CZECH UNIVERSITY OF LIFE SCIENCES PRAGUE

FACULTY OF ECONOMICS AND MANAGEMENT

DEPARTMENT OF SYSTEM ENGINEERING

DIPLOMA THESIS

Web Application of Earthquake Based on Machine Learning

Rabendra Pd. Shrestha

(Master's in informatics)

SUPERVISOR: doc. Ing. Vojtěch Merunka, Ph.D.

Declaration

I declare that I have worked on my master thesis title “Web application of earthquake based on Machine learning” by myself. I have developed using Python as coding and using the Moving Average (MA) method. It is the classical machine learning approach easy to use and easy to figure out the trend and understandable. I have used very simple frontend using only html and CSS but there is also additional google map which make easy for the map integration. I have solved my the work issued by my own, where researched from publication author or online articles, pdfs, related to my title has been included. I have included the screen shot of my practical part as well as some of the figures from google as an example. My contribution and those authors to this thesis have been explicitly sourced or indicated at the end of the writing.

06.04.29 ____________________________

Signature

Acknowledgement

First of all, I would like to express my gratitude by thanking my supervisor “doc. Ing. Vojtěch Merunka, Ph.D.” from the “Department of Information Technology” at “Czech University of Life Science” for granting this opportunity and be my supervisor and his advice, guidelines, support. And, I would also like to thank all the people whose assistance was a milestone in the completion of this thesis.

Summery

This is the application base project and the complete package for the end users so they can use in their devices. There are the methods and modules are used to complete this project. Using moving average (MA) method based on time series data and analysis the trend that which is going towards make it easy to predict. The forecasting is the short time period. In this project forecasting is five next days periods. The data is the real direct attached from the USGC. This application is like the pre alarm that inform the earthquake might happen in next five days in where and how much magnitude might be face.

But that does not mean it ring the bell to the email or get the notification to the mobile devices. It just provides the pre inform and possible danger to the places and try to make prevention activities by users own.

Since this is the based-on machine learning, so data is the time series data. Before the application there is the machine learning analysis and the data visualization to see the trend of the flow of earthquake. After that applied model and methods, test accuracy and only than the code converted into the web application.

Keywords

Earthquake prediction, Artificial neural networks, Time series, Machine learning, Web Application, Data Analysis, Data Visualization.

Abstract

The main objective of this thesis is the end product of predicted earthquake based on machine learning. This is the attempt of early alert to the people. Earthquake is very uncertain due to its highly complex nature. Earthquake brings the devastating and natural calamities. Financially and human losses as well as the economically brings weak to the nation to nation. More than 100 years have been conducted researching on predicting earthquake. Still, it is hard to high prediction like the magnitude of 8 or more. But mostly predicted accurately after shocked as per the tested and research based. Also, in the past some algorithms have been developed for testing those are M8 and MSC (i.e., the Mendocino Scenario). During 1992 to 1997, five earthquakes were occurred of magnitude 8 and above in test area. M8 and MSC were capable of predicting earthquake and correctly identified the location.

Earthquake has various seismic signals that should be identified which are analyzed through past historical data. The studies bring to develop the model more suited for real-world to forecast earthquake. The machine learning algorithm and neural networks bring changes of forecasting like ARIMA, MA, LSTM for time series data.

Table of Contents

Introduction

Earthquake is the natural disaster which cause lots of damaged in terms of human life and financial loses. Main cause of earthquake is the movement of seismic tectonic plates. Earthquake has been studied from the very long periods of time. It is one of the most challenging tasks to analysis and predict from that period of time. There are two concepts related to the prediction. Which, one comes from the impossible result as a prediction and another one trying to achieve the task.

This is the true that more than a century, seismologist community is trying to develop a method to predict the earthquake. Hence the lack of scientific instruments and technology which could not monitory accurately and could not achieve the comprehensive data. The instruments could be monitor on the stress changes, pressure and temperature variation beneath the crust which might be possible to get the accurate data.

There might be very valuable information left behind buried under the crust when seismic waves travel. Those might be very useful in various fields like geological study, mining, oil and gas industries etc. Also, seismic data processing can be very applicable in earthquake and after shock detection.

It is not possible to prevent the earthquake instead people must prepare for. And using computer simulation helps to build safer structures and better understanding the mechanism of seismic waves which minimize the damage during earthquake. And modern computer based on intelligent algorithms achieve significant results on different sectors like weather forecast, disease diagnosis etc.

Earthquake prediction and earthquake forecasting are slightly different things. As per the author’s perspective forecasting shows the probability of future occurrence while prediction means in the form of Yes or No without any associated probability factor.

Generally, earthquakes predictions have two different approaches those are **precursors based** and **trend based**. Precursors are anomalous phenomena that might signal an impending such as radon gas emissions, unusual animal behaviour, electromagnetic anomalies etc. Trend based methods involve identifying patterns of seismicity that precede an earthquake. Collect the data as time series on trend-based approach. In this paper, a trends-based approach is adopted and the Moving Average (MA) neural network is used to capture the trend involving statistical techniques.

After the predicting and analysis by MA than the analysis prediction converts to the web based interactive application. In this paper using current one-month data from USGSC, predict the data and forecast of the five days and finally convert to web base interactive application. So, the people would be alert and precaution on the future impending.

Objectives and Methodology

Objectives

The main objective of the thesis is to develop the web based application to end user using analysis and find out the behavior of historical data. The medium of analysis is the Moving Average (MA) specially used for time series data forecasting. Using this application, the end user will be pre alert and caution for in near future calamities.

Adapting the machine learning algorithm will help to better understand of analysis which also based on data visualization will help to more clarify.

Methodology

The thesis will have two parts. The first part of the thesis is based on study and research of the academic papers, journal, Books, online articles and research subjects.

The second part of the thesis will be a real project documentation such as the practical methodology, UML diagram, proven data, proven algorithm, data analysis, data visualisation and python project. Based on the practical and using machine learning module, accuracy will be measured and the conclusion as a web base application will be framed.

Tools

Some of the tools which help to complete for this thesis.

Python

Python is a high-level programming and it is widely used in the programming world. It's origin by Guido van Rossum in 1991 and after than its further developed by the Python Software Foundation. Its main purpose to design is emphasis on code readability, and with help of syntax, programmers can easily write their code fewer lines.

Python is one of the programming languages. It is used in multiple purposes. Here are some of the list below where the python can be used.

1. Web development – Popular frameworks like Django and Flask are widely used for web development. They help to write server-side code and to manage database, write backend programming logic, mapping URLs, create REST API and many more.
2. Machine learning – Machine learning codes are written in many languages and one of them is Python. There are many machine learning applications can be seen and written in Python. Python makes small codes are fast processing that is why it is most widely used. So, it is also an easy to write a logic so that a machine can quickly learn and solve a particular problem on its own. For example, products recommendation for various websites like Amazon, Flipkart, eBay etc. are based on machine learning algorithm. So, as per user's interest it is displayed in list without user's more effort on searching. Another application is Face recognition and Voice recognition which are mostly used in mobile phone is also another example of machine learning.
3. Data Analysis – Python is also more popular for handling data so as Data analysis and data visualization. The charts and plots for displaying data trend are also developed by using Python.
4. Scripting – Scripting is small and so powerful mostly used for system. Mostly is used to automate simple tasks for example sending automated response emails etc. Python scripting is one of them which is also possible to write a such type of applications.
5. Game development – By using python it is also possible to develop games.
6. Some of the Embedded applications are also developed in Python.
7. Desktop applications – Some of the library like TKinter or QT are used for desktop application.

Jupyter Notebook

It is an open source web application. It is hybrid application used as IDE in web and get output instantly without run any command in console as well as it also stored the project in the folder. It can able to use to create and share documents during the live code, equations, visualizations, and text. It is maintained by the people at Project Jupyter.

It is a spin-off project from the IPython project that is why it has an IPython Notebook project itself. Jupyter transport with the IPython kernel so this brings to allow write programs in Python. There are currently over 100 other kernels running in Jupyter that can also use. The purpose of Jupyter notebooks is to support researcher or pedagogy and make them easily to write code also to provide a more accessible by easy interface.

Visual Studio Code IDE

Visual Studio Code is a tool for editing the code and save for reused. It very powerful and easy-to-use. It supports all kind of programming language support. It is highly customization with various extensions, and it is for free. It is good package for beginners and more advanced programmers also.

It is developed by Microsoft but can use any operating system like Windows, Linux and macOS. There are some features available such as support for debugging, embedded for Git control and GitHub, easy to read using syntax highlighting, intelligent code completion for less errors, snippets, and code refactoring. It can be customizable, allowing users to change the theme, keyboard shortcuts, preferences, and install extensions as per the users' preferences. There is the source code which is free and also available open source and released under the permission of MIT License.

There's combine features that makes Visual Studio more intelligent with a source code editor and the powerful developer tooling, such as IntelliSense code completion and debugging. That makes frictionless edit-build-debug cycle. Which is very handful for less time fiddling with the own environment preference, and more time executing on new ideas.

It is a lightweight fast source code editor, perfect for daily basis. VS Code helps for instant productive syntax such highlighting, bracket-matching, auto-indentation, box-selection, snippets, and more for any of the high-level languages including python. It makes code to use easy like accustomed used of instinctive keyboard shortcuts, easy to customization and community-contributed keyboard shortcut mappings. Visual Studio Code also has some inbuilt intelligent such IntelliSense code completion, rich semantic code understanding and navigation, and code refactoring.

Debugging is most important for written code. To test code that run successfully or not debugging helps to get error and which piece of code and where and which line numbers, what kind of error etc. are all shows by debugging. Visual Studio Code also includes an interactive debugger. So, it is easy to step through source code, inspect and test variables, view call stacks, and can execute commands in the console. It is like a complete package.

Due to VS Code, it is easy to perform of common tasks and merging individual code using by different users from different location. That is the Git so any user can work with source control without going to other tools for editor including viewing pending changes diffs.

Python Modules used

- **Scikit-learn:** For data mining and analysis which optimizes Python's machine learning usability. Used for handling basic Machine Learning algorithms like clustering, regression, linear and logistic regressions, classification, etc.
- **XGBoost**: XGBoost is an algorithm used in Machine Learning. XGBoost is an implementation of gradient boosted decision trees designed for speed and performance.

  XGBoost is an open source library providing a high-performance implementation of gradient boosted decision trees. An underlying C++ codebase combined with a Python interface sitting on top makes for an extremely powerful yet easy to implement package.
- **Numpy:** It is specially used for scientific calculation
- **Pandas:** This model widely used for data analysis. It offers developers with high-performance structures and data analysis tools. So, this helps them reduce the project implementation time.
- **Matplotlib:** It is widely used for plotting specially for creating 2D plots for example histograms, charts, scatter plot, etc.
- **Basemap:** It is most popular tool for creating maps. It is used in python module. It is an extension of a matplotlib, so it has so the same features of matplotlib library to create for data visualizations. Most importantly it provides more good visualization specially for map, unlike google map. Some of its features are adds the geographical projections, from the datasets to plot coast lines, countries, and so on.

Google Map

It is a web mapping service developed by Google to integration word map to the applications. It has various features such as using satellite imagery, aerial photography, street maps, 360° interactive panoramic views of streets, real-time traffic conditions, and route planning for traveling by foot, car, bicycle and air, or public transportation.

It is used in frontend side. It can be used in various form. JavaScript or iframe are used to integration in the html tag. It has provided the api key to integration. This is applicable to any programing language.

The main purpose for using google map is for the clear view and easy to handle. It is used in web. So, it can access from anywhere only needed is the internet access. Through using google map can easily see all the countries and location. Also, easy to know where and what is happening based on the dataset using, handling and integration and other functions.

Literature Review

This chapter covers all the related the earthquake prediction and forecast. It focuses all the process behind to complete the application like the machine learning algorithm MA, UML diagram, python for web application and the data visualization.

Time Series

Before dive into the machine learning and algorithm let's first see the time series, time series analysis and time series forecasting.

A set of data in numeric form of an individual or collective time order on the based of sequential direction, in the most of cases in equal distance or period of time. According to (Chatfield, 2000) "A time-series is a collection of observations made sequentially through time" (p.1). Time series has the main quality of sequential order of data as per the time variables. Any variable is changed as per the change on time.

In statistic has three kind of data first is time-series, second is cross-section and third is pooled data. Time series data depends on time changes specially in sequential order. That can be daily basic or monthly or yearly or seasonally. But the data are all in particularly time basic. Cross-sectional data are all collected in one point of time. And the last one pooled data is the combined of both data structure that are the time-series and cross-sectional data.

Some of the examples of the time series are as follow:

i. sales of a particular product in successive months

ii. the temperature at a particular location at noon on successive days, and

iii. electricity consumption in a particular area for successive one-hour periods

iv. track security the price

v. stock marketing in a day

vi. death on the period of earthquake

vii. consumption of gas and water

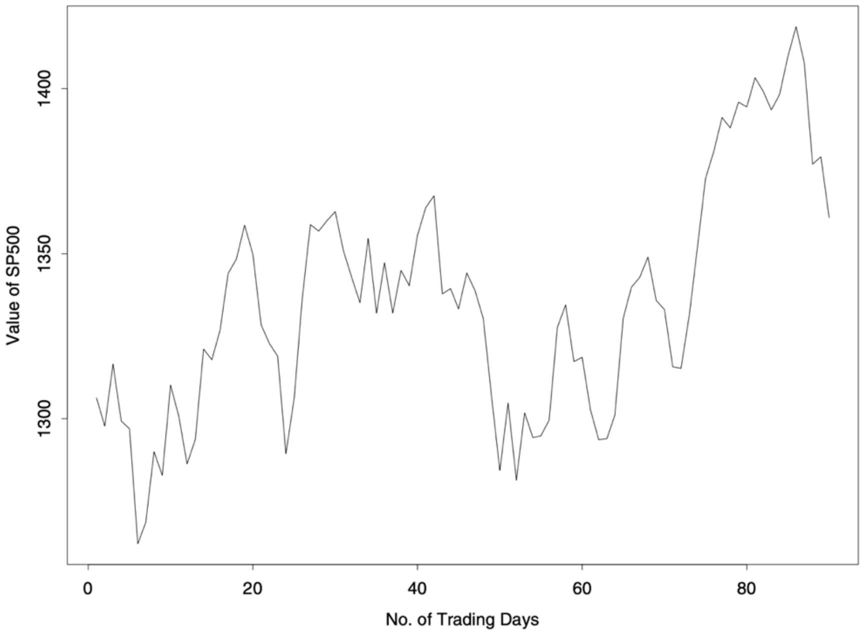


Fig 1: Time Series Forecasting

this figure a graph showing the Standard & Poor (S&P) 500 index for the U.S. stock market for 90 trading days starting on March 16, 1999.

(Note that values for successive trading days are plotted at equal intervals even when weekends or public holidays intervene.)

Applications of time-series forecasting include:

1. Economic planning

2. Forecasting

3. Inventory (or stock) control

4. Production and capacity planning

5. The evaluation of alternative economic strategies

6. Budgeting

7. Financial risk management

8. Model evaluation

Time Series Analysis

Any given variable can be seen through any diagram changes over the time of that variables. From one period to another changing period of time will affect the any variable. That changing variable can be study through the time series analysis.

Let's say the daily electricity consumption in a company. If gathering data of electricity on the period of two years, then analysis those periods like the consumption in January of previous year and this year. If the slight variation might occur than the consumption goes smooth. If there are changes in between two period than there might be something happened there might be more consumption or might be used in holiday or might be used in night period of time. This is so helpful to get the information and can be possible to prevent the same near future occurrences.

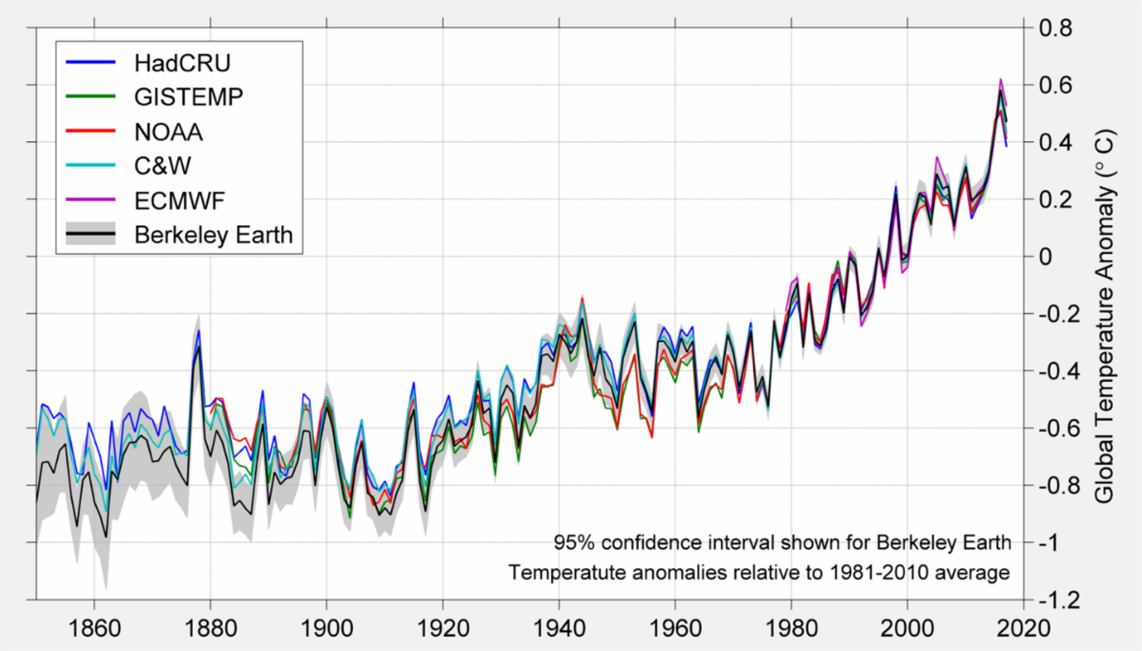


Fig 2: Time Series Analysis

Time series analysis is very useful. It solves the problem that might happen the same thing again and again. Comparing the any two or more than two period of time of any occurrences in previous time helps to clear view of analysis and prevent the future circumstances.

Time Series Forecast

According to Kenton (2019), "Time series forecasting uses information regarding historical values and associated patterns to predict future activity. Most often, this relates to trend analysis, cyclical fluctuation analysis, and issues of seasonality. As with all forecasting methods, success is not guaranteed." Forecasting takes the data of historical and present data. From those data forecast used to make predictions along with analyzing trends. Forecasting is the process of involving models which fit on historical data and using them to predict future observations (Shiva, 2018).

In forecasting has the time that is t and estimating variable Y(t+h) using under available information at the time t (Burba, 2019). The main common goal of time series forecasting is extrapolating from past behavior into the future.

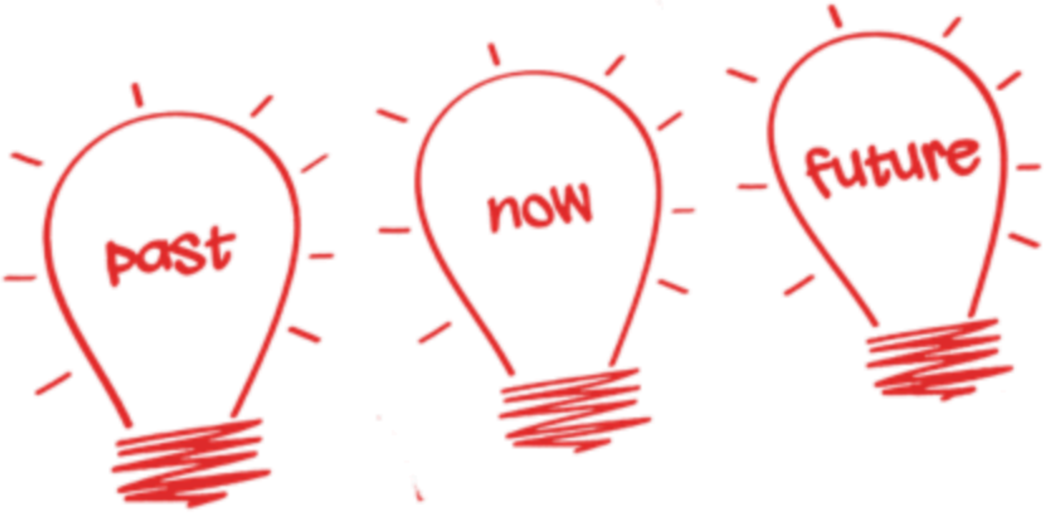


Fig 3: Time Series Forecasting

Time series forecasting is one of the most important problem in all sectors for example business and industry, government, economics, environmental sciences, medicine, social science, politics, and finance. Based on time, forecasting problems can be into three terms such as short-term, medium-term, and long-term. In the case of short-term, forecasting problems predicting events only a few time periods that can be days, weeks, and months for the future time period. Medium-term forecasts extend more than short-term to predict future events from 1 to 2 years, and so as long-term forecasting problems extend beyond the medium-term by many years. Short- and medium-term forecasts are used for the budgeting and new research like the range from operations management to budgeting so as selecting new research and development projects. Also, long-term forecasts used for strategic planning issues. Short-term and medium-term forecasting are typically based on identifying, modeling, and extrapolating the patterns from the historical data. Because unchangeable nature of the historical data, statistical methods are very useful for short- and medium-term forecasting and results are far greater than expected.

