SPACE SCIENCE WITH PYTHON

# Space Science with Python — A Data Science Tutorial Series

Thomas Albin Follow
Jun 13, 2020 · 7 min read ★

Python is an amazing language for data science and machine learning and has a lot of great community driven Open Source libraries and projects. How can we use Python to explore and analyse the wonders and mysteries of Space?

Photo by Shot by Cerqueira on Unsplash

Near-Earth Objects, Meteors, ESA's Rosetta/Philae mission to a comet, the spacecraft Cassini exploring the ring worlds of Saturn … I worked in great projects during my academic studies and later as a doctorate student in the university. As a modern astrophysicist or space scientist, the major work is done in front of the screen: data exploration, data storage and maintenance, as well as the scientific analysis and publication of fascinating results and insights.

I learned a lot during these times and I am very grateful for that. Grateful for the opportunities and the time to explore cosmic wonders at academia's final frontier.

I used data scientific methods, machine learning and neural network architectures that can be developed and used by virtually anybody thanks to great publication sites, passionate users and a strong open source community. Now, I want to create a link between Data Science and Space Science. On Medium, Twitter, Reddit or at my Public Outreach presentations: People are amazed and fascinated by our cosmos! And I want to contribute something back for the community: A tutorial series that links ***Space Science with Python***.

## Overview

This article is an overview and provides short summaries of all articles that I publish here on Medium. This article will be updated continuously and provides a table of contents. All code examples are uploaded on my GitHub repository. Bookmark it to get future updates.

### 0. An Introduction

The very first article contains no coding parts. It was written and published as an initial introduction.

### 1. Setup and first steps

Setup of a virtual environment for Python. Installation of the NASA toolkit SPICE, respectively the Python Wrapper *spiceypy*. Explanation of some so-called SPICE kernels.

### 2. A look at Kepler's first law

Computation of the Solar System Barycentre with respect to the Sun (using SPICE). The tutorial shows that the gravitational centre of our Solar System moves within and outside the Sun. Consequently, the Sun "wobbles" around this common centre.

### 3. The Solar System centre

The outer gas giants (Jupiter, Saturn, Uranus and Neptune) are the major gravitational influencers in our Solar System. The computations and visualisations of miscellaneous angular parameters reveal that these planets are the main reason of the movement of the Solar System Barycentre as introduced in tutorial session 2.

### 4. The dance of Venus

April / May 2020: The Venus is visible to the naked eye in the evening; right after sunset our neighbor planet appears as a star above the horizon. Close angular distances with the Moon create a nice photo shoot. Here, the tutorial explains how to compute the angular distance between the Venus, Moon and Sun to determine optimal observation parameters (using SPICE).

### 5. Space maps

A tutorial that explains a core analysis and visualisation part of astronomy and space science: maps. SPICE and *matplotlib* are used to explain, compute, draw and interpret these maps. Further, two different reference systems are explained that are used in future sessions, too.

### 6. Around the Sun

SPICE provides so-called kernels that allow one to determine the position and velocity vector of planets, asteroids or spacecraft. The vector computation procedure is shown for the dwarf planet Ceres. Based on the position and velocity vector the corresponding orbital elements are calculated. Further, it is shown how close the asteroid 1997BQ passed by Earth in May 2020.

### 7. Comets — Visitors from afar

Comets are a remnant of the formation of our Solar System. Hundreds are known, documented and free available as a dataset. In this session, an SQLite database is created with data from the Minor Planet Center and some parameters are derived using SPICE. Further, the Great Comet Hale-Bopp is used as an example to derive positional information.

### 8. The Origin of Comets

Two types of comets are known: P and C Types. The different statistical variations are shown and discussed as well as their possible source of origin.

### 9. A Rendezvous with Jupiter

P Type comets are dynamically associated with Jupiter. This dynamical link is described with the Tisserand Parameter that is introduced and explained. A data scientific analysis of the distribution reveals the significant dynamical differences between C and P Type comets.

### 10. Supplements for Papers

This tutorial session is a supplementary article. It describes how one can create animations of the multi-dimensional Tisserand Parameter. These kind of visualisations help one to understand more easily multi-input functions. Online supplementary materials are often provided in publications to support the reader with additional information.

