FINAL REPORT

Module: Strategic Thinking
Strategic Thinking Project

Project Group: DAB-FT-N921-004
Student Names and Student Numbers
Sylwia Kmiec, sba21659
Hannah Lange, sba21692
Joanne O'Shaughnessy, sba21686

TABLE OF CONTENTS

Table of Figures

Table of Figures

Table of Tables

Introduction

Earthquakes happen daily around the world. The US "National Earthquake Information Center now locates about 20,000 earthquakes around the globe each year, or approximately 55 per day" (USGS, 2021).

Even if we (luckily) do not hear about earthquakes and other natural disasters in the news daily, they are affecting millions of people from around the world and their choices on where and how to live. Most of the daily earthquakes can only be registered by highly tuned sensors and cannot be felt by humans at all. However, periodically when stronger earthquakes do occur, they have a high impact on local people and places.

In the following, we will present the report for the project that was motivated by the objective to analyze historical and current earthquake and natural disaster data and the possibilities for the application in the field of data analytics and machine learning.

Without any prior background knowledge in geology and seismology, we are mainly interested in exploring the field of earthquake predictions and forecasting from a data analytics point of view. This project's goal is not about using data analytics and its tools and models to predict future earthquakes, as we recognize that for the near future the prediction of earthquakes including the date and time as well as the location and magnitude of an earthquake will not be possible and claiming otherwise would be foolish (USGS, 2021).

For this project, the focus is on the analysis of earthquake and other natural disaster data to learn about possible correlations and the relationship and impact of earthquakes on other geological disasters. We would like to emphasize that the purpose is not to predict the occurrence of other natural disasters such as volcanic eruptions and tsunamis as a result of earthquakes. The intention instead is merely to add to the existing research and work done in this field from a data analytics point of view. The objective is the application of machine learning models to explore if their application can be a suitable instrument for learning more about the relationship and interconnectivity. As per the research done in this domain has shown an increase in volcanic activity in the local areas after earthquakes. The idea is to use machine learning to detect and reproduce those observed patterns in our data as well.

In the following, we will base the report for the phase our project on the Cross-Industry Standard Process for Data Mining (CRISP-DM) method. We will follow the life cycle of the CRISP-DM reference model (as displayed in Fig. 1), which includes the individual phases of our project, as well as the respective tasks associated with each phase of the project.

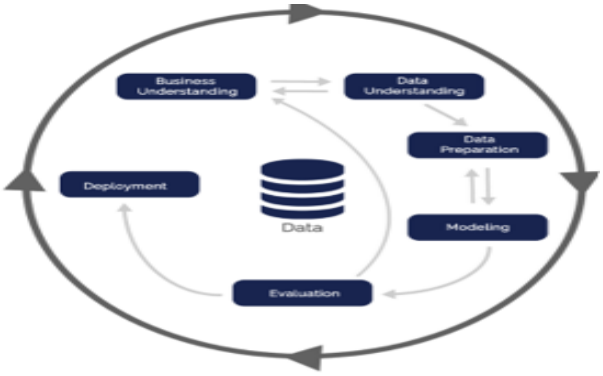


Figure 1: CRISP-DM model

We will present the steps taken within the individual tasks of each project. As this is a circular model, we have acknowledged that in the course of the project, we were going through several reiterations of the individual phases to evaluate progress and objectives based on our original project objective as well as on the results and further insights gained. Due to the nature of the project, we will also consider putting more emphasis on certain phases over others.

1. Business Understanding

Following the CRISP-DM model, we will start with the Business Understanding. However, before we can dive deeper into the business objectives and outline the project plan, we will give a brief overview of what has been produced in the previous phases of the project already, and then continue from this point onwards.

PROJECT HISTORY:

After the initial data collection process, while selecting and exploring the data, we realized that the specific research questions guiding phase one of our project were not feasible to build the project upon. The original intention and objective of this project was to look into earthquakes and their effects and implications with a geographic focus on the US and New Zealand. We were interested in exploring the following two research questions:

1. *Earthquakes and potential impact on housing prices in the local area*
2. *Earthquakes and the potential impact on tourism in the local area*

As main lessons learned from completing the first project phase, we decided that with the existing and available data and set project timeline and associated milestones, this would not be a feasible project and that we would have to go back and decide on a new and more practical research approach. Under the provision, that it needs to be possible for our data and research objective to be analyzed with Machine Learning models, we redirected our main research objective respectively. The projective objective and our research ambition pivoted towards the following question:

How likely it is that an earthquake leads to volcanic eruptions or tsunamis, and vice versa?

This has been further refined with the decision made to focus on the following:

How likely is it that an earthquake will lead to volcanic eruptions?

Instead of looking at the direct impact of an earthquake in a specified geographical area as originally planned, we decided on a more inclusive and global approach to the topic. Without any geographical restrictions, we want to look into potential correlations between the geographical location of plate tectonics, volcanic eruptions, and earthquakes. At the beginning of this new approach, we also considered including other natural disasters such as tsunamis, however after data exploration, we decided to solely

focus on volcano eruptions related to earthquakes. The idea as a new basis of the project is to use Machine Learning models to evaluate the relationship between the earthquake and volcano data.

Before we can dive deeper into the presentation of the available data, we have to state some general background information and research about plate tectonics, and tectonic movements as well as related consequences thereof, as this is the theoretic basis for our initial data exploration of the project.

PLATE TECTONICS

The below image shows the world's major tectonic plates including the African, Antarctic, Eurasian, Indo-Australian, North-American, South-American, and the Pacific Plates. The black line marks where the tectonic plates meet, the so-called plate boundaries or plate margins. Major plates generally cover an area of about 20 million km2 (National Geographic Society, 2021).

Not pictured are the surfaces minor as well as microplates, as they cover less surface area and sometimes even might be grouped together. These tectonic plates are in constant motion, and on average each plate can move between 2-5 cm per year (Explore Plate Tectonics, 2021).

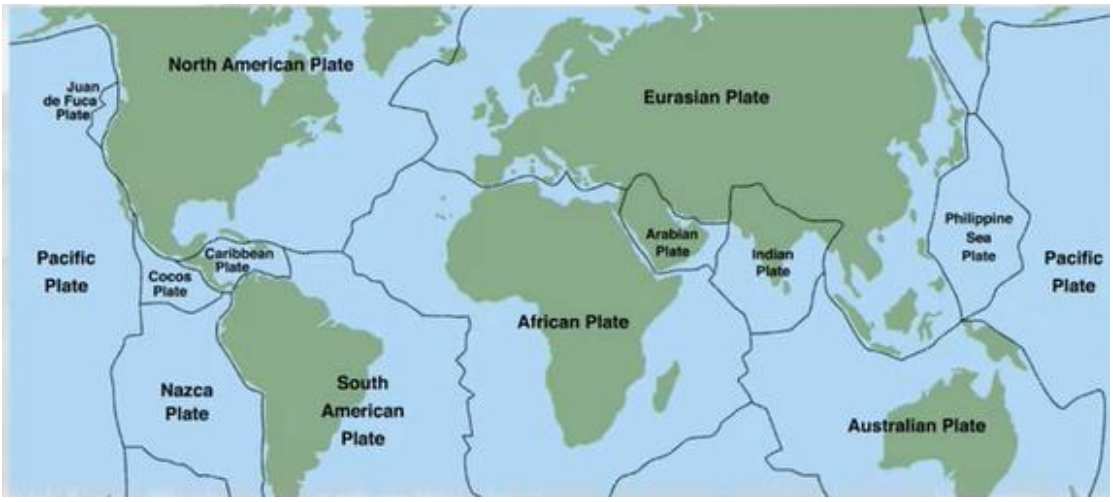


Figure 2: Tectonic Plates (Source: Sawe, 2020)

These tectonic plates are in constant motion, and on average each plate can move between 2-5 cm per year (Explore Plate Tectonics, 2021).

There are different possible directions of movement that can occur where the tectonic plates meet, the below image shows how the main tectonic plates can move against or towards each other.

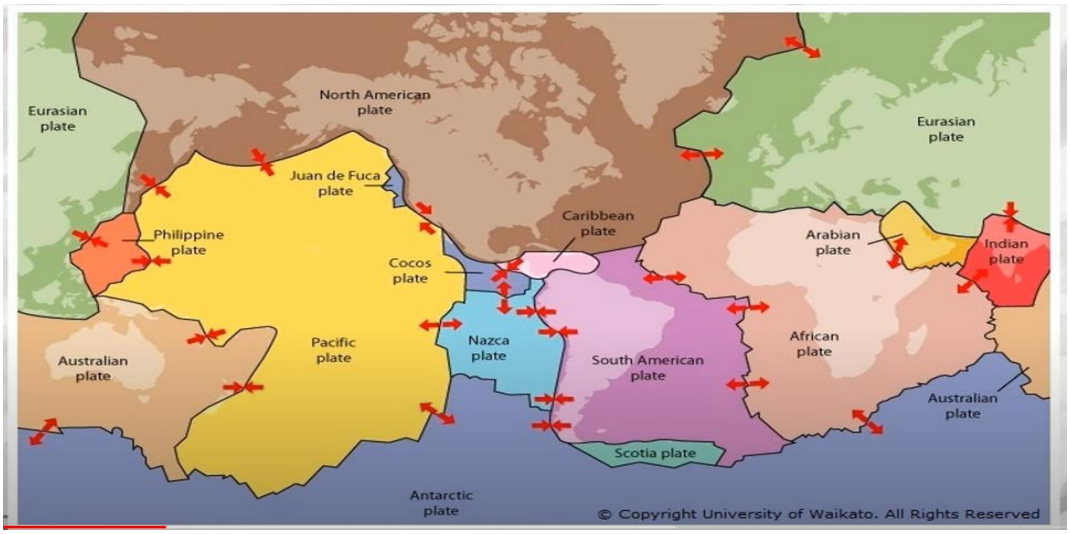


Figure 3: How plates move against each other (Source: SCI, 2007)

Along the plate boundaries, is where most tectonic hazards such as earthquakes and volcanic activities occur. E.g., there are about 500 active volcanoes in the world, and most of the volcanic eruptions occur on plate margins (Pacific Plate specifically) (National Geographic Society, 2021).

Research has shown, that there is a connection between plate tectonics, earthquakes, and volcanoes: Specifically, that earthquakes can cause volcanic eruptions, volcanic eruptions can cause earthquakes, and plate tectonic movements can lead to earthquakes. Besides, plate tectonics can lead to volcanic eruptions, and plate tectonic can also form volcanoes.

For this project, we are focusing on analyzing volcanic eruptions as a result of earthquakes only.

Other important theoretic definitions include looking into how earthquakes are being measured. Here we must look at the magnitude of earthquakes and the respective intensity. Magnitude refers to the energy that is being released by an earthquake and thereby the power that it has. Magnitude is measured on an exponential scale, where each magnitude is basically 33 times more powerful than the previous (PNSN, 2021).

As per the Richter magnitude scale (Fig. 4), we can see that level 5 earthquakes are categorized as “light” earthquakes on the scale. In real life, however, we would already be able to strongly feel the impact of such a magnitude 5 earthquakes. The likelihood of damage to buildings and infrastructure is high (Natural Hazard Portal, 2022).

RELATIONSHIP BETWEEN EARTHQUAKES AND VOLCANOES

Earthquakes can trigger volcanic eruptions through the severe movement of tectonic plates. Volcanic eruptions can have devastating economic and social consequences. To be able to set parameters for our machine learning analysis of the data, we looked into the existing body of research for both natural hazards.

Much of the work examining the relationship between earthquakes and volcanoes involves looking at volcano data for a long period following earthquakes, often focusing on specific regions.

Sawi and Manga (2018) report a 5-12% increase in the number of explosive volcanic eruptions in the two months to 2 years following earthquakes of magnitude 8 and higher, classified as major earthquakes.

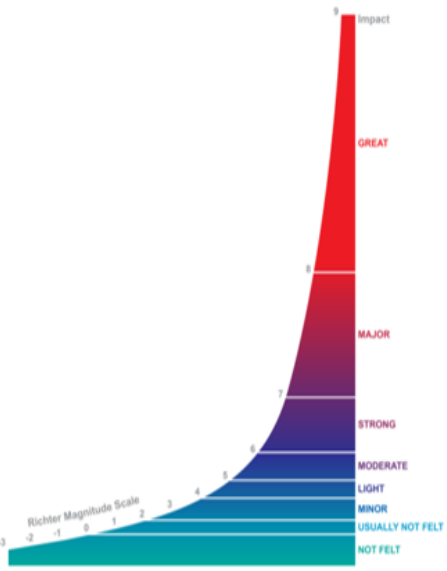


Figure 4: Richter Magnitude Scale (Source: British Geological Survey, 2021)

Marzocchi et al. (2002) modelled volcanic eruptions of VEI ≥ 5 induced by earthquakes of magnitude ≥ 7 in the last century following previous work which found a statistically significant influence occurring 0-5 and about 30 years after, at distances up to 1000 km from the earthquake epicenter,

Fault lengths of magnitude 6.5 to magnitude 9.0 earthquakes range from about 20 to more than 1000 km, respectively, and help inform distance selection to research (Hill et al. 2002).