Usually forecasting is applicable as daily, weekly, monthly, quarterly, or annually to many business applications using time series data. For business reporting play a crucial part and data visualization makes the clear of the business situation. Time interval of the data can be used for reporting and display the changes.

Some of the examples that data may change instantly, such as the viscosity of a chemical product at the point in time where it is measured; it may be cumulative, such as the total sales of a product during the month; or it may be a statistic that in some way reflects the activity of the variable during the time period, such as the daily closing price of a specific stock on the New York Stock Exchange. Forecasting is the very important and reason behind specially for decision making and planner that prediction of future events which are applicable in many areas (Montgomery, Jennings, & Kulahci, 2015).

Some of the areas are covered as follow:

1. Operations Management. To make success business organizations need to forecast for future situation and must observe routinely for example demand and supply in order to further production and sales and its outcome and control over inventories. This helps to manage the supply chain, recruitment of personnel so to determine staffing requirements, and plan capacity. Forecast not only determine the products and services but also it determines the locations that which product going to produce.
2. Marketing. On the various marketing decisions impact on various factors that can be forecast and easy to help to support the further decision making. For example, main factor on changing the price policies and its effectiveness on business evaluation. Determine the next level of planning and adjustment on to determine the goals of business whether goals are being met or not. Determine the impact of the products to

the customers by new promotions and advertisements. All are the small forecasting and helps for new planning for next level on impact to business success.

3. Finance and Risk Management. Forecast are the most used in financial sectors in return their investments what investors have invested in their assets. Expected returns might be stocks, bonds, and commodities. Decision can be made by the forecast looking on the interest rates, options and currency exchange rates.

   Forecast also can be predicted in the Risk management in Finance so the continuously looking after the volatility of assets thereafter apply the plan for the right decision and secure on the associated investment.

4. Economics. The major backbone of the any nation is the Economics. Proper forecast in the economic sector brings the stable economic conditions, good health, good life of the people in the country. Those economic variables such as gross domestic product, population growth, unemployment, interest rates, inflation, job growth, production, and consumption. Those all variables are to be forecast by Governments, financial institutions and policy organizations.

   Depends on those variables and forecasting decisions and policies are made. Policies might be monetary and fiscal policy. Those economic variables are the instruments of strategic planning and play a major role of decision making.

5. Industrial Process Control. Industries determine the production and its quality. Depends on the quality characteristics of the production process helps to determine various process of the production. Forecast on the qualities make the control over the unnecessary process such as the to determine the change of the process or shut down. Two types of control schemes are used for monitoring and adjustment for the industrial process they are Feedback and Feedforward control. Prediction is the integral part of those schemes for the process of output.

6. Demography. It is maintained by Government get the proper data, planning and forecast depends on gender, age and race. So, every country has their own statistic on those things and depends on those statistic forecast and planning for policy to maintain the balance and economic as well as business forecast as per the gender, age for their products. Birth rate, death, migration patterns, age group etc. are the major factors to forecast and planning the government policies like the retirement program, educations, antipoverty planning, health care sectors, social service actions and so on.

There are the much more problems which require the forecast but there are only two broad types of forecasting techniques—*qualitative* methods and *quantitative* methods.

- *Qualitative* forecasting is based on experienced and subjective who have good knowledge on the related field. The condition might be the less historical data or no historical data. For example, the marking strategies who has the high experienced on the related field can easily determine the taste of the customers based on gender, price, size and income for the new product going to launch in the market. Delphi Method is the well-known of the quality forecasting technique. This technique was developed by the RAND Corporation (Dalkey. N., 1969).
- *Quantitative* forecasting is based on historical data. Forecast is made by under the pattern of data summarized and expressed the statistical relationship of the values of the data. Then choose the appropriate model based on data patterns to predict the future. In other words, to find the future forecasting model is used as conclude the past and current behavior.

In general use there are various types of forecasting model but widely used are regression models, smoothing models, and general time series models.

- *Regression model* is the well-known model in the case of prediction. It uses the relationship between the variables depends on related predictor variables and the predictor variables describes the cause on the forces of the observed values of the variable of interest. For an example the house purchases which is as a predictor variable which predict to forecast the furniture sales. The most well-known method is the *least squares* of the regression models.
- *Smoothing models* are the simple function specially to employ the previous observations so to get the forecast of the variable of interest. Those methods are based on statistic, so they are easy to use for the satisfactory results.
- *General time series* models are the statistical properties of the historical data. It is used to employ to specify a formal model. And generally, using the least squares to estimate the unknown parameters of this model.

There are other features of forecasting problem and they are the forecast horizon and the forecast interval.

The forecast horizon used the number of future periods so the forecasts can be made. The horizon is often dictated by the nature of the problem. As an example, let's take a production planning. In production planning, products demand is forecast may be made by a monthly basis. Because a production needs the more time to change or modify its schedule. So, it will be easy to ensure to know the status of sufficient raw materials and component parts will be available from the supply chain. Also, the proper plan to deliver the completed goods to customers so the customers will facility the inventory. For those things it would be necessary to forecast up to 3 months ahead. *The forecast lead time* is another calling name of the forecast horizon.

Every new Forecasts are to be made as per the frequency that is call the *Forecast Interval*. Let's take a same example from forecast horizon of production planning. Here, the demand of the forecast be made on a monthly basic same as up to 3 months in the future, so the new forecast is prepared for each month. The forecast time interval is the one month of every time cycle when the forecast is made. So, there is always the same length of forecast lead time. Let's say used this in T periods. A rolling or moving horizon forecasting is approached whenever it employs these two things. And every time if this system updates, the forecast will be T-1 of periods in the horizon and then it computes the forecast in the newest period that is the T. If the longer period for lead time is used than to forecast there will apply the rolling horizon approach.

The forecast is the series of process that connected every activity. This is no differentiate than other activities it is simple like input process and output. So, the input will be one or more so as the output also will be one or more. The activities in the forecasting process are:

1. Problem definition

2. Data collection

3. Data analysis

4. Model selection and fitting

5. Model validation

6. Forecasting model deployment

7. Monitoring forecasting model performance

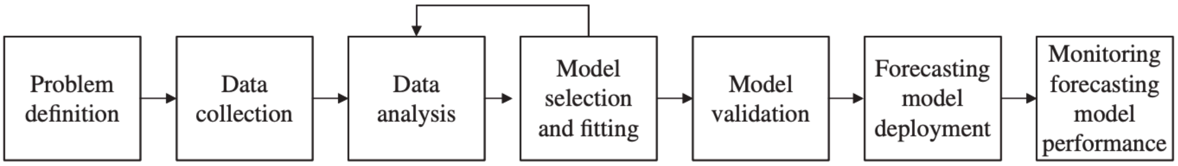
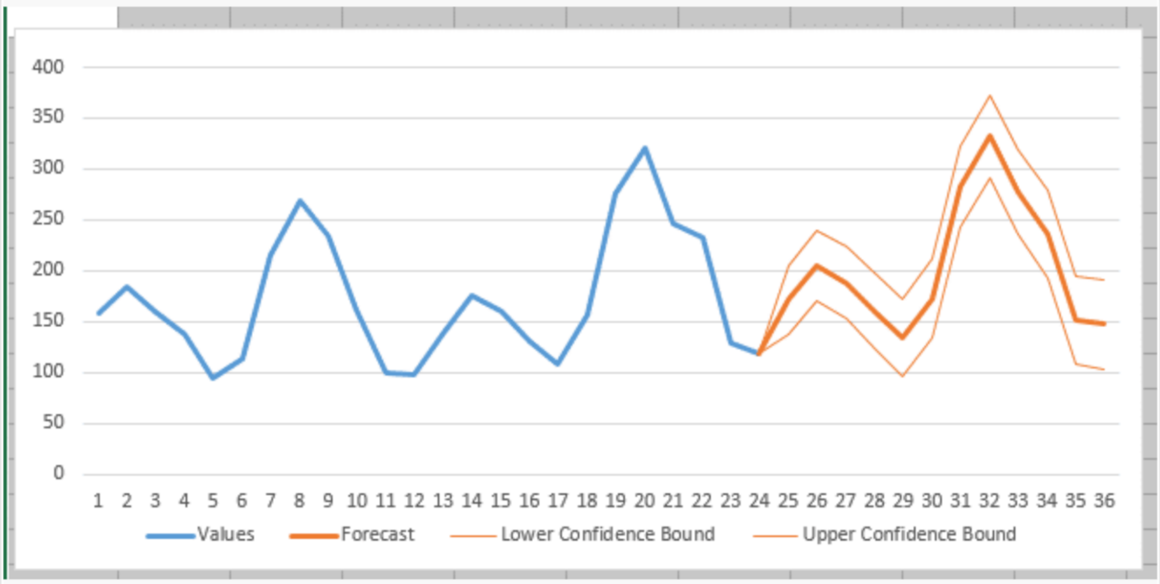

Problem definition
Data collection
Data analysis
Model selection and fitting
Model validation
Forecasting model deployment
Monitoring forecasting model performance

Fig 4: Introduction to Time Series Analysis and Forecasting

Model Evaluation is one of the important factors for time series forecast. There must meet some statistical criteria for model evaluation for example, mean error, mean absolute error, or root mean squared error, etc. Those are the base on the statistical criteria. To fit on time-series model during for forecast model evaluation will apply during that process (Gerlow, Scott & Liu, 2002).

For example, fitted models are checked and examining by the goodness-of-fit. Generally sample data from the population is used to estimate the model parameters. In time-series data, the data is used from all available data and provide the new forecast values of new time dependency. So, the new values are compared from the actual observation. The accuracy will find out from those comparison.



The blue curve represents known data. Forecasted data is represented in orange (lighter orange curves give the bounds of the confidence intervals).

Fig 5: Time Series Analysis and Forecasting Definition and Examples

Let's we have time series observation x_1, x_2, …, x_N and require forecasting of the future values let's say x_{N+h}. Here, h is the integer which represent the lead time or the forecasting horizon (h for horizon). The forecast of x_{N+h} made at time $_N$ for $_h$ steps ahead will be denoted by x hat i.e. in symbol $\hat{x}_N$ (h).

A forecast method is a procedure using past and present data. An algorithm is applied using those data. So, a particular model is arising from the behavior. Depends on the conditions of that model an optimal forecast will be find out. Thus, the two terms `method' and `model' should be kept clearly distinct.

Forecasting methods may be broadly classified into three types:

1. *Judgemental forecasts* based on subjective judgement, intuition, `inside' commercial knowledge, and any other relevant information.
2. *Univariate methods* where forecasts depend only on present and past values of the single series being forecasted, possibly augmented by a function of time such as a linear trend.
3. *Multivariate methods* where forecasts of a given variable depend, at least partly, on values of one or more additional time series variables, called predictor or explanatory variables. Multivariate forecasts may depend on a multivariate model involving more than one equation if the variables are jointly dependent.

In general approach there are combined of more than one forecasting method. For example, there will not be any mathematical model to express for univariate or multivariate than to adjust the model using subjectively to take account of external information.

Mostly forecasting is based on techniques to implement the particular methods. There might need the forecasting strategies for using. Here are some of the methods to fit in a class of time-series model. They are listed as follow: Autoregression (AR), Moving Average (MA), Autoregressive Moving Average (ARMA), Exponential Smoothing (SES), Autoregressive Integration Moving Average (ARIMA), Neural Network (NN) as LSTM.

Autoregression (AR)

Autoregression uses the observations from the previous time steps to predict the next time step. Input are the previous time steps which use to a regression equation to predict the value for the next time step. And the relationship between the variables are called correlation.

The correlation is the positive or negative correlation depends on the variable's natures. For example, if the both variables go up or down together that's mean in same direction together, this is called a positive correlation. If the one variable goes up and another variable goes down, that's mean in opposite directions, this is called the negative correlation.

Calculation of correlation between the output variable and previous time steps there is the statistical measures for. Calculations of correlation are in different time lags. If the correlations between the output variable and a specific lagged variable will be strong, the more weight can be used of autoregression on the variables when modeling.

Time series data structure is the sequential and using that data by autoregression which is also called serial correlation. The correlation statistics can also help to choose which lag variables will be useful in a model and which will not.

When we have the new dataset and the lag variables of that dataset shows low or no correlation with output variable, then it suggests that the time series problem may not be predictable (Brownlee J. , 2019)

The autoregression (AR) method models use the sequence of main prior time as a linear function of the observations.

It has some notation for the model to specifying for the model let's say p as a parameter to the AR function and the notation denote as AR(p). For example, AR (1) is a first order autoregression model (Sagar, 2019)

The method is suitable for univariate time series without trend and seasonal components.

Take a look an example of plotted below.

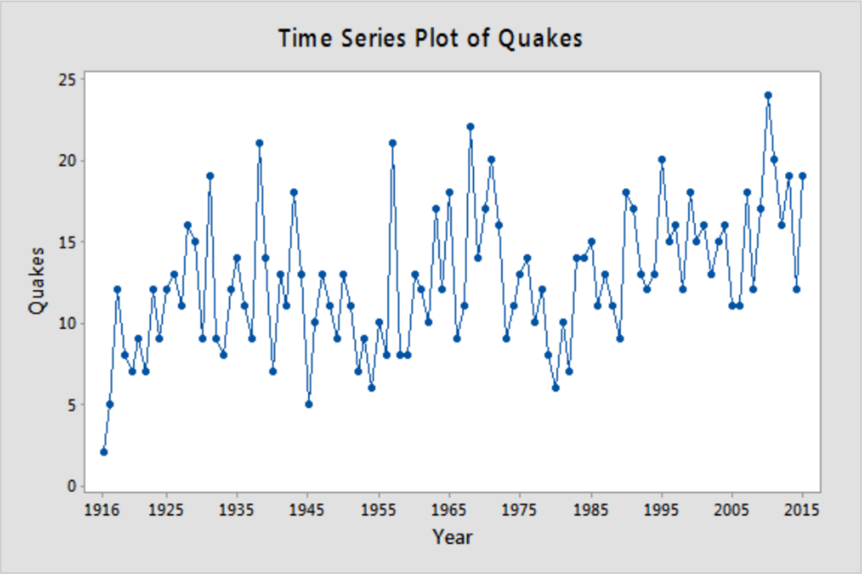


Fig 6: Autoregressive Models

Here in this figure we have the data of earthquake from USGS website. Let's plot the data. Let we have y_t that is the annual number of worldwide earthquakes with magnitude greater than 7 on the Richter scale for n = 100 years.

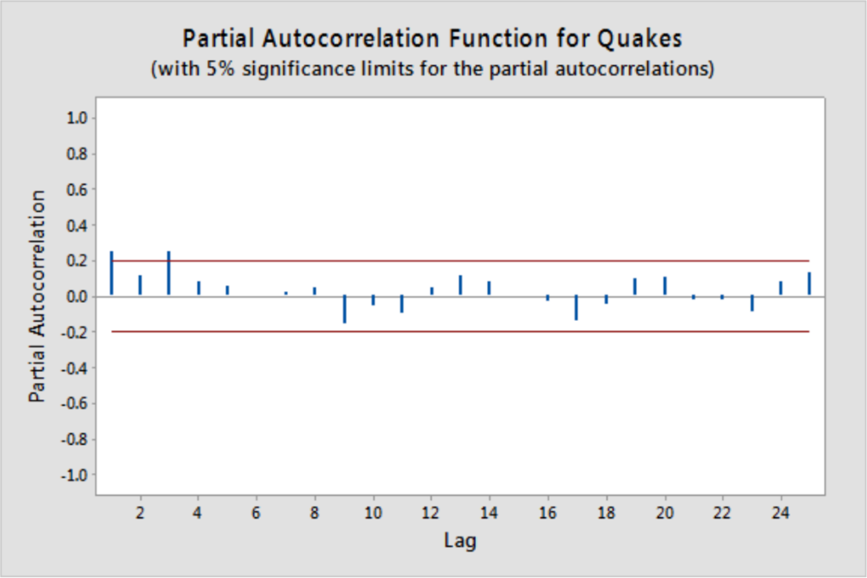


Fig 7: Partial Autoregressive Models

Here is the plot of Partial Autocorrelation Function in short PACF. Here is the interpretation of the mean which is the third order autoregression. Maybe it is the necessary since there are notable partial autocorrelations for lags 1 and 3.

Moving Average (MA) Model:

History of classical method of time series originated in 1920s. It was widely used until 1950s. It is still used for many time series decomposition methods. Moving average method is the one of them and it used to estimate the trend-cycle (Hyndman, 2018).

It provides the general idea how the trend of the data set looks like and going through it. It provides the average of any data set or subset of numbers. This is one of the most extremely useful to understand and start also for forecasting long-term trend, so it helps to calculate for any period of time. Let's take a look for a twenty-year period of sales data. In the dataset we can calculate a five-year moving average, a four-year moving average, a three-year moving average and so on. It is very helpful for the stock market analysts. Generally, they use the 50- or 200-days moving average so they can trend of the data and helps to forecast where the stocks are heading towards.

Average means the "middle" value of any set specially for numbers. Moving average concept is also the same. But here average is calculated in the subset of data set for several times. Let's take a look for the two years moving average of the data set from 2000 to 2003 serialization. So here the averages of subsets look like 2000/2001, 2001/2002 and 2002/2003. It is the best visualization of the data in plotted format (Glen, 2013).

In statistic viewpoint, moving average creates a series of averages. Series of averages come from the subset of data of the full dataset. Moving average also called rolling average or running average or moving mean (MM) or rolling mean. Moving average is a type of finite impulse response filter. It has variations, they are simple, and cumulative, or weighted forms. Here are the notified of two things "Moving" and "Average" and those had to be defined mathematically.

Moving is the additive operation. It adds every time as continuous format or in discrete format. Continuous is also called vector space so as discrete known as group. So, the additive operation takes in the "reference position" and add in the available space which moves.

Basically, *Average* is the **mean** of any dataset. Here the mean is created for "reference position" in the available space from the subset of the dataset. The first initial moving average is gained from the given fix subset size. Then after another mean is gained using "shifting forward" from another fix subset of data from the same dataset as serialization and added new value. It continues till the end of the series of numbers of dataset that mean, excluding the first number of the series and including the next value in the subset.

By using movie average find out the short-term fluctuations. Studying the short-term highlight, the longer-term trends or cycles. It is depending on the application and set of parameters for moving average to get the threshold between short-term and long-term. Let's take a look of financial data and moving average is used for technical analysis. Analysis might be financial data, stock prices, returns or trading volumes. Not only that moving average is used in economics. It examines the gross domestic product, employment or other macroeconomic time series. It is also quickly found out the selling trends that will be uptrend or a downtrend. Because of the pattern that captured by moving average, it is easy to find out the trend and easily predict for next moves.

