KRITI'25

Martian Core Analysis

Organizer :
Equinox and CNA

High Prep
|600 Pts

Martian Core Analysis

Start : 10/01 End : 31/01

Welcome!

This is a month-long challenge where you will create a modular solution for analyzing extraterrestrial signals to detect potential anomalies and infer their origins. Inspired by NASA's planetary exploration challenges, your task is to integrate signal processing, machine learning, and advanced visualization techniques to simulate, analyze, and interpret these signals.

The focus is on building an efficient, interactive system capable of detecting patterns, filtering noise, and rendering a 3D representation of the source environment.

Good luck and have fun exploring the cosmos! 🌌

THEME

The task is to determine the size and state of the Martian core by analyzing seismic wave data. This involves the following modular steps:

Module 1: Understanding Seismic Wave Velocities

1.What are P-waves and S-waves? Briefly describe their properties and how they differ in terms of the materials they can propagate through.

2.Using the following equations for wave velocities, calculate the P-wave(v_p) and S-wave (s_v) velocities if the bulk modulus (K) is 2.5×10^{10} Pa, the shear modulus (G) is 1.0×10^{10} Pa, and the density (ρ) is 3000 kg/m^3

3. Derive P and S waves in a homogeneous, isotropic medium.

a)Start from this wave equation:

$$\rho \frac{\partial^2 u_i}{\partial t^2} = \frac{\partial \sigma_{ij}}{\partial x_j}$$

.

b) Use Hooke's law,

$$\sigma_{ij} = \lambda\delta_{ij}(\nabla \cdot \mathbf{u}) + \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$$

Derive:

$$\rho\frac{\partial^2 u_i}{\partial t^2} = (\lambda + \mu)\frac{\partial}{\partial x_i}(\nabla \cdot \mathbf{u}) + \mu\nabla^2 u_i$$

Where λ and μ are known as Lamé parameters

From this wave equation derive P and S waves where P are compressive waves and S are transverse waves.

Hint: You may split the displacement variable into shear and compressive components comprising of vector and scalar potentials, respectively. Get the final differential forms only and from these differential forms obtain the velocities of S and P waves.

Module 2:Identifying Shadow Zones:

1.Explain what is meant by the "S-wave shadow zone" and why its existence indicates the presence of a liquid core.

2.If S-waves are not detected on the opposite side of the planet from a seismic event, what can you infer about the core's state?

Module 3:Calculating the Core-Mantle Boundary (CMB):

1.Use Snell's Law to explain how P-waves refract at the core-mantle boundary.

2.Given the following data:

- v1=10 km/s (mantle P-wave velocity)
- v2=8 km/s (core P-wave velocity)
- $i=30^\circ$

Calculate the angle of refraction (r).

Module 4:Determining Core Radius:
The total radius of Mars is R=3390 km. The depth to the core-mantle boundary is d=560 km. Calculate the core radius (R_c) using the formula:

$R_c = R - d$

Verify whether this calculated core radius is consistent with seismic data observations.

Module 5:Verifying Core State:
Based on your calculations and observations from previous modules, summarize whether the Martian core is solid or liquid. Justify your answer using the absence of S-waves and the reduction in P-wave velocity within the core.

Module 6: Seismic Signal Processing Using ML

1. Explain how features such as amplitude, frequency, and phase shifts can be extracted from seismic waveforms.
2. Train a supervised ML model to classify seismic data as "shadow zone" or "non-shadow zone" to infer core properties.
3. Dataset:
    - Simulated seismic data with labeled shadow zones.
    - Real seismic data (if available) for validation.
    - Refere to the resources section in https://www.spaceappschallenge.org/nasa-space-apps-2024/challenges/seismic-detection-across-the-solar-system/?tab=resources

Module 7: Predicting Core Radius Using Regression

1. Train a regression model (e.g., Random Forest or Linear Regression) to predict the Martian core radius based on seismic wave velocities and density.
2. Dataset:
    - Input: P_v, S_v, density (ρ).
    - Output: Core radius (R_c).

Module 8: Anomaly Detection in Seismic Data

1. Use an unsupervised learning approach to identify anomalies in seismic wave data that could indicate subsurface geological features.
2. Algorithms: k-means clustering, DBSCAN, or autoencoders.
3. Dataset:
   - Synthetic seismic waveforms with known anomalies.

Module 9: Simulation of Wave Propagation with Neural Networks

1. Create a physics-informed neural network (PINN) to simulate P-wave and S-wave velocities across the Martian interior.
2. Use the PINN to estimate refraction angles and shadow zones for varying core-mantle boundary properties.

RULES

1. Sequential Modular Completion:
   a. The problem is divided into 9 modules, 5 theoretical and 4 ML. Participants must complete them all.
2. Programming Tools:
   a. You can use any programming language or framework, but Python (with libraries like NumPy, SciPy, TensorFlow, PyTorch, etc.) is recommended.
3. Physics and ML Integration:
   a. Participants are expected to combine traditional physics-based seismic analysis with modern machine learning techniques.
4. Deliverables:
   a. Well-documented code for each module.
   b. A report summarizing the results, challenges faced, and methodologies used.
   c. A final 3D visualization of the Martian core and seismic wave propagation.

5. Original Work:
    a. All work submitted must be original and created during the event. Any external resources or datasets must be credited appropriately.
6. GitHub Repository:
    a. Maintain a well-structured GitHub repository with regular commits, clear documentation, and step-by-step instructions for replicating your solution.
7. Datasets:
    a. You are allowed to use the provided dataset sources and any publicly available real seismic datasets for validation purposes.

JUDGING CRITERIA

1. Scientific Accuracy and Implementation (20%):
    - Are the calculations based on sound physical and mathematical principles?
    - Are the seismic properties and core properties (state, size, etc.) calculated accurately?
2. Integration of ML Techniques (30%):
    - Are the machine learning models implemented correctly and effectively?
    - Are appropriate algorithms chosen for classification, regression, and anomaly detection?
    - Is the performance of ML models well-justified with metrics?
3. 3D Rendering (30%):
    - Is the 3D visualization clear, accurate, and visually appealing?
    - Does it effectively communicate the insights gained from the analysis?
4. Documentation and Presentation (20%):
    - Is the GitHub repository well-organized, with clear documentation and instructions?
    - Is the final report comprehensive and easy to understand?