Hill et al (2020) identified the response of 14 volcanoes to earthquakes of a magnitude of ≥ 8 that occurred between 2001 and 2011.

The parameters selected by Linde and Sacks (1998) were an eruption of VEI of ≥ 2 that occurred within 5 days and 800 km of an earthquake of magnitude ≥ 8, looking at data from 1500.

Lemarchand and Grasso (2008) used data from 1973 to 2005, selecting earthquakes of magnitude ≥ 4.8, and found an increase in volcanic eruptions on the day of the earthquake event.

Based on a review of the published research and limitations given our data set, it was concluded to select the parameters of 7 days following an earthquake of magnitude 5 and a distance of 1500 km. A balance was required between distance, magnitude, and timeline to ensure we would have matches to allow us to complete the machine learning component of our project.

It is noted that the parameters selected in this project are not ideal, and were more data available, selection of a higher magnitude, shorter distance, and over a longer timeframe would have been optimal. That said, we were able to find matches with the parameters selected. Many research papers focus on the volcano parameter VEI of two and higher, however, in this instance, we select all volcanoes due to sample size and many missing values, we focus on any volcano event rather than the explosivity of the earthquake, as measured with VEI.

2. Data Understanding

For the collection and selection of data to use for our project, one of our main priorities was to use data from a reliable and verifiable source. We have used the below-mentioned sources to collect data that is suitable for our project goal. As there is no "ready-to-use" dataset available for our project, we had to collect data from different sources and build the dataset from there.

The main data sources we used are listed below:

DATA COLLECTION

Earthquake dataset, source:

https://earthquake.usgs.gov/earthquakes/search/

Volcano's dataset, source:

https://volcano.si.edu/search_eruption.cfm

Tsunami's dataset, source:

https://www.ngdc.noaa.gov/hazel/view/hazards/tsunami/event-data?maxYear=2021&minYear=1960

Summary of the initial data collection process

For the initial data collection, we reviewed available data from the Atmospheric Administration (NOAA) for the number of records for earthquakes with a magnitude of 5 and higher. The table below summarizes the available data.

Table 1: NCEI Hazard Earthquake Results

	Years	Region	Magnitude	Records
1	All	All	All	6269
2	All	All	5+	4140
3	1960+	All	5+	2130
4	1960+	CA, USA	5+	41
5	1960+	NZ	5+	39

For earthquake data, the initial source was the Emergency Events Database (EM-DAT). We collected the available data for earthquakes between 1901 and 2020 respectively.

Table 2: EM-DAT Emergency Events Database

	Years	Region	Magnitude	Records
1	1901*-2020	All	All	1830
2	1960-2020	All	All	1475
3	1901-2020	All	5+	1414
4	1960-2020	All	5+	1153
5	1960-2020	CA, USA	5+	19
6	1960-2020	NZ	5+	8

*first record = 1901

Ultimately, we decided to move forward with data provided from the U.S. Geological Surveys earthquake catalog for the earthquake data. The volcano data was collected from the "Volcanos of the world" database provided by the Smithsonian Institution: National Museum of Natural History - Global Volcanism Program.

At the beginning of the project, the data collection process had its challenges. The first sources we considered for our project provided earthquake data already filtered. However, only certain earthquakes were reported. We have noticed this while performing Exploratory Data Analysis on the dataset. After having discovered a better suitable and more reliable data source, the next challenge was to find a *workaround* to getting the data reports downloaded. One major obstacle was, that there were limitations in getting the reports downloaded. Eventually, we decided to narrow the period of the data for our research to two years only. Even then, to get the data for two years from 2019-2020 inclusive we had to download more than 30 single reports, which then had to be combined manually. This resulted, as expected in issues related to data duplication and missing observations in the combined final data file.

To be able to employ Machine Learning analysis of the available data, we have opted to only select the data for the most recent two years 2019-2020. At the time of the data collection process in September 2021, the data for the full year was not yet available, and we have decided to exclude the 2021 data respectively.

DATA UNDERSTANDING SUMMARY:

Unfortunately, there is no dataset including all necessary information required for our analysis readily available. For the data understanding and exploration, we have collected datasets for each natural hazard individually as previously explained. In the following, we will present the main findings from the first round of Exploratory Data Analysis of the earthquake, volcano, and tsunami dataset sourced from the USGS and Smithsonian databases referenced above.

The objective here was to get a better insight into the collected data and its shape to be able to take further steps in merging to one dataset that can be used for the machine learning implementation of the project.

In the following, we will give an overview of the individual datasets first that will be used to build the dataset for the modeling later on.

1. Earthquakes

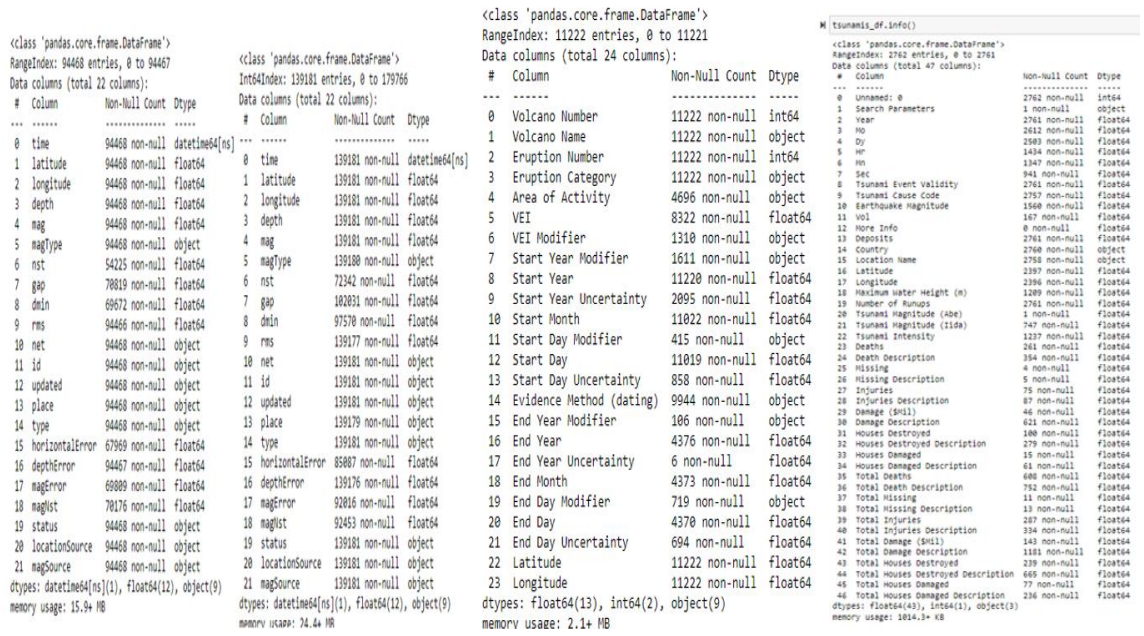
- The dataset including the earthquakes from 2019 and 2020 we have:
    - There are 94468 earthquakes recorded of the magnitude 1 and more in 2019.
    - There are 139181 earthquakes recorded of the magnitude 1 and more in 2020 (excluding duplicate rows).

2. Volcanos

The volcano dataset includes 11222 observations, across 24 variables.

3. Tsunamis

The Tsunami dataset includes 2767 observations, across 47 variables.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 94468 entries, 0 to 94467
Data columns (total 22 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   time            94468 non-null  datetime64[ns]
 1   latitude        94468 non-null  float64
 2   longitude       94468 non-null  float64
 3   depth           94468 non-null  float64
 4   mag             94468 non-null  float64
 5   magType         94468 non-null  object
 6   nst             54225 non-null  float64
 7   gap             70819 non-null  float64
 8   dmin            69672 non-null  float64
 9   rms             94466 non-null  float64
 10  net             94468 non-null  object
 11  id              94468 non-null  object
 12  updated         94468 non-null  object
 13  place           94468 non-null  object
 14  type            94468 non-null  object
 15  horizontalError 67969 non-null  float64
 16  depthError      94467 non-null  float64
 17  magError        69089 non-null  float64
 18  magNst          70176 non-null  float64
 19  status          94468 non-null  object
 20  locationSource  94468 non-null  object
 21  magSource       94468 non-null  object
dtypes: datetime64[ns](1), float64(12), object(9)
memory usage: 15.9+ MB
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 139181 entries, 0 to 179766
Data columns (total 22 columns):
 #   Column          Non-Null Count   Dtype
---  ------          --------------   -----
 0   time            139181 non-null  datetime64[ns]
 1   latitude        139181 non-null  float64
 2   longitude       139181 non-null  float64
 3   depth           139181 non-null  float64
 4   mag             139181 non-null  float64
 5   magType         139180 non-null  object
 6   nst             72342 non-null   float64
 7   gap             102031 non-null  float64
 8   dmin            97570 non-null   float64
 9   rms             139177 non-null  float64
 10  net             139181 non-null  object
 11  id              139181 non-null  object
 12  updated         139181 non-null  object
 13  place           139179 non-null  object
 14  type            139181 non-null  object
 15  horizontalError 85987 non-null   float64
 16  depthError      139176 non-null  float64
 17  magError        92016 non-null   float64
 18  magNst          92453 non-null   float64
 19  status          139181 non-null  object
 20  locationSource  139181 non-null  object
 21  magSource       139181 non-null  object
dtypes: datetime64[ns](1), float64(12), object(9)
memory usage: 24.4+ MB
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11222 entries, 0 to 11221
Data columns (total 24 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Volcano Number            11222 non-null  int64
 1   Volcano Name              11222 non-null  object
 2   Eruption Number           11222 non-null  int64
 3   Eruption Category         11222 non-null  object
 4   Area of Activity          4696 non-null   object
 5   VEI                       8322 non-null   float64
 6   VEI Modifier              1310 non-null   object
 7   Start Year Modifier       1611 non-null   object
 8   Start Year                11220 non-null  float64
 9   Start Year Uncertainty    2095 non-null   float64
 10  Start Month               11022 non-null  float64
 11  Start Day Modifier        415 non-null    object
 12  Start Day                 11019 non-null  float64
 13  Start Day Uncertainty     858 non-null    float64
 14  Evidence Method (dating)  9944 non-null   object
 15  End Year Modifier         106 non-null    object
 16  End Year                  4376 non-null   float64
 17  End Year Uncertainty      6 non-null      float64
 18  End Month                 4373 non-null   float64
 19  End Day Modifier          719 non-null    object
 20  End Day                   4370 non-null   float64
 21  End Day Uncertainty       694 non-null    float64
 22  Latitude                  11222 non-null  float64
 23  Longitude                 11222 non-null  float64
dtypes: float64(13), int64(2), object(9)
memory usage: 2.1+ MB
```