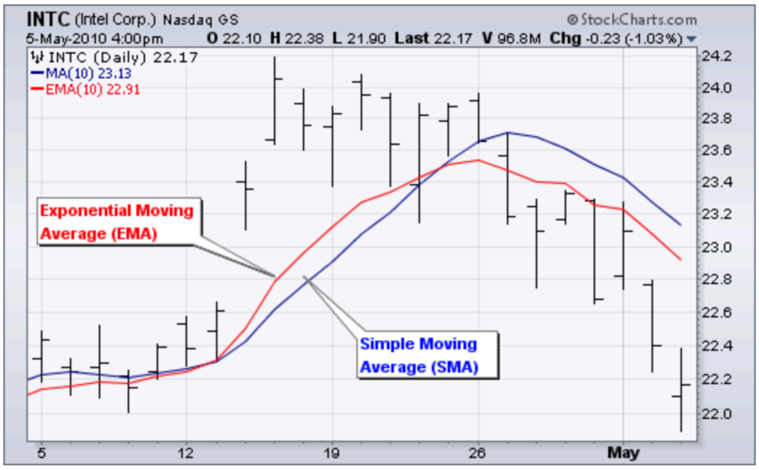
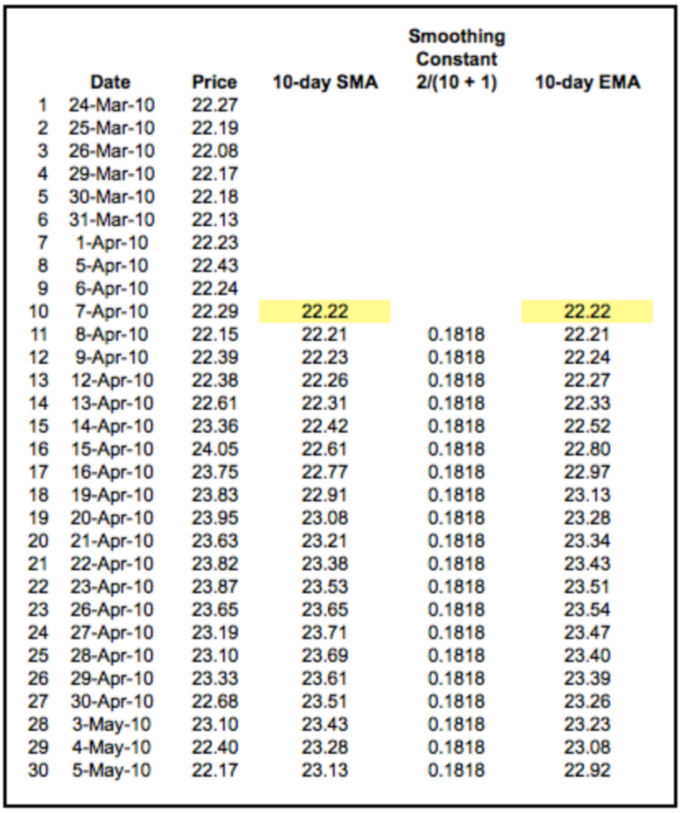


Fig 8: Moving Average

Moving average forecasting technique is also simple. Its process is calculated by adding up the last 'n' period's values and then dividing that number by 'n'. So, the moving average value is considering as the forecast for next period.



	Date	Price	10-day SMA	Smoothing Constant 2/(10 + 1)	10-day EMA
1	24-Mar-10	22.27			
2	25-Mar-10	22.19			
3	26-Mar-10	22.08			
4	29-Mar-10	22.17			
5	30-Mar-10	22.18			
6	31-Mar-10	22.13			
7	1-Apr-10	22.23			
8	5-Apr-10	22.43			
9	6-Apr-10	22.24			
10	7-Apr-10	22.29	22.22		22.22
11	8-Apr-10	22.15	22.21	0.1818	22.21
12	9-Apr-10	22.39	22.23	0.1818	22.24
13	12-Apr-10	22.38	22.26	0.1818	22.27
14	13-Apr-10	22.61	22.31	0.1818	22.33
15	14-Apr-10	23.36	22.42	0.1818	22.52
16	15-Apr-10	24.05	22.61	0.1818	22.80
17	16-Apr-10	23.75	22.77	0.1818	22.97
18	19-Apr-10	23.83	22.91	0.1818	23.13
19	20-Apr-10	23.95	23.08	0.1818	23.28
20	21-Apr-10	23.63	23.21	0.1818	23.34
21	22-Apr-10	23.82	23.38	0.1818	23.43
22	23-Apr-10	23.87	23.53	0.1818	23.51
23	26-Apr-10	23.65	23.65	0.1818	23.54
24	27-Apr-10	23.19	23.71	0.1818	23.47
25	28-Apr-10	23.10	23.69	0.1818	23.40
26	29-Apr-10	23.33	23.61	0.1818	23.39
27	30-Apr-10	22.68	23.51	0.1818	23.26
28	3-May-10	23.10	23.43	0.1818	23.23
29	4-May-10	22.40	23.28	0.1818	23.08
30	5-May-10	22.17	23.13	0.1818	22.92

Fig 9:10 days Moving Average Chart and Calculation Process

Moving average is like tricky part for mathematically, it is kind of twister that is why to it is used as low-pass filter taking as an example for signal processing. But when it is used in non-time series data its filters are higher frequency without using time. Because of its simplification the smooth data can be seen. Moving average smoothing the data even for the irregularities data of peaks and valleys so the trends can easily recognize (Wikiversity., 2018).

Autoregressive Moving Average (ARMA)

ARMA is the short form of Autoregressive Moving Average (ARMA) method models. It combines both Autoregression (AR) and Moving Average (MA) models (Brownlee, 2018). ARMA is a linear function of the observations with the residual errors at prior time steps.

To forecast by the ARMA model, there should be autoregression (AR) and moving average (MA) behaved well. They applied in ARMA model. They are the combined model of ARMA. ARMA model assumed that the time-series data is stationary. But when the data won't stationary it only performs uniformly around in a particular time period.

Since, ARMA assumed that the time-series data must be stationary, so the first thing, stationarity has to be defined in the sense of the behavior of the autocorrelation function (ACF). Before autocorrelation function, partial autocorrelation function (PACF) also being defined. It analyzed many simple cases of ARMA model. After then building and estimation of ARMA model can be developed and for that there are many a sequence of examples that already designed to demonstrate. And those are helped for selection an appropriate model to explain the evolution of an observed time series (Mills, 2019).

The ARMA concept is developed by Box and Jenkins (1970) for using the time series analysis method. In economic and financial theory has the explanatory variables to be defined but Box and Jenkins could not imply on those theory. But they used in time series based on the changing law of the time series itself. And the reason behind for developing is that the time series is stationary.

ARMA is one of the crucial for time series. Mostly it is used for market research for long term tracking data. Let’s take an example of retail research. So, it’s important to analysis the sales volume depends on the variation on the seasonal characteristic.

This model is among the high-resolution spectral analysis methods of the model parameter method, which is used in studying the rational spectrum of the stationary stochastic processes. It is more meaningful or suited for a large volume of class of practical problems.

The notation for the model involves specifying the order for the AR(p) and MA(q) models as parameters to an ARMA function, e.g. ARMA (p, q). An ARIMA model can be used to develop AR or MA models.

The method is suitable for univariate time series without trend and seasonal components.

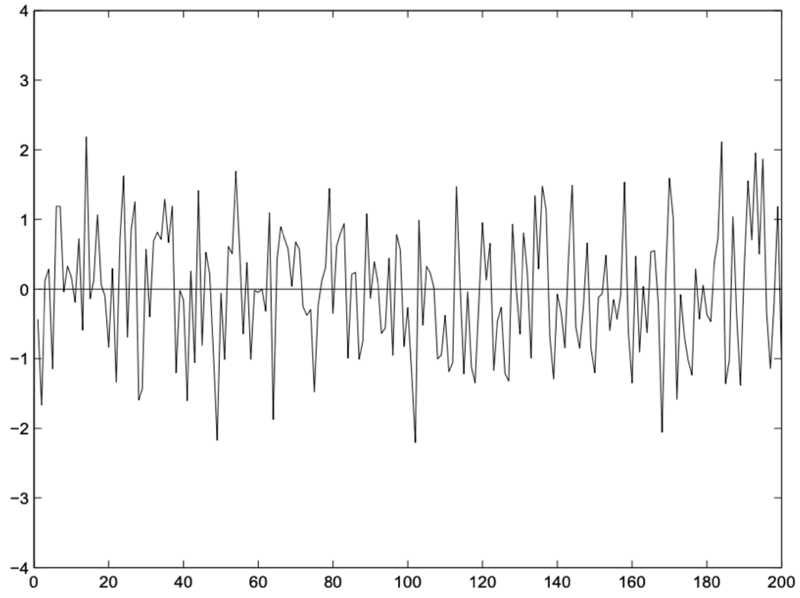


Fig 10: A Gaussian white noise time serie

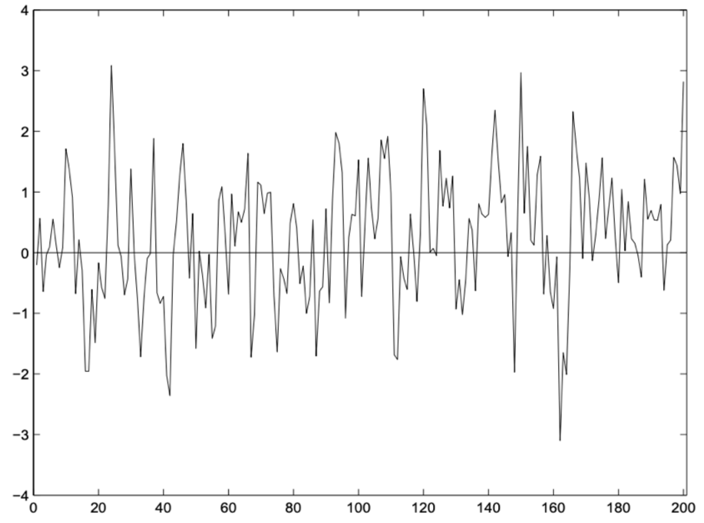


Fig 11: A simulated AR (1) process, with φ = 0.4

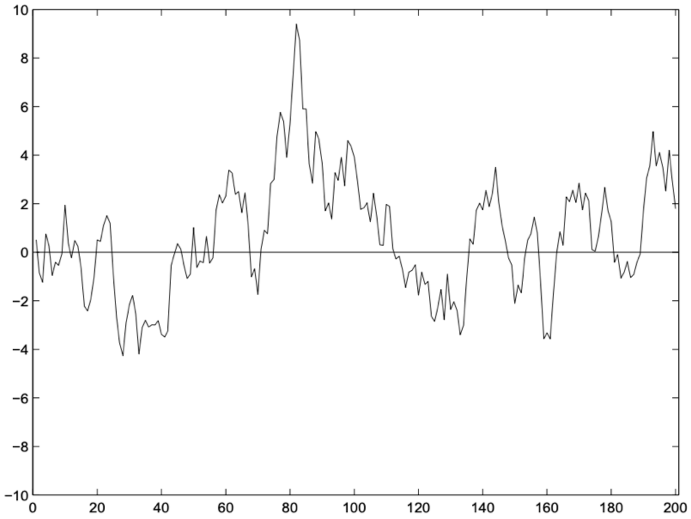


Fig 12: A simulated AR(1) process, with φ = 0.9

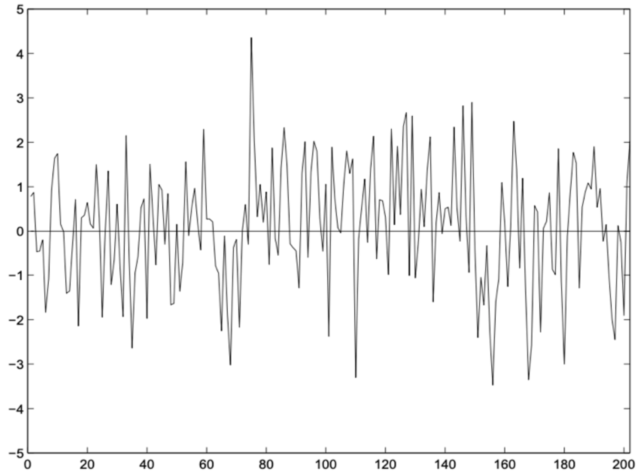


Fig 13: A simulated MA(3) process, with θ1 = 0.6, θ2 = −0.5, and θ3 = 0.4

ARIMA model:

ARIMA method is also introduced by Box and Jenkins (1976) and the similar idea for stationary time series using both autoregressive (AR) and a moving average (MA) component. Since it also handled the non-stationary for that it has to be reduced to stationarity beforehand by differencing the data (Chatfield, 2000).

ARIMA model used a time series data to statistically analysis and predict future trends. It is a form of regression analysis. It measures the one dependent variable and its related variables. So, it is easily found out the changes on related variables effect to the dependent variable.

The components of ARIMA makes easily to understand the model. Here are the components as follows:

- Autoregression (AR): it shows a changing variable that returns on its own lagged, or prior, values.
- Integrated (I): It is the difference of raw observations, so the time series become the stationary. Take an example that data values are replaced by the difference between the data values and the previous values.
- Moving average (MA): It takes the dependency between observation and residual error when the lagged observations.

Those all above components have the standard notation as a parameter. Those parameters for ARIMA model denoted as p, d, and q and each integer values in replace of those parameters denoted the type of ARIMA model used. Also, the parameters can be defined as:

- p: It is the lag order. It checks the number of lag observations in the model.
- d: It is known as the degree of differencing. It is differentiated the how many times or the number of times for the raw of observations.
- q: It is the order of moving average which is the size of the moving average window.

In a linear regression model will happen the number of type of terms. Let’s take an example, if zero is used as a parameter which means that a particular component is not used in the model so that defined the ARIMA may perform an ARMA model or even simple AR, I, or MA models.

ARIMA defines everything is consistency in order to change in data over time to make it stationary. But the economic and market data need the trends to follow to see the structure of data flow, so the purpose of differencing is to remove any trends or seasonal structures. But the seasonal or regular data can be predictable pattern over each year, it could bring the negatively effect on the regression model. That means if a trend appears than the stationarity won’t be clearly visible, so the result may not be good visible by the many computations process (Chen, J. 2019).

Seasonal Autoregressive Integrated Moving-Average (SARIMA)

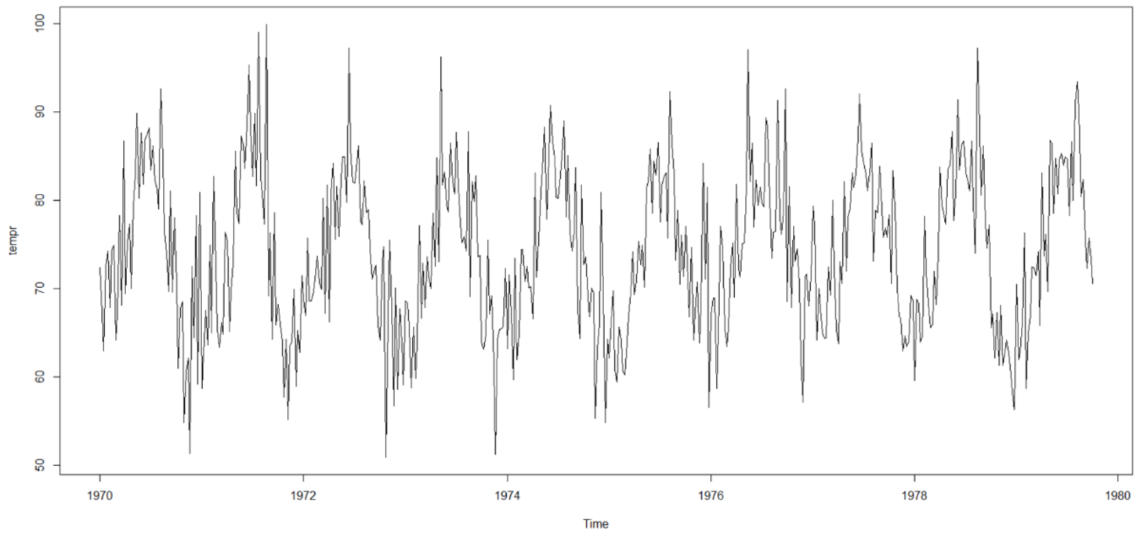
The Seasonal Autoregressive Integrated Moving Average or SARIMA is the method in the sequence of a linear function. It is differentiated the observations, errors, differenced seasonal observations, and seasonal errors at prior time steps.

It is the combin of the ARIMA model with seasonal level. ARIMA has the same autoregression, differencing, and moving average modeling now it included seasonal components (Brownlee J. , 2019). It supports explicitly univariate time series data with a seasonal component.

Unlike AIRMA, it also has three parameters but for seasonal component of the series. Those three parameters are the autoregression (AR), differencing (I) and moving average (MA). In addition, it has additional parameter for the period of the seasonality (Brownlee J. , 2019).

It is a worthy to analysis the seasonal data depends on seasonal effects. For an example the temperature can be measured as a seasonal basis. Basically, there are four seasons in a year. If analysis the temperature as per the yearly basis on a particular season, definitely there has found the strong correlation between the measured temperature on the same season (Foo, 2018).

Let's take a look in a plot of temperature cycle.



Yearly cycles observed with temperature data

Fig 14: Seasonal lags: SARIMA modelling and forecasting and Examples

The above plot has the cycle of yearly rise and fall temperature. Assuming that there is the 60-degree Fahrenheit in the end of the year, but the temperature will rise 80-degree Fahrenheit near the mid of the year.

There are two notations involved for this model one is the parameters for the ARIMA. They are AR(p), I(d), and MA(q). Another notation is the parameter for seasonal level. They are AR(P), I(D), MA(Q) and m parameters. In combined SARIMA (p, d, q) (P, D, Q) m where "m" stands for the number of time steps in each season (the seasonal period). SARIMA model can be used to develop AR, MA, ARMA and ARIMA models.

The method is suitable for univariate time series with trend and/or seasonal components.

Seasonal Autoregressive Integrated Moving-Average with Exogenous Regressors (SARIMAX)

Seasonal Autoregressive Integrated Moving Average with external variables in short SARIMAX model. SARIMAX forecast takes all the effects because the demand influencing factors also it includes the modeling of exogenous variables.

Exogenous variables are also known the covariates that means the independent variables in terms of statistic. SARIMAX uses the parallel input sequences which have observations of time steps same as the original time series. The first primary series may be referred to as endogenous data in order to contrast it from the exogenous sequence(s). Then the observations are directly included in the model for exogenous variables at each time step that means not changed which is in the same way as the primary endogenous sequence (e.g. as an AR, MA, etc. process).

The SARIMAX is also used to absorb other models with its exogenous variables, such as ARX, MAX, ARMAX, and ARIMAX.

The method is suitable for univariate time series with trend and/or seasonal components and exogenous variables (Brownlee J. , 2019).

Aburto & Weber, (2003, 2007) adopted the SARIMA model and included the neural network to forecast. They try to forecast the demand in a Chilean supermarket. They used the inputs neurons, as payment, intermediate payment, before holidays, holidays, festivals, school vacation, climate, price as variables for their forecasting model which is modeled by neural network for the errors of SARIMA process. They compared their proposed model with naïve, seasonal naïve, unconditional average, SARIMAX, and several neural network models.

SARIMAX model not only forecast the market demand, it is also used in diverse fields of applications as a forecasting tool. Cools et al. (2009) developed ARIMAX and SARIMAX models to forecast the daily traffic counts. They studied in the daily traffic data and its impact by holidays in the different locations when the seasonal changes and analyzed in their study. There are the better results from using ARIMAX and SARIMAX framework which are revealed from the inclusion of weekly seasonality and holiday effects at different site locations (Arunraj, 2016).

