

Exploratory Data Analysis and Data Visualization on Meteorite Landings on Earth Using Python

SAFIA HINA

Computer Teacher, International School, Shaikpet, Hyderabad.

DOI: 10.29322/IJSRP.13.04.2023.p13625

<http://dx.doi.org/10.29322/IJSRP.13.04.2023.p13625>

Paper Received Date: 6th March 2023

Paper Acceptance Date: 7th April 2023

Paper Publication Date: 15th April 2023

Abstract

During the initial evolving days of programming languages, Programmers performed basic to complex mathematical computations using programming languages, but with the emergence of competitive programming languages data can be analysed, cleaned and visualized for better understanding. In this paper, we will use NASA's meteorite landings dataset(Open source data from NASA's Meteoritical Society) and perform the data visualization to find the unique chondrites that have fallen on the earth surface. Long before the Earth and other planets existed, the solar system consisted of a protosun and an immense, dense cloud of gases and silicate/metal dust particles. This pre-planetary system is called the Proto Solar nebula (PSN). Over time the dust grains began to stick together (accrete) to form fluffy dust balls that accumulate (accrete). High-temperature events during the collapse of the PSN melted these dust balls into molten drops that quickly cooled (quenched) into a glass and tiny minerals. These millimetre-sized spherical objects resemble "fiery drops of rain" and are called chondrules. The meteorites that contain chondrules are called chondrites.

Keywords

Chondrites, Chondrules, Numpy arrays, Pandas series, Box-plots, Distribution-plot, Meteoroid, Meteor, Meteorites, Folium Maps, Annotated Count Plots.

1. INTRODUCTION

In the early years of the solar system formation, collisions among objects in space, e.g., dust particles, planetesimals, asteroids, planetoids and protoplanets, were very frequent. Evidence of these collisions is visible today on the Moon's surface, which shows thousands of ancient impacts that left craters ranging from centimetres to hundreds of kilometres in size. Even today, ejected materials from recent impacts on asteroids, Mars, and the Moon continue to rain down on the Earth as meteorites. The Meteoritical Society collects data on meteorites that have fallen to Earth from outer space. In this paper, detailed steps to perform data gathering on meteorites

landing, cleaning, analysis and visualizations are done using python programming language.

2. DATA REQUIREMENT

The dataset contains the following attributes of meteorites that have struck our planet:

- name: the name of the place where a meteorite was found or observed.
- id: a unique identifier for a meteorite.
- nametype: one of the following:
 - valid: a typical meteorite.
 - relict: a meteorite that has been highly degraded by the weather on Earth.
- reclass: the class of the meteorite; one of a large number of classes based on physical, chemical, and other characteristics.
- mass: the mass of the meteorite, in grams
- fall: whether the meteorite was seen falling, or was discovered after its impact; one of the following:
 - Fell: the meteorite's fall was observed.
 - Found: the meteorite's fall was not observed.
- year: the year the meteorite fell, or the year it was found.
- reclat: the latitude of the meteorite's landing.
- reclang: the longitude of the meteorite's landing.
- GeoLocation: a comma-separated tuple that combines reclat and reclang.

3. DATA COLLECTION

The data is collected from NASA's meteorite landings dataset(Open source data from NASA's Meteoritical Society) using Python's numpy and pandas module. It consists of 45,716 rows and 10 columns.

```

1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4
5 met_df = pd.read_csv('meteorite-landings/meteorite-landings.csv')
6 met_df
7