```
tsunamis_df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2762 entries, 0 to 2761
Data columns (total 47 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Unnamed: 0                      2762 non-null   int64
 1   Search Parameters               1 non-null      object
 2   Year                            2761 non-null   float64
 3   Mo                              2612 non-null   float64
 4   Dy                              2583 non-null   float64
 5   Hr                              1434 non-null   float64
 6   Mn                              1347 non-null   float64
 7   Sec                             941 non-null    float64
 8   Tsunami Event Validity          2761 non-null   float64
 9   Tsunami Cause Code              2757 non-null   float64
 10  Earthquake Magnitude            1560 non-null   float64
 11  Vol                             167 non-null    float64
 12  More Info                       0 non-null      float64
 13  Deposits                        2761 non-null   float64
 14  Country                         2760 non-null   object
 15  Location Name                   2758 non-null   object
 16  Latitude                        2397 non-null   float64
 17  Longitude                       2396 non-null   float64
 18  Maximum Water Height (m)        1209 non-null   float64
 19  Number of Runups                2761 non-null   float64
 20  Tsunami Magnitude (Abe)         1 non-null      float64
 21  Tsunami Magnitude (Iida)        747 non-null    float64
 22  Tsunami Intensity               1237 non-null   float64
 23  Deaths                          261 non-null    float64
 24  Death Description               354 non-null    float64
 25  Missing                         4 non-null      float64
 26  Missing Description             5 non-null      float64
 27  Injuries                        75 non-null     float64
 28  Injuries Description            87 non-null     float64
 29  Damage ($Mil)                   46 non-null     float64
 30  Damage Description              621 non-null    float64
 31  Houses Destroyed                100 non-null    float64
 32  Houses Destroyed Description    279 non-null    float64
 33  Houses Damaged                  15 non-null     float64
 34  Houses Damaged Description      61 non-null     float64
 35  Total Deaths                    688 non-null    float64
 36  Total Death Description         752 non-null    float64
 37  Total Missing                   11 non-null     float64
 38  Total Missing Description       13 non-null     float64
 39  Total Injuries                  287 non-null    float64
 40  Total Injuries Description      334 non-null    float64
 41  Total Damage ($Mil)             143 non-null    float64
 42  Total Damage Description        1181 non-null   float64
 43  Total Houses Destroyed          239 non-null    float64
 44  Total Houses Destroyed Description  665 non-null  float64
 45  Total Houses Damaged            77 non-null     float64
 46  Total Houses Damaged Description  236 non-null  float64
dtypes: float64(43), int64(1), object(3)
memory usage: 1014.1+ KB
```

Figure 5: Variables of the original earthquake, volcano, and tsunamis datasets

3. Data Preparation

The data preparation process of our project has been the most time-consuming during the project. When selecting which data to use for the project we had to keep in mind that not all available data was necessarily relevant to our project objective. Selecting the relevant data and cleaning the dataset(s) has been a high priority.

Additionally, had to ensure we have a “clean” dataset of good quality to be able to implement Machine Learning models.

The following initial steps have been taken to prepare the datasets for potential usage in the implementation of Machine Learning models:

Data Construction:

Earthquakes:

- for analysis purposes, the column 'time' has to be brought to the shape: YYYY-MM-DD

- year, month, and day then each needed to be in separate columns, therefore the time column was split, and new columns were created accordingly: year, month, and day. This was done, when comparing to the format of the dataset for tsunamis and volcanos, this format was applied, and our objective is to prepare the individual datasets as much as possible to make it feasible to combine them together.
- We constructed a new dataframe combining the cleaned and pre-processed 2019 and 2020 datasets together with the concat method
- For the years 19/20 combined we have a total of 233649 observations, across 22 variables.

Volcanos:

- The following variables have been removed from the dataset as they would not add any additional information for our analysis:
    - 1. Area of Activity (missing values, not known information)
    - 2. VEI
    - 3. VEI Modifier
    - 4. Start Year Modifier
    - 5. Start Year Uncertainty (re historical dates)
    - 6. Start Day Modifier
    - 7. Start Day Uncertainty (re historical dates)
    - 8. End Year Modifier
    - 9. End Year Uncertainty
    - 10. End Day Modifier, 11. End Day Uncertainty
    - 11. End Day Uncertainty
    - 12. Evidence Method (dating)
    - 13. End Year
    - 14. End Month
    - 15. End Day
- From the available data we selected only events that occurred in the years 2019-2020, this leaves us with 54 entries and 13 variables in total.
- VEI index is a measure of the explosivity index, it was decided to remove this variable as our project focuses on the relationship between earthquakes of defined parameters and all volcanoes and so this variable was dropped. In addition, there were over 25% missing values for VEI Index.
- All columns with a modifier in the title are not relevant as are minor mathematical adjustments to the associated column.

T*sunamis:*

- For the tsunami dataset, we kept only columns related to date, country, location, event validly, event cause, tsunami intensity, and earthquake magnitude and created a new data frame only containing those features.

- Thereby, we have decided to exclude the following variables from our analysis for now:

  Unnamed, Search Parameters, Mo, Dy, Hr, Mn, Sec, Vol, More Info, Deposits, Maximum Water Height (m), Number of Runups, Tsunami Magnitude (Abe), Tsunami Magnitude (Iida), Tsunami Intensity, Deaths, Death Description, Missing, Missing Description, Injuries, Injuries Description, Damage ($Mil), Damage Description, Houses Destroyed, Houses Destroyed Description, Houses Damaged, Houses Damaged Description, Total Deaths, Total Death Description, Total Missing, Total Missing Description, Total Injuries, Total Injuries Description, Total Damage ($Mil), Total Damage Description, Total Houses Destroyed, Total Houses Destroyed Description, Total Houses Damaged, Total Houses Damaged Description

DATA CLEANING PROCESS

We conducted several iterations of data cleaning. First on our individual datasets, as described below, and once we had created a new dataset, we did another iteration of data cleaning. We were focusing on missing values and duplicated data.

Missing Values

Earthquakes

- After the datasets for 2019 and 2020 have been combined, we checked for missing values
- Columns with the most missing values in the data frame are not imperative for the analysis, and these columns were dropped from the data frame (see figure 6)
- Additionally, two rows with missing values were dropped from the data frame

```
date                  0
latitude              0
longitude             0
depth                 0
mag                   0
magType               1
nst               94538
gap               56551
dmin              62003
rms                   6
net                   0
id                    0
updated               0
place                 2
type                  0
horizontalError   76005
depthError            6
magError          67199
magNst            66465
status                0
locationSource        0
magSource             0
dtype: int64
```

Figure 6: List of dropped variables from the earthquake data frame

Volcanos

- Dataset was filtered for volcano occurrences in 2019 and 2020 only – 54 records
- Any missing data under the columns “End Year, Month, and Day” can be replaced with the last date of 2020 (missing dates might be due to the fact that the volcano is still active, and/ or the eruption is not completed)

Tsunamis

- After filtering the dataset for the observation from 2019 and 2020 respectively, there was only one column with missing values which was then filled with the new value: 'missing'

Duplicated data

Earthquakes

- No duplicate data for the 2019 dataset
- In the 2020 dataset, there were duplicated data points that were created by the manual junction of the individual datasets and consequently had to be dropped from the data frame

For the volcanoes and tsunamis dataset, there was no further maintenance concerning duplicated data required.

SUMMARY REPORT FIRST PHASE OF DATA PREPARATION

Earthquakes

After completion of the first EDA and data cleaning process, we had 216502 rows over 15 variables as follows:

date, latitude, longitude, depth, mag, magType, net, id, updated, place, type, status, locationSource, magSource.

Volcanoes

The Volcano data frame consists of 54 observations in the Volcano data set for the period 2019 – 2020, and 10 variables as shown below.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 54 entries, 22 to 75
Data columns (total 10 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Volcano Number     54 non-null     int64
 1   Volcano Name       54 non-null     object
 2   Eruption Number    54 non-null     int64
 3   Eruption Category  54 non-null     object
 4   year               54 non-null     float64
 5   month              54 non-null     float64
 6   day                54 non-null     float64
 7   Latitude           54 non-null     float64
 8   Longitude          54 non-null     float64
 9   date               54 non-null     datetime64[ns]
dtypes: datetime64[ns](1), float64(5), int64(2), object(2)
memory usage: 4.6+ KB
```

