

Project Lab - Load Full Data-Set to Hive

Deliverables

Screen shot of provisioned EMR cluster from AWS EMR dashboard:


Cluster: EMR_1_4 Waiting Cluster ready after last step completed.


[Summary](#) [Application history](#) [Monitoring](#) [Hardware](#) [Configurations](#) [Events](#) [Steps](#) [Bootstrap actions](#)

Connections: [Enable Web Connection](#) – Hue, Spark History Server, JupyterHub, Resource Manager ... (View All)

Master public DNS: ec2-18-224-199-115.us-east-2.compute.amazonaws.com [SSH](#)

Tags: Name = EMR_1_4 [View All](#) / [Edit](#)

Summary	Configuration details
ID: j-25O7Y1IDRN7LT	Release label: emr-5.22.0
Creation date: 2019-04-09 21:56 (UTC-5)	Hadoop distribution: Amazon 2.8.5
Elapsed time: 36 minutes	Applications: Hive 2.3.4, Pig 0.17.0, Hue 4.3.0, JupyterHub 0.9.4, Spark 2.4.0
Auto-terminate: No	Log URI: s3://aws-logs-930103565420-us-east-2/elasticmapreduce/ 
Termination protection: Off Change	EMRFS consistent view: Disabled
	Custom AMI ID: --

Network and hardware	Security and access
Availability zone: us-east-2c	Key name: EMR_EC2_key_ADD_MORE_HERE
Subnet ID: subnet-d201df9e external link icon	EC2 instance profile: EMR_EC2_DefaultRole
Master: Running 1 m4.large	EMR role: EMR_DefaultRole
Core: Running 2 m4.large Spot (max on-demand)	Auto Scaling role: EMR_AutoScaling_DefaultRole
Task: --	Visible to all users: All Change
	<div> This feature will be deprecated soon.</div>
	Security groups for sg-6d3eb403 external link icon (default) Master:
	Security groups for sg-6d3eb403 external link icon (default) Core & Task:

Screen shot/shots from the EMR Master terminal, clearly showing all executed commands:

1) Download full dataset and save in Windows:

```
[hadoop@ip-172-31-43-170 ~]$ wget --no-check-certificate https://data.consumerfinance.gov/api/views/s6ew-h6mp/rows.csv?accessType=DOWNLOAD
--2019-04-11 04:18:27-- https://data.consumerfinance.gov/api/views/s6ew-h6mp/rows.csv?accessType=DOWNLOAD
Resolving data.consumerfinance.gov (data.consumerfinance.gov)... 52.206.140.205, 52.206.68.26, 52.206.140.199
Connecting to data.consumerfinance.gov (data.consumerfinance.gov)|52.206.140.205|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/csv]
Saving to: 'rows.csv?accessType=DOWNLOAD'

rows.csv?accessType=DOWNLOAD [ 690.49M 5.45MB/s in 86s]
2019-04-11 04:19:57 (8.01 MB/s) - 'rows.csv?accessType=DOWNLOAD' saved [724033536]
[hadoop@ip-172-31-43-170 ~]$ mv 'rows.csv?accessType=DOWNLOAD' cust_complains_all_through_3_31.csv
```

2) Pre-process data set to remove special characters and reformat to tab delimited format

The .csv file that was downloaded in the above step pre-processed to format it to tab delimited format using python3 as given below:

```
In [1]: import csv

In [2]: input_file_name = "cust_complains_all_through_3_31.csv"
        output_file_name = "cust_complains_all_through_3_31_tab.csv"

In [3]: def remove_n_r(column):
        column = column.replace('\n','')
        column = column.replace('\r','')
        column = column.replace('\t','')
        return column

In [4]: in_txt = csv.reader(open(input_file_name, encoding="utf8"), delimiter = ',')

In [5]: out_csv = open(output_file_name, 'w', newline='\n', encoding="utf8")
        csv_writer = csv.writer(out_csv, delimiter = '\t', quotechar='|', quoting=csv.QUOTE_MINIMAL)
        line_count = 0
        for row in in_txt:
            if line_count == 0:
                print('Column names are:', row)
                line_count += 1
            else:
                new_row = []
                for col in row:
                    new_row.append(remove_n_r(col))
                csv_writer.writerow(new_row)
                line_count += 1
        out_csv.close()

Column names are: ['Date received', 'Product', 'Sub-product', 'Issue', 'Sub-issue', 'Consumer complaint narrative', 'Company public response', 'Company', 'State', 'ZIP code', 'Tags', 'Consumer consent provided?', 'Submitted via', 'Date sent to company', 'Company response to consumer', 'Timely response?', 'Consumer disputed?', 'Complaint ID']

In [6]: print('Processed',line_count, ' lines.')
Processed 1256553 lines.
```

