Big Data Tools & Techniques for MSc

Analysis of clinical trial data

Assessment Title:	Analysis of clinical trial data				
Module Title:	Big Data Tools & Techniques				
Module CRN Code:	41141 (September) 50194 (January)	**Level:**	7	**Semester:**	2
Programme Code(s):	MST/DS/F1, MST/DS/P1 (September) MST/DS1/F1, MST/DS1/P2 (January)	**Issue date:**[1]		Week 7 (14/03/2022)	
Weighting:	100% of the total module mark	**Submission date:**[2]		End of week 12 (13/05/2022 16:00)	
Assessor(s):	Judita Preiss	**Return date:**		An unratified final mark will be available 3 weeks after submission or submission date (whichever is later).	

[1] Date on which brief is to be given to students.
[2] Date by which assessment is to be submitted.

Learning outcomes of this assessment

The learning outcomes covered by this assignment are:

- Provide a broad overview of the general field of 'big data systems'
- Developing specialised knowledge in areas that demonstrate the interaction and synergy between ongoing research and practical deployment of this field of study.

Key skills to be assessed

This assignments aims at assessing your skills in:

- The usage of common big data tools and techniques
- Your ability to implement a standard data analysis process
  - Loading the data
  - Cleansing the data
  - Analysis
  - Visualisation / Reporting
- Use of Python, SQL and Linux terminal commands

Recommended Reading

The module notes complimented by tools and techniques covered in other modules are sufficient literature for completing this assignment successfully.

For reference documentation:

- Spark documentation (`https://spark.apache.org/documentation.html`)
- Hive documentation (`https://cwiki.apache.org/confluence/display/Hive/Home`)
- MySQL documentation (`https://dev.mysql.com/doc/refman/5.5/en/`)
- Python documentation (`https://developers.google.com/edu/python/introduction` and `https://matplotlib.org/users/intro.html`)
- AWS documentation (`https://docs.aws.amazon.com/`)

Equipment and Facilities to be Used

For this assignment, only the platforms used in the course (Databricks and Amazon Web Services) are to be used. All processing must be done via executable notebooks, scripts and code, and these must be stored and included with the submission. Terminal commands must be stored in shell scripts, language specific code has to be stored in separate files (for example, HiveQL code must be stored in .sql scripts).

Your solution must be implemented using both HiveQL and PySpark (note that the PySpark version **cannot** use SQL queries directly – such a solution will not receive any marks).

Workload

For the successful completion of this assignment, a total of 120 hours should be budgeted.

Task

You will be given a dataset and a set of problem statements. You are required to implement your solution to each problem in both HiveQL and PySpark.

General instructions

You will follow a typical data analysis process:

1. Load / ingest the data to be analysed
2. Prepare / clean the data
3. Analyse the data
4. Visualise results / generate report

For steps 1, 2 and 3 you will use environments that have been used within this module. The data necessary for this assignment will be downloadable as .csv files.

The .csv files have a header describing the file's contents. They are:

1. `clinicaltrial_<year>.csv`: each row represents an individual clinical trial, identified by an *Id*, listing the sponsor (*Sponsor*), the status of the study at time of the file's download (*Status*), the start and completion dates (*Start* and *Completion* respectively), the type of study (*Type*), when the trial was first submitted (*Submission*), and the lists of conditions the trial concerns (*Conditions*) and the interventions explored (*Interventions*). Individual *conditions* and *interventions* are separated by commas. (Source: ClinicalTrials.gov)
2. `mesh.csv`: the conditions from the clinical trial list may also appear in a number of hierarchies. The hierarchy identifiers have the format [A-Z][0-9]+(.[A-Z][0-9]+)* (such as, e.g., D03.633.100.221.173) where the initial letter and number combination designates the root of this particular hierarchy (in the example, this is D03) and each "." descends a level down the hierarchy. The rows of this file contain condition (*term*), hierarchy identifier (*tree*) pairs. (Source: U.S. National Library of Medicine.)

3. `pharma.csv`: the file contains a small number of a publicly available list of pharmaceutical violations. For the puposes of this work, we are interested in the second column, *Parent_Company*, which contains the name of the pharmaceutical company in question. (Source: `https://violationtracker.goodjobsfirst.org/industry/pharmaceuticals`)

When creating tables for this work, you **must** name them as folows:

- clinicaltrial_2021 (and clinicaltrial_2019, clinicaltrial_2020 for the sample data)
- mesh
- pharma

The uploaded datasets, if used, must exist (and be named) in the following locations:

- /FileStore/tables/clinicaltrial_2021.csv (similarly for sample data)
- /FileStore/tables/mesh.csv
- /FileStore/tables/pharma.csv

This is to ensure that we can run your notebooks when testing your code (marks are allocated for your code running).