Figure 7: List of variables for volcano dataframe 2019-2020

Tsunamis

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 22 entries, 0 to 21
Data columns (total 14 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Unnamed: 0              22 non-null     int64
 1   year                    22 non-null     float64
 2   month                   22 non-null     float64
 3   day                     22 non-null     float64
 4   Tsunami Event Validity  22 non-null     float64
 5   Tsunami Cause Code      22 non-null     float64
 6   Earthquake Magnitude    15 non-null     float64
 7   Country                 22 non-null     object
 8   Location Name           22 non-null     object
 9   Latitude                22 non-null     float64
 10  Longitude               22 non-null     float64
 11  Date                    22 non-null     datetime64[ns]
 12  MercatorX               22 non-null     float64
 13  MercatorY               22 non-null     float64
dtypes: datetime64[ns](1), float64(10), int64(1), object(2)
memory usage: 2.5+ KB
```

Figure 8: List of variables for tsunamis dataframe 2019-2020

The shape of the new data frame: 22 observations and 14 features.

As the number of observations for our pre-selected time period is very small, we decided to drop the tsunami dataset from our machine learning modeling plans and focus on earthquakes and volcanoes only. Due to the workload involved in creating the earthquake dataset and the significant challenges faced with labelling, we needed to narrow our focus and define our problem statement. Based on published research in this space, the logical choice was to discard tsunamis and focus on the occurrences of volcanoes following an earthquake. The low number of tsunamis during our selected time frame would mean we would have low confidence in any potential findings.

VISUALIZATIONS

One main focus during the initial data understanding as well as the data preparation process was to produce visualizations of our data. As we were working with geospatial data, initially we have mainly used the Folium library as well as bokeh plots.

General observations:

- The majority of earthquakes occur in lines along tectonic plate boundaries
- Generally: larger number of earthquakes in Asia and along North and South America.
- A smaller number of earthquakes in Africa and Australia which do not occur on plate boundaries

This can be compared to the outline of the major plates in Fig. 2.

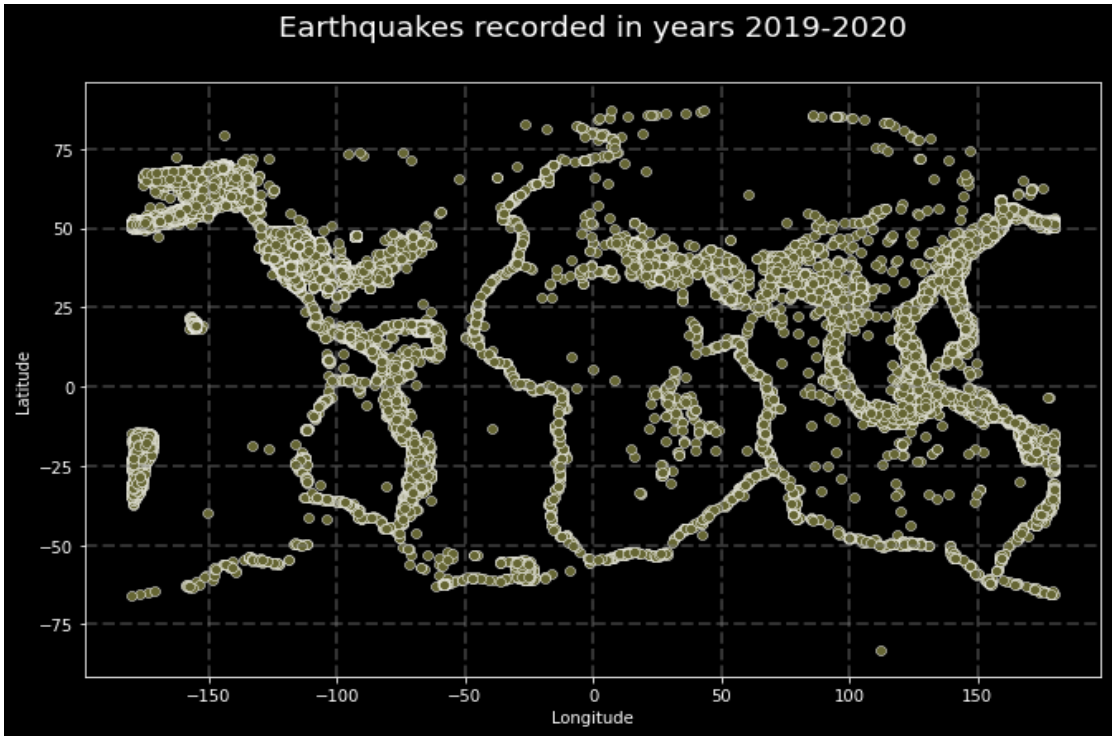


Figure 9: Scatterplot Earthquake data 2019-2020

Additionally, we also checked the distribution of earthquakes in our chosen period for analysis by magnitude. As per Fig. 10, below we can see that the highest number of earthquakes are of magnitude 1 (96583 earthquakes). For magnitude 5+, which is the magnitude that has been selected for this project, we have a total of 13679 earthquakes.

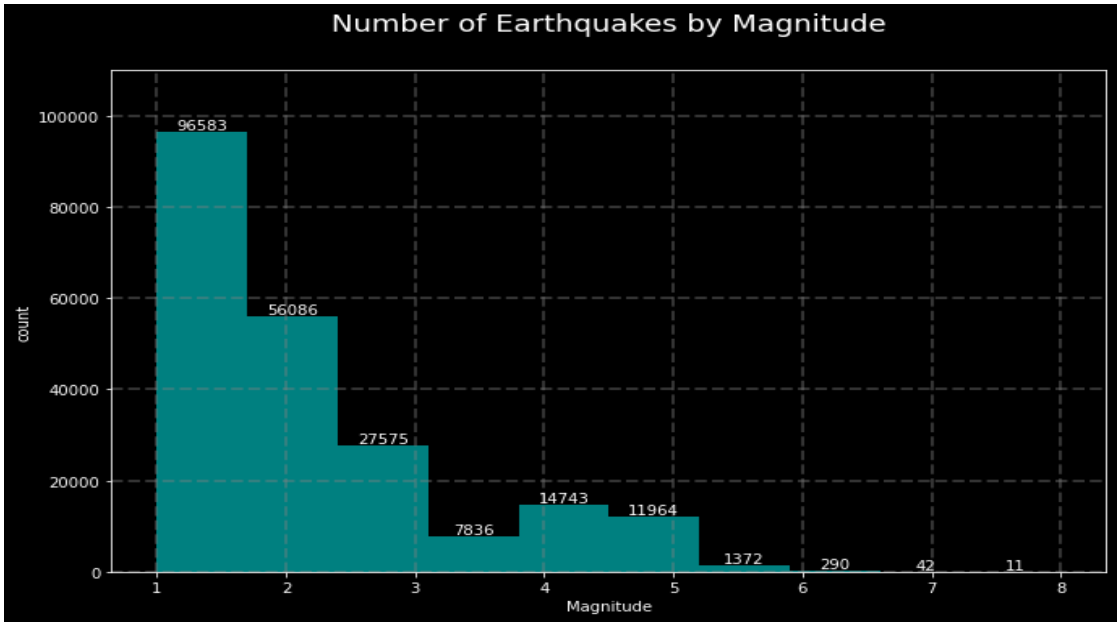


Figure 10: Histogram of number of Earthquakes by Magnitude

Earthquake depth ranges from the earth's surface to approximately 800 km deep. The strength of shaking from an earthquake is greatest closer to the earth's surface. This parameter can reveal important information on the state of the tectonic plates and can provide insight into deformation in the subduction zone. This can be a difficult parameter to accurately measure however and the error of depth is generally greater than the error for the location of an earthquake. The depth of earthquakes in our dataset are on the lower end and can be considered shallow earthquakes.

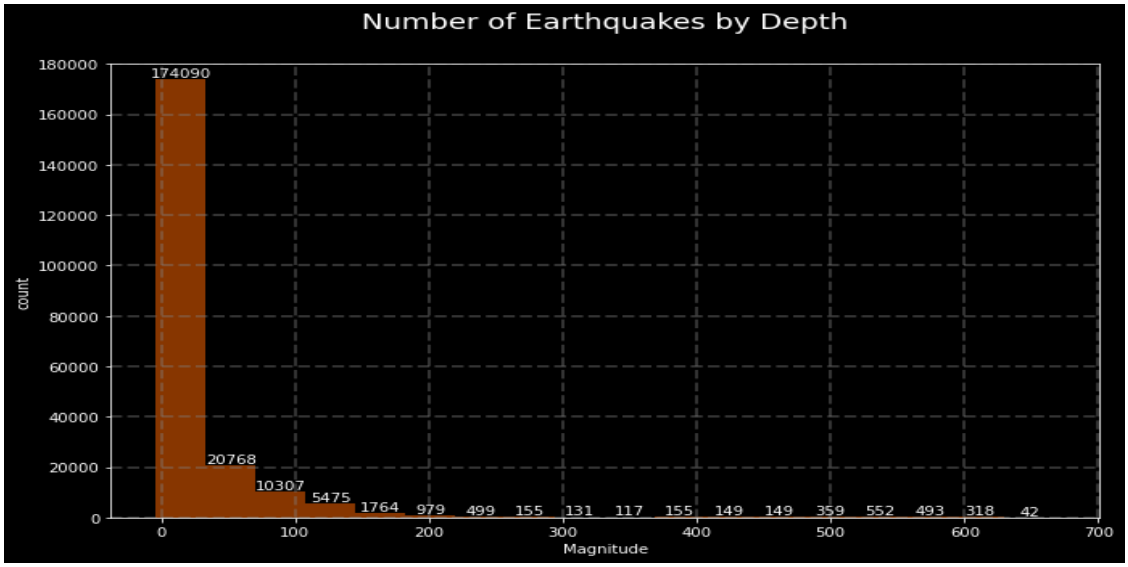


Figure 11: Histogram of number of Earthquakes by Depth

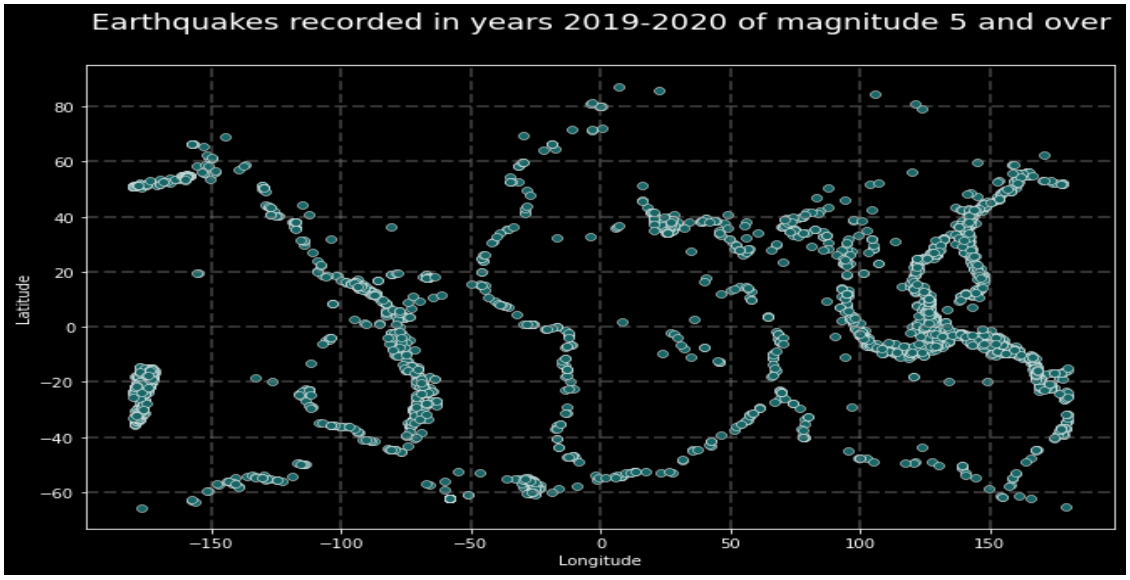


Figure 12: Scatterplot of Earthquake data, magnitude 5 and higher

When filtering the earthquake data for 2019 and 2020 for magnitude 5 earthquakes only, we can see that the general observations from the previous scatterplot still holds true. The majority of earthquakes occur in lines along tectonic plate boundaries and we have

a larger number of earthquakes in Asia and along the plat tectonic lines for North and South America.

Next, we looked at the data from the same time period of volcanic eruptions.

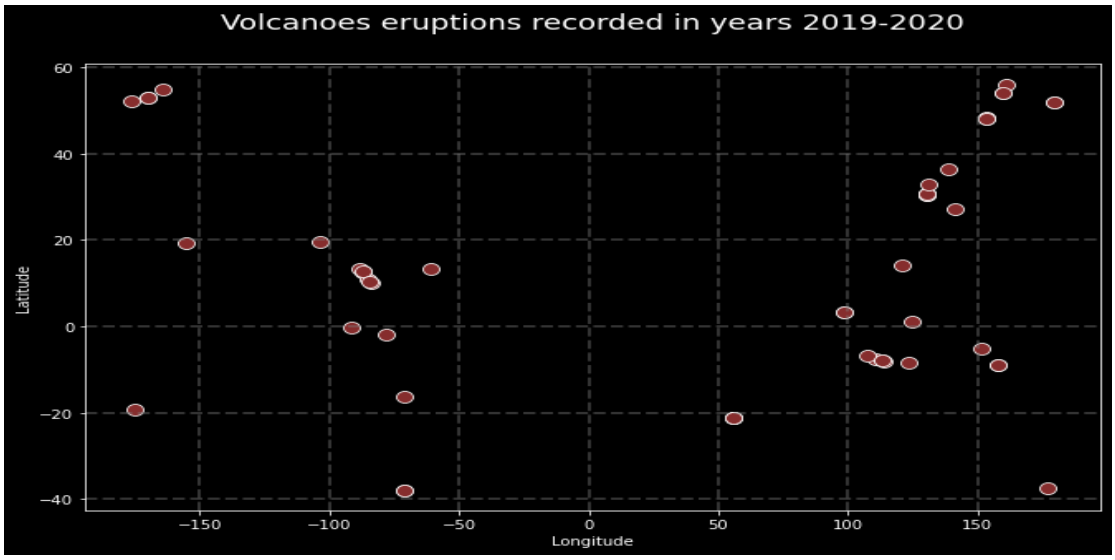


Figure 13: Scatterplot of Volcanic eruption, 2019-2020

General observations:

- as per data exploration and data preprocessing, we have much fewer events for volcanic eruptions during the years 2019 and 2020
- we can mainly observe the volcanic activity around the edge of the Pacific Ocean

To visualize the connectivity of earthquake and volcanic eruptions, both types of geological disasters were plotted with a Bokeh plot.

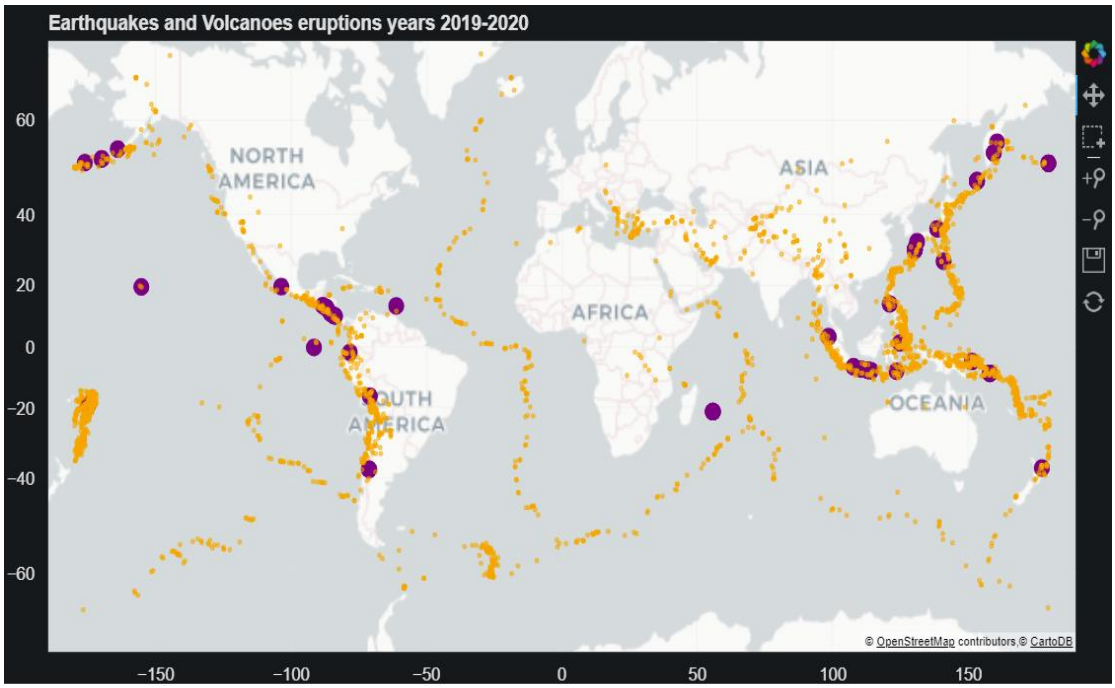


Figure 14: Bokeh Plot Geological Disasters

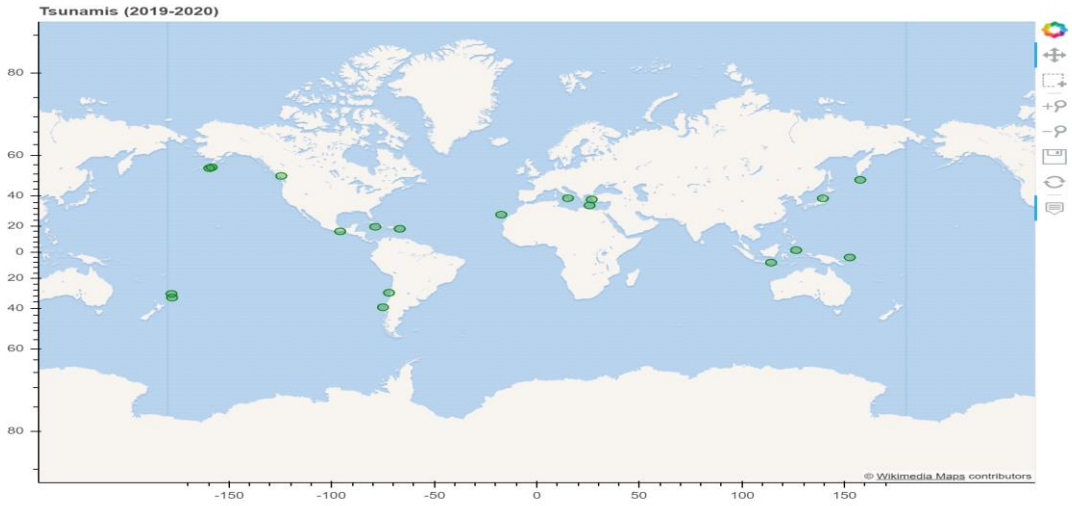


Figure 15: Bokeh Plot Tsunamis

LABELING STAGE - STEPS TAKEN:

A key component of our project was to design and develop a method to allow for a 'match to be determined', indicating there is a potential link between a volcano and a previous earthquake. This was challenging on two fronts, firstly an understanding of the science of earthquakes and volcanoes and research of published literature was required to determine appropriate parameters to apply to test for association. Further, much research and trial and error were required to develop the code for this task.

VOLCANO DATASET

1. Volcano, 54 observations, date: volcano eruption
2. Multiplication of the above dataset, with 6 days pre-eruption, duplication of records with new intendent date:

	Volcano Number	Volcano Name	Eruption Number	Eruption Category	year	month	day	Latitude	Longitude	date
74	345040	Poas	22303	Confirmed Eruption	2019.0	2.0	7.0	10.200	-84.233	2019-02-04
74	345040	Poas	22303	Confirmed Eruption	2019.0	2.0	7.0	10.200	-84.233	2019-02-07
74	345040	Poas	22303	Confirmed Eruption	2019.0	2.0	7.0	10.200	-84.233	2019-02-03
74	345040	Poas	22303	Confirmed Eruption	2019.0	2.0	7.0	10.200	-84.233	2019-02-05
74	345040	Poas	22303	Confirmed Eruption	2019.0	2.0	7.0	10.200	-84.233	2019-02-02
74	345040	Poas	22303	Confirmed Eruption	2019.0	2.0	7.0	10.200	-84.233	2019-02-01
74	345040	Poas	22303	Confirmed Eruption	2019.0	2.0	7.0	10.200	-84.233	2019-02-06

Figure 16: One volcanic eruption recorded on seven consecutive days

3. Concatenation of all produced volcano datasets into one main that contains 378 observations
4. Merging Volcano All dataset with the main Earthquake dataset on the column: 'date'. This has produced a new data frame with earthquake observations that have relation to a volcano on the date variable. The number of observations: 110426.
5. Creation of new Variable: Distance Between Earthquake and Volcano.
6. Filtering the above Data frame by condition: Distance_km <=1500 km.
7. This has produced a frame of 2565 observations. At that stage, we have got 2565 earthquakes that are related to volcanoes based on 7 days date frame and distance

from each other less or equal to 1500km. These are our future True labels for the whole Earthquake dataset.

8. The next part was to add these True to the whole Earthquake dataset and for those earthquake observations that are not True to label them as False. We have used for loop to do this.
9. In the end we have got an Earthquake dataset with 213944 False labels and 2558 True labels.
10. The next step was to filter the whole dataset by a Magnitude of 5 or over.
11. After doing this our labeled dataset has 3061 observations total, where 2987 are False and only 74 observations are True.
12. To prepare this dataset for Machine learning we had to make sure that all the variables are numeric.

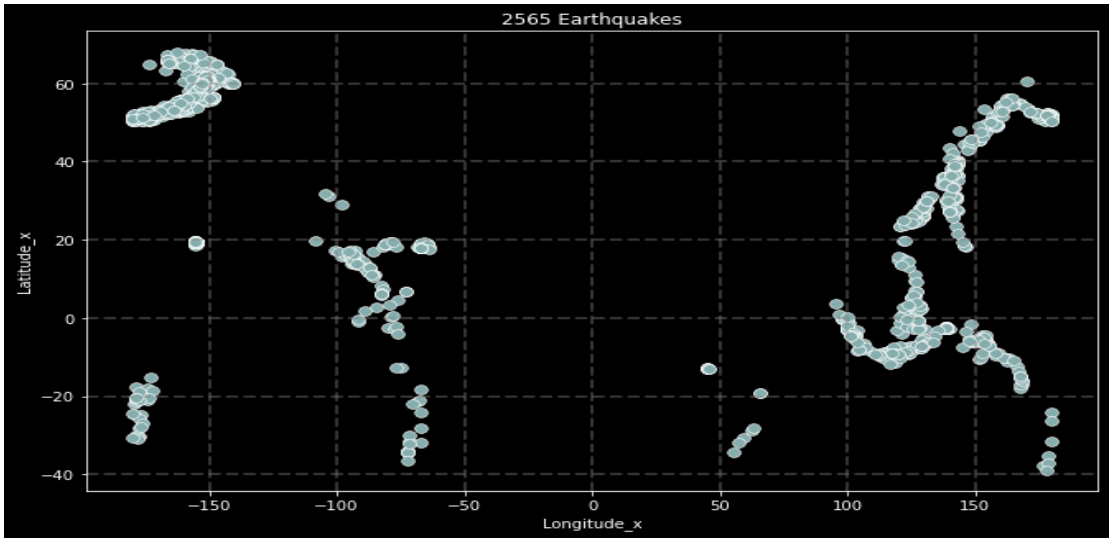


Figure 17: Scatterplot of Earthquakes that matched to a Volcano (mag 5+, 7 days, ≤1500km)

Figure 18: Geospatial Visualization of True labels – no mag restriction

Figure 19: Geospatial Visualization of Earthquakes that may lead to Volcanic eruption – True labels for mag 5+

Fig. 19 is another way to visualize the matches, earthquakes of mag 5+ that had a volcano eruption occur within day of and 6 days subsequent and distance of 1500 km.

Fig. 20 shows the final data set for ML, there are 3061 observations and 5 variables, latitude, longitude, depth, magnitude, and Vol_match (target variable, indicating if a volcano occurred within 7 days and 1500 km.

Fig. 21 is a correlation heatmap of the final dataframe, showing poor correlation of the variables, which is expected, given the variables remaining.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3061 entries, 0 to 3060
Data columns (total 5 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Latitude   3061 non-null   float64
 1   Longitude  3061 non-null   float64
 2   depth      3061 non-null   float64
 3   mag        3061 non-null   float64
 4   vol_match  3061 non-null   int64
dtypes: float64(4), int64(1)
memory usage: 119.7 KB
```

