Celestial Ai: Interactive Exoplanet Discovery with AI, 3D Exploration, and Gamified Learning

Team ExoPlantDetectors
Leader: Omar Abdel Motalb
Members: Abdel Rahman Abdel Motalb, Mohamed Samir, Mohand Fouda

NASA Space Apps Challenge 2025

1. High-Level Project Summary

Celestial Ai is an interactive machine learning system that classifies potential exoplanets using data from NASA's **Kepler Mission** [2]. Our platform employs an AI model trained on ten scientifically relevant Kepler features to predict whether a candidate is *CONFIRMED*, *CANDIDATE*, or *FALSE POSITIVE*.

The system combines a **trained machine learning pipeline** with a visually dynamic **Streamlit web application** [4], allowing users to:

- Upload and analyze datasets.
- Retrain the model with new data.
- Explore results in 3D and artistic galaxy modes.
- Play an interactive educational game.

Celestial Ai brings scientific rigor and creative engagement together to make exoplanet discovery accessible and inspiring.

2. Project Details

Architecture Overview

The system includes:

- **train_pipeline.py:** Prepares NASA data, trains the classifier, and exports the model artifacts.

- **app.py:** An advanced Streamlit interface that integrates model inference, retraining, visualization, and interactivity.

Why Streamlit?

Streamlit was chosen as the backbone of our user interface due to its ability to combine interactive components with data science pipelines seamlessly [4]. It allowed the team to translate machine learning outputs into engaging visualizations and create real-time interactivity without requiring complex front-end development. Through Streamlit, our model's predictions, 3D plots, and artistic galaxy visualizations become fully explorable for both researchers and learners. The framework also ensures rapid deployment and accessibility across platforms, empowering citizen scientists to engage with NASA data instantly.

Core ML Pipeline

Listing 1: Model Training Pipeline

```
gb_pipeline = Pipeline([
    ("preprocessor", preprocessor),
    ("classifier", HistGradientBoostingClassifier(
        learning_rate=0.05,
        max_iter=500,
        max_depth=6,
        min_samples_leaf=20,
        l2_regularization=1.0,
        early_stopping=True,
        validation_fraction=0.1,
        random_state=42,
        class_weight="balanced"
    ))
])
gb_pipeline.fit(X_train, y_train)
joblib.dump(gb_pipeline, "gb_pipeline.joblib")
```

This classifier achieved balanced accuracy across all three Kepler disposition categories and generalizes well to unseen candidates. The pipeline leverages the `scikit-learn` library [3] for modular and reproducible machine learning.

App Integration — Core Streamlit Logic

The Streamlit app dynamically loads the trained pipeline and label encoder, processes user input, and visualizes predictions. It also includes a parallax starfield and three orbiting planets that blur over content, a metaphorical nod to "worlds beyond."

Listing 2: Header and CSS Integration

```python
st.set_page_config(page_title="Celestial Ai", layout="wide")

def inject_custom_css():
    st.markdown("""
    <style>
    .stApp { background: linear-gradient(-45deg,#0a0118,#1a0033,#0f0c29,#24243e);
              animation: gradientShift 15s ease infinite; }
    @keyframes gradientShift {
        0%{background-position:0% 50%}
        50%{background-position:100% 50%}
        100%{background-position:0% 50%}
    }
    .title-wrap h1 {
        font-family:'Orbitron'; font-weight:900;
        background:linear-gradient(120deg,#00ffff,#ff00ff);
        -webkit-background-clip:text; -webkit-text-fill-color:transparent;
        text-shadow:0 0 40px rgba(0,255,255,0.55);
    }
    </style>
    """, unsafe_allow_html=True)

def show_header():
    st.markdown("""
        <div class="title-wrap">
            <h1>      Celestial Ai</h1>
            <p style="color:#9bb3ff;">AI Exoplanet Detection     NASA Space Apps 2025</p>
        </div>
    """, unsafe_allow_html=True)

inject_custom_css()
show_header()
```

Prediction Workflow

Listing 3: Candidate Classification Logic

```python
def classify_candidates(df, pipeline, encoder):
    X = df[SELECTED_FEATURES]
    y_pred = pipeline.predict(X)
    y_proba = pipeline.predict_proba(X)
    df["prediction"] = encoder.inverse_transform(y_pred)
    df["confidence"] = (y_proba.max(axis=1) * 100).round(2)
    return df
```

This function is used in both batch mode (for CSV uploads) and interactive single-candidate mode.

User-Focused Features

- Batch CSV analysis with automatic alignment and visualization.
- Retrain tab for continual learning from user data.
- 3D Exoplanet Explorer (radius, period, and signal-to-noise).
- Artistic Mode mapping planetary metrics to a stylized galaxy.
- "Guess the Exoplanet" game reinforcing classification understanding.

3. NASA Data

The model is trained on the publicly available **Kepler Mission Cumulative Dataset** from the NASA Exoplanet Archive [1].

Top 10 Features:

`koi_score, koi_fpflag_nt, koi_model_snr, koi_fpflag_co, koi_fpflag_ss, koi_fpflag_ec, koi_impact, koi_duration, koi_prad, koi_period`

Each feature was selected based on astrophysical relevance, ensuring interpretability and scientific validity.

4. Space Agency Partner & Other Data

All tools and resources used are open-source and freely accessible:

- Python, scikit-learn [3], Streamlit [4], Plotly, Pandas, NumPy
- Google Fonts (Orbitron, Space Grotesk)
- NASA Kepler Exoplanet Archive dataset [1]

5. Use of Artificial Intelligence (AI)

We used **ChatGPT (OpenAI)** for:

- Structuring and commenting Python code.
- Optimizing the Streamlit interface layout and CSS animations.

- Drafting and refining project documentation.

All model training, data analysis, and application logic were implemented independently by the team.

Conclusion

Celestial Ai merges scientific rigor and creative design to make exoplanet exploration engaging and transparent. By automating detection and visualizing discoveries interactively, it represents a forward step in how the public can interact with NASA's exoplanetary data.

References

[1] NASA Exoplanet Archive. (2025). *Kepler Mission Cumulative Dataset.* Retrieved from `https://exoplanetarchive.ipac.caltech.edu`

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

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

[4] Streamlit, Inc. (2025). *Streamlit — The fastest way to build data apps.* Retrieved from `https://streamlit.io`