### 11. Did we observe everything?

Bias effects are present in virtually any statistical or data scientific research topic. Smaller, respectively fainter comets are more difficult to detect and their detectability scales with the distance and activitiy to the Sun.

### 12. A comet in 3 D

ESA's Rosetta/Philae mission explored the comet 67P/Churyumov–Gerasimenko from 2014 to 2016. During its 2 years mission the camera instruments took several images of the comet's core and derived a 3 D shape model. With the package *visvis* a Python renderer is programmed to interactively explore this icy world.

### 13. Turbulent times of a comet

There are several sources to predict the trajectory of a comet (here: 67P). We established an SQLite database with data from the Minor Planet center (see part 7) and we learned how to derive data from the SPICE kernels. Both data provide different and also non-static results that are described and compared here.

### 14. An Invisible Visitor

Part 13 has shown that the orbital elements of 67P from the SPICE kernels change for different Ephemeris times. One possible reason: 67P is a P Type and Jupiter-Family-

Comet (part 9) that is being influenced significantly by Jupiter. With the support of SPICE we can show the gravitational influence of the gas giant by computing a simple 2-body solution.

### 15. The Solar Orbiter and comet ATLAS

A few weeks ago (End May / Beginning of June 2020) ESA's Solar Orbiter crossed parts of the dust and ion tail of comet ATLAS. What kind of geometric requirements must be fulfilled *to be sure* that the spacecraft crossed the ion tail? Using SPICE and the most recent kernels of the spacecraft help us to answer this question.

### 16. Bright Dots in the Dark Sky

Brightness, flux density, irradiance, radiance … there are a lot of confusing words and definitions to describe light sources. In astronomy and space science one uses another definition: Magnitude. We create in this basic concept tutorial some functions that are used for future sessions (e.g., brightness computation of asteroids or meteors).

### 17. Uncertain Movement of an Asteroid

It is the 30th June 2020: Asteroid Day! Today we start with some asteroid related articles, beginning with an asteroid that passed by at a distance of 3 Lunar Distances: 2020 JX1. Computing the position of an asteroid is not as simple as shown in the past, we need the covariance matrix to determine a possible solution space of the asteroid's location.

### 18. Density Estimators in the Sky

2020 JX1 left the vicinity of our home planet! A distance of 3 Lunar Distances was small in cosmic scales, but large enough to miss us. The error-bars in the orbit solution space (see last session) propagate through the computation. Consequently, the sky coordinates of the asteroid are a solution space, too! A 2D Kernel Density Estimator will help us to determine an area of uncertainty in the sky, to answer the question: *Where could the asteroid be?*

### 19. A very bright Opposition

The brightness of asteroids can be computed by using the so called H-G magnitude function. An empirically determined equation that depends on the distance between the asteroid and the Earth and Sun, the phase angle, its absolute magnitude and the slope parameter. What are the special features of this equation? Let's see …

### 20. Ceres in the Sky

Tutorial #20 links several topics together: distance and phase angle determination, the apparent magnitude, sky coordinates and so on. The task: Visualising the path of Ceres in the sky for the year 2020 (considering its brightness trend, too). After this article we are good to go to start our first space science project about asteroids and Near-Earth Objects.

## Science Project #1

### 21. Space Science with Python — Asteroid Project (Part 1)

The first part of the project is an introduction into the Near-Earth Object (NEO) topic and does not include any coding yet. The structure of the upcoming weeks is being described.

### 22. Asteroid Project (Part 2) — Test Driven Development

Our project shall lead to a Python library that can be later used by amateur and professional astronomers and scientists alike. To ensure a credible and sustainable software package the library shall be written in a Test Driven Development (TDD) coding framework. What is TDD exactly? We will figure it out in this session.

### 23. Space Science with Python — Asteroid Project (Part 3)

A generic TDD example is provided in this step-by-step guide. Using a simple equation (computation of the enclosed angle between 2 vectors) we will try to find a solution based on example for all required computational steps.

Python Space Science Programming Data Science