An Iterative Machine Learning Approach to Identify Gravitational Anomalies through Time Dilation Measurement between Jupiter's Lagrange Points

Simulating Planet X Orbits and Validating Deep Learning Models in Gravitational Time Dilation Detection

Jonathan Nacionales

September 25, 2024

Abstract

This paper presents an iterative, fail-fast approach to detecting gravitational anomalies, such as hypothetical Planet X, by analyzing time dilation between satellites at Jupiter's L4 and L5 Lagrange points. Using simulated sinusoidal patterns of gravitational time dilation, we propose a method to train a machine learning model that can infer the most likely orbit of unknown gravitational sources. The paper outlines the step-by-step process of incrementally adding complexity—beginning with the Sun, Jupiter, and Planet X—before expanding the system to include other planets, the asteroid belt, and nearby galactic influences. By validating the model at each stage, we aim to develop a scalable and computationally efficient method for gravitational anomaly detection.

Concept

The maximum entropy principle states that given limited information, the macrostate with the highest entropy (i.e., the most probable) should be chosen as the best representation of a system's state. It's a method for making inferences based on incomplete information.

For a system with many degrees of freedom, the principle implies that when we observe a particular microstate, we should infer the macrostate that corresponds to the maximum possible entropy, given the constraints imposed by that microstate. This approach maximizes uncertainty and avoids assuming more structure than the available data supports.

This principle is rooted in the second law of thermodynamics.

Deep learning can be considered a tactic that leverages a form of inference consistent with the maximum entropy principle. It attempts to find the most likely model (macrostate) that fits the observed data (microstate) by balancing complexity and generalizability.

In deep learning, models often deal with high-dimensional data (analogous to systems with many degrees of freedom). The goal is to find patterns (analogous to macrostates) that best explain or predict the observed data (analogous to microstates) without assuming too much or too little about the underlying structure.

During training, a neural network optimizes its parameters to fit the data while generalizing to unseen data. This process can be viewed as maximizing the model's capacity to capture patterns (similar to maximizing entropy) while constrained by the data.

In our case:

- We have simulated data from satellites at Jupiter's L4 and L5 Lagrange points, which provide a **baseline** for measuring tidal variations.
- Each planet produces a unique **sinusoidal pattern** of tidal forces, allowing us to extract identifiable signatures from the data.
- By applying the maximum entropy principle, when we observe a certain tidal pattern (the microstate), we want to infer the most probable **gravitational source** or anomaly (macrostate) that could have produced this pattern, while accounting for known gravitational influences.

Given that multiple orbits and gravitational anomalies (like Planet X) could influence the data, we seek to maximize entropy by considering all possible explanations for the observed patterns and choosing the explanation that fits best while making the fewest assumptions.

Here, **deep learning** helps us build a model capable of learning these patterns from vast amounts of simulated data:

- **Training Phase**: We simulate many possible orbits for Planet X and record the **sinusoidal patterns** those orbits would generate in the tidal data, considering how Planet X's gravity would interact with other gravitational sources.
- The deep learning model learns to identify and extract these patterns, building a "map" of likely gravitational configurations.
- **Inference Phase**: Once trained, the model can be fed real data from satellites (L4 and L5) and infer which orbit (out of many possible simulated orbits) is most likely responsible for any observed gravitational anomalies.

This process mirrors the **maximum entropy principle** in that deep learning helps identify the macrostate (the most probable orbit of Planet X) that is most consistent with the observed data (the tidal patterns), based on the knowledge encoded from training simulations. By analyzing the patterns, deep learning enables the system to infer the most likely gravitational source while balancing complexity and uncertainty, just as the maximum entropy principle suggests.

Experiment

Here's one approach that can be used to train a machine learning model to infer the most probable orbit for a hypothetical Planet X from real data, based on the sinusoidal patterns produced by gravitational time dilation.

Step 1: Simulating the Data

First, we begin by simulating the solar system, including known planets and hypothetical orbits for Planet X. This simulation involves:

- **Known Planetary Orbits**: All known planets in the solar system (e.g., Earth, Jupiter, Mars) will contribute gravitational effects that influence time dilation between the two satellites. For each planet, we calculate the gravitational time dilation they would induce based on their mass, distance, and orbit.
- **Hypothetical Planet X**: We also introduce hypothetical orbits for Planet X. Each orbit configuration is varied (e.g., different distances from the Sun, eccentricities, and orbital periods), and the gravitational time dilation contribution from this hypothetical planet is recorded for the two satellites. By varying Planet X's parameters, we generate a wide range of potential gravitational signatures.
- **Time Dilation**: For each simulation, we calculate the time dilation between Satellite A (L4) and Satellite B (L5) as they exchange signals. This time dilation results from both gravitational effects and relative motion. Each planet, including Planet X, produces its own distinct sinusoidal pattern of time dilation over time, reflecting its contribution to the gravitational field.

Step 2: Generating Sinusoidal Patterns

The difference in time recorded by Satellite A and Satellite B produces sinusoidal patterns due to the periodic orbits of the planets. These sinusoidal patterns:

- Are influenced by the positions and masses of each planet.
- Vary depending on the strength of the gravitational pull and the relative positions of the satellites to the planets.
- Contain signatures from all gravitational sources, including hypothetical Planet X.

For each configuration of Planet X’s orbit, the combined sinusoidal pattern is recorded, which includes the known planets and the additional influence of Planet X.

