IML term project 2025

Team: Overfit & Chill (Group 52)
Members:
Christos Zonias
Kokoro Horiuchi
Mikhail Golubkov

Date: December 7, 2025

Preliminary Report

1. Introduction

This project focuses on the classification of atmospheric new particle formation (NPF) events using environmental sensor data collected at the Hyytiälä forestry field station (SMEAR II mast). NPF refers to the process by which small molecular clusters form into larger aerosol particles, influencing cloud formation, weather patterns, and air quality. Understanding the conditions under which NPF occurs is of significant scientific and environmental importance, particularly for assessing air pollution dynamics and climate interactions.

The dataset contains daily aggregated meteorological and aerosol measurements, including temperature, humidity, and condensation sink, recorded at multiple heights. Each day is labeled with an event type: nonevent (no NPF) or one of three NPF event classes (Ia, Ib, II). The primary task is to develop a binary classifier to distinguish between event (any NPF type) and nonevent days. The secondary task involves multi-class classification across the four event categories.

From a machine learning perspective, this represents a supervised classification problem with structured, clean data. The objective is not to achieve state-of-the-art predictive performance, but to systematically apply and evaluate data preprocessing, feature selection, model training, and validation techniques. The project emphasizes methodological rigor, interpretability, and reasoned decision-making over pure accuracy, providing a practical exercise in building, analyzing, and reporting on a real-world classification pipeline.

2.1 Data Acquisition and Initial Setup

Libraries Used:

- pandas for data manipulation
- numpy for numerical operations
- matplotlib.pyplot for core plotting library

Column Structure:

- Each sensor measurement appears as pairs of .mean and .std columns.
- Sensors include: CO2168, CO2336, CO242, CO2504, Glob, T672, T84, UV_A, UV_B, CS, and others.
- Additional columns: id, class4, date, partlybad.

Removed Columns (Data Handling):

- date - since test data does not include date, it is not possible to utilize it for predictions
- partlybad
- id

Feature Analysis:

- Identified 10 features with minimal variance (top 4 are CS.std, CS.mean, PTG.std, PTG.mean)
- Variance range: 4.6e-07 (CS.std) to 2.1e-02 (SO2168.std)

2.2 Basic Ideas and Solution Approach

Proposed Solution Strategy #1: Simple Supervised Learning Methods
After analysing the scoring criteria, it was clear for us that the classifier should be probabilistic and handle multiple classes. We decided to evaluate the following models:

- Naive Bayes
- Softmax logistic regression
- LDA, QDA
- Random Forest
- SVM
- Gradient boosted trees

The hyperparameters would be tuned automatically based on cross-validation scores. The models would be evaluated on multiple dataset variants: plain (all features), reduced (with only selected, skew-corrected features), and reduced with PCA applied. All data was standardized.

Well-performing models could then be combined into an ensemble.

Proposed Solution Strategy #2: Two-stage approach
After analyzing the scoring criteria, we observed that the evaluation combines both binary (event vs. nonevent) and multiclass (II, Ia, Ib, nonevent) predictions. This motivated a two-stage approach:

Stage 1: Binary Classification (Event Detection) First, we simplify the problem to detecting whether an event occurred (class2: event vs. nonevent). Two complementary models are trained:

- XGBoost - gradient boosted trees with strong predictive power
- SVM (RBF kernel) - captures nonlinear decision boundaries

The models are combined via weighted probability averaging, with the optimal mixing weight α tuned on validation data to minimize perplexity.

Stage 2: Multiclass Classification (Event Type) For the full 4-class prediction, we again leverage an ensemble:

- XGBoost with `multi:softprob` objective for calibrated probabilities
- SVM with probability calibration

The ensemble is used only if it outperforms XGBoost alone on validation accuracy.

