CA4022 - Starbucks Dataset Analysis using Apache Pig and Hive

Robert Sparks - 20480484

November 2023

Github repository - https://github.com/rs337/Apache-Pig-and-Hive-Analysis

1. Introduction

We have been tasked in this assignment to use a dataset to perform some cleaning and analysis on using Apache Pig and Hive. The dataset I chose was the Starbucks dataset, it is a collection of nutritional information for the Starbucks food and drinks items. There were three csv files in the dataset with various cleaning tasks needed to be performed to have clean data to perform the querying and analysis on.

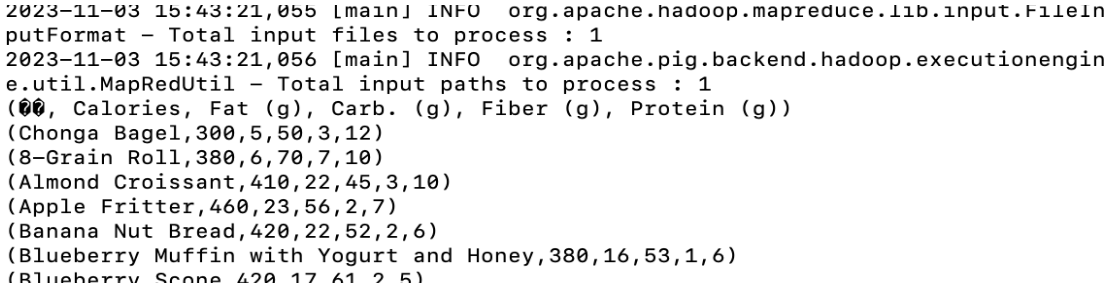
We had to perform our data cleaning in Pig and do 2 simple queries there and then use Hive to perform the same 2 queries and then 3 extra complex queries. I do not include my cleaning commands or queries in this write up however they can be found in the cleaning.pig, simple_queries.pig and hive_queries.hive respectively.

2. Cleaning the data - Pig

In order to clean our data we must use Apache Pig. I used Apache pig locally. It is essential to clean your data before using it as inconsistent data can lead to unreliable results. It can also stop you from doing any queries at all. To understand what cleaning I needed to do I loaded the files one by one and used Pig's dump function to see what the output looked like. From this I had a fairly good idea what needed to be cleaned. Below I list some of the main processes that I carried out, however you can see the full process carried out in my cleaning.pig file.

- LOAD (path/to/csv) USING PigStorage(',') AS (columnName:valueType) this command was used for each of the 3 csv files to get them into Pig.
- STORE data INTO 'starbucks_clean/folder_name' USING org.apache.pig.piggybank.storage.CSVExcelStorage(); was used to export each of the 3 files once they were clean.
- FILTER data BY Name != ''; was used to skip the headers line for each of the files.

One issue I had when loading in the food menu in was there was some sort of encoding on the file that meant when I dumped the output there was a symbol in the first row:



```
2023-11-03 15:43:21,055 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileIn
putFormat - Total input files to process : 1
2023-11-03 15:43:21,056 [main] INFO  org.apache.pig.backend.hadoop.executionengin
e.util.MapRedUtil - Total input paths to process : 1
(��, Calories, Fat (g), Carb. (g), Fiber (g), Protein (g))
(Chonga Bagel,300,5,50,3,12)
(8-Grain Roll,380,6,70,7,10)
(Almond Croissant,410,22,45,3,10)
(Apple Fritter,460,23,56,2,7)
(Banana Nut Bread,420,22,52,2,6)
(Blueberry Muffin with Yogurt and Honey,380,16,53,1,6)
(Blueberry Scone,420,17,61,2,5)
```

Upon doing some research I found that there must be some sort of different encoding on the file. This led me to open the file in sublime text which opened fine, strangely, the same happened with Google Docs. I then found a feature on Sublime Text that allowed me to open it with encoding. From manually going through a few different encodings I found that it was UTF-16-LE. I then used the command below in my terminal to convert the file into UTF-8 which Pig loaded in perfectly.

