NAANMUDHALVAN-IBM SKILL

ARTIFICIAL INTELLIGENCE

GROUP PROJECT

Project Title: Earthquake Prediction model using python.

Phase III . Submission

S.No	Group Members Name	**Naan Mudhalvan** ID	Email ID
1	S Abirami	au820321106001	veeravelabirami@gmail.com
2	M.Kalaimathi	au820321106019	muhilvanangu@gmail.com
3	S.Amirtha	au820321106006	amirthasaravanakumar04@gmail.com
4	S.Aishwarya	au820321106003	aishwaryasivakumar12112003@gmail.com
5	P.Bavadharani	au820321106010	dharanibava89@gmail.com
6.	B.Gnanasri	au820321106014	gnanasribaskar23@gmail.com

Abstract

Earthquakes are natural disasters that can cause significant damage and loss of life. Accurate prediction of earthquakes is essential for developing early warning systems, disaster planning, risk assessment, and scientific research. This project aims to predict the magnitude and probability of Earthquake occurring in a particular region (California, United States) from the historic data of that region using various Machine learning models.

Dataset

The dataset used in this project is called the "SOCR Earthquake Dataset", and it contains information about earthquakes that have occurred with a magnitude of 3.0 or greater in California, United States.

Each row in the dataset represents a single earthquake event and includes the following information:

- Date and time of the earthquake in UTC (Coordinated Universal Time)
- Latitude and longitude(in degree) of the epicentre, which is the point on the Earth's surface directly above where the earthquake occurred
- Depth of the earthquake, measured in kilometers
- Magnitude of the earthquake on the Richter scale
- SRC = source
- nst - number of stations used for solution (range: 0 to ...)
- close - distance of closest station to epicenter (range: 0 to ...)
- rms - root-mean-squared residual of solution (range: 0. to 1.)
- gap - azimuthal gap (range: 0 to 360)

The Richter scale is a logarithmic scale that measures the magnitude of an earthquake based on the energy released by the earthquake. Each increase of one unit on the Richter scale represents a tenfold increase in the amplitude of the seismic waves generated by the earthquake.

The dataset contains earthquake events from January 2, 2017, to December 31, 2019, which includes a total of 37,706 earthquakes. This dataset could be used for a variety of purposes, such as studying earthquake patterns and trends over time or for predicting future earthquake activities.

Introduction

The SOCR Earthquake Dataset can be used to build machine learning models to predict earthquakes or to better understand earthquake patterns and characteristics. Here are a few possible ways machine learning models can be used with this dataset:

1. Earthquake prediction: You can use this dataset to build a model that predicts when and where an earthquake might occur based on past earthquake data. You could use techniques such as time series analysis, clustering, or classification to identify patterns in the data and make predictions.
2. Magnitude prediction: You can use this dataset to build a model that predicts the magnitude of an earthquake based on other factors such as location, depth, or the number of seismic stations that recorded the earthquake. You could use regression techniques to build this model.

3. Risk assessment: You can use this dataset to identify areas that are at higher risk of earthquakes based on historical earthquake data. You could use clustering or classification techniques to identify patterns in the data and identify areas with similar characteristics.
4. Anomaly detection: You can use this dataset to detect anomalies or outliers in the data, which could represent earthquakes that are unusual or unexpected. You could use techniques such as clustering or classification to identify patterns in the data and detect anomalies.
5. Data visualization: You can use this dataset to create visualizations of earthquake data, which could help you identify patterns and relationships in the data. You could use techniques such as scatter plots, heat maps, or geographic information systems (GIS) to visualize the data.

These are just a few examples of the many ways that machine learning models can be used with the SOCR Earthquake Dataset. The specific approach you take will depend on your research question and the goals of your analysis. In this project we focus mainly on Earthquake prediction and Magnitude prediction.