Figure 20: Final dataframe for ML

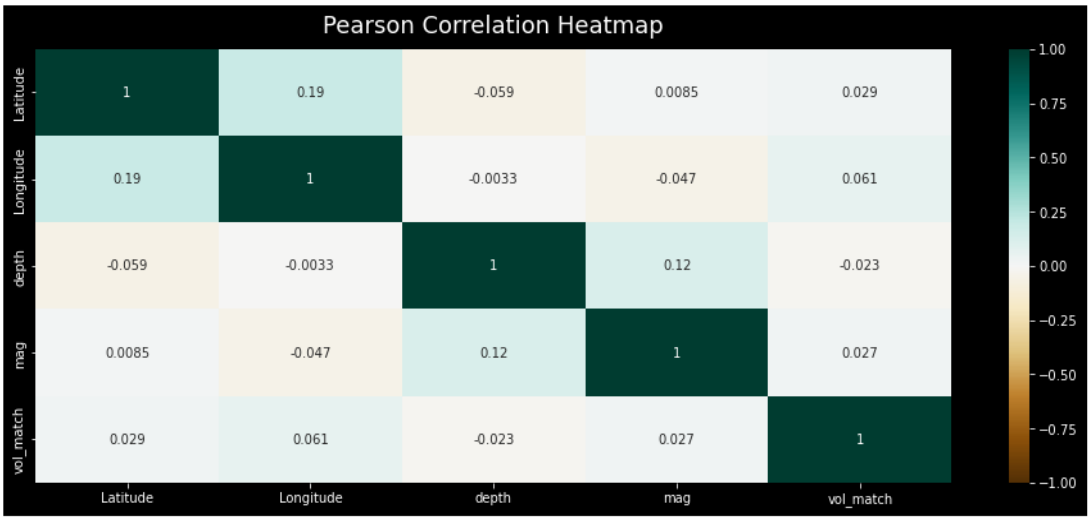


Figure 21: Pearson Correlation heatmap of final dataframe for Machine Learning

4. Modeling

For the modeling phase, we continued working with the newly created dataset earthquakes19_20_ML_7.

From previous research, the objective for the modeling phase was set to find potential choices of an ML algorithm and method that would be suitable for our data and research objective in general. Commonly for labelled data, a supervised training method can be implemented.

The following models have been implemented: KNN, SVM, and Logistic Regression

Before the implementation, the dataset was split into independent and dependent variables with the following details:

Independent variables: 4 variables across 3061 observations (3061, 4), and the target variable with 3061 observations for 1 variable.

One major challenge for the implementation of ML models is the is the imbalance in our target variable. This had to be accounted for. Additionally, as we are implementing among others the KNN algorithm, the data had to be normalized as well. Generally, models are performing better when the numerical input variables have been scaled to a standard range beforehand. We used the Standard Scaler from the sklearn library on our data.

We then split the dataset into training and testing with a split of 30% for testing data. See Fig. 22, below for the shape of datasets for machine learning models.

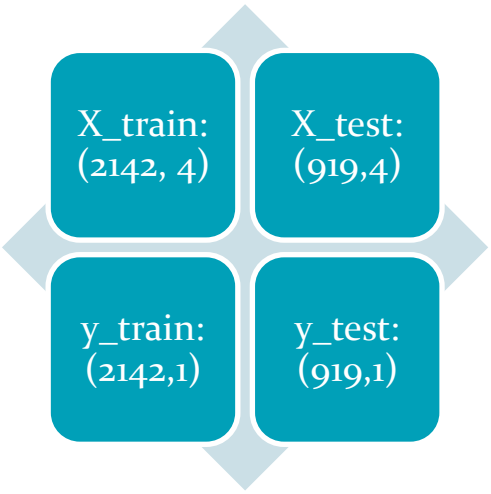


Figure 22: Datasets for Machine Learning Model (test & train)

We also checked the distribution of the target variable in both the training and testing dataset, see below:

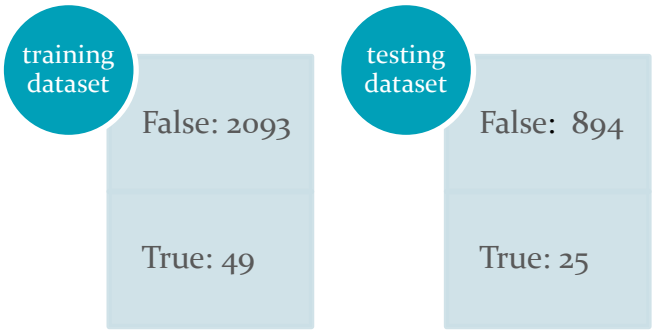


Figure 23: Split of the target variable

One of the major issues in dealing with highly imbalanced data like ours directly relates to the metrics used to evaluate the model. Without considering the imbalance, metrics such as the accuracy score can be misleading.

Generally, in a dataset with a highly unbalanced class variable, the classifier will be biased, and "predict" the most common class without performing any analysis of the features (Brownlee, 2015). As the classifiers are mostly trained on data that is not representing the minority class well. We will determine if this is the case for our data. The results will have a high accuracy rate, which incorrectly indicates good model performance. There are different options to deal with this, one of the more widely used solutions to this problem is to employ: Resampling.

There are different options available, ranging from removing samples from the majority class (under-sampling) and/or adding more examples from the minority class (over-sampling). For the following, we have chosen to apply Over-sampling with the SMOTE method.

SMOTE which stands for Synthetic Minority Oversampling Technique, is only one of many and consists of synthesizing elements for the minority class, based on those that already exist (Alencar, R. 2018). The objective of this is to synthetically generate a nearly class-balanced training set, which then can be used to train the model classifier respectively (Blagus, 2013).

