Classical Machine Learning Baselines for Deepfake Audio Detection on the Fake-or-Real Dataset

Faheem Ahmad, Ajan Ahmed, Masudul Imtiaz
Clarkson University, Potsdam, New York, USA
Email: fahmad@clarkson.edu, aahmed@clarkson.edu, mimtiaz@clarkson.edu

***Abstract*—Deep learning has made it easy to synthesize highly realistic human voices, enabling so–called "deepfake audio" that can be exploited for fraud, impersonation, and disinformation. Despite rapid progress on neural detectors, there remains a need for transparent baselines that reveal which acoustic cues most reliably separate real from synthetic speech. This chapter presents an interpretable classical machine learning baseline for deepfake audio detection using the Fake-or-Real (FoR) dataset. We extract a rich set of prosodic, voice-quality, and spectral features from two-second speech clips at two sampling rates: a high-fidelity 44.1 kHz condition and a 16 kHz re-recorded condition that mimics telephone-quality audio. We perform statistical analysis (ANOVA, missing-value inspection, correlation heatmaps) to understand which features differ significantly between real and fake speech, and we then train multiple models—Logistic Regression, Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Gaussian Naïve Bayes, Support Vector Machines (SVMs), and Gaussian Mixture Models (GMMs). Performance is reported using accuracy, ROC–AUC, equal error rate (EER), and Detection Error Trade-off (DET) curves. Pairwise McNemar's tests are used to confirm that differences between models are statistically significant. The best model, an SVM with an RBF kernel, achieves approximately 93% test accuracy and an EER around 7% on both sampling rates, while simpler linear models reach around 75% accuracy. Analysis of feature importances and distributions shows that pitch variability and spectral richness (e.g., spectral centroid and bandwidth) are key cues distinguishing real from synthetic speech. These results provide a strong, interpretable baseline for future deepfake audio detectors and highlight which aspects of speech synthesis still diverge from human speech.**

I. Introduction

Recent advances in neural text-to-speech (TTS) and voice conversion have made it possible to generate highly convincing human-like speech from text or from short recordings of a target speaker. While these technologies have clear positive applications, they also enable malicious use cases such as impersonating individuals in phone calls, generating fake audio evidence, or automating social engineering attacks. Such synthetic or "deepfake" audio presents an emerging security and trust challenge.

A growing body of work addresses deepfake audio detection using a range of features and models [1], [3], [4]. Many high-performing approaches rely on deep neural networks trained on spectrograms, raw waveforms, or embeddings from speaker recognition systems. Although powerful, deep models can be computationally expensive and often function as black boxes, making it difficult to understand exactly which acoustic cues they exploit.

In contrast, classical machine learning models built on hand-engineered features provide two important benefits. First, they are computationally efficient and can potentially run on low-power devices (for example, in telephony gateways or browser plug-ins). Second, they are relatively interpretable: by examining which features are most discriminative, we can gain insight into how synthetic speech differs from genuine human speech and how TTS systems might evolve to evade detection.

The goal of this chapter is to build a detailed, interpretable baseline for deepfake audio detection on the *Fake-or-Real* (FoR) dataset [2]. Rather than focusing on deep learning, we emphasize:

- Carefully designed acoustic features that capture prosody, voice quality, and spectral structure.
- Statistical analysis (ANOVA, missing-value patterns, correlation heatmaps) to select and justify features.
- A suite of classical models (Logistic Regression, LDA, QDA, Naïve Bayes, SVMs, and GMMs).
- Robust evaluation including accuracy, ROC–AUC, EER, DET curves, and McNemar's tests.

We also explore two sampling conditions: (i) high-quality audio at 44.1 kHz for two-second clips, and (ii) the 16 kHz *re-recorded* condition where clips are played back and recorded in a physical room, mimicking realistic channel effects. This dual setup allows us to ask whether deepfake detection remains effective when audio is transmitted over a noisy channel.

The remainder of this chapter is organized as follows. Section II describes the FoR dataset and the two experimental conditions. Section III explains the acoustic features we extract and the statistical tests used for feature selection. Section IV outlines the modeling pipeline and evaluation metrics. Section V presents the empirical results, including model performance, DET curves, and McNemar's tests. Section VI discusses the implications of our findings, and Section VII concludes with suggestions for future work.

II. Fake-or-Real Dataset

A. Overview

The Fake-or-Real (FoR) dataset [2] is a large public corpus designed specifically for synthetic speech detection. It contains short utterances of both real human speech and computer-generated speech from multiple TTS engines. The real speech originates from several publicly available corpora (such as

CMU Arctic, LJSpeech, and VoxForge), while the fake speech is produced by modern TTS systems including neural models like WaveNet. This diversity makes FoR a useful benchmark for studying generalizable deepfake detection.

FoR is released in several variants:

- **for-original:** Original collected clips with varying sampling rates and durations.
- **for-norm:** A normalized set with consistent sampling rate, volume, and channel configuration.
- **for-2sec:** Clips truncated or segmented to exactly two seconds, based on the normalized set.
- **for-rerec:** Re-recorded versions of the two-second clips, captured by playing them back in a room and recording with a microphone, simulating telephone-like channel effects.

In this work, we focus on the **for-2sec** and **for-rerec** variants. Each variant supplies predefined training, validation, and test sets with balanced numbers of real and fake utterances. In our experiments we use all training and test partitions, totaling approximately 31,138 clips, with roughly equal numbers of real and fake samples in each condition.

