Milestone–2

Machine Learning Model Training for Exoplanet Habitability Prediction

1. Introduction

The discovery of exoplanets has significantly increased in recent years due to advancements in astronomical observation technologies. However, identifying which of these planets may be capable of supporting life remains a complex challenge. Machine Learning (ML) provides an effective approach to analyze large astronomical datasets and identify patterns related to planetary habitability.

This milestone focuses on the **training, evaluation, and selection of the best machine learning model** to predict the habitability potential of exoplanets using preprocessed planetary and stellar data. The goal is to build a reliable and interpretable ML model that can classify exoplanets as **potentially habitable or non-habitable**, along with a probability-based habitability score.

2. Problem Formulation

2.1 Learning Type

The problem is formulated as a **Supervised Machine Learning** task, as the dataset contains labeled examples indicating whether an exoplanet is habitable or not.

2.2 Prediction Type

This project primarily addresses a **Binary Classification problem**, defined as:

Class Value	Description
1	Potentially Habitable
0	Non-Habitable

The model learns from historical data and predicts the habitability class for unseen exoplanets.

2.3 Input and Output

Input:

- Cleaned and engineered numerical and categorical features

- Planetary features such as radius, mass, orbital period, equilibrium temperature
- Stellar features such as star type and luminosity

Output:

- Habitability class (0 or 1)
- Habitability probability score indicating confidence in prediction

---

3. Dataset Preparation for Machine Learning

3.1 Dataset Description

The machine learning-ready dataset is stored at:

data/processed/exoplanet_preprocessed.csv

This dataset is the result of thorough preprocessing performed in Milestone-1, including missing value handling, encoding, and feature engineering.

3.2 Feature and Target Separation

The dataset is divided into:

- **Features (X):** Independent variables used for prediction
- **Target (y):** Habitability label

3.3 Train–Test Split

To evaluate model generalization, the data is split into:

- **80% Training data**
- **20% Testing data**

A fixed random_state is used to ensure reproducibility of results.

---

4. Baseline Model Selection

4.1 Purpose of Baseline Models

Baseline models are used to establish a **minimum performance benchmark**. They help determine whether advanced models genuinely improve prediction quality.

4.2 Baseline Models Used

- **Logistic Regression**
- **Decision Tree Classifier (with limited depth)**

4.3 Observations

- Logistic Regression provides a linear decision boundary
- Decision Trees capture simple non-linear relationships
- These models give initial accuracy and recall values for comparison

---

5. Primary Machine Learning Models

5.1 Random Forest Classifier

Random Forest is an ensemble learning method that combines multiple decision trees.

Advantages:

- Handles non-linear relationships effectively
- Robust to noise and overfitting
- Provides feature importance for interpretability

5.2 XGBoost Classifier

XGBoost (Extreme Gradient Boosting) is a powerful boosting-based algorithm.

Advantages:

- High performance on structured tabular data
- Handles complex feature interactions
- Efficient and scalable

Both models were trained and evaluated to identify the best-performing classifier.

---

6. Feature Scaling and Pipeline Design

6.1 Need for Feature Scaling

Since features have different units and ranges, scaling is required to:

- Ensure fair contribution of each feature

- Improve model convergence and stability

6.2 Pipeline Usage

Scikit-learn **Pipelines** were used to combine:

- Feature scaling
- Model training

This approach:

- Prevents data leakage
- Ensures clean and reproducible training
- Reflects industry best practices

---

7. Model Training Process

Each model was trained using the training dataset with default hyperparameters initially. The training process allows the model to learn patterns that differentiate habitable and non-habitable exoplanets.

After training, the models were saved using joblib for reuse without retraining.

---

8. Model Evaluation Metrics

To comprehensively evaluate model performance, the following metrics were used:

Metric	Description
Accuracy	Overall correctness of predictions
Precision	Reliability of habitable predictions
Recall	Ability to detect habitable planets
F1-score	Balance between precision and recall
ROC-AUC	Probability-based discrimination

Importance of Recall

In exoplanet studies, **recall is critical** because failing to identify a potentially habitable planet is more costly than a false positive.

---

9. Hyperparameter Tuning

9.1 Purpose

Hyperparameter tuning improves model performance and generalization.

9.2 Method Used

- **GridSearchCV** with cross-validation

9.3 Tuned Parameters

- Random Forest: n_estimators, max_depth
- XGBoost: learning_rate, max_depth, n_estimators

Tuning was performed only after baseline evaluation to avoid premature optimization.

---

10. Model Comparison and Final Selection

10.1 Selection Criteria

- High F1-score and Recall
- Stable performance on test data
- Minimal overfitting
- Interpretability

10.2 Final Model Selected

✅ **Random Forest Classifier**

Justification:

- Balanced precision and recall
- Better generalization
- Feature importance supports scientific interpretation

---

11. Habitability Scoring and Ranking

Instead of only class labels, the final model generates **habitability probability scores**. These scores are used to rank exoplanets from most to least habitable.

The ranked output is stored as:

data/processed/habitability_ranked.csv

---

12. Model Interpretability

Feature importance analysis revealed that:

- Planetary radius
- Equilibrium temperature
- Stellar luminosity
- Orbital characteristics

are the most influential factors affecting habitability predictions.

These results align with known astrophysical principles, increasing confidence in the model.

---

13. Conclusion

In this milestone, multiple machine learning models were trained and evaluated to predict exoplanet habitability. Through systematic comparison and tuning, the Random Forest classifier was selected as the final model due to its strong performance, robustness, and interpretability. The resulting system provides both classification and habitability ranking, making it suitable for scientific analysis and future extensions.