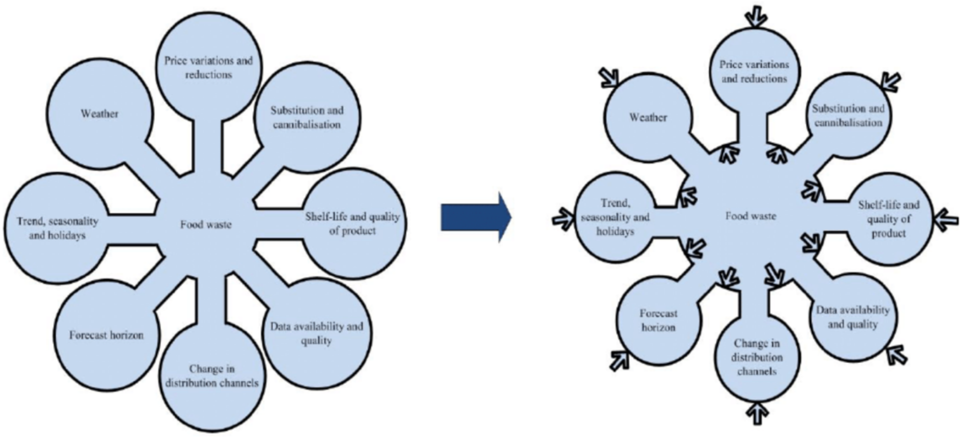

Price variations and reductions
Weather
Substitution and cannibalisation
Trend, seasonality and holidays
Food waste
Shelf-life and quality of product
Forecast horizon
Data availability and quality
Change in distribution channels
Price variations and reductions
Weather
Substitution and cannibalisation
Trend, seasonality and holidays
Food waste
Shelf-life and quality of product
Forecast horizon
Data availability and quality
Change in distribution channels

Fig 15: Factors influencing food waste

Vector Autoregression (VAR)

Actually, vector autoregression (VAR) is a variation of autoregressive models (AR) where we extend the autoregressive scheme and then convert to multiple variables with linear dependencies between them. That is why it is start from the from a univariate AR and then we will extend it to multiple variables (Alfonso, 2017).

VAR or Vector Autoregression is the forecasting algorithm used for the relationship between the time series. They are in bi-directional. Those time series influenced to two or more time series to each other. So, it can be said that it is it is a multivariate forecasting algorithm since it is used when two or more time series influence each other.

Since this method is suitable for multivariate time series but without trend and seasonal components (Brownlee J. , 2019).

That means, the basic requirements in order to use VAR are:

- You need at least two time series (variables)
- The time series should influence each other.

Let's consider as an Autoregressive (AR) model where each variable of time series is modeled as a function of the past values. So, the predictors are time delayed value or can say the lags variables of the series. Then how is VAR different from other Autoregressive models like AR, ARMA or ARIMA?

The main difference is those models are uni-directional and bi-directional. In uni-directional the predictors or exogenous variables influence the Y or endogenous variables and not effected by both variables that's mean vice-versa. Whereas, Vector Auto Regression (VAR) is bi-directional. That is, the variables endogenous and exogenous variables influence each other.

Each variable of VAR model modeled depend on **linear combination of past values of itself and the past values of other variables in the system**. Since in the multiple time series that the endogenous and exogenous variables influence each other, it is modeled as per a system of equations with one equation per variable in time series (Prabhakaran.S., 2019).

A vector autoregression (VAR) model is a multivariate time series model that mean it contains a system of 'n' equations of 'n' distinct. And the stationary response variables depend on the linear functions. Linear function has the lagged responses and other terms. The notation 'p' as the degree is also characterized of the VAR model. Using notation 'p', let's say each equation in a VAR (p) model contains 'p' lags of all variables in the system (Mathworks, 2020).

The next step of VAR model uses the AR model in each time series. This generalize the AR to multiple parallel time series, for example multivariate time series. The notation for the VAR model uses the specifying the order for the AR (p) model as parameters to a VAR function which is VAR (p).

The vector autoregression (VAR) model is very easy to use because of its flexible for the analysis of multivariate time series. Univariate autoregressive model is a natural extension process which makes multivariate time series dynamized. Since VAR model described the dynamic behavior so it is especially useful for dynamic behavior of economic and financial time series and for forecasting. Due to dynamic behavior, it also provides the superior forecasts. That means forecast from univariate time series models and then elaborate to theory-based simultaneous equations models. It has the conditional of specified variables which makes VAR model to be more flexible to forecast on the potential future paths.

Not only VAR model describes the data and forecasting, it is also used for structural inference and policy analysis. And in structural analysis has some certain assumptions about the causal structure of the data, which is under investigation are forcedly to make done. So, the results are all summarized. The results can be causal impacts of unexpected shocks, or innovations to specified variables on the variables in the model. These causal impacts that are mostly summarized with impulse response functions and forecast to error variance which is decomposition (faculty.washington, 2020).

In econometrics, VAR is widely well-known and used model. It is the key point of VAR model that the values are differentiated between the value of a variable at a time point which depends on linearly and the value of different variables at previous instants of time. Let's take an example of the birth rate. The births numbers might be predicted in a given month which is from the fertile population value of nine months earlier.

There's some computational cost for these kinds of models. It's a fortunate, there's the computing capacity which is apply to the various other machine learning to huge datasets. Furthermore, this is not the limit to apply only to econometrics but also to different fields like health or weather or simply to any problem that works with time series.

Vector Autoregression Moving-Average (VARMA)

Vector autoregressive moving average (VARMA) processes is also a part of linearly regular processes using with a wide range of applications, but it is more a flexible. In the case of parsimonious parametrization VARMA model is more suitable comparison than VAR models. However, comparison VARMA and VAR processes, VARMA estimation is more harder and the relation between internal parameters and external characteristics like the autocovariance function is more involved but in general the maximum likelihood method only needs numerical optimization (Scherrer, 2019).

In every next step VARMA method model used the ARMA model in each time series. VARMA is the generalization of ARMA model which multiple parallel time series like a multivariate time series and the method is more suitable for multivariate time series but without trend and seasonal components (Brownlee J. , 2019).

VARMA has the notation and it has the specifying the order like the AR (p) and MA (q) models are as parameters to a VARMA function. Using full notation seem like this: VARMA (p, q). VARMA model not only used the ARMA it can also be used to develop VAR or VMA models.

Theoretically VARMA model expected that VAR is used in a high order so the VARMA structure will approximate the true. But the recent literatures result tells different things that it might cause the problem. Fern´andez-Villaverde et al. (2007), and Chari, Kehoe and McGrattan (2007) examined to (S)VAR of conditions. The conditions are under the economic shocks and impulse responses are from which an economic model. The researchers bring the light upon the result and conclude on their analysis. Their result conclude that the available current data must prohibited that misleading to VARs. Because it is too short of a lag length which provide the poor approximations and unreliable inferences.

In addition, Kapetanios, Pagan and Scott (2007) have suggested on VAR to get the sample of 30000 which have 50 orders required if to get the adequately capture the effect. The effects might be the structural shocks in order to a data dynamic and DSGE elements both has for a simulated model.

Ravenna (2007) also has points of using VAR and warns to researchers that VAR model misleading to convert characterize in the dynamics model which is lead to a VARMA structure. So that could be problematic and be cautious on relying on evidence from VARs to build such models.

VAR model has the limitations even if it is well framed documented. Instead macroeconomic researchers compelled to use VARMA model. But the complexities to identification and estimation of VARMA models, practitioners hesitate to use it. In contrast VAR model is easy and accessibility.

Hannan and Deistler (1988) and L¨utkepohl (2005) state that about multivariate time series model that it has many methods and techniques are available to solve problems in the past and one of the noticeable series. But here the question arises that how to determine internal structure of a VARMA model, if in a direct and straightforward manner has not been completely resolved.

Here are two techniques of identification predominate:

1. **The scalar-component methodology**: This methodology is main pioneered by Tiao and Tsay (1989), and further it is developed in Athanasopoulos and Vahid (2008). This method adapts the canonical correlation analysis which is introduced in Akaike (1974b) to detect various linear dependencies which implied by different structures. There is the solution of different eigenvalue problems and which solves the underlying multiple decision problem via a sequence of hypothesis tests are believed by this method.
2. **The echelon form methodology**: This methodology is developed in Hannan and Kavalieris (1984) and Poskitt (1992). This approach determined the Kronecker indices using the regression techniques also it uses the model selections criteria those are AIC (Akaike, 1974a), BIC (Schwarz, 1978) or HQ (Hannan and Quin, 1979). With this approach, the coefficients of a VARMA model are expressed in echelon canonical form which are also estimated. (Poskitt, 2011).

In macroeconomic theory, VARMA is link with linearized dynamic stochastic general equilibrium (DSGE) models (Kascha, 2012; Fern´andez-Villaverde et al., 2007). Notation of VARMA d (p, q) model describes a stationary d-dimensional vector time series where y_t is modeled as a function of its own so as p past values and q lagged error terms.

Because of identifiability problem and lots of challenges comes during computation VARMA models used less used in practice. Because of this Vector Autoregressive (VAR) models are primary choice of multivariate time series rather than VARMA models. A VARMA is the extension of VAR model and there is the special case of the VARMA in which the time series are only modeled as a function of their own 'p' past values and there are no moving average coefficients are included. VAR models are used in different applications and they are found in diverse fields such as biostatistics (e.g., Kirch et al. (2015)), finance (e.g., Tao et al. (2011)), economics (e.g., Matteson and Tsay (2011)), and marketing (e.g., Gelper et al. (2016)). If in those above fields using large-scale multivariate time series data which are used commonly and increasing way, there are two conditions that VAR is follow for and they are:

1. understanding how the component time series interact with each other
2. increasing forecast accuracy by using information on interactions among multiple time series.

However, classical time series theory suggests that the more general VARMA models can be equally effective, or even better, for achieving these objectives compared to VAR models (Wilmsa. I, 2019).

Vector Autoregression Moving-Average with Exogenous Regressors (VARMAX)

The Vector Autoregression Moving-Average with Exogenous Regressors in short VARMAX is an extension of the VARMA model. It also includes the modeling of exogenous variables. VARMAX uses in a multivariate time series without trend and seasonal components with exogenous variables of the ARMAX method (Brownlee J. , 2019)

Exogenous variables which are also called covariates. They are used the parallel input sequences and the parallel between observations and the original series. The main series are differentiated between endogenous data and exogenous data. The observations for exogenous variables can directly include in the model at each time step. Even though it is not modeled in the same way as the primary endogenous sequence for example AR, MA, etc. process. There are some other models such as VARX and VMAX which are also used by VARMAX method, but it is the subsumed models with exogenous variables.

VARMAX is a multivariate time series to procedure and estimates the model parameters. It generates forecasts which is associated with vector autoregressive moving-average (VARMA) processes with exogenous regressors models. Let's have a look that economic and financial variables. They are often not correlated or happening in the same time to each other, but they are also correlated to each other's past values. The VARMAX procedure can be used to model in these types of time relationships. The endogenous variables of economic and financial application influenced by external variables in the system. The endogenous variables also said dependent or response. So as exogenous variables are also said external, independent, input, predictor or regressor variables. The VARMAX procedure provides the dynamic relationship to both between the dependent variables and also between the dependent and independent variables to the model.

VARMAX models are in orders of the autoregressive or moving-average process or can be both processes. Those orders can be specified by options or they can be automatically determined when using the VARMAX procedure.

Criteria to follow for automatically determining these orders are listed below:

- Akaike's information criterion (AIC)
- corrected AIC (AICC)
- Hannan-Quinn (HQ) criterion
- final prediction error (FPE)
- Schwarz Bayesian criterion (SBC), also known as Bayesian information criterion (BIC)

But if using the automatic order selection there are the list provided by VARMAX procedure for autoregressive order identification aids:

- partial cross-correlations
- Yule-Walker estimates
- partial autoregressive coefficients
- partial canonical correlations

VARMAX has some of the tests to aid for determining the presence of unit roots and cointegration in the time series for the stationarity. These tests include the following:

- Dickey-Fuller tests
- Johansen cointegration test for nonstationary vector processes of integrated order one
- Stock-Watson common trends test for the possibility of cointegration among nonstationary vector processes of integrated order one
- Johansen cointegration test for nonstationary vector processes of integrated order two

VARMAX procedure provides for vector autoregressive and moving-average (VARMA) and Bayesian vector autoregressive (BVAR) models to those data which is stationary vector time series. If the data is not nonstationary series that there is the procedure to make nonstationary to stationary by using appropriate differencing. Here VARMAX procedure provides both the vector error correction model (VECM) and the Bayesian vector error correction model (BVECM) for solving problem. The problem will be high dimensionality in the parameters of the VAR model.

There are the Bayesian models which are used when prior information about the model parameters is available. Since VARMAX is the multivariant time series, VARMAX procedure also allows independent or exogenous variables with their distributed lags to influence dependent or said endogenous variables in various models for example VARMAX, BVARX, VECMX, and BVECMX models.

Since, multivariate time series' main object is forecasting for that there's another step of fitting model. Fitting model are VARMAX, BVARX, VECMX, and BVECMX. After fitting the model, the VARMAX procedure next step to compute prediction values based on the parameter estimates and the past values of the vector time series (support.sas, 2020).

The model parameter estimation methods are the following:

- least squares
- maximum likelihood

Exponential Smoothing (SES):

To get the immediate result of forecasting exponential smoothing (SES) is widely used procedures for smoothing discrete time series. Because of its qualities like its simplicity, its computational efficiency, its ease of adjusting its responsiveness to changes in the process being forecast, and its reasonable accuracy.

The main idea behind of the exponential smoothing is to get the smooth the original series the same way as the moving average does. So, using the smoothed series in forecasting get the future values of the variable. In exponential smoothing allow the more recent values of the series which have impact the greater influence on the forecast of future values comparison to the more distant observations.

Exponential smoothing is the simple and practical approach which helps to forecast. The forecast is constructed from an exponentially weighted average of past observations. There are the sequences of weights have been provided. The less weight is given to present observation the immediately preceding will happen that means exponential decay of influence of past data.

This forecasting method is most widely used of all forecasting techniques. It requires little computation. This method is used when data pattern is approximately horizontal (i.e., there is no neither cyclic variation nor pronounced trend in the historical data).

The idea exponential smoothing is the most recent observations which will usually provide the best guide to the future. So, the weighting scheme that has decreasing weights when the observations get older. The choice of the smoothing constant is to determine the operating characteristics of exponential smoothing. If there's the smaller the value of α there will be the slower response. As contrast larger values of α because there will be more chances of the smoothed value to react quickly. But not only to real changes but also random fluctuations will also impact. There is the simple exponential smoothing model. This model is only good for non-seasonal patterns. There is approximately zero-trend for short-term forecasting. Because, if there will extend past in the next period, there might be the forecasted value for that period has to be used as a surrogate for the actual demand for any forecast for past the next period. So, as a result, there is no need to add corrective information that means the actual demand and any error grows exponentially (Ostertagova. E & Ostertag. O, 2011).

Exponential smoothing is a more “smoothing” out of the data because of removing much of the “noise” that is the random effect from the data by giving a better forecast.

There are three types of Exponential Smoothing (SES). They are as follow:

i. **Simple Exponential Smoothing:**
   Simple exponential smoothing described using an additive model with constant level but no seasonality then it can be used to make short-term forecast
ii. **Holt's Exponential Smoothing:**
   A time series using an additive model which has increasing or decreasing trend but no seasonality then there can be used Holt's exponential smoothing to make short-term forecasts.
iii. **Winters' Three Parameter Linear and Seasonal Exponential Smoothing:**
   Combining of both that means if this model used both first an additive model with increasing or decreasing trend and second seasonality then there can be used Holt-Winters exponential smoothing to make short-term forecasts.

Long-Short Term Memories (LSTM):

Long-Short Term Memories is called LSTM in short. It is used to forecast the time series data base on RNN. It is a type of Recurrent Neural Networks (RNNs). LSTM has the unique ability to learn and remember over long sequences of input data through the use of “gates” which regulate the information flow of the network.

LSTM has the state and it is represented a state space vector. The state tracks the dependencies of every changing new observations with past ones. LSTM mostly used the unstructured data like audio, video, text etc. It will get more benefit from transfer learning techniques even if it applied to standard time series.

LSTMs has the following limitations of RNNs.

- Short-term memory — Earlier time information are discarded when moving forward to next steps this will cause the loss of important information.
- Vanishing gradient — The term gradient is the value used. It is used to update the weight used in a neural network. If a gradient value becomes extremely small, it doesn't contribute too much to learning. In the vanishing gradient problem, gradient shrinks as it back propagates through time.
- Exploding gradient — This occurs when the network assigns unreasonably high importance to the weights.

Both RNNs and LSTMs has the same type of behavior for passing data. They both have the propagates forward data. But LSTMs makes use of gates to decide if it should keep or forget information different approach then RNNs.

An LSTM cell has two types of stage one is “a cell state and input” another one is “forget and output gates” both make use of several activation functions (Krishni., 2019).

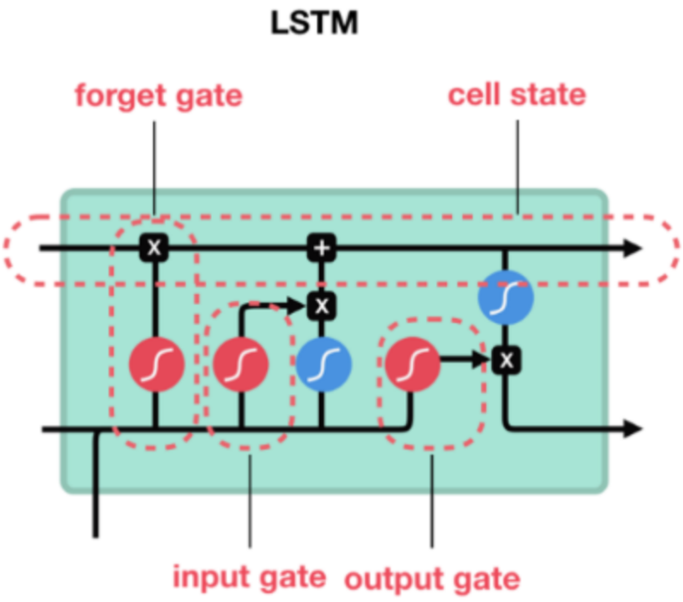


Fig 16: 1 A High-Level Introduction to LSTMs

- Forget gate — Decides which information should be kept and which should be discarded.
- Input gate — Updates the cell state.
- Output gate — Decides what the next hidden state(contains information on previous inputs) should be.
- Cell state — Acts as a highway that transports relative information along the sequence chain.

The two activation functions used,

- Sigmoid — squishes values between 0 and 1.
- Tanh — squishes values between -1 and 1.

Practical Part

UML Diagram

Simply UML stands for Unified Modeling Language. UML is a visually clear modern approach. In fact, it is the modeling and documenting software. In the software world it is the most popular business process modeling techniques.

Software has lots of components. Generally, people could not understand all the return code. So, all those components are represented as in diagram form is the main objective of UML. As the old proverb says: “a picture is worth a thousand words”. It is the clear and better understanding by the visual and the flow. So, all those visual representations help to understand the possible flaws or errors in software or business processes.

First UML was created in confusing manner as documentation. There were several visual representations introduced to document the software systems in 1990s period. The need for this system goes much more and final result comes in 1994-1996 in more unified way to visually. For this system was developed by three software engineers working at Rational Software. After that it was officially adopted in 1997 and it still remained the standard ever since, but few updates are included as the time periods.

What is the use of UML?

Main purpose of UML is the modeling for software engineering. Now in a different angle and different way of using as the documentation of workflow or process for the business or the software world. Let’s take an example of using activity diagram which is a type of UML diagram. If the activity diagram will replacement against flowchart, it has the better result comparison to flowchart. It provides the more standard modeling workflow and better readability and efficiency features.

Here some of the different uses of UML in software development and business process documentation:

- **Sketch:** UML diagrams in this level is used for overall view. It used before the coding and to communicate different aspects and characteristics of a system. Probably it will not include all the necessary details to execute the project until the very end.
- **Forward Design:** This is developed for the better view for the system or the workflow for the business to get the better ideas. Thus, this way many design issues or flaws can

be revealed. So, those issues can improve in time and make the overall project health and well-being.

- **Backward Design:** UML diagram is converted the code in different diagrams as a form of documentation for the different activities, roles, actors, and workflows.

Types of UML Diagram

There are different types of UML diagram depends on its uses and purposes. So, it can be designed before implementation or after (as part of documentation).