```

	name	id	nametype	recclass	mass	fall	year	reclat	reclong	GeoLocation
0	Aachen	1	Valid	L5	21.0	Fell	1880.0	50.77500	6.08333	(50.775000, 6.083330)
1	Aarhus	2	Valid	H6	720.0	Fell	1951.0	56.18333	10.23333	(56.183330, 10.233330)
2	Abee	6	Valid	EH4	107000.0	Fell	1952.0	54.21667	-113.00000	(54.216670, -113.000000)
3	Acapulco	10	Valid	Acapulcoite	1914.0	Fell	1976.0	16.88333	-99.90000	(16.883330, -99.900000)
4	Achiras	370	Valid	L6	780.0	Fell	1902.0	-33.16667	-64.95000	(-33.166670, -64.950000)
...
45711	Zillah 002	31356	Valid	Eucrite	172.0	Found	1990.0	29.03700	17.01850	(29.037000, 17.018500)
45712	Zinder	30409	Valid	Pallasite, ungrouped	46.0	Found	1999.0	13.78333	8.96667	(13.783330, 8.966670)
45713	Ziin	30410	Valid	H4	3.3	Found	1939.0	49.25000	17.66667	(49.250000, 17.666670)
45714	Zubkovsky	31357	Valid	L6	2167.0	Found	2003.0	49.78917	41.50460	(49.789170, 41.504600)
45715	Zulu Queen	30414	Valid	L3.7	200.0	Found	1976.0	33.98333	-115.68333	(33.983330, -115.683330)

45716 rows x 10 columns

Table 1. Data Retrieval on Meteorite Landings on Earth with 45716 rows and 10 columns

4. DATA RETRIEVAL AND DATA CLEANING

Firstly, we need to check the range of values in the year column in the Data Frame.

```

1 met_df['year'].describe()

```

```

count    45428.000000
mean     1991.772189
std       27.181247
min       301.000000
25%      1987.000000
50%      1998.000000
75%      2003.000000
max       2501.000000
Name: year, dtype: float64

```

Figure 1: Descriptive Statistics of year column

As you can see in the Figure 2, we have got the summary of descriptive statistics for the series. You can also spot some of the values against the 25%, 50% and 75% parameters. They are called percentiles (or quartiles).

Let's visualize the spread of the values in the year column in the Data Frame. For this, we will create a boxplot. It plots the minimum, first quartile, second quartile, third quartile and maximum values in the form of a box.

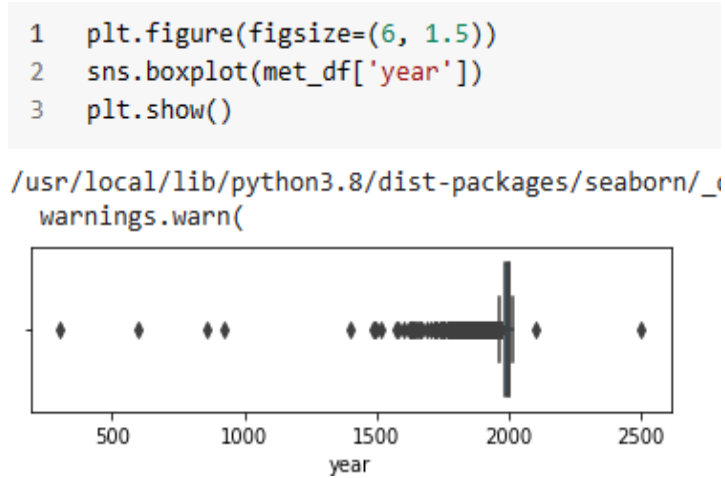


Figure 2: Boxplot for the year column of the met_df dataset

Figure 3, shows the boxplot for the year column in the Data Frame which is tightly squeezed. It shows that most of the year values are very close to each other. Specifically, most of the year values are roughly around the year 2000.

Similarly, you create a boxplot for the mass column in the Data Frame to observe the spread of the values.

