ANALYZING US BIRTH DATA FROM 2016 TO 2021

Eyram Kueviakoe

Professor Nasheb Ismaily

May 30, 2024

Bellevue University
DSC 650 – Big Data

Outline

Introduction

Birth rates are a key indicator of population growth and can impact various sectors like healthcare and education. Understanding birth trends is important to improve the educational system and healthcare. It is also useful in making policies.

This project analyzes birth data from 2016 to 2021in the US to discover trends and patterns. We will be focusing on how the educational level of parents affects birth rates.

During this project, we will use big data tools to handle and analyze the dataset efficiently. We will use HDFS to store the data, spark to process the data, and Hive to query the data.

Data source

The dataset used for this project was obtained from Kaggle. It contains detailed information about the birth records from 2016 to 2021. It includes information on the number of births grouped by year, state and the educational level of the parents. The goal of the project is to use big data tools to analyze and query the data.

The link to the dataset is: https://www.kaggle.com/datasets/danbraswell/temporary-us-births?select=us_births_2016_2021.csv

Presentation of the dataset

Columns description:

State: Full name of the state

State Abbreviation: 2-character abbreviation of the state.

Year: The 4-digit year in which the births were recorded.

Gender: The gender of the baby (Male/Female).

Education Level of Mother: The education level of the mother.

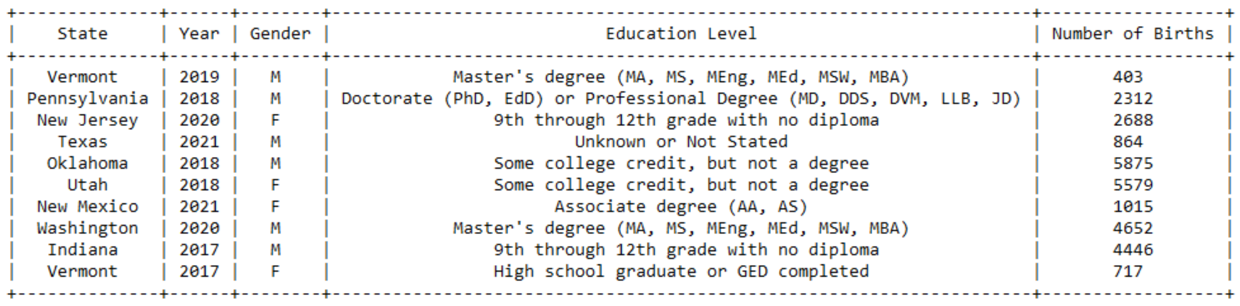
Education Level Code: The corresponding code for the education level of the mother

Number of Births: The number of births for the specified category.

Average Age of Mother (years): The average age of the mother in the specified category.

Average Birth Weight (g): The average birth weight in grams for the specified category.

Showing a sample of the dataset (with relevant columns)



```
+--------------+------+--------+--------------------------------------------------------------+-----------------+
|    State     | Year | Gender |                       Education Level                        | Number of Births |
+--------------+------+--------+--------------------------------------------------------------+-----------------+
|   Vermont    | 2019 |   M    |          Master's degree (MA, MS, MEng, MEd, MSW, MBA)          |       403       |
| Pennsylvania | 2018 |   M    | Doctorate (PhD, EdD) or Professional Degree (MD, DDS, DVM, LLB, JD) |      2312       |
|  New Jersey  | 2020 |   F    |             9th through 12th grade with no diploma              |      2688       |
|    Texas     | 2021 |   M    |                     Unknown or Not Stated                       |       864       |
|   Oklahoma   | 2018 |   M    |             Some college credit, but not a degree               |      5875       |
|     Utah     | 2018 |   F    |             Some college credit, but not a degree               |      5579       |
|  New Mexico  | 2021 |   F    |                   Associate degree (AA, AS)                     |      1015       |
|  Washington  | 2020 |   M    |          Master's degree (MA, MS, MEng, MEd, MSW, MBA)          |      4652       |
|   Indiana    | 2017 |   M    |             9th through 12th grade with no diploma              |      4446       |
|   Vermont    | 2017 |   F    |               High school graduate or GED completed             |       717       |
+--------------+------+--------+--------------------------------------------------------------+-----------------+
```

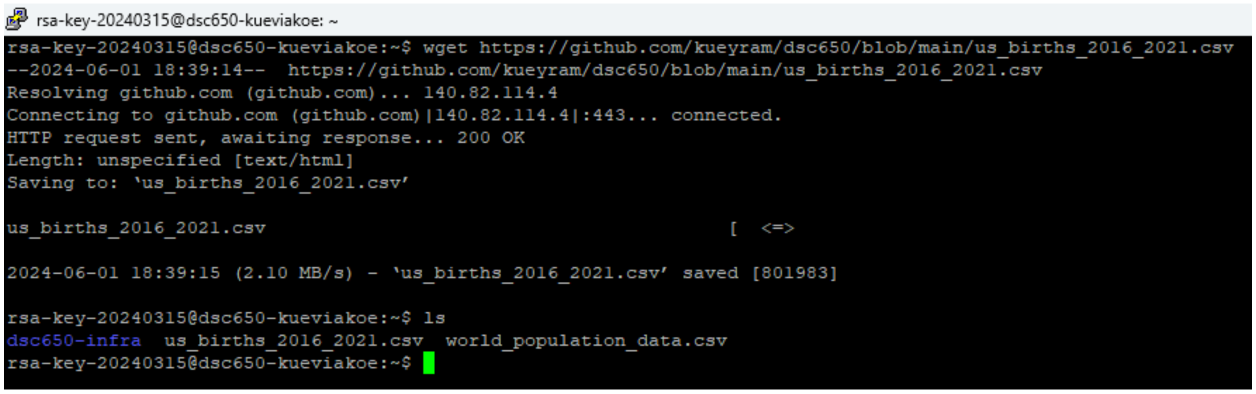
Data Ingestion

The first step of this project is to import the dataset into HDFS (Hadoop Distributed File System). To achieve this, we will connect to the HDFS instance in our google cloud virtual machine and upload the csv file containing the data.

