ACADGILD

Learn. Do. Earn.

CATEGORIES ▾

□ Home / Spark / Analyzing New York Crime Data Using SparkSQL

28
APRIL
2016

Analyzing New York Crime Data Using SparkSQL

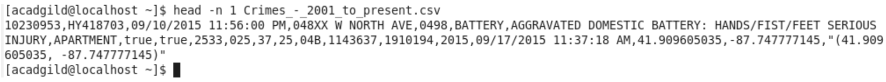
In this post, we will be analyzing the crimes dataset of New York using SparkSQL. In case you are not familiar with SparkSQL, please refer to our post on Introduction to SparkSQL.

Dataset Description:

This dataset is available publically, reflects the reported incidents of crime (with the exception of murders, where data exists for each victim) that has occurred in the City of Chicago from 2001 to present. The data is extracted from the New York Police Department's CLEAR (Citizen Law Enforcement Analysis and Reporting) system.

You can download the dataset from here.

Below is a sample record from the dataset



```
[acadgild@localhost ~]$ head -n 1 Crimes_-_2001_to_present.csv
10230953,HY418703,09/10/2015 11:56:00 PM,048XX W NORTH AVE,0498,BATTERY,AGGRAVATED DOMESTIC BATTERY: HANDS/FIST/FEET SERIOUS
INJURY,APARTMENT,true,true,2533,025,37,25,04B,1143637,1910194,2015,09/17/2015 11:37:18 AM,41.909605035,-87.747777145,"(41.909
605035, -87.747777145)"
[acadgild@localhost ~]$
```

You can click here for the complete data set column wise description.

In this post, we will be using pyspark shell for writing our queries.

Problem Statement:

1. **Find number of crimes that happened under each FBI code.**

Code:

```
#import SQLContext and row

from pyspark.sql import SQLContext,Row

sqlContext=SQLContext(sc)

#load the data set and split the records

lines =sc.textFile("hdfs://localhost:9000/Crime_dataset")

parts = lines.map(lambda l: l.split(","))

# construct the Rows by by passing a list of key/value pairs as kwargs

Crimes = parts.map(lambda p:Row(Id =p[0],case_no=p[1],date=p[2],block=p[3],IUCR=p[4],Primary_type=p[5],description=p[6],Loc_des =p[7],arrest=p[8],domestic= p[9],beat=p[10],district=p[11],ward=p[12],community=p[13],fbicode=p[14],XCor=p[15],YCor=p[16],year=p[17],Updated_on=p[18],latitude=p[19],longi=p[20],loc=p[21]))

# Create the DataFrame and register it has Table

schema1=sqlContext.createDataFrame(Crimes)

schema1.registerTempTable("Crimes")

#run the query for getting the required result

result=sqlContext.sql("select fbicode,count(fbicode) as count from Crimes group by fbicode")

result.show()
```

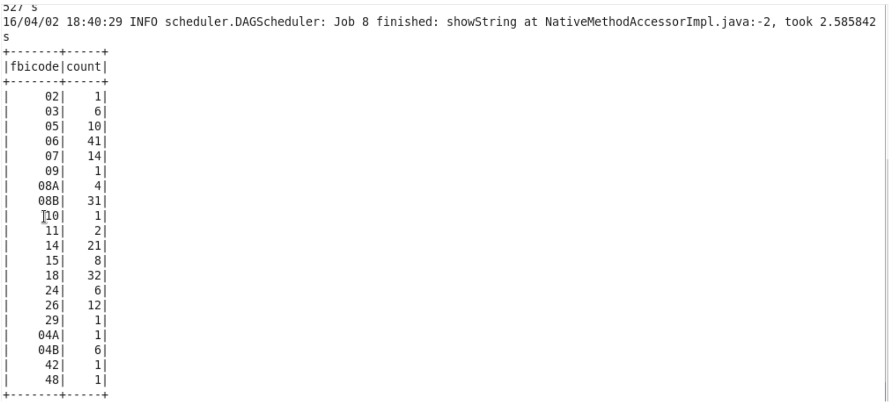
SEARCH

ﬂ Search Now

CATEGORIES

- AcadGild
- Android
- Android For Kids
- AngularJS
- Big Data and Hadoop
- Careers
- Cloud computing
- Database
- Digital Marketing
- Front End
- Full Stack
- Hadoop Administration
- IOS
- Java
- Kids
- Linux Administration
- NodeJS
- Others
- Python
- Quiz

