Overall Description of the Codebase

This project implements an **end-to-end machine learning workflow** to identify **potentially habitable exoplanets** using astrophysical data. The pipeline progresses from data preparation to baseline modeling, advanced model training, and finally a stacked ensemble optimized for high recall.

Section 1: Data Preparation & Balancing

This section handles **all preprocessing before modeling**.

- Loads a preprocessed dataset containing planetary and stellar features.
- Removes:
    - **Target leakage features** that directly encode habitability physics.
    - **Metadata/identifiers** that have no predictive value.
- Defines habitability_score as the regression target and also creates a **binary habitability label** using a fixed threshold.
- Performs a **stratified train–test split** to preserve class proportions.
- Reduces feature redundancy by:
    - Removing highly correlated features.
    - Applying domain-driven feature removal.
- Evaluates multicollinearity using **Variance Inflation Factor (VIF)**.
- Balances the training data using **SMOTE-ENN**, combining oversampling and noise removal.
- Saves all cleaned and balanced datasets for reuse.
- Visualizes class distributions before and after balancing.

Purpose:
Produce statistically clean, leakage-free, and balanced datasets suitable for reliable model training.

Section 2: Baseline Classification Models

This section establishes **reference performance benchmarks**.

- Loads the cleaned datasets from preprocessing.

- Converts the regression target into a binary classification label.
- Defines two baseline models:
    - Logistic Regression
    - Shallow Decision Tree
- Uses pipelines with feature scaling and class weighting.
- Trains each model and evaluates performance using:
    - Accuracy
    - Precision
    - Recall
    - F1-score
    - F2-score (recall-focused)
- Generates:
    - Classification reports
    - Confusion matrices
    - ROC curves
- Compares baseline models using recall-weighted metrics.

Purpose:
Provide interpretable and low-complexity benchmarks to justify more advanced models.

Section 3: Primary Machine Learning Models

This section trains and evaluates **high-performance models**.

- Uses Random Forest and XGBoost pipelines.
- Trains models on balanced training data.
- Saves trained models to disk for reuse and deployment.
- Generates predictions and probability scores.
- Evaluates models using recall-priority metrics, including F2-score.
- Produces mandatory:
    - Classification reports

    - Confusion matrices
    - ROC curve comparisons
- Summarizes model performance in a ranked comparison table.

Purpose:
Identify the strongest standalone models for habitability prediction.

Section 4: Stacked Ensemble & Threshold Optimization

This section builds the **final predictive system**.

- Constructs a **stacking ensemble** using:
    - XGBoost
    - Random Forest
      as base learners.
- Uses Logistic Regression as a meta-learner.
- Trains the ensemble within a standardized pipeline.
- Computes probability outputs on test data.
- Optimizes the classification threshold using the **precision–recall curve** to enforce high recall.
- Evaluates final predictions with:
    - Classification report
    - Confusion matrix
- Saves the optimized ensemble model for deployment.

Purpose:
Maximize detection of potentially habitable exoplanets while maintaining controlled precision.

Final Summary

The codebase implements a **modular, recall-focused machine learning system** with:

- Strong preprocessing discipline
- Clear baseline comparisons

- Advanced ensemble modeling
- Explicit decision-threshold control

The final output is a **deployable hybrid model** optimized to prioritize scientific discovery over raw accuracy.