## **Spoilage Stage Classification using XGBoost: Technical Overview**

### **1. Context and Classification Rationale**

The objective is to classify the spoilage stage of strawberries into three biological stages: Fresh, Spoiling, and Late Spoilage, based on real-time gas sensor data collected over time. These classes were defined after visual and temporal inspection of the Time\_to\_Spoilage\_Minutes column plotted against key indicators such as VOC resistance (BME\_VOC\_Ohm) and ethanol-related gas concentration (MQ3 PPM). We observed consistent patterns in VOC evolution as spoilage progressed.

The spoilage stages were defined as follows:

| Spoilage Stage | Time\_to\_Spoilage\_Minutes (mean) | Biological Rationale |
| --- | --- | --- |
| Fresh | ≥ 2400 minutes | VOC readings are relatively stable; fermentation hasn’t started. |
| Spoiling | 600–2399 minutes | Ethanol production begins; MQ3 increases, VOC fluctuates. |
| Late Spoilage | < 600 minutes | Fermentation peaks; high ethanol and VOC instability observed. |

This classification reflects microbial activity progression and aligns with the expected biological timeline of strawberry deterioration in a sealed chamber.

### 2. Data Input and Feature Engineering

Sensor data from two batches were used: - Batch 1 (training): Strawberries monitored from fresh to spoilage - Batch 2 (testing): Similar progression, but treated as unseen data

Each record contains: - BME\_VOC\_Ohm – Volatile Organic Compounds (VOC) - MQ3\_Top\_PPM, MQ3\_Bottom\_PPM – Ethanol / Alcohol concentrations - BME\_Temp, BME\_Humidity – Ambient conditions

To introduce time-aware dynamics, we segment the data using non-overlapping 3-minute windows, then extract:

* Statistical features: mean, standard deviation for each sensor
* Delta features: change from start to end of the window
* Gas Ratios: e.g., VOC / MQ3\_Top as an indicator of fermentation ratio

These engineered features become the input variables X, while the target label y is derived from the Time\_to\_Spoilage\_Minutes average in each window.

### 3. XGBoost Model Application

XGBoost is a gradient boosting algorithm known for its high performance on structured/tabular data.

Training Logic: - Initialization: Starts with a base probability for each class (e.g., uniform distribution). - Boosting rounds: - At each step, XGBoost trains a new tree to correct the residual errors of the previous ones. - Trees focus on hard-to-classify samples (e.g., borderline “Spoiling” cases). - Final prediction: Aggregates all tree outputs using a weighted sum and applies softmax to produce class probabilities.

Optimization Techniques: - Second-order gradient boosting (uses gradient + Hessian) - L1/L2 regularization to reduce overfitting - Shrinkage (learning rate) to improve stability - Column subsampling for generalization

### 4. Per-Tree Decision Logic

Each decision tree: - Splits based on thresholds in features like VOC\_to\_MQ3\_Top\_Ratio or delta\_VOC\_Ohm - Assigns a logit score at each leaf - The final prediction is the softmax of the summed scores across all trees

Example Scenario: If a sample shows: - High VOC resistance - Gradual ethanol increase (MQ3 Top and Bottom rising) - Stable delta VOC Then: - Trees likely classify it as Fresh, because these patterns indicate early-stage chemical stability.

### 5. Training Pipeline Summary

Raw sensor data ➔ 3-min windows ➔ Feature extraction ➔ Spoilage Stage labels  
↓  
Train (X, y) ➔ XGBoost ➔ Tree ensemble  
↓  
Learns thresholds: VOC ↓ + MQ3 ↑ rapidly → Spoiling or Late Spoilage

### 6. Outcomes and Evaluation

* The model trained on Batch 1 was tested on Batch 2 (held-out, unseen data).
* Despite class imbalance, the classifier learned to associate sensor evolution trends with spoilage progression.
* Evaluation included:
  + Accuracy
  + Precision/Recall/F1-score per class
  + Multiclass ROC Curve
  + Confusion Matrix