B. Sampling Rate Conditions

We evaluate two sampling-rate settings:

a) 44.1 kHz (high fidelity).: The *for-2sec* clips are high-quality audio sampled at 44.1 kHz. Each clip contains a two-second segment of either real or synthetic speech. This condition provides detailed spectral information up to 22 kHz, including subtle high-frequency cues such as sibilants, breath noise, and microphone artifacts.

b) 16 kHz (re-recorded).: The *for-rerec* clips are the same underlying utterances replayed through speakers in a room and recorded by a microphone. The effective bandwidth is narrower, roughly similar to telephone or conferencing audio. We downsample these clips to 16 kHz for analysis. This condition tests whether detection remains robust when audio is transmitted over a realistic channel with environmental noise and frequency limitations.

For each condition we use the dataset's predefined train/test split, which ensures that speakers do not overlap between sets. This is crucial to avoid overestimating performance by memorizing speaker-specific characteristics.

C. Class Balance and Data Balancing Decisions

The FoR dataset is carefully constructed to be approximately class-balanced: the number of real and fake clips is similar in each partition. As a result, we *do not* perform any additional data balancing such as oversampling, undersampling, or synthetic minority over-sampling (SMOTE).

There are three reasons for this decision:

1) **No severe imbalance.** With roughly equal numbers of real and fake samples, standard classification metrics like accuracy and ROC–AUC are meaningful without balancing.
2) **Avoiding synthetic artifacts.** Methods like SMOTE generate synthetic feature vectors in feature space. For high-dimensional acoustic statistics, these artificial points may not correspond to physically plausible audio and can encourage models to learn artifacts of the oversampling process rather than genuine class differences [5], [6].
3) **Preserving real distribution.** For detection problems, we often care about performance under the natural distribution of traffic. Distorting that distribution through aggressive resampling could lead to misleading estimates of false alarm or miss rates in deployment.

Because the classes are well balanced, we simply train all models on the original data and, where applicable, use class weights to handle any minor imbalance.

```
===== Checking class balance for 16k_fullclip =====
Unique speakers in TRAIN: 24447
Unique speakers in TEST : 6112
Class counts in TRAIN (0=real, 1=fake): [12404 12509]
Class counts in TEST  (0=real, 1=fake): [3144 3081]
Train % fake: 50.21%
Test  % fake: 49.49%
```

Fig. 1: Console output showing the class-balance and speaker-balance check for the 16 kHz configuration. After the group-aware train–test split, the training set contains approximately 24,447 unique speakers and the test set 6,112 speakers, with nearly equal numbers of real and fake clips in both splits (about 50% fake in train and 49% fake in test). This confirms that the split is both speaker-disjoint and approximately class-balanced.

```
===== Checking class balance for 44k_25ms =====
Unique speakers in TRAIN: 24447
Unique speakers in TEST : 6112
Class counts in TRAIN (0=real, 1=fake): [12404 12509]
Class counts in TEST  (0=real, 1=fake): [3144 3081]
Train % fake: 50.21%
Test  % fake: 49.49%
```

Fig. 2: Class-balance and speaker-balance check for the 44.1 kHz two-second FoR clips. The group-aware split yields the same number of unique speakers in train and test as in the 16 kHz setting, and again the class distribution remains almost perfectly balanced between real and fake clips. Because of this, no additional oversampling or undersampling was applied.

III. Feature Extraction and Statistical Analysis

We extract a comprehensive set of acoustic features designed to capture differences in prosody, voice quality, and spectral structure between real and synthetic speech. For each two-second clip we compute frame-level measurements and then summarize them with statistics such as mean, standard deviation, range, percentiles, and coefficients of variation.

A. Prosodic Features

Prosodic features describe the perceived melody and rhythm of speech. We compute the fundamental frequency (F_0) using

the YIN algorithm and derive the following statistics over voiced frames:

- $f0\_mean\_v$: Mean F_0 in Hz during voiced segments.
- $f0\_std\_v$: Standard deviation of F_0, indicating pitch variability.
- $f0\_range\_v$: Range (max–min) of F_0.
- $f0\_iqr\_v$: Interquartile range (IQR) of F_0.
- $f0\_cv\_v$: Coefficient of variation ($f0\_std\_v/f0\_mean\_v$).
- $f0\_p10\_v$**,** $f0\_p90\_v$: 10th and 90th percentile of F_0.

We also extract timing-related prosodic features based on voiced/unvoiced segmentation:

- dur_s: Duration of the trimmed clip in seconds.
- **voice_pct**: Proportion of time that is voiced.
- $n\_voiced\_seg\_per\_s$: Number of distinct voiced segments per second.
- **mean_voiced_seg_ms**: Average voiced segment duration in ms.
- **pause_ratio**: Ratio of unvoiced (pause) duration to voiced duration.
- $f0\_slope\_hz\_per\_s$: Slope of a linear regression fit to the F_0 contour, capturing overall rising or falling intonation.

These features are motivated by the observation that human speech often exhibits rich, context-dependent prosody, whereas synthetic speech may have flatter intonation or overly regular timing. For example, many TTS systems historically struggled to mimic natural pitch variability, resulting in more monotone voices.

B. Voice-Quality Features

Voice quality describes fine-grained characteristics of vocal fold vibration and the resulting glottal source. We include:

- **jitter_local**: Relative cycle-to-cycle variation in pitch period.
- **shimmer_local**: Relative cycle-to-cycle variation in amplitude.

In natural speech, jitter and shimmer arise from small irregularities in vocal fold vibration. Synthetic speech generated from vocoders can be overly smooth, with very low jitter and shimmer. Thus, these measures can indicate how "perfect" or "natural" the glottal source appears.

We also attempted to compute a harmonics-to-noise ratio (HNR), but due to extraction issues the resulting feature (`hnr_mean`) was missing for all samples and was therefore discarded.

