Meteorite Landing Data Visualization with Python

Ileana M. Atiles Gil de Lamadrid
Master of Engineering in Computer Engineering
Advisor: Jeffrey Duffany, Ph.D.
Electrical and Computer Engineering and Computer Science Department
Polytechnic University of Puerto Rico

Abstract *— For billions of years, the Earth has witnessed meteorite falls. Meteorites enter the Earth's atmosphere and land in more places than others. Recording on meteorites has been based on sighting and recovery. Many meteorites recoveries had been possible to Earth's conditions (such as cold temperatures) and human dedication. This project aims to corroborate data findings using Python programming language in a JupyterLab web environment. Data visualization of meteorites sample data will help to analyze where they land the most, which is the heaviest in mass and what year had the most landings. Zipf's law is considered as an approach to investigate if meteorites hitting the Earth follow any distribution pattern.*

Key Terms *— Data Visualization, JupyterLab, Meteorite, Python3.*

INTRODUCTION

A meteorite is defined as a piece of rock or other matter from space that has landed on Earth without being destroyed by the atmosphere. Meteorites are mainly classified due to its composition as Iron or Stony. Iron meteorites composition is high (~90%) in Iron and Nickel. Stony meteorites composition is higher with of other elements [1]. Meteorites have been falling on Earth for billions of years and meteorite recovery has been done for many centuries. Recording, collection and analysis of meteorite landing took popularity in the 1960s and 70s, but there is also have a high record of meteorite recovery during the 1800s [2]. Lately, there has been more effort to analyze and find patterns in scientific using data mining and big data analysis techniques.

BACKGROUND

The need for data analysis and visualization has been growing for the last years. As the world of the internet has grown, also has the amount of information. Social media has increased the amount of data to be analyzed. It's estimated that at least 1,200 petabytes of data are held by the big tech companies such as Google, Microsoft, Facebook and Amazon. The amount of data is not counting what is held internally by companies or other service cloud storage companies. In order to discover, solve problems and move our technology forward, statistical analysis and visualizations are needed. Data analysis and visualization is a trending domain in computer science, in both academia and business fields. The surge of using quicker and easier-to-program tools such as JupyterLab, R Studio, PowerBI, and Tableau are in high demand today for such purposes.

Statistical Methods

Several statistical methods and data visualization techniques have been applied to meteorite landing data using different tools and programming languages. In this case, Python programming language will be used to analyze such data. As per studies, meteorites tend to land more frequently in the southern high latitudes such as Antarctica than anywhere else [3] [4]. Meteorites that land on the Antarctica's ice cap are preserved for long periods of time under very clean conditions as they are carried toward the continents. Some scientists believe most of Antarctica's rock might be built of meteorites [5].

Zip's Law

In 1949, George Kingsley Zipf, an American linguist, found that frequency patterns of English words from a large text such a book, emerge if you keep track of them. There's a higher amount of words used than others. These are ranked by the number of occurrences in the text. A distribution can be obtained by using the frequency of each time the word is used vs its rank [6]. This behavior is a type of power distribution called Zipf's Law. It has been observed the same law applies to city size populations. The population of the city has been found to be inversely proportional to its rank. Population values were decreasing as their rank's. This power-law distribution appears in different domains such as size of earthquakes, frequency in use of words, and meteorites hitting the Earth [7] [8].

PROBLEM

This report aims to answer the following questions using Python programming:

- What latitude range does meteorites land the most?
- Where does the heavier meteorites land the most?
- Which years had the greatest number of landings?
- Which was the heaviest meteor to land between the 1970s until 2013? Where did its land?
- Does the data follow Zipf's law?

EQUIPMENT AND MATERIALS

A desktop computer was used to prepare the work environment. The technical specifications for hardware are the following:

- Desktop model: Acer Predator G3-710
- Processor: Intel® Core™ i7-7700 CPU @ 3600MHz
- Physical Memory: 32 GB
- Storage: one SSD (solid-state drive) 250GB and one HDD (hard-disk drive) 1 TB

The operative system used for this project was Windows 10 Home version. The additional software used were free and open source. An environment was created for JupyterLab using Anaconda Python distribution. JupyterLab is a web-based interactive development environment for Jupyter notebooks, code, and data. JupyterLab has many uses in data science and machine learning [9].

