Machine Learning & Data Mining
Miniproject 1

Caltech CS/CNS/EE 155
Released February 3, 2026

---

1 Overview

For this miniproject, you will use audio features of songs to predict whether a song is likely to be low popularity or high popularity. The dataset focuses exclusively on audio-based features such as energy, danceability, and acousticness, ensuring that predictions are made based on the sonic and structural characteristics of the songs.

In miniproject 1, you will predict whether a song is of low popularity or high popularity based solely on its audio features. This task eliminates identifying features (such as track name, artist, or album) to focus exclusively on understanding how a song's sonic characteristics correlate with its popularity.
We will host the competition on Kaggle, a site for data science competitions. There can be 10 submissions per day per team, and the submission window will close on Tuesday February 10th at 5:00 PM PST. **You may not use any additional data sources.**

In this competition, use the training data (`train.csv`) to come up with predictions for the test data (`test.csv`). To help you estimate your model's generalization, there will be a public leaderboard to help you estimate your performance on the "public" test set. The leaderboard ranking on the separate private test set will only be revealed when the whole competition ends.

The competition website, which includes the datasets and guidebooks describing the datasets, can be found here: https://www.kaggle.com/t/01f984f104294f4baf9a583c70e5f112

There will be a benchmark submission added by the TAs. We encourage you to beat this benchmark.

Your task:

Each row in `train.csv` represents the different features of a song. The last column in `train.csv` is `Popularity_Type`, which indicates whether the song is of high or low popularity, with high popularity being represented by 1 and low popularity being represented by 0.
Your task is to predict the target in `test.csv` to the best of your ability. A few helpful notes on the dataset:

- It is generally encouraged to submit probabilities from your models instead of 0/1 predictions, as you get rewarded for having a non-zero probability of the correct class even if it is not the highest probability for that sample. This also represents real world conditions, where you would like to know the relative likelihoods of each class being the right one.
- If you submit many times, be careful of overfitting to the public test set.

Please follow the format in the sample submission files (`sample_submission.csv`) when generating your submissions to Kaggle.

Performance metric:

Your model will be evaluated using AUC: the Area Under the receiver operating characteristic Curve.

2 Competition Logistics

- The competition ends on Tuesday, February 10th at 5:00 PM PST. Late hours cannot be used for the competition.
- You should work in groups of two to four students (or auditors). To join teams, each participant should sign up for the competition separately, then go to the `Team` tab and `Send Invitation`.
- You can make up to 10 submissions a day. However, at the end, you will select the two submissions that you think will perform the best on the private test set.
- If you have questions, please ask on Piazza! We also encourage starting early and making a preliminary submission as quickly as possible.
- You can use open-source tools, concepts you learned in class, and other additional techniques you find online (except for existing code written to model this particular loan dataset) to get the best score that you can.
- **You may collaborate fully within your team, but no collaboration is allowed between teams.**
- **You may not search for additional data related to this task; you may only train and validate your models using the provided training set.**
- The top 50% of groups (private leaderboard) will receive a small extra-credit bonus (up to 5 points, based on ranking). However, we care more about your process and rationale than your model performance. The report and Colab are worth the large majority of your grade.

3 Written Deliverables

- **Report (90 points):** Additional details below. The report should be written to the length specifications. If you have additional insights, you can include them in the extra credit section. We encourage to use graphs in your report and Colab demo, as visualization is very helpful!
- **Colab Demo (10 points):** To help the class learn from each other, you should share a Colab notebook that runs your best-performing model and generates several of the visualizations included in your report. You can include exploratory data analyses, feature engineering, parameter curves, model ensembling, and/or more. Here is a particularly extensive example. Please share the public, read-only Colab link on Piazza in a public note with your team name, and attach the Piazza post link and the Colab link in your report.
- **Due date:** Wednesday, February 11th at 11:59 PM PST, via Gradescope. Late submissions will use late hours from all group members.
- **Please submit your report in groups rather than submitting it once per student.** You can see how to submit in groups here.

4 Report Guidelines

We recommend that you use the LaTeX template provided to you and simply fill in the blanks. To collaborate on the report writing, we recommend using Overleaf (https://www.overleaf.com/edu/caltech), an online LaTeX editor. Caltech students can get a pro account for free using caltech.edu emails.