- Iconv -f UTF-16LE -t UTF-8 'starbucks_raw/starbucks-menu-nutrition-food.csv' > 'starbucks_raw/starbucks-menu-nutrition-food-no_encoding.csv'

3. Querying the Data

I carried out two simple queries in Pig and then in Hive. The results will be shown below first with the Pig results and then the Hive results. In order to know that I did the querying and setup correctly the results from all queries should match. I then also show the results of my complex queries where I do one standard complex query, an outer join and one random sampling query. All of the code used to perform these queries will be in my Pig and Hive query files.

3.1 What food item has the highest protein?

As someone who is passionate about the gym and healthy eating I was curious to see what food item had the highest protein from the starbucks menu.

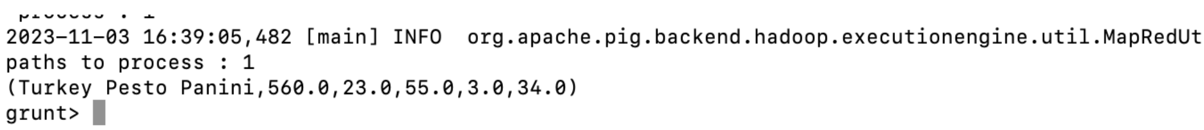
Pig

To find out the highest protein I ordered the table by Protein in a descending order and as I was only interested in the highest protein count I added a ‘LIMIT 1’ to the query.

highest_protein = ORDER food BY Protein DESC;

high_protein_food = LIMIT highest_protein 1;

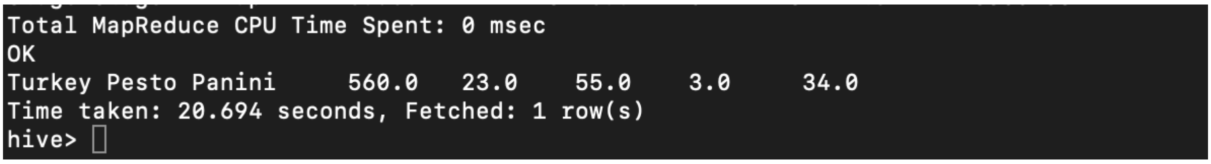
This resulted in the following output once I dumped it out of Pig.



```
process : 1
2023-11-03 16:39:05,482 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUt
paths to process : 1
(Turkey Pesto Panini,560.0,23.0,55.0,3.0,34.0)
grunt>
```

