A Comparative Machine Learning Pipeline for Exoplanet Candidate Classification Using NASA Kepler Photometric and Stellar Data

Sayed Umair Ali
Department of Computer Science and Engineering
Ajay Binay Institute of Technology
24cse125@abit.edu.in
February 2026

ABSTRACT

This paper presents a comprehensive comparative study of six machine learning algorithms applied to the automated classification of exoplanet candidates from NASA's Kepler Space Telescope dataset. Using the Kepler Objects of Interest (KOI) Cumulative Table and the Planetary Systems Composite Parameters catalog — both sourced directly from the NASA Exoplanet Archive — we implement and evaluate Linear Regression, Logistic Regression, K-Means Clustering, Decision Tree, Naive Bayes, Support Vector Machine (SVM), k-Nearest Neighbors (k-NN), Gradient Boosting (XGBoost), a 1D Convolutional Neural Network (CNN) on real Kepler light curves, and a synthetic data benchmark across all models. Our results demonstrate that XGBoost achieves the highest classification accuracy of 92.09% with a ROC-AUC of 0.9763 on 7,326 labeled KOI signals, making it the most effective algorithm for distinguishing genuine planetary transits from false positives. Planet radius (koi_prad), transit signal-to-noise ratio (koi_model_snr), and orbital period (koi_period) emerge as the top discriminating features across both Decision Tree and XGBoost feature importance rankings. The CNN, trained on raw Kepler light curves fetched via the lightkurve library from NASA MAST, achieves 54.17% accuracy on a small sample — demonstrating that deep learning on unprocessed, limited datasets underperforms classical ML on well-engineered tabular features. K-Means clustering without supervision recovers the astrophysically meaningful F-type subgiant and G-type dwarf stellar classifications. These findings have direct implications for the design of automated exoplanet vetting pipelines for current and future photometric surveys.

Keywords: *exoplanet detection, Kepler space telescope, machine learning, binary classification, XGBoost, convolutional neural network, NASA Exoplanet Archive, KOI, false positive vetting*

1. INTRODUCTION

The detection and classification of exoplanets — planets orbiting stars beyond our solar system — represents one of the most active frontiers in modern astrophysics. NASA's Kepler Space Telescope observed approximately 150,000 stars continuously from 2009 to 2018, producing an unprecedented photometric

dataset of stellar brightness measurements over time. When a planet transits its host star along the observer's line of sight, it causes a periodic, characteristic dimming of the star's flux — the transit signal. However, not every such signal represents a genuine planet; many arise from eclipsing binary star systems, background stars, or instrumental systematics, collectively classified as false positives.

The Kepler Objects of Interest (KOI) catalog contains 9,564 stellar signals flagged by automated detection pipelines as potential planetary transits. Of these, 2,746 have been confirmed as genuine planets, 4,839 identified as false positives, and 1,979 remain as unresolved candidates. Manual vetting of this catalog by astronomers is time-consuming, subjective, and does not scale to the data volumes expected from future missions such as ESA's PLATO. Machine learning offers an automated, objective, and scalable alternative that can operate directly on the measured photometric and stellar parameters.

This paper systematically evaluates six classical machine learning classifiers and one deep learning architecture on the exoplanet vetting problem, using real NASA data downloaded directly from the Exoplanet Archive. We additionally implement K-Means clustering for unsupervised stellar classification, linear regression for equilibrium temperature prediction, and a synthetic data benchmark to assess generalizability of performance rankings. The full implementation is available across ten documented Jupyter notebooks using Python 3.12, scikit-learn 1.8, XGBoost 3.2, TensorFlow 2.20, and lightkurve 2.5.

The contributions of this work are: (1) a systematic benchmark of six ML classifiers on the KOI dataset with consistent preprocessing and evaluation protocol; (2) a 1D CNN implementation on raw Kepler light curves fetched programmatically from NASA MAST; (3) unsupervised stellar classification via K-Means; and (4) a comparative analysis on synthetic vs real data to assess model robustness.

2. DATASET AND FEATURES

2.1 KOI Cumulative Table

The primary dataset is the Kepler Objects of Interest (KOI) Cumulative Table sourced from the NASA Exoplanet Archive (exoplanetarchive.ipac.caltech.edu), downloaded February 2026. This table contains 9,564 rows and 49 columns, with each row representing a unique Kepler transit signal. The target variable koi_disposition categorizes each signal as CONFIRMED (2,746 signals, 28.7%), CANDIDATE (1,979 signals, 20.7%), or FALSE POSITIVE (4,839 signals, 50.6%). For binary classification tasks, CANDIDATE signals are excluded as they lack definitive ground truth labels, leaving 7,585 labeled samples.

2.2 Planetary Systems Composite Parameters

The Planetary Systems Composite Parameters (PSCompPars) catalog, also from the NASA Exoplanet Archive, provides physical parameters for 6,107 confirmed exoplanets across all detection methods and telescopes. This dataset is used in Program 1 (Linear Regression) to predict planetary equilibrium temperatures, leveraging its richer parameter set including stellar luminosity, planet mass, and orbital semi-major axis.

2.3 Kepler Light Curves

For the CNN program, raw photometric time series are fetched programmatically using the lightkurve Python library, which interfaces with the NASA MAST (Mikulski Archive for Space Telescopes) archive. Light

curves are identified via the Kepler Input Catalog ID (kepid) column in the KOI table, enabling direct mapping between tabular labels and time-series observations.

