Rock Mine Prediction

A MINI PROJECT REPORT

Submitted by

Vinay Poddar [RA2011032010061]
Kartikey Mahawar[RA2011032010041]
Jonal Suthar[RA2011032010066]

Under the guidance of

Dr. Balasaraswathi V R
(Assistant Professor, Department of Networking and Communication)

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING
With specialization in Internet of Things

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956

SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR - 603203

MAY 2023

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
COLLEGE OF ENGINEERING & TECHNOLOGY
S.R.M. NAGAR, KATTANKULATHUR – 603 203

BONAFIDE CERTIFICATE

Certified that this project report **Rock Mine Prediction** is the bonafide work of **Vinay Poddar(RA2011032010061) , Kartikey Mahawar(RA2011032010041) and Jonal Suthar (RA2011032010066)** of III Year/VI Sem B. Tech(CSE) who carried out the miniproject work under my supervision for the course 18CSE392T – Machine Learning - I in SRMInstitute of Science and Technology during the academic year 2022-2023(Even sem).

Dept. of Networking and Communications Kattankulathur 603 203 * SRM IST *

IGNATURE

r. Balasaraswathi V R
SUPERVISOR
Assistant Professor
Department of Networking and Communication
SRMIST

SIGNATURE

Dr. Annapurani Panaiyappan
K.Head of Department
Professor
Department of Networking and Communications
SRMIST

ABSTRACT

Underwater Mine usage by the naval defense system provides great security but also possesses a threat to the marine life and submarine vessels as the mines can be easily mistaken for rocks. We need a much more accurate system to predict the object as it is very dangerous if a mistake is made. To have a great accuracy we need accurate data to generate accurate results. We worked on the data set which is provided by Gorman, R. P., and Sejnowski, T. J. (1988). The data is used to train the machine. This paper presents a method for the prediction of underwater mines and rocks using Sonar signals. Sonar signals are used to record the various frequencies of underwater objects at 60 different angles. We constructed three binary classifier models according to their accuracy. Then, prediction models are used to predict the mine and rock categories. Python and Supervised Machine Learning Classification algorithms are used to construct these prediction models.

TABLE OF CONTENTS

1.1 Introduction

Rock mine prediction refers to the process of using data analysis techniques to forecast the likelihood of finding valuable mineral deposits within a geological formation. This is a critical task in the mining industry, as it helps companies make informed decisions about where to invest resources for exploration and extraction.

The prediction of rock mines involves analyzing geological data such as rock types, mineral composition, and structural characteristics. This data is typically collected through various methods such as geological surveys, drilling, and geophysical measurements. Machine learning algorithms and statistical models are then applied to this data to create predictive models that can identify areas with high potential for mineral deposits.

Accurate rock mine prediction is essential for ensuring the sustainability of mining operations and minimizing environmental impact. It also plays a vital role in supporting economic development and creating job opportunities in mining communities.

1.2 Objectives

Identify areas with high potential for valuable mineral deposits: Accurately predicting the location and quantity of mineral deposits can help mining companies identify areas that are most likely to yield significant returns on investment.

Reduce exploration costs: Mining companies often invest substantial amounts of time and resources in exploring for mineral deposits. Predictive models can help reduce exploration costs by directing exploration efforts towards areas with the highest potential for success.

Minimize environmental impact: Accurate rock mine prediction can help minimize the environmental impact of mining activities by directing exploration and extraction efforts away from environmentally sensitive areas.

Optimize resource utilization: Predictive models can help mining companies optimize resource utilization by directing exploration and extraction efforts towards areas with the highest potential for mineral deposits.

Improve safety: Accurate rock mine prediction can help improve safety by directing mining activities towards areas that are less likely to pose a risk to workers or the environment.

Enhance decision-making: Predictive models can provide mining companies with valuable insights and information to make more informed decisions about exploration and extraction activities, leading to better overall outcomes.

1.3 General and Unique Services

General services of rock mine prediction include:

Accurate and reliable mineral deposit predictions: Predictive models can provide accurate and reliable predictions of mineral deposits, reducing the risks and costs associated with exploration and extraction.

