Data Science Project Report

1. Principal Investigator

Carlo Rizzo Email: **rizzoc3@newpaltz.edu**

1.1 Individual Contribution Breakdown (list the percentage)

Task	Member 1	Total
Introduction	Carlo	100%
Background	Carlo	100%
Implementation	Carlo	100%
Experiment Results and Discussion	Carlo	100%
Conclusion	Carlo	100%
Other contribution and explain	Carlo	100%

2. Title of Project

Titanic - Machine Learning from Disaster using MapReduce

3. Mentoring

Professor Min Chen, Department of Computer Science, SUNY-New Paltz

chenm@newpaltz.edu

4. Introduction

4.1 Project Motivation:

The main motivation for the project was to apply my new-found knowledge in data science to an age old problem. The Titanic is a disaster that everyone knows about, and society has already learned so much from the tragedy in regards to public safety and the construction of boats. I thought it would be interesting to see if it could teach us another lesson about the types of people who survived the wreck, and we can determine this through data science. One of our main tools will the K-Means algorithm; an algorithm designed to calculate the centroids of clusters which can be analysed and then used to determine an outcome. The next tool will be Python to

clean the data and do some preliminary work and run our initial K-Means to receive the initial centroids from the cleaned data. Finally, we will use Hadoop to run the MapReduce function, giving it our initial centroids from the Python code, and manipulating them to receive new centroids that we can use to compare with Python's K-Means function to test accuracy.

4.2 Aims and Objectives

Goals:

- Clean the data
- Run K-Means on the data to receive centroids
- Use these centroids in our MapReduce function
- Analyse and plot the data
- Determine which types of people lived through the sinking of the Titanic
- Compare the outcome received through my analysis with the actual outcome of the tragedy
- Draw conclusions based on the data obtained

5. Background/History of the Study

In 1910, England began building what they claimed to be the most luxurious cruise ship to ever be made and even boasted it to be unsinkable. Despite the claims of being unsinkable, on April 15, 1912, the ship hit an iceberg and sunk, leaving 1503 people dead and becoming one of the world's greatest tragedies to this day.

The Titanic – Machine Learning From Disaster is a very popular problem on Kaggle and there is a lot of documentation about it. However, I did not see anyone try to solve this problem using MapReduce, so I figured that would be an interesting take on a problem that has already been solved so many times over. I believe that using MapReduce can be valuable in this situation because it is enough entries to be able to reduce, but not such a large amount that it would be difficult to handle, leading this to be a great project for beginners and anyone new to Hadoop.

6. Approach and Implementation

The main approach to the problem is to first clean the data, run a K-Means with Python to get the initial centroids, put these centroids into the MapReduce function, map the index with a centroid value, calculating the new center points which are used in the next iteration and outputting the final centroids.

- Cleaning the data: In order to clean the data, we must first only use the columns which pertain to our problem at hand. We can do this by dropping columns such as Name that have no bearing on whether or not a person would have lived or died. In our case, we only examined Age and Sex, but there are many other combinations of columns someone could examine to make an inference about the survival of the passengers.
- Python K-Means: This is a preliminary step which we use to get our initial centroids from the data, and then we will import these centroids into the MapReduce program.
- Putting Centroids in MapReduce: Ideally, the MapReduce program would not need the centroids hard coded in, but in my case, I could not get it to work without an input file containing them and hard coding them in. Only one or the other resulted in errors or an incorrect output.
- Mapping the Index: the variable “i” represents the closest center point, and this is mapped using the index variable.
- Mapping the Centroid Value: Using the computeDis function, we are able to calculate the euclidean distance between the centroids and create sample values mapped with the index for the Reduce function.
- Calculating New Center Points with Reduce: By feeding the Reduce function our key and value, we can calculate the new center points for the next iteration until the Reduce is completed.

7. Experiment Results and Discussion

The results actually were identical to the centroids output by the Python K-Means which was very shocking. I assumed they would be close or a little off, but after fixing my calculations they were exactly the same which was amazing to see.

One part of this program that I am very proud of is the versatility of it. Someone could easily give it different columns and with a few minor adjustments, it

would output the centroids for completely different combinations of columns. This project was an eye opener as to how powerful MapReduce can be, and I would love to test it on an extremely large dataset and see what it is truly capable of. It feels a little bitter-sweet seeing as how my input data wasn't large enough to warrant a Combine function, but if I ever revisit a data science project that can utilize it, I definitely will write another MapReduce program.

This project taught me so much about Hadoop and Linux; I feel like an expert navigating and manipulating my VM now.

8. Conclusion

In conclusion, the centroids suggest that younger women were less likely to die, which is true if we reference the data file and it is well documented that women and children were the first to be evacuated while the Titanic was sinking. Given this program, any of the other conditions could be clustered and graphed to be analyzed also.

I am extremely happy with the results. I've never felt something as satisfying getting the MapReduce to work and seeing the results at the end with no errors. Hadoop is very powerful which of course requires it to have many moving parts, and it feels so great to get them all working and getting an output. In hindsight, I wish I did my project on something more interesting, but regardless this was a good place to start for a beginner like myself, so it only made sense.

9. References

K-Means Map Reduce: https://bbnewpaltz.sln.suny.edu/bbcswebdav/pid-3586885-dt-content-rid-16482487_2/courses/fall21_CPS593_04/parallelkmeansmapreduce_zhao.pdf

Python Code to Clean Data: https://www.youtube.com/watch?v=pUSi5xexT4Q