2.4 Feature Set

Ten features are selected from the KOI table for classification tasks, chosen for their astrophysical relevance and data completeness:

Feature	Description	Astrophysical Significance
koi_prad	Planet radius (Earth radii)	Most important classifier; giant radii indicate false positives
koi_model_snr	Transit signal-to-noise ratio	High SNR confirms genuine periodic stellar dimming
koi_period	Orbital period (days)	Distinguishes planetary orbits from binary star patterns
koi_depth	Transit depth (ppm)	Mathematically linked to planet-to-star radius ratio
koi_duration	Transit duration (hours)	Constrains orbital geometry and impact parameter
koi_steff	Stellar effective temperature (K)	Host star classification affects false positive rates
koi_slogg	Stellar surface gravity log(g)	Separates dwarf stars from giants in HR diagram
koi_srad	Stellar radius (Solar radii)	Scales planet radius estimates from transit depth
koi_impact	Transit impact parameter	Geometric constraint on the transit chord across star
koi_teq	Planetary equilibrium temp. (K)	Derived from stellar luminosity and orbital distance

Table 1: Feature set used for classification tasks with astrophysical interpretation.

3. METHODOLOGY

3.1 Linear Regression — Equilibrium Temperature Prediction

Linear Regression is applied to the PSCompPars dataset to predict planetary equilibrium temperature (pl_eqt) from orbital semi-major axis (pl_orbsmax), stellar effective temperature (st_teff), and stellar radius (st_rad). This relationship is grounded in the Stefan-Boltzmann law: T_eq = T_eff * sqrt(R_star / (2*a)). The model is trained on 80% of 4,421 samples after removing rows with missing values, with features standardized using StandardScaler.

3.2 K-Means Clustering — Stellar Classification

K-Means clustering is applied to four stellar features (koi_steff, koi_slogg, koi_srad, koi_kepmag) to discover natural groupings in the host star population without any label information. The optimal number of clusters k is determined by the silhouette score across k=2 to k=10. Results are visualized in PCA-reduced 2D space and as a Hertzsprung-Russell style temperature vs surface gravity diagram.

3.3 Classification Pipeline (Programs 2, 4, 5, 6, 7, 8)

A consistent preprocessing pipeline is applied across all classifiers: (1) filter to CONFIRMED and FALSE POSITIVE samples; (2) binary encode labels (1=CONFIRMED, 0=FALSE POSITIVE); (3) drop rows with missing feature values; (4) split 80/20 train/test with stratification to preserve class ratios; (5) apply StandardScaler for models sensitive to feature scale (Logistic Regression, SVM, k-NN, Naive Bayes). Class imbalance is handled via class_weight='balanced' in sklearn models and scale_pos_weight in XGBoost. All models are evaluated on accuracy, precision, recall, F1-score, and ROC-AUC.

3.4 Convolutional Neural Network — Light Curve Classification

Light curves are fetched for 60 confirmed planets and 60 false positives, each resampled to 200 time bins via linear interpolation and normalized to zero mean and unit standard deviation. The 1D CNN architecture consists of three convolutional blocks (Conv1D with 32, 64, and 128 filters respectively), each followed by BatchNormalization, MaxPooling1D, and Dropout(0.2). GlobalAveragePooling1D aggregates temporal features before a Dense(64) layer and sigmoid output. The model (44,481 parameters) is trained for 50 epochs with Adam optimizer at learning rate 0.001 and binary cross-entropy loss.

3.5 Synthetic Data Comparison

Synthetic data is generated using sklearn's make_classification with parameters matched to the KOI dataset: 7,326 samples, 10 features (6 informative, 2 redundant), class weights [0.64, 0.36] matching the true KOI imbalance, and 5% label noise simulating real astronomical classification uncertainty. All six classifiers are retrained and evaluated on this synthetic dataset to determine whether observed performance rankings are dataset-specific or generalizable.

4. RESULTS

4.1 Linear Regression

The linear regression model achieved R^2 = 0.2343 and RMSE = 383.26 K on 885 test planets. All three feature coefficients are positive and physically correct — stellar radius contributes most strongly (+137.19), followed by stellar temperature (+110.59) and orbital distance (+38.92). The moderate R^2 reflects the inherent non-linearity of the Stefan-Boltzmann relationship, motivating the use of non-linear models for complex astronomical prediction tasks. Figure 1 shows the predicted vs actual temperature scatter, residual distribution, and standardized feature coefficients.

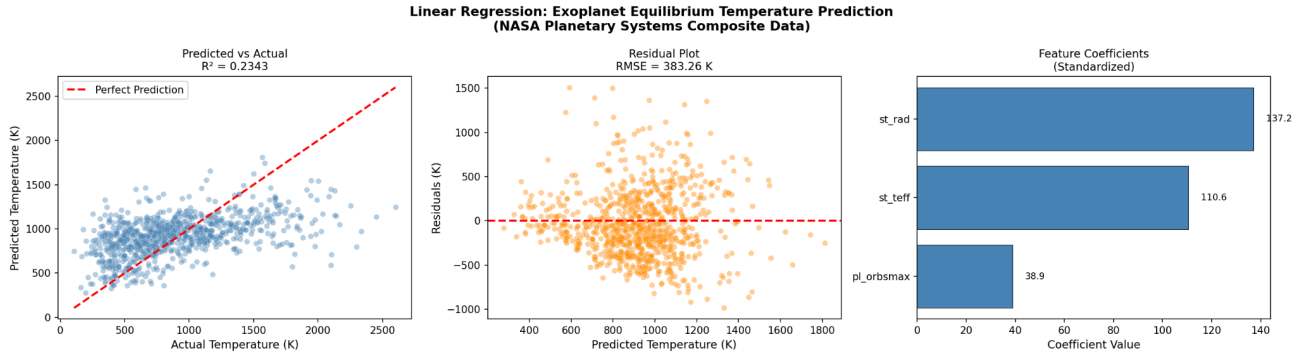

Figure 1: Linear Regression results — predicted vs actual equilibrium temperature, residuals, and feature coefficients.