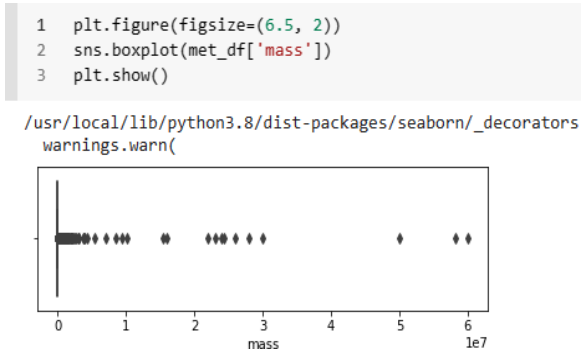


Figure 3: Boxplot for the mass column of the met_df dataset

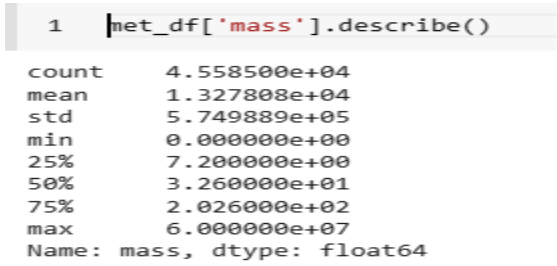


Fig.4 Descriptive Statistics of mass column of met_df dataset

```

found_valid_df = correct_lat_long_df[(correct_lat_long_df['fall'] == 'Found') & (correct_lat_long_df['nametype'] == 'Valid')]
found_valid_df
    
```

	name	id	nametype	recclass	mass	fall	year	reclat	reclong	GeoLocation
1108	Abajo	4	Valid	H5	331.00	Found	1982	26.80000	-105.41667	(26.800000, -105.416670)
1109	Abar al' Uj 001	51399	Valid	H3.8	194.34	Found	2008	22.72192	48.95937	(22.721920, 48.959370)
1110	Abbott	5	Valid	H3-6	21100.00	Found	1951	36.30000	-104.28333	(36.300000, -104.283330)
1111	Abermathy	7	Valid	L6	2914.00	Found	1941	33.85000	-101.80000	(33.850000, -101.800000)
1112	Abo	8	Valid	H	1.20	Found	1840	60.43333	22.30000	(60.433330, 22.300000)
...
45711	Zillah 002	31356	Valid	Eucrite	172.00	Found	1990	29.03700	17.01850	(29.037000, 17.018500)
45712	Zinder	30409	Valid	Pallasite, ungrouped	46.00	Found	1999	13.78333	8.96667	(13.783330, 8.966670)
45713	Ziin	30410	Valid	H4	3.30	Found	1939	49.25000	17.66667	(49.250000, 17.666670)
45714	Zubkovsky	31357	Valid	L6	2167.00	Found	2003	49.78917	41.50460	(49.789170, 41.504600)
45715	Zulu Queen	30414	Valid	L3.7	200.00	Found	1976	33.98333	-115.68333	(33.983330, -115.683330)

30871 rows x 10 columns

Table 2. Data about the meteorites which has been found(found) and in the good(valid) condition

Let's make a cartogram for the meteorites which are found in good condition (found_valid_df). According to the retrieved data as shown in Table 2. 30,871 meteorites were found in good condition. Hence, creating markers

for each of these meteorites is impractical. So, we can slice the found_valid_df Data Frame based on year. Let's estimate how many meteorites were found in good condition in between the years 2004 and 2007.

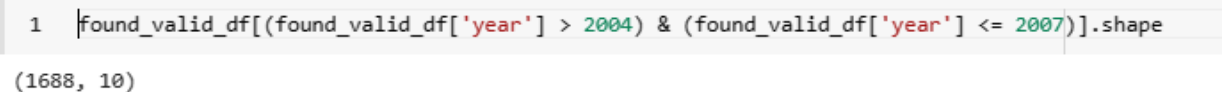


Figure 5. Data showing 1688 meteorites found in good(valid) condition in between the years 2005 and 2007.

So, 1688 meteorites were found in good condition from

2005 to 2007 (both inclusive). Now, let's create a cartogram for these meteorites.

As you can see in the Fig. 4, the min = 0 grams, 25% = 7.2 grams, 50% = 32.6 grams and 75% = 202.6 grams. These values are not so close to each other. But because the other mass values are very high (especially the maximum mass value is 60,000 kg which is a lot compared to most of the mass values), the boxplot is extremely squeezed.

