ExoHabitAI

Milestone 2 - Dataset Preparation and ML Model for Habitability Prediction

Name: Satya Sri Dheeraj M

Date: 25 January 2026

0. Overview of Milestone 2

This milestone focuses on building a scientifically valid machine learning pipeline using the preprocessed dataset developed in Milestone 1. The primary objective is not to claim definitive habitability, but to train and evaluate models that can rank exoplanets according to their likelihood of satisfying standard habitable zone criteria. Special emphasis is placed on preventing data leakage, handling extreme class imbalance, and using evaluation metrics appropriate for real-world astronomical prioritization tasks.

---

1. Scientific Reasonability of Results

The final results are scientifically reasonable and reflect a correctly implemented machine learning pipeline under realistic constraints. The most important improvement is the deduplication of the dataset, where 8,859 duplicate planet entries were removed, reducing the data from 12,894 rows to 4,035 unique planets. This step eliminated memorization effects that had previously caused unrealistically perfect performance. The subsequent removal of 8 rows with missing values resulted in a final dataset of 4,027 planets, which is a negligible reduction and does not affect statistical validity.

The use of star system–aware splitting further strengthens the methodology. Since planets orbiting the same star share identical stellar properties, random splitting would allow leakage through shared stellar features. By grouping planets by star system and ensuring no overlap between training and test sets, the model is forced to generalize to entirely unseen stellar environments. This choice is critical for scientific validity and reflects how the model would be used in practice for newly discovered exoplanets.

a. Performance Interpretation

The dataset is extremely imbalanced, with only 27 habitable planets (0.67%). As expected, this makes standard classification metrics misleading. Logistic Regression assigns higher probabilities to habitable planets but does not produce positive predictions at the default 0.5 threshold, resulting in zero binary recall. This behavior is appropriate given the rarity of positive examples and highlights why binary classification is not the correct evaluation framework for this problem.

Tree-based models show different limitations. Random Forest struggles with the imbalance and exhibits poor generalization, while XGBoost achieves moderate and consistent recall across cross-validation and test data. The disagreement between models indicates genuine learning rather than leakage, which is a positive outcome.

b. Evaluation Metrics and Ranking Quality

Because the goal is to prioritize promising candidates rather than make hard classifications, ranking-based metrics are the most appropriate. The model successfully ranks most habitable planets near the top of the candidate list, demonstrating practical utility despite weak binary classification performance.

---

2. Model Performance Summary (Test Set)

Model	CV Recall (Mean ± Std)	Test Recall	Test F1	ROC-AUC
Logistic Regression	0.87 ± 0.17	0.00*	0.00	0.99
Random Forest	0.26 ± 0.16	0.14	0.25	0.92
XGBoost	0.44 ± 0.20	0.43	0.43	0.95

* Logistic Regression produces no positive predictions at threshold 0.5 but provides useful probability rankings.

a. Ranking Performance (Final Model)

Metric	Value	Interpretation
Recall@10	0.43	3 of 7 habitable planets in top 10
Recall@20	0.71	5 of 7 habitable planets in top 20
Recall@50	0.86	6 of 7 habitable planets in top 50
Avg Precision	0.47	Strong ranking quality under imbalance

These results show that the model is effective at prioritizing habitable candidates even when binary classification is unreliable.

b. Interpretation of Model Behavior

The observed differences in performance across models highlight the impact of model choice under extreme class imbalance and leakage-free constraints. Logistic Regression assigns elevated probabilities to habitable planets but remains conservative at the default classification threshold, making it suitable for ranking rather than binary decisions. Random Forest exhibits difficulty generalizing from a small number of positive examples, while XGBoost provides a balanced trade-off between recall and stability. These behaviors are consistent with expectations for imbalanced scientific datasets and indicate genuine learning rather than memorization.

---

3. Notebook Explanation

a. Introduction

This project aims to build a machine learning pipeline to rank potentially habitable exoplanets using NASA TESS data. Since true habitability cannot be directly observed, labels are derived from rule-based habitable zone criteria. This makes data leakage a central concern and requires careful methodological design.

b. Data Leakage and Proxy Leakage

Initial models achieved unrealistically perfect performance due to direct data leakage. Features such as insolation, equilibrium temperature, planet radius, orbital eccentricity, and stellar temperature were used both to create the labels and to train the model. Including these features allowed the model to trivially learn the labeling rules.

After removing these features, proxy leakage was identified. Certain physical properties (for example, stellar mass and radius) allow reconstruction of excluded features through known astrophysical relationships. These proxy features were also removed to prevent indirect leakage, even though they are scientifically meaningful observables.

c. Feature Selection

After excluding direct labeling features, derived habitability indices, and proxy features, the final model uses 15 features. These include orbital parameters, planet mass, stellar metallicity and surface gravity, discovery year, and one-hot encoded categorical variables for planet and stellar types. This restricted feature set forces the model to learn indirect correlations rather than explicit habitability rules.

d. Data Splitting Strategy

To prevent star system–level leakage, planets were split using group-aware methods so that all planets from a given star system appear in only one split. Cross-validation was performed using GroupKFold, producing conservative but realistic performance estimates.

e. Handling Class Imbalance and Evaluation

Due to extreme class imbalance, accuracy and standard recall are not meaningful metrics. Instead, ranking-based metrics such as Recall@k and Average Precision are used. These metrics directly measure how well the model prioritizes habitable planets near the top of the candidate list, which aligns with the real-world use case of allocating telescope observation time.

f. Conclusion

The final pipeline trades inflated performance for scientific validity. While binary classification performance is limited, the model provides meaningful rankings that can efficiently prioritize promising exoplanet candidates. This approach demonstrates responsible machine learning practice in a scientific setting, emphasizing leakage prevention, honest evaluation, and clear communication of limitations.