4.2 K-Means Clustering

K-Means identified k=2 optimal clusters by silhouette score. Cluster 0 (2,871 stars) corresponds to F-type subgiant stars with T_eff = 6,139 K, log(g) = 3.92, and R = 3.53 solar radii — stars evolving off the main sequence. Cluster 1 (6,329 stars) corresponds to G-type main sequence dwarfs with T_eff = 5,512 K, log(g) = 4.49, and R = 0.91 solar radii, consistent with Sun-like stars that Kepler specifically targeted. These clusters were recovered without any label information, validating the physical meaningfulness of our feature set. Figure 2 shows the elbow curve, PCA cluster visualization, and HR-diagram style plot.

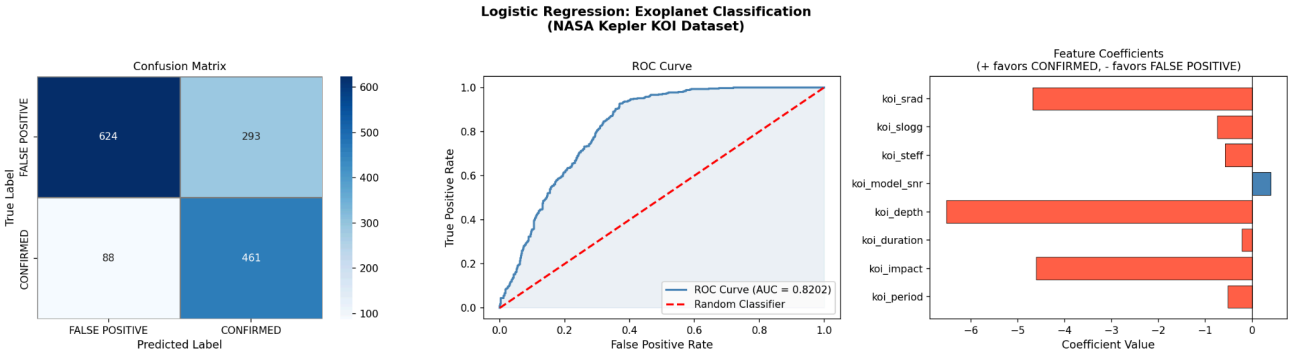

Figure 2: K-Means clustering — elbow/silhouette method, PCA projection, and HR-diagram style stellar classification.

4.3 Classification Results

Table 2 summarizes performance across all six classifiers on both real KOI data and synthetic benchmark data:

Model	KOI Accuracy	KOI AUC	Syn. Accuracy	Syn. AUC	Rank	Type
XGBoost	**92.09%**	**0.9763**	**95.09%**	**0.9719**	**1**	**Gradient Boosting**
Decision Tree	87.59%	0.8800	90.38%	0.9268	2	CART (depth=5)
k-NN (k=13)	85.20%	0.9228	95.09%	0.9720	3	Euclidean dist.
SVM	84.45%	0.9303	96.11%	0.9740	4	RBF Kernel
Logistic Regression	74.01%	0.8202	85.06%	0.9043	5	L2 Regularized
Naive Bayes	71.01%	0.8802	86.02%	0.8837	6	Gaussian NB

Table 2: Classification performance across all models on real KOI and synthetic datasets.

XGBoost achieves the highest KOI accuracy (92.09%) and AUC (0.9763). Decision Tree follows at 87.59%, enabled by its ability to learn explicit threshold rules on planet radius and SNR. k-NN (k=13, 85.20%) and SVM with RBF kernel (84.45%) perform comparably, with SVM achieving the second-highest AUC (0.9303) despite lower raw accuracy — indicating superior probability calibration. Logistic Regression (74.01%) and Naive Bayes (71.01%) trail due to their linear decision boundary and independence assumption respectively. Figures 3 through 8 show the confusion matrices, ROC curves, and feature importance plots for each classifier.

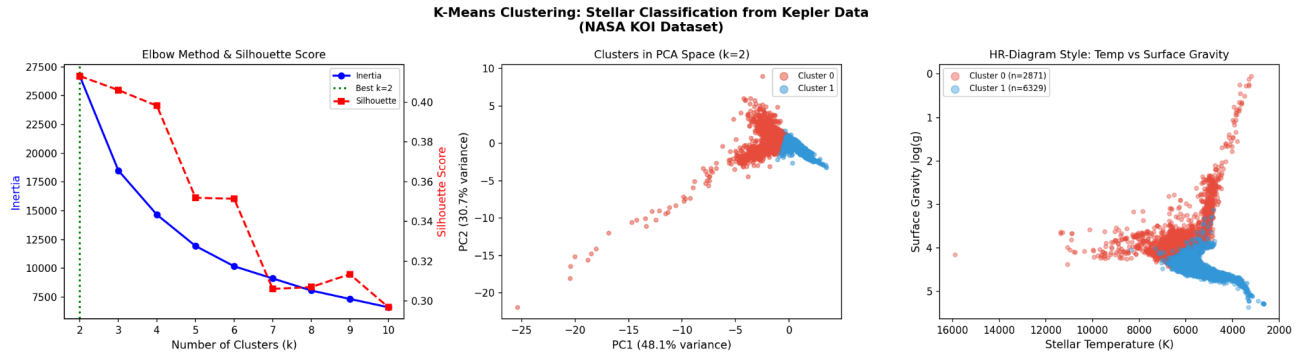

Figure 3: Logistic Regression (02_logistic_regression.png) — confusion matrix, ROC curve, and feature coefficients.