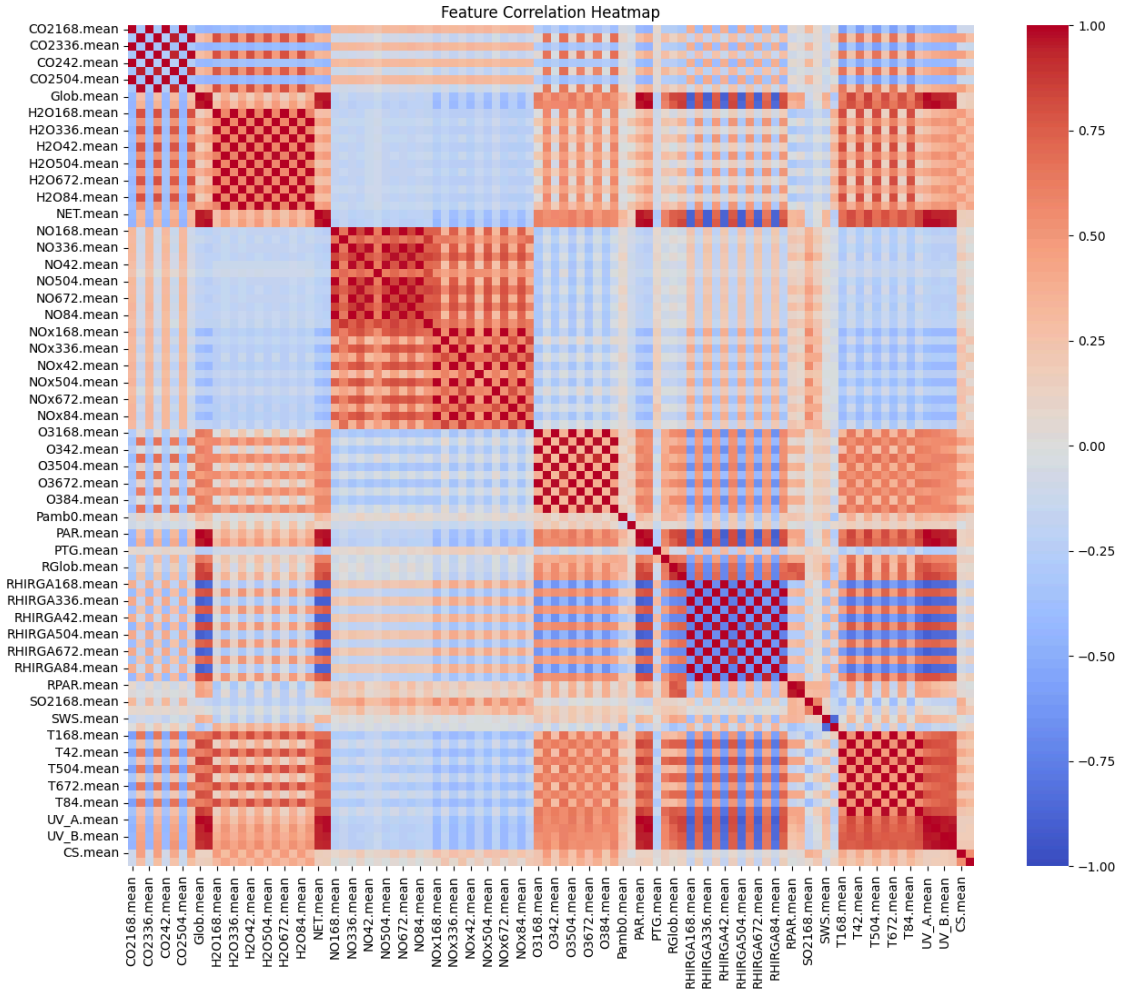
The predicted **class** is the one with the highest probability from the multiclass classifier. The output **probability** is taken from the binary classification stage (event/nonevent probability).

Feature Engineering: Standard scaling applied to all features.

3. Data Analysis Stages

3.1 Initial Data Exploration

- Created feature correlation heatmap. This would help identifying and removing dependent variables.



- To identify which features are most promising for distinguishing between the four event classes:

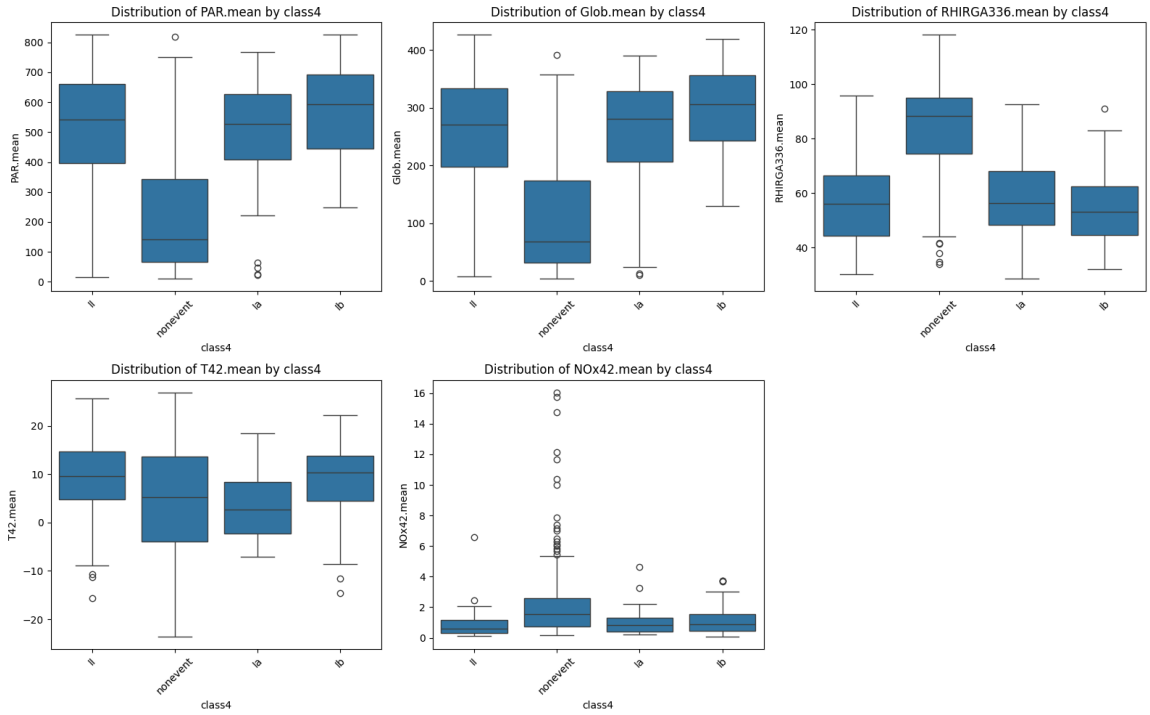


Figure 1: EDA

- Scatterplots of key feature pairs reveal that NPF events (Ia, Ib, II) primarily occur under high radiation and low humidity conditions. Nonevent days cluster in regions of low `PAR.mean`/ `Glob.mean` and high `RHIRGA336.mean`. Temperature provides secondary separation, with nonevents frequently occurring on cold, humid days and events on warmer, sunnier days. The three event classes partly overlap, indicating that subclassification (Ia, Ib, II) is inherently more difficult than binary event/nonevent classification.