<u>Step 1</u>: We download the csv file from Kaggle and upload it to our github repository.

<u>Step 2</u>: Uploading the csv file into our virtual machine using the command

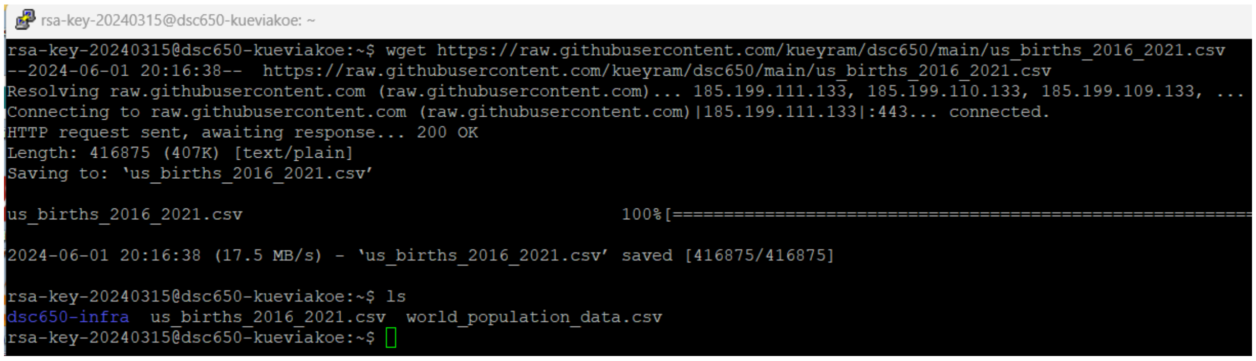
wget https://raw.githubusercontent.com/kueyram/dsc650/main/us_births_2016_2021.csv



```
rsa-key-20240315@dsc650-kueviakoe: ~
rsa-key-20240315@dsc650-kueviakoe:~$ wget https://github.com/kueyram/dsc650/blob/main/us_births_2016_2021.csv
--2024-06-01 18:39:14--  https://github.com/kueyram/dsc650/blob/main/us_births_2016_2021.csv
Resolving github.com (github.com)... 140.82.114.4
Connecting to github.com (github.com)|140.82.114.4|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: ‘us_births_2016_2021.csv’

us_births_2016_2021.csv                                 [  <=>

2024-06-01 18:39:15 (2.10 MB/s) - ‘us_births_2016_2021.csv’ saved [801983]

rsa-key-20240315@dsc650-kueviakoe:~$ ls
dsc650-infra  us_births_2016_2021.csv  world_population_data.csv
rsa-key-20240315@dsc650-kueviakoe:~$
```

Fig1: Downloading the csv file onto the virtual machine

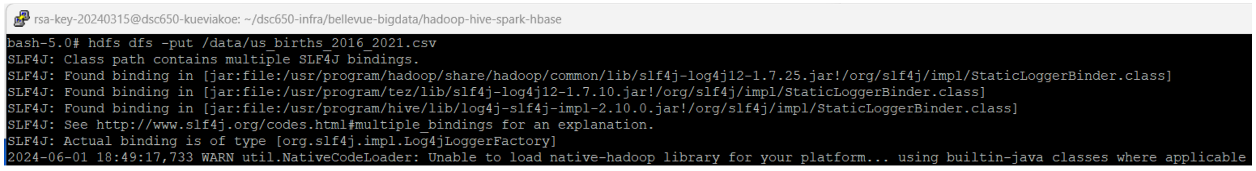
<u>Step 3</u>: Starting the Docker container and then accessing the master container



```
rsa-key-20240315@dsc650-kueviakoe: ~
rsa-key-20240315@dsc650-kueviakoe:~$ wget https://raw.githubusercontent.com/kueyram/dsc650/main/us_births_2016_2021.csv
--2024-06-01 20:16:38--  https://raw.githubusercontent.com/kueyram/dsc650/main/us_births_2016_2021.csv
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.111.133, 185.199.110.133, 185.199.109.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.111.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 416875 (407K) [text/plain]
Saving to: ‘us_births_2016_2021.csv’

us_births_2016_2021.csv                          100%[========================================================

2024-06-01 20:16:38 (17.5 MB/s) - ‘us_births_2016_2021.csv’ saved [416875/416875]

rsa-key-20240315@dsc650-kueviakoe:~$ ls
dsc650-infra  us_births_2016_2021.csv  world_population_data.csv
rsa-key-20240315@dsc650-kueviakoe:~$
```

Fig2: Starting docker container

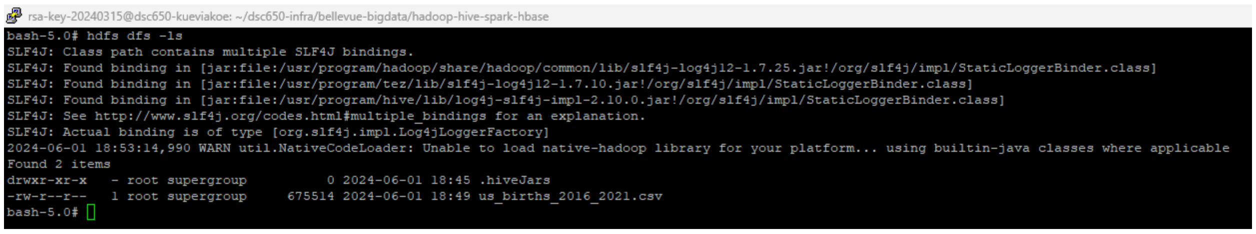
Step 4: Load the csv file into HDFS



```
rsa-key-20240315@dsc650-kueviakoe: ~/dsc650-infra/bellevue-bigdata/hadoop-hive-spark-hbase
bash-5.0# hdfs dfs -put /data/us_births_2016_2021.csv
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/program/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/program/tez/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/program/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
2024-06-01 18:49:17,733 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
```

Fig3: Loading the csv file into hdfs

Step 5: Checking if the file was successfully uploaded.



