# Data engineering project

Report on the MongoDB + OULAD dataset

**Put together by** **Kory Frankee** (fran0618)

**Word count: ~** 575of 3,000 words

**Page count: ~** 2 of 10

(Excludes references, figures, title page, and appendices)

## What I’ve done

#### I. Data loading

I wanted to load the data from the Mongo DB and OULAD datasets straight into a tibble (as that would be more efficient that loading it into a dataframe and then converting it into a tibble)

To accomplish this, I developed a “data loader” class. **Within it, I wrote functions for:**

* Putting data retrieved from the account of a MongoDB user > from a cluster > from a MongoDB collection into a tibble
* Putting data from a .csv file into a tibble

#### II. Data observations

To help understand the data, I wrote a “data checker” class. Within it, I wrote functions for:

* Checking for duplicate primary key IDs
* Checking that at least one of the student ID, assessment ID, presentation ID, and course IDs was unique in each row even if there may be duplicate key values in the column

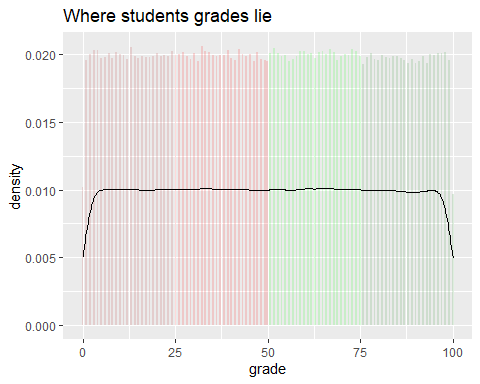
After loading in the data from the MongoDB dataset, I discovered that the scores were nested

A screenshot of a graph

Description automatically generated

Of the 400,000 rows, there were 10,000 students and 501 classes with a unique ID, 3 assignment types (‘exam’, ‘homework’ and ‘quiz’), and 400,000 unique scores in the range of 0 to 100.

(Conclusion) The strange scores show that the mock data is unrealistic. (P1) Markers don’t tend to provide scores with upwards of 7 decimal places. (P2) Marks don’t tend to be all different to one another when the number of assignments marked is high, which it is. (P3) Marks tend to have a minimum of 0 as some students receive a 0 for not handing in their work. Here, the minimum score is ~ 0.0002. (P4) Marks tend to have a maximum of 100 as some students receive full marks for their work. Here, the maximum is ~ 99.99. (P5) Scores tend to have a normal distribution. Here, I’ve provided a visualization of the distribution of scores for all the assignments to show how uniform it is



After looking at the data from the OULAD dataset, I had several problems with it.

1. Missing scores.

A screenshot of a computer

Description automatically generated

1. Use of quantitative data for if result is transferred from a previous presentation instead of nominal data (‘0’/’1’ values are less clear than ‘no/’yes’ values)
2. Undescriptive column names[[1]](#endnote-1).
3. Wrongly entered age values (‘55<=’)

A screenshot of a computer

Description automatically generated

1. Wrongly entered IMB band values (’10-20’)
2. Missing IMD band values

A screenshot of a computer

Description automatically generated

1. Missing when student unregistered values

A screenshot of a computer

Description automatically generated

1. Missing when student registered values

A screenshot of a computer

Description automatically generated

1. Missing exam due dates

A screenshot of a computer

Description automatically generated

#### III. Data cleanup

I wanted to address a few of the major problems that I had with the data, so I developed a “data cleanup” class. **Within it, I wrote functions for:**

* Unnesting a column
* Finding and replacing values
* Removing columns
* Putting values in a column into a vector of length 1 with comma-separated terms and quotation marks between each term
* Setting the names of columnsand rows

I replaced missing or incorrect values with sensible values. I, for example, wrote a function to work out the most common IMD value for each of the regions with missing values. I then filled in the missing IMD values for a particular region with the most common IMD value for that region. I, for another example, knew that the missing exam dates were *“on the last week of the course”* .

#### IV. Data additions

I added a grades column to the MongoDB dataset, which I filled in with tertiary grades (‘F’, ‘P’, ‘Cr’, ‘D’, and ‘HD’) based on what score students received for their assessments and based on how Australians universities apply these grades[[2]](#endnote-2) [[3]](#endnote-3).

I added a new weighted score column for the OULAD dataset, which I derived from [[4]](#endnote-4)

#### V. Data subsets and supersets

To help divide up the data, I wrote a “data subsetter” class with functions for:

* Querying the data using SQL

To help join data together, I wrote a “data supersetter” class with functions for:

* Merging tables

I joined the OULAD tables that I was able to join (without R Studio freezing) into one big table

**(Appendix 04)**

#### VI. Data analysis

To help analyse the data, I wrote a “data analysis” class with functions for:

* Working out the cleanliness of the data (number of missing values, number of rows with unique values, number of unique values in a column, and number of duplicate values)
* Working out statistical information (mean, mean of group, median, maximum, minimum, quartile value, quartile value for group, standard deviation, z-score, & range of values). **By “mean of group”, I mean this**

A screenshot of a table

Description automatically generated

**To do:**

* Work out what the easiest class is
* Work out what the most difficult class is

#### VII. Data visualization

To help visualize the data, I wrote a “data visualization” class with functions for:

* Graphing a scatter plot
* Graphing a complicated scatter plot that makes use of two datasets

**(Appendix 01)**

* Graphing a histogram
* Graphing a bar chart

**(Appendix 02)**

* Graphing multiple bar charts side by side

(**Appendix 03**).

* Getting the inverted colours for a list of colours (used for distinguishing the colours of labels from colours of the plot objects)

**To do:**

* Utilize other graphs

#### VIII. Data modelling

**To do:**

* Simple linear model
* General linear model w/
  + Categorical predictors
  + Categorical and continuous predictors
  + Continuous predictors
* Evaluate the performance of the models
* Evaluate how interpretable the models are

#### Appendices

##### MongoDB dataset visualizations

###### Appendix 01

|  |
| --- |
| Here, I’ve how the selected student has performed compared to other students in the same classes using an average of their scores for all their assessment items as the basis of comparison. |
| Showing it in action...  With student ID of 0    With student ID of 1 |

###### Appendix 02

|  |
| --- |
| Visualization of the least to most common grades students receive for their assessment items. This reveals that more students succeed with their assignments than fail at them by a small margin. This means that students are underperforming or that the markers are quite harsh. |
|  |

###### Appendix 03

|  |
| --- |
| Visualization of student scores across two different classes. **Useful insights could be gathered from this side by side analysis if:**   * We could compare the grades for a given class with previous years to see if there has been an improvement in student performance. |
|  |

##### OULAD dataset schema

###### Appendix 04

|  |
| --- |
| Shows the original schema |
|  |

|  |
| --- |
| Shows the new schema |
|  |

##### OULAD dataset visualizations

#### References

1. https://analyse.kmi.open.ac.uk/open\_dataset [↑](#endnote-ref-1)
2. Australian Education Info 2024, *“Grading system in Australia”,* <https://www.australiaeducation.info/Education-System/Grading-System.html>, viewed March 2024 [↑](#endnote-ref-2)
3. University of Sydney, *“How do I work out my final grade?”,* <https://usqassist.custhelp.com/app/answers/detail/a\_id/2615/~/how-do-i-work-out-my-final-grade%3F>, viewed March 2024 [↑](#endnote-ref-3)
4. <https://usqassist.custhelp.com/app/answers/detail/a_id/2615/~/how-do-i-work-out-my-final-grade%3F> [↑](#endnote-ref-4)