There are few null values in the dataframe which are replaced by median values of that particular column of the dataset.

5. Data Visualization using Folium Maps

Our next task is to make a cartogram for the landing locations of the meteorites found on Earth. Meteorites are found in two conditions: the good and the withered condition meteorites. The condition of the meteorites is described in the "nametype" column. If the "nametype" value is Valid, then it means the condition of the meteorite found is good whereas if the "nametype" value is Relict, then it means the condition of the meteorite found is withered.

Firstly, let us get the data about the meteorites, which are in good condition(valid) and has been found(Found)

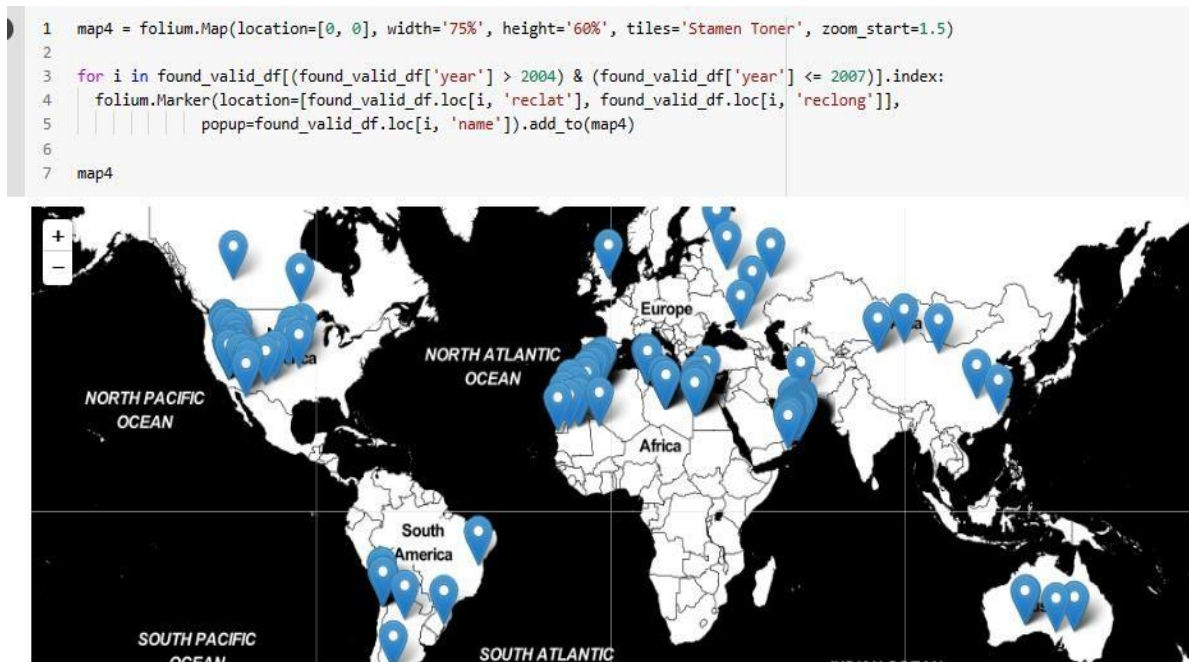


Figure 6: World map showing 1688 meteorites found in between the years 2005 and 2007(both inclusive)

As can be seen in the Figure 6 and Figure.7, 1688 meteorites are plotted on maps using the folium Marker and folium Circular markers. Multiple markers are overlapped due to space constraint and can be seen clearly when the map is zoomed. When the pointer is

pointed on the markers, name of meteorite can be read. Figure 7 shows the circular Markers. The larger the circle, it indicates the larger density of the meteorites found in that location which provides an easy way of analysing and visualising the data set.