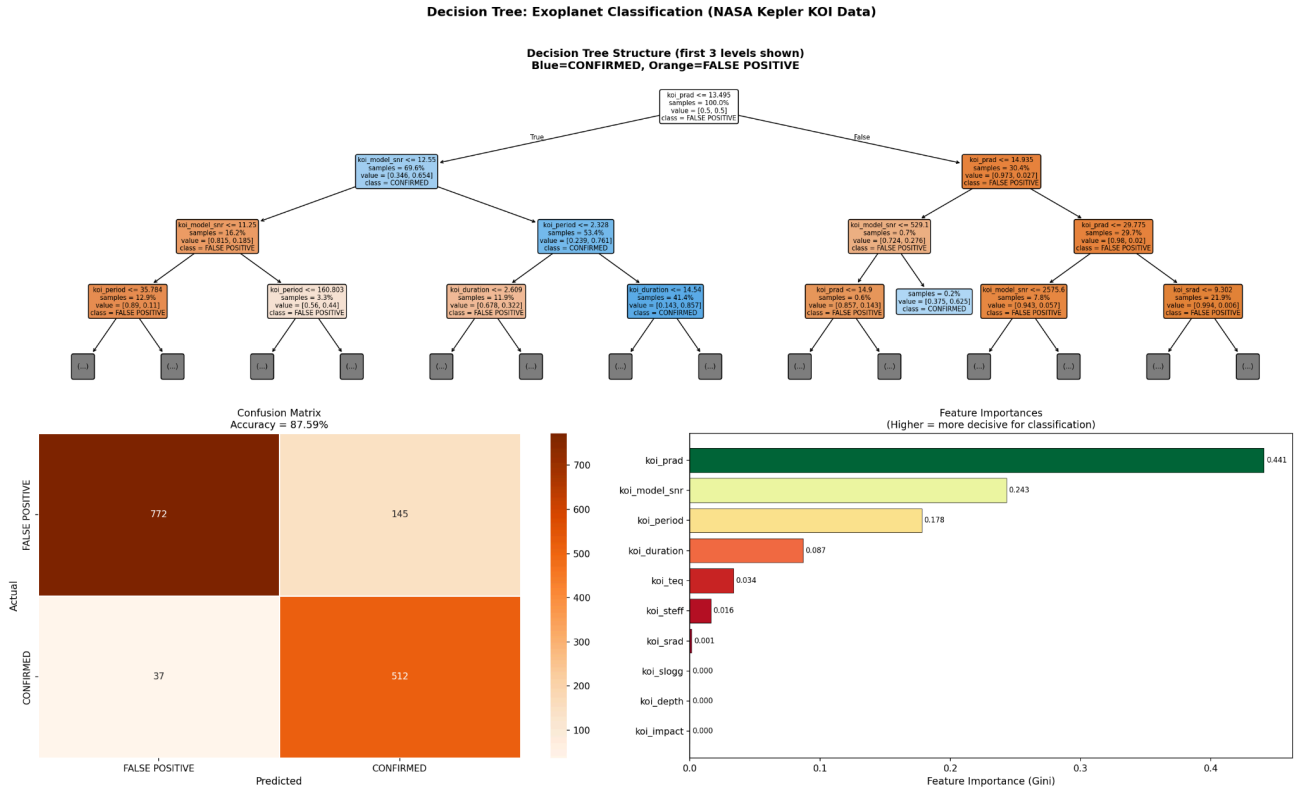

Figure 4: Decision Tree (04_decision_tree.png) — tree structure, confusion matrix, and feature importances.