C. Energy and Spectral Features

We compute the short-time magnitude spectrum using the STFT, with window and hop lengths appropriate to each sampling rate (25 ms window, 10 ms hop at 44.1 kHz; 1024-sample window, 10 ms hop at 16 kHz). At 44.1 kHz, we compute short-time spectral features using 25 ms windows with 10 ms hop over each two-second utterance, and aggregate them via mean and variance to obtain one feature vector per clip. From this we derive several spectral descriptors:

a) RMS Energy.: Frame-level RMS energy is summarized by:

- **rms_mean**, **rms_std**, **rms_range**, **rms_iqr**, **rms_cv**.

These capture the loudness and dynamics of the signal. Synthetic speech may be more uniformly normalized in amplitude, whereas real recordings can have greater variation.

b) Spectral Centroid.: The spectral centroid is the "center of mass" of the magnitude spectrum and correlates with perceived brightness. We compute:

- **spec_centroid_mean**, **spec_centroid_std**, **spec_centroid_rng**.

Real speech often contains more high-frequency content due to sibilants, plosives, and microphone noise, which can raise the centroid compared to synthetic voices.

c) Spectral Bandwidth.: Spectral bandwidth measures the spread of energy around the centroid. We use:

- **spec_bandwidth_mean**, **spec_bandwidth_std**, **spec_bandwidth_rng**.

Broadband noise and a wide range of formant structures in natural speech can result in larger bandwidths.

d) Spectral Contrast and Rolloff.: Finally, we compute:

- **spec_contrast_mean**, **spec_contrast_std**: Difference between spectral peaks and valleys across frequency bands.
- **spec_rolloff_mean**, **spec_rolloff_std**, **spec_rolloff_rng**: Frequency below which a fixed percentage (85%) of spectral energy resides.

These features help capture how sharply energy concentrates around formants and how far into the high frequencies the energy extends.

D. Handling Missing Values and Feature Selection

After feature extraction, we conducted a systematic inspection of missing values and feature relevance.

1) Missing Values: We computed, for each feature, the number and percentage of missing entries. The overall missing rate was low (around 2.8%), almost entirely due to the `hnr_mean` feature which failed to compute for every file. All other features had 0% missing values. Consequently, we dropped `hnr_mean` from further analysis and modeling rather than attempting to impute a constant or meaningless value.

2) ANOVA Feature Significance: To test whether each feature's mean differed significantly between real and fake speech, we applied a one-way ANOVA on the training set. For the 16 kHz configuration, the ten most significant features (smallest p-values) included:

- `f0_std_v`, `f0_cv_v`, `f0_range_v`, `f0_iqr_v`,
- `rms_mean`,
- `spec_centroid_mean`, `spec_bandwidth_mean`,
- `spec_rolloff_mean`,

all with p-values effectively zero. These features strongly separate the two classes and align with the intuition that pitch variability and spectral richness are key indicators of natural speech.

By contrast, some features such as `shimmer_local`, `spec_bandwidth_rng`, and `spec_rolloff_rng` had

much larger p-values. In particular, `shimmer_local` showed no significant difference between real and fake (e.g., $p \approx 0.91$ in one configuration), suggesting that amplitude cycle irregularity is not a strong discriminator in this dataset. We therefore elected to drop features with $p \geq 0.05$ from the final models.

3) Final Feature Sets: After removing non-significant and fully-missing features, we obtained:

- **44.1 kHz (for-2sec):** 29 features used for modeling. Dropped features include `hnr_mean`, `spec_bandwidth_rng`, and `spec_rolloff_rng`.
- **16 kHz (for-rerec):** 30 features used. Dropped features include `hnr_mean` and `shimmer_local`.

This selection strikes a balance between capturing essential prosodic and spectral differences and avoiding redundant or uninformative features.

E. Feature Visualizations

To build intuition, we visualized feature distributions and correlations separately for real and fake speech.

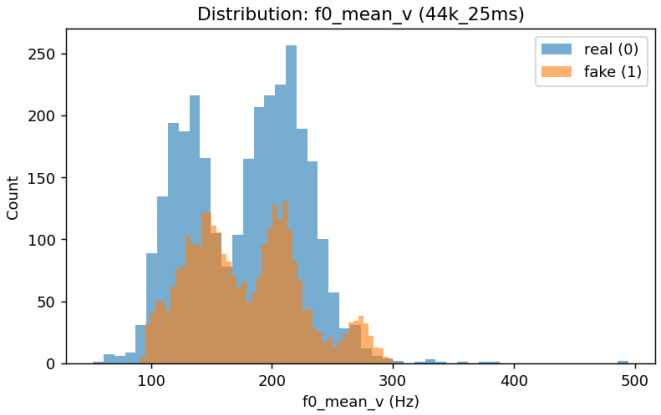

Fig. 3: Distribution of mean fundamental frequency ($f0\_mean\_v$) for real (blue) and fake (orange) speech at 44.1 kHz. Real speech exhibits two broad peaks corresponding roughly to male and female pitch ranges; fake speech covers a similar range but with a slightly different distribution.

Analogous plots for the 16 kHz condition show similar qualitative patterns, illustrating that key discriminative cues survive the re-recording process.

We also computed correlation heatmaps for the final feature sets, separately for real and fake speech, to understand how features interact. A representative example for the 16 kHz configuration is shown in Figure 8.

IV. Modelling Pipeline and Evaluation

A. Data Splitting and Preprocessing

For each sampling-rate condition, we use the dataset's predefined training and test sets. Speaker identities do not overlap between splits, preventing speaker leakage. The training set contains approximately 24,913 clips and the test set 6,225 clips.