See our example file for guidelines. The structure is as follows:

1. **Introduction (15 points):** This section is purely for the TAs and should be brief. Maximum of 1 page.
   - Group members
   - Kaggle team name
   - Ranking on the private leaderboard
   - AUC score on the private leaderboard
   - Colab link
   - Piazza link
   - Division of labor: Your team should collaborate so that each member has a similar workload. If there is a noticeable discrepancy in the division of labor, team members may receive differing grades.
2. **Overview (15 points):** Concisely summarize your attempts. Detailed explanations should go in the next section. Recommended length of half-page. Maximum of 1 page.
   - Models and techniques tried: What models did you try? What techniques did you use along with your models? Did you implement anything out of the ordinary?
   - Work timeline: What did your timeline look like for the competition?
3. **Approach (20 points):** This section should be a more detailed explanation of how you approached the competition. Maximum of 1 page.
   - Data exploration, processing and manipulation: Did you manipulate the data or the features in any way, such as data cleaning or feature engineering? What techniques and libraries did you use to accomplish such manipulation? Please justify your methodologies.
   - Details of models and techniques: Why did you try the models and techniques that you used? What was that process like? What are the advantages and disadvantages of using such methods?
4. **Model Selection (20 points):** This section should outline how you chose the best models. Maximum of 1 page.
   - Scoring: What optimization objectives did you use, and why? How did you score your models, and why? Which models scored the best?
   - Validation and test: How did you split your data? Did you use validation techniques? How did you test your models? What were the results of these tests, and what did the results tell you?

5. **Conclusion (20 points):** Summarize the report and your experience. Maximum of 1 page.
   - Insights: Please answer the following questions
     - Among all the features in the data, which features have the most influence on the prediction target? How did you determine which features were important? List the top 10 features. (Bonus points if you can analyze whether these 10 features positively or negatively influence the prediction target.)
     - Overall, what did you learn from this project?
   - Challenges: What could you have done differently? What obstacles did you encounter during the process?
6. **Additional insights (optional, 5 extra-credit points):** you can mention any insights you found interesting that you didn't have space to mention above. Be creative! Examples:
   - Why do we use AUC as our Kaggle competition metric? Do you think there is a better metric for this project? Why, or why not?
   - Among the machine learning methods that your group used, are there any methods or pipelines that can be parallelized? If so, how? If not, why not? (You are not required to actually parallelize your code)
   - What additional data would be useful to collect for this problem, and why?

5 Conclusion

Feel free to ask questions on Piazza and in office hours. Most of all, have fun!

6 LLM Usage

This section has an example of LLM usage reporting below. Please follow this format to report LLM usage. Indicate each usage clearly.

- Name of LLM(s) Used: ChatGPT
- Components of Project involving LLM: Data processing.

Data preprocessing (example)

Interaction (example)

Only include the parts of the response that you integrated into your project. You may submit screenshots. For example, Figure 1 illustrates the interaction with LLM for data preprocessing. Alternatively, you can also write out the prompt and response in text.

Usage/Integration of Responses (example)

As shown in Figure 2, We used the response as follows, modifying the name of the encoder object and applying it to our training dataframe data, in column some_column.

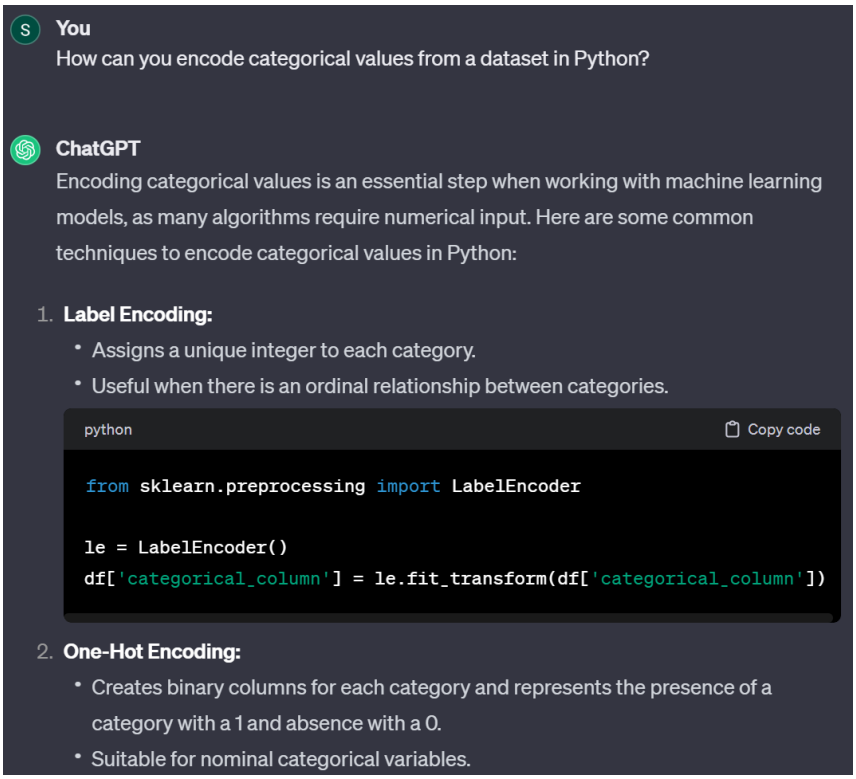

Figure 1: Interaction with LLM for data preprocessing.

```
from sklearn.preprocessing import LabelEncoder

enc = LabelEncoder()
data['some_column'] = enc.fit_transform(data['some_column'])
```

Figure 2: Integration of LLM's output for data preprocessing