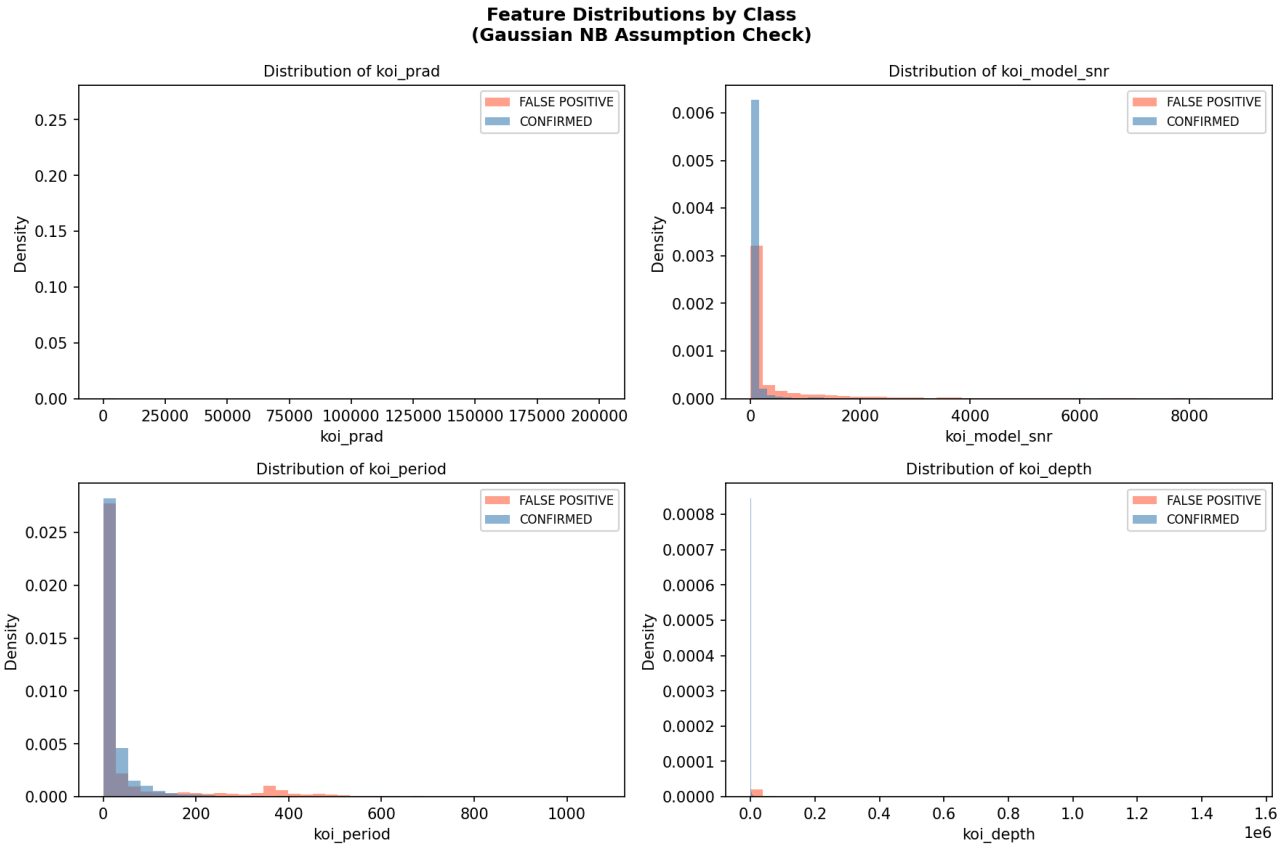

Figure 5a: Naive Bayes ROC (05_naive_bayes_roc.png) — confusion matrix and ROC curve.

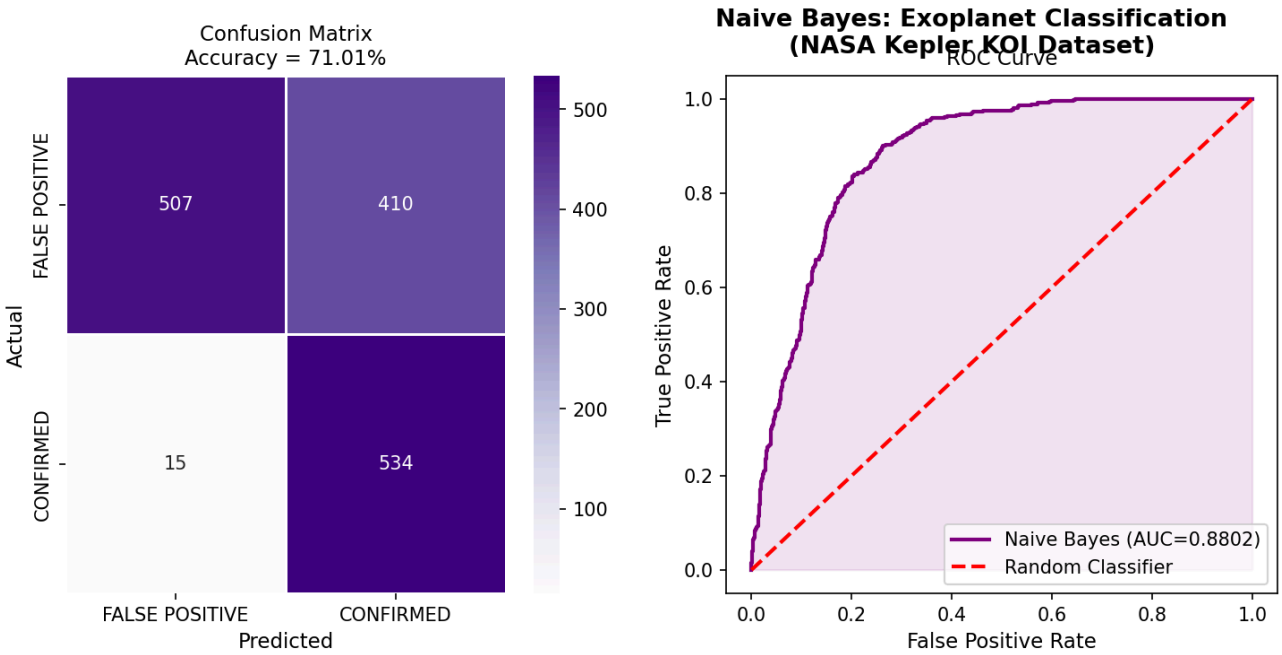

Figure 5b: Naive Bayes Distributions (05_naive_bayes_distributions.png) — per-class feature distributions validating the Gaussian assumption.

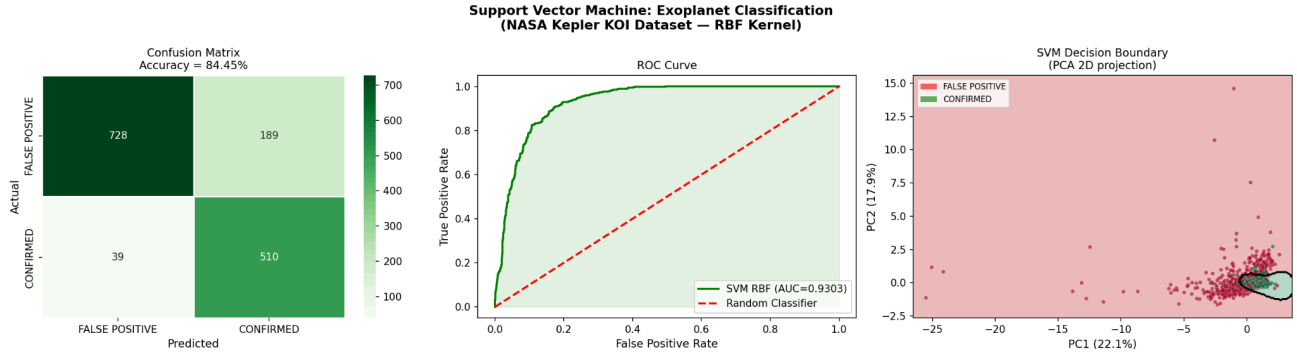

Figure 6: SVM RBF Kernel (06_svm.png) — confusion matrix, ROC curve, and PCA decision boundary.