From the above screenshot it can be said that 1256553 records were processed. Thus, the output file “cust_complains_all_through_3_31_tab.csv” consists of 1256552 records and a header. This output file is then uploaded in aws S3 in location : S3://dataset330/

3) Create database customer_complaints and External table cust_complaints_all_raw as below:

```
hive> create database customer_complaints;
OK
Time taken: 0.074 seconds
hive> CREATE EXTERNAL TABLE customer_complaints.cust_complaints_all_raw
> (date_received      string,
> product             string,
> sub_product         string,
> issue              string,
> sub_issue           string,
> consumer_complaint_narrative string,
> company_public_response string,
> company            string,
> state              string,
> zip_code           string,
> tags               string,
> consumer_consent_provided string,
> submitted_via      string,
> date_sent_to_company string,
> company_response    string,
> timely_response     string,
> consumer_disputed   string,
> complaint_id        string)
> ROW FORMAT DELIMITED
> FIELDS TERMINATED BY '\t'
> STORED AS
> INPUTFORMAT
> 'com.amazonaws.emr.s3select.hive.S3SelectTableTextInputFormat'
> OUTPUTFORMAT
> 'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat'
> LOCATION 's3://dataset330/Complaints'
> TBLPROPERTIES (
>   "s3select.format" = "csv",
>   "s3select.headerInfo" = "ignore"
> );
OK
Time taken: 0.631 seconds
hive> █
```

The data that will be loaded in this table must be tab delimited. Also, the data that will be loaded will be saved in location “S3://dataset330/Complaints” as shown in the screenshot below:

Amazon S3 > dataset330

Overview	Properties	Permissions	Management
----------	------------	-------------	------------

Q Type a prefix and press Enter to search. Press ESC to clear.

Upload Create folder Download Actions

US East (Ohio)

Viewing 1 to 4			
<input type="checkbox"/> Name	Last modified	Size	Storage class
<input type="checkbox"/> Complaint_processed.csv	Apr 5, 2019 4:54:10 PM GMT-0500	18.8 MB	Standard
<input type="checkbox"/> Complaints_\$folder\$	Apr 9, 2019 10:57:09 PM GMT-0500	0 B	Standard
<input type="checkbox"/> cust_complaints_all_through_3_31.csv	Apr 9, 2019 4:36:21 PM GMT-0500	690.0 MB	Standard
<input type="checkbox"/> cust_complaints_all_through_3_31_tab.csv	Apr 9, 2019 11:01:49 PM GMT-0500	687.7 MB	Standard

Viewing 1 to 4

4) Load data in the External Table:

The below code was executed to load data in the External Table.

```
hive> LOAD DATA inpath 's3://dataset330/cust_complains_all_through_3_31_tab.csv'
> INTO TABLE customer_complaints.cust_complaints_all_raw;
Loading data to table customer_complaints.cust_complaints_all_raw
OK
Time taken: 5.115 seconds
```

This is a one-time process and this step is not required to be executed every time this table is created as location is mentioned while creating this external table.

As cluster is terminated, all the tables and databases created get lost. So, whenever a new cluster is created, step 2 is required to be executed every time. As data is now already stored in the location mentioned while creating the external table, data is automatically loaded in the table.

The status of AWS S3://dataset330/ looks like below:

The screenshot displays the Amazon S3 console interface. The top navigation bar shows 'Amazon S3' and the current path 'dataset330'. Below this, there are tabs for 'Overview', 'Properties', 'Permissions', and 'Management'. The 'Overview' tab is selected, showing a search bar and a list of objects. The objects listed are 'Complaints', 'Complaint_processed.csv', 'Complaints_\$folder\$', and 'cust_complains_all_through_3_31.csv'. The 'Complaints' folder is expanded, showing its contents: 'Complaint_processed.csv', 'Complaints_\$folder\$', and 'cust_complains_all_through_3_31.csv'. The 'Overview' tab is selected, showing a search bar and a list of objects. The objects listed are 'Complaints', 'Complaint_processed.csv', 'Complaints_\$folder\$', and 'cust_complains_all_through_3_31.csv'. The 'Complaints' folder is expanded, showing its contents: 'Complaint_processed.csv', 'Complaints_\$folder\$', and 'cust_complains_all_through_3_31.csv'.