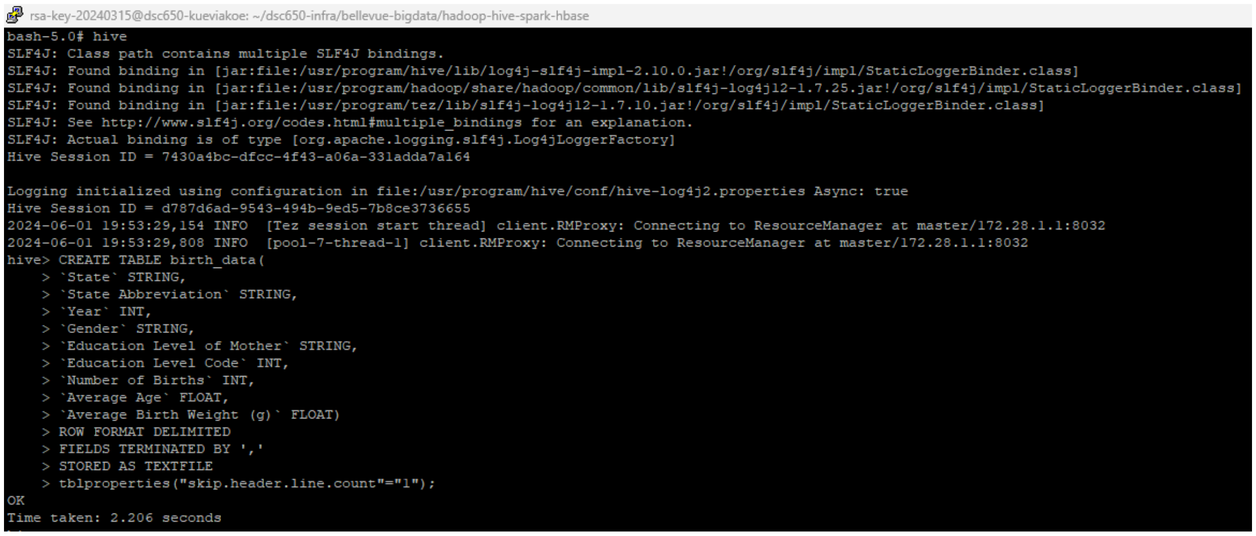
```
rsa-key-20240315@dsc650-kueviakoe: ~/dsc650-infra/bellevue-bigdata/hadoop-hive-spark-hbase
bash-5.0# hdfs dfs -ls
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/program/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/program/tez/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/program/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
2024-06-01 18:53:14,990 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 2 items
drwxr-xr-x   - root supergroup          0 2024-06-01 18:45 .hiveJars
-rw-r--r--   1 root supergroup     675514 2024-06-01 18:49 us_births_2016_2021.csv
bash-5.0# 
```

Fig4: Showing the csv was uploaded into hdfs

Create table and load data in Hive

Step6: Let's start a hive session and create a table using this command

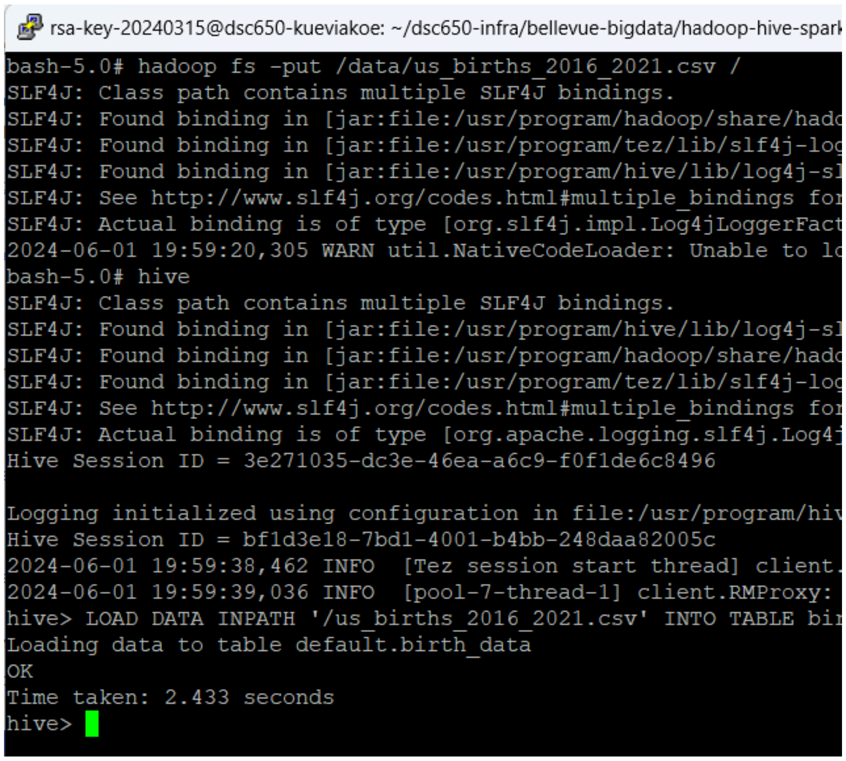
```
CREATE TABLE birth_data(
`State` STRING,
`State Abbreviation` STRING,
`Year` INT,
`Gender` STRING,
`Education Level of Mother` STRING,
`Education Level Code` INT,
`Number of Births` INT,
`Average Age` FLOAT,
`Average Birth Weight (g)` FLOAT)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
STORED AS TEXTFILE
tblproperties("skip.header.line.count"="1");
```



```
rsa-key-20240315@dsc650-kueviakoe: ~/dsc650-infra/bellevue-bigdata/hadoop-hive-spark-hbase
bash-5.0# hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/program/hive/lib/log4j-slf4j-impl-2.10.0.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/program/hadoop/share/hadoop/common/lib/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/program/tez/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
Hive Session ID = 7430a4bc-dfcc-4f43-a06a-331adda7a164

Logging initialized using configuration in file:/usr/program/hive/conf/hive-log4j2.properties Async: true
Hive Session ID = d787d6ad-9543-494b-9ed5-7b8ce3736655
2024-06-01 19:53:29,154 INFO  [Tez session start thread] client.RMProxy: Connecting to ResourceManager at master/172.28.1.1:8032
2024-06-01 19:53:29,808 INFO  [pool-7-thread-1] client.RMProxy: Connecting to ResourceManager at master/172.28.1.1:8032
hive> CREATE TABLE birth_data(
    > `State` STRING,
    > `State Abbreviation` STRING,
    > `Year` INT,
    > `Gender` STRING,
    > `Education Level of Mother` STRING,
    > `Education Level Code` INT,
    > `Number of Births` INT,
    > `Average Age` FLOAT,
    > `Average Birth Weight (g)` FLOAT)
    > ROW FORMAT DELIMITED
    > FIELDS TERMINATED BY ','
    > STORED AS TEXTFILE
    > tblproperties("skip.header.line.count"="1");
OK
Time taken: 2.206 seconds
```