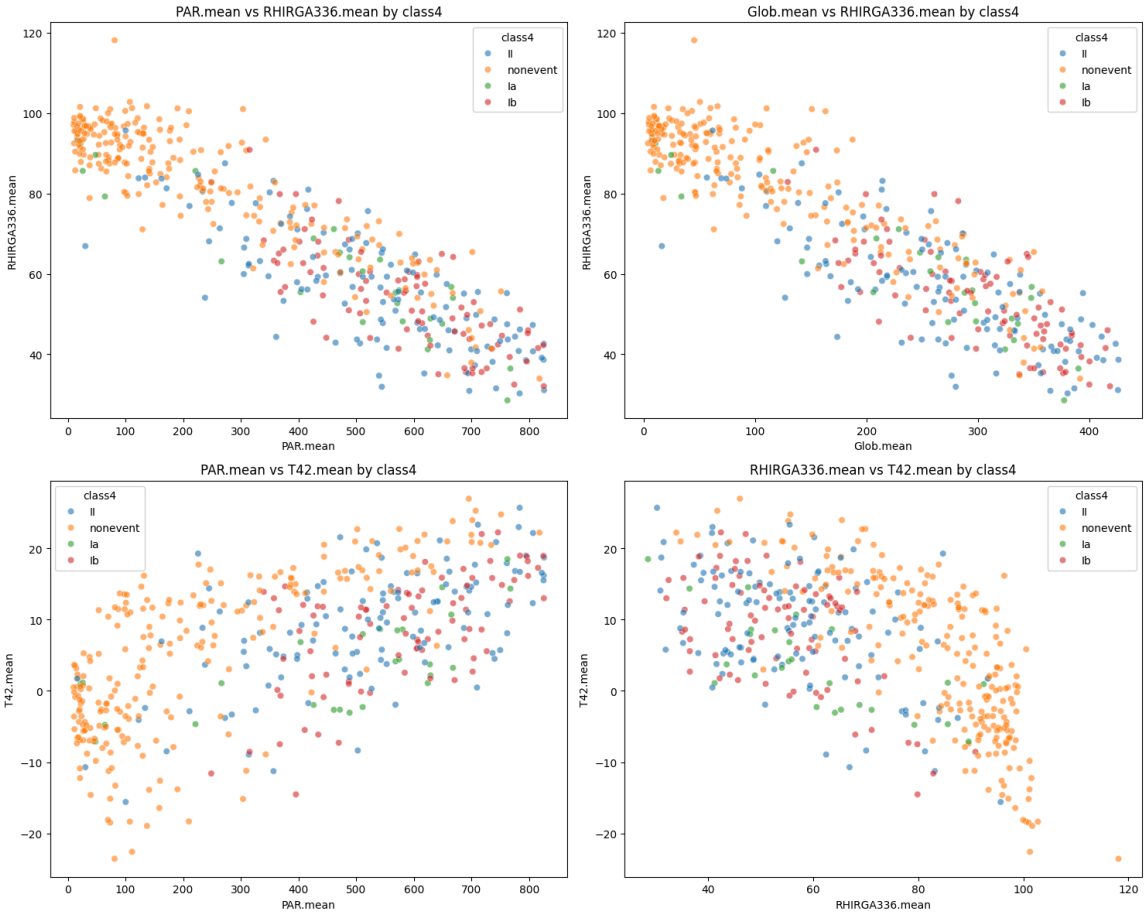
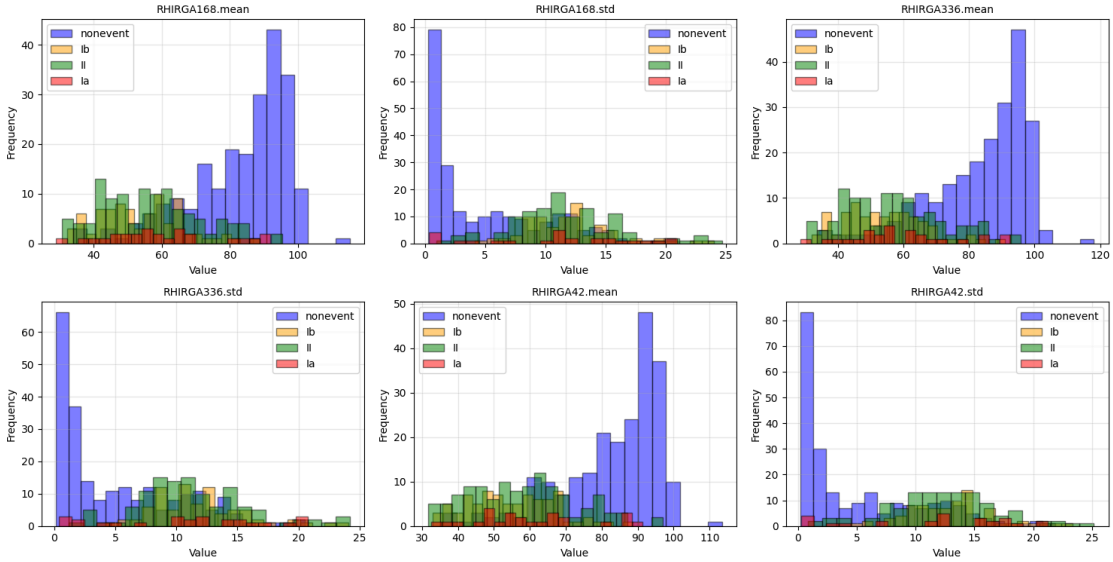