Time and cost savings: By directing exploration and extraction efforts towards areas with the highest potential for mineral deposits, predictive models can save time and reduce exploration costs.

Improved decision-making: Predictive models can provide valuable insights and information to help mining companies make more informed decisions about exploration, extraction, and investment.

Environmental protection: Accurate rock mine prediction can help minimize the environmental impact of mining activities by directing extraction efforts towards areas that are less likely to cause harm.

Safety improvements: Rock mine prediction can help improve safety by directing mining activities towards areas that pose a lower risk of safety hazards.

Unique services of rock mine prediction include:

Customization: Predictive models can be customized to suit the specific needs and requirements of mining companies, taking into account geological formations, mineral deposits, and other factors.

Advanced technology: Rock mine prediction often involves advanced technology, such as machine learning and artificial intelligence, which can provide more accurate and reliable predictions than traditional methods.

Continuous monitoring: Predictive models can be used to continuously monitor mining activities and provide feedback, helping to optimize operations and reduce risks.

Early warning systems: Rock mine prediction can be used to develop early warning systems for potential safety hazards or environmental impact, allowing mining companies to take action before a problem occurs.

Multidisciplinary approach: Rock mine prediction often involves collaboration among experts in geology, data analysis, and mining operations, leading to a more comprehensive understanding of geological formations and mineral deposits.

1.4 Software Requirements Specification

The software requirements for rock mine prediction can vary depending on the specific techniques and methods used for data analysis. However, here are some common software requirements for rock mine prediction:

Data processing and management: Software tools for data processing, management, and analysis are essential for rock mine prediction. This includes tools for cleaning, filtering, and processing geological data, such as Python, R, and MATLAB.

Machine learning and artificial intelligence: Machine learning and artificial intelligence algorithms are commonly used in rock mine prediction. Software tools for implementing these algorithms include TensorFlow, Keras, PyTorch, and scikit-learn.

Geographic Information Systems (GIS): GIS software is used for spatial analysis and visualization of geological data. This includes software such as ArcGIS, QGIS, and MapInfo.

Data visualization and reporting: Software tools for data visualization and reporting are critical for communicating results and insights from rock mine prediction. This includes tools such as Tableau, Power BI, and ggplot2.

High-performance computing: High-performance computing infrastructure may be required for processing and analyzing large volumes of geological data. Software tools such as Apache Hadoop and Apache Spark can be used for distributed computing and processing.

Collaboration and project management: Collaboration and project management software tools such as GitHub, JIRA, and Trello can be used to manage data analysis projects and ensure efficient collaboration among team members.

2. Literature Survey

2.1 Existing System

There have been numerous studies and developments in the field of rock mine prediction, using a variety of methods and technologies. Here are some examples of existing systems for rock mine prediction:

Geostatistical modeling: Geostatistical modeling is a widely used method for rock mine prediction. It involves the use of statistical analysis and mathematical models to estimate the quantity and quality of mineral deposits in a geological formation. Geostatistical models are typically based on data gathered from geological surveys, drilling, and other sources, and can be used to create maps and models of mineral deposits.

Machine learning and artificial intelligence: Machine learning and artificial intelligence (AI) are increasingly being used for rock mine prediction. These methods involve the use of algorithms and statistical models to analyse large amounts of data and identify patterns and relationships between variables. Machine learning and AI can be used to develop predictive models that can accurately identify areas with high potential for mineral deposits.

Remote sensing: Remote sensing techniques, such as aerial photography, satellite imagery, and LiDAR (Light Detection and Ranging), can be used to gather data about geological formations and mineral deposits. This data can be used to create 3D models of the terrain and identify areas with high potential for mineral deposits.

Seismic surveys: Seismic surveys involve the use of seismic waves to map the subsurface of a geological formation. Seismic data can be used to create 3D models of the terrain and identify areas with high potential for mineral deposits.