For comparison, here are the dataset details before and after the application of sampling:

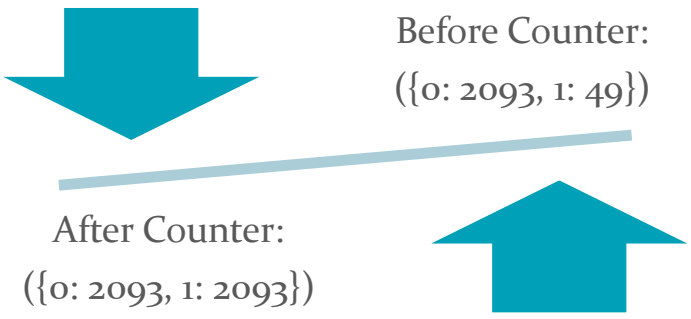


Figure 24: Counter before and after sampling

Generally, applying SMOTE is beneficial for models such as KNN classifiers in high-dimensional data, as it otherwise would be biased towards the minority class. In addition, it potentially performs better than simple oversampling (Blagus, 2013).

To be able to evaluate the effect of balancing the data on the model performance, we will implement the models on the balanced as well as on the imbalanced data. Thereby we will be able to compare and contrast the performance of our data.

ML MODELS

Before we present the results and details of the ML model implementations on our data, here is a brief high-level overview of the implemented models. The general approach was to first implement the model with the default parameter and then again after hyperparameter tuning.

1. K-Nearest Neighbors Algorithm (KNN)

This supervised algorithm can be used for classification problems. The idea behind the KNN is, that it assumes that similar things exist in close proximity e.g., are near to each other (Harrison, 2019).

The algorithm is used for predicting the correct class in the testing data by calculating the distance between the test data and all the training observations (Christopher, 2021). We applied Gridsearch to find the best parameters for tuning the KNN model. Gridsearch was selected, as it allows for the definition of a grid of parameters that will be searched using K-fold cross-validation. The results were: {'leaf_size': 25, 'n_neighbors': 1, 'p': 1}

2. Support Vector Machine (SVM)

First proposed by Vapnik et al. in 1995, support vector machine (SVM) is a supervised linear classification learning method. SVM is an algorithm that takes the data as an input and outputs a line that separates classes if possible (Sunil 2017). “SVM tries to make a decision boundary in such a way that the separation between the two classes [...]is as wide as possible (Pupale, 2019). A decision boundary is defined using SVM in addition to a maximal margin that divides into two classes almost all the data points.

Gridsearch is used to tune the hyperparameters. There are three main types of kernels that can be used: linear, polynomial, and radial basis function kernel (RBF). Here we use the RBF kernel, a function whose value depends on the distance from the origin or from some point. The width of the RBF kernel is controlled by the gamma parameter and governs scale. The regularization parameter is the c parameter which controls for the importance of each point.

The algorithm can be tuned with the following hyperparameter:

C: this parameter affects the tradeoff between a smooth decision boundary and classifying training points correctly. Generally, a high value of c means more complex

decision curves trying to fit in all the points. For tuning the objective is to try different values of c for the data to get the perfectly balanced curve. The main goal is to avoid overfitting of the model.

Gamma: this parameter "defines how far the influence of a single training example reaches." (Pupale, 2019) For gamma with a low value indicates, that also far away points still pull substantial weight.

We also used the Gridserach method to find the best parameters, with the following re sults for our SVM model: `{'C': 1000, 'gamma': 0.01}`

3. Logistic Regression

The last model implemented is Logistic Regression, as this is a suitable model for our binary classification problem (Brownlee, 2020). The model can be used to predict the probability of an instance belonging to the default class.

Linear models are fast to train and robust. In logistic regression, a weighted sum of inputs is passed through a sigmoid function. A S-curve is obtained, and this is used for sample classification. A linear relationship between the variables is not required to use logistic regression.

For this model, we decided to only compare the performance results on balanced and imbalanced data respectively.

No tuning was performed for this algorithm as the results were very poor. We decided to focus on the better-performing models.

5. Evaluation

Model Performance Evaluation

The ML model performance has been evaluated based on the following metrics. For each model, we checked the training and testing score in addition to the classification report that will be presented thereafter.

In order to keep a better overview and compare results, the sensitivity and specificity scores have been added to the below table as well.

The sensitivity score is the metric used to evaluate the model's ability to predict the true positives of each category, and therefore can be used to determine the proportion of the actual positive cases that got predicted correctly. It is calculated by Sensitivity = True Positives / (True Positives + False Negatives). In short: (TP/(TP+FN). A sensitivity score of 100% would indicate that all instances have been correctly identified and that there are no false negatives. Generally, the higher the value of sensitivity means a higher value of true positive and thereby a lower value of false negatives. In contrast, a lower value for the sensitivity score indicates a lower value of true positives and a higher value of

false negatives. Overall, the objective would be to have models with high sensitivity scores.

In comparison to that, the specificity score evaluates the model's ability to predict true negatives of each category and can be applied to determine the proportion of actual negative cases which got predicted correctly as negative. The formula for this score is Specificity = True Negatives / (True Negatives + False Positives) or: (TN/(TN+FP). A score of 100% for specificity would indicate no false positives.

Table 3: ML Evaluation

	Balanced Training Data	Imbalanced Training Data
KNN_1	Training set score: 0.9594 Test set score: 0.8803 **Sensitivity**: 0.8948545861297539 **Specificity**: 0.087378640776699O3	Training set score: 0.9818 Test set score: 0.9717 **Sensitivity**: 0.9921700223713646 **Specificity**: 0.46153846153846156
KNN_2	Training set score: 1.0000 Test set score: 0.9140 **Sensitivity**: 0.9306487695749440 **Specificity**: 0.11428571428571428	Training set score: 1.0000 Test set score: 0.9706 **Sensitivity**: 0.9910514541387024 **Specificity**: 0.42857142857142855
SVM_1	Training set score: 0.5000 Test set score: 0.9728 **Sensitivity**: 0.941834451901566 **Specificity**: 0.14754098360655737	Training set score: 0.9771 Test set score: 0.9728 **Sensitivity**: 1.0 **Specificity**: nan
SVM_2	Training set score: 0.9924 Test set score: 0.9260 **Sensitivity**: 1.0 **Specificity**: nan	Training set score: 0.9935 Test set score: 0.9641 **Sensitivity**: 0.98993288590604O2 **Specificity**: 0.1
Logistic Regression	Training set score: 0.5000 Test set score: 0.9728 **Sensitivity**: 1.0 **Specificity**: nan	Training set score: 0.9771 Test set score: 0.9728 **Sensitivity**: 1.0 **Specificity**: nan

We can see that the training score alone for some models such as Logistic Regression, and SVM with default parameter highly differs between balanced and imbalanced data. In contrast, Logistic Regression as well as SVM with tuned hyperparameter is not picking up a specificity score at all. The low value of specificity here means a lower value of true negatives and implies a higher value of false positives.

In the following the classification report for the imbalanced and balanced data will be presented. For easier comparison, we will add a table with the summary of the scores for all models as well.

The classification report shows the model metrics for precision, recall and f1-score on a per-class basis. The metrics are based on the true and false positives, as well as true and false negatives. The metrics have been indicated by:

TN – true negative: predicted negative for actual negative result

FP – false positive: predicted positive for actual negative result

FN – false negative: predicted negative for actual positive result

TP – true positive: predicted positive for actual positive result

The precision score is the ratio between the true positives and all the positives and refers to the percentage of the relevant results.

Recall is the ability of a classifier to find all positive instances and denotes the fraction of positives that were correctly identified.

The F1 score is a weighted mean of precision and recall scores. The best score is 1.0 and the worst would be 0.0 respectively.

Support is the number of actual occurrences of the class in the specified dataset and can be used for further evaluation of the process. Here it shows that the data is not well balanced for class 0 and 1.

KNN model_imbalanced

	precision	recall	f1-score	support
0	0.98	0.99	0.99	894
1	0.46	0.24	0.32	25
accuracy			0.97	919
macro avg	0.72	0.62	0.65	919
weighted avg	0.96	0.97	0.97	919

KNN model_balanced

	precision	recall	f1-score	support
0	0.98	0.89	0.94	894
1	0.09	0.36	0.14	25
accuracy			0.88	919
macro avg	0.53	0.63	0.54	919
weighted avg	0.96	0.88	0.91	919

KNN model_tunned_imbalanced

	precision	recall	f1-score	support
0	0.98	0.99	0.98	894
1	0.43	0.24	0.31	25
accuracy			0.97	919
macro avg	0.70	0.62	0.65	919
weighted avg	0.96	0.97	0.97	919

Knn model best parameters_balanced

	precision	recall	f1-score	support
0	0.98	0.93	0.95	894
1	0.11	0.32	0.17	25
accuracy			0.91	919
macro avg	0.55	0.63	0.56	919
weighted avg	0.96	0.91	0.93	919

SVC model_imbalanced

	precision	recall	f1-score	support
0	0.97	1.00	0.99	894
1	0.00	0.00	0.00	25
accuracy			0.97	919
macro avg	0.49	0.50	0.49	919
weighted avg	0.95	0.97	0.96	919

SVC model_balanced

	precision	recall	f1-score	support
0	0.97	1.00	0.99	894
1	0.00	0.00	0.00	25
accuracy			0.97	919
macro avg	0.49	0.50	0.49	919
weighted avg	0.95	0.97	0.96	919

SVC model best parameters_imbalanced

	precision	recall	f1-score	support
0	0.97	0.99	0.98	894
1	0.10	0.04	0.06	25
accuracy			0.96	919
macro avg	0.54	0.51	0.52	919
weighted avg	0.95	0.96	0.96	919

SVC model best parameters_balanced

	precision	recall	f1-score	support
0	0.98	0.94	0.96	894
1	0.15	0.36	0.21	25
accuracy			0.93	919
macro avg	0.56	0.65	0.59	919
weighted avg	0.96	0.93	0.94	919

Logreg model_imbalanced

	precision	recall	f1-score	support
0	0.97	1.00	0.99	894
1	0.00	0.00	0.00	25
accuracy			0.97	919
macro avg	0.49	0.50	0.49	919
weighted avg	0.95	0.97	0.96	919

Logistic Regression model_balanced

	precision	recall	f1-score	support
0	0.97	1.00	0.99	894
1	0.00	0.00	0.00	25
accuracy			0.97	919
macro avg	0.49	0.50	0.49	919
weighted avg	0.95	0.97	0.96	919

Figure 25: Classification Report for imbalanced and balanced ML models

For balanced and imbalanced data, the Confusion matrix was plotted, see below in figures 26 and 27. The confusion matrix was used as the basis for Table 4 to compare the results.