Figure 7: Plotting 1688 meteorites with circular markers found in between the years 2005 and 2007(both inclusive)

6. Data visualisation using Count plots

A count plot is simply a type of bar graph which presents the number of occurrences of a value in a visual format. It helps in visualising the variation of values for a particular feature. For e.g., if you want to see graphically whether the number of meteorites falling over the years increasing or decreasing year-on-year basis, then you can choose to create a count plot. They are very easy to

create. Let us annotate the count plot for the meteorites fallen between the years 2000 and 2012. As can be evaluated from the graph the number of meteorites found in the year 2000 are 1506, in the year 2001 1266 meteorites have been found.

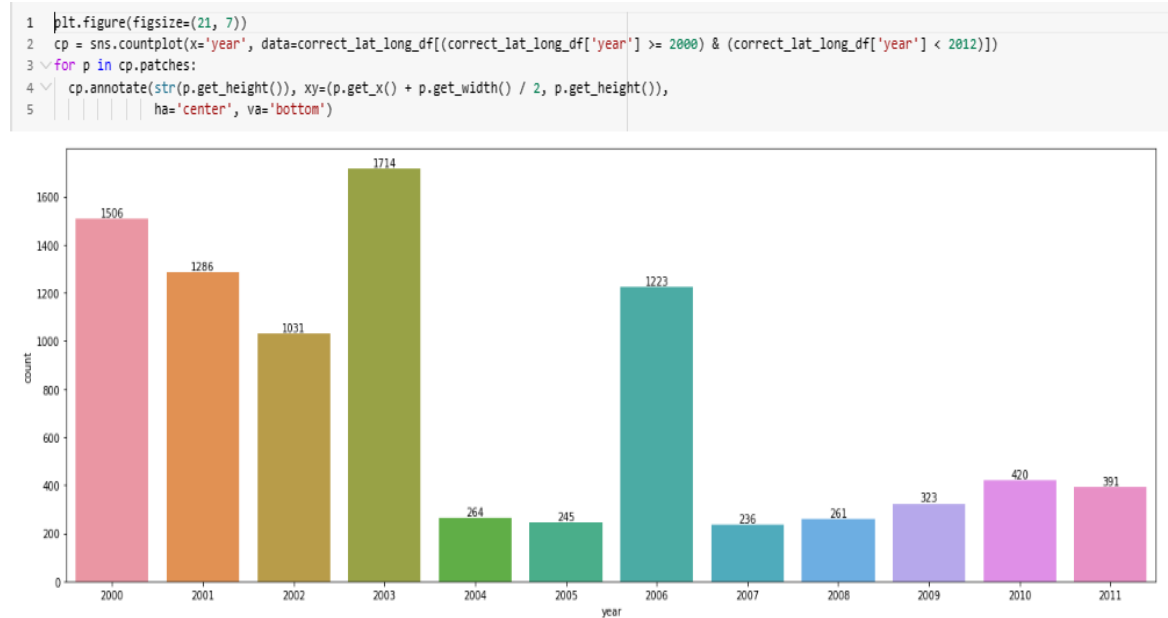


Figure 8: Count plot for the meteorites found in between the years 2000 and 2011(both inclusive)

For a histogram created using the seaborn module, the same process is followed.