Fig5: Accessing Hive and creating a table to store the data

Step 7: Loading the data into the Hive table:



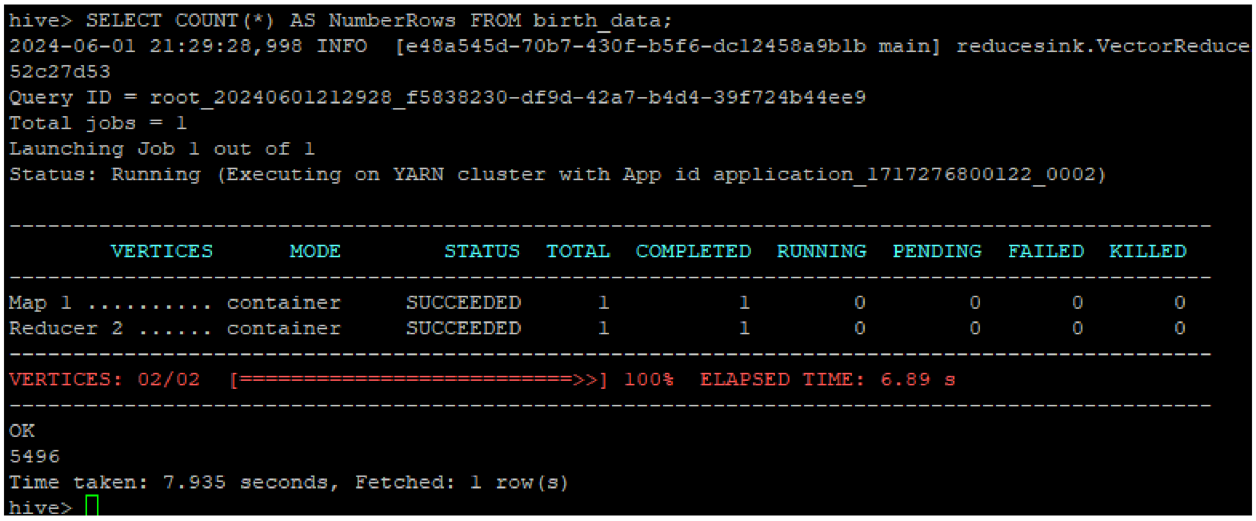
```
rsa-key-20240315@dsc650-kueviakoe: ~/dsc650-infra/bellevue-bigdata/hadoop-hive-spark
bash-5.0# hadoop fs -put /data/us_births_2016_2021.csv /
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/program/hadoop/share/hadc
SLF4J: Found binding in [jar:file:/usr/program/tez/lib/slf4j-log
SLF4J: Found binding in [jar:file:/usr/program/hive/lib/log4j-sl
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFact
2024-06-01 19:59:20,305 WARN util.NativeCodeLoader: Unable to lc
bash-5.0# hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/program/hive/lib/log4j-sl
SLF4J: Found binding in [jar:file:/usr/program/hadoop/share/hadc
SLF4J: Found binding in [jar:file:/usr/program/tez/lib/slf4j-log
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4j
Hive Session ID = 3e271035-dc3e-46ea-a6c9-f0f1de6c8496

Logging initialized using configuration in file:/usr/program/hiv
Hive Session ID = bf1d3e18-7bd1-4001-b4bb-248daa82005c
2024-06-01 19:59:38,462 INFO  [Tez session start thread] client.
2024-06-01 19:59:39,036 INFO  [pool-7-thread-1] client.RMProxy:
hive> LOAD DATA INPATH '/us_births_2016_2021.csv' INTO TABLE bir
Loading data to table default.birth_data
OK
Time taken: 2.433 seconds
hive>
```

Fig 6: Loading the data into the Hive table

Step 8: Running queries on the data

- Query 1: Let's count the number of rows in the dataset

SELECT COUNT(*) AS NumberRows FROM birth_data;



