# DATA420-25S2 (C)

## Assignment 2

## The Million Song Dataset (MSD)

Due on Friday, October 17 by 5:00 PM.

If you want to discuss the assignment material you can use the Discord server where the discussion will benefit all. If you have a question that requires an official answer you can use the forum on LEARN. If you have a more personal question you can email me or contact the class rep as needed.

A reminder that the Discord server is for discussion of concepts only, not for sharing code or answers to assignment questions.

**Links**

Report upload (pdf)
Supplementary material upload (zip, limited to 10 MB)

Discord server
Help forum for Assignment 2

## Instructions

- Your report should be submitted as a single pdf file on LEARN. Any additional code, images, and supplementary material should be submitted separately as a single zip file on LEARN. You should **not** submit any outputs as part of your supplementary material, leave these in cloud storage.

- The body of your report should be between 3,000 and 5,000 words long, excluding your cover page, table of contents, references, appendices, and supplementary material. You need to be accurate and concise and you need to demonstrate depth of understanding.

- You should make sensible choices concerning margins, font size, spacing, and formatting. For example, margins between 0.5” and 1”, a sans-serif font e.g. Arial with font size 11 or 12, line spacing 1 or 1.15, and sensible use of monospaced code blocks, tables, and images.

- You should reference any external resources using a citation format such as APA or MLA, including any online resources which you used to obtain snippets of code or examples. You must reference any use of Grammerly, ChatGPT, or any other generative AI tools to **improve** the quality of your own original work.

- You **must not** use any content generated by AI directly in your report or your supplementary material. You are encouraged to use AI to help you solve problems and develop code, but you should take time to understand any content that you use so that you develop accurate depth of understanding.

# GRADING

The assignment is graded across a number of categories which are summarised in the table below.

| | |
|---|---|
| **Answers** | 14 |
| **Reasoning** | 45 |
| **Tables** | 9 |
| **Visualizations** | 11 |
| **Writing** | 11 |
| **Coding** | 10 |

The marks for the answers, reasoning, tables, and visualizations categories are distributed across the sections in the assignment, the marks for the writing section are based on your report as a whole, and the marks for the coding section are based on your code as a whole.

**Writing**

- Was the report well structured overall?
- Did the report go into a suitable amount of detail and demonstrate depth of understanding?
- Was writing concise and was the report easy to understand?
- Was writing natural and professional?
- Was any code included in the report or was it only in the supplementary material?
- Were any external resources appropriately cited and referenced?
- Was any use of AI appropriately acknowledged?

**Coding**

- Were notebooks well structured and easy to navigate and understand?
- Were any empty cells, exceptions, or other anomalies left in the notebook?
- Was code style consistent and readable overall?
- Was code commented appropriately?
- Was supplementary material provided and was it well structured?

You should structure your assignment report based on the high level comments below, and then check that you have satisfied the grading criteria the processing, analysis, and visualization sections across each of the answers, reasoning, tables, and visualizations categories.

## Structure

Your report should have the following sections within which you can also use question numbers as subheadings to group paragraphs, tables, and figures that you use to answer the questions that have been asked. You should keep your writing concise and easy to understand, and you should provide enough detail to demonstrate depth of understanding.

### Background

- You should give a brief overview of the purpose of the assignment and what you have achieved or understood with your processing, analysis, and modeling.
- You should provide a high level summary of the Million Song Dataset (MSD), similar to that already included in the assignment brief, which will provide context for the description of the structure and content of the data in the processing section below.

### Processing

- You should describe the structure and content of the datasets which you can refer back to in the sections below. You should describe the steps that you took to load, join, and check the data, answer the questions that have been asked, and discuss anything else that you discovered.
- You should **not** include outputs other than answers to the questions that you have been asked.

### Audio similarity

- You should describe the target for your classification algorithms, the features that you used, and how you trained the algorithms to predict binary and multiclass outcomes. You should describe the strengths and weaknesses of the algorithms and how you chose the hyperparameter values that you used. You should explain any decisions that you made about feature selection, splitting, sampling, hyperparameters, and metrics. You should discuss the performance of the algorithms and talk about how you would do hyperparameter tuning.
- You should answer the questions that have been asked, give a high level summary of what you have done, and discuss any insights that you had. You should talk about any tasks that you were unable to complete and explain why.