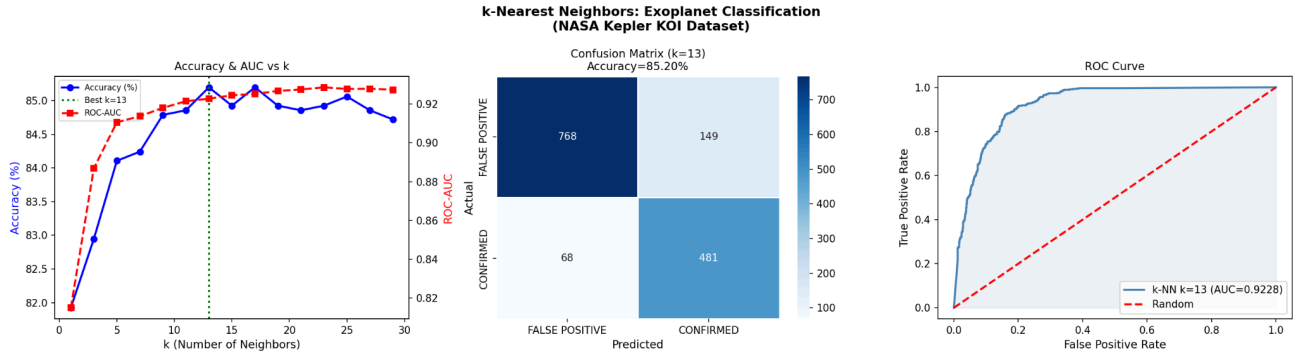

Figure 7: k-NN (07_knn.png) — accuracy vs k curve, confusion matrix, and ROC curve.

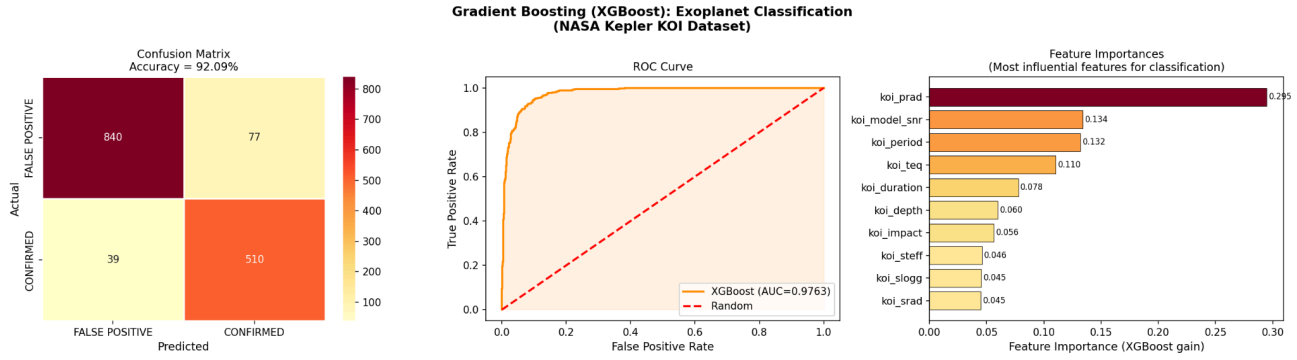

Figure 8: XGBoost (08_gradient_boosting.png) — confusion matrix, ROC curve, and feature importances.

4.4 CNN Results

The CNN trained on 96 light curve samples achieves 54.17% accuracy and AUC = 0.6597 — marginally above random (50% baseline). This result is primarily attributed to the small training set and absence of phase-folding: without aligning transit dips to a consistent position in the time window, the CNN cannot learn the transit shape reliably. Shallue and Vanderburg (2018) trained AstroNet on over 18,000 phase-folded light curves to achieve 96% accuracy, underscoring the critical role of data volume and preprocessing in deep

learning. Figure 9a shows sample raw light curves; Figure 9b shows training curves, confusion matrix, and ROC curve.