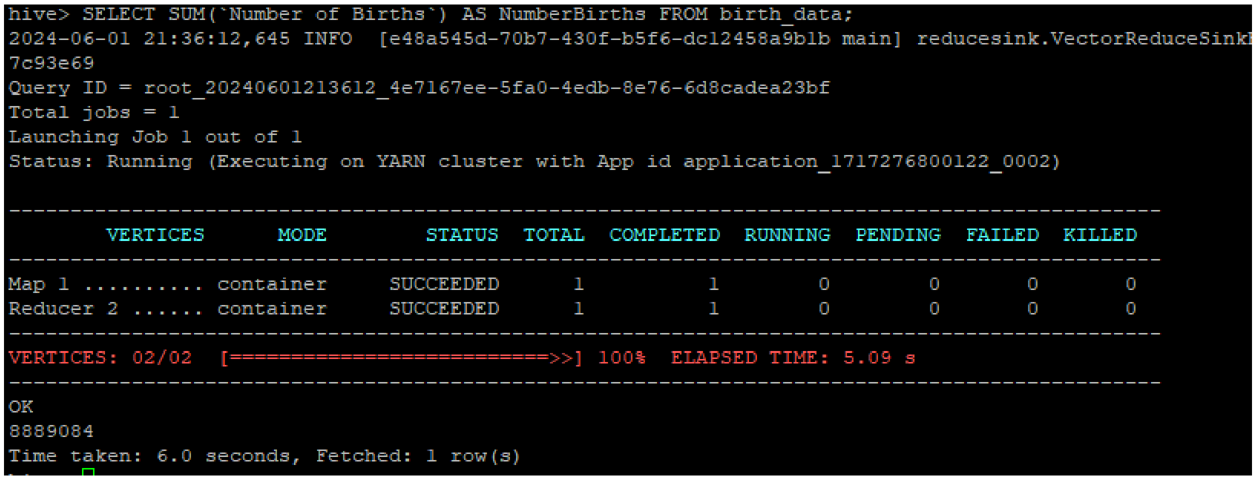
```
hive> SELECT COUNT(*) AS NumberRows FROM birth_data;
2024-06-01 21:29:28,998 INFO  [e48a545d-70b7-430f-b5f6-dc12458a9b1b main] reducesink.VectorReduce
52c27d53
Query ID = root_20240601212928_f5838230-df9d-42a7-b4d4-39f724b44ee9
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1717276800122_0002)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 6.89 s
----------------------------------------------------------------------------------------------
OK
5496
Time taken: 7.935 seconds, Fetched: 1 row(s)
hive>
```

Fig 7: Number of rows in the dataset

We have 5496 rows in the dataset

- Query 2: Let's count the number of births in the dataset

SELECT SUM(`Number of Births`) AS NumberBirths FROM birth_data;



```
hive> SELECT SUM(`Number of Births`) AS NumberBirths FROM birth_data;
2024-06-01 21:36:12,645 INFO  [e48a545d-70b7-430f-b5f6-dc12458a9b1b main] reducesink.VectorReduceSink
7c93e69
Query ID = root_20240601213612_4e7167ee-5fa0-4edb-8e76-6d8cadea23bf
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1717276800122_0002)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 5.09 s
----------------------------------------------------------------------------------------------
OK
8889084
Time taken: 6.0 seconds, Fetched: 1 row(s)
```

Fig8: Number of births in the dataset

There are 8889084 births in the dataset

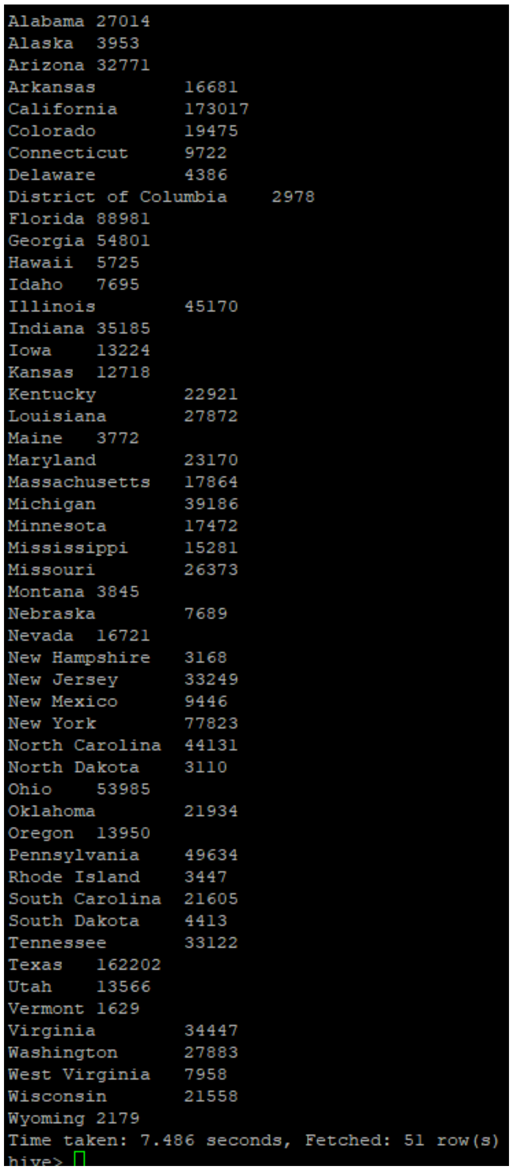
- Query 3: Number of births in each state in 2020

SELECT State, SUM(`Number of Births`) AS total_births_2020

FROM birth_data

WHERE Year = 2020

GROUP BY State;



```
Alabama 27014
Alaska  3953
Arizona 32771
Arkansas        16681
California      173017
Colorado        19475
Connecticut     9722
Delaware        4386
District of Columbia    2978
Florida 88981
Georgia 54801
Hawaii  5725
Idaho   7695
Illinois        45170
Indiana 35185
Iowa    13224
Kansas  12718
Kentucky        22921
Louisiana       27872
Maine   3772
Maryland        23170
Massachusetts   17864
Michigan        39186
Minnesota       17472
Mississippi     15281
Missouri        26373
Montana 3845
Nebraska        7689
Nevada  16721
New Hampshire   3168
New Jersey      33249
New Mexico      9446
New York        77823
North Carolina  44131
North Dakota    3110
Ohio    53985
Oklahoma        21934
Oregon  13950
Pennsylvania    49634
Rhode Island    3447
South Carolina  21605
South Dakota    4413
Tennessee       33122
Texas   162202
Utah    13566
Vermont 1629
Virginia        34447
Washington      27883
West Virginia   7958
Wisconsin       21558
Wyoming 2179
Time taken: 7.486 seconds, Fetched: 51 row(s)
hive>
```