The environment was set up as a local server on the desktop computer. Anaconda was chosen to launch the Jupyter environment and handle the necessary packages to manage data and create graphics in Python version 3. Anaconda® is a package manager (conda), an environment manager, a Python/R data science distribution, and a collection of open source packages [10]. Versions for all used environments, distributions and packages are listed in Table 1.

Table 1
Environments, Distribution, Packages, and Libraries

Environments, Distribution, Packages and Libraries	Version Number
Anaconda	2019.10
Anaconda Navigator	1.9.7
JupyterLab	1.1.4
Jupyter Notebook	6.0.1
Conda	4.7.12
Python	3.7.4
Pandas	0.25.1
Matplotlib	3.1.1
NumPy	1.16.5
Seaborn	0.9.0

METHODOLOGY

Python programming was the approach used to handle data analysis and visualization. Python-based libraries Pandas and Matplotlib were mainly used. Pandas library was used to load, read and display data from the csv file into tabular data (data frame data type). Pandas also allowed to calculate basic statistical values and made easier to manage data as necessary to create the figures with Matplotlib later. Matplotlib library allowed to create all the graphic visualizations to analyze meteorite data. This library contains a set of functions familiar to MATLAB users. Other libraries such as NumPy and Seaborn were used to create a heatmap chart.

The dataset was public data available at DATA.GOV as a CSV (comma separated values) format file [11]. The mentioned website it's a U.S. Government's open data catalog that contains data, tools, and resources to conduct research, develop web and mobile applications, design data visualizations.

The file only contained data until 2013. Sample data from meteorite landing used had a window of 40 years. The data chosen to create the visualizations was from 1973 until 2013. The data had the following attributes: (1) meteorite id; (2) meteorite name; (3) mass in grams; (4) meteorite type; (5) latitude; (6) longitude; (7) geolocation; (8) date of fall; and (9) a category to determine if the meteorite was found or not.

The first step to organize the data to determine the approach to use was to look at the raw data to properly identify the formats, datatypes, etc. Data was cleaned to choose the sample. Date formats were changed for easy extracting of the landing year using Microsoft Excel. The analysis and visualization to perform chosen was to make comparison between meteorites mass, year frequency and where they land the most. The place of landing was defined by latitude: high, medium or low latitude at the South and North hemisphere. Antarctica is in the South hemisphere at latitude -75.2509766 and longitude -0.071389 [12].

RESULTS AND DISCUSSION

Several charts were obtained. The first chart created was a pin Earth map to locate the areas where the meteorites land the most. Unfortunately, so many points were heavily clustered that it was difficult to see the real amount. The chart was not immediately discarded but it was clear another approach to answer the questions had to be taken. A comparison made using a heatmap chart showed meteorites are greatly found in high latitudes at the South hemisphere such Antarctica. Figure 1 shows the South high latitudes darker which indicates the meteorites generally land on that part of the globe.

The top numbers for landings at the different latitude ranges are shown in Table 2. A higher tendency of landing around Antarctica is shown by comparing the data with the heatmap chart.

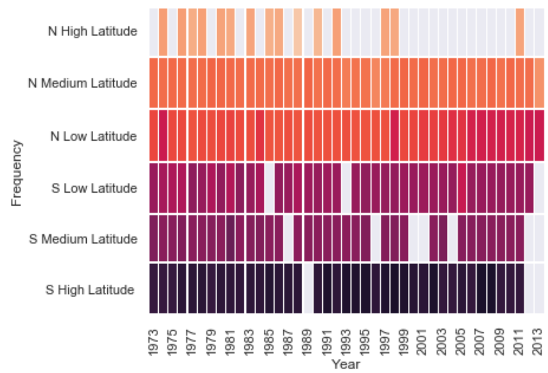

Figure 1
Latitude Frequency Heatmap

Table 2
Number of Landings per Year by Latitude Range

Latitude	Year	Landing Count
(-90.0, -60.0)	1979	3,018
	1988	2,274
	2003	1,422
(-60.0, -30.0)	1991	73
	1993	70
	1994	41
(-30.0, 0.0)	1998	1,667
	2009	1,200
	2007	814
(0.0, 30.0)	2001	711
	2000	603
	1999	397
(30.0, 60.0)	1999	101
	2000	93
	2001	71
(60.0, 90.0)	1981	3
	1974	1
	1976	1

The histogram in Figure 2 shows a tendency of meteorites with higher mass landing more in the North hemisphere rather than in the South.

