Hyperspectral Vomitoxin Prediction: Model Evaluation Report

1. Preprocessing Steps

- **Standardization:** The spectral data was standardized using StandardScaler to ensure uniform feature scaling.
- **Dimensionality Reduction:** Principal Component Analysis (PCA) was applied to reduce dimensionality while preserving important variance.
- **Feature Selection:** Only relevant principal components were retained to improve model efficiency and reduce overfitting.
- **Train-Test Split:** The dataset was split into training and testing subsets to evaluate model performance effectively.

2. Insights from Dimensionality Reduction

- **PCA Transformation:** PCA reduced the high-dimensional spectral data to a smaller set of principal components, capturing the most significant variance.
- **Variance Retention:** The top principal components retained over 95% of the variance, enabling effective feature compression.
- **Improved Model Efficiency:** Using reduced features led to faster model training and inference times without significantly impacting prediction accuracy.

3. Model Selection, Training, and Evaluation

- **Convolutional Neural Network (CNN):** Used for deep feature extraction from hyperspectral images.
- **Deep Learning (ANN):** Implemented as a multi-layer perceptron for direct regression.
- **XGBoost:** Gradient boosting model for structured tabular data.
- **Random Forest:** Ensemble-based regression model with bootstrap aggregation.
- **GridSearchCV:** Used for optimization and hyper parameter tuning for the Random Forest and XGBoost.

Observations

Model	MAE	RMSE	R2
CNN	3001.23	6089.36	0.82
XgBoost	2209.8	4023.19	0.78
Random Forest	2106.86	5099.63	0.91

- Random Forest performed best, achieving the highest R^2 score (0.91) and the lowest error metrics compared to other models. So only Random Forest Implementation alone is used in code that's submitted via Git Hub.
- XGBoost was a strong competitor, but slightly underperformed compared to Random Forest.
- CNN and Deep Learning models performed moderately well but required more computational resources and training time.

4. Key Findings and Suggestions for Improvement

Key Findings

- Dimensionality reduction via PCA significantly improved model training efficiency.
- Random Forest outperformed other models, achieving the best prediction accuracy.
- Deep learning models, while effective, required extensive tuning for optimal performance.

Suggestions for Improvement

- Hyperparameter Optimization: Fine-tuning hyperparameters for Random Forest and XGBoost could further enhance performance.
- Feature Engineering: Experimenting with different feature extraction techniques before PCA.
- Hybrid Models: Combining deep learning with traditional ensemble models for improved prediction accuracy.
- Additional Data Augmentation: Expanding the dataset with synthetic variations to improve model generalization.