Fig8: Number of births in each state in 2020

Query 4: Top 5 education level with the highest births

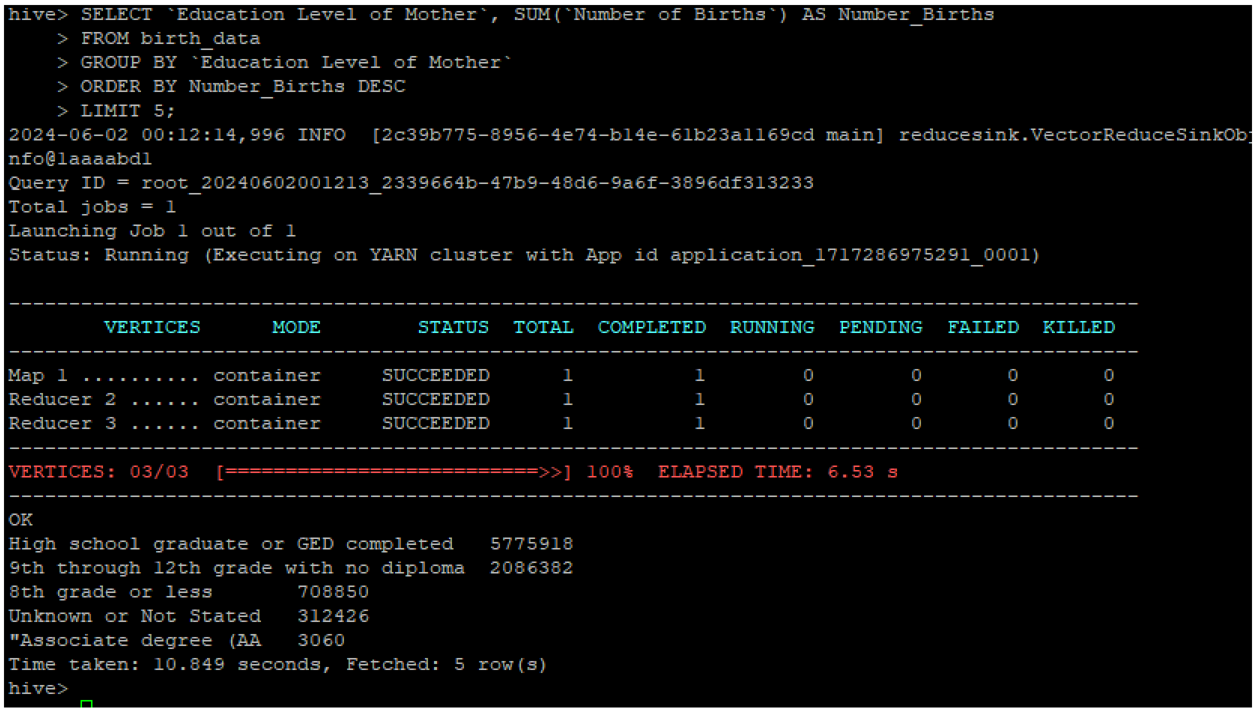
SELECT `Education Level of Mother`, SUM(`Number of Births`) AS Number_Births

FROM birth_data

GROUP BY `Education Level of Mother`

ORDER BY Number_Births DESC

LIMIT 5;



```
hive> SELECT `Education Level of Mother`, SUM(`Number of Births`) AS Number_Births
    > FROM birth_data
    > GROUP BY `Education Level of Mother`
    > ORDER BY Number_Births DESC
    > LIMIT 5;
2024-06-02 00:12:14,996 INFO  [2c39b775-8956-4e74-b14e-61b23a1169cd main] reducesink.VectorReduceSinkOb
nfo@1aaaabd1
Query ID = root_20240602001213_2339664b-47b9-48d6-9a6f-3896df313233
Total jobs = 1
Launching Job 1 out of 1
Status: Running (Executing on YARN cluster with App id application_1717286975291_0001)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        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: 6.53 s
----------------------------------------------------------------------------------------------
OK
High school graduate or GED completed   5775918
9th through 12th grade with no diploma  2086382
8th grade or less       708850
Unknown or Not Stated   312426
"Associate degree (AA   3060
Time taken: 10.849 seconds, Fetched: 5 row(s)
hive>
```

Fig 9: Top 5 education level with the highest births

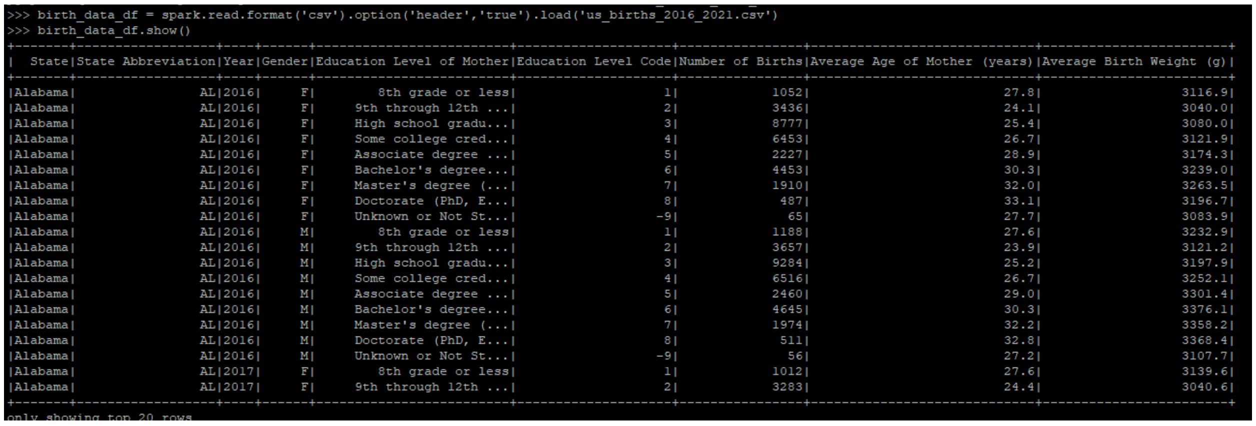
Using Spark

Step 8: Loading data into spark and creating a dataframe

birth_data_df = spark.read.format('csv').option('header','true').load('us_births_2016_2021.csv')

Let's very the dataframe was created.