You are to implement all steps (at least) twice: once in HiveQL and once in Spark using PySpark.

For the visualisation of the results you are free to use any tool that fulfils the requirements, which can be tools you have learned about such as Python's matplotlib, SAS or Qlik, or any other free open source tool you may find suitable. Using built in visualizations directly is permitted, it will however not yield a high number of marks. Your report needs to state the software used to generate the visualization, otherwise a built in visualization will be assumed.

Extra features to be implemented

To get more than a "Satisfactory" mark for the module, a number of extra features should be implemented. Features include, but are not limited to:

- Writing general and reusable code: for example, ensuring that switching to a different version of the data requires only the change of one variable.
- Two separate implementations of the Spark part, one via dataframes, the other via RDDs (note that using SQL directly in the Spark part will not count towards your mark).
- Implementation using an AWS cluster. In this case, scripts and screenshots need to be supplied to ensure reproducibility.
- Refinement (with justification) of the basic implementations.
- Further analyses of the data, motivated by the questions asked.
- Creation of additional visualizations presenting useful information based on your own exploration which is not covered by the problem statements.

The data

You will be using clinical trial datasets in this work and combining the information with a list of pharmaceutical companies and condition hierarchy information. You will be given the answers to the questions, for a basic implementation, for two historical datasets, so you can verify your basic solution to the problems. Your final submission will need to consist of results executed on the third, 2021, release of the data. All data will be available from Blackboard.

Problem statements

You are a data analyst / data scientist whose client wishes to gain further insight into clinical trials. You are tasked with answering these questions, using visualisations where these would support your conclusions.

You should address the following problem statements. You should use the solutions for historical datasets (available on Blackboard) to test your implementation.

1. The number of studies in the dataset. You must ensure that you explicitly check distinct studies.
2. You should list all the types (as contained in the *Type* column) of studies in the dataset along with the frequencies of each type. These should be ordered from most frequent to least frequent.
3. The top 5 conditions (from *Conditions*) with their frequencies.
4. Each condition can be mapped to one or more hierarchy codes. The client wishes to know the 5 most frequent roots (i.e. the sequence of letters and numbers before the first full stop) after this is done.

   To clarify, suppose your clinical trial data was:

   ```
   NCT01, ... ,"Disease_A,Disease_B",
   NCT02, ... ,Disease_B,
   ```

   And the mesh file contained:

   ```
   Disease_A A01.01 C23.02
   Disease_B B01.34.56
   ```

   The result would be

   ```
   B01 2
   A01 1
   C23 1
   ```

5. Find the 10 most common sponsors that are not pharmaceutical companies, along with the number of clinical trials they have sponsored. *Hint:* For a basic implementation, you can assume that the *Parent Company* column contains all possible pharmaceutical companies.
6. Plot number of completed studies each month in a given year – for the submission dataset, the year is 2021. You need to include your visualization as well as a table of **all** the values you have plotted for each month.

Report

A 3000 word report that documents your solution should be included with your submission. In this module, a background, literature review or citations are **not** required. The format of the report should be as follows:

1. Description of any setup required
2. Data cleaning and preparation (including descriptions, justifications and screenshots of all code)
3. Problem answers
   (a) Question 1
   - Assumptions made
   - PySpark implementation outline (description of main ideas in words and screenshot of code)
   - HiveQL implementation outline (description of main ideas in words and screenshot of code)
   - (Optional) PySpark implementation outline (the second - DF or RDD - PySpark implementation, description of main ideas in words and screenshot of code)
   - (Optional) AWS implementation commands (description of main ideas in words, screenshot of code and results)
   - Result on the submission (final) data – results for all implementations must be included (screenshots are fine)
   - Discussion of result

   (b) Question 2
   - Assumptions made
   - PySpark implementation outline (description of main ideas in words and screenshot of code)
   - HiveQL implementation outline (description of main ideas in words and screenshot of code)
   - (Optional) PySpark implementation outline (the second - DF or RDD - PySpark implementation, description in words and code)
   - (Optional) AWS implementation commands
   - Result on the submission (final) data – results for all implementations must be included (screenshots are fine)
   - Discussion of result

   (c) Same arrangement for remaining questions

   (d) (Optional) further analysis 1 (to include an explanation of why the analysis is being / should be performed, implementation description and code, result and discussion)

   (e) (Optional) further analysis 2 (to include an explanation of why the analysis is being / should be performed, implementation description and code, result and discussion)

Further analyses can range from implementations to program outlines, marks will be awarded in proportion to the complexity of the analysis and the amount of work done. At a minimum, a program outline needs to also include a justification of why this would be useful, what would be gained by it, and a rough indication of how it would be expected to be performed.