Output:



```
527 s
16/04/02 18:40:29 INFO scheduler.DAGScheduler: Job 8 finished: showString at NativeMethodAccessorImpl.java:-2, took 2.585842
s
+-------+-----+
|fbicode|count|
+-------+-----+
|     02|    1|
|     03|    6|
|     05|   10|
|     06|   41|
|     07|   14|
|     09|    1|
|    08A|    4|
|    08B|   31|
|     10|    1|
|     11|    2|
|     14|   21|
|     15|    8|
|     18|   32|
|     24|    6|
|     26|   12|
|     29|    1|
|    04A|    1|
|    04B|    6|
|     42|    1|
|     48|    1|
+-------+-----+
```

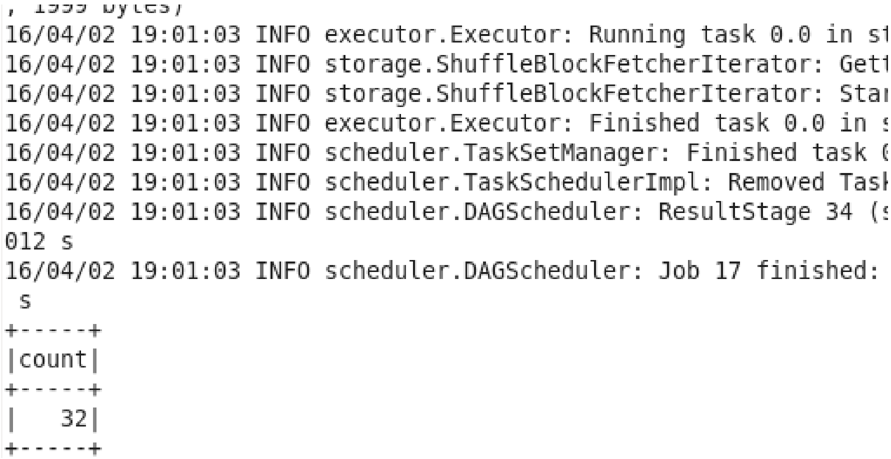
2. Find number of ‘NARCOTICS’ cases filed in the year 2015.

We have already read the data created from the Data Frame and registered as a table with the name ‘Çrimes’, in the first problem statement. Now, we can directly run the queries on this table.

Query:

```
result=sqlContext.sql("select count(*) as count from Crimes where Primary_type ='NARCOTICS' and year = 2015 ")

result.show()
```



```
, 1999 bytes)
16/04/02 19:01:03 INFO executor.Executor: Running task 0.0 in st
16/04/02 19:01:03 INFO storage.ShuffleBlockFetcherIterator: Gett
16/04/02 19:01:03 INFO storage.ShuffleBlockFetcherIterator: Star
16/04/02 19:01:03 INFO executor.Executor: Finished task 0.0 in s
16/04/02 19:01:03 INFO scheduler.TaskSetManager: Finished task 0
16/04/02 19:01:03 INFO scheduler.TaskSchedulerImpl: Removed Task
16/04/02 19:01:03 INFO scheduler.DAGScheduler: ResultStage 34 (s
012 s
16/04/02 19:01:03 INFO scheduler.DAGScheduler: Job 17 finished:
 s
+-----+
|count|
+-----+
|   32|
+-----+
```

