

# Mini Project 3

## Junctional Rhythm Prediction

Due: June 12, 2026, 11:59 PM PT

*Student Name:*

*Instructor: John Lipor*

**Instructor Note:** I will not answer questions after June 5, 2026, so be sure to start early.

## 1 Introduction

For the final mini project of the quarter, you will tackle a brand new open research problem: predicting junctional rhythm (JR) from pediatric ambulatory electrocardiogram (ECG) recordings. For our purposes, JR is a result of congenital heart disease in which only one ventricle functions and is indicated by the absence of the P wave in ECG recordings.<sup>1</sup> The JR burden denotes the percent of time a patient spends in JR and may be a clinically useful quantity for identifying interventions to improve outcomes for such patients. However, quantifying JR burden is difficult, requiring long ECG recordings and expert analysis to be accurately determined. Thus, accurately and automatically quantifying JR burden could lead to improved outcomes for patients with this condition.

To obtain a labeled dataset for predicting JR burden, we (myself and collaborators) recruited a group of ten electrophysiologists to label individual beats as “junctional” (P wave absent), “sinus” (P wave present), or “unknown” (less than 75% confident in beat classification). Each labeler was shown 500 beats from each of 10 patients, for a total of 5000 labeled beats in the dataset. Since the ground truth is unknown for these beats, we used the Dawid-Skene expectation-maximization algorithm [1] to estimate the sensitivity and specificity of each labeler. The resulting average sensitivity was 0.957 and specificity was 0.962, indicating broad agreement among the experts. Further, we found that JR burden ranged from 2–97% across patients.

Your task is to train a three-class classifier to automatically predict whether beats correspond to JR, sinus, or are too difficult to make a determination, using the above dataset for training. A successful algorithm will obtain near expert-level performance. For this project, you will receive all available labeled data (no separate training and test sets), and it is up to you to develop an evaluation approach, including how to split, whether to average over many splits, and what scoring metric(s) to use. *Regardless of your algorithm’s final performance, you must make a compelling case that you have accurately estimated the generalization performance of your model.*

---

<sup>1</sup>This is likely a gross oversimplification.

## 1.1 Notes

- (a) DSS rules apply for all mini projects.
- (b) Be sure to cite any sources you used, including/especially Kaggle notebooks.
- (c) You may use an LLM (e.g., ChatGPT) to aid with this project. However, for any section that you find an LLM useful, you must specify how you used it. For example, you may need to state something like, “ChatGPT helped me understand the components of exploratory data analysis, and I used some lines of generated code to complete this task.” You also must understand every line of code used in your project.
- (d) For this project, you are free to use any type of predictor or available library, including neural networks.
- (e) If your results in any way inform my thinking about this problem, I will invite you to be a co-author on any research papers I ultimately write.

## 2 Getting Started

Since we have not studied training of neural networks, I am providing an example solution (`JR_demo.ipynb`) that trains a simple one-dimensional convolutional neural network. This example demonstrates how to work with the data and a neural network and is not intended to be a strong solution in terms of network architecture or generalization.

The demo also provides an overview of the dataset and plots a few examples of correctly classified and misclassified traces. The dataset fields include `X`, the ECG waveform segments; `y` and `y_encoded`, the expert labels in string and integer form; `img_names`, the corresponding image names; `patient_index`, a zero-based patient identifier; `labelers`, the anonymized expert labeler IDs; `label_names` and `label_codes`, the class mapping; and `missing_label_code`, the integer code used for missing labels.

## 3 Requirements

You must create a report in either  $\text{\LaTeX}$  or a similar (instructor approved) typesetting language that contains the following sections:

- (a) Problem description (5 pts)
  - Explain what data you have to work with, what algorithms you will use, what your goal is, and why anyone should care.
- (b) Exploratory data analysis (EDA, 20 pts)
  - **(NEW)** For this dataset, feature engineering will likely be limited.
  - **(NEW)** You must perform a data analysis across beats and across patients.