### Song recommendations

- You should describe the distributions of user-song play counts, describe any choices you had to make to use this data for collaborative filtering, and talk about the performance of the collaborative filtering model using specific examples and the ranking metrics that you have evaluated.
- You should explain the implications of the choices you had to make, discuss any other systems that you would need to generate recommendations for **all** users of your service, and discuss any other considerations for using the collaborative filtering model to generate recommendations.
- You should answer the questions that have been asked, include visualizations, give a high level summary of what you have done, and discuss any insights that you had.

## Conclusions

- You should give an overview of what you have achieved and what you have learned.

## References

- You should list all references that you have used or referenced.

## Processing

This section is about developing your understanding of the **structure** of the data and setting up code that you will need to load and join the datasets so that you can use them to develop models and answer questions in the audio similarity and song recommendations sections.

### Answers

- Names, sizes, formats, data types, and number of rows in each of the datasets.

### Reasoning

- A brief summary of the information that is contained in the audio attributes datasets and why these are separate from the audio features datasets.
- A clear explanation of how you used these to create a StructType automatically, including any additional processing.
- A clear explanation of how you renamed columns in the audio feature datasets. You should explain any decisions that you made and why. Answers to any other questions that you have been asked.

### Tables

- A table containing names, sizes, formats, data types, and number of rows in each of the datasets.

### Visualizations

- A directory tree showing how the datasets are organized.

## Audio similarity

This section is about demonstrating that you can develop binary and multiclass classification algorithms to predict the genre of a track from audio based features.

You should demonstrate your understanding of the audio feature datasets, how to prepare suitable features for training, how to train binary and multiclass classification algorithms, and how to evalute their performance effectively. You should also demonstrate your knowledge of hyperparameter tuning in theory.

### Answers

- The class balance of the binary classification target as a count and a proportion or ratio.
- Sensible values for binary accuracy, precision, and recall.
- Sensible values for multiclass performance metrics that account for class balance in some way.

### Reasoning

- A brief summary of the audio features datasets based on your background reading.
- Any conclusions that you can make from the descriptive statistics and if they will influence how you train or inference your model.
- A clear explanation of any choices you need to make based on how the features are correlated.
- A comment on the distribution of genres and how it will affect your performance metrics.
- A clear explanation of how you have split the dataset into training and test sets, including any stratification or resampling techniques that you have used.
- A clear explanation of each classification algorithm that takes into account their explainability, interpretability, predictive accuracy, training speed, hyperparameter tuning, dimensionality, and any additional preprocessing that might be required.
- A clear explanation of how logistic regression can be used for multiclass classification and if there are any additional assumptions that need to be satisfied.
- A clear explanation of how cross-validation works and why it is used for hyperparameter tuning, and how you would use it to tune the hyperparameters of a classification algorithm in general based on this understanding.
- A comment on the relative performance of each binary and multiclass model.

### Tables

- A table containing the descriptive statistics for the audio features or at least a representative sample of the audio features if there are too many to fit on one page.
- A table containing a summary of the classification algorithms, their advantages, disadvantages, and any comments on how they should be used.

- A table of per genre performance metrics.
- A table of hyperparameters for each of the classification algorithms, with concise definitions of the hyperparameters and how they affect the performance of each model.

**Visualizations**

- A heatmap of feature correlations before and after correlated features have been removed.
- A visualization of the distribution of genres.
- A visualization of per genre performance metrics.

## Song recommendations

This section is about demonstrating that you can develop a viable song recommendation system based on collaborative filtering.

### Answers

- How many unique songs and unique users there are in the Taste Profile dataset overall.
- How many unique songs has the most active user played.
- Any other descriptive statistics that are relevant to how the data is distributed.
- Examples of recommendations and songs that a user has actually played, including any relevant song or artist metadata that you need to make a qualitative comparison between the items and the recommendations.

### Reasoning

- A brief discussion about the advantages and disadvantages of repartitioning and caching the dataset taking into account the inherent properties of the dataset and how it will be used.
- A clear explanation of how you have defined song popularity and user activity.
- A clear explanation of how you filtered users and songs to achieve the thresholds specified and any other steps that you took to prepare your dataset for training.
- An explanation of why every user in the test set must have some user-song plays in the training set as well and how you achieved this given the number of users and items involved.
- A clear explanation of each ranking metric and why they are useful even if we only have a historical snapshot of user interactions.
- A detailed explanation of how you would evaluate the performance of recommendation systems in the real world in an online way.

### Tables

- A table of descriptive statistics and an explanation of their meaning.
- A table of the number of users and songs that remain in the dataset or have been excluded.
- A table of ranking metrics for your model.

### Visualizations

- A visualization of the distributions of song popularity and user activity.