There are mostly used two types of UML diagram and they are **Behavioral** UML diagram and **Structural** UML diagram. Structural diagram illustrates the structure of a system or process. Behavioral diagram describes the behavior of the system like its actors, and its building components.

The different types of diagrams are categorized as below:

Behavioral UML Diagram

1. Activity Diagram
2. Use Case Diagram
3. Interaction Overview Diagram
4. Timing Diagram
5. State Machine Diagram
6. Communication Diagram
7. Sequence Diagram

Structural UML Diagram

1. Class Diagram
2. Object Diagram
3. Component Diagram
4. Composite Structure Diagram
5. Deployment Diagram
6. Package Diagram
7. Profile Diagram

Here are some of them are illustrated based on the practical part of web application.

Activity Diagram

Activity diagrams describe the flow of different activities and actions. Those activities or actions can be both sequential and in parallel. They describe the objects used, consumed or produced by an activity and the relationship between the different activities.

In the below activity diagram display the flow of the activities of this thesis. Data load from the USGC server that stored in csv format than user request for the specific date for the next activity. The requested date read by server, process and calculation than fetch and finally display data in the webpage.

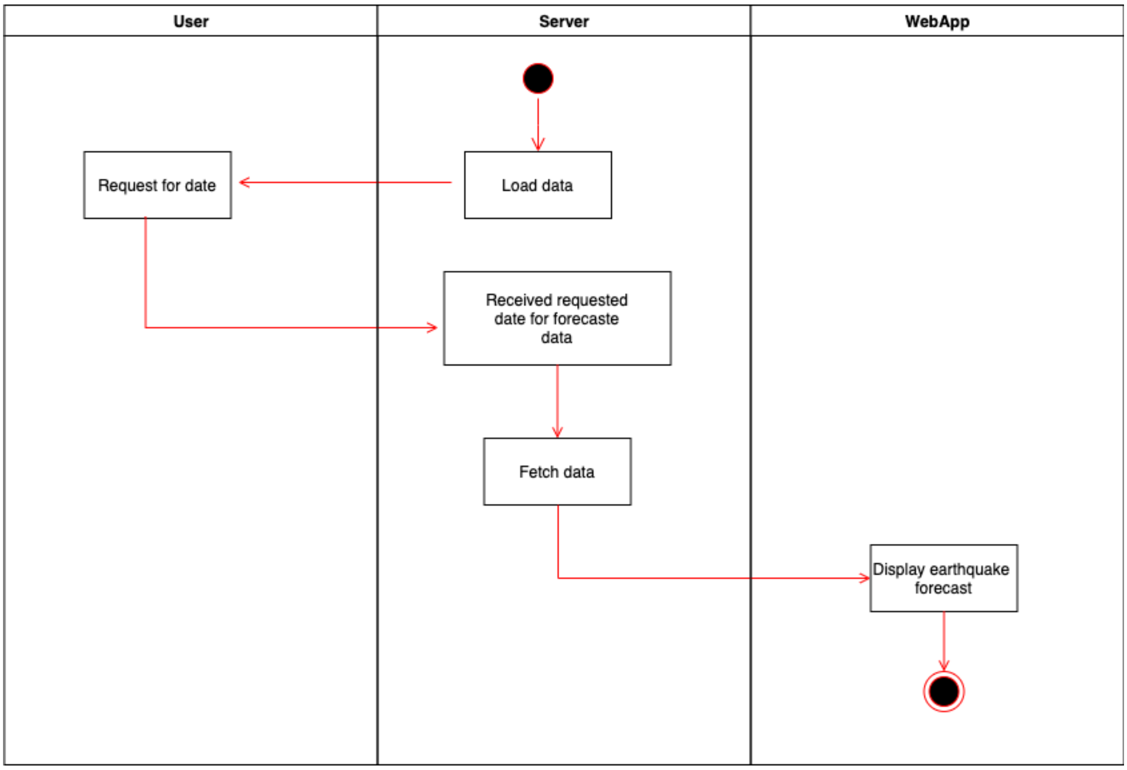

User
Server
WebApp
Request for date
Load data
Received requested date for forecaste data
Fetch data
Display earthquake forecast

Fig 17: Activity Diagram for earthquake application

Use Case Diagram

Main fundamental part that required for the system is the functional requirement which fulfill by system. Use case diagram fulfill this requirement where it is used to analyze the system’s high-level requirements. Using different Use Cases, these requirements can be fulfilled.

There are the **three** main components of this UML diagram:

- **Functional requirements** – It represent the use cases. In use cases a verb describs an action.
- **Actors** – An actor can be anything, a human being, an organization or an internal or external application. They interact with the system.
- **Relationships** – It establishes the relationship between actors and use cases. In diagram it represented straight arrows.

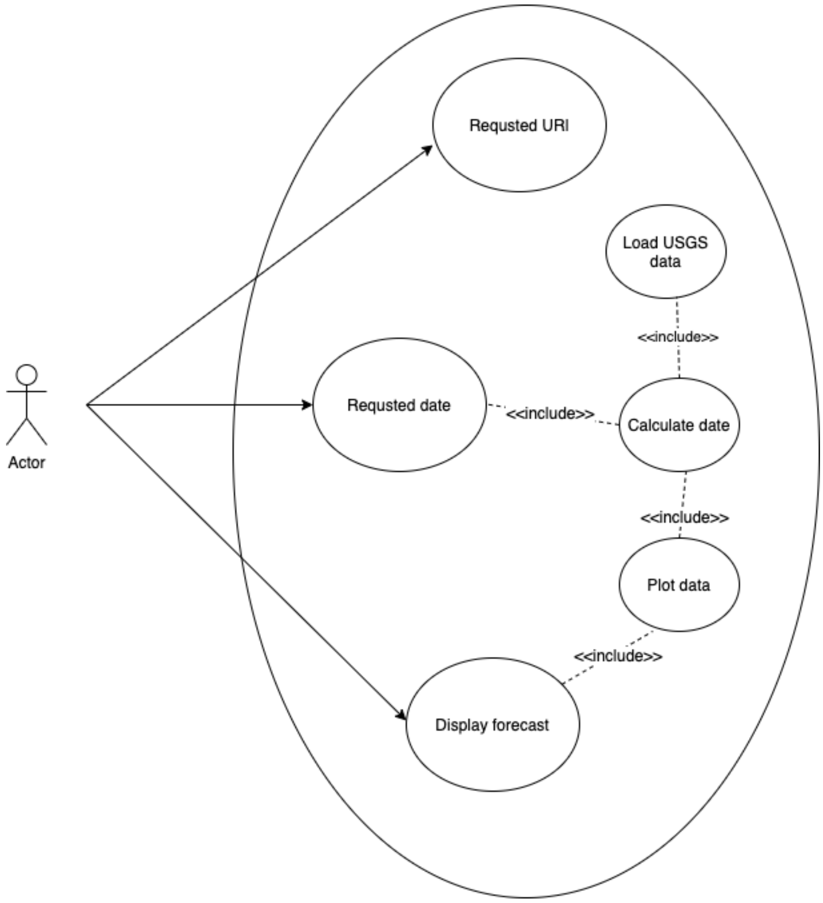


Fig 18: Use case diagram for earthquake application

The above UML diagram represent of this thesis. There might be one or more users, but the user uses the app only one at the time. The diagram clearly shows the function flows and their relationships using different notations like straight arrows, <<include>> and connectivity, and actor.

State Machine UML diagram

State machine UML diagrams used for the different states of a component within a system. It is also known as State chart diagrams. State machine describes the several states of an object and how it changes based on internal and external events.

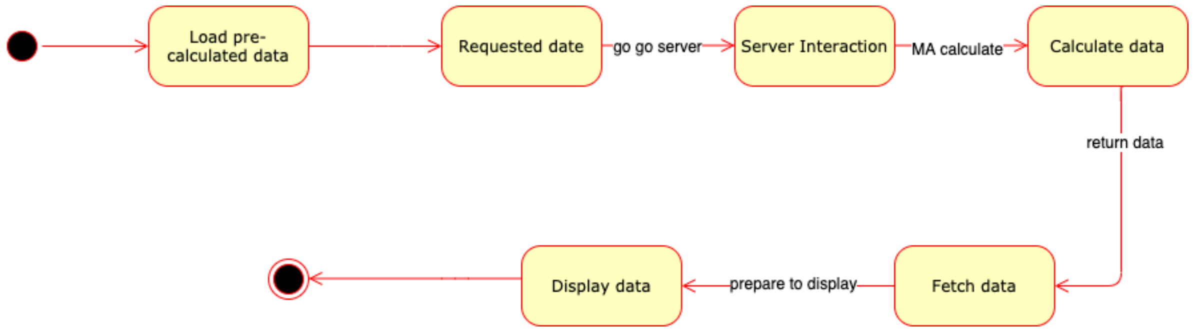


Fig 19: State machine UML diagram for earthquake application

As described the state changes based on internal and external events the above thesis diagram also displays the various states from start to end that how the states are flowed and changed steps by steps from begin to finish state like loading date, requested data, interaction with server, than the data are calculated from the based on request, fetch data from the calculated data and in the end display data to user.

Sequence UML Diagram

Sequence UML diagram is the sequential flow of the messages of interactions that what is happening between actors and objects. Activation depends on another object only if that object wants to communicate with actors or objects. All communication is represented in a chronological manner.

Sequence or structural diagrams illustrate the structure of a system. This is specially used in software development to represent the architecture of the system. So, it is easy to visualize that how the different components are interconnected that does not mean how they behave or communicate, simply where they stand.

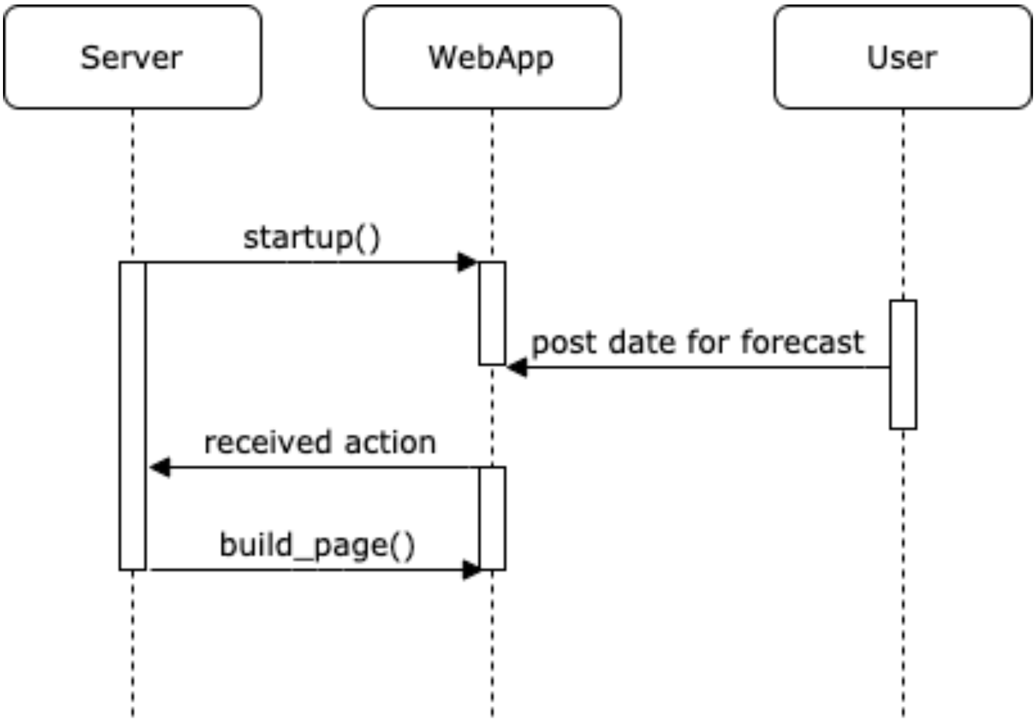


Fig 20: Sequence diagram for earthquake application

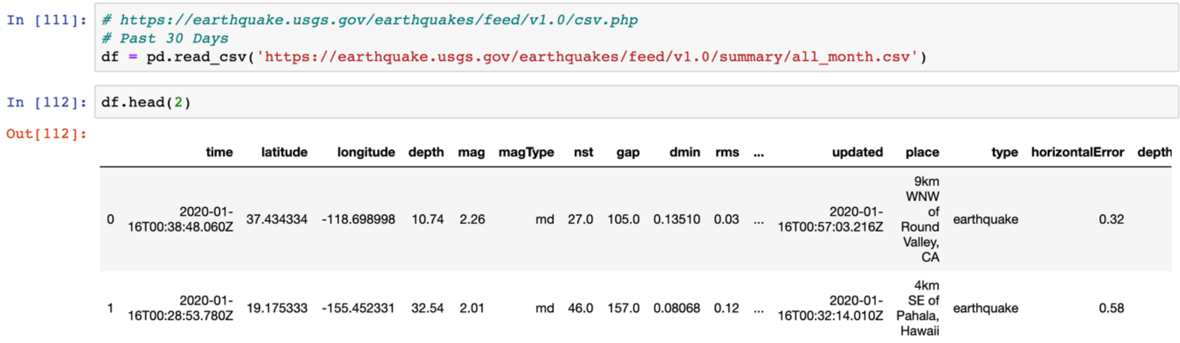
The above diagram represents for this thesis application. This shows the flow and interconnection with server, app and user. The sequential way it flows the activities. There is the function of startup() called from the server so it load in the server than send the data to app. After user called the requested date for related data on that date to the app. The app again sends the received action to server and call the build_page() function and display data to application.

Analysis and Visualizations:

In this section the data fetch from the USCS of U.S will import, analysis and visualization than construct the module, split the data into train and test than prediction and find the accuracy of the prediction so the module will predict fine.

Earthquake before feature engineering

First of all the data import from the source[usgs] and test the content so the graph can plot easily.

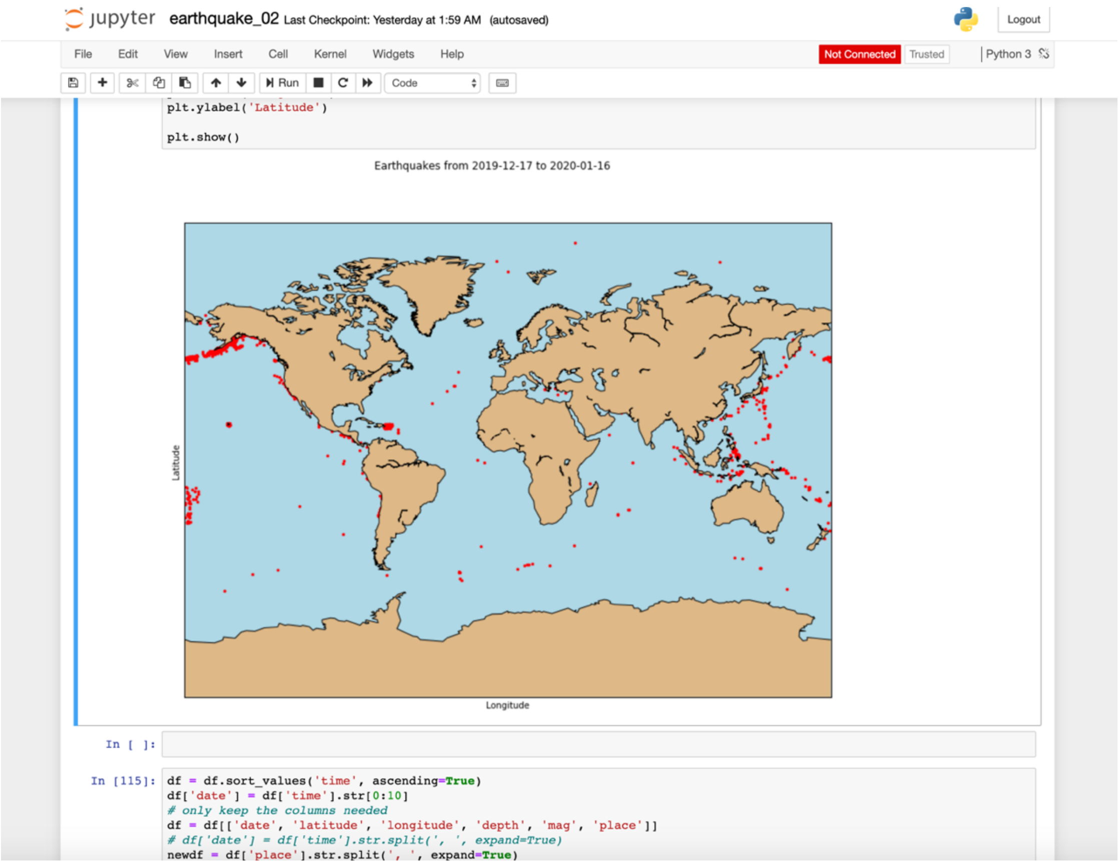


```
In [111]: # https://earthquake.usgs.gov/earthquakes/feed/v1.0/csv.php
          # Past 30 Days
          df = pd.read_csv('https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_month.csv')

In [112]: df.head(2)
```

Out[112]:

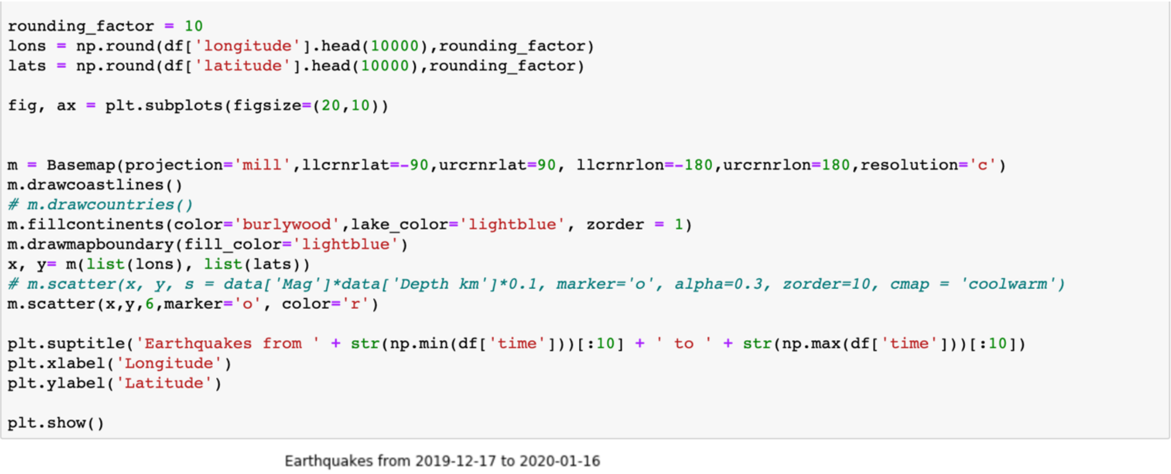
	time	latitude	longitude	depth	mag	magType	nst	gap	dmin	rms	...	updated	place	type	horizontalError	depth
0	2020-01-16T00:38:48.060Z	37.434334	-118.698998	10.74	2.26	md	27.0	105.0	0.13510	0.03	...	2020-01-16T00:57:03.216Z	9km WNW of Round Valley, CA	earthquake	0.32	
1	2020-01-16T00:28:53.780Z	19.175333	-155.452331	32.54	2.01	md	46.0	157.0	0.08068	0.12	...	2020-01-16T00:32:14.010Z	4km SE of Pahala, Hawaii	earthquake	0.58	

After checking the content of the imported data, it is easy to plot the into the diagram. Here is the code for plotting data into map using Basemap.


jupyter
earthquake_02 Last Checkpoint: Yesterday at 1:59 AM (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Connected
Trusted
Python 3
Run
Code
Earthquakes from 2019-12-17 to 2020-01-16
Latitude
Longitude

```
plt.ylabel('Latitude')

plt.show()
```

```
In [ ]:

In [115]: df = df.sort_values('time', ascending=True)
          df['date'] = df['time'].str[0:10]
          # only keep the columns needed
          df = df[['date', 'latitude', 'longitude', 'depth', 'mag', 'place']]
          # df['date'] = df['time'].str.split(', ', expand=True)
          newdf = df['place'].str.split(', ', expand=True)
```



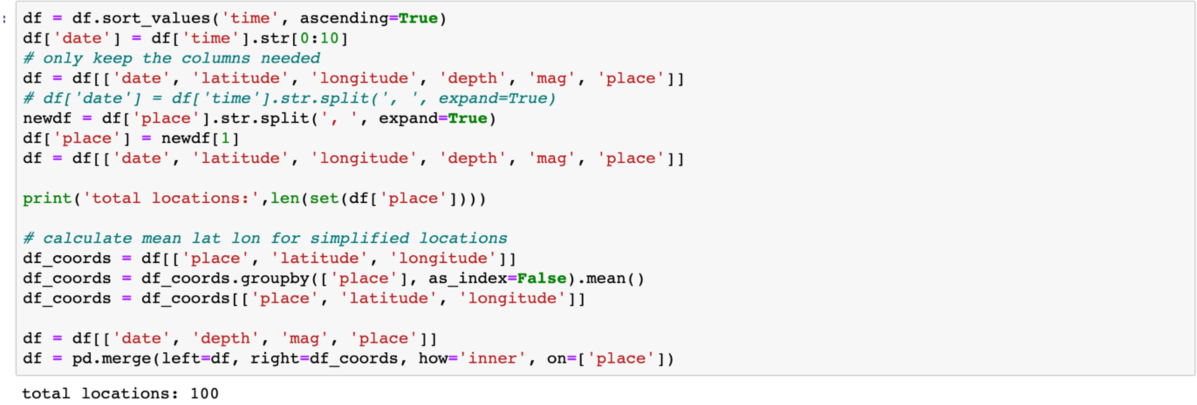
```
rounding_factor = 10
lons = np.round(df['longitude'].head(10000),rounding_factor)
lats = np.round(df['latitude'].head(10000),rounding_factor)

fig, ax = plt.subplots(figsize=(20,10))


m = Basemap(projection='mill',llcrnrlat=-90,urcrnrlat=90, llcrnrlon=-180,urcrnrlon=180,resolution='c')
m.drawcoastlines()
# m.drawcountries()
m.fillcontinents(color='burlywood',lake_color='lightblue', zorder = 1)
m.drawmapboundary(fill_color='lightblue')
x, y= m(list(lons), list(lats))
# m.scatter(x, y, s = data['Mag']*data['Depth km']*0.1, marker='o', alpha=0.3, zorder=10, cmap = 'coolwarm')
m.scatter(x,y,6,marker='o', color='r')

plt.suptitle('Earthquakes from ' + str(np.min(df['time']))[:10] + ' to ' + str(np.max(df['time']))[:10])
plt.xlabel('Longitude')
plt.ylabel('Latitude')

plt.show()
```

Earthquakes from 2019-12-17 to 2020-01-16

This is the one month illustrate data of earthquake without any feature engineering. Small dots mean low earthquake whereas big dots means high earthquake.