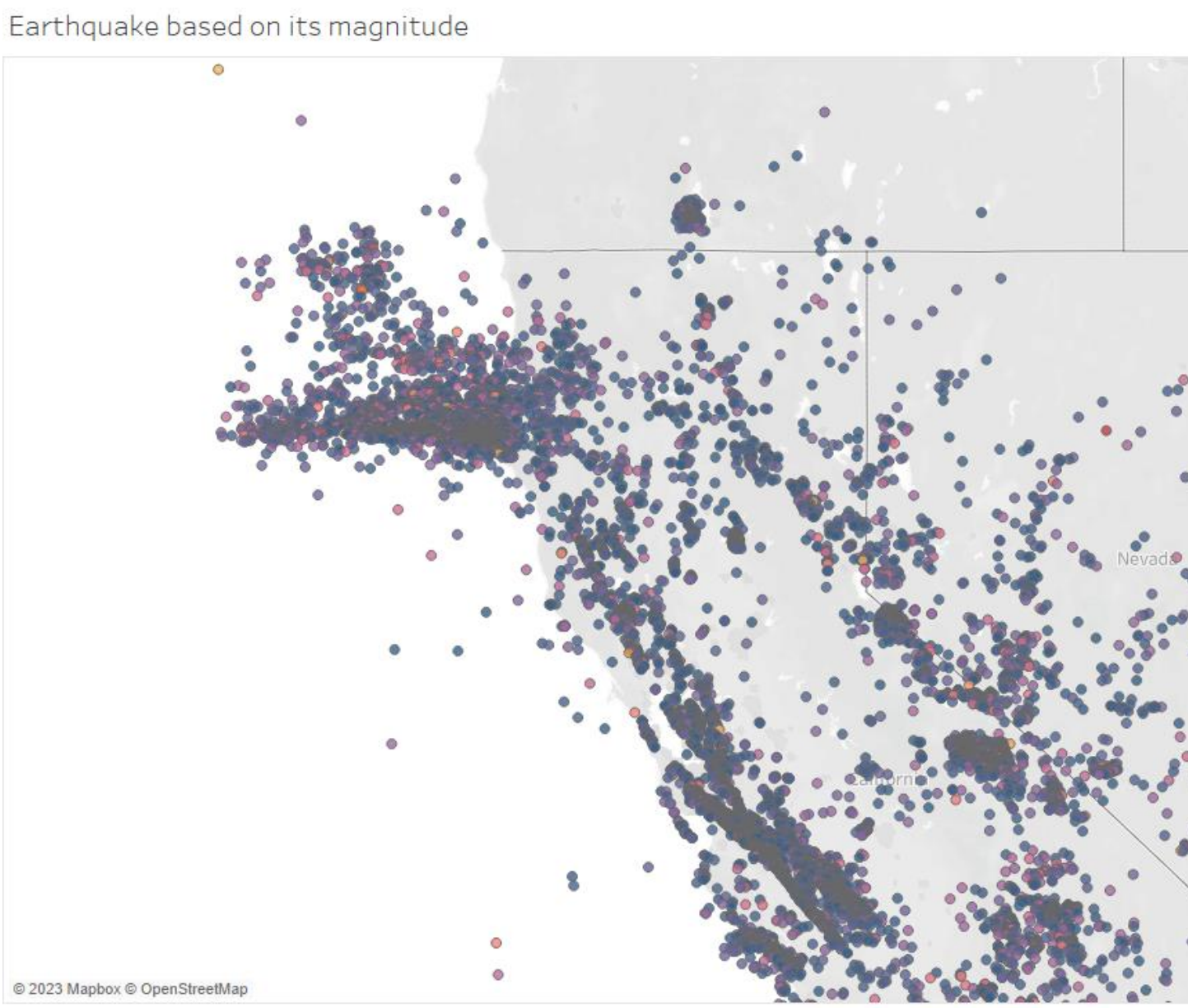


Figure 1

Earthquake based on its magnitude

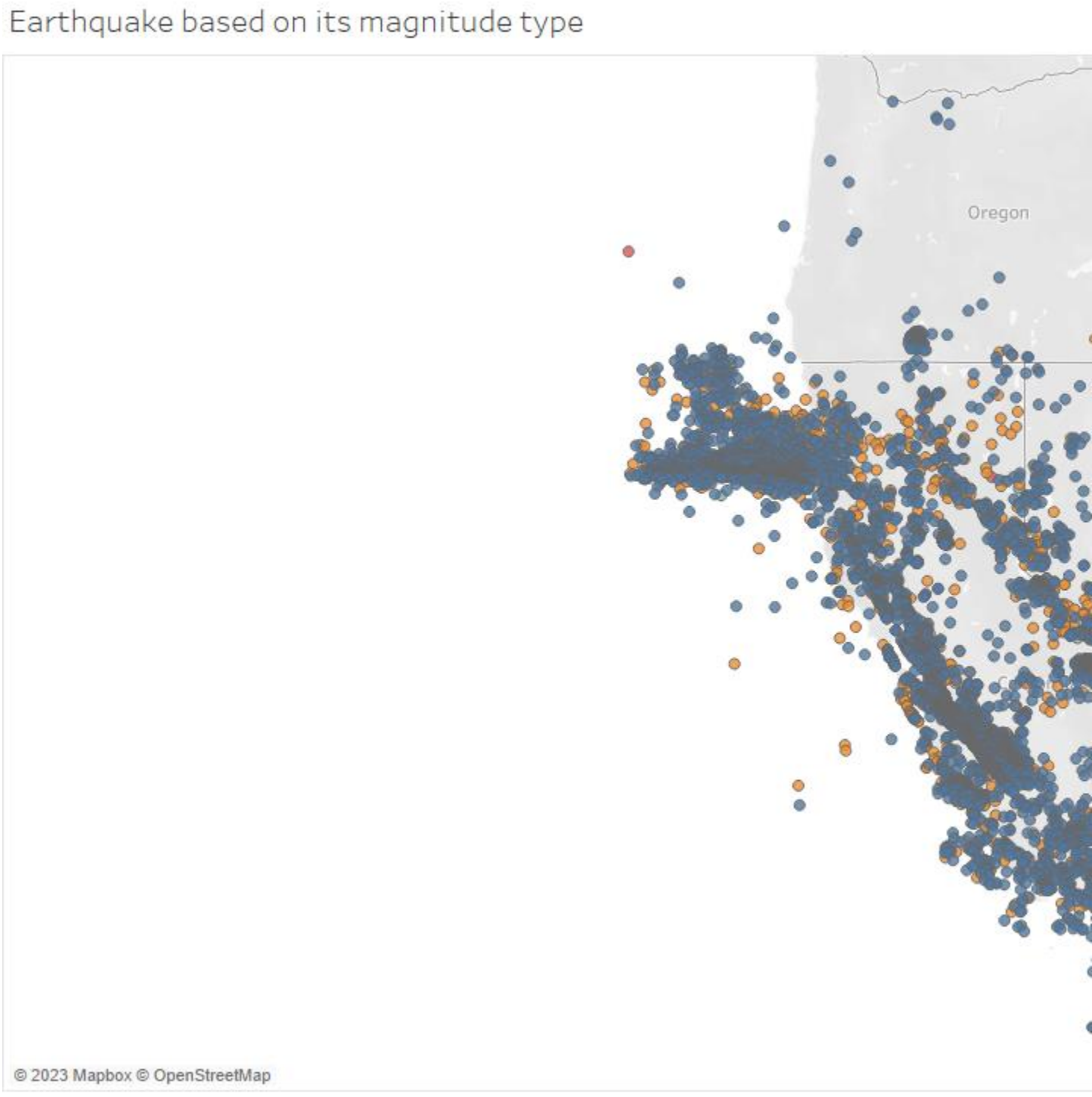


Figure 2

EEarthquake based on its magnitude type

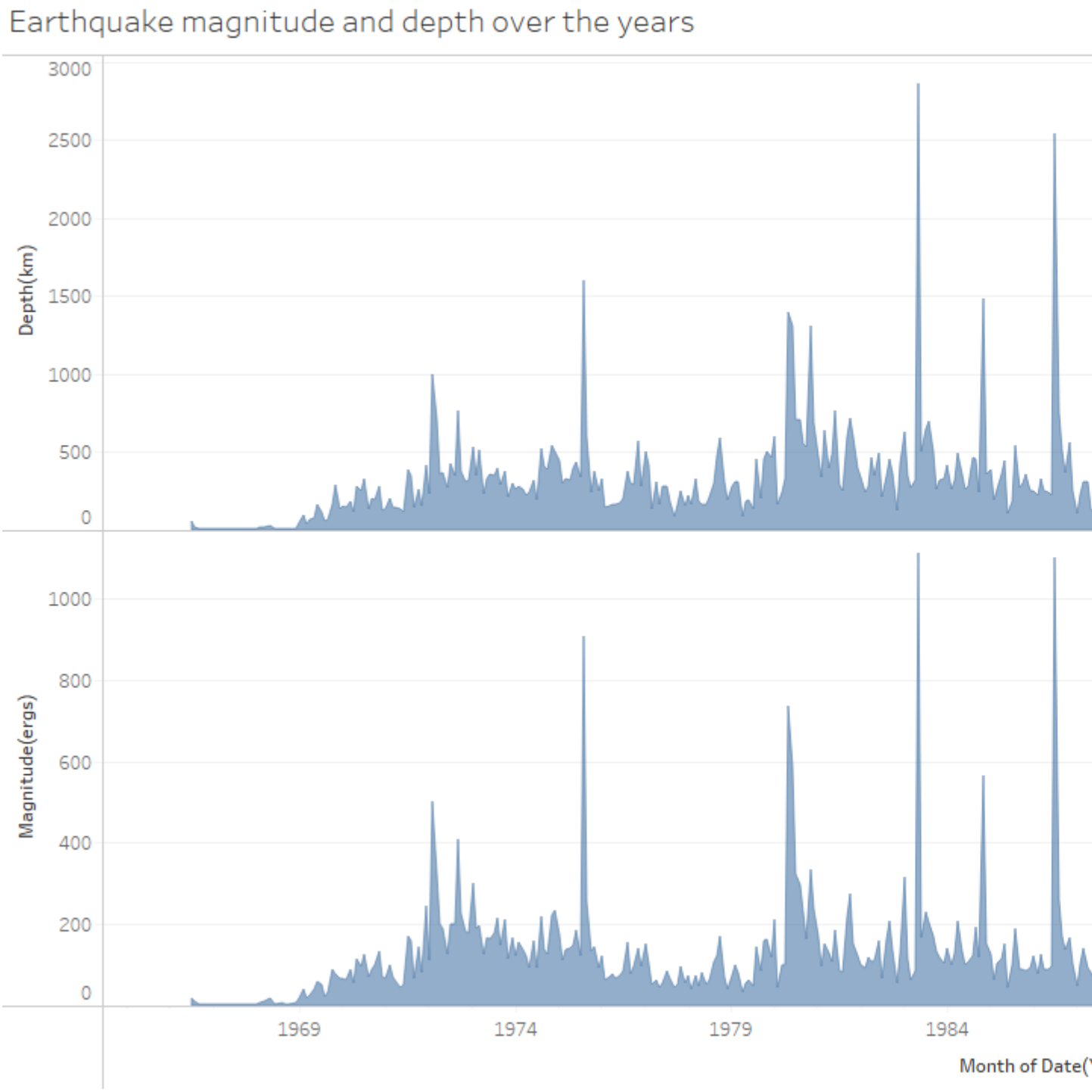


Figure 3

Earthquake magnitude and depth over the years

Implementation

We will use four models in this project:

1. Linear regression
2. Support Vector Machine(SVM)
3. Naive Bayes
4. Random Forest

Linear Regression

Linear regression is a type of supervised machine learning algorithm that is used to model the linear relationship between a dependent variable (in this case, earthquake magnitude) and one or more independent variables (in this case, latitude, longitude, depth, and the number of seismic stations that recorded the earthquake).

The basic idea behind linear regression is to find the line of best fit through the data that minimizes the sum of the squared residuals (the difference between the predicted and actual values of the dependent variable). The coefficients of the line of best fit are estimated using a method called ordinary least squares, which involves minimizing the sum of the squared residuals with respect to the coefficients.

In this situation, we have used multiple linear regression to model the relationship between earthquake magnitude and latitude, longitude, depth, and the number of seismic stations that recorded the earthquake. The multiple linear regression model assumes that there is a linear relationship between the dependent