If the distinct implementations for a question are along similar lines, you may find it easier to highlight the differences.

Requirements & Marking Scheme

Requirement	Assessment Method	Weight (%)
Homework	Programs / demos throughout semester	20%
Data loading and preparation	Report	20%
Data analysis	Report	30%
Report	Report	30%

Notes

- The assignment must be completed on your own: this includes all the programs and report. By the act of following these instructions and handing your work in, it is deemed that you have read and understand the rules on plagiarism as written in the academic handbook (see also the Unfair means point below).
- The assignment must be completed on time. If you submit work late, it will be marked according to the University's late submission policy.

Unfair means

The University has strict policies on unfair means. It is your responsibility to ensure that you both understand these and adhere to them in the production of your assignment. Any submitted works with such content identifiable will be penalised in accordance with the University of Salford regulations

`https://www.salford.ac.uk/governance-and-management/academic-handbook`

Submission

The assignment needs to be uploaded in two parts. The report is uploaded via Turnitin (so that an originality report is generated for you), while all your code / notebooks and scripts need to be gathered together in a zip file and submitted via Blackboard. Note that both items are required for a complete submission!

All your filenames should start with your last name. E.g. if your last name is "Smith" and you have produced a file named `report.pdf`, you should upload this as `smith_report.pdf`.

The following items must be included in your submissions:

- **Turnitin submission** – a single text file:
    1. A report in either Word or PDF, such that Turnitin generates an originality score for it.
- **Blackboard submission** – a zip folder containing:
    1. A PySpark notebook named with your name prepended, for example `smith_rdd.ipynb` and / or `smith_df.ipynb`. This should include the results of your run on the 2021 dataset. An HTML download from Databricks including the results of the execution should also be included, named `smith_rdd.html` and / or `smith_df.html`.
    2. A HiveQL solution to the questions with your name prepended, for example `smith_hiveql.sql`. An HTML download from Databricks including the results of the execution should also be included, named `smith_hiveql.sql`.
    3. Optionally, a folder named `scripts` containing any script files for data loading and / or AWS implementation. All scripts must contain comments where appropriate.

Note that submissions must be made to both Turnitin and the Blackboard area for a complete submission!

It is assumed that you will also address any social / legal and ethical issues surrounding the implementation of the project such as copyright, references, licenses, and web law.

Assessment Criteria

The following assessment criteria are provided as a guide to the criteria that you need to satisfy in order to get a grade within each of the following ranges.

Extremely poor (0-9)

- Totally inadequate demonstration of required knowledge.
- Not able to apply the practical and analytical skills from their programmes.
- No appropriate design methodology.
- No demonstration of analysis evaluation or synthesis.
- No evidence of the ability to self-manage a significant piece of work and critical self-evaluation of the process.
- Little academic value; presentation is extremely poor; work has no structure or clarity; extremely poor use of language; no references; no attempt to provide evidence of sources used.

Very Poor (10-19)

- Virtually no relevant knowledge demonstrated.
- Fails to adequately apply the practical and analytical skills from their programme.
- Very poor use of design methodology.
- No meaningful analysis or evaluation or synthesis.
- Unable to self-manage a significant piece of work and to identify appropriate issues for critical self-evaluation of the process for reflection.

- Academic arguments presented are inappropriate or very poorly linked; presentation is very poor; work has little discernible structure or clarity; very poor use of language; lack of ability to source adequate material; very poor referencing.

Poor (20-29)

- Inconsistent or inaccurate knowledge.
- Limited and inappropriate and inaccurate application of the practical and analytical skills from their programme.
- Poor use of methodology.
- Descriptive, occasional attempts to analysis or evaluate material but lacks critical approach to evaluation or synthesis.
- Identifies issues for reflection but lacks evidence of reflective processes.
- Some but inconsistent ability to self-manage a significant piece of work or critical self-evaluation of the process.
- Confusion or weakness in academic argument; presentation is poor; work is disorganised and lacks clarity; poor use of language; poor use of reference material; inappropriate or out dated sources with numerous referencing errors.