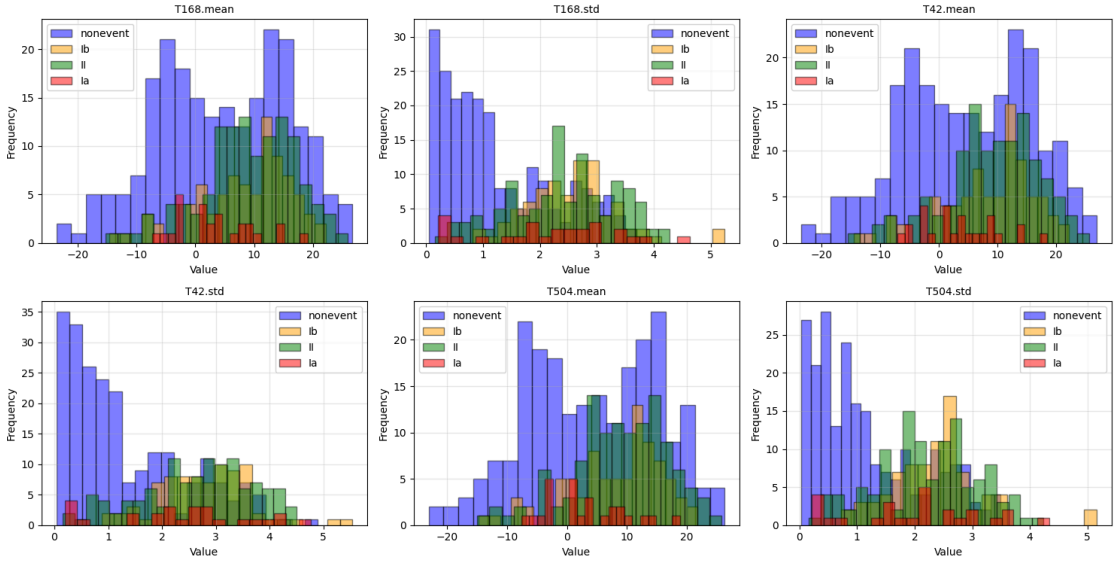
Figure 2: Scatterplot

- We calculated correlations between variables and event classes. The analysis showed that the RHIRGA feature group separated event/nonevent well, but for multiclass classification, no single variable provided clear separation. We plotted per-class distributions of each variable to identify those with the best class separation. Some of the strongest discriminators are shown below:

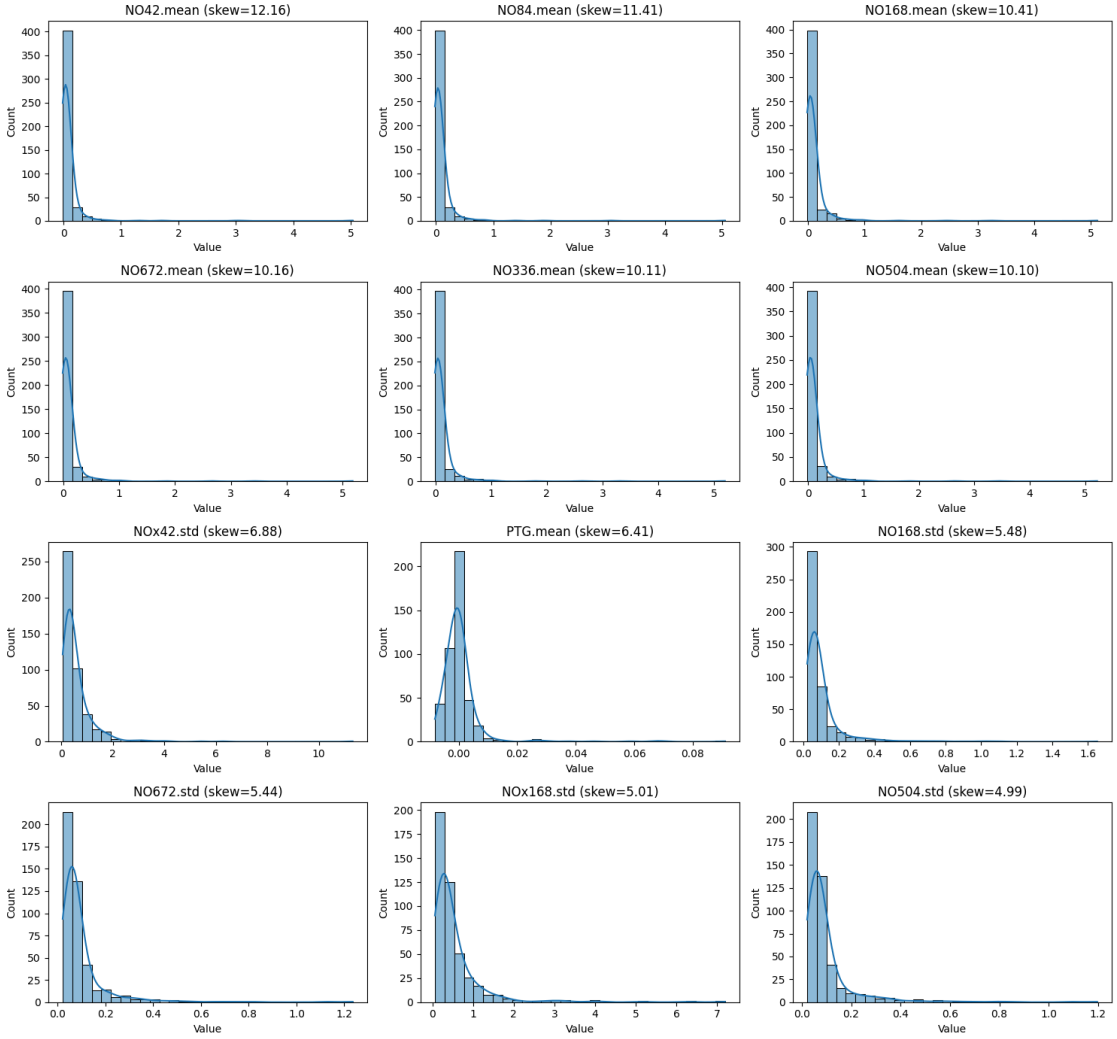
RHIRGA Distributions by Class



T Distributions by Class



- We also observed that many features are skewed, not following a normal distribution closely.



- We verified that the data does not contain any missing values or empty rows.

Based on these observations, we designed the following preprocessing pipeline to evaluate alongside the raw data: 1. Drop non-feature columns 2. Drop features with low variance 3. Fix skewness 4. Remove highly correlated features 5. Standardize 6. (Optional) Apply PCA

As a result, we obtained two new datasets with 32 and 15 features, respectively.

4.1 Results: Simple supervised methods

We evaluated a range of classical machine learning models on the 4-class NPF event classification task, testing each model across different feature preprocessing strategies: plain (all features), selected features (reduced set), and dimensionality reduction via PCA.

Note: Accuracy values below refer to cross-validation class prediction accuracy, not Kaggle scores.

Linear Discriminant Analysis (LDA): LDA achieved 0.633 accuracy with all features, and feature selection approach yielded the best LDA result at 0.722. This suggests that LDA benefits significantly from reduced, decorrelated feature spaces where its linear decision boundaries can operate more effectively.

Quadratic Discriminant Analysis (QDA): QDA performed slightly worse than LDA, achieving 0.622 with all features and 0.6 with filtered data.

Naive Bayes (NB): Naive Bayes showed the most dramatic variation across preprocessing strategies. With all features, it achieved only 0.467 accuracy-the worst among all models-reflecting the violation of its feature independence assumption in the highly correlated sensor data. Performance improved substantially to 0.678 when combined with intelligent dimensionality reduction and PCA, which decorrelates features and better satisfies the independence assumption. Notably, it was the only classifier capable of predicting Ia class events, which motivated its inclusion in ensemble experiments.

Random Forest (RF): Random Forest achieved 0.700 accuracy on plain data, demonstrating its robustness to feature scaling and correlation. However, performance dropped to 0.678 with reduced features and PCA, suggesting that the tree-based splits benefit from the full feature space and that PCA's linear transformations may obscure decision boundaries that RF naturally captures.

Softmax Logistic Regression: Softmax regression proved to be one of the most consistent performers. Plain features yielded 0.700 accuracy, while L1 regularization improved this to 0.733-the best single-model result. The L1 penalty effectively performs implicit feature selection by shrinking irrelevant coefficients to zero. The reduced feature dataset achieved 0.711 accuracy, and adding PCA dropped performance to 0.644, indicating that explicit feature selection slightly underperforms L1′s automatic selection.