Birth_date_df.show()



```
>>> birth_data_df = spark.read.format('csv').option('header','true').load('us_births_2016_2021.csv')
>>> birth_data_df.show()
+-------+------------------+----+------+-------------------------+--------------------+----------------+-----------------------------+------------------------+
|  State|State Abbreviation|Year|Gender|Education Level of Mother|Education Level Code|Number of Births|Average Age of Mother (years)|Average Birth Weight (g)|
+-------+------------------+----+------+-------------------------+--------------------+----------------+-----------------------------+------------------------+
|Alabama|                AL|2016|     F|        8th grade or less|                   1|            1052|                         27.8|                  3116.9|
|Alabama|                AL|2016|     F|     9th through 12th ...|                   2|            3436|                         24.1|                  3040.0|
|Alabama|                AL|2016|     F|     High school gradu...|                   3|            8777|                         25.4|                  3080.0|
|Alabama|                AL|2016|     F|     Some college cred...|                   4|            6453|                         26.7|                  3121.9|
|Alabama|                AL|2016|     F|     Associate degree ...|                   5|            2227|                         28.9|                  3174.3|
|Alabama|                AL|2016|     F|     Bachelor's degree...|                   6|            4453|                         30.3|                  3239.0|
|Alabama|                AL|2016|     F|     Master's degree (...|                   7|            1910|                         32.0|                  3263.5|
|Alabama|                AL|2016|     F|     Doctorate (PhD, E...|                   8|             487|                         33.1|                  3196.7|
|Alabama|                AL|2016|     F|     Unknown or Not St...|                  -9|              65|                         27.7|                  3083.9|
|Alabama|                AL|2016|     M|        8th grade or less|                   1|            1188|                         27.6|                  3232.9|
|Alabama|                AL|2016|     M|     9th through 12th ...|                   2|            3657|                         23.9|                  3121.2|
|Alabama|                AL|2016|     M|     High school gradu...|                   3|            9284|                         25.2|                  3197.9|
|Alabama|                AL|2016|     M|     Some college cred...|                   4|            6516|                         26.7|                  3252.1|
|Alabama|                AL|2016|     M|     Associate degree ...|                   5|            2460|                         29.0|                  3301.4|
|Alabama|                AL|2016|     M|     Bachelor's degree...|                   6|            4645|                         30.3|                  3376.1|
|Alabama|                AL|2016|     M|     Master's degree (...|                   7|            1974|                         32.2|                  3358.2|
|Alabama|                AL|2016|     M|     Doctorate (PhD, E...|                   8|             511|                         32.8|                  3368.4|
|Alabama|                AL|2016|     M|     Unknown or Not St...|                  -9|              56|                         27.2|                  3107.7|
|Alabama|                AL|2017|     F|        8th grade or less|                   1|            1012|                         27.6|                  3139.6|
|Alabama|                AL|2017|     F|     9th through 12th ...|                   2|            3283|                         24.4|                  3040.6|
+-------+------------------+----+------+-------------------------+--------------------+----------------+-----------------------------+------------------------+
only showing top 20 rows
```

Fig 9: Checking that the dataframe was created and populated

<u>Step 9</u>: Let's remove the redundant columns from the dataframe

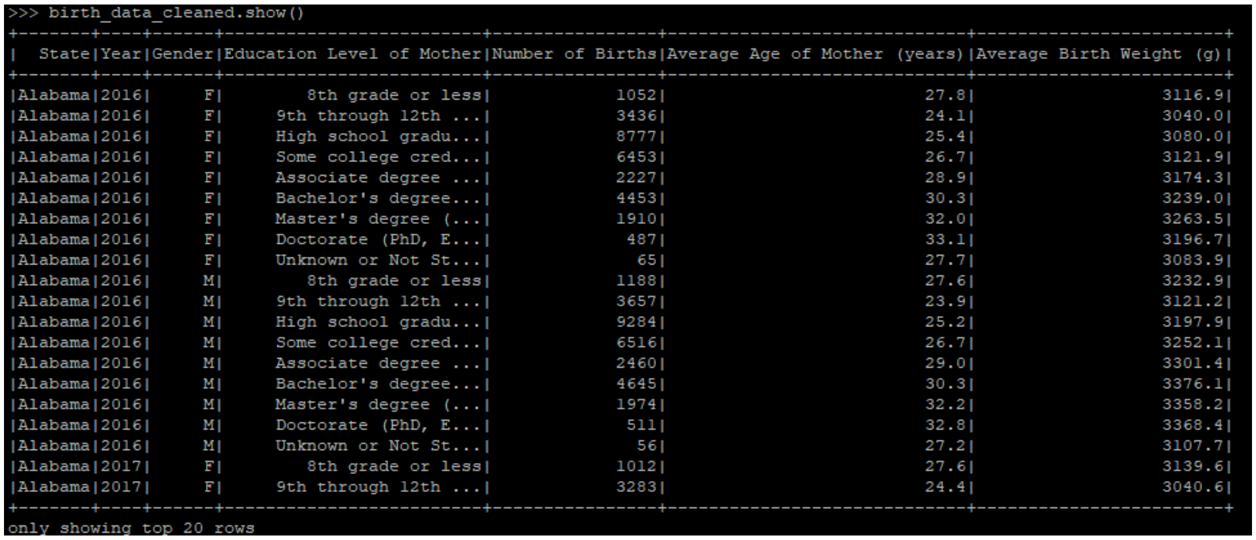
The dataframe has a column called State and another column which has the state names abbreviated. It also has the educational level and the code that corresponds to to the educational level.