variable (magnitude) and each of the independent variables (latitude, longitude, depth, and number of seismic stations), and that the relationship is additive (i.e., the effect of each independent variable on the dependent variable is independent of the other independent variables).

Once the model has been fit to the data, we can use it to predict the magnitude of a new earthquake given its latitude, longitude, depth, and the number of seismic stations that recorded it. This can be useful for earthquake monitoring and early warning systems, as well as for understanding the underlying causes of earthquakes and improving our ability to predict them in the future.

Figure 4
Multiple linear regression plot using seaborn library(python)

The linear regression equation used in our multiple linear regression model for earthquake magnitude prediction with latitude, longitude, depth, and number of seismic stations as independent variables can be written as:

Magnitude = -0.6028 * Latitude + 1.2012 * Longitude - 0.0008 * Depth + 0.0239 * No_of_stations + 0.1573

Where:

- Magnitude is the dependent variable, representing the magnitude of the earthquake
- Latitude, Longitude, Depth, and No_of_stations are the independent variables
- The coefficients (-0.6028, 1.2012, -0.0008, and 0.0239) represent the slopes of the regression line for each independent variable
- The intercept (0.1573) represents the predicted magnitude when all independent variables are zero.
- This equation allows us to predict the magnitude of an earthquake based on its latitude, longitude, depth, and the number of seismic stations that recorded it. By plugging in the values of the independent variables for a given earthquake, we can obtain an estimate of its magnitude.

The results we obtained from the linear regression model were as follows:

- Mean squared error (MSE): 0.17562
- R-squared (R2) score: 0.03498

SVM

Support Vector Machines (SVM) is a type of supervised machine learning algorithm that can be used for both regression and classification tasks. The basic idea behind SVM is to find the best boundary that separates the data into different classes or predicts a continuous output variable (in this case, earthquake magnitude).

In SVM, the data points are mapped to a higher-dimensional space where the boundary can be easily determined. The best boundary is the one that maximizes the margin, which is the distance between the boundary and the closest data points from each class. This boundary is called the "hyperplane."

For regression tasks, SVM uses a similar approach but instead of a hyperplane, it finds a line (or curve in higher dimensions) that best fits the data while maximizing the margin. This line is the "support vector regression line."

SVM can handle both linear and non-linear data by using different kernels that transform the data into a higher-dimensional space. Some commonly used kernels include linear, polynomial, and radial basis function (RBF) kernels.