Support Vector Machine (SVM): SVM with RBF kernel achieved 0.711 on plain data, competitive with softmax regression. The reduced feature dataset yielded the same accuracy of 0.711, while adding PCA degraded performance.

HistGradientBoostingClassifier: The gradient boosting model achieved 0.644 accuracy with a log loss of 0.793.

Ensemble Methods: Combining models yielded mixed results. A simple softmax + Naive Bayes ensemble achieved 0.689 accuracy, slightly below either model's best individual performance. However, stacking softmax with Naive Bayes-using a logistic regression meta-learner with both

models' predictions as features-achieved 0.733 accuracy, matching the best softmax result but with potentially better generalization.

We also experimented with shuffling data entries to ensure models were not learning temporal trends, but in all cases this only harmed performance.

Overall, the best model from this approach was softmax logistic regression with L1 regularization trained on raw data, yielding a Kaggle score of 0.529.

4.2 Results: Two-Stage Approach

Training binary XGBoost and SVM classifiers yielded 0.91 validation accuracy. The learned mixing weight α was 0.11, meaning XGBoost's predictions received approximately 90% weight in the ensemble.

Binary Classification (class2) - Ensemble Results:

Class	Precision	Recall	F1-Score	Support
0 (nonevent)	0.89	0.87	0.88	45
1 (event)	0.87	0.89	0.88	45
Accuracy			0.88	90
Macro Avg	0.88	0.88	0.88	90
Weighted Avg	0.88	0.88	0.88	90

For the multiclass stage, the standalone XGBoost classifier outperformed the XGBoost+SVM ensemble, with the former reaching 0.64 accuracy and the latter 0.63.

Multiclass Classification (class4) - XGBoost Only:

Class	Precision	Recall	F1-Score	Support
nonevent	0.83	0.89	0.86	45
Ia	0.00	0.00	0.00	5
Ib	0.40	0.38	0.39	16
II	0.50	0.50	0.50	24
Accuracy			0.64	90
Macro Avg	0.43	0.44	0.44	90
Weighted Avg	0.62	0.64	0.63	90

The model struggles significantly with the minority class Ia (0% precision/recall), likely due to its small support of only 5 samples. The nonevent class is predicted most reliably, while event subtypes (Ib, II) show moderate performance.

This approach achieved a Kaggle score of 0.739.

5. Conclusion

This project explored two approaches to classifying NPF events from atmospheric sensor data, revealing several insights about the problem structure and model selection.

Challenges Encountered:

- **Class imbalance:** The Ia class represents only 6% of training data, making it extremely difficult to predict with standard methods.
- **Feature correlation:** Many features are highly correlated, violating independence assumptions (Naive Bayes) and potentially causing instability in other models.
- **Feature correlation:** Many features are skewed, violating distributional assumptions.
- **Dataset size:** Possibly the results would be better with more data available for training.

Key Findings:

The two-stage approach achieved the best Kaggle score (0.739), demonstrating that problem decomposition can outperform direct multiclass classification when the class structure supports it. By separating binary event detection (91% accuracy) from event classification, we leveraged the strong separability of event vs. nonevent days while acknowledging the inherent difficulty of distinguishing between event subtypes.

Among simple supervised methods, L1-regularized softmax regression achieved the best cross-validation accuracy (0.733), but this did not translate to strong Kaggle performance (0.529).

Lessons Learned:

- Tree-based ensembles (XGBoost) provide robust performance with minimal tuning
- Simple, well-regularized models can match complex ensembles on cross-validation but may not generalize as well
- The two-stage probability assignment strategy proved critical for the combined scoring metric

Future Directions:

- Experiment with probability calibration techniques (Platt scaling, isotonic regression)
- Add hyperparameter tuning to the two-stage approach

Notes A large language model (Claude) was used to correct spelling, grammar and formatting.

Self-grading: We think this report is grade 5.

The report demonstrates solid understanding of the problem, suggesting original work. Multiple model families were evaluated, exploratory data analysis with sufficient visualisations was concluded. The successes and failures were reported honestly. The report is well-formatted and overall readable. It would be better if there was more information on hyperparameter tuning in the models.