3. Find the number of theft related arrests that happened in each district.

- R & Machine Learning
- Scala
- Spark
- Uncategorized

GET SOCIAL

AcadGild
Local Business · Bangal
92,286 likes

Like Page

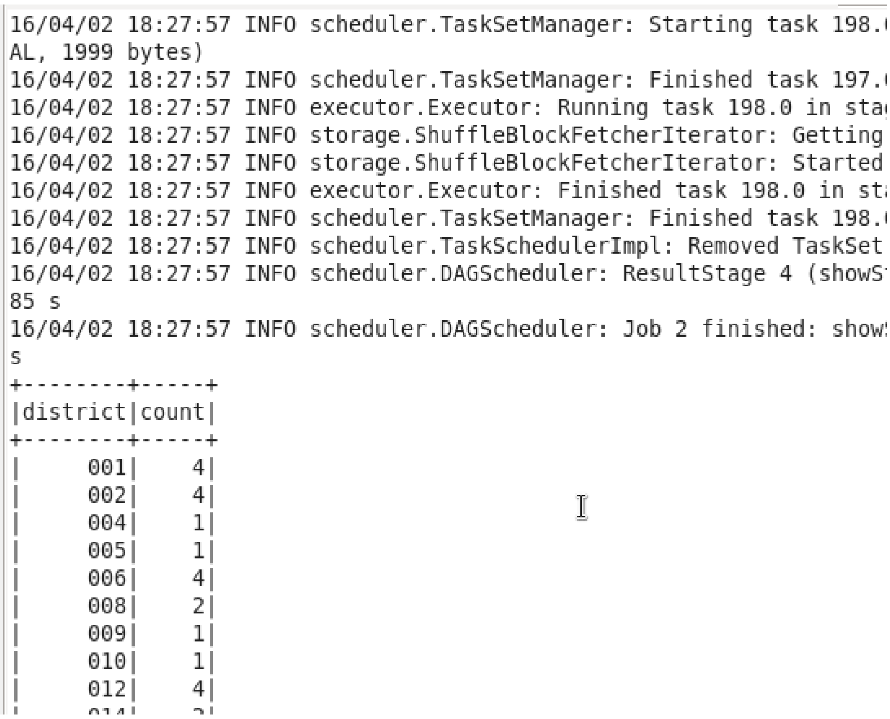
3 friends like this

AcadGild shared a li
1 hr

Spark Use Case – U
In this post, we will be perform
ACADGILD.COM

WHAT'S TRENDING

result=sqlContext.sql(“select district ,count(*) as count from Crimes where Primary_type =’THEFT’ and arrest = ‘true’ group by district “) result.show()



```
16/04/02 18:27:57 INFO scheduler.TaskSetManager: Starting task 198.(
AL, 1999 bytes)
16/04/02 18:27:57 INFO scheduler.TaskSetManager: Finished task 197.(
16/04/02 18:27:57 INFO executor.Executor: Running task 198.0 in sta
16/04/02 18:27:57 INFO storage.ShuffleBlockFetcherIterator: Getting
16/04/02 18:27:57 INFO storage.ShuffleBlockFetcherIterator: Started
16/04/02 18:27:57 INFO executor.Executor: Finished task 198.0 in st
16/04/02 18:27:57 INFO scheduler.TaskSetManager: Finished task 198.(
16/04/02 18:27:57 INFO scheduler.TaskSchedulerImpl: Removed TaskSet
16/04/02 18:27:57 INFO scheduler.DAGScheduler: ResultStage 4 (showS
85 s
16/04/02 18:27:57 INFO scheduler.DAGScheduler: Job 2 finished: show
s
+--------+-----+
|district|count|
+--------+-----+
|     001|    4|
|     002|    4|
|     004|    1|
|     005|    1|
|     006|    4|
|     008|    2|
|     009|    1|
|     010|    1|
|     012|    4|
|     014|    2|
```

We hope this blog helped you in getting grip over SparkSQL concepts.Keep visiting our website for more blogs on Big Data,Spark and other technologies.