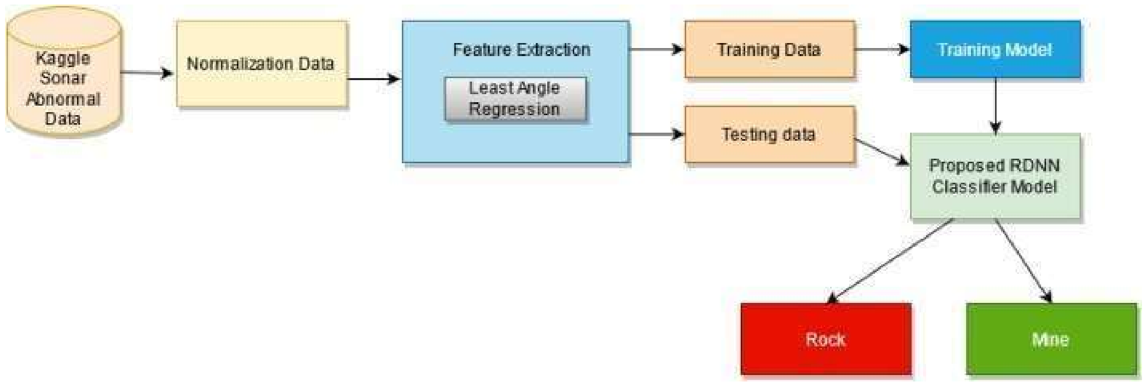
Geochemical analysis: Geochemical analysis involves the analysis of rock and soil samples to identify the presence of minerals and other elements. Geochemical data can be used to create maps and models of mineral deposits.

Hybrid methods: Some researchers have proposed hybrid methods that combine multiple techniques, such as geostatistical modelling and machine learning, to create more accurate and reliable predictive models.

2.2 Comparison of Existing vs Proposed System

Existing systems for rock mine prediction typically involve the use of established techniques such as geostatistical modelling, machine learning, remote sensing, seismic surveys, and geochemical analysis. These techniques have been developed over many years and have been tested and validated in a variety of mining contexts. They offer a range of advantages, including accuracy, reliability, and efficiency. On the other hand, proposed systems for rock mine prediction may involve newer, cutting-edge technologies or techniques that have not yet been fully validated or tested. These proposed systems may offer advantages such as increased accuracy, speed, or cost-effectiveness, but they also come with some degree of uncertainty and risk. Another key difference between existing and proposed systems is the availability and quality of data. This can be a significant challenge, as data collection can be time-consuming and expensive. Overall, while existing systems for rock mine prediction offer many advantages, there is always room for improvement, and proposed systems may offer new and exciting opportunities for the mining industry. The choice of system will depend on various factors, including the specific needs and requirements of the mining operation, the available data, and the resources available for exploration and extraction.

3. System Architecture and Design



Data mining is the process of sorting through large data sets to identify patterns and relationships that can help solve business problems through data analysis. Data mining techniques and tools enable enterprises to predict future trends and make more-informed business decisions. Data mining is a key part of data analytics overall and one of the core disciplines in data science, which uses advanced analytics techniques to find useful information in data sets. At a more granular level, data mining is a step in the knowledge discovery in databases (KDD) process, a data science methodology for gathering, processing and analyzing data. Data mining and KDD are sometimes referred to interchangeably, but they're more commonly seen as distinct things. Data mining is a crucial component of successful analytics initiatives in organizations. The information it generates can be used in business intelligence (BI) and advanced analytics applications that involve analysis of historical data, as well as real-time analytics applications that examine streaming data as it's created or collected. After the data selection process we will mine the data and look for the important attributes in the data and their values, the spending score of the customer that each customer how much he can spend upon, the annual income how each Individual Customer earns and how much he will be capable to spend and how each actually spends, so based upon all of the values we will mine the data and use it for clustering.

import dependencies

In [17]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

Fig 3 DEPEDENCIES

Numpy :- NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, Fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely NumPy stands for Numerical Python. In Python we have lists that serve the purpose of arrays, but they are slow to process. NumPy aims to provide an array object that is up to 50x faster than traditional Python lists. The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy. Arrays are very frequently used in data science, where speed and resources are very important.