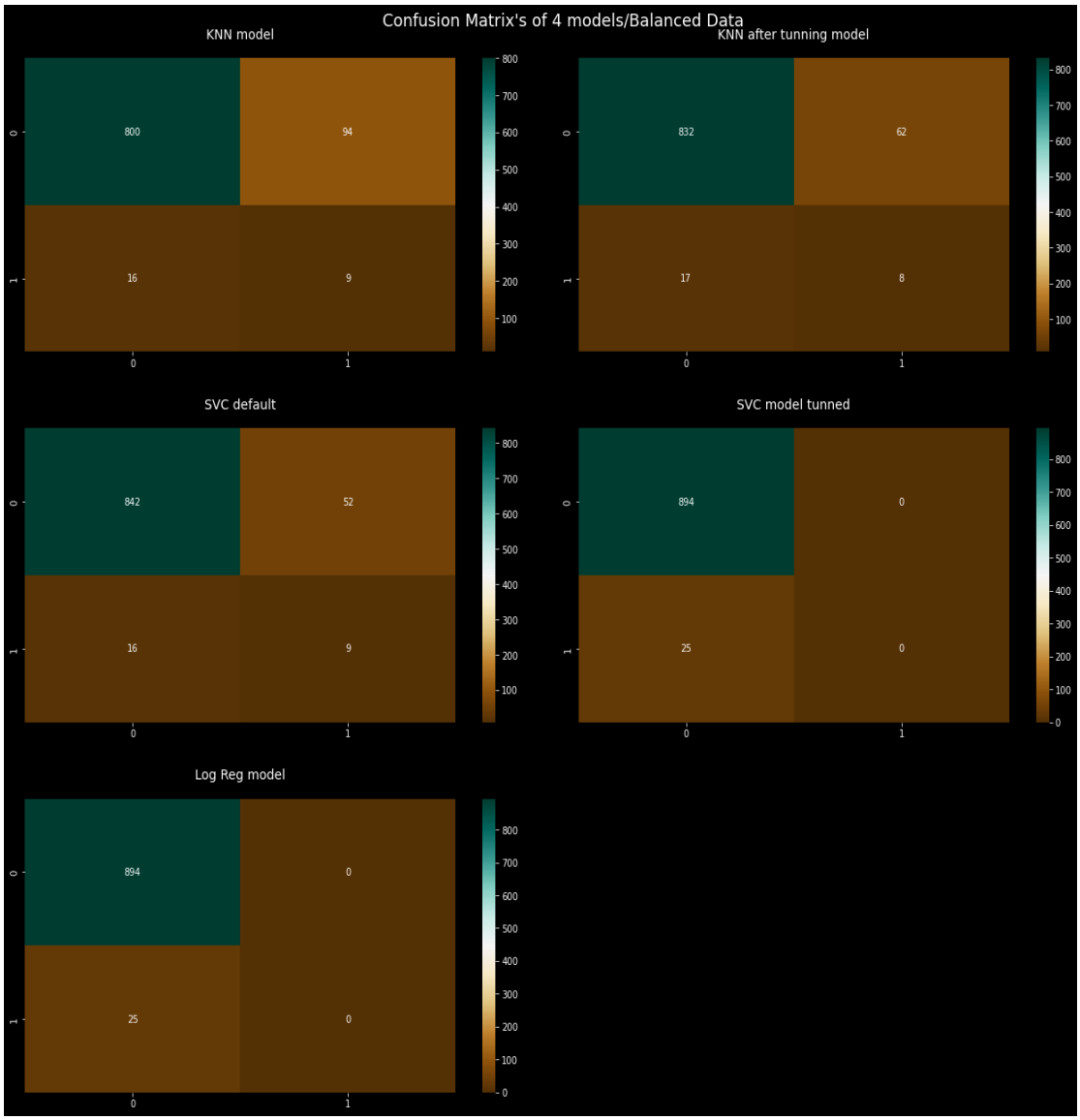


Figure 26: Confusion Matrix of balanced ML models

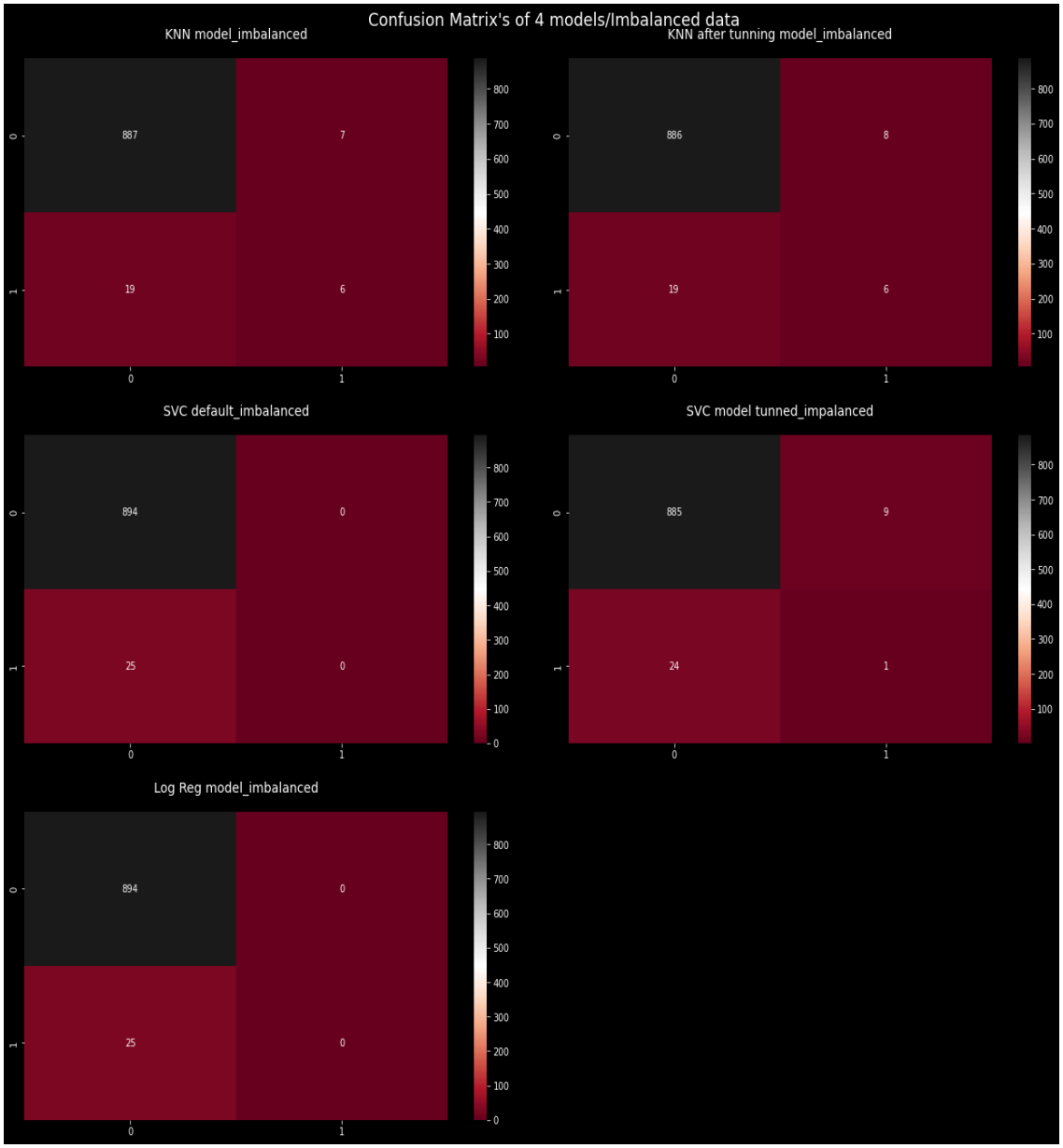


Figure 27: Confusion Matrix of imbalanced ML models

Table 4: Comparison of Confusion Matrix Results:

	Balanced ML models	Imbalanced ML models
KNN_1	TN 800 FP 94 FN 16 TP 9	TN 887 FP 7 FN 19 TP 6
KNN_2	TN 832 FP 62 FN 17 TP 8	TN 886 FP 8 FN 19 TP 6
SVM_1	TN 842 FP 52 FN 16 TP 9	TN 894 FP 0 FN 25 TP 0
SVM_2	TN 894 FP 0 FN 25 TP 0	TN 885 FP 9 FN 24 TP 1
Logistic Regression	TN 894 FP 0 FN 25 TP 0	TN 894 FP 0 FN 25 TP 0

See highlighted above the rates for True Negatives (TN) and True Positives (TP). We can see that those were better recognized in the models with the balanced data.

In order to visualize the performance of the models, the Receiver Operator Characteristic (ROC) curve was plotted. The ROC curve is built by plotting the rate of the true positives (TP) against the rate of the false positives (FP). By plotting, the trade-off between sensitivity and specificity in our models can be observed (Chan, 2020).

Generally, the good performance of a model is indicated by ROC classifiers that give curves closer to the top-left corner (AUC =1).

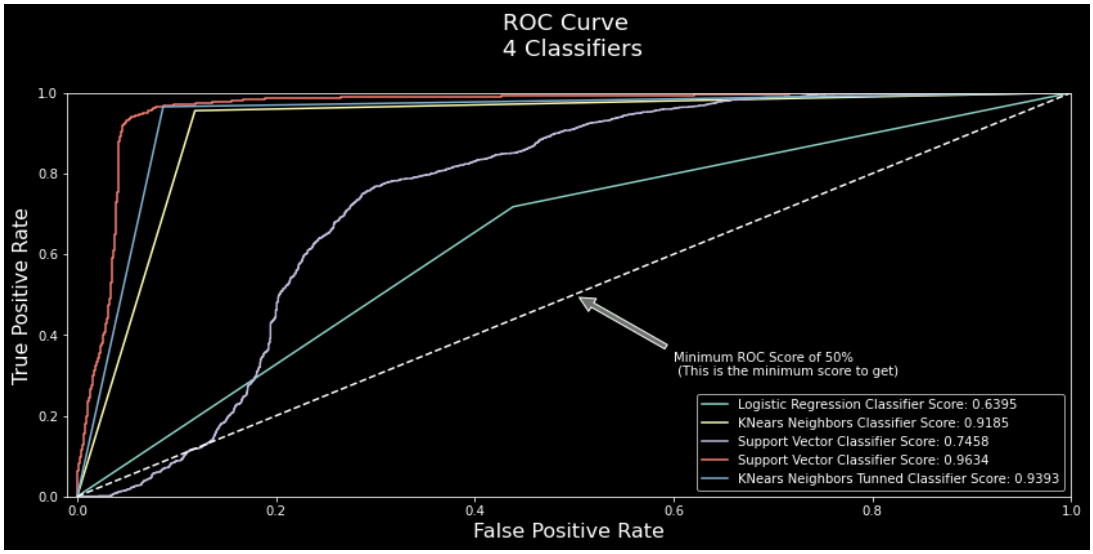


Figure 28: ROC Curve balanced ML Models

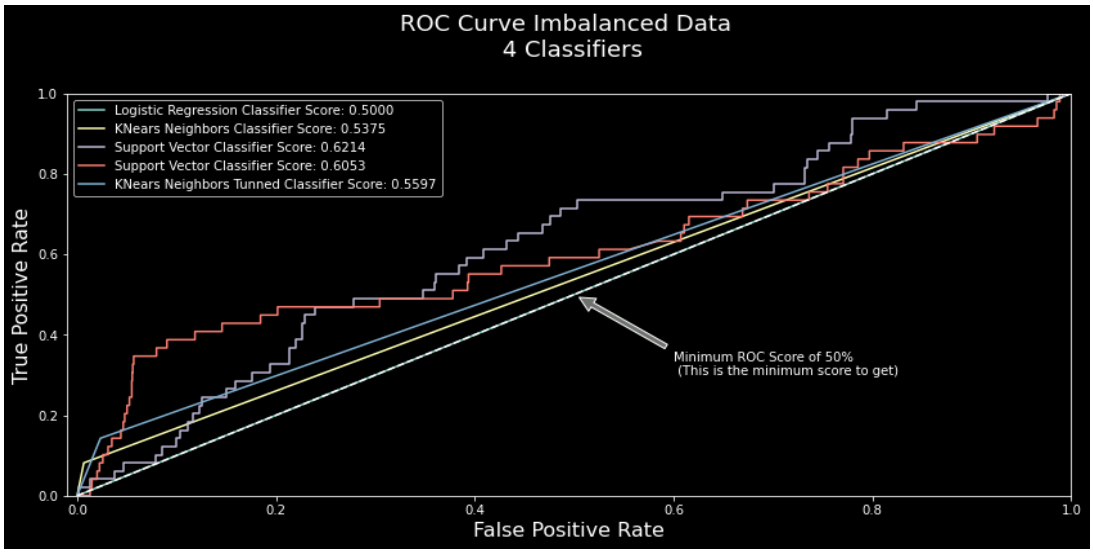


Figure 29: ROC Curve imbalanced ML Models

We can see that the ROC curves have better results for the balanced data. Here SVM, KNN. For the imbalanced data, the performance of the models is worse. The curves from the models are closest to the minimum ROC score.

Additionally, the Area under curve (AUC) was calculated to compare the results and to summarize the different models. The AUC is used to determine the model performance. Generally, the AUC score is "equivalent to the probability that a randomly chosen positive instance is ranked higher than a randomly chosen negative instance" (Chan, 2020)

The higher the AUC, the better the performance of the chosen model in distinguishing between the positive and negative classes.

When AUC = 1, then the classifier is able to perfectly distinguish between all the Positive and the Negative class points correctly, and for AUC = 0, the classifier is incorrectly predicting all negatives as positives, and all positives as negatives.

Table 5: AUC Scores

	AUC balanced ML Models	AUC imbalanced ML Models
KNN	0.9185379837553751	0. 5374718449252611
KNN tuned	0.9393215480172001	0. 55972288580984 23
SVM	0.74579428755876 13	0. 6214105326793881
SVM best params	0.9633641042685686	0. 6053072925300076
Logistic Regression	0.6395126612517916	0.5

From the above plots and tables, we can see that those models which have been fitted to balanced data (after applying SMOTE) recognized the True Negative and True Positives better and did a better classification job. The best results were achieved with KNN and KNN after hyperparameter tuning.

Generally, we were able to observe that specificity and sensitivity scores were good at correctly recognizing data that was synthetically balanced.

By adding those performance metrics to the evaluation, we accounted for the effect of imbalanced data on our model performance. Relying on accuracy scores alone does not show the true results. In a highly imbalanced dataset, a 99% accuracy can be meaningless. Therefore, precision and recall scores have to be accounted for as well.

