ASSIGNMENT #1: Data Featurization and Modeling

Due electronically January 25th at midnight (11:59pm Toronto time)

Goal: For the Materials Project Database, implement a standard data analysis and modeling pipeline using machine learning following these steps:

1. Research the dataset for this project.
2. Define the research question: regression or classification? Choose one.
    - Regression Task: Predict the bandgap of a material
    - Classification: Classify a material as a conductor (bandgap=0), semiconductor (0 < bandgap <=4), or insulator (bandgap > 4)
3. Preprocess the dataset so that it is clean and ready for use.
4. Featurize the dataset using a method of your choice, e.g. Matminer or Magpie featurizers
5. Perform dimensionality reduction on the featurized dataset using a method of your choice (e.g., PCA, TSNE, UMAP, Spectral Embedding, etc.).
6. Apply a cluster labeling algorithm to the reduced-dimension data
7. Use feature selection to down-select features suitable for your task. Explain how you selected these features.
8. Decide on a cross-validation strategy, e.g. K-fold, leave-one-cluster-out (LOCO), etc.
9. Train either Random Forest Regressor or Random Forest Classifier models to answer the research question.

In a research setting, a materials scientist cannot perform every experiment. Instead, the scientist uses their domain expertise to recognize the most promising experiments and performs as few as possible to obtain the desired conclusion. In a self-driving laboratory, the scientist's domain expertise must be transferred to an algorithm that minimizes the number of experiments performed while maximizing the ability to predict the outcome of the experiments which are **not** performed. You will practice this kind of thinking as the final goal of this assignment. Your training dataset represents the experiments which have been completed, and your test dataset represents the experiments which have not been completed but which you want to know something about.

10. Report and implement a method to identify the minimum number of samples in the training dataset required to predict the test dataset. Here are some resources to help you think about this problem:
    - Minimax
    - Information Gain and Mutual Information for Machine Learning
    - Reduced Data Sets and Entropy-Based Discretization

Submission instructions: The expected deliverables must be in the form of Jupyter Notebooks. Please do not convert it to a PDF file. Your notebook must run to receive credit for the assignment.

Make sure you have a registered GitHub account, then set up a **private** GitHub repository named "MSE403H/MSE1003H_YourName_YourStudentNumber". After that, please add both Jae (jae3goals) and me (daleas0120) as collaborators so we can access your submissions. Instructions for adding collaborators are here.

Reminders

- Prevent data leakage! Use good statistical practices to prevent contamination of your test data set. Before you begin your analysis, think about how you will ensure this and document your code to make it clear that this has happened.
- Keep good repository hygiene: commit changes often and make sure there are instructions for how to run all code. This includes how to access data downloads or install any needed python packages.
- You are encouraged to use a coding assistant (Cursor, GitHub CoPilot, etc.). However, code comments and written explanations must be your own. Replace any AI-generated code comment with your own explanation and add more comments as appropriate.
- Cite your resources. If there is a research paper or code repository which you referenced as part of your work, make sure to provide a citation in your notebook (a hyperlink is fine). Reading scientific literature is encouraged.
- To ensure reproducibility, set the random seed to 403 or 1003 for all tasks involving randomness.
- You are welcome to message Ashley with assignment questions on Quercus or at her email (Ashley.dale@utoronto.ca).

Notebook Contents: Your Jupyter notebook is a technical writing assignment as much as it is a coding assignment and science experiment. Emphasize conciseness and clear explanations in your report, along with visualizations and other details necessary for a reader to understand your thought process.

In addition to the code, the notebook should have the following sections:

- **Introduction:** A brief description of the dataset and your research objectives
- **Analysis:** present your code and results, organized logically with markdown explanations
- **Summary:** Key insights, conclusions, and reflections on your findings

Figures: Your notebook should contain the following visualizations. Figures should have axis labels and legends.

1. Dataset Contents and Featurization. This can be through statistical measures, example samples from the dataset, etc.
2. Reduced Dimensionality Dataset. Use a color bar to indicate how the property of interest is distributed among this space
    a. For PCA, please also justify your selection of the number of PCs using cumulative explained variance plots.
3. Reduced Dimensionality Dataset labeled by cluster. Use a color bar to indicate cluster ID.
4. Model performance visualizations for the training data and testing data.
    a. For regression tasks please use parity plots.
    b. For classification tasks, please use confusion matrices.
5. Cross validation results.
6. One or more figures showing the method and results of minimizing the number of required training samples.

Evaluation Criteria

	Excellent (A+)	**Good (A)**	**Acceptable (B)**	**Poor (C)**	**Unacceptable (D)**
Coding	Strong organization indicating logic. Markdown shows separate major steps with detailed comments including relevant equations and theory.	Good organization, comments, code implements important methods but lacks some small details. Highly readable.	Some organization, comments indicate major sections but do not explain logical flow or important details	Little organization or logical flow. Few comments, or comments are AI generated. Missing results.	No organization or comments. Missing most results.
Correctness of Experiment	Methods indicate deep understanding of underlying assumptions and/or domain expertise. Cleverness and novelty.	Code can run with a single click and no errors. All methods implemented are correct for the task. Strong justification is provided for method choices.	Code runs but might use patches to accommodate a failed earlier step. A few poor method choices or weak justification. No random seed.	Some methods are inappropriate for the task, more than one method is incorrectly implemented.	Code does not run, or there are significant methodological errors including data leakage between train and test.
Analysis	Informed skepticism that identifies key assumptions and tests them. Work identifies novel research directions.	Formulates a valid scientific question. Correct conclusions based on results and correct reporting of implications for future work.	Comments on most results but misses a key aspect or assumption. Analysis does not consider implications for future work.	Comments miss most key results, do not comment on major trends. Conclusions are trivial. Conclusions do not reference results.	No analysis, incorrect conclusions, unfounded conclusions