The predicted values from SVM model when evaluated using mse and r2 metrics:

- Mean squared error (MSE): 0.53166
- R-squared (R2) score: -1.92129

Naive Bayes

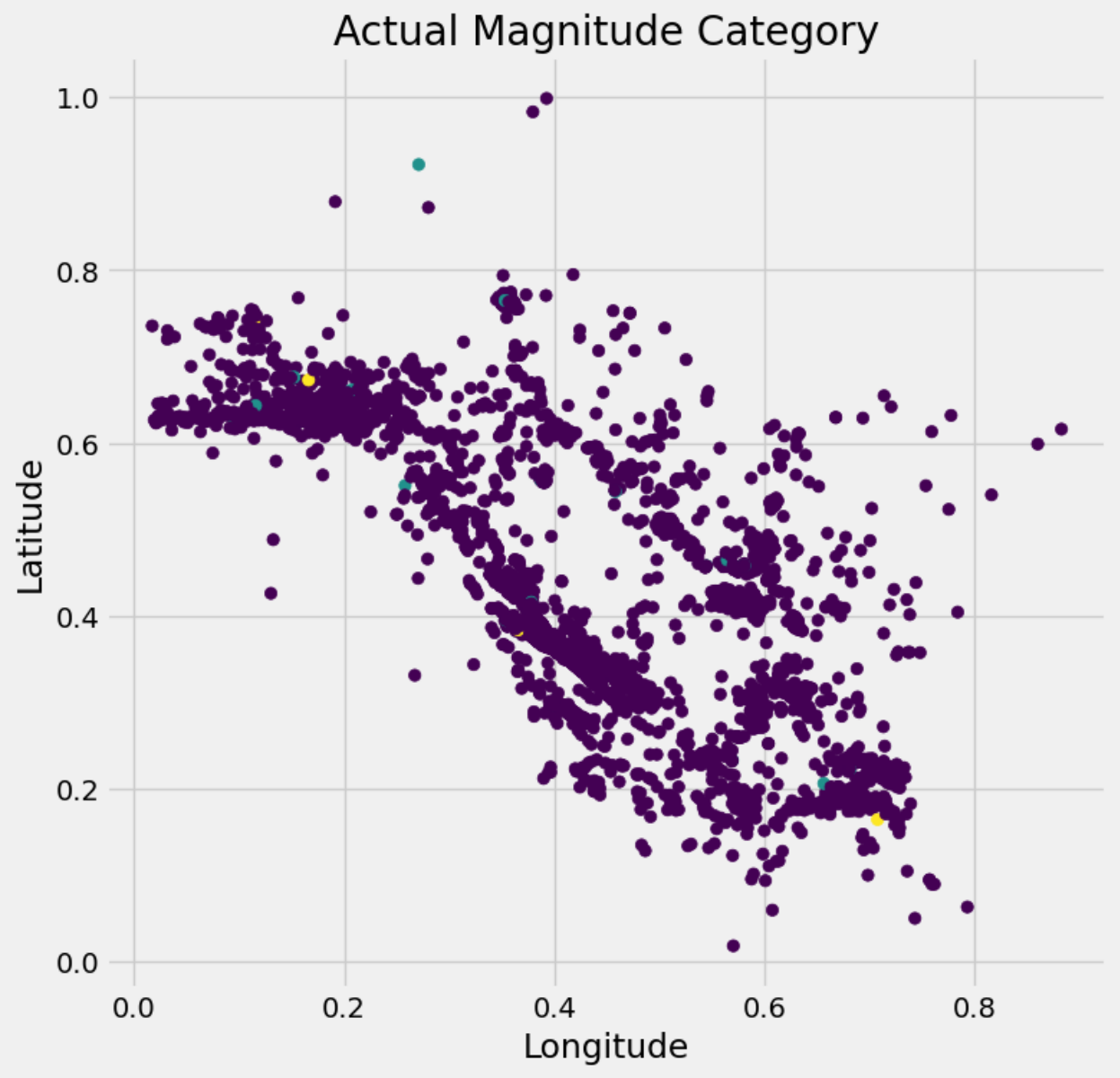
In statistics, naive Bayes classifiers are a family of simple "probabilistic classifiers" based on applying Bayes' theorem with strong (naive) independence assumptions between the features (see Bayes classifier). They are among the simplest Bayesian network models,[1] but coupled with kernel density estimation, they can achieve high accuracy levels.

Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression,[3]:718 which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers.

