**MSc in Data Analytics – Integrated Continuous Assessment**

Valentina Martucci SBS23006

[sbs23006@student.cct.ie](mailto:sbs23006@student.cct.ie)

November 10th, 2023

Table of Contents

[1. Introduction 3](#_Toc150026569)

[2. Data processing and storage 3](#_Toc150026570)

[2.1 Methods and tools 3](#_Toc150026571)

[2.2 Data storage and processing 4](#_Toc150026572)

[2.2.1 Database selection 4](#_Toc150026573)

[3 Time series 6](#_Toc150026574)

[3.1 Analysis 6](#_Toc150026575)

[3.2 Models 7](#_Toc150026576)

[3.3 Dashboard 8](#_Toc150026577)

[4 Conclusion 8](#_Toc150026578)

[References 9](#_Toc150026579)

[Appendix – Dashboard user guide 12](#_Toc150026580)

# Introduction

In the era of digital transformation, the ability to process and analyze large volumes of data, or “Big Data”, has become a critical success factor for businesses across various sectors. Through the application of advanced data processing techniques and forecasting models, this project strives to provide a comprehensive solution for handling Big Data and conducting robust time series analysis.

This document is structured as follows. Section 2 describes the tools used during this project in relation to data management. Section 3 focuses more on the actual data, its analysis and forecasting models. Conclusions for this work are then summarised in Section 4.

# Data processing and storage

## 2.1 Methods and tools

A dataset containing 1.6M observations has been provided for this project. The initial step has been choosing an appropriate database to store the data, process its content, and then store the processed data back in the database for further analysis and modelling. The outcome of the big data processing, along with modelling techniques, has been then summarised in an interactive dashboard. The diagram below represents the data phases, along with the tools used. Additionally details on those steps are provided in the next section.

A diagram of a software company

Description automatically generated

Figure - Project Structure

Three Jupyter notebooks have been produced for this study and, as this is a data analysis project, modules like *pandas*, *matplotlib*, *sns* and other have been used for data manipulation and visualization.

Report’s notebooks, along with datasets and report can be found at

<https://github.com/sbs23006/MsC_BigData_AdvAnalytics/>

The report wordcount (including titles, references, and all sections excluding Appendix) is 3480.

## Data storage and processing

The initial step has been the evaluation of one SQL database (MySQL) and one noSQL (HBase). As the number of records is quite high, some benchmark tests have been run against those two databases to evaluate the best performance for this dataset. *Yahoo! Cloud Servicing Benchmark (YCSB)* framework has been used, being a widely recognized tool for evaluating the performances of different database management systems, including both SQL and NoSQL databases [https://benchant.com/blog/ycsb].

### Database selection

Tests with different workload have been performed on both databases to evaluate their performance sensitivity to data size change. YCSB metrics have been collected for each and compared.

The first metric that has been analysed is runtime, that indicates the total execution time for the test operations. It was observed that MySQL runtime grows significantly as the size of the test data increases, making HBase more suitable for big data storage.

A graph showing a number of numbers

Description automatically generated with medium confidence

Figure - Runtime comparison between MySQL and HBase

The second metric compared has been throughput, that indicates the measure of how many units of information a system can process in a given amount of time. Also in this case, HBase results in being the best choice as it can process more units than MySQL.

A graph with a line and a red line

Description automatically generated

Figure - Throughput comparison between HBase and MySQL

Another important metric when looking at databases performances is latency, that is the total amount of time that it will take for the database to receive a request, process the underlying transq, and return the correct response. Also in this case, HBase results in being quicker that MyQSL when inserting a bigger amount of data.

A graph showing the difference between latency and num operations

Description automatically generated

Figure - Latency comparison between HBase and MySQL

A NoSQL database is clearly the best choice for this project. Besides HBase, MongoDB has been also considered but, despite the great support from the open-source community, it required more configurations to be able to work with tools like Spark or pymongo to process the data in a Jupyter notebook. HBase has therefore been chosen due to its native integration with Apache Hadoop. Evidence of all those evaluations can be found in the accompanying Jupyter notebook *DB benchmark + Evaluation.ipynb*, as well as within the Big Data section in the file *DataProcessing.ipynb*.

### Data processing

Bla Blab la

# Time series

## 3.1 Analysis

Blab la bla to be completed.

## 3.2 Models

Blab la bla to be completed.

## 3.3 Dashboard

Bla Bla to be completed.

# Conclusion

Blab la to be completed.

# References