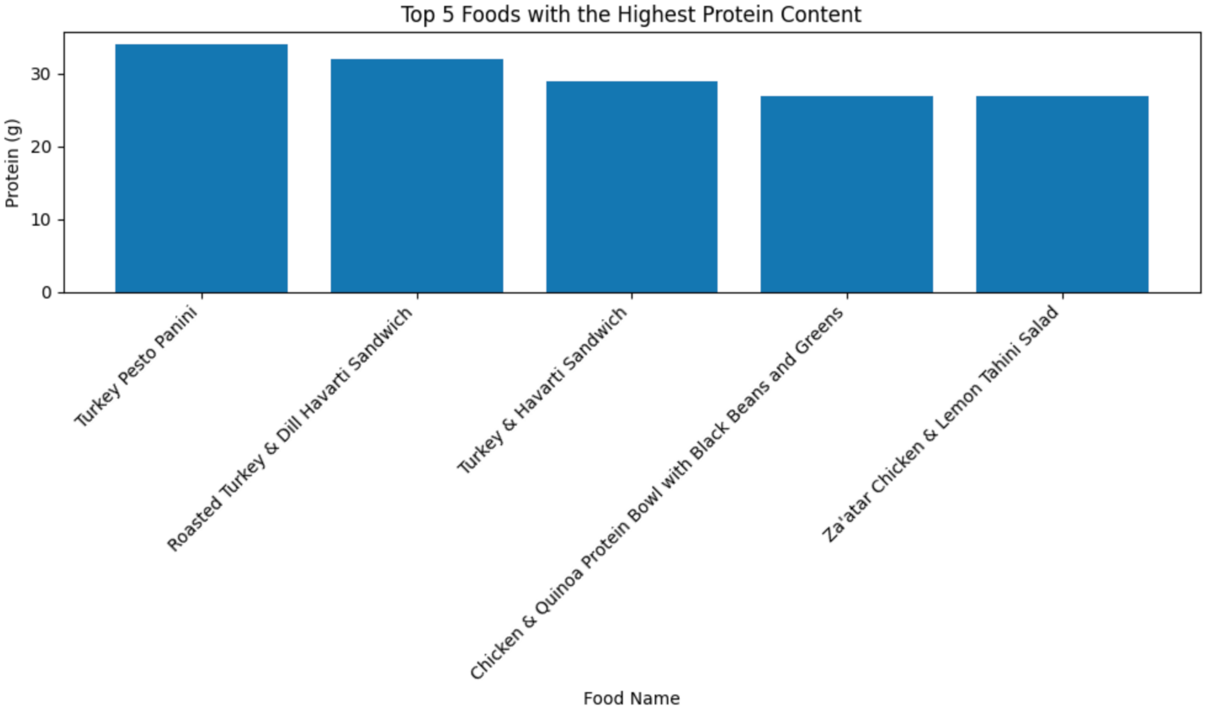
Hive

I did a similar query in Hive which resulted in this:



```
Total MapReduce CPU Time Spent: 0 msec
OK
Turkey Pesto Panini      560.0    23.0     55.0    3.0      34.0
Time taken: 20.694 seconds, Fetched: 1 row(s)
hive> 
```

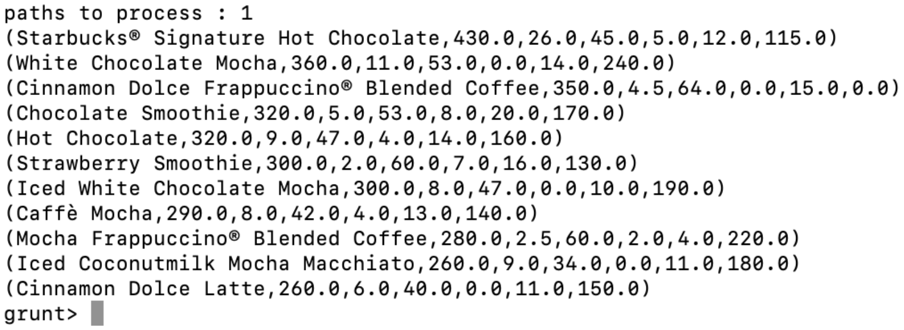
As can be seen the output was Turkey Pesto Panini for both queries which meant they ran successfully. The last column is the Protein which had a count of 34g which is pretty good. Here is a visualisation of the top five highest protein foods on the menu created in Google Colab using the matplot and pandas library. This also backs up our query from Pig and Hive.



3.2 What drinks have more than 250 calories?

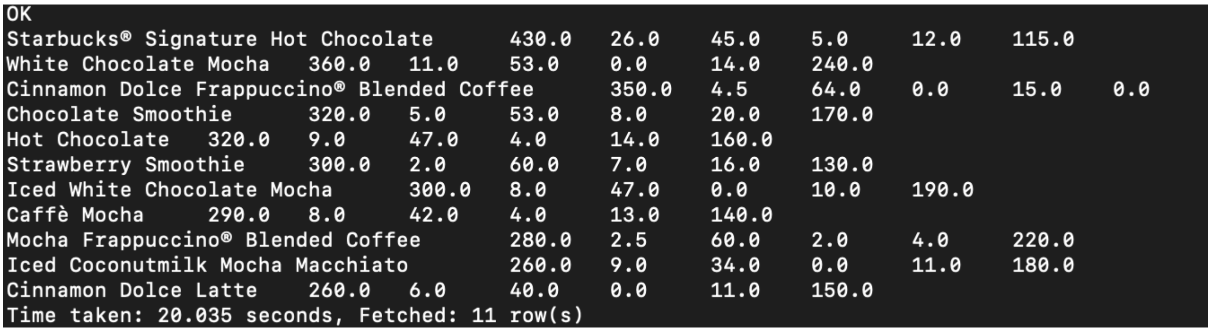
Next, I wanted to see which drinks were high in calories. To be more precise, I wanted to see what drinks/ how many drinks had more than 250 calories. To do this I selected all drinks greater than 250 calories and then put them in descending order so I could see what had the highest calories. The results from Pig and Hive can be seen below:

Pig



```
paths to process : 1
(Starbucks® Signature Hot Chocolate,430.0,26.0,45.0,5.0,12.0,115.0)
(White Chocolate Mocha,360.0,11.0,53.0,0.0,14.0,240.0)
(Cinnamon Dolce Frappuccino® Blended Coffee,350.0,4.5,64.0,0.0,15.0,0.0)
(Chocolate Smoothie,320.0,5.0,53.0,8.0,20.0,170.0)
(Hot Chocolate,320.0,9.0,47.0,4.0,14.0,160.0)
(Strawberry Smoothie,300.0,2.0,60.0,7.0,16.0,130.0)
(Iced White Chocolate Mocha,300.0,8.0,47.0,0.0,10.0,190.0)
(Caffè Mocha,290.0,8.0,42.0,4.0,13.0,140.0)
(Mocha Frappuccino® Blended Coffee,280.0,2.5,60.0,2.0,4.0,220.0)
(Iced Coconutmilk Mocha Macchiato,260.0,9.0,34.0,0.0,11.0,180.0)
(Cinnamon Dolce Latte,260.0,6.0,40.0,0.0,11.0,150.0)
grunt> 
```

Hive

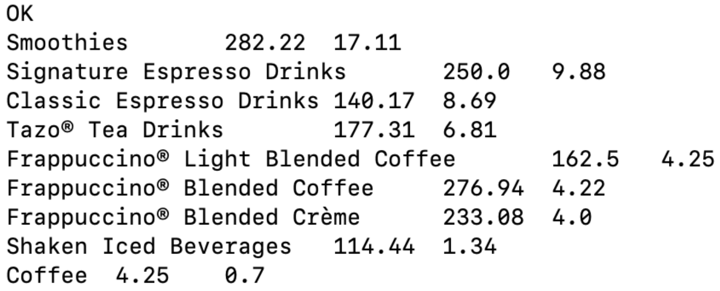


```
OK
Starbucks® Signature Hot Chocolate      430.0   26.0    45.0    5.0     12.0    115.0
White Chocolate Mocha   360.0   11.0    53.0    0.0     14.0    240.0
Cinnamon Dolce Frappuccino® Blended Coffee      350.0   4.5     64.0    0.0     15.0    0.0
Chocolate Smoothie      320.0   5.0     53.0    8.0     20.0    170.0
Hot Chocolate   320.0   9.0     47.0    4.0     14.0    160.0
Strawberry Smoothie     300.0   2.0     60.0    7.0     16.0    130.0
Iced White Chocolate Mocha      300.0   8.0     47.0    0.0     10.0    190.0
Caffè Mocha     290.0   8.0     42.0    4.0     13.0    140.0
Mocha Frappuccino® Blended Coffee       280.0   2.5     60.0    2.0     4.0     220.0
Iced Coconutmilk Mocha Macchiato        260.0   9.0     34.0    0.0     11.0    180.0
Cinnamon Dolce Latte    260.0   6.0     40.0    0.0     11.0    150.0
Time taken: 20.035 seconds, Fetched: 11 row(s)
```

Thankfully, we received the same output for both Hive and Pig. We can see that the Starbucks Signature Hot Chocolate had the highest caloric content. We can also notice that there are 11 drinks on the menu that are over 250 calories. As was pointed out from the dataset the nutritional information is given 12 oz drinks.

3.3 Beverage Category - Calories vs. Protein

For this query I wanted to see per Beverage Category what the average calories were in comparison to the average protein content. In order to do this I had to use the AVG() and the ROUND(,2) function to round the averages to 2 decimal places. I also ordered the output based on the protein content in descending order.