Name	Last modified	Size	Storage class
Complaints	--	--	--
Complaint_processed.csv	Apr 5, 2019 4:54:10 PM GMT-0500	18.8 MB	Standard
Complaints_\$folder\$	Apr 9, 2019 10:57:09 PM GMT-0500	0 B	Standard
cust_complains_all_through_3_31.csv	Apr 9, 2019 4:36:21 PM GMT-0500	690.0 MB	Standard

Viewing 1 to 4

Amazon S3 > dataset330 > Complaints

Overview

Name	Last modified	Size	Storage class
cust_complains_all_through_3_31_tab.csv	Apr 9, 2019 11:14:01 PM GMT-0500	687.7 MB	Standard

Viewing 1 to 1

cust_complains_all_through_3_31_tab.csv file is moved to the new bucket – “Complaints”.

5) Create table to store data in PARQUET format

```
hive> CREATE TABLE customer_complaints.cust_complaints_all
> (date_received      string,
> product             string,
> sub_product         string,
> issue              string,
> sub_issue           string,
> consumer_complaint_narrative string,
> company_public_response string,
> company            string,
> state              string,
> zip_code           string,
> tags               string,
> consumer_consent_provided string,
> submitted_via       string,
> date_sent_to_company string,
> company_response    string,
> timely_response     string,
> consumer_disputed   string,
> complaint_id        string)
> ROW FORMAT DELIMITED
> FIELDS TERMINATED BY ','
> STORED AS PARQUET
> TBLPROPERTIES ("parquet.compression"="SNAPPY");
OK
Time taken: 0.111 seconds
```

Create a table “cust_complaints_all” and insert data in it from table cust_complaints_all_raw as shown below:

```
hive> INSERT INTO customer_complaints.cust_complaints_all
> SELECT * FROM customer_complaints.cust_complaints_all_raw;
Query ID = hadoop_20190410042039_26406640-7b82-4d98-af04-2b2e7fa7df33
Total jobs = 1
Launching Job 1 out of 1
Tez session was closed. Reopening...
Session re-established.
Status: Running (Executing on YARN cluster with App id application_1554865673450_0003)

-----
VERTICES      MODE      STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
-----
Map 1 ..... container  SUCCEEDED    11      11         0         0         0         0
-----
VERTICES: 01/01  [=====>>>] 100%  ELAPSED TIME: 54.50 s
-----
Loading data to table customer_complaints.cust_complaints_all
OK
Time taken: 65.861 seconds
hive> █
```

Run basic queries:

Data has been loaded into table cust_complaints_all. Let's run some basic queries:

- (i) Display first 5 records:

```
hive> select * from customer_complaints.cust_complaints_all limit 5;
OK
02/13/2015      Mortgage      Conventional fixed mortgage      Loan servicing, payments, escrow account 0
CWEN LOAN SERVICING LLC VA      239XX      N/A      Web      02/13/2015      Closed with explanation Ye
s      No      1239425
02/13/2015      Debt collection Medical False statements or representation      Attempted to collect wrong
amount      Global Financial Services Group SC      29073      N/A      Web      02/21/2015
Closed with monetary relief      Yes      No      1239104
02/13/2015      Debt collection I do not know      Cont'd attempts collect debt not owed      Debt is not mine L
egal Prevention Services, LLC. NY      109XX      N/A      Web      02/13/2015      Closed with non-mo
netary relief      Yes      Yes      1240053
02/13/2015      Debt collection Other (i.e. phone, health club, etc.)      Cont'd attempts collect debt not o
wed      Debt is not mine      AllianceOne Recievables      Management      WA      98021      N
/A      Web      02/24/2015      Closed with explanation Yes      No      1240181
02/13/2015      Debt collection Credit card      Cont'd attempts collect debt not owed      Debt resulted from
identity theft      ENCORE CAPITAL GROUP INC.      IL      608XX      N/A      Web      02
/13/2015      Closed with non-monetary relief Yes      No      1239003
Time taken: 0.15 seconds, Fetched: 5 row(s)
hive>
```