Inadequate (30-39)

- Limited evidence of knowledge.
- Inappropriate application of the practical and analytical skills from their programme.
- Unsatisfactory design methodology.
- Mainly descriptive evidence of analysis, inconsistent critical approach, little evaluation or synthesis.
- Follows processes of reflection but fails to demonstrate insight; lacks coherence in the self-management of a significant piece of work.
- Presentation is unsatisfactory; work is limited in terms of structure, coherence or clarity; limitations in academic style; unsatisfactory referencing with errors; limited ability to support content with relevant sources.

Unsatisfactory (40-49)

- Basic knowledge with occasional inaccuracies.
- Appropriate yet basic application of the practical and analytical skills from their programme.
- Superficial depth or limited breadth, but an overall adequate identification of design methodology.
- Critical analysis evident, with some evaluation and synthesis, although limited evidence of reflection.
- Some evidence of an ability to self-manage a significant piece of work and critical self-evaluation of the process.
- Some appropriate academic argument although not well applied and lacking in clarity; presentation of work is adequate in terms of structure, coherence, clarity and academic style; some inconsistencies; some grammar and syntax errors which detract from the content; narrow range of sources; referencing in presented work is adequate with some inconsistencies or inaccuracies; over utilises secondary sources; references used are inappropriate in terms of currency.

Satisfactory (50-59)

- Mostly accurate knowledge with satisfactory depth and breadth of knowledge.
- Solid application of the practical and analytical skills from their programme
- Fair use of design methodology.
- Sound critical analysis and evaluation or synthesis.
- Demonstrates basic ability of synthesise information in order to formulate appropriate questions and conclusions; reflective process is utilised, with insight demonstrating planning for future practice; shows the ability to self-manage a significant piece of work and critical self-evaluation of the process.
- Relevant academic argument; presentation of work is fair in terms of structure coherence, clarity and academic style; some inconsistencies in grammar and syntax; fair range of sources identified with appropriate referencing and few inaccuracies; appropriate use of primary and secondary sources.

Good (60-69)

- Consistently relevant accurate knowledge with good depth and breadth.
- Clear and relevant application of the practical and analytical skills from their programme.
- Good use of design methodology.
- Clear, in depth critical analysis, evaluation and academic argument with synthesis of different ideas and perspectives.
- Utilises reflection to develop self and practice; aware of the influence of varied perspectives and time frames; demonstrates an ability to self-manage a significant piece of work and critical self-evaluation of the process.
- Presentation of work is well organised with good use of language to express ideas or argument; very few inconsistencies in grammar and syntax good; good range of sources; well referenced with very few inaccuracies; good use of primary and secondary sources.

Very Good (70-79)

- Comprehensive knowledge demonstrating very good depth and breadth.
- Clear insight into links between the practical and analytical skills from their programme.
- Strong use of design methodology.
- Very good analysis and synthesis of material with evidence of critical and independent thought.
- Demonstrates ability to transfer knowledge between different contexts appropriately; balanced and mature approach to reflection used to enhance practice and performance; clear ability to self-manage a significant piece of work and critical self-evaluation of the process.
- Presentation is of a very good standard, demonstrating a scholarly style. Very good grammar and syntax. Clear evidence of referencing to a wide range of primary and secondary sources which are used effectively in supporting the work.

Excellent (80-89)

- Excellent depth of knowledge in a variety of contexts.
- Coherent and systematic application of the practical and analytical skills from their programme.
- Excellent use of design methodology.
- Excellent critical analysis and synthesis.

- Integrates the complexity of a range of knowledge and excellent understanding of its relevance; confident in their ability to self-manage a significant piece of work and critical self-evaluation of the process
- Arguments handled skilfully with imaginative interpretation of material; presentation is excellent, well-structured and logical; demonstrates a scholarly style; excellent grammar and syntax.

Outstanding (90-100)

- Outstanding knowledge.
- Exceptional application of the practical and analytical skills from their programme.
- Excellent professional execution of design methodology.
- Outstanding critical analysis and synthesis.
- Excels in self-managing a significant piece of work and critical self-evaluation of the process show an aptitude to formulate new questions, ideas or challenges.
- Incorporates evidence of original thinking; presentation is outstanding demonstrating a fluent academic style.