- EDA is the first step in any data science pipeline and is a systematic effort to understand your data *before* modeling it. Think of it as an iterative, question-driven process: you form a hypothesis about the data, generate a visualization or statistic to test it, draw a conclusion, and let that conclusion raise new questions. The goal is not to produce a gallery of plots, but to develop working knowledge of your data that directly informs the decisions you make in your Approach.
- A useful self-check: every figure or table in your EDA should connect to at least one decision you describe in your Approach section. If a finding did not change anything about how you modeled the data, ask yourself whether it belongs in your report.
- There are many excellent tutorials on EDA online; be sure to cite whichever ones you found helpful and reflect on them in your “What I Learned” section. Some starting points are [2, 3, 4, 5].
- Do not simply provide plots and tables without interpretation. Describe the significance of each finding and, where applicable, what action it prompted.

(c) Challenges (10 pts)

- What were the challenges you encountered when applying machine learning to this dataset?
- Did these challenges mainly result from the data? From results? Installing libraries to perform preprocessing?

(d) Approach (30 pts)

- **(NEW)** For this project, you must decide on and justify your scoring metric. Describe the scoring metric both qualitatively and mathematically, as well as why you believe this is the correct scoring metric for this problem.
- **(NEW)** Describe your data splitting approach and how it will be used to estimate generalization error. For example, is a single train/test split sufficient? Your goal is to be confident that your evaluation results are an accurate estimation of real-world performance.
- **(NEW)** You may wish to generate synthetic data to enlarge the training set. I have included a simple synthetic data generator (built entirely by AI) that may or may not be a good start. Note that your final scoring/validation must be measured only on held out real data.
- Provide a thorough but concise description of your approach, including (but not limited to) your approach to wrangling, preprocessing, and improving the performance of your predictor.
- This should be much clearer than what is typically found in Kaggle notebooks, but make sure it stays concise. A summary paragraph at the beginning of this section could be helpful.

- Since we have now formally studied model selection, you must use either a validation set or cross validation to tune your model parameters. Be sure to describe how you went about this and why.

(e) Evaluation and summary (20 pts)

- **(NEW)** Your evaluation must include a comparison with the expert labelers. You must decide how to perform this comparison and justify your approach.
- **(NEW)** Your evaluation must consider per-patient performance in addition to overall performance.
- Describe why/where your approach falls short in terms of estimation and approximation error. You may wish to generate a learning curve, as described by Ch. 11 of the text.
- Dive into the data to search for trends regarding which points were misclassified and why.
- Summarize your solution, describing what worked, what didn't, what the main limitations are, and any general conclusions about the dataset.

(f) What I learned (5 pts)

- Describe the main skills/tools that you learned and used for this project and how you learned them.
- You must run your project through an AI system to obtain feedback, then incorporate this feedback where appropriate. My recommendation is to upload this document and your report, then prompt, "Evaluate my report with respect to the grading criteria," or something similar. Give a high-level description of any feedback you found particularly helpful.

## 4 Grading

The grading breakdown is below. Each section will be graded according to technical correctness, effort, and creativity. Note that clarity of writing is a major component. You should put yourself in the place of writing for a boss or senior in a workplace. If your writing is terrible, they will assume your work is terrible.

Item	Percentage
Description	5%
EDA	20%
Challenges	10%
Approach	30%
Evaluation	20%
What I learned	5%
Clarity/conciseness of written communication	10%

## References

- [1] A. P. Dawid and A. M. Skene, “Maximum likelihood estimation of observer error-rates using the em algorithm,” *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 28, no. 1, pp. 20–28, 1979.
- [2] (2023) Data wrangling in python. [Online]. Available: <https://www.geeksforgeeks.org/data-wrangling-in-python/>
- [3] (2023) Exploratory data analysis in python. [Online]. Available: <https://www.geeksforgeeks.org/exploratory-data-analysis-in-python/>
- [4] (2022) A simple tutorial on exploratory data analysis. [Online]. Available: <https://www.kaggle.com/code/spscientist/a-simple-tutorial-on-exploratory-data-analysis>
- [5] S. Ray. (2016) A comprehensive guide to data exploration. [Online]. Available: <https://www.analyticsvidhya.com/blog/2016/01/guide-data-exploration/>