- (ii) Display total no of records in the table:

```
hive> select count(*) as Total_Complaints from customer_complaints.cust_complaints_all;
OK
1256552
Time taken: 0.436 seconds, Fetched: 1 row(s)
hive>
```

- (iii) Display top 10 companies against which most complaints have been filed along with count:

```
hive> select company, count(*) as Count from customer_complaints.cust_complaints_all group by company orde
r by Count desc limit 10;
Query ID = hadoop_20190410043331_f4047638-1c24-497f-8941-2b5ea915e08b
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1554865673450_0004)

-----
VERTICES      MODE      STATUS      TOTAL      COMPLETED      RUNNING      PENDING      FAILED      KILLED
-----
Map 1 ..... container      SUCCEEDED      7      7      0      0      0      0
Reducer 2 ..... container      SUCCEEDED      1      1      0      0      0      0
Reducer 3 ..... container      SUCCEEDED      1      1      0      0      0      0
-----
VERTICES: 03/03 [=====>>>] 100% ELAPSED TIME: 17.65 s
-----
OK
EQUIFAX, INC.      112203
Experian Information Solutions Inc.      100584
TRANSUNION INTERMEDIATE HOLDINGS, INC.      93334
BANK OF AMERICA, NATIONAL ASSOCIATION      81346
WELLS FARGO & COMPANY      70098
JPMORGAN CHASE & CO.      59332
CITIBANK, N.A.      48324
CAPITAL ONE FINANCIAL CORPORATION      33892
Navient Solutions, LLC.      28923
OCWEN LOAN SERVICING LLC      27642
Time taken: 18.25 seconds, Fetched: 10 row(s)
hive>
```

- (iv) Display top 10 product and sub_product that have been complained about along with count:

```
hive> select product, sub_product, count(*) as Count from customer_complaints.cust_complaints_all group by
product, sub_product order by Count desc limit 10;
Query ID = hadoop_20190410043753_48493c06-395e-4b0b-9ed2-07e6e7665ba2
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1554865673450_0004)
```

VERTICES	MODE	STATUS	TOTAL	COMPLETED	RUNNING	PENDING	FAILED	KILLED
Map 1	container	SUCCEEDED	7	7	0	0	0	0
Reducer 2	container	SUCCEEDED	1	1	0	0	0	0
Reducer 3	container	SUCCEEDED	1	1	0	0	0	0

```
VERTICES: 03/03 [=====>>>] 100% ELAPSED TIME: 16.47 s
OK
Credit reporting, credit repair services, or other personal consumer reports    Credit reporting    20
9739
Credit reporting    140432
Credit card    89190
Mortgage    Other mortgage    86636
Mortgage    Conventional fixed mortgage    70613
Bank account or service    Checking account    59045
Debt collection I do not know    49134
Debt collection Other (i.e. phone, health club, etc.)    44555
Credit card or prepaid card    General-purpose credit card or charge card    35071
Mortgage    FHA mortgage    31334
Time taken: 17.623 seconds, Fetched: 10 row(s)
hive>
```

- (v) Display top 10 issue and sub_issue that have been reported along with count:

```
hive> select issue, sub_issue, count(*) as Count from customer_complaints.cust_complaints_all
> group by issue, sub_issue
> order by Count desc
> limit 10;
Query ID = hadoop_20190411043032_3d14da60-e988-4e27-a165-83f2a40750c2
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1554955694799_0001)
```

VERTICES	MODE	STATUS	TOTAL	COMPLETED	RUNNING	PENDING	FAILED	KILLED
Map 1	container	SUCCEEDED	7	7	0	0	0	0
Reducer 2	container	SUCCEEDED	1	1	0	0	0	0
Reducer 3	container	SUCCEEDED	1	1	0	0	0	0

```
VERTICES: 03/03 [=====>>>] 100% ELAPSED TIME: 18.28 s
OK
Loan modification, collection, foreclosure    112311
Loan servicing, payments, escrow account    77333
Incorrect information on your report    Information belongs to someone else    54500
Account opening, closing, or management    37961
Incorrect information on credit report    Account status    37057
Cont'd attempts collect debt not owed    Debt is not mine    36731
Problem with a credit reporting company's investigation into an existing problem    Their investigation did not fix an error on your report3
4517
Incorrect information on credit report    Information is not mine    32384
Incorrect information on your report    Account status incorrect    26043
Deposits and withdrawals    22851
Time taken: 19.663 seconds, Fetched: 10 row(s)
```