Share this:

7

Related

Spark Use Case – The Daily Show
April 16, 2016
In "Spark"

Spark SQL - Module for Structured Data Processing
April 22, 2016
In "Big Data and Hadoop"

Bucketing in Hive
April 7, 2016
In "Big Data and Hadoop"

Tweets by @acadgild

ACADGILD @acadgild

#Spark Use Case - Uber #DataAnalysis buff.ly/1NvpXDN

Spark…
In this …
acadg…

8s

ACADGILD @acadgild

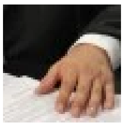
#TechTipTuesday:

Embed View on Twitter

RECENT POSTS

Spark Use Case – Uber Data Analysis
□ May 16, 2016

Why Learning MongoDB Will Boost Your Career
□ May 14, 2016



Job Responsibilities of Hadoop Professionals
□ May 13, 2016

Graphical Exploratory Data Analysis-II
□ May 13, 2016

BRUNDESH

Brundesh R currently working at AcadGild is an expert in Big Data domain with 3.5 years of Industry experience. He has rich experience in Hadoop, R, Python, Java . He has published several blogs and articles on Hadoop,Spark and have undertaken projects on Hadoop platform. AcadGild was founded with the vision of "Learn. Do. Earn". We provide skill development courses based on current industry needs. But what sets us apart is earning opportunities we provide after successful completion of course. We also provide live mentoring and 24x7 support. Our mentors are industry thought leaders in their respective fields

PREVIOUS ARTICLE

Spark RDD Operations in Scala Part – 2

NEXT ARTICLE

Beginners guide to FTP & SFTP Server Configuration

RELATED POSTS

Spark Use Case – Uber Data Analysis

May 16, 2016

Integrating SparkSQL with MySQL

May 12, 2016

Spark Use Case – Travel Data Analysis

May 8, 2016

LEAVE A REPLY

COMMENTS *

ARCHIVES

- May 2016
- April 2016
- March 2016
- February 2016
- January 2016
- December 2015
- November 2015
- September 2015
- August 2015
- July 2015
- June 2015
- May 2015
- November 2014
- October 2014
- September 2014
- August 2014

NAME *

EMAIL *

WEBSITE

SUBMIT

□ **NOTIFY ME OF FOLLOW-UP COMMENTS BY EMAIL.**

□ **NOTIFY ME OF NEW POSTS BY EMAIL.**

CATEGORIES

- AcadGild
- Android
- Android For Kids
- AngularJS
- Big Data and Hadoop
- Careers
- Cloud computing
- Database
- Digital Marketing
- Front End
- Full Stack

TAGS

- ANDORID PROFILING TOOLS
- ANDROID APP FOR SPEECH TO TEXT
- ANDROID APP FOR TEXT TO SPEECH
- ANDROID DEVELOPMENT
- ANDROID MEMORY ANALYZER
- ANDROID MEMORY MANAGEMENT
- BANGALORE SUMMER CAMP
- BEST SUMMER CAMPS 2016

LIKE WHAT YOU SEE? SUBSCRIBE TO OUR BLOG

We send only 1 email in a week

Enter your email...

Subscribe

- Hadoop Administration
- IOS
- Java
- Kids
- Linux Administration
- NodeJS
- Others
- Python
- Quiz
- R & Machine Learning
- Scala
- Spark
- Uncategorized

BIG DATA DEVELOPEMENT

COMMISSIONING AND DECOMMISSIONING OF DATANODE IN HADOOP

DEPENDENCY INJECTION

DIFFERENCE BETWEEN ANDROID VS IOS

FEATURES OF DDMS

FILE FORMATS

FILE FORMATS IN HADOOP

HADOOP

HADOOP ADMINISTRATION

HDFS

HIVE WITH MYSQL

INTRODUCTION TO SPARK

JAVASCRIPT MVC FRAMEWORK

JOB OPPORTUNITIES IN HADOOP

JOB TRENDS IN BIG DATA BLOG

JOB TRENDS IN HADOOP

LINUX

LINUX BASIC

LINUX BASIC COMMANDS

LINUX COMMANDS

MYSQL

MYSQL-CONNECTOR-JAVA-5.1.2.JAR

MYSQL-CONNECTOR-JAVA.JAR

MYSQL WITH HIVE

MYSQL WITH SQOOP

PYTHON

RACK AWARENESS

RECYCLE BIN

RESILIENT DISTRIBUTED DATASET (RDD)

SPARK

SQOOP

SQOOP WITH MYSQL

STYLING A RESPONSIVE WEB PAGE

SUMMER CAMP

SUMMER CAMP 2016

TOP 10 RECORDS IN MAPREDUCE

TRASH CONFIGURATION