Image Noise Distribution Matching

Machine Learning Project Report

Ionescu Victor Marian

1 Introduction

The objective of this project is to design and evaluate a machine learning pipeline capable of determining whether two given image noise samples originate from the same underlying probability distribution. This task is highly relevant in image forensics and sensor identification, where noise characteristics act as a fingerprint of the acquisition device.

Unlike traditional image similarity tasks, noise samples lack semantic structure and exhibit stochastic behavior. Therefore, the proposed approach focuses on modeling distribution-level properties rather than spatial patterns.

2 Data Preprocessing and Feature Engineering

2.1 Robust Scaling and Outlier Handling

Noise samples frequently contain extreme values. To mitigate their influence, features were scaled using the *RobustScaler*, which relies on the interquartile range rather than the mean and variance. This choice ensures stability under heavy-tailed noise distributions.

2.2 Distribution Profile Extraction

Each noise image is transformed into a fixed-length statistical descriptor capturing its underlying distribution. The extracted features include:

- Core statistics: mean, variance, median, standard deviation, median absolute deviation
- Quantile-based features (1% to 99%)
- Tail behavior descriptors
- Higher-order moments: skewness and kurtosis
- Multi-scale histogram entropy and dispersion (8, 16, 32, 64 bins)
- Frequency-domain statistics via FFT magnitude
- Local quadrant-based statistics
- Edge magnitude statistics

Each image is represented by a vector of statistical features capturing both global and local distribution characteristics.

2.3 Feature Representation and Dimensionality

For each pair of noise samples, a comparison vector is constructed using:

- Raw feature differences
- Absolute and normalized differences
- Log-ratio features
- Distance metrics: Euclidean, Manhattan, Chebyshev
- Similarity metrics: cosine similarity and Pearson correlation

This results in a final feature vector of **245 dimensions per noise pair**. The processed dataset contains **6078 samples**.

3 Machine Learning Models

3.1 Base Models

The following classifiers were evaluated:

- Random Forest
- Gradient Boosting
- XGBoost
- LightGBM

Tree-based models were selected due to their robustness to feature scaling and ability to capture non-linear relationships between statistical descriptors.

3.2 Stacked Ensemble Architecture

To leverage model diversity, a stacked ensemble was implemented. Base classifiers generate probabilistic predictions on the validation set, which are used as meta-features for a Logistic Regression meta-learner.

This approach improves generalization by combining complementary decision boundaries learned by individual models.

4 Hyperparameter Optimization

Hyperparameters were tuned using iterative manual search guided by validation performance. Table 1 summarizes the selected configurations.

Model	Hyperparameter	Selected Value
XGBoost	Learning rate	0.05
XGBoost	Max depth	6
LightGBM	Num leaves	25
Random Forest	Min samples leaf	5
Ensemble	Decision threshold	0.45

Table 1: Selected hyperparameters after manual tuning

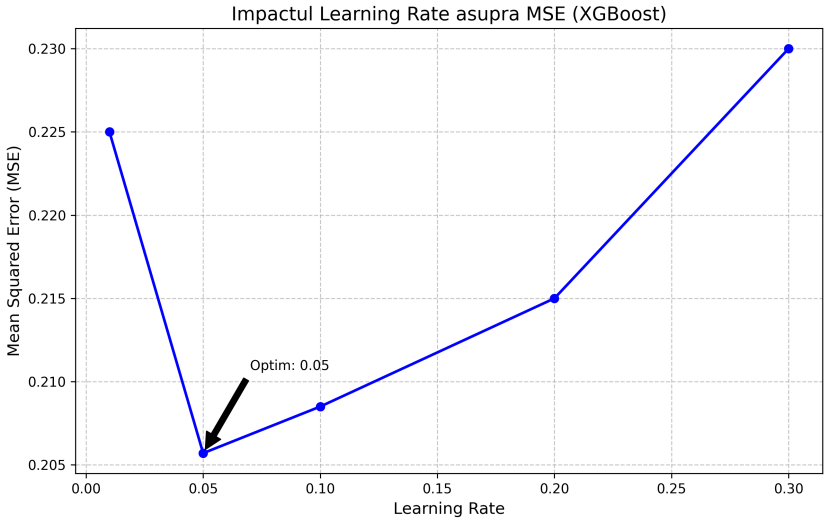

Figure 1: Impact of the learning rate on the validation MSE for the XGBoost model. The optimal value was observed at a learning rate of 0.05.

Figure 1 illustrates the influence of the learning rate on the validation Mean Squared Error for the XGBoost classifier. Very small learning rates lead to underfitting, while larger values increase the optimization noise and degrade performance.

A learning rate of 0.05 achieved the lowest validation error and was therefore selected for the final model configuration.

5 Performance Evaluation

Models were evaluated on the validation set using MAE, MSE, Spearman, and Kendall correlation metrics.

Model	MAE	MSE	Spearman	Kendall
XGBoost	0.4160	0.2057	0.3997	0.3265
LightGBM	0.4213	0.2077	0.3855	0.3148
Gradient Boosting	0.4321	0.2126	0.3616	0.2953
Random Forest	0.4516	0.2158	0.3854	0.3147
Stacked Ensemble	**0.4111**	**0.2048**	**0.4047**	**0.3306**

Table 2: Validation performance across models

5.1 Metric Interpretation

MAE and MSE evaluate numerical prediction accuracy, while Spearman and Kendall correlations assess ranking consistency. The stacked ensemble outperforms all base models across all metrics, indicating improved robustness and ranking reliability.

6 Unsuccessful Approaches

Several approaches were tested and discarded:

- **Raw pixel K-NN**: Failed due to lack of spatial correspondence in noise.

- **Convolutional Neural Networks**: Overfitted rapidly due to limited data.
- **Raw subtraction**: Highly sensitive to mean shifts.

These experiments confirmed the importance of distribution-level representations.

7 Conclusion

This project demonstrates that noise distribution matching can be effectively addressed using statistical feature engineering combined with ensemble learning. The stacked model achieved the best overall performance, particularly in rank-based metrics critical for forensic matching tasks.

Future work may explore learned representations or self-supervised feature extraction techniques.