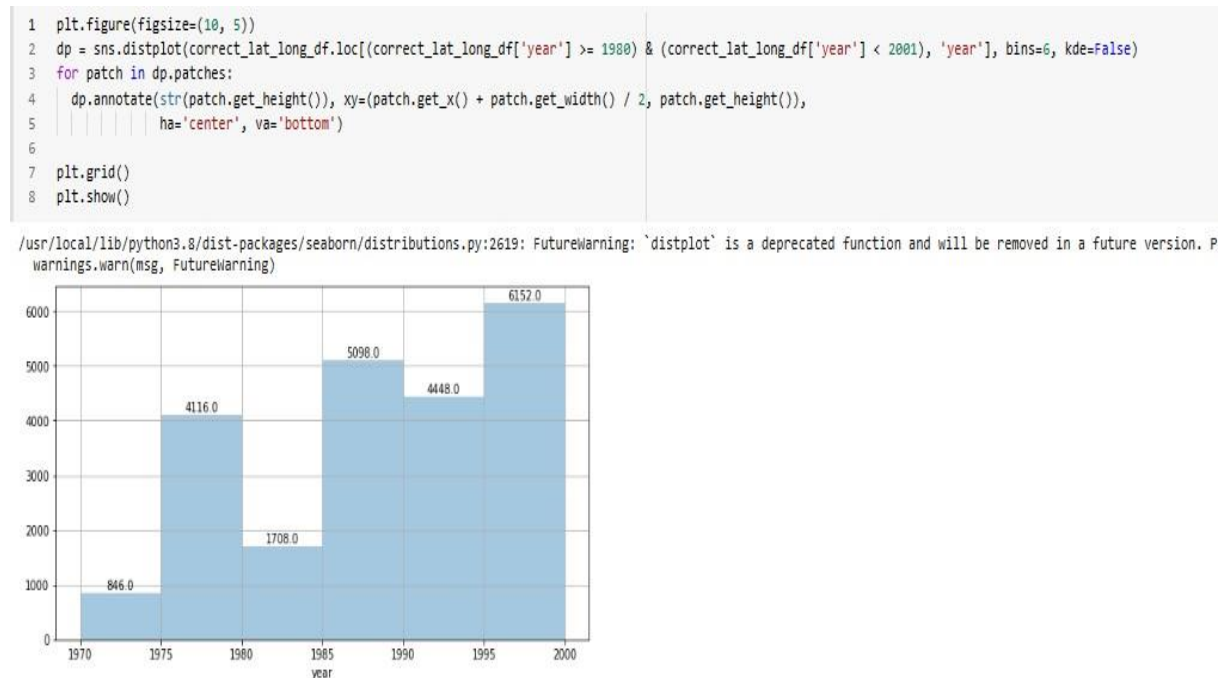


Figure 9 : Histogram for the meteorites found in between 1970 to 2000(both inclusive)

As per our calculation, at least 79% per cent of the meteorites fallen on Earth are ordinary chondrites. Our calculation is very close to Wikipedia's claim. There are

more chondrites in the Data Frame, but finding them out manually is a laborious task. If we display all the classes of meteorites, you will realize how messed-up the data is.

```
1 ord_chon = ['H3-H3,9', 'H4', 'H5', 'H6', 'H7', 'L3-L3,9', 'L4', 'L5', 'L6', 'L7', 'LL3-LL3,9', 'LL4', 'LL5', 'LL6', 'LL7', 'H', 'L', 'LL', 'H-H']
2 total = 0
3 for oc in ord_chon: # 'oc' stands for ordinary chondrites
4     for c in correct_lat_long_df['recclass']: # 'c' stands for class.
5         if oc == c:
6             total += 1
7
8 print("Total number of ordinary chondrites =", total,
9       "\nPercentage of number of ordinary chondrites =", round(total * 100 / correct_lat_long_df.shape[0], 2), '%')
```

Total number of ordinary chondrites = 25589
Percentage of number of ordinary chondrites = 79.88 %

Figure 10. Showing the total number of ordinary chondrites and percentage of number of ordinary chondrites

As you can see, the list containing the different classes of meteorites is very long. They are also annotated with some other type of meteorites. Hence, segregating the ordinary class of meteorites from the above list is quite difficult and a very time-consuming exercise. So, we will accept the fact that at least 79% of the meteorites that fall on Earth are ordinary chondrites

established that the study of chondrules allows us to understand the creation of our solar system. That's why the study of meteorites is important. If you ever get an ordinary chondrite, protect it. Don't lose it because you essentially will be carrying the history of the solar system in your hands and Visualization of data is made much more simpler with Python Data analysis and data mining.

Conclusion

So, the more ordinary chondrites fall on Earth, the more will be the availability of chondrules. We have already

Bibliography:

1. <https://data.nasa.gov/>
2. <https://www.geeksforgeeks.org/data-analysis-with-pyhton/>
3. <https://www.coursera.org/learn/data-analysis-with-python>