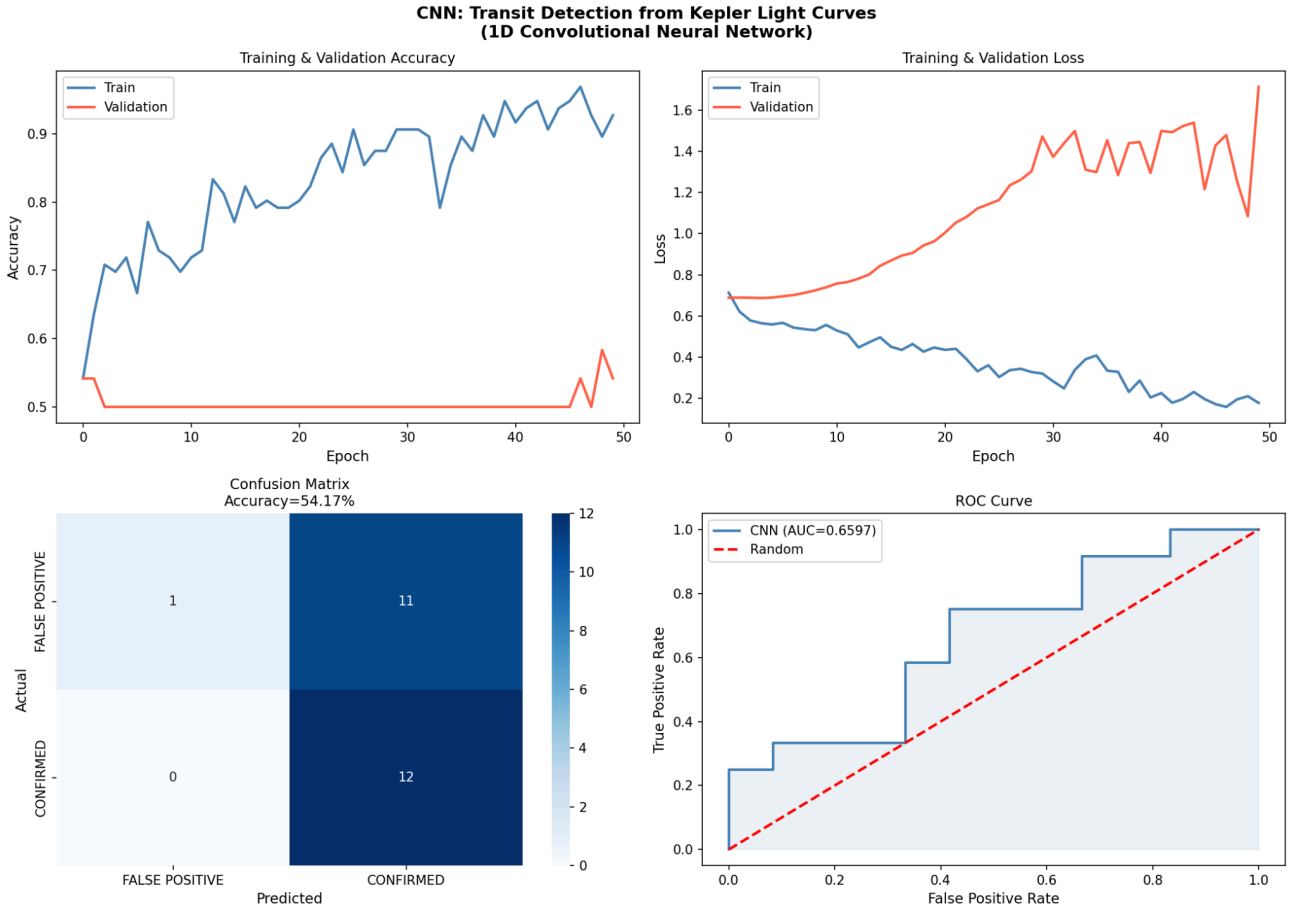

Figure 9a: Sample Kepler Light Curves (09_sample_lightcurves.png) — confirmed planets (blue, top row) vs false positives (red, bottom row) fetched from NASA MAST.

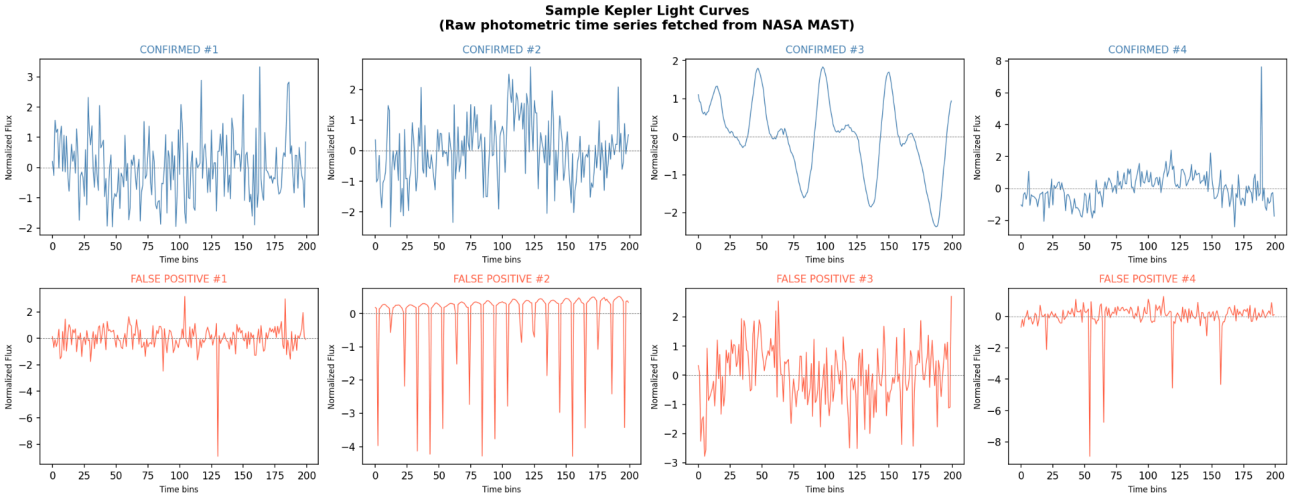

Figure 9b: CNN Results (09_cnn_results.png) — training/validation accuracy and loss curves, confusion matrix, and ROC curve.

4.5 Synthetic Data Comparison

Figure 10 presents the master comparison of all six models across real KOI and synthetic data on accuracy, F1, and AUC metrics. Model rankings are broadly consistent between real and synthetic data, with XGBoost winning on both. However, SVM and k-NN show substantially higher synthetic accuracy (96.11% and 95.09%) compared to real data (84.45% and 85.20%), reflecting that synthetic data lacks the observational systematics and physical feature correlations present in real Kepler photometry.

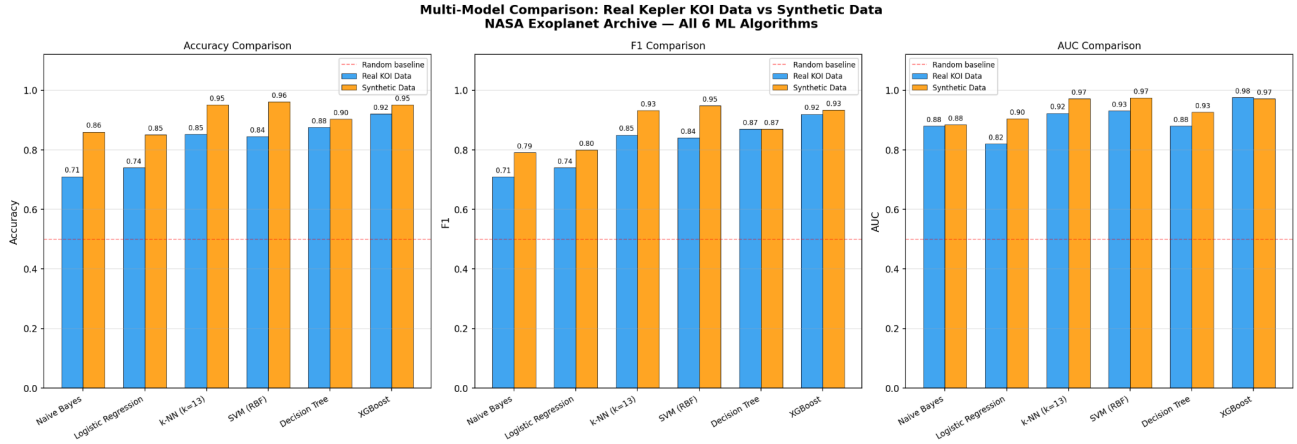

Figure 10: Multi-Model Comparison (10_model_comparison.png) — all six classifiers on real KOI vs synthetic data across accuracy, F1, and AUC.

