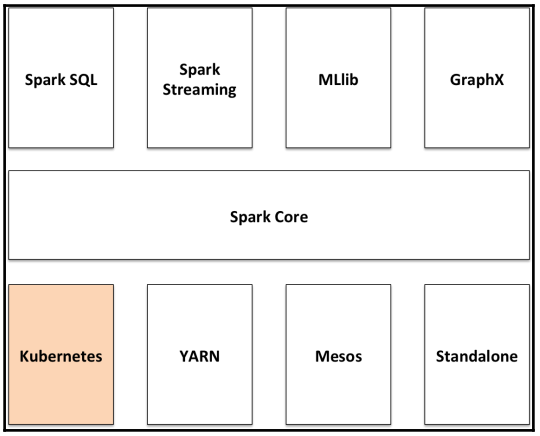


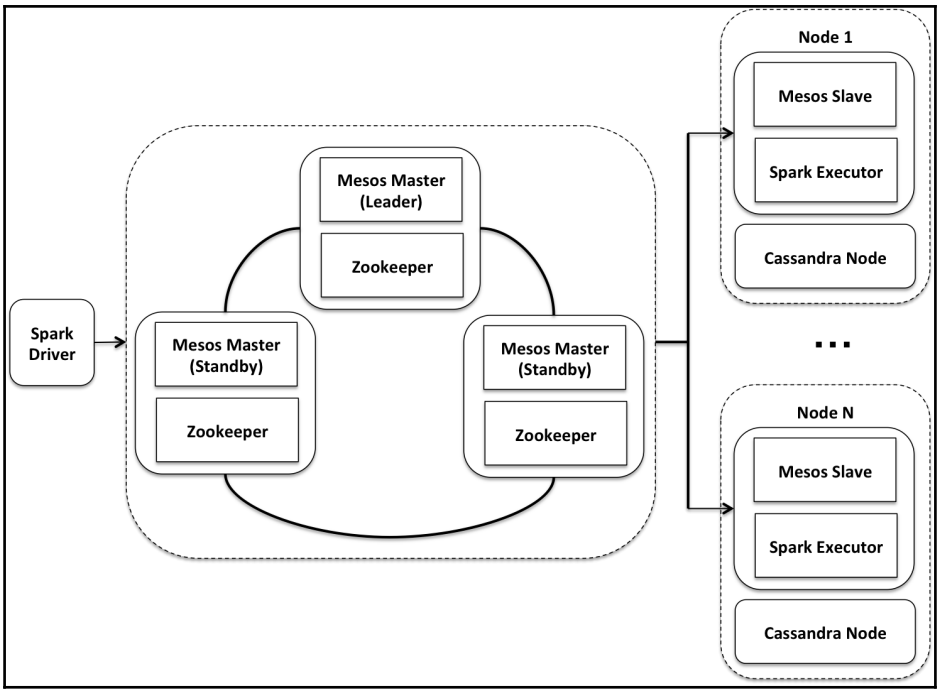


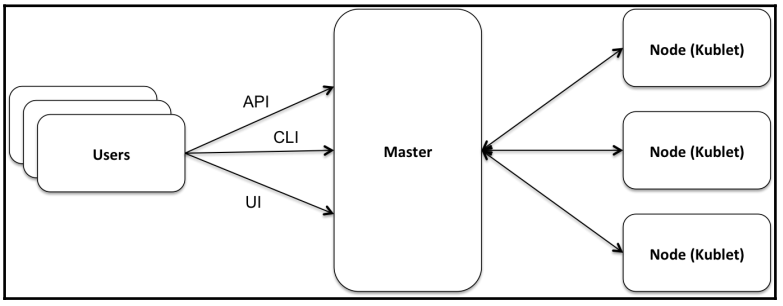


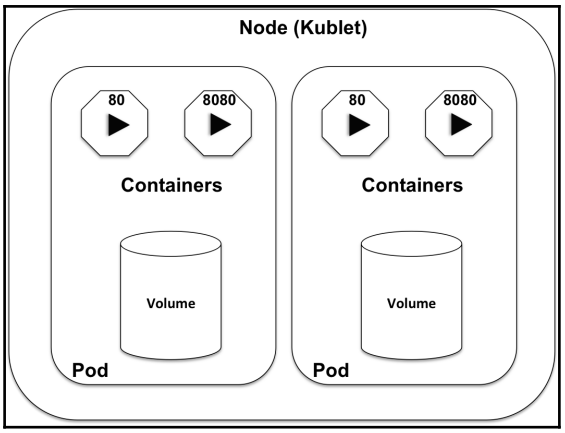
```
{"numClasses":2,"numFeatures":262144,"interceptVector":{"type":1,"values":[-
1.5906514317596054]},"coefficientMatrix":{"type":0,"numRows":1,"numCols":262144,"colPt
rs":[0,11],"rowIndices":[15554,17222,24152,27526,28698,30913,42633,51505,155117,227410
,234657],"values":[1.6097382089819763,2.7668185824796496,1.6097382089819763,-
1.9454856869974295,2.7668185824796496,-1.9454856869974295,-
2.381670949064051,1.6097382089819763,-
2.381670949064051,2.7668185824796496,3.28241759359622],"isTransposed":true},"isMultino
mial":false}
```




Node 1
Mesos Slave
Spark Executor
Cassandra Node
Mesos Master
(Leader)
Zookeeper
Spark
Driver
Mesos Master
(Standby)
Zookeeper
Mesos Master
(Standby)
Zookeeper
Node N
Mesos Slave
Spark Executor
Cassandra Node


Users
API
CLI
UI
Master
Node (Kublet)
Node (Kublet)
Node (Kublet)


Node (Kublet)
80
8080
Containers
Volume
Pod
80
8080
Containers
Volume
Pod

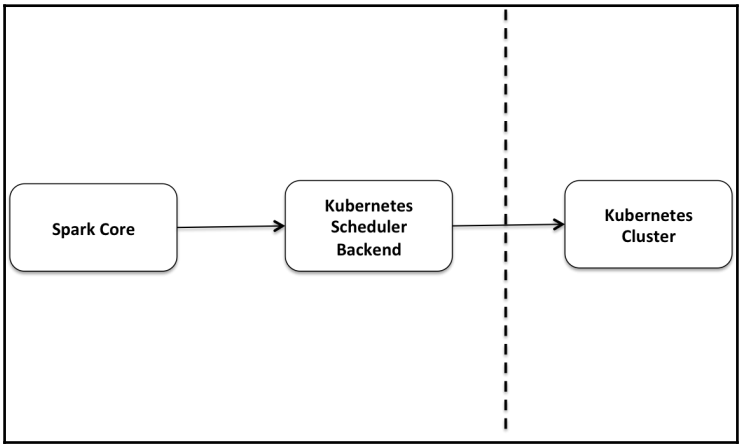

Spark Core
Kubernetes Scheduler Backend
Kubernetes Cluster

Table of Contents

Index