# **Comparative Study of Supervised Learning Algorithms on Gas Sensor Array Dataset**

Report B: Moli Agrawal, 5th Sem CSE

## **1. Introduction**

This report presents an enhanced comparative analysis of multiple supervised learning algorithms applied to the UCI Gas Sensor Array Drift Dataset. The dataset captures the complex challenge of sensor drift over time---an issue that significantly impacts long-term gas classification accuracy. The goal is to benchmark and evaluate various machine learning models representing linear, ensemble, probabilistic, and neural paradigms under consistent preprocessing and evaluation conditions. This work aims to identify the most effective and generalizable approach for gas classification in drift-prone environments.

## **2. Dataset Overview**

### **2.1 Dataset Description**

The UCI Gas Sensor Array Drift Dataset was collected using 16 metal-oxide gas sensors exposed to six gases under different concentrations. The dataset reflects sensor drift across multiple months, introducing gradual changes in sensor responses. This makes it ideal for assessing model adaptability. The dataset was preprocessed to remove outliers, handle missing values, and standardize feature distributions.

### **2.2 Dataset Statistics**

| **Characteristic** | **Value** |
| --- | --- |
| Total Samples | 13,910 |
| Original Features | 128 |
| Engineered Features | 11 (statistical descriptors) |
| Total Features | 139 |
| Number of Classes | 6 |
| Training Samples | 11,128 (80%) |
| Testing Samples | 2,782 (20%) |

## **3. Methodology and Algorithms**

The experiment followed a rigorous methodology, ensuring fair comparison and reproducibility across all models. Each algorithm underwent identical preprocessing and evaluation protocols.

### **3.1 Data Preprocessing**

Data cleaning involved Z-score-based outlier detection (|Z| > 3.0) and linear interpolation for missing values. StandardScaler was applied to normalize all features, critical for algorithms like SVM and KNN. The data was then split into 80% training and 20% testing subsets using stratified sampling to preserve class balance across six gas categories.

### **3.2 Feature Engineering**

In addition to the 128 raw sensor readings, 11 engineered features were derived to enhance discriminative power. These included mean, standard deviation, variance, skewness, kurtosis, minimum, maximum, and signal range. Feature importance was later assessed using the Gini importance measure from the Random Forest model, revealing that certain sensors contributed disproportionately to the overall classification accuracy.

### **3.3 Model Training and Evaluation**

All models were trained using 5-fold cross-validation for robustness. Evaluation metrics included Accuracy, Precision, Recall, F1-Score, and ROC-AUC. Non-probabilistic classifiers such as Linear SVM and Ridge Classifier do not provide probability estimates, which is why ROC-AUC was not computed for them. Training times were recorded to assess computational efficiency. Neural networks and SVMs required longer training durations compared to tree-based models, while Naive Bayes and Logistic Regression completed training within seconds.

### **3.4 Algorithms and Hyperparameters**

The following algorithms and configurations were tested:

1. Logistic Regression -- solver='lbfgs', max\_iter=500
2. Linear SVM -- C=1.0, kernel='linear'
3. K-Nearest Neighbors -- n\_neighbors=5, metric='minkowski'
4. Decision Tree -- criterion='gini', max\_depth=None
5. Random Forest -- n\_estimators=200, max\_depth=None, random\_state=42
6. Ridge Classifier -- alpha=1.0
7. Gaussian Naive Bayes -- default priors
8. Bernoulli Naive Bayes -- binarize=0.0
9. Multi-Layer Perceptron -- hidden\_layers=(100,100,50), activation='relu', solver='adam', max\_iter=300

## **4. Results and Performance Analysis**

### **4.1 Overall Model Performance**

The comparative performance of all algorithms is summarized below. The Neural Network achieved the highest accuracy and ROC-AUC, while Naive Bayes models performed poorly due to strong feature correlations that violate the independence assumption.

| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-Score** | **ROC-AUC** | **CV Mean** | **CV Std** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Neural Network | 0.9957 | 0.9957 | 0.9957 | 0.9957 | 0.9993 | 0.9936 | 0.0018 |
| Random Forest | 0.9946 | 0.9946 | 0.9946 | 0.9946 | 0.9991 | 0.9927 | 0.0016 |
| Linear SVM | 0.9932 | 0.9932 | 0.9932 | 0.9932 | --- | 0.9922 | 0.0013 |
| Logistic Regression | 0.9917 | 0.9918 | 0.9917 | 0.9917 | 0.9984 | 0.9899 | 0.0007 |
| KNN (k=5) | 0.9899 | 0.9900 | 0.9899 | 0.9899 | 0.9979 | 0.9865 | 0.0029 |
| Ridge Classifier | 0.9720 | 0.9727 | 0.9720 | 0.9719 | --- | 0.9737 | 0.0044 |
| Decision Tree | 0.9709 | 0.9709 | 0.9709 | 0.9709 | 0.9815 | 0.9718 | 0.0032 |
| Gaussian NB | 0.5859 | 0.7028 | 0.5859 | 0.6107 | 0.8463 | 0.5829 | 0.0123 |
| Bernoulli NB | 0.5421 | 0.6068 | 0.5421 | 0.5345 | 0.8421 | 0.5430 | 0.0056 |

### **4.2 Training Time Comparison**

Training time comparison revealed that Naive Bayes and Decision Tree models completed in under 2 seconds, Random Forest required approximately 10 seconds, while SVM and Neural Network exceeded 40 seconds due to iterative optimization. Despite longer training times, their superior performance justifies computational cost.

### **4.3 Cross-Validation Stability**

The cross-validation standard deviation values indicate the stability and reliability of each model across different data folds. Neural Network and Random Forest exhibited very low standard deviations (0.0018 and 0.0016 respectively), demonstrating consistent performance across all folds. Linear SVM also showed remarkable stability with a standard deviation of 0.0013. In contrast, Ridge Classifier had the highest variability among top performers (0.0044), though still maintaining acceptable stability.

### **4.4 ROC-AUC Performance Analysis**

Models capable of producing probability estimates demonstrated excellent ROC-AUC scores. The Neural Network achieved the highest ROC-AUC of 0.9993, followed closely by Random Forest at 0.9991. These near-perfect scores indicate exceptional discriminative capability across all six gas classes. Logistic Regression and KNN also performed admirably with ROC-AUC scores of 0.9984 and 0.9979 respectively. Even the poorly performing Naive Bayes models showed ROC-AUC scores above 0.84, suggesting that while their classification accuracy was low, they maintained some discriminative power in probability space.

## **5. Discussion**

### **5.1 Model Performance Interpretation**

The experiment demonstrates that advanced models like Random Forest and Neural Networks effectively capture complex nonlinear dependencies within the sensor data. Naive Bayes models underperformed because they assume feature independence, which is unrealistic for correlated sensor readings. Ensemble techniques such as Random Forest excelled by averaging multiple decision paths, mitigating overfitting while retaining interpretability.

### **5.2 Sensor Drift Challenges**

Sensor drift posed a unique challenge---models that rely on static decision boundaries (e.g., Logistic Regression, SVM) still performed well, but their long-term robustness remains uncertain. Neural models are better equipped to adapt to evolving data distributions if retrained periodically. Additionally, class distribution was found to be approximately balanced, ensuring that performance metrics were unbiased.

### **5.3 Computational Efficiency Trade-offs**

The trade-off between computational efficiency and model performance was evident throughout the analysis. Naive Bayes and Decision Tree models offered rapid training and inference, making them suitable for resource-constrained environments or applications requiring real-time predictions. However, their accuracy limitations make them less viable for high-stakes applications. Random Forest provided an optimal balance, achieving near-optimal accuracy while maintaining reasonable training times. Neural Networks, despite requiring the most computational resources, delivered the best overall performance, justifying their use in applications where accuracy is paramount.

### **5.4 Ensemble Methods Advantage**

Ensemble methods demonstrated clear advantages in this classification task. Random Forest, by aggregating predictions from 200 decision trees, achieved superior generalization compared to a single Decision Tree. The ensemble approach reduced variance and improved robustness to noisy sensor readings. This highlights the importance of ensemble techniques in real-world sensor applications where individual sensors may produce unreliable measurements.

### **5.5 Feature Correlation Impact**

The strong performance of models capable of handling correlated features (Neural Networks, Random Forest, SVM) versus the poor performance of Naive Bayes models underscores the importance of understanding feature relationships. Sensor arrays naturally produce highly correlated readings as multiple sensors respond to the same environmental conditions. This correlation structure provides valuable information that sophisticated models can exploit, while simpler probabilistic models that assume independence fail to leverage these relationships.

## **6. Conclusion and Future Work**

### **6.1 Summary of Findings**

This enhanced comparative study confirms that Neural Networks and Random Forests are the most effective classifiers for gas sensor data under drift conditions, achieving near-perfect accuracy and generalization. Simpler models like Logistic Regression and Linear SVM also deliver strong results with lower computational demands, making them suitable for embedded or real-time applications.

### **6.2 Practical Implications**

The findings have significant implications for industrial gas monitoring systems. Organizations must balance accuracy requirements against computational constraints when selecting appropriate models. For laboratory settings where computational resources are abundant, Neural Networks offer the best performance. For embedded IoT devices with limited processing power, Logistic Regression or Linear SVM provide excellent accuracy with minimal resource consumption.

### **6.3 Future Research Directions**

Future work will focus on integrating domain adaptation and incremental learning to address real-world drift. Techniques such as transfer learning, adaptive normalization, and RNN-based temporal modeling could help maintain accuracy without frequent retraining. A key direction is deploying lightweight ensemble models in IoT-enabled gas monitoring systems for practical scalability.

### **6.4 Long-term Deployment Considerations**

Long-term deployment of gas sensor classification systems requires addressing several challenges. Periodic model retraining or online learning approaches will be necessary to adapt to continuous sensor drift. Developing automated drift detection mechanisms could trigger retraining only when necessary, optimizing computational resources. Additionally, investigating hybrid approaches that combine the interpretability of tree-based models with the accuracy of neural networks represents a promising research avenue.

## **7. References**

1. Vergara, A., et al. (2012). UCI Gas Sensor Array Drift Dataset. UCI Machine Learning Repository.
2. Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825--2830.
3. Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.
4. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
5. Brownlee, J. (2020). Machine Learning Mastery with Python. Machine Learning Mastery Publications.