Feature Engineering:

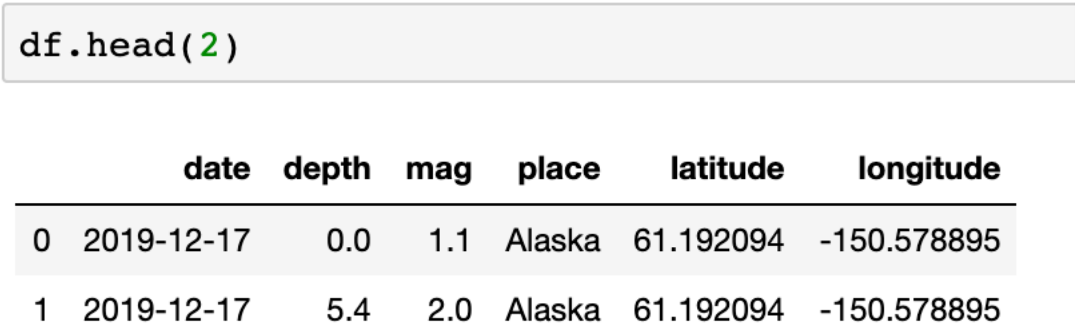
Here the data is filtered. Unnecessary columns are removed and used only necessary dependent and independent variables. After that data are grouped by place.



```
df = df.sort_values('time', ascending=True)
df['date'] = df['time'].str[0:10]
# only keep the columns needed
df = df[['date', 'latitude', 'longitude', 'depth', 'mag', 'place']]
# df['date'] = df['time'].str.split(', ', expand=True)
newdf = df['place'].str.split(', ', expand=True)
df['place'] = newdf[1]
df = df[['date', 'latitude', 'longitude', 'depth', 'mag', 'place']]

print('total locations:',len(set(df['place'])))

# calculate mean lat lon for simplified locations
df_coords = df[['place', 'latitude', 'longitude']]
df_coords = df_coords.groupby(['place'], as_index=False).mean()
df_coords = df_coords[['place', 'latitude', 'longitude']]

df = df[['date', 'depth', 'mag', 'place']]
df = pd.merge(left=df, right=df_coords, how='inner', on=['place'])
```

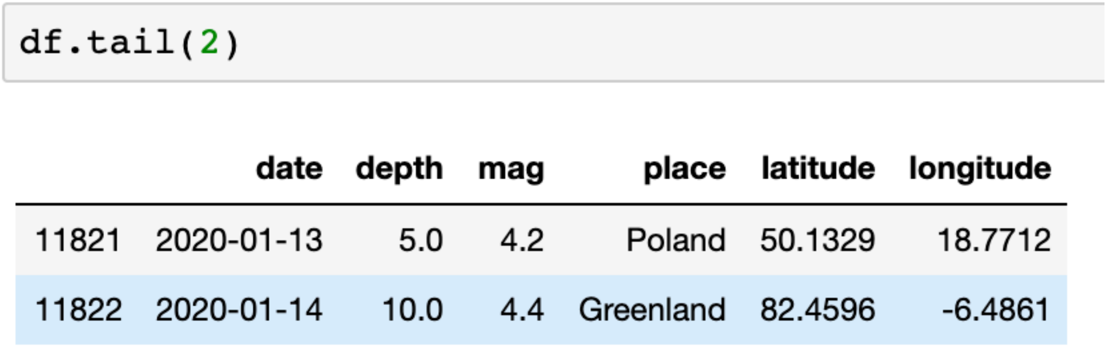
```
total locations: 100
```

After the feature engineering the data and outcome looks like as follow figures:



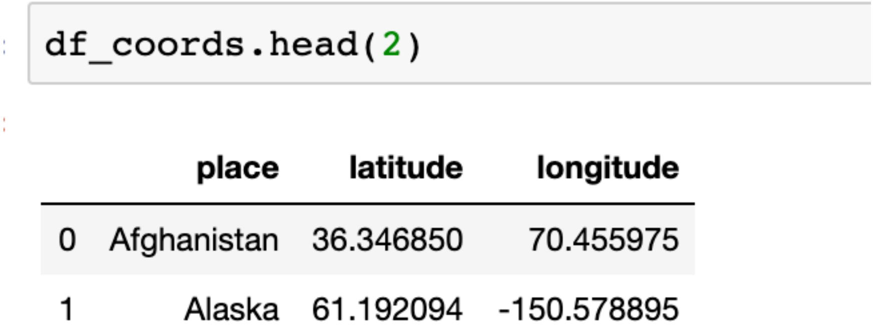
```
df.head(2)
```

	date	depth	mag	place	latitude	longitude
0	2019-12-17	0.0	1.1	Alaska	61.192094	-150.578895
1	2019-12-17	5.4	2.0	Alaska	61.192094	-150.578895



```
df.tail(2)
```

	date	depth	mag	place	latitude	longitude
11821	2020-01-13	5.0	4.2	Poland	50.1329	18.7712
11822	2020-01-14	10.0	4.4	Greenland	82.4596	-6.4861

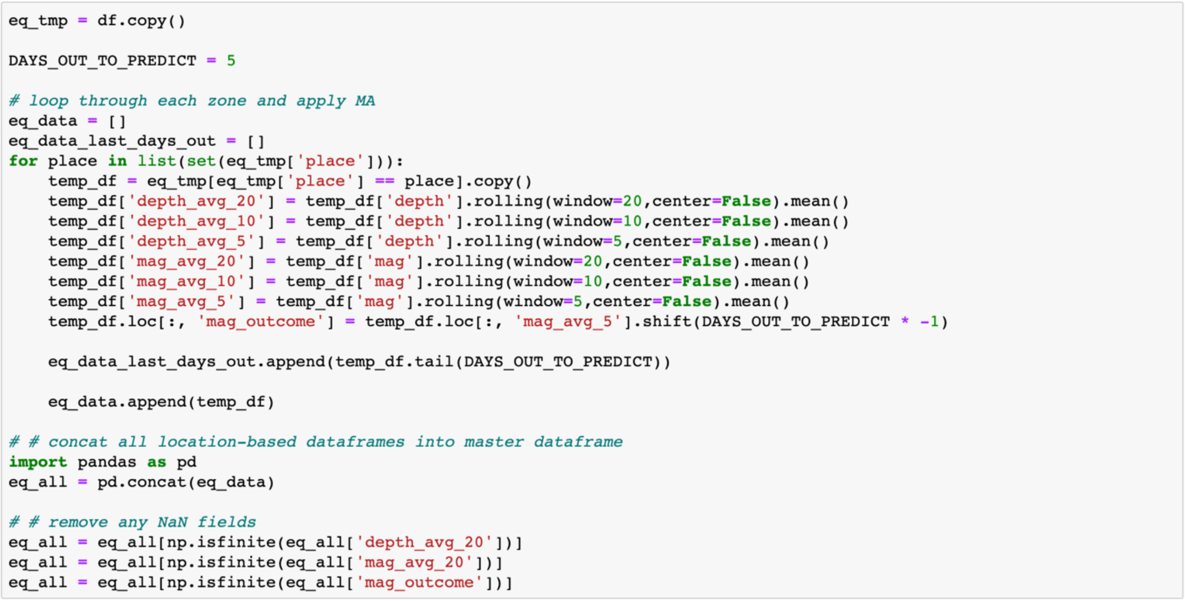


```
df_coords.head(2)
```

	place	latitude	longitude
0	Afghanistan	36.346850	70.455975
1	Alaska	61.192094	-150.578895

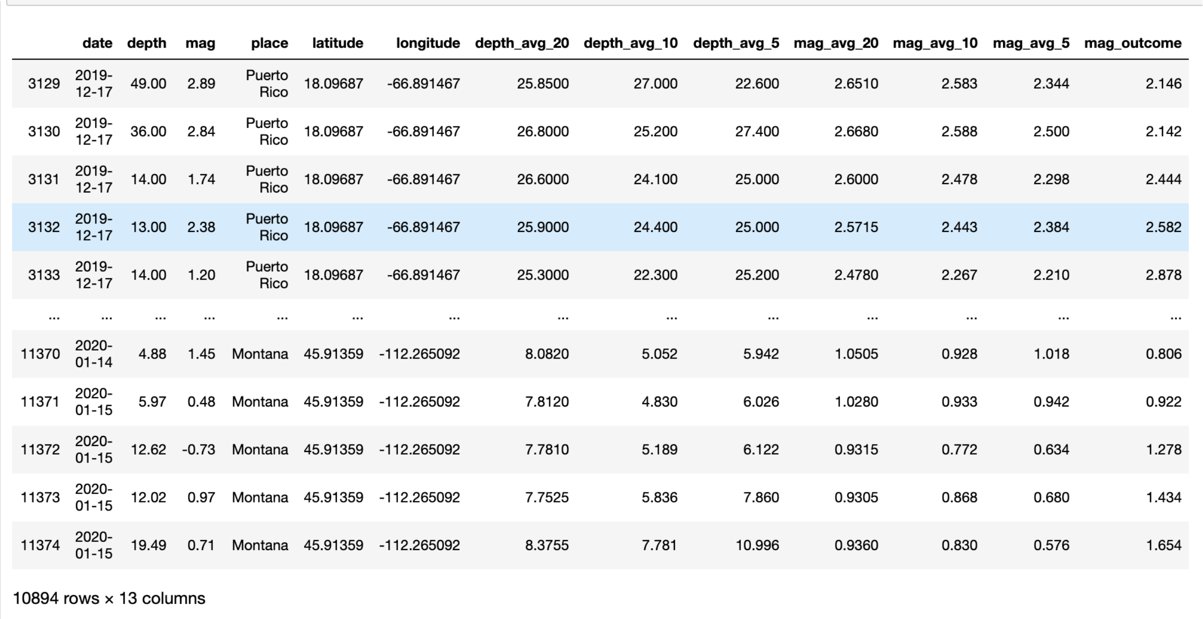
Using Moving Average (MA)

This section demonstrates the Moving Average (MA) model. Here, featuring the data using moving average of depth and magnitude. Depth and magnitude moving averages are 20, 10, and 5 respectively. The data will use from those values.



```
eq_tmp = df.copy()

DAYS_OUT_TO_PREDICT = 5

# loop through each zone and apply MA
eq_data = []
eq_data_last_days_out = []
for place in list(set(eq_tmp['place'])):
    temp_df = eq_tmp[eq_tmp['place'] == place].copy()
    temp_df['depth_avg_20'] = temp_df['depth'].rolling(window=20,center=False).mean()
    temp_df['depth_avg_10'] = temp_df['depth'].rolling(window=10,center=False).mean()
    temp_df['depth_avg_5'] = temp_df['depth'].rolling(window=5,center=False).mean()
    temp_df['mag_avg_20'] = temp_df['mag'].rolling(window=20,center=False).mean()
    temp_df['mag_avg_10'] = temp_df['mag'].rolling(window=10,center=False).mean()
    temp_df['mag_avg_5'] = temp_df['mag'].rolling(window=5,center=False).mean()
    temp_df.loc[:, 'mag_outcome'] = temp_df.loc[:, 'mag_avg_5'].shift(DAYS_OUT_TO_PREDICT * -1)

    eq_data_last_days_out.append(temp_df.tail(DAYS_OUT_TO_PREDICT))

    eq_data.append(temp_df)

# # concat all location-based dataframes into master dataframe
import pandas as pd
eq_all = pd.concat(eq_data)

# # remove any NaN fields
eq_all = eq_all[np.isfinite(eq_all['depth_avg_20'])]
eq_all = eq_all[np.isfinite(eq_all['mag_avg_20'])]
eq_all = eq_all[np.isfinite(eq_all['mag_outcome'])]
```

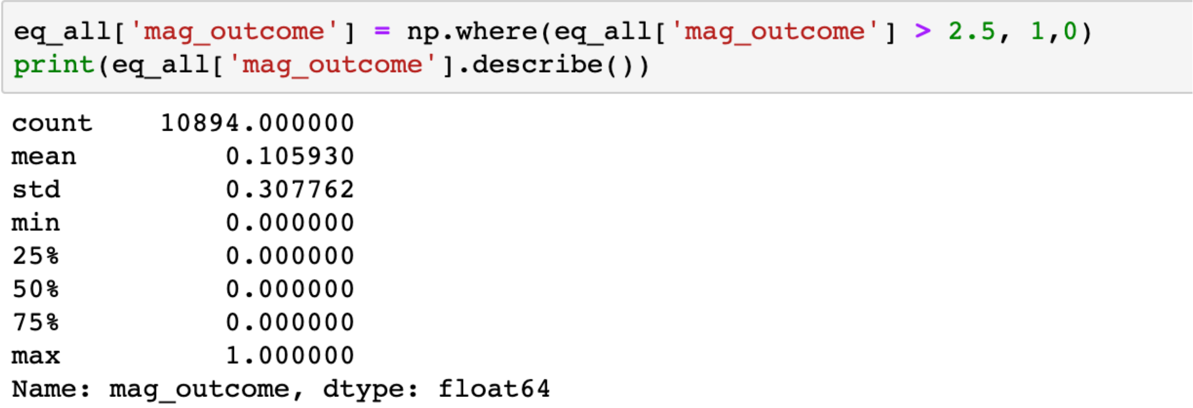
After calculating of moving average of depth and magnitude the data frame extended with all calculated average values.



	date	depth	mag	place	latitude	longitude	depth_avg_20	depth_avg_10	depth_avg_5	mag_avg_20	mag_avg_10	mag_avg_5	mag_outcome
3129	2019-12-17	49.00	2.89	Puerto Rico	18.09687	-66.891467	25.8500	27.000	22.600	2.6510	2.583	2.344	2.146
3130	2019-12-17	36.00	2.84	Puerto Rico	18.09687	-66.891467	26.8000	25.200	27.400	2.6680	2.588	2.500	2.142
3131	2019-12-17	14.00	1.74	Puerto Rico	18.09687	-66.891467	26.6000	24.100	25.000	2.6000	2.478	2.298	2.444
3132	2019-12-17	13.00	2.38	Puerto Rico	18.09687	-66.891467	25.9000	24.400	25.000	2.5715	2.443	2.384	2.582
3133	2019-12-17	14.00	1.20	Puerto Rico	18.09687	-66.891467	25.3000	22.300	25.200	2.4780	2.267	2.210	2.878
...	...	...	...	...	...	...	...	...	...	...	...	...	...
11370	2020-01-14	4.88	1.45	Montana	45.91359	-112.265092	8.0820	5.052	5.942	1.0505	0.928	1.018	0.806
11371	2020-01-15	5.97	0.48	Montana	45.91359	-112.265092	7.8120	4.830	6.026	1.0280	0.933	0.942	0.922
11372	2020-01-15	12.62	-0.73	Montana	45.91359	-112.265092	7.7810	5.189	6.122	0.9315	0.772	0.634	1.278
11373	2020-01-15	12.02	0.97	Montana	45.91359	-112.265092	7.7525	5.836	7.860	0.9305	0.868	0.680	1.434
11374	2020-01-15	19.49	0.71	Montana	45.91359	-112.265092	8.3755	7.781	10.996	0.9360	0.830	0.576	1.654

10894 rows × 13 columns

Condition for the magnitude output that if the mag is more than 2.5 it's true otherwise false. And get the statistical description like mean, standard deviation etc.