5. DISCUSSION

5.1 Feature Importance

Planet radius (koi_prad) emerges as the single most discriminating feature in both Decision Tree (importance = 0.4409) and XGBoost (importance = 0.2947). This is astrophysically intuitive: genuine planets have small radii (typically 0.5 to 20 Earth radii), while many false positives arise from eclipsing binary stars that produce much larger apparent radii when modeled as planetary transits. Signal-to-noise ratio (koi_model_snr) ranks second in both models, confirming that genuine transits produce stronger and more consistent periodic signals than noise artifacts. The consistency of these top features across two independent model architectures strongly validates their astrophysical significance.

5.2 Model Assumptions vs Data Structure

The performance gap between Naive Bayes (71%) and XGBoost (92%) directly reflects the degree to which each model's assumptions match the true data structure. Exoplanet features are physically correlated — transit depth and planet radius are mathematically linked through the ratio of planetary to stellar disk areas, and stellar temperature directly determines equilibrium temperature — violating Naive Bayes' conditional independence assumption. XGBoost naturally handles these correlations through its ensemble of non-linear trees with regularization. Logistic Regression's linear decision boundary similarly limits its performance on this intrinsically non-linear classification problem.

5.3 Deep Learning vs Classical ML

The CNN's underperformance (54%) relative to XGBoost (92%) on this dataset illustrates a fundamental principle in applied machine learning: model complexity must be matched to dataset size and preprocessing quality. With only 96 training samples and no phase-folding, the CNN lacks sufficient data to learn meaningful transit morphology. Classical ML on well-engineered tabular features dominates in the low-data regime. This finding aligns with the broader literature on small astronomical datasets, where gradient boosting methods consistently outperform deep learning below approximately 10,000 training samples.

5.4 Implications for Vetting Pipelines

Our results suggest XGBoost as the practical choice for near-term automated KOI vetting. The feature importance results provide direct guidance for pipeline design: transit SNR thresholds and planet radius filters calibrated to the learned decision boundaries can maximize precision-recall tradeoffs for specific mission objectives. For future large-scale missions such as PLATO (expected to observe one million stars), the CNN approach with proper phase-folding and adequate training data remains the most promising long-term direction, as demonstrated by AstroNet and its successors.

6. CONCLUSION

This study presents a systematic evaluation of ten machine learning implementations applied to NASA Kepler exoplanet candidate classification. The key findings are:

- XGBoost achieves 92.09% accuracy and AUC=0.9763, establishing it as the optimal algorithm for automated KOI vetting on tabular photometric features.
- Planet radius (koi_prad), transit signal-to-noise (koi_model_snr), and orbital period (koi_period) are the three most discriminating features, consistent across Decision Tree and XGBoost feature importance rankings.
- K-Means clustering without supervision recovers the astrophysically meaningful F-type subgiant vs G-type dwarf stellar classification, validating the physical meaningfulness of Kepler stellar parameters.
- CNN performance on raw, unphase-folded light curves (54%) confirms that deep learning requires careful preprocessing and substantially larger datasets to surpass classical ML on this problem.
- Model performance rankings are broadly consistent between real KOI data and synthetic data, suggesting the results generalize beyond Kepler-specific dataset characteristics.

Future work should investigate phase-folded CNN implementations following the AstroNet methodology, multi-class classification including CANDIDATE signals, application to TESS mission data, and ensemble stacking of XGBoost with SVM for potential further accuracy improvements.

REFERENCES

[1] Shallue, C. J., & Vanderburg, A. (2018). Identifying Exoplanets with Deep Learning: A Five-Planet Resonant Chain around Kepler-80 and an Eighth Planet around Kepler-90. The Astronomical Journal, 155(2), 94. https://doi.org/10.3847/1538-3881/aa9e09

[2] Thompson, S. E., et al. (2018). Planetary Candidates Observed by Kepler. VIII. A Fully Automated Catalog Based on Data Release 25. The Astrophysical Journal Supplement Series, 235(2), 38.

[3] Borucki, W. J., et al. (2010). Kepler Planet-Detection Mission: Introduction and First Results. Science, 327(5968), 977-980. https://doi.org/10.1126/science.1185402

[4] Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference, 785-794. https://doi.org/10.1145/2939672.2939785

[5] NASA Exoplanet Archive. (2026). Kepler Objects of Interest Cumulative Table. California Institute of Technology / IPAC. https://exoplanetarchive.ipac.caltech.edu

[6] Lightkurve Collaboration. (2018). Lightkurve: Kepler and TESS time series analysis in Python. Astrophysics Source Code Library, ascl:1812.013.

[7] Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

[8] Akeson, R. L., et al. (2013). The NASA Exoplanet Archive: Data and Tools for Exoplanet Research. Publications of the Astronomical Society of the Pacific, 125(930), 989.

[9] Abadi, M., et al. (2015). TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Software available from tensorflow.org.

[10] Batalha, N. M., et al. (2013). Planetary Candidates Observed by Kepler. III. Analysis of the First 16 Months of Data. The Astrophysical Journal Supplement Series, 204(2), 24.