We will drop State Abbreviation, and Education Level Code

birth_data_cleaned = birth_data_df.drop("State Abbreviation", "Education Level Code")

Checking the new dataframe

Birth_data_cleaned.show()



```
>>> birth_data_cleaned.show()
+-------+----+------+-------------------------+----------------+-----------------------------+------------------------+
|  State|Year|Gender|Education Level of Mother|Number of Births|Average Age of Mother (years)|Average Birth Weight (g)|
+-------+----+------+-------------------------+----------------+-----------------------------+------------------------+
|Alabama|2016|     F|        8th grade or less|            1052|                         27.8|                  3116.9|
|Alabama|2016|     F|     9th through 12th ...|            3436|                         24.1|                  3040.0|
|Alabama|2016|     F|     High school gradu...|            8777|                         25.4|                  3080.0|
|Alabama|2016|     F|     Some college cred...|            6453|                         26.7|                  3121.9|
|Alabama|2016|     F|     Associate degree ...|            2227|                         28.9|                  3174.3|
|Alabama|2016|     F|     Bachelor's degree...|            4453|                         30.3|                  3239.0|
|Alabama|2016|     F|     Master's degree (...|            1910|                         32.0|                  3263.5|
|Alabama|2016|     F|     Doctorate (PhD, E...|             487|                         33.1|                  3196.7|
|Alabama|2016|     F|     Unknown or Not St...|              65|                         27.7|                  3083.9|
|Alabama|2016|     M|        8th grade or less|            1188|                         27.6|                  3232.9|
|Alabama|2016|     M|     9th through 12th ...|            3657|                         23.9|                  3121.2|
|Alabama|2016|     M|     High school gradu...|            9284|                         25.2|                  3197.9|
|Alabama|2016|     M|     Some college cred...|            6516|                         26.7|                  3252.1|
|Alabama|2016|     M|     Associate degree ...|            2460|                         29.0|                  3301.4|
|Alabama|2016|     M|     Bachelor's degree...|            4645|                         30.3|                  3376.1|
|Alabama|2016|     M|     Master's degree (...|            1974|                         32.2|                  3358.2|
|Alabama|2016|     M|     Doctorate (PhD, E...|             511|                         32.8|                  3368.4|
|Alabama|2016|     M|     Unknown or Not St...|              56|                         27.2|                  3107.7|
|Alabama|2017|     F|        8th grade or less|            1012|                         27.6|                  3139.6|
|Alabama|2017|     F|     9th through 12th ...|            3283|                         24.4|                  3040.6|
+-------+----+------+-------------------------+----------------+-----------------------------+------------------------+
only showing top 20 rows
```

Fig 10: New dataframe after redundant columns were removed

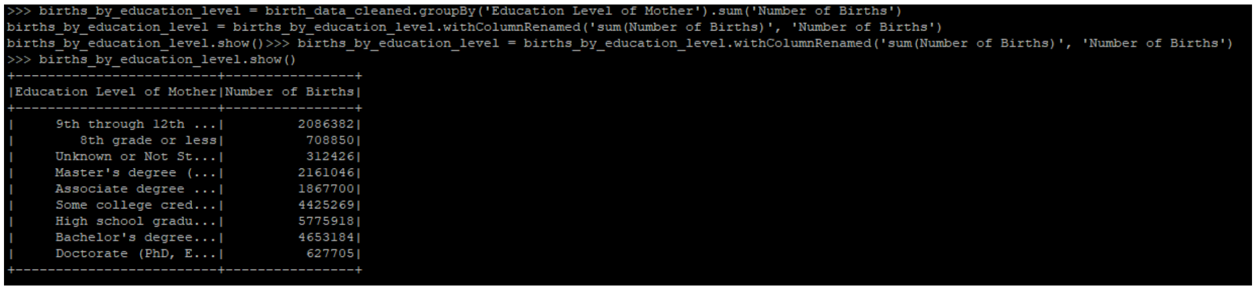
With the cleaned dataframe, we can create visualizations and find possible correlations between the number of births and the education level of the mother.

<u>Step 10</u>: Number of births grouped by education level

births_by_education_level = birth_data_cleaned.groupBy('Education Level of Mother').sum('Number of Births')

births_by_education_level = births_by_education_level.withColumnRenamed('sum(Number of Births)', 'Number of Births')

births_by_education_level.show()



```
>>> births_by_education_level = birth_data_cleaned.groupBy('Education Level of Mother').sum('Number of Births')
births_by_education_level = births_by_education_level.withColumnRenamed('sum(Number of Births)', 'Number of Births')
births_by_education_level.show()>>> births_by_education_level = births_by_education_level.withColumnRenamed('sum(Number of Births)', 'Number of Births')
>>> births_by_education_level.show()
+-------------------------+----------------+
|Education Level of Mother|Number of Births|
+-------------------------+----------------+
|     9th through 12th ...|         2086382|
|        8th grade or less|          708850|
|     Unknown or Not St...|          312426|
|     Master's degree (...|         2161046|
|     Associate degree ...|         1867700|
|     Some college cred...|         4425269|
|     High school gradu...|         5775918|
|     Bachelor's degree...|         4653184|
|     Doctorate (PhD, E...|          627705|
+-------------------------+----------------+
```

Fig 11: Births grouped by educational level

Conclusion

After ingesting the data into HDFS, we were able to define our table schema in Hive and import the data into the table. This gave us the opportunity to use HiveQL to run different queries on the table.

For advanced data manipulation and transformation, we use PySpark to create a dataframe that can be used to create visualization and graphs. PySpark can be used for data analysis, complex transformation, and machine learning.

The analysis of the data shows that as the mother's education level rises, there is a decrease in the number of births. This means that there is a possible correlation between higher education among mothers and reduced fertility rates. This can be used by healthcare professionals and educators to promote family planning.

In the future, we could include real-time data processing to ensure the analysis is up-to-date and responsive to changing trends. We could also use machine learning to make predictions. Finally, we could implement advanced visualization and interactive dashboards to present and communicate our findings.