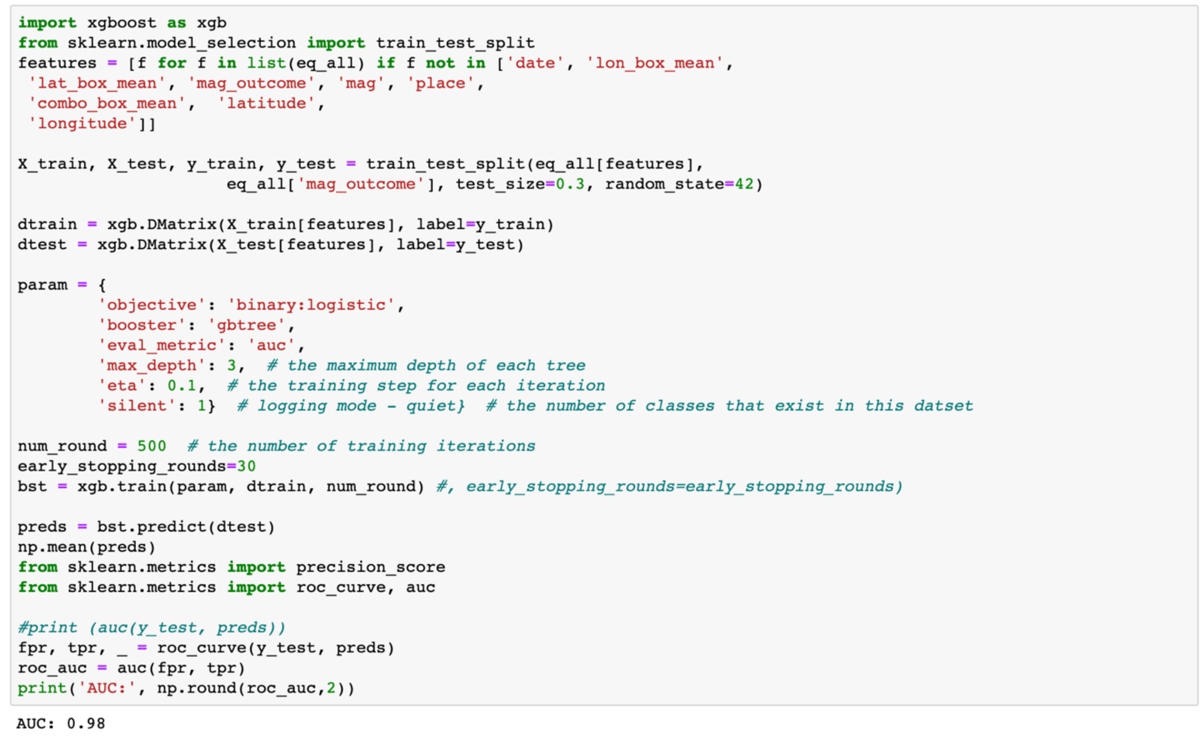


```
eq_all['mag_outcome'] = np.where(eq_all['mag_outcome'] > 2.5, 1,0)
print(eq_all['mag_outcome'].describe())
```

```
count    10894.000000
mean         0.105930
std          0.307762
min          0.000000
25%          0.000000
50%          0.000000
75%          0.000000
max          1.000000
Name: mag_outcome, dtype: float64
```

Using XGBoost Module:

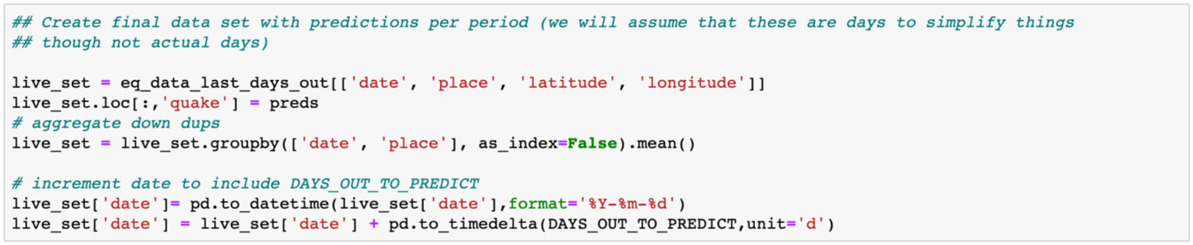
Dataset obtained for the earthquake is divided into training and testing sets using train_test_split using sklearn model_selection. For training and validation purposes, 70% of the dataset is selected, while testing is performed on rest of 30% hold out dataset. Training and test set value measured using XGBoost model. The model used the classification model. The test set is used for prediction and calculate the accuracy. Here using AUC for accuracy. The accuracy can be seen 98 percentage. That's mean model predicted very good.



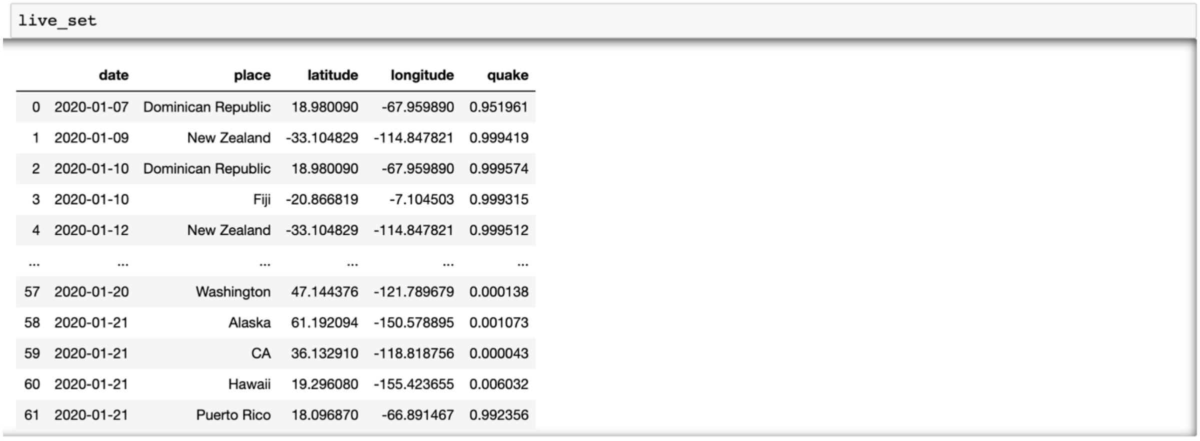
```
import xgboost as xgb
from sklearn.model_selection import train_test_split
features = [f for f in list(eq_all) if f not in ['date', 'lon_box_mean',
 'lat_box_mean', 'mag_outcome', 'mag', 'place',
 'combo_box_mean',  'latitude',
 'longitude']]

X_train, X_test, y_train, y_test = train_test_split(eq_all[features],
                     eq_all['mag_outcome'], test_size=0.3, random_state=42)

dtrain = xgb.DMatrix(X_train[features], label=y_train)
dtest = xgb.DMatrix(X_test[features], label=y_test)

param = {
        'objective': 'binary:logistic',
        'booster': 'gbtree',
        'eval_metric': 'auc',
        'max_depth': 3,  # the maximum depth of each tree
        'eta': 0.1,  # the training step for each iteration
        'silent': 1}  # logging mode - quiet}  # the number of classes that exist in this datset

num_round = 500  # the number of training iterations
early_stopping_rounds=30
bst = xgb.train(param, dtrain, num_round) #, early_stopping_rounds=early_stopping_rounds)

preds = bst.predict(dtest)
np.mean(preds)
from sklearn.metrics import precision_score
from sklearn.metrics import roc_curve, auc

#print (auc(y_test, preds))
fpr, tpr, _ = roc_curve(y_test, preds)
roc_auc = auc(fpr, tpr)
print('AUC:', np.round(roc_auc,2))
```

```
AUC: 0.98
```

Now the final data set is maintained where added quake column. Quake column has the predicted values. Also added the future dataset that means added 5 more rows which are the future predicted values of earthquake.



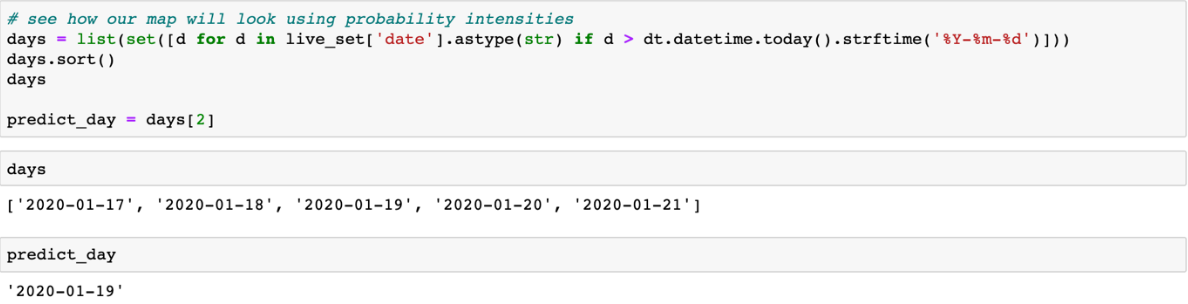
```
## Create final data set with predictions per period (we will assume that these are days to simplify things
## though not actual days)

live_set = eq_data_last_days_out[['date', 'place', 'latitude', 'longitude']]
live_set.loc[:,'quake'] = preds
# aggregate down dups
live_set = live_set.groupby(['date', 'place'], as_index=False).mean()

# increment date to include DAYS_OUT_TO_PREDICT
live_set['date']= pd.to_datetime(live_set['date'],format='%Y-%m-%d')
live_set['date'] = live_set['date'] + pd.to_timedelta(DAYS_OUT_TO_PREDICT,unit='d')
```



```
live_set
```

	date	place	latitude	longitude	quake
0	2020-01-07	Dominican Republic	18.980090	-67.959890	0.951961
1	2020-01-09	New Zealand	-33.104829	-114.847821	0.999419
2	2020-01-10	Dominican Republic	18.980090	-67.959890	0.999574
3	2020-01-10	Fiji	-20.866819	-7.104503	0.999315
4	2020-01-12	New Zealand	-33.104829	-114.847821	0.999512
...	...	...	...	...	...
57	2020-01-20	Washington	47.144376	-121.789679	0.000138
58	2020-01-21	Alaska	61.192094	-150.578895	0.001073
59	2020-01-21	CA	36.132910	-118.818756	0.000043
60	2020-01-21	Hawaii	19.296080	-155.423655	0.006032
61	2020-01-21	Puerto Rico	18.096870	-66.891467	0.992356

Let's check the list of future predicted date.



```
# see how our map will look using probability intensities
days = list(set([d for d in live_set['date'].astype(str) if d > dt.datetime.today().strftime('%Y-%m-%d')]))
days.sort()
days

predict_day = days[2]
```

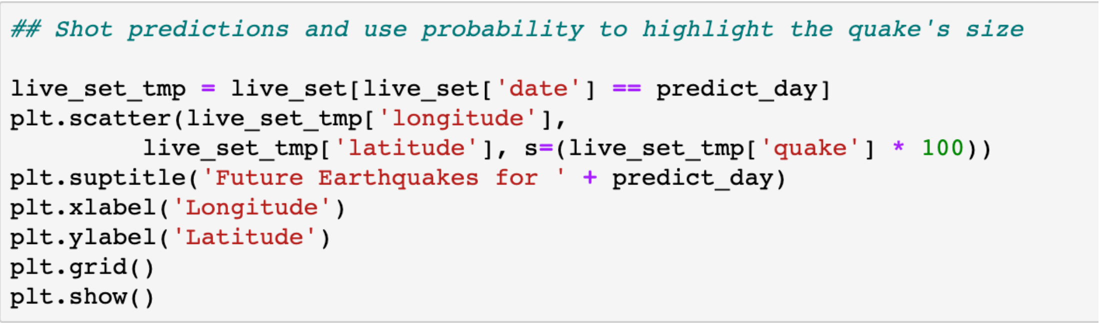
```
days
```

```
['2020-01-17', '2020-01-18', '2020-01-19', '2020-01-20', '2020-01-21']
```

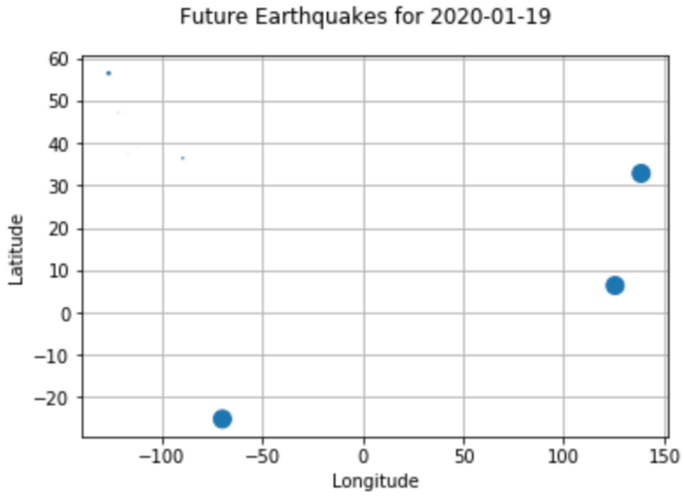
```
predict_day
```

```
'2020-01-19'
```

The future predicted value from predicted day is plotted using scatter plot.

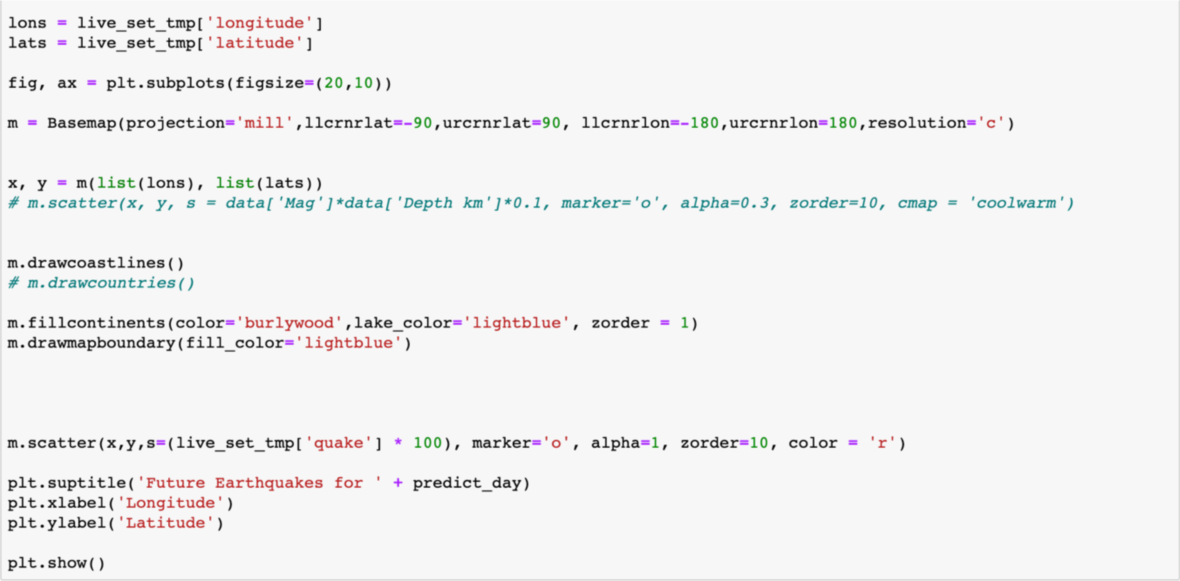


```
## Shot predictions and use probability to highlight the quake's size

live_set_tmp = live_set[live_set['date'] == predict_day]
plt.scatter(live_set_tmp['longitude'],
          live_set_tmp['latitude'], s=(live_set_tmp['quake'] * 100))
plt.suptitle('Future Earthquakes for ' + predict_day)
plt.xlabel('Longitude')
plt.ylabel('Latitude')
plt.grid()
plt.show()
```



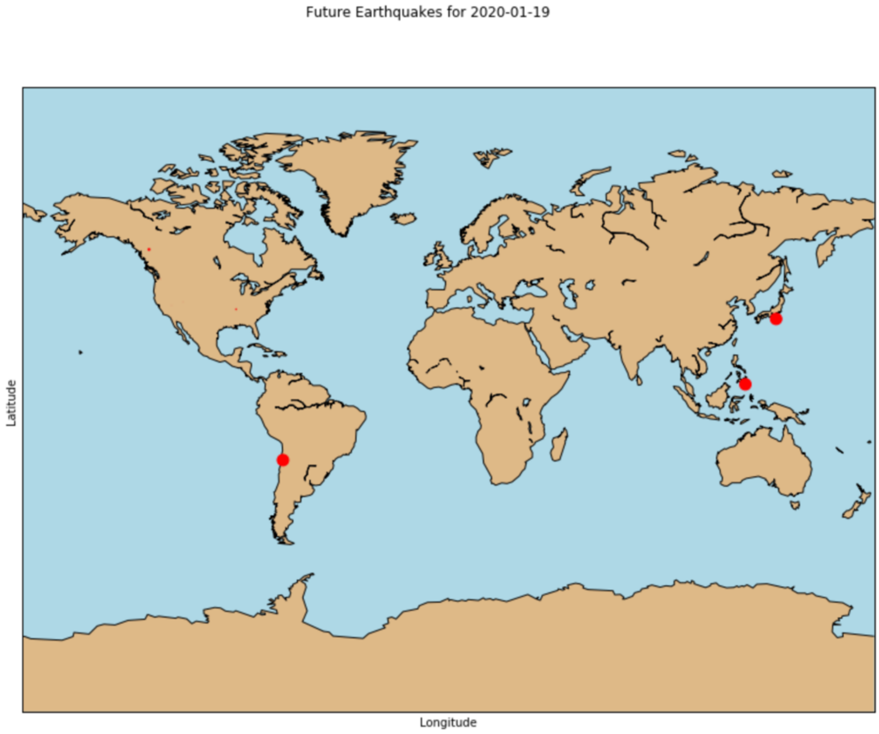
Final outcome:

All the code is summarized from the last final data which is using the Basemap for the actual map which can see the actual location from the predicted date of 2020-01-19.



```
lons = live_set_tmp['longitude']
lats = live_set_tmp['latitude']

fig, ax = plt.subplots(figsize=(20,10))

m = Basemap(projection='mill',llcrnrlat=-90,urcrnrlat=90, llcrnrlon=-180,urcrnrlon=180,resolution='c')


x, y = m(list(lons), list(lats))
# m.scatter(x, y, s = data['Mag']*data['Depth km']*0.1, marker='o', alpha=0.3, zorder=10, cmap = 'coolwarm')


m.drawcoastlines()
# m.drawcountries()

m.fillcontinents(color='burlywood',lake_color='lightblue', zorder = 1)
m.drawmapboundary(fill_color='lightblue')




m.scatter(x,y,s=(live_set_tmp['quake'] * 100), marker='o', alpha=1, zorder=10, color = 'r')

plt.suptitle('Future Earthquakes for ' + predict_day)
plt.xlabel('Longitude')
plt.ylabel('Latitude')

plt.show()
```

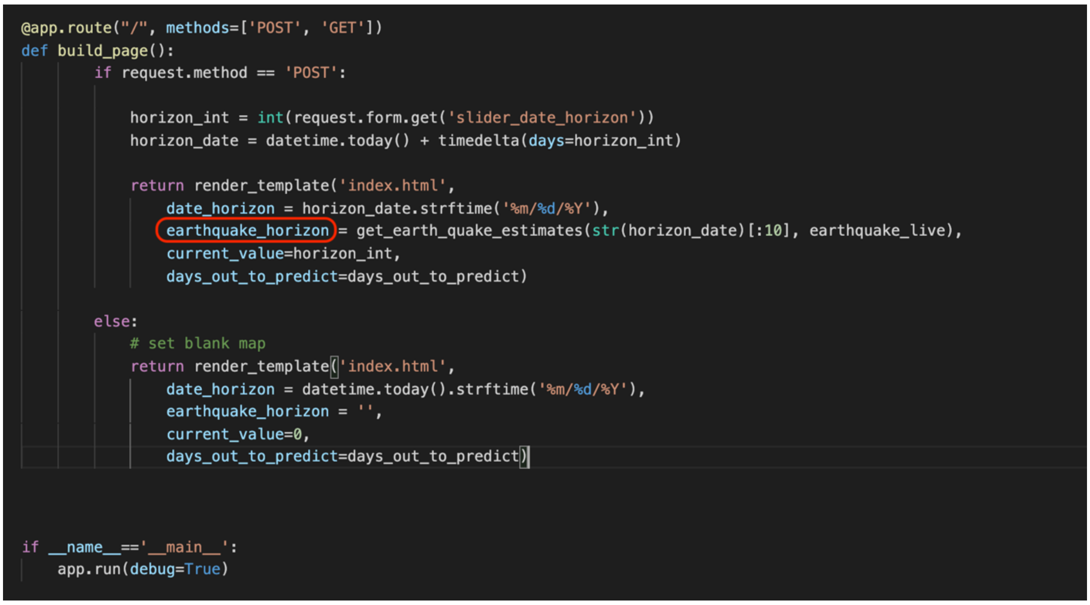
Future Earthquakes for 2020-01-19



Web Application

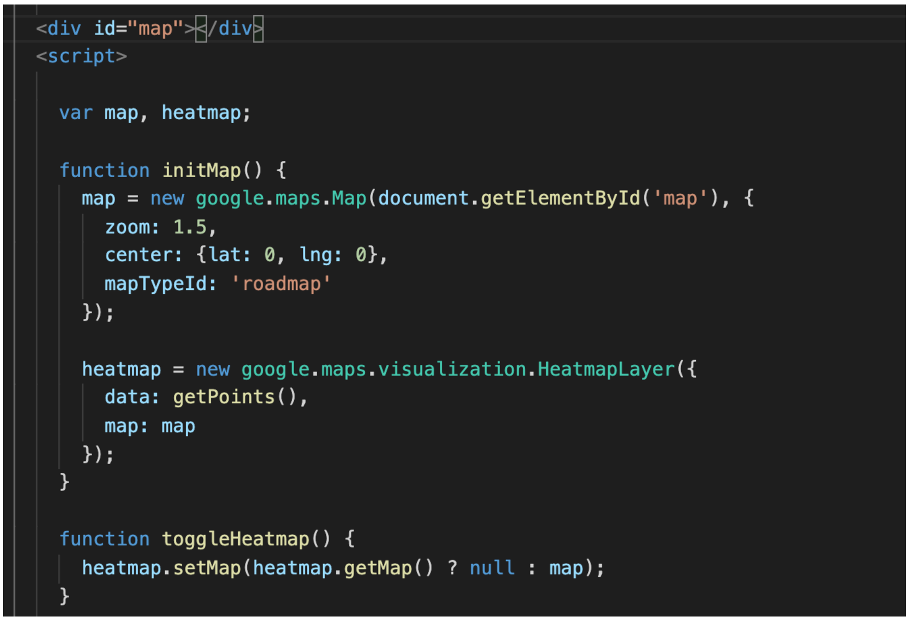
This is the base for user interaction. This is the application based on web any user can see the earthquake or find the location that is going to happen in next 5 days. This is based on python code using the Flask and basic HTML, CSS and google map api with JavaScript. User can use the slider from next day to other four days.