In the code, we used the Naive Bayes classifier to predict the magnitude of earthquakes based on their latitude, longitude and number of monitoring stations. We split the data into training and testing sets, trained the Naive Bayes model on the training data, and evaluated its performance on the test data using the accuracy score, confusion matrix and classification report

Figure 5
Actual vs Predicted

Figure 5
Actual vs Predicted



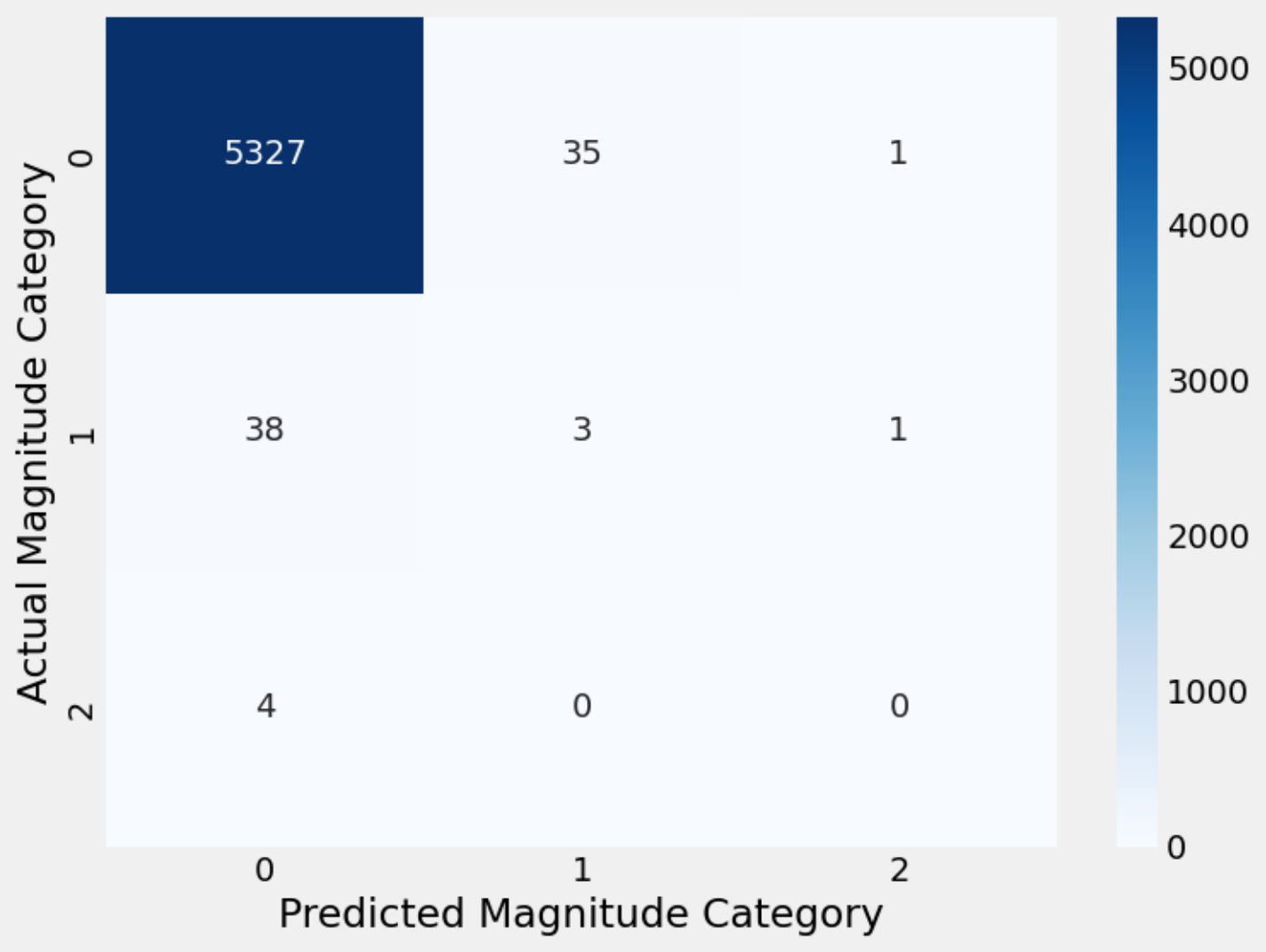


Figure 6
Heatmap of Confusion Matrix

Figure 6
Heatmap of Confusion Matrix

- Accuracy: 0.9853947125161767
- Confusion Matrix: [[5327 35 1] [38 3 1] [4 0 0]]

Random Forest

Random forest is a machine learning algorithm that is used for both classification and regression tasks. It is an ensemble learning method that combines multiple decision trees to create a more accurate and robust model.

The basic idea behind random forest is to create multiple decision trees, each trained on a subset of the data and a random subset of the features. Each tree makes a prediction, and the final prediction is the average (for regression) or the mode (for classification) of the individual tree predictions. By creating many trees and taking their average, random forest can reduce the impact of overfitting and improve the accuracy and stability of the model.

