**Short Report: Machine Learning Pipeline for Vomitoxin Prediction**

1. **Introduction :**

This report outlines the **data preprocessing, model selection, training, and evaluation** for predicting **vomitoxin levels in corn** using spectral data. The workflow includes **data cleaning, feature engineering, model training, and API deployment**.

**2. Data Preprocessing :**

**2.1 Handling Missing Data**

* Checked for missing values and replaced empty values with NaN.
* Verified the dataset after imputation.

**2.2 Outlier Detection**

* Used **boxplots** and the **IQR method** to detect and handle extreme values.
* Applied **log transformation** to normalize skewed distributions.

**2.3 Feature Engineering**

* Normalized spectral data using **standardization (Z-score scaling)**.
* Selected relevant features using **correlation analysis**.

**3. Model Training & Evaluation :**

**3.1 Model Selection**

* Trained a **regression model ( XGBoost )** to predict vomitoxin levels.
* Used **train-test split (80-20)** for evaluation.

**3.2 Performance Metrics**

* **Mean Absolute Error (MAE):** 6.51
* **Mean Squared Error (MSE):** 378.18
* **R² Score:** 0.9999

These results indicate that the model performs **exceptionally well**.

**4. Deployment & API Integration :**

* Deployed the model using Flask to create an API for real-time predictions.
* Implemented a test script (test\_api.py) to send sample inputs and receive predictions.

**5. Key Findings & Future Improvements :**

**5.1 Findings**

* **Dimensionality reduction was not heavily needed**, as feature selection was already applied.
* The model generalizes well, with an **R² of 0.9999**, showing high accuracy.

**5.2 Future Improvements**

* **Hyperparameter tuning** (Grid Search, Optuna) to optimize performance.
* **Deploy the model to a cloud service** (AWS, GCP, or Render) for broader access.

**6. Conclusion :**

This machine learning pipeline successfully predicts **vomitoxin levels in corn** with **high accuracy**. The trained model is now **deployable** and can be used for real-time predictions via an API. Further improvements can enhance performance and interpretability.