Here is the python code when the post will happen and load the data. After that return the value and load the html file. Main the value has been loaded in "earthquake_horizon" variable and called in html file.



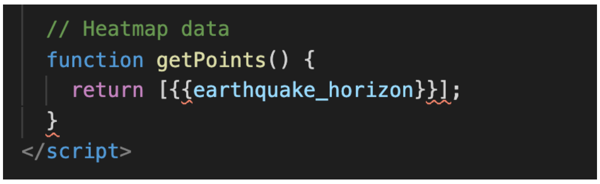
```python
@app.route("/", methods=['POST', 'GET'])
def build_page():
        if request.method == 'POST':

            horizon_int = int(request.form.get('slider_date_horizon'))
            horizon_date = datetime.today() + timedelta(days=horizon_int)

            return render_template('index.html',
                date_horizon = horizon_date.strftime('%m/%d/%Y'),
                earthquake_horizon = get_earth_quake_estimates(str(horizon_date)[:10], earthquake_live),
                current_value=horizon_int,
                days_out_to_predict=days_out_to_predict)

        else:
            # set blank map
            return render_template('index.html',
                date_horizon = datetime.today().strftime('%m/%d/%Y'),
                earthquake_horizon = '',
                current_value=0,
                days_out_to_predict=days_out_to_predict)



if __name__=='__main__':
    app.run(debug=True)
```

Here is the code that load the google map through id and the points are display as heatmap.



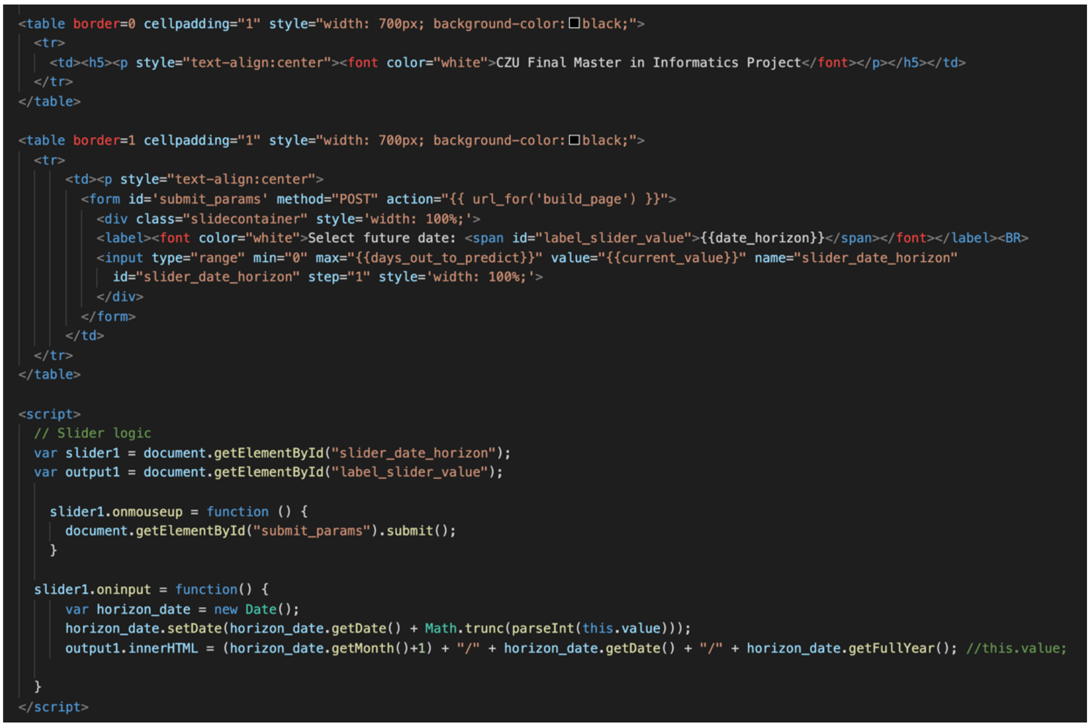
```html
<div id="map"></div>
<script>

  var map, heatmap;

  function initMap() {
    map = new google.maps.Map(document.getElementById('map'), {
      zoom: 1.5,
      center: {lat: 0, lng: 0},
      mapTypeId: 'roadmap'
    });

    heatmap = new google.maps.visualization.HeatmapLayer({
      data: getPoints(),
      map: map
    });
  }

  function toggleHeatmap() {
    heatmap.setMap(heatmap.getMap() ? null : map);
  }
```

This is the JavaScript function which load the data from python.



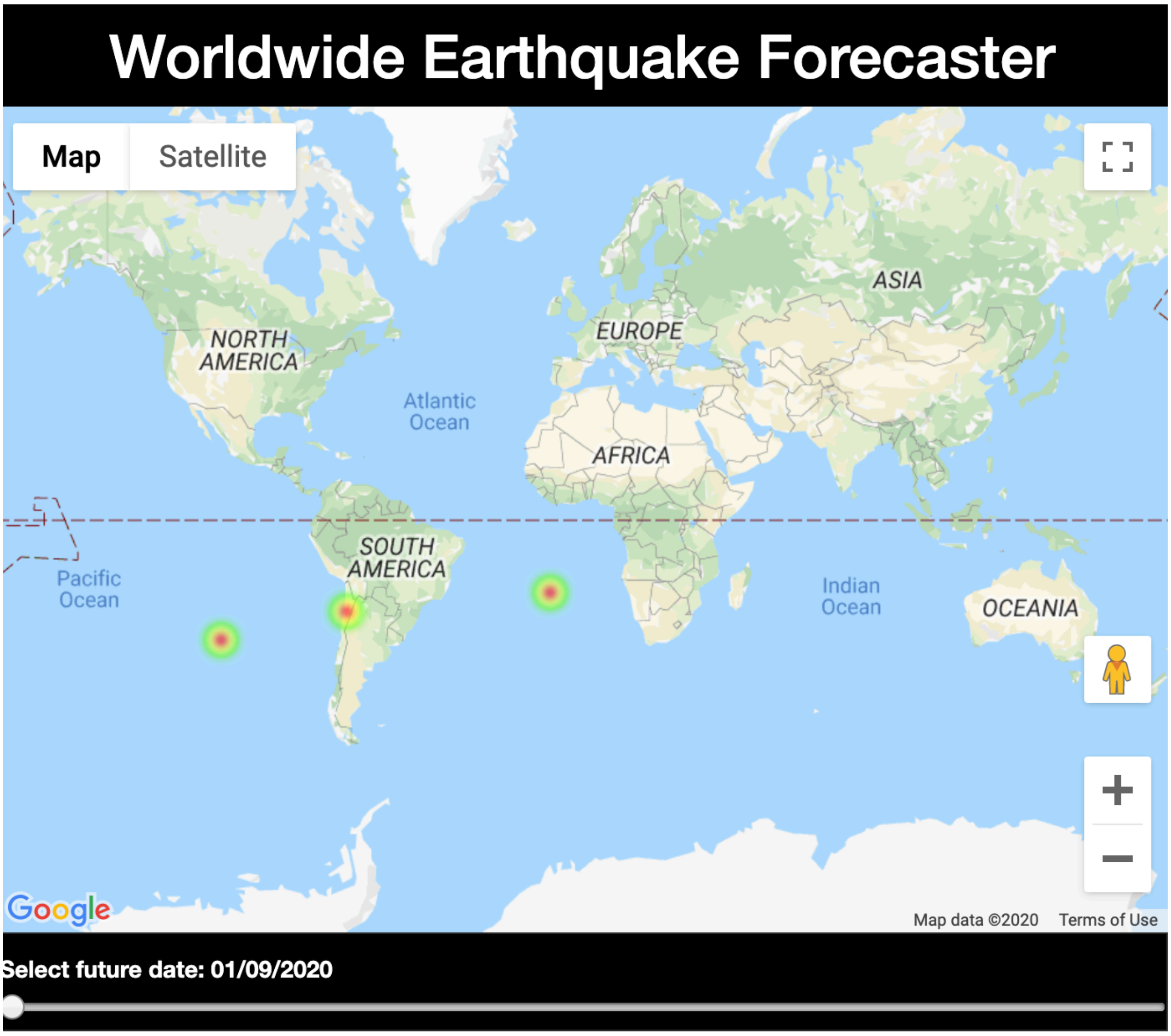
```
  // Heatmap data
  function getPoints() {
    return [{{earthquake_horizon}}];
  }
</script>
```

Now the all the html and JavaScript functions and code of slider and submit the post action when user slides the slider and the action will happen.



```
<table border=0 cellpadding="1" style="width: 700px; background-color:□black;">
  <tr>
    <td><h5><p style="text-align:center"><font color="white">CZU Final Master in Informatics Project</font></p></h5></td>
  </tr>
</table>

<table border=1 cellpadding="1" style="width: 700px; background-color:□black;">
  <tr>
      <td><p style="text-align:center">
        <form id='submit_params' method="POST" action="{{ url_for('build_page') }}">
          <div class="slidecontainer" style='width: 100%;'>
          <label><font color="white">Select future date: <span id="label_slider_value">{{date_horizon}}</span></font></label><BR>
          <input type="range" min="0" max="{{days_out_to_predict}}" value="{{current_value}}" name="slider_date_horizon"
            id="slider_date_horizon" step="1" style='width: 100%;'>
          </div>
        </form>
      </td>
  </tr>
</table>

<script>
  // Slider logic
  var slider1 = document.getElementById("slider_date_horizon");
  var output1 = document.getElementById("label_slider_value");

    slider1.onmouseup = function () {
      document.getElementById("submit_params").submit();
    }

  slider1.oninput = function() {
      var horizon_date = new Date();
      horizon_date.setDate(horizon_date.getDate() + Math.trunc(parseInt(this.value)));
      output1.innerHTML = (horizon_date.getMonth()+1) + "/" + horizon_date.getDate() + "/" + horizon_date.getFullYear(); //this.value;

  }
</script>
```

This is the main application where the date has changed when the slider run. Heatmap for points can been seen in the different places. Those heatmap indicate the earthquake in the future. Here is the figure the earthquake date is 2020-01-19 and can been seen the heatmap as earthquake.



Worldwide Earthquake Forecaster

Select future date: 01/09/2020

Conclusion

Forecasting is one of the hardest jobs. Lots of things impact to forecast. Furthermore, the correct data and filtering for right choices and correct the data. With changing the time, improvising and experiencing and also new technologies makes prediction possible. Computing skills and new technologies also added to help for prediction.

There are lots of different models have been developed. Every model has their own pros and cons. Selecting their model based on different things like univariant and multivariant time series data. Factors and variables also effect to the forecasting for example the endogenous and exogenous variables. Those variables could be impact to each other or not depends on the data type like univariant and multivariant. The lag variables at the same time and the previous variable on the prime time all are the factors that impact on the forecasting methods.

Best practice is that to see the trend that what the data looks like and where and how it is going towards to. So, here it comes the data analysis and data visualization. Since, earthquake is the hardest to predict. It won't affect by seasonal and it is also not frequently occurring things. It is the natural calamities. Its data is irregular.

But that doesn't mean it is not possible to predict or forecast. Since, there's already build tons of practical models based on machine learning. So, no human effort or manual process that makes failure for predictions. But it also doesn't mean it is perfectly hundred percentage accurate predictions. The short period of time, small magnitude and aftereffects can be predicted well and most cases those are predicted and got the good results. With combining seismologic with computer technology, the outcome is far better than previous time.

Even using neural networks like LTMS, the outcome is far superiors. But the classical methods and modules are also far beyond when using those in proper way. There are many classical methods are available to support the forecast and they are all good in their own way like AR, MA, ARMA, ARIMA, SES, SARIMA and so on.

Not only using those models but it is also possible that any user can use it through the application. It is possible that creating application for the general users without any analysis or using those models. They can use from anywhere any time from their mobile or their own any devices using internet access. There are many applications are available now these days and every time it is improving, and computational capacities are also becoming stronger. So, this is the thesis for all of those with demonstrate on practical part.

In this this, the data is the real time data from USCG. That data updated every day. Here it is used for a month data, so data processing won't take much more time in local environment. The method is used of classical approach. That method is moving average (MA). Generally, it is easy to see the trend of data flow from this model and possible to predict for short time periods since in this thesis has only a month data, so the prediction time period is also shortened. In this thesis prediction is only for next five days. So lagged variables and shifting procedure will be there for prediction.

Finally, predicting methods and new technologies will have far more superior in next future periods with new improving modules like LSTM using neuron networks. More accurate and more reliable output will come in future. But still these pre-build models and technologies are still providing the far better result and make possibilities even the hardest to predict like earthquake. Predicting earthquake and new update application can be found from the trusted application also. Those are the applicable and, in this paper, has the demo, sample and the process that can be found.

References

Chatfield, C. (2000). *Time-Series Forecasting.* London: Chapman & Hall/CRC.

Kenton, W. (2019, July 16). *What is the time series?* Retrieved from What is the time series?: https://www.investopedia.com/terms/t/timeseries.asp#understanding-time-series

Shiva, C. (2018, December 06). *Time Series Forecasting.* . Retrieved from Time Series Forecasting.: https://dzone.com/articles/time-series-forecasting

Burba, D. (2019, October 30). *An overview of time series forecasting models.* Retrieved from An overview of time series forecasting models.: https://towardsdatascience.com/an-overview-of-time-series-forecasting-models-a2fa7a358fcb

Gerlow, M. E. (1993). Economic evaluation of commodity price forecasting models. . *International Journal of Forecasting,*, 387-397.

Montgomery, D. J. (2015). *Introduction to Time Series Analysis and Forecasting.*

Hyndman, R. &. (2018). *Forecasting: Principles and Practice.* Australia: Monash University.

Brownlee, J. (2019, September 18). *Autoregression Models for Time Series Forecasting With Python.* . Retrieved from Autoregression Models for Time Series Forecasting With Python. : https://machinelearningmastery.com/autoregression-models-time-series-forecasting-python/

Sagar, R. (2019, January 04). *9 Essential Time-Series Forecasting Methods in Python.* . Retrieved from 9 Essential Time-Series Forecasting Methods in Python.: https://analyticsindiamag.com/time-series-forecasting-methods-in-python/

Glen, S. (2013, September 24). *Moving Average: What it is and How to Calculate it.* Retrieved from Moving Average: What it is and How to Calculate it.: https://www.statisticshowto.datasciencecentral.com/moving-average/

Wikiversity. (2018, November 27). *Wikiversity.* . Retrieved from Moving Average.: https://en.wikiversity.org/wiki/Moving_Average

Mills, T. (2019). A Practical Guide to Modeling and Forecasting. In *Applied Time Series Analysis.* (pp. 31-56). Loughborough: Loughborough University.

science, P. E. (n.d.). *PennState Eberly college of science*. Retrieved from PennState Eberly college of science: https://online.stat.psu.edu/stat501/lesson/14/14.1

Chen, J. (2019, April 13). *Technical Analysis: Autoregressive Integrated Moving Average (ARIMA).* Retrieved from Technical Analysis: Autoregressive Integrated Moving Average (ARIMA).: https://www.investopedia.com/terms/a/autoregressive-integrated-moving-average-arima.asp

Foo, K. (2018, January 4). *Seasonal lags: SARIMA modelling and forecasting*. Retrieved from Seasonal lags: SARIMA modelling and forecasting: https://medium.com/@kfoofw/seasonal-lags-sarima-model-fa671a858729

Brownlee, J. (2019, August 21). *A Gentle Introduction to SARIMA for Time Series Forecasting in Python*. Retrieved from A Gentle Introduction to SARIMA for Time Series Forecasting in Python: https://machinelearningmastery.com/sarima-for-time-series-forecasting-in-python/

Brownlee, J. (2019, August 21). *11 Classical Time Series Forecasting Methods in Python (Cheat Sheet)*. Retrieved from 11 Classical Time Series Forecasting Methods in Python (Cheat Sheet): https://machinelearningmastery.com/time-series-forecasting-methods-in-python-cheat-sheet/

faculty.washington. (2020, February). *Vector Autoregressive Models for Multivariate Time Series*. Retrieved from Vector Autoregressive Models for Multivariate Time Series: https://faculty.washington.edu/ezivot/econ584/notes/varModels.pdf

Arunraj, N. (2016, April). *Application of SARIMAX Model to Forecast Daily Sales in Food Retail Industry*. Retrieved from Application of SARIMAX Model to Forecast Daily Sales in Food Retail Industry: https://www.researchgate.net/publication/296477650_Application_of_SARIMAX_Model_to_Forecast_Daily_Sales_in_Food_Retail_Industry

Alfonso, C. (2017, August, 10). *An Introduction to Vector Autoregression*. Retrieved from An Introduction to Vector Autoregression: https://dzone.com/articles/vector-autoregression-overview-and-proposals

Prabhakaran.S. (2019, July 7). *Vector Autoregression (VAR) – Comprehensive Guide with Examples in Python*. Retrieved from Vector Autoregression (VAR) – Comprehensive Guide with Examples in Python: https://www.machinelearningplus.com/time-series/vector-autoregression-examples-python/

Mathworks. (2020, February). *Vector Autoregression (VAR) Models*. Retrieved from Vector Autoregression (VAR) Models: https://www.mathworks.com/help/econ/introduction-to-vector-autoregressive-var-models.html#bsxcdon-1

Scherrer, M. W. (2019). Handbook of Statistics. In S. &. M., *Chapter 6 - Vector autoregressive moving average models* (pp. 145-191).

Poskitt, D. S. (2011, March 22). *Econometrics & Business Statistics Version*. Retrieved from Vector Autoregresive Moving Average Identification for Macroeconomic Modeling: A New Methodology: https://www.eui.eu/Documents/DepartmentsCentres/Economics

Wilmsa. I, B. S. (2019, April 25). *Sparse Identification and Estimation of Large-Scale Vector AutoRegressive Moving Averages*. Retrieved from Sparse Identification and Estimation of Large-Scale Vector AutoRegressive Moving Averages: https://arxiv.org/pdf/1707.09208.pdf

support.sas. (2020, February). *The VARMAX Procedure. RESOURCES / FOCUS AREAS.* Retrieved from The VARMAX Procedure. RESOURCES / FOCUS AREAS.: https://support.sas.com/rnd/app/ets/procedures/ets_varmax.html

Ostertagova. E & Ostertag. O. (2011, September). *The Simple Exponential Smoothing Model*. Retrieved from The Simple Exponential Smoothing Model: https://www.researchgate.net/publication/256088917_The_Simple_Exponential_Smoothing_Model

Boehmke. B. (2020, February). *Exponential Smoothing*. Retrieved from Exponential Smoothing: http://uc-r.github.io/ts_exp_smoothing

Smith. M & Agrawal. R., S. (2020, February). *A Comparison of Time Series Model Forecasting Methods on Patent Groups*. Retrieved from A Comparison of Time Series Model Forecasting Methods on Patent Groups: http://ceur-ws.org/Vol-1353/paper_13.pdf

Krishni. (2019, April 06). *A High-Level Introduction to LSTMs*. Retrieved from A High-Level Introduction to LSTMs: https://medium.com/datadriveninvestor/a-high-level-introduction-to-lstms-34f81bfa262d

Dalkey. N. (1969). The Delphi method. *An experimental study of group opinion*, 408-426.