Step 3: Training the Machine Learning Model

Next, a machine learning model is trained on these simulated patterns:

- **Input Data**: The input to the model is the sinusoidal pattern of time dilation between the two satellites, resulting from the gravitational effects of all the planets in the solar system.
- **Target Output**: The model's goal is to learn the relationship between the observed sinusoidal pattern and the configuration of Planet X's orbit. Specifically, the model is trained to predict the orbital parameters (e.g., distance, period, eccentricity) of Planet X that produce a given time dilation pattern.
- **Learning Process**: Through training, the model learns to identify subtle variations in the time dilation pattern that correspond to different orbital configurations of Planet X. It maps specific patterns of gravitational time dilation to the orbit that most likely produces them. This learning process involves recognizing the combined signatures of known planets and distinguishing them from the signature of Planet X.

Step 4: Real Data and Inference

Once the model is trained, we can apply it to real-time dilation data collected by Satellite A and Satellite B. When real data is fed into the model:

- The machine learning model compares the real sinusoidal pattern with the patterns it learned during training.
- Based on the similarities between the real data and the simulated patterns, the model can **infer the most probable orbit** for Planet X. It does this by finding the orbit configuration that best matches the time dilation pattern observed in the real data.
- Because the model has been trained on a wide range of hypothetical orbits, it can identify which configuration of Planet X's orbit is most likely responsible for any anomalies in the time dilation pattern.

Step 5: Maximizing Probability via Maximum Entropy

The machine learning model's goal is to select the **most probable orbit** that fits the observed data. This connects with the maximum entropy principle by focusing on the orbit that requires the fewest assumptions beyond the observed data and best explains the gravitational time dilation. The model balances the need to fit the data while not overfitting to noise or making unwarranted assumptions.

Strategy

We adopt a **"fail fast" strategy** to mitigate risk and minimize computational cost while incrementally testing and refining the model. The idea is to break down the problem into manageable, lower-risk steps, allowing us to validate the effectiveness of our model at each stage before proceeding to more complex simulations.

This is how our strategy works:

Step 1: Simplify the Simulation (Start Small)

Instead of simulating the entire solar system with all the planets, we begin with a simplified scenario that includes only the **Sun**, **Jupiter**, and **Planet X**.

These three bodies are chosen because they provide the core gravitational influences that would be significant in identifying time dilation patterns.

By isolating these key players, we reduce computational complexity and make it easier to control and analyze the results.

Step 2: Build and Train the Model Iteratively

We generate sinusoidal patterns of time dilation from this simplified system and train a machine learning model to detect the presence of Planet X based on this data.

During this initial phase, we cross-validate the model entirely on simulated data, ensuring the model can **"discover" Planet X** using only the gravitational influence of the Sun and Jupiter.

This step serves as a proof of concept: if the model can identify Planet X in a simplified scenario, it gives confidence that it can handle more complexity later.

If the model fails to detect Planet X at this stage, we can quickly halt and refine the approach without investing heavily in simulating all planets or additional astronomical bodies. This minimizes risk by allowing us to adjust early in the process before escalating the complexity.

Step 3: Gradual Introduction of Additional Planets

Once we validate the model in the simplified system, we incrementally introduce additional planets (e.g., Mars, Saturn) into the simulation.

After adding each new planet, we re-run the process of simulating time dilation, retraining the model, and validating its ability to detect Planet X amid more gravitational influences.

Each successful iteration further confirms the model's ability to differentiate between the gravitational signatures of known planets and that of Planet X, giving us confidence to proceed.

Step 4: Incremental Expansion of Complexity

As the model successfully handles the **known planets**, we can extend the simulation to include:

- The **asteroid belt**, which introduces additional gravitational noise.
- The **galactic center** and **nearby galaxies**, which contribute smaller but significant gravitational influences over time.

By continually cross-validating the model at each stage, we ensure that it can handle increasingly complex data without overwhelming the system or producing unreliable results.

Advantages of this Approach:

1. **Low-Risk, Iterative Testing**: By starting small and gradually increasing complexity, we minimize the computational cost and reduce the risk of failure, following the **fail fast** philosophy.
2. **Proof of Concept**: At each stage, we are building confidence in the model's capacity to "discover" Planet X, proving that the approach works before expanding to more resource-intensive simulations.
3. **Focused Learning**: Initially isolating key gravitational influences (Sun, Jupiter, Planet X) allows the model to develop a robust understanding of time dilation patterns in a simple context, making it easier to scale this understanding when new planets or bodies are added.
4. **Cross-Validation**: Using simulated data in controlled increments provides a built-in way to evaluate the model's performance, ensuring that it consistently delivers accurate predictions as new factors are introduced.

How It Fits Into the Overall Strategy:

- **Maximum Entropy Principle**: This approach aligns with the maximum entropy strategy because at each stage, we are identifying the **most probable macrostate** (the presence and orbit of Planet X) based on the data from a gradually increasing number of gravitational influences. By avoiding overfitting in complex simulations too early, we are ensuring that the model focuses on the data it has rather than unnecessary complexity.
- **Deep Learning Tactic**: The incremental learning process allows the deep learning model to build knowledge step by step. Each stage adds new patterns for the model to learn, without overwhelming it with excessive complexity early on. This tactic mirrors the fail fast approach in model training, as it ensures the system learns efficiently and correctly.
- **Scalability**: Once the model demonstrates success in simpler scenarios, it becomes more scalable. We are able to confidently simulate the entire solar system, introduce

other gravitational influences, and trust that the model has already been well-trained and validated at each stage of complexity.