In the code we provided earlier, we used the random forest algorithm to predict the magnitude of earthquakes based on their latitude, longitude, depth, and number of monitoring stations. We split the data into training and testing sets, trained the random forest model on the training data, and evaluated its performance on the test data using the mean squared error (MSE) and R-squared (R2) score.

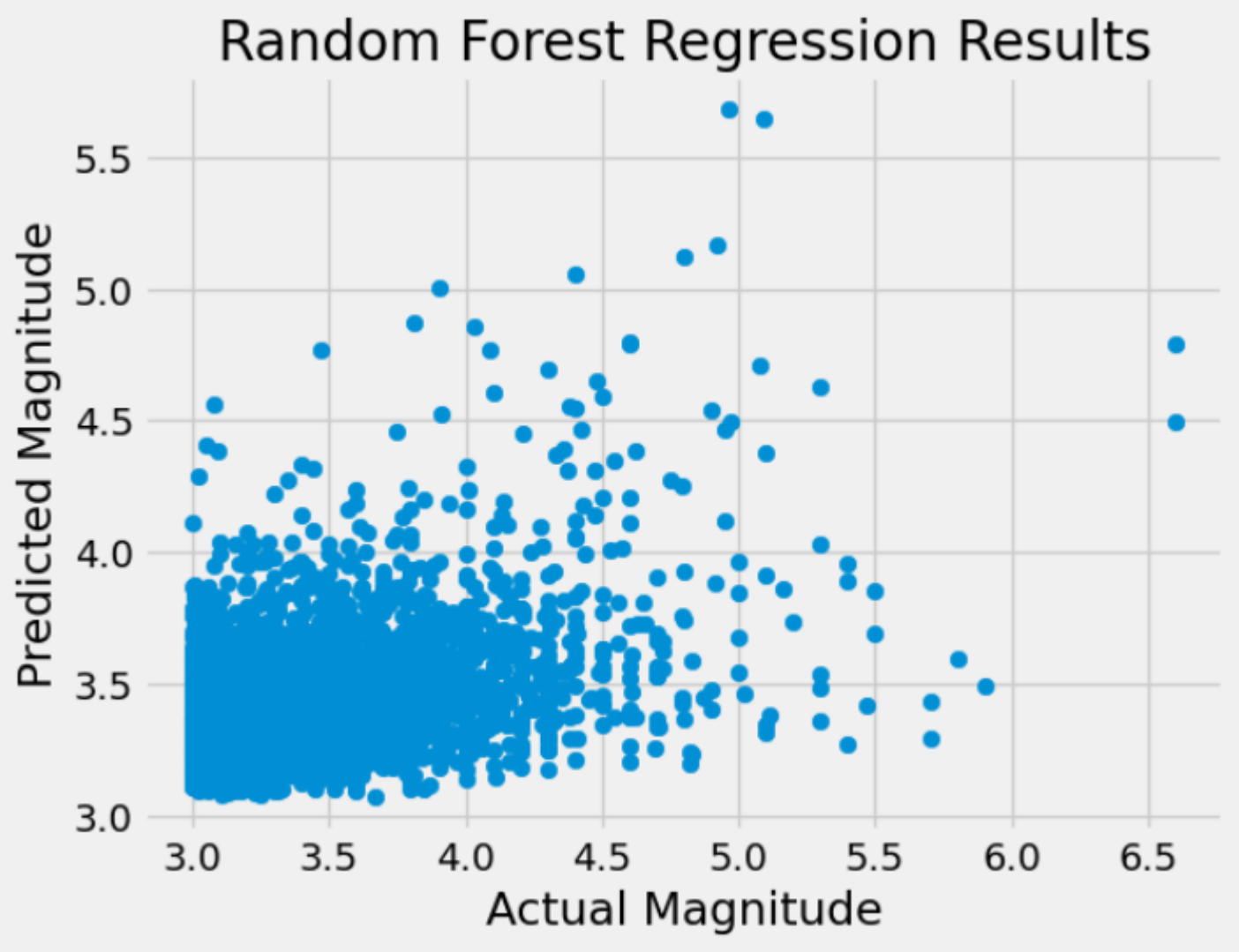


Figure 7
Actual vs Predicted

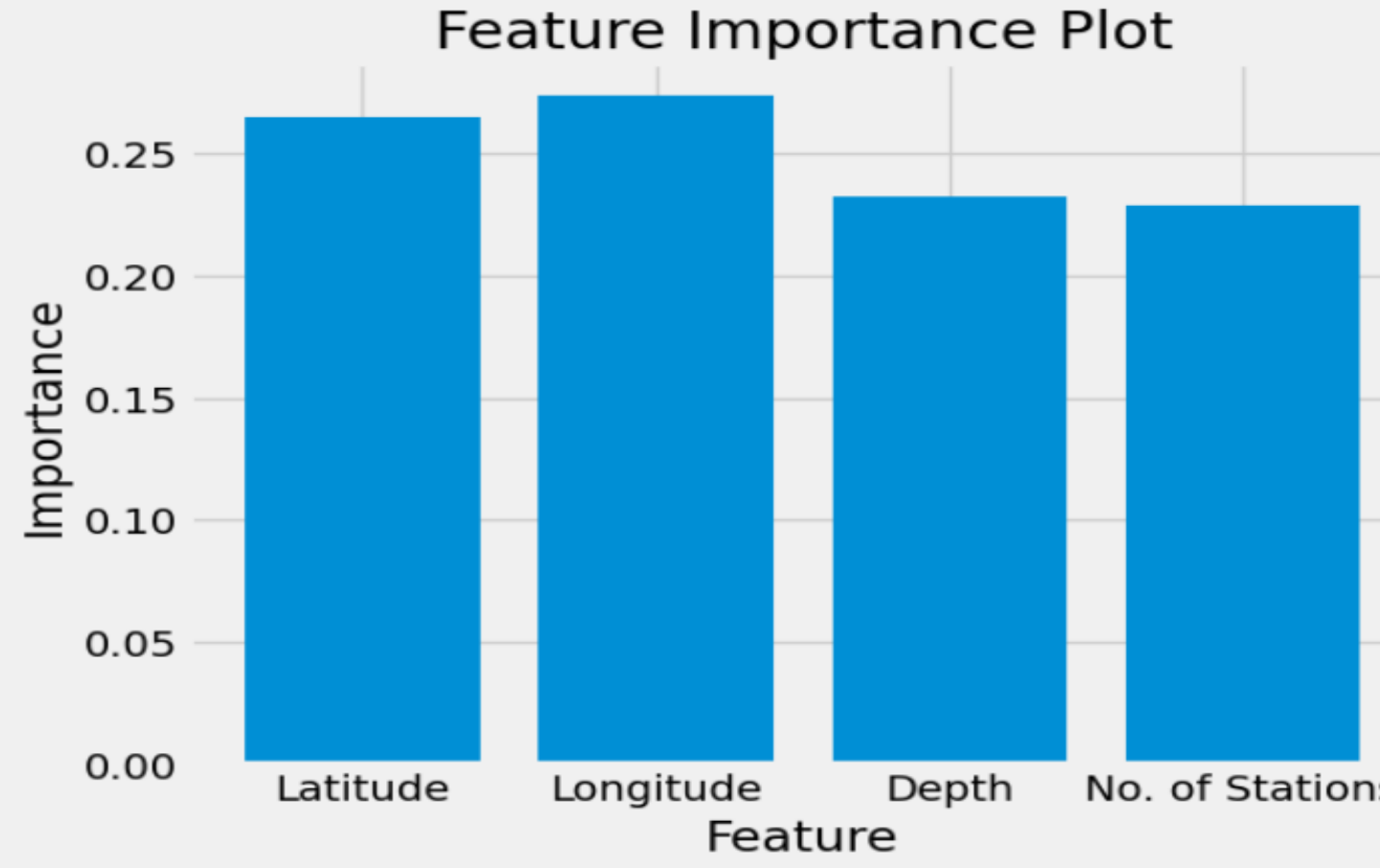


Figure 8
Feature Importance Plot

The results we obtained from the random forest model were as follows:

- Mean squared error (MSE): 0.15599
- R-squared (R2) score: 0.14288

These results indicate that the random forest model was able to accurately predict the magnitude of earthquakes based on the given features. The low MSE and high R2 score indicate that the model was making accurate predictions, and was able to explain a large proportion of the variance in the target variable.

Overall, the random forest algorithm is a powerful tool for machine learning tasks, and can be used in a variety of applications, including finance, healthcare, and image recognition

Conclusion

When comparing two models, both the mean squared error (MSE) and R-squared (R2) score can be used to evaluate the performance of the models.

In general, a model with a lower MSE and a higher R2 score is considered a better model. This is because the MSE measures the average difference between the predicted and actual values, and a lower MSE indicates that the model is making more accurate predictions. The R2 score measures the proportion of the variance in the target variable that is explained by the model, and a higher R2 score indicates that the model is able to explain more of the variability in the target variable.

From the results of this project we can conclude that random forest is the most accurate model for predicting the magnitude of Earthquake compared to all other models used in this project.

However, it's important to keep in mind that the relative importance of MSE and R2 score may vary depending on the specific problem and the context in which the models are being used. For example, in some cases, minimizing the MSE may be more important than maximizing the R2 score, or vice versa. It's also possible that one model may perform better on one metric and worse on another, so it's important to consider both metrics together when evaluating the performance of the models.