To finalize the model evaluation, all performance metrics will be summarized in Table 6, below.

Table 6: Ranking Model Comparison

	Model	Accuracy Score	Precision	Recall	Roc_auc	Sensitivity	Specificity
0	SVM_tunned	0.926007	0.147541	0.36	0.963364	1.000000	NaN
1	KNN_tunned	0.914037	0.114286	0.32	0.939322	0.930649	0.114286
2	KNN	0.880305	0.087379	0.36	0.918538	0.894855	0.087379
3	SVM	0.972797	0.000000	0.00	0.745794	0.941834	0.147541
4	Log_Reg	0.972797	0.000000	0.00	0.639513	1.000000	NaN
5	SVM_unb	0.972797	0.000000	0.00	0.621411	1.000000	NaN
6	SVM_tunned_unb	0.964091	0.100000	0.04	0.605307	0.989933	0.100000
7	KNN_tunned_unb	0.970620	0.428571	0.24	0.559723	0.991051	0.428571
8	KNN_unb	0.971708	0.461538	0.24	0.537472	0.992170	0.461538
9	Log_Reg_unb	0.972797	0.000000	0.00	0.500000	1.000000	NaN

Table 6 is ordered by ROC AUC score in descending order, with SVM tuned having the highest score. Regardless of this, we conclude the best models are the KNN models. We can see that the KNN with default and KNN after hyperparameter tuning have the best, most robust results: very high ROC AUC scores, good sensitivity, and fair specificity results. SVM after tuning while having a high accuracy score, and the best ROC score, has poorly recognized positive results. Specificity score is missing as TP and TN are not recognized.

Our results follow the observation by Brownlee (2015), in that the classifiers are mostly trained on data that is not representing the minority class well, and in this highly unbalanced dataset, the classifier will be biased and "predict" the most common class without performing any analysis of the features.

Additionally, the model comparison has also been plotted as a heatmap to visualize the results in color across two dimensions.

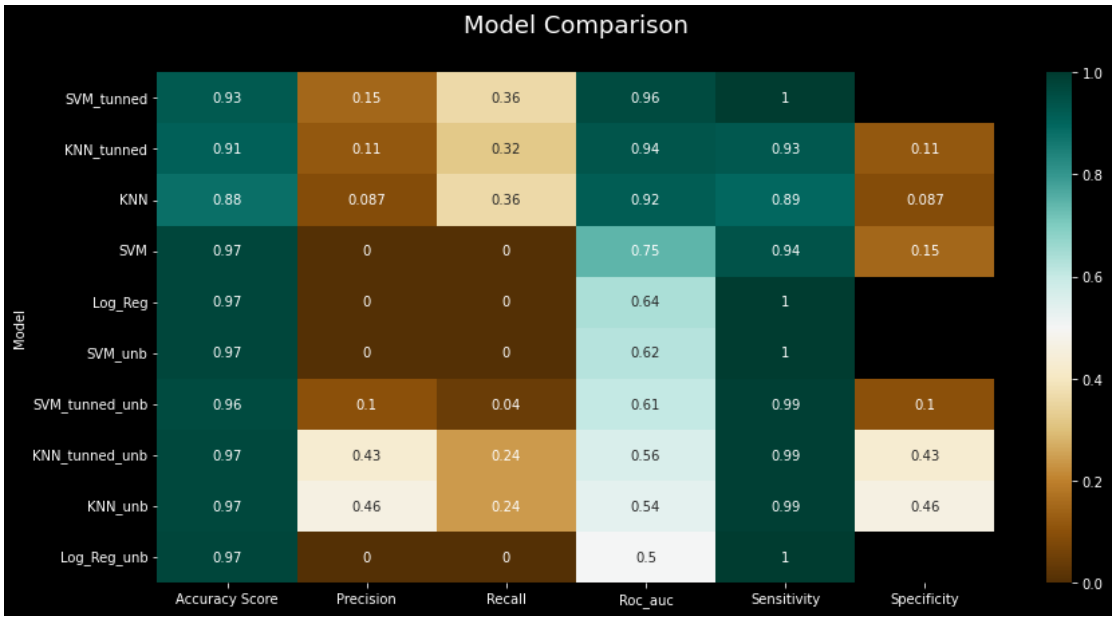


Figure 30: Heatmap Model Comparison

To sum up, the accuracy score alone cannot be used to determine which is the "best performing model". By relying on accuracy score alone, would have chosen the actual "worst performing model" (once other metrics have been taken into consideration).

To have a full picture, we cannot base the results on accuracy scores alone, but have to take all performance metrics into account, such as ROC AUC, sensitivity and specificity scores.

Conclusion

In the preceding report, we presented our approach to analyze earthquake and volcano data with machine learning models. As stated in the introduction when we started the project, the objective was not to be able to predict future earthquakes or volcanic eruptions as a result of earthquakes. Instead, we focused primarily on exploring the possibility to implement machine learning models to further understand the potential causal relationship between both natural phenomena.

Creating the dataset for our project and implementing the machine learning models as described above has shown that this would be a possible approach towards the analysis of their relationship.
When validating our results, we have to take the following into account: for this project, we were only looking into a rather small period of recent historic data. As we only analyzed data from the years 2019 and 2020, there are plenty of different possibilities to expand this small sample to a bigger scale. If we were not constrained by time and resources, expanding the time frame to include not only the two recent years but go back and include the past hundred years, would be a worthwhile option.
Alternatively, it would be an interesting future option to also apply the models to new data. Also considering that we decided not to include tsunamis in our present project, this would be an additional avenue to explore in future research.

We considered the approach of current research in this field and followed appropriate methodologies; however, we note the limitations of our dataset. At the point this became known, it was agreed we did not have sufficient time or resources to step back to the data-gathering phase as we created a dataset ourselves which was a considerable effort. Our idea to explore the relationship between earthquakes and volcanoes, specifically, examining if there is an increase in instances of volcanic eruption following an earthquake has supported research by Sawe and Manga (2018) among others. As our earthquake data was for the years 2019 and 2020, we had 54 volcanoes for this time period, and due to the years involved, we were unable to expand this. We understood at this point, it would have been preferable to have the option to have 30 years of volcanic eruption data to analyze the potential relationship between earthquakes and volcanoes. We also needed to exclude the volcano VEI index as we could not afford to further reduce our volcano records. In any event, we focused on adopting the most suitable machine learning algorithms and believe we have succeeded here. Following this methodology with another dataset would be an interesting endeavor.

This project has been a lesson in sunk cost, we had spent so much time creating a data set, that it would not have been possible to finish the project if we were to repeat or redo this step. It has also been a lesson in resilience and the challenges faced to complete a project that has major constraints.

We also learned the relevance of initial instincts. Our initial ideas were to focus only on earthquakes of magnitude 5+. Other initial ideas were to examine the house prices of an area following earthquakes or their impact on tourism with the idea of using flight and hotel price data. On reflection, eliminating smaller non-relevant earthquakes was a good

idea and some of our earlier ideas would have negated the need to research earthquake and volcano science extensively.

In conclusion, our work does not conclude that there is any causation between earthquakes and subsequent volcanoes. The heatmap concludes there is no correlation between the ML variables. Our work shows early indicators of a potential relationship, the ability to identify a relationship is hugely impacted by the dataset limitations.

The next stage would be to expand the dataset and apply the same methodologies. Some possible avenues would be selecting earthquakes of magnitude 7+ and mapping volcanic activity for several years after. Another avenue would be to focus on a specific region. Focusing on a high magnitude and also selecting for the volcanic activity of VEI ≥2 would be a good approach.

APPENDIX

Presentation

https://prezi.com/i/view/jY97svuqCOCILQpCS3Jw

References

Alencar, R., 2018. Resampling strategies for imbalanced datasets. [online] Available at: <https://www.kaggle.com/code/rafjaa/resampling-strategies-for-imbalanced-datasets/notebook> [Accessed Mar 10th, 2022].

Blagus, R. and Lusa, L. (2013). SMOTE for high-dimensional class-imbalanced data. *BMC Bioinformatics*, 14(1). doi:10.1186/1471-2105-14-106.

Brownlee, J., 2015. 8 Tactics to Combat Imbalanced Classes in Your Machine Learning Dataset. [online] Machine Learning Mastery. Available at: <https://machinelearningmastery.com/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/> [Accessed 12 May 2022].

Brownlee, J., 2020. Logistic regression for machine learning. Machine Learning Mastery. Available at: https://machinelearningmastery.com/logistic-regression-for-machine-learning/ [Accessed May 11, 2022].

Chan, C., 2020. What is a ROC curve and how to interpret it. Displayr. Available at: https://www.displayr.com/what-is-a-roc-curve-how-to-interpret-it/#:~:text=The%20ROC%20curve%20shows%20the,diagonal%20(FPR%20%3D%20TPR). [Accessed May 11, 2022].

Christopher, A., 2021. K-Nearest Neighbor. Medium. Available at: https://medium.com/swlh/k-nearest-neighbor-ca2593d7a3c4 [Accessed May 11, 2022].

CRISP-DM 1.0- Step-by-step data mining guide. Available online at: https://the-modeling-agency.com/crisp-dm.pdf [Accessed: 01 Nov 2021].

Global Report on internal displacement. 2021 IDMC | GRID 2021 | 2021 Global Report on Internal Displacement. Available at: https://www.internal-displacement.org/global-report/grid2021/ [Accessed May 11, 2022].

Harrison, O., 2019. Machine learning basics with the K-nearest neighbors algorithm. Medium. Available at: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761 [Accessed May 11, 2022].

Hill, D., Pollitz, F. and Newhall, C. (2002) 'Earthquake–Volcano Interactions', Physics Today, 55(11), 41-47.

Hill-Butler, C. et al. (2020) 'The co-incidence of earthquakes and volcanoes: assessing global volcanic radiant flux responses to earthquakes in the 21st century', Journal Of Volcanology and Geothermal Research, 393(2020), 106770.

Lemarchand, N. and Grasso, JR. (2007) 'Interactions between earthquakes and volcano activity', Geophysical Research Letters, 34(24): 24303

Linde, A.T., Sacks, I.S., (1998) 'Triggering of volcanic eruptions', Nature, (395) 888-890.

Marzocchi, W., Casarotti, E., and Piersanti, A. (2002) 'Modeling the stress variations induced by great earthquakeson the largest volcanic eruptions of the 20th century', Journal of Geophysical Research, 107, B11, 2320.

Natural hazards portal. Danger levels earthquakes - Natural Hazards Portal. Available at: https://www.natural-hazards.ch/home/dealing-with-natural-hazards/earthquakes/danger-levels.html [Accessed May 11, 2022].

National Geographic Science. 2021. Explore Plate Tectonics. [online] Available at: <https://www.nationalgeographic.com/science/article/plate-tectonics> [Accessed 8 December 2021].

National Geographic Society, N., 2021. Plate Tectonics and Volcanic Activity. [online] National Geographic Society. Available at: <https://www.nationalgeographic.org/article/plate-tectonics-volcanic-activity/#:~:text=The%20two%20types%20of%20plate,boundaries%20and%20convergent%20plate%20boundaries.> [Accessed 8 December 2021].

Pupale, R., 2019. Support vector machines(svm) - an overview. Medium. Available at: https://towardsdatascience.com/https-medium-com-pupalerushikesh-svm-f4b42800e989#:~:text=SVM%20or%20Support%20Vector%20Machine,separates%20the%20data%20into%20classes./ [Accessed May 11, 2022].

Radečić , D. 2020. Here's How To Calculate Distance Between 2 Geolocations in Python. [online] Available at: <https://towardsdatascience.com/heres-how-to-calculate-distance-between-2-geolocations-in-python-93ecab5bbba4> [Accessed 14 May 2022].

Sawe. (2020). How Many Tectonic Plates Are There?. Available: https://www.worldatlas.com/articles/major-tectonic-plates-on-earth.html. [Accessed 8th December 2021.]

Sawi, T.M., Manga, M. (2018) 'Revisiting short-term earthquake triggered volcanism', Bulletin of Volcanology, (2018) 80: 57.

SCI. (2007). Plate tectonics. Available: https://www.sciencelearn.org.nz/resources/339-plate-tectonics. [Accessed 8th December 2021.]

Science Learning Hub. 2021. Plate tectonics. [online] Available at: https://www.sciencelearn.org.nz/resources/339-plate-tectonics [Accessed 8 December 2021].

Sunil, 2017. SVM: Support Vector Machine Algorithm in machine learning. Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/ [Accessed May 11, 2022].