The meteor with greatest mass landed in 1976 as shown in Figure 3. It weighted 4,000kg (4.4 US ton).

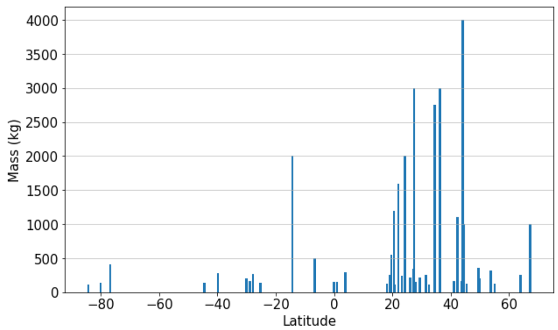

Figure 2
Top 50 Meteorite Landing by Mass (kg)

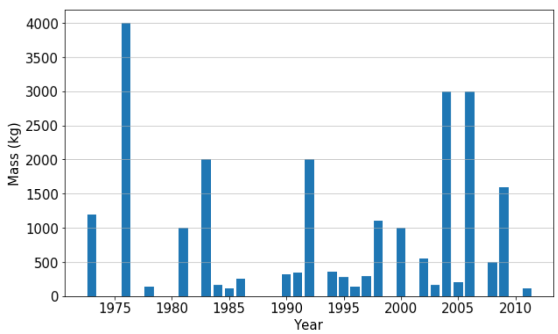

Figure 3
Top 50 Yearly Meteorite Landing by Mass (kg)

A XY-line plot in Figure 4 shows the single year with most findings and sightings corresponds to 1979 with 3,046 landings. Followed by 1988 with 2295 landings and 1998 with 2,151.

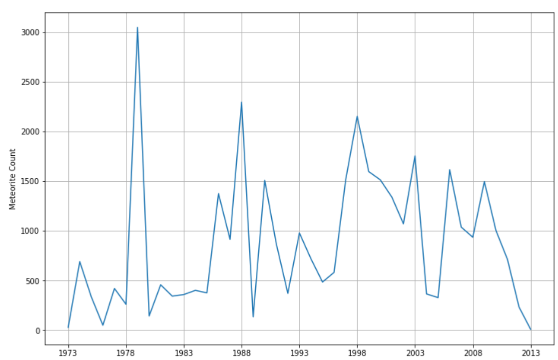

Figure 4
Meteorite Landing by Year

Basic statistical calculations were executed for the data using Pandas library. Obtained values can be seen at Table 3. The standard deviation (40850) suggests there's a big gap between the mass values of the data. The median (24.5) suggests there might be a higher number of low mass meteorites in the data.

Table 3
Basic Statistical Calculations on Mass (g)

Statistics	Mass (g)
Count	35777
Mean	1449
Standard deviation	40850
Median	24.5
Min	0
Max	4000000
Mode	1.3

The Zipf's law distribution analysis was applied to the data using the top 100 meteorites in mass (kg), to confirm the suggestions from basic statistics. The meteorite mass decreases as rank decreases. A small count of very heavy mass meteorites and a big count of lighter meteorites can be observed in Figure 5.

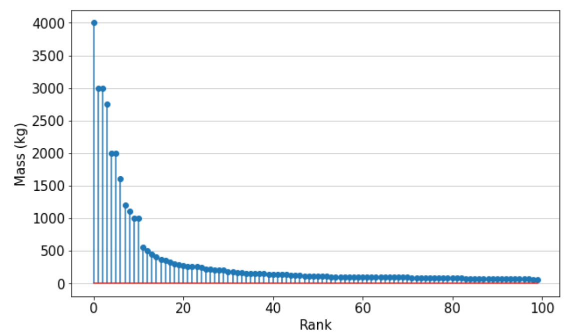

Figure 5
Zipf's Law for Top 100 Meteorites

Conclusion

The questions established at the beginning of the project were answered successfully. The year 1979 had the most landings. The range of latitude where meteorites land the most is in the Southern hemisphere, specifically in higher latitudes where Antarctica is located.

Meteorites with higher mass tend to land in the North hemisphere at medium latitudes. The meteorite with greatest mass from the sample landed at the North hemisphere at the latitude 44.05. The meteorite landed in 1976 and weighted 4,000kg (4.4 US ton). This meteorite was identified after looking at the complete dataset. The id is 12171 and the meteorite's name was Jilin. The name is based in the same city which is located at the latitude 44.05 and longitude 126.16667 in China. The result coincides with references found in the Meteoritical Bulletin Database [13].