Pandas:- Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data. The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008. Pandas allows us to analyze big data and make conclusions based on statistical theories. Pandas can clean messy data sets, and make them readable and relevant. Relevant data is very important in data science

Matplotlib:- Matplotlib: Visualization Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.

1.Create publication quality plots.

2.Make interactive figures that can zoom,

pan, update.

3. Customize visual style and

layout.

4. Export to many file formats.

5. Embed in Jupyter Lab and Graphical User Interfaces.

6.Use a rich array of third-party packages built on Matplotlib

7.Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.

Scikit Learn:- Scikit-learn is an open source data analysis library, and the gold standard for Machine Learning (ML) in the Python ecosystem. Key concepts and features include: Algorithmic decision-making methods, including:

Classification: identifying and categorizing data based on patterns.

Regression: predicting or projecting data values based on the average mean of existing and planned data.

Clustering: automatic grouping of similar data into datasets. Algorithms that support predictive analysis ranging from simple linear regression to neural network pattern recognition.

Interoperability with NumPy, pandas, and matplotlib libraries.

4. Methodology

4.1 Problem Statement

The mining industry is always in search of new mineral deposits to meet growing global demand. However, the exploration and extraction of minerals can be costly, time-consuming, and environmentally impactful. Therefore, the industry requires efficient and accurate methods for predicting the likelihood of finding valuable mineral deposits within geological formations.

The problem with rock mine prediction is that it requires a comprehensive understanding of the geological features and mineral deposits of an area, which can be challenging to obtain. Geological data collection is often expensive, time-consuming, and may require sophisticated equipment and expertise.

Moreover, traditional methods of geological data analysis may not be efficient in processing the vast amounts of data available. Therefore, there is a need for innovative data analysis techniques, such as machine learning and artificial intelligence, to help identify areas with high potential for mineral deposits and make more informed decisions about exploration and extraction.

Therefore, the problem statement for rock mine prediction is to develop accurate, efficient, and cost-effective methods to analyze geological data and predict the likelihood of finding valuable mineral deposits within geological formations.

4.2 Scope

The scope of rock mine prediction is vast, covering a wide range of geological formations, mineral deposits, and mining operations. The applications of rock mine prediction are numerous and include:

Mineral exploration: Rock mine prediction can be used to identify areas with high potential for valuable mineral deposits, reducing the time and resources required for exploration.

Mine planning: Predictive models can be used to optimize mine planning by directing extraction efforts towards areas with the highest potential for mineral deposits.

Resource estimation: Predictive models can help estimate the quantity and quality of mineral deposits in a geological formation, providing valuable information for decision-making.

Risk management: Rock mine prediction can be used to identify areas that pose a higher risk of environmental impact or safety hazards, helping to manage risk.

Environmental impact assessment: Predictive models can be used to assess the environmental impact of mining activities and direct extraction efforts towards areas that are less likely to cause harm.

Economic analysis: Accurate rock mine prediction can help mining companies make more informed decisions about investment and resource allocation, leading to better economic outcomes.

Geological research: Predictive models can be used to further scientific understanding of geological formations and mineral deposits, contributing to the advancement of geological research.

4.3 Advantages of Rock Mine Prediction

Increased efficiency: Rock mine prediction enables mining companies to identify areas with high potential for mineral deposits, allowing them to focus their exploration efforts and resources in the most promising areas. This can result in significant cost savings and increased efficiency in the exploration process.

Improved accuracy: By using advanced technologies such as machine learning, remote sensing, and geostatistical modeling, rock mine prediction can provide highly accurate predictions of the location and quality of mineral deposits. This can help mining companies make more informed decisions about where to mine and how to extract resources, leading to increased productivity and profitability.

Reduced environmental impact: By accurately predicting the location and quantity of mineral deposits, rock mine prediction can help mining companies minimize their environmental impact. By reducing the amount of exploration and drilling required, rock mine prediction can also help reduce the disruption to local ecosystems and communities.