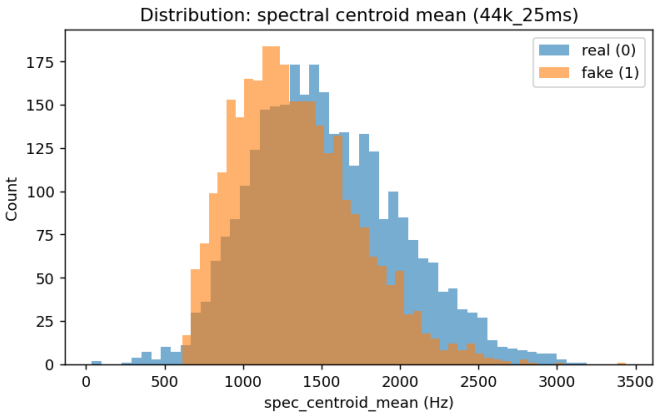

Fig. 4: Distribution of mean spectral centroid for real and fake speech at 44.1 kHz. Real speech tends to have a higher centroid, indicating more high-frequency energy (e.g., sibilants and microphone noise) than the synthetic speech.

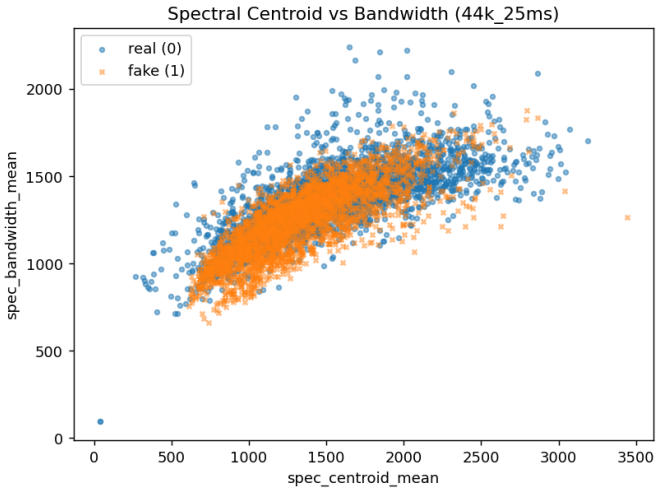

Fig. 5: Scatter plot of mean spectral centroid versus mean spectral bandwidth at 44.1 kHz. Real clips (blue) and fake clips (orange) overlap, but many real clips occupy the high-centroid, high-bandwidth region that fake clips rarely reach.

We apply the following preprocessing steps:

1) **Feature cleaning:** Replace any occurrences of $\pm\infty$ with `NaN` and drop features that are entirely missing (e.g., `hnr_mean`).
2) **Imputation:** For remaining missing values (rare in practice), we use median imputation based on the training set.
3) **Scaling:** All features are standardized to zero mean and unit variance using statistics computed on the training set. The same transformation is applied to the test set.

B. Classical Machine Learning Models

We evaluate the following models:

- **Logistic Regression (LR):** A linear model with L2 regularization and class weights to handle minor imbalance.

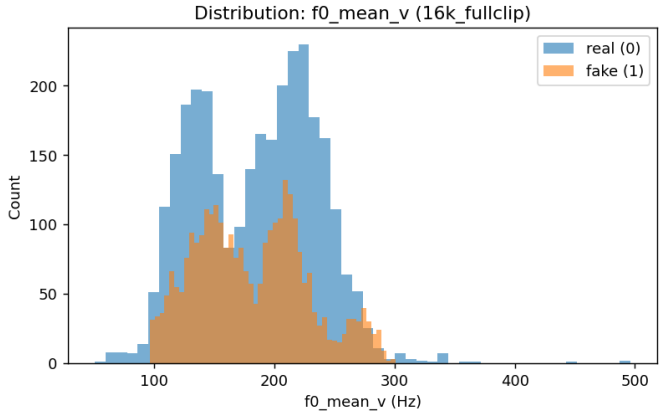

Fig. 6: Distribution of $f0\_mean\_v$ for real and fake speech at 16 kHz. The overall structure resembles the 44.1 kHz case, indicating that downsampling and re-recording do not remove pitch-based cues.

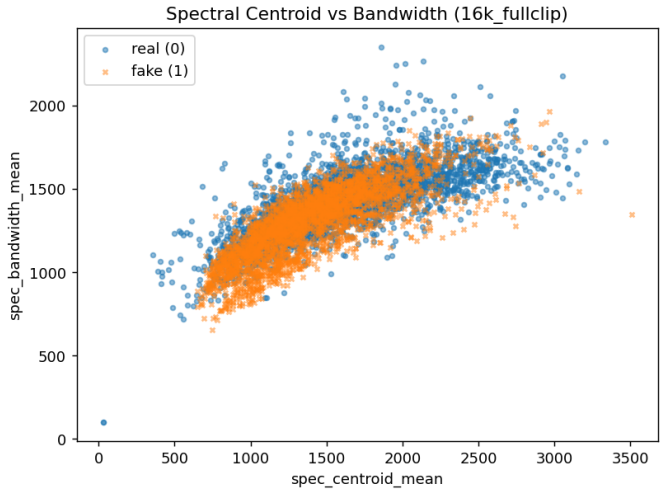

Fig. 7: Spectral centroid versus bandwidth at 16 kHz. Real speech still tends to occupy regions with higher centroid and bandwidth, reflecting broader spectral content even after re-recording.

- **Linear Discriminant Analysis (LDA):** Assumes each class is Gaussian with shared covariance, yielding a linear decision boundary.
- **Quadratic Discriminant Analysis (QDA):** Similar to LDA but with class-specific covariance matrices, allowing quadratic boundaries.
- **Gaussian Naïve Bayes (GNB):** Models each feature as an independent Gaussian given the class. This is simple but often underfits when features are correlated.
- **Support Vector Machines (SVM):** We use a linear SVM and an SVM with a radial basis function (RBF) kernel. The RBF SVM can capture complex non-linear decision surfaces.
- **Gaussian Mixture Model (GMM) classifier:** We train separate GMMs for the real and fake classes and classify by comparing the log-likelihoods of each sample under the two models.