Meteorite landing follows Zipf's distribution law meaning the meteorite landing is more frequent for those with less mass than high higher mass.

FUTURE WORK

The CSV data contained the exact geolocation of a meteorite landing but not the city or country name. Meteorite names are usually named after the city they land. This is not enough since cities with the same name exists in different countries. In addition, a meteorite could land more than once at the same city. The CSV data often concatenates more information after the meteorite's name when this happens. Hence, making programming and the data analysis extensive and prone to errors.

Using Google's Geocoding API (application programming interface) would make easier to determine the city and country of landing. The API has a reverse geocoding aid to convert geographic coordinates into a human-readable address. Other APIs from Google could be also used to create a better map without using Python's Matplotlib library, which it's very limited.

REFERENCES

[1] A. Krot, K. Keil, E. Scott, C. Goodrich and M. Weisberg, "1.05 Classification of Meteorites," *Treatise on Geochemistry*, vol. 1, pp. 83 - 128, 2007. [Online serial]. Available: https://books.google.com.pr/books?id=kYtksEUxw0oC&pg=PA83&lpg=PA83&dq=1.05+Classification+of+Meteorites&source=bl&ots=NH3E918ut5&sig=ACfU3U2EZO_xShe1U6lwLGg3CB5Cc8it-A&hl=en&sa=X&ved=2ahUKEwjqluuk7rrnAhUJrVkKHY_TCQ8Q6AEwCnoECAgQAQ#v=onepage&q&f=false. [Accessed: Jan. 30, 2020].

[2] G. J. H. McCall, A. Bowden and R. J. Howarth, *The History of Meteoritics and Key Meteorite Collections: Fireballs, Falls and Finds*, London: Geological Society of London, 2006.

[3] R. L. Korotev, "Some Meteorite Statistics," in *Washington University in St. Louis*, 2020. [Online]. Available: https://sites.wustl.edu/meteoritesite/items/some-meteorite-statistics/. [Accessed: December 12, 2019].

[4] Cornell University. (n. d.). *Do More Meteorites Fall at Low-latitudes...?* [Online]. Available: http://curious.astro.cornell.edu/our-solar-system/73-our-solar-system/comets-meteors-and-asteroids/meteors/301-do-more-meteorites-fall-at-low-latitudes-intermediate. [Accessed: December 12, 2019].

[5] W. Cassidy, E. Olsen and K. Yanai, "Antarctica: A Deep-Freeze Storehouse for Meteorites," in *Science*, vol. 198, no. 4318, pp. 727-731, 1977. [Online serial]. Available: https://science.sciencemag.org/content/198/4318/727. [Accessed: 30 January 2020].

[6] J. Tranquillo, in *MATLAB for Engineering and the Life Sciences*, Morgan & Claypool Publishers, 2011, pp. 97.

[7] P. M. Todd and G. Gigerenzer, "Rarity and Skewness in the World," in Ecological Rationality: Intelligence in the World, New York, *Oxford University Press*, 2012, pp. 381 - 382.

[8] Cornell University. (2016). *Zipf's Law for Cities – A Simple Explanation for Urban Populations* [Online]. Available: https://blogs.cornell.edu/info2040/2016/11/13/zipfs-law-for-cities-a-simple-explanation-for-urban-populations/. [Accessed: January 30, 2020].

[9] Proyect Jupyter. (n. d.). *Jupyter* [Online]. Available: https://jupyter.org/. [Accessed: November 20, 2019].

[10] Anaconda, Inc. (n. d.). *Anaconda* [Online]. Available: https://docs.anaconda.com/anaconda/. [Accessed: Nov. 20, 2019].

[11] NASA. (2019). *Meteorite Landings* [Online]. Available: https://catalog.data.gov/dataset/meteorite-landings. [Accessed: October 18, 2019].

[12] GeoDatos. (2020). *Geographic Coordinates of Antarctica* [Online]. Available: https://www.geodatos.net/en/coordinates/Antarctica. [Accessed: January 30, 2020].

[13] International Society for Meteoritics and Planetary Science. (2016). *The Meteoritical Society* [Online]. Available: https://www.lpi.usra.edu/meteor/?code=12171. [Accessed: January 30, 2019].