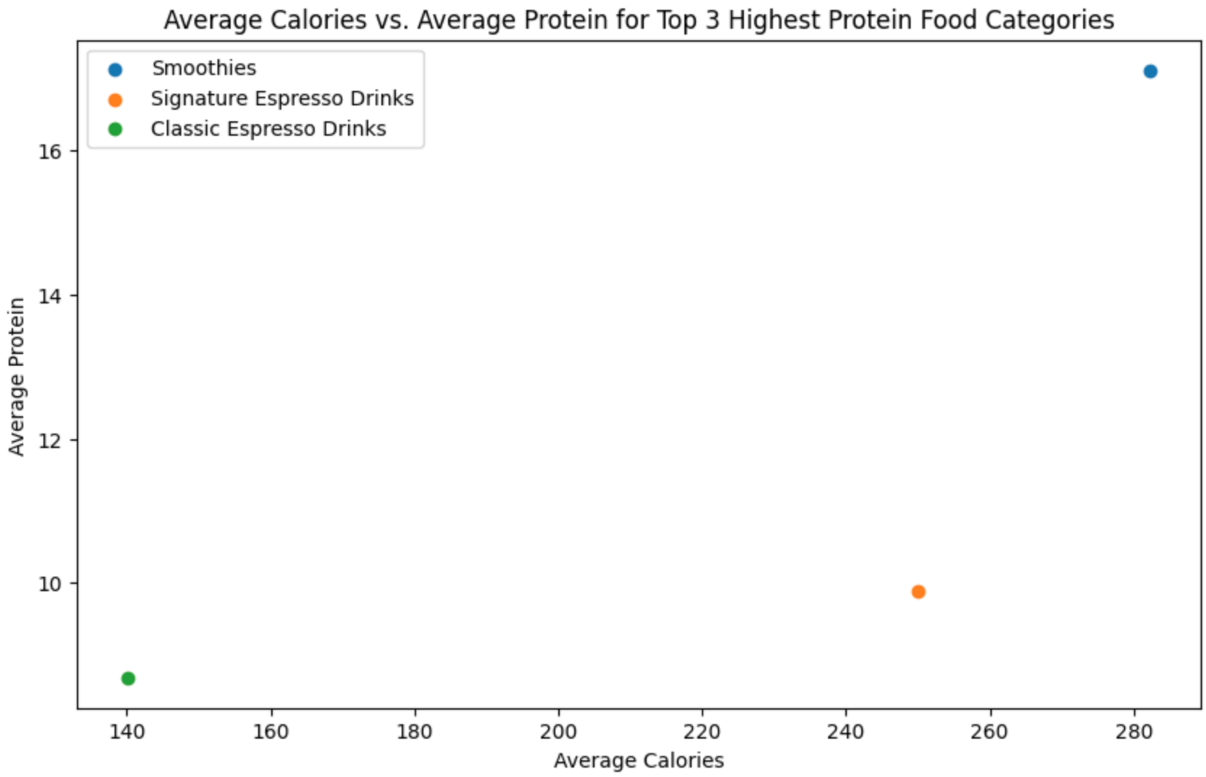


```
OK
Smoothies       282.22  17.11
Signature Espresso Drinks       250.0   9.88
Classic Espresso Drinks 140.17  8.69
Tazo® Tea Drinks        177.31  6.81
Frappuccino® Light Blended Coffee       162.5   4.25
Frappuccino® Blended Coffee     276.94  4.22
Frappuccino® Blended Crème      233.08  4.0
Shaken Iced Beverages   114.44  1.34
Coffee  4.25    0.7
```

The output from Hive looked like this which had the category on the left and then average calories and the average protein. Something I noticed was that Smoothies and Frappuccino Blended Coffee had similar average calorie contents, 282.22 and 276.94 respectively. However they had very different average protein contents with 17.11g for smoothies and only 4.22g for the Frappuccino Blended Coffee.

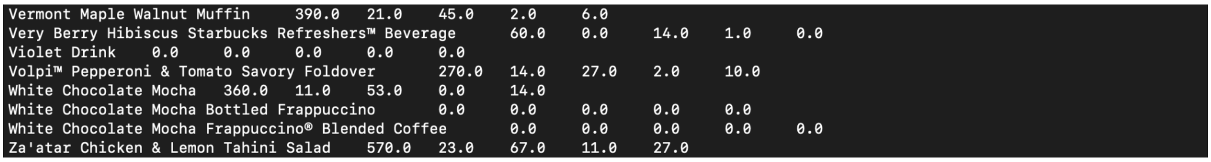
When we compare this with the highest protein food, the Turkey Pesto Panini with 34g of protein even the Smoothie does not compare too well. I would have liked to do a comparison between food and drinks categories. Unfortunately, there was no category for the food so I could not have performed a join on the two tables. Also, the expanded menu only had drinks in it.

Looking below, there is a visualisation I created of the top 3 highest average protein drink categories. It backs up the findings from the queries which I am happy with.



3.4 Join Query

I performed a full outer join on my food and drinks table. I made use of a new function to me called the COALESCE() function. Which when used in a join allows you to take only the values from either the food or drinks table. This allowed me to make a clean looking table with the Name, Calories, Fat, Carb, Fiber and Protein. The output can be seen below:

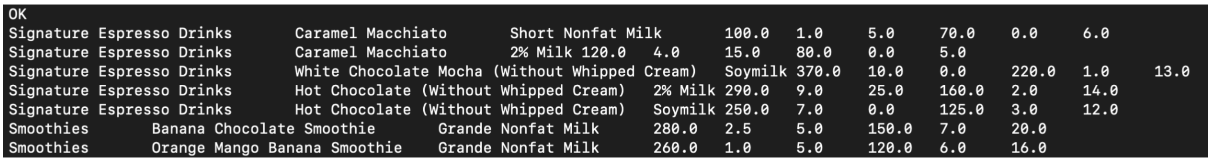


```
Vermont Maple Walnut Muffin     390.0   21.0    45.0    2.0     6.0
Very Berry Hibiscus Starbucks Refreshers™ Beverage      60.0    0.0     14.0    1.0     0.0
Violet Drink    0.0     0.0     0.0     0.0     0.0
Volpi™ Pepperoni & Tomato Savory Foldover       270.0   14.0    27.0    2.0     10.0
White Chocolate Mocha   360.0   11.0    53.0    0.0     14.0
White Chocolate Mocha Bottled Frappuccino       0.0     0.0     0.0     0.0     0.0
White Chocolate Mocha Frappuccino® Blended Coffee       0.0     0.0     0.0     0.0     0.0
Za'atar Chicken & Lemon Tahini Salad    570.0   23.0    67.0    11.0    27.0
```

From this we can see drinks and food items clearly displayed with only 6 columns in total. Without the use of COALESCE function I was ending up with a lot more and many of the columns had NULL values as there was no direct join available.

3.5 Sampling Query

I was unable to come up with some interesting analysis with a sampling query. However, I decided to do a UNION between the Signature Espresso Drinks and Smoothies beverage categories instead. When I ran the query it resulted in this:



```
OK
Signature Espresso Drinks       Caramel Macchiato       Short Nonfat Milk       100.0   1.0     5.0     70.0    0.0     6.0
Signature Espresso Drinks       Caramel Macchiato       2% Milk 120.0   4.0     15.0    80.0    0.0     5.0
Signature Espresso Drinks       White Chocolate Mocha (Without Whipped Cream)   Soymilk 370.0   10.0    0.0     220.0   1.0     13.0
Signature Espresso Drinks       Hot Chocolate (Without Whipped Cream)   2% Milk 290.0   9.0     25.0    160.0   2.0     14.0
Signature Espresso Drinks       Hot Chocolate (Without Whipped Cream)   Soymilk 250.0   7.0     0.0     125.0   3.0     12.0
Smoothies       Banana Chocolate Smoothie       Grande Nonfat Milk      280.0   2.5     5.0     150.0   7.0     20.0
Smoothies       Orange Mango Banana Smoothie    Grande Nonfat Milk      260.0   1.0     5.0     120.0   6.0     16.0
```

The results of this confused me a little as no Smoothies were returned when the sampling percentage was set to 10% so I upped it to 25% and saw 2 smoothies returned this time. I ran a COUNT query and saw that there were only 9 smoothies in the menu and 40 signature espresso drinks, which clarifies the results of this for me.

4. Conclusion

To conclude, I am more of a believer of the true power of Apache Pig and Hive. I was initially sceptical after having a lot of difficulty with the setup of Hadoop and struggled to see why it was so important. However, once it is all configured correctly it is very easy to use with some previous SQL experience. I think I am yet to see the true capabilities of the system as it is designed for dealing with much larger data. I now have a bit of confidence in my abilities to operate on the system and look forward to seeing how and where it will be used in future workplaces.

There are various points worth mentioning from my analysis, such as various nutritional information. However, it was only after getting to the complex queries that I felt I could have found a larger dataset to work with as I feel I was a bit limited with what I could do. I still believe I have shown the power of the system and some various insights.