C. Hyperparameter Tuning

Most models have few hyperparameters and rely on maximum-likelihood estimates. For the RBF SVM we perform a grid search over the penalty parameter C and kernel bandwidth γ, using stratified 3-fold cross-validation on the training set. ROC–AUC is used as the scoring metric. In both sampling conditions the best configuration was $C = 10$ and $\gamma = \texttt{scale}$.

The GMM uses a fixed number of mixture components per class (chosen by examining validation likelihood and avoiding overfitting). Linear models use default regularization parameters unless otherwise noted.

D. Evaluation Metrics

We evaluate both training and test performance using:

- **Accuracy:** Fraction of correctly classified clips at the chosen decision threshold.
- **ROC–AUC:** Area under the receiver operating characteristic curve, summarizing ranking quality independent of threshold.
- **Equal Error Rate (EER):** Error rate at the threshold where false acceptance rate (FAR) equals false rejection rate (FRR). This is common in biometric and spoofing evaluations.
- **DET curves:** Plots of FAR versus FRR across thresholds, highlighting the trade-off between security (low FAR) and usability (low FRR).

To assess whether differences between models are statistically significant, we also compute pairwise McNemar's tests on both training and test predictions.

V. Results

A. Model Performance at 44.1 kHz

Table I summarizes the performance of all models on the 44.1 kHz (for-2sec) condition. We report training and test accuracy, ROC–AUC, and test EER.

The linear models (Logistic Regression, LDA, Linear SVM) cluster around 75% accuracy and ROC–AUC ≈ 0.82. QDA and GMM improve this to $\approx$ 77–78% accuracy and ROC–AUC ≈ 0.85–0.87. The RBF SVM substantially outperforms all others, achieving nearly 93% test accuracy and an EER of only 7.3%.

B. Model Performance at 16 kHz

Table II shows analogous results for the 16 kHz (for-rerec) condition.

Interestingly, performance does not degrade in the re-recorded, lower-bandwidth condition; in fact, most models slightly improve. The RBF SVM reaches an EER of 6.6%, while GMM and QDA also see modest gains. This indicates that the channel effects introduced in the re-recorded data do not remove the discriminative cues; they may even introduce additional artifacts that differ between real and fake clips.

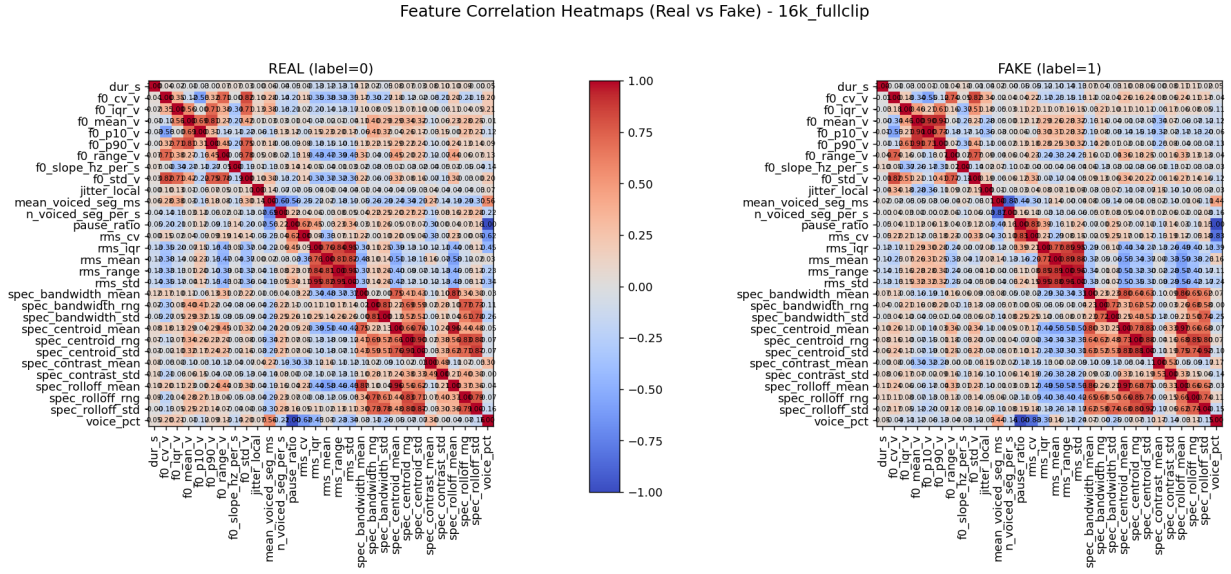

Fig. 8: Feature correlation heatmaps for real (left) and fake (right) speech at 16 kHz. Blocks of high correlation appear among spectral features (centroid, bandwidth, rolloff) and among energy features. Differences between the two heatmaps indicate how feature relationships shift between genuine and synthetic speech.

TABLE I: Model performance on FoR (44.1 kHz, 2-second clips).

Model	Train		Test		
	Accuracy	ROC–AUC	Accuracy	ROC–AUC	EER (%)
Logistic Regression	0.751	0.823	0.753	0.828	24.7
LDA	0.747	0.816	0.753	0.819	24.7
QDA	0.768	0.854	0.771	0.854	22.9
Gaussian Naïve Bayes	0.698	0.768	0.696	0.764	30.4
Linear SVM	0.754	0.823	0.755	0.826	24.5
RBF SVM	**0.953**	**0.990**	**0.927**	**0.980**	**7.3**
GMM (log-likelihood)	0.787	0.875	0.783	0.869	21.7

TABLE II: Model performance on FoR (16 kHz, re-recorded clips).

Model	Train		Test		
	Accuracy	ROC–AUC	Accuracy	ROC–AUC	EER (%)
Logistic Regression	0.756	0.831	0.754	0.834	24.6
LDA	0.753	0.826	0.752	0.827	24.8
QDA	0.774	0.858	0.771	0.857	22.9
Gaussian Naïve Bayes	0.701	0.772	0.698	0.769	30.2
Linear SVM	0.758	0.829	0.758	0.830	24.2
RBF SVM	**0.957**	**0.991**	**0.934**	**0.981**	**6.6**
GMM (log-likelihood)	0.811	0.895	0.805	0.888	19.5

C. Detection Error Trade-off Curves

We visualize the trade-off between FAR and FRR for the RBF SVM using DET curves. The following figure placeholders correspond to the 44.1 kHz and 16 kHz conditions:

The curves show that one can operate at FAR and FRR around 5% simultaneously, or drive FAR even lower at the cost of somewhat higher FRR, depending on application requirements.

D. McNemar's Tests

To establish whether differences between models are statistically significant, we computed McNemar's tests on pairwise model disagreements for both train and test sets. Key findings include:

- Logistic Regression, LDA, and Linear SVM show no significant pairwise differences on the test set (e.g., $p \approx 1.0$ for LDA vs. Logistic and $p > 0.4$ for Logistic vs. Linear SVM). Their small accuracy differences are not meaningful.
- QDA and GMM are significantly better than the linear

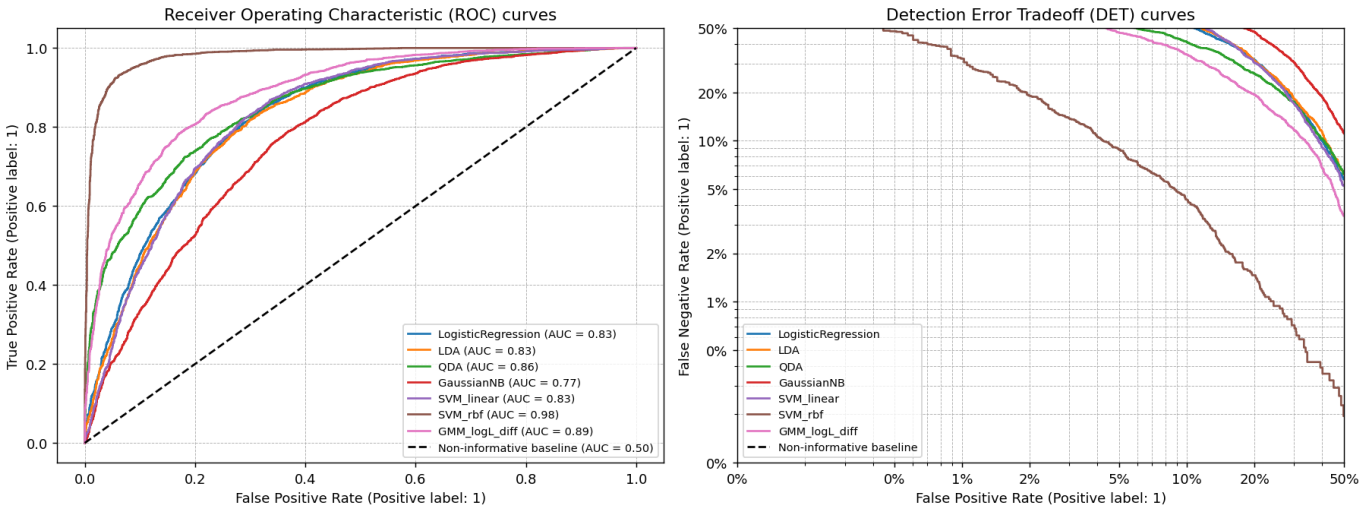

Fig. 9: Receiver Operating Characteristic (ROC) and Detection Error Trade-off (DET) curves for all classical models on the 16 kHz re-recorded FoR condition. Each colored line corresponds to one model (Logistic Regression, LDA, QDA, Gaussian Naïve Bayes, linear SVM, RBF SVM, and GMM log-likelihood difference). The ROC plot (left) shows that the RBF SVM dominates the other models with AUC close to 0.98, while the DET plot (right) illustrates that its operating points lie well below the non-informative baseline over a wide range of false positive rates.

models (e.g., Logistic vs. QDA or Logistic vs. GMM with $p \ll 0.05$), confirming that non-linear boundaries improve detection.

- On the 16 kHz condition, the GMM significantly outperforms QDA, reflecting its ability to model multimodal class distributions.
- The RBF SVM is significantly better than every other model in all pairwise comparisons (with p-values effectively zero), supporting the conclusion that its performance advantage is not due to random variation.
- Gaussian Naïve Bayes is significantly worse than all other models, consistent with its oversimplified feature-independence assumptions.

Together, these tests confirm the ranking:

$$\text{RBF SVM} \gg (\text{GMM} \approx \text{QDA}) > (\text{LR} \approx \text{LDA} \approx \text{Linear SVM}) \gg \text{GNB}.$$

VI. Discussion

A. What Makes Speech Sound "Real"?

The ANOVA and visualization results point to several key differences between real and synthetic speech in the FoR dataset:

- **Pitch variability.** Features such as $f0\_std\_v$, $f0\_range\_v$, and $f0\_cv\_v$ were among the most significant. Real speech displays richer, context-dependent intonation, whereas synthetic speech often exhibits flatter pitch contours. Even if the average pitch is realistic, the fine dynamics of rising and falling intonation are harder to synthesize.
- **Spectral richness.** Real recordings typically have greater high-frequency content and broader spectral bandwidth. This manifests in higher values for `spec_centroid_mean`, `spec_bandwidth_mean`, and `spec_rolloff_mean`. Synthetic speech, especially vocoder-based, may sound slightly smoother or muffled, lacking some of the sharp consonant bursts and recording noise that characterize natural speech.
- **Voice quality.** Jitter shows meaningful differences, suggesting that synthetic voices are often too "perfect" in their periodicity. Shimmer, however, was not a discriminative feature, indicating that amplitude cycle irregularities are either similar between classes or overshadowed by other amplitude effects.
- **Temporal structure.** Features related to pauses and voiced segment counts (voice_pct, pause_ratio, $n\_voiced\_seg\_per\_s$) were only mildly significant. Because all clips were two seconds long and contained fluent speech, both real and fake utterances had similar high-level timing patterns.

These findings align with prior work that uses prosody and spectral cues for synthesized speech detection [1], [3]. They also suggest directions for future TTS systems seeking to be less detectable: increasing pitch variability in a controlled way, adding realistic high-frequency content, and incorporating micro-perturbations in the glottal source.

B. Why Classical Models Perform Well

The strong performance of the RBF SVM (AUC ≈ 0.98, EER $\approx 7\%$) indicates that, in this handcrafted feature space,

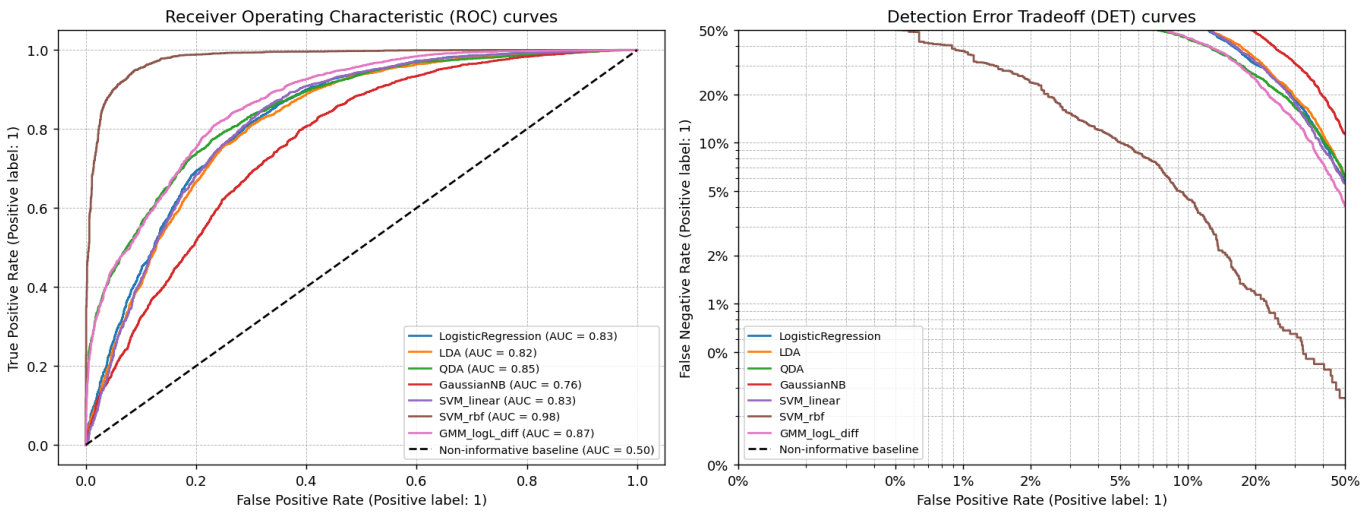

Fig. 10: ROC and DET curves for all classical models on the 44.1 kHz FoR two-second clips. The pattern closely mirrors the 16 kHz case: linear models (Logistic Regression, LDA, linear SVM) cluster together, QDA and the GMM provide moderate gains, and the RBF SVM clearly outperforms the others with the highest ROC–AUC and the lowest DET curve, indicating strong robustness across operating points.

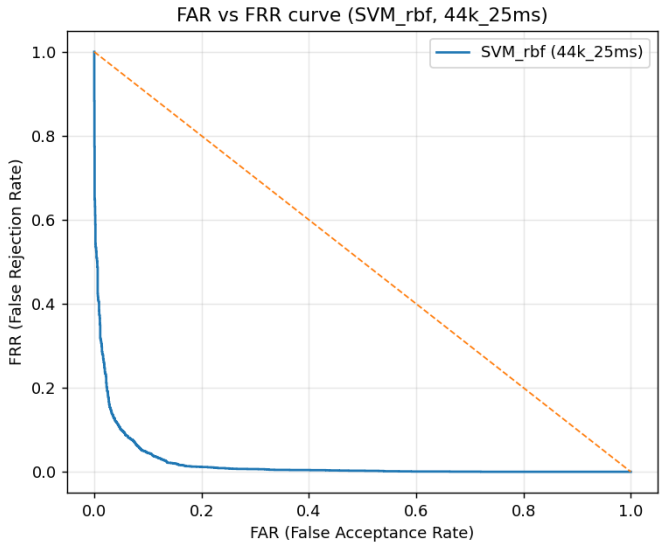

Fig. 11: Detection Error Trade-off (DET) curve for the RBF SVM on the 44.1 kHz condition. The EER point occurs where FAR and FRR intersect at approximately 7.3%.

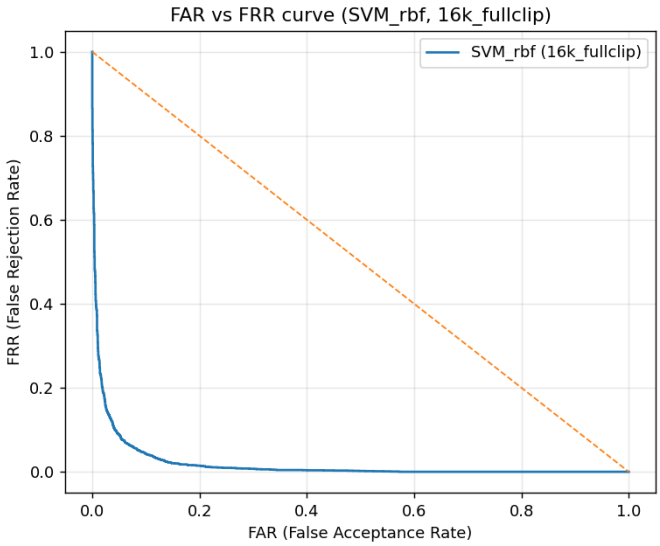

Fig. 12: DET curve for the RBF SVM on the 16 kHz condition. The equal error rate is slightly lower, around 6.6%.

real and fake speech are almost linearly separable after a modest non-linear transformation. The GMM and QDA also perform well, suggesting that class-conditional feature distributions are multi-modal and not strictly linearly separable.

Despite their simplicity, classical models provide several advantages:

- **Interpretability.** It is straightforward to inspect which features contribute most to the decision boundary and to visualize how real and fake samples differ along those dimensions.
- **Data efficiency.** Classical models require far fewer parameters than deep neural networks and can often achieve high performance with relatively small training sets.
- **Computational efficiency.** Feature extraction and model inference are fast enough for real-time applications on modest hardware.

These advantages make classical models attractive as base-

lines and as components in hybrid detection systems.

C. Generalization and Dataset-Specific Artifacts

One caveat is that the FoR dataset uses a specific set of TTS engines and recording conditions. The RBF SVM may exploit artifacts that are highly consistent within these engines but may not generalize to future synthesis methods. For example, if all fake clips from a particular TTS engine share a subtle pattern in spectral rolloff versus pitch, the model might key on that pattern. A new TTS engine that avoids this particular artifact could evade detection.

By contrast, the linear models, which plateau around 75% accuracy, may capture more fundamental differences (e.g., overall pitch variability and average spectral centroid) that are likely to persist across synthesis methods. In practice, a robust system might combine interpretable classical features with deeper representations or ensembles over multiple datasets.

D. Why We Did Not Balance the Data

As noted earlier, the FoR dataset is already approximately balanced between real and fake classes. Introducing additional balancing (especially synthetic oversampling) would risk:

1) Distorting the natural distribution of features.
2) Encouraging the models to learn patterns specific to synthetic oversampled points.
3) Making evaluation metrics such as accuracy and EER less representative of real-world deployment scenarios.

Because our focus is on understanding the natural separability of real vs fake speech under realistic conditions, we intentionally avoided such manipulations.

VII. Conclusion

This chapter presented a comprehensive classical machine learning baseline for deepfake audio detection on the Fake-or-Real dataset. Using a carefully engineered set of prosodic, voice-quality, and spectral features, we trained multiple models and evaluated them on both high-fidelity (44.1 kHz) and re-recorded (16 kHz) conditions.

The main findings can be summarized as follows:

- Pitch variability and spectral richness are the most discriminative cues; real speech exhibits broader and more irregular patterns in both domains compared to synthetic speech.
- Linear models such as Logistic Regression, LDA, and Linear SVM achieve around 75% accuracy (AUC ≈ 0.82), while QDA and GMM further improve performance to approximately 77–80% accuracy (AUC ≈ 0.86–0.89).
- An RBF SVM significantly outperforms all other models, achieving around 93% test accuracy and EER near 7% on both sampling-rate conditions.
- Detection remains strong even after re-recording and downsampling to 16 kHz, indicating that key cues survive realistic channel distortions.
- McNemar's tests confirm that the performance gains of non-linear models, particularly the RBF SVM, are statistically significant rather than arising by chance.

These results demonstrate that classical models built on interpretable acoustic features can form a strong baseline for deepfake audio detection, sometimes rivaling or complementing more complex deep learning approaches. Future work could extend this baseline by:

- Incorporating additional features (e.g., phase-based or cepstral features such as MFCCs and CQCCs).
- Evaluating on other datasets, including ASVspoof and FakeAV-celeb, to test cross-corpus generalization.
- Integrating these features into hybrid systems that combine classical and deep architectures.

Ultimately, understanding *why* current synthetic speech sounds different from real speech is as important as achieving high classification accuracy. The analyses presented here provide a step in that direction and offer a transparent reference point for future research on robust, explainable deepfake audio detection.

References

[1] A. Hamza, A. R. R. Javed, F. Iqbal, *et al.*, "Deepfake audio detection via MFCC features using machine learning," *IEEE Access*, vol. 10, pp. 134018–134028, 2022.

[2] R. Reimao and V. Tzerpos, "The Fake-or-Real dataset: A benchmark for audio deepfake detection," APTLY Lab, York University, 2021. [Online]. Available: https://aptly.ca

[3] J. Yi, X. Liu, and X. Chen, "Audio deepfake detection: A survey," *arXiv preprint* arXiv:2308.14970, 2023.

[4] H. Patino, A. L. N. Delgado, and T. Kinnunen, "ASVspoof 2021: Leveraging automatic speaker verification for deepfake speech detection," in *Proc. ASVspoof Workshop*, 2021.

[5] A. W. Storey, "Binary cough detection using classical machine learning on the COUGHVID dataset," M.S. Project Report, Clarkson University, Potsdam, NY, USA, 2021.

[6] A. W. Storey, "Noise classification with classical ML models on the MUSAN dataset," Technical Report, Clarkson University, Potsdam, NY, USA, 2021.

[7] Z. Khalid, S. H. Yoon, and A. C. Kot, "FakeAVCeleb: A novel audio-video multimodal deepfake dataset," in *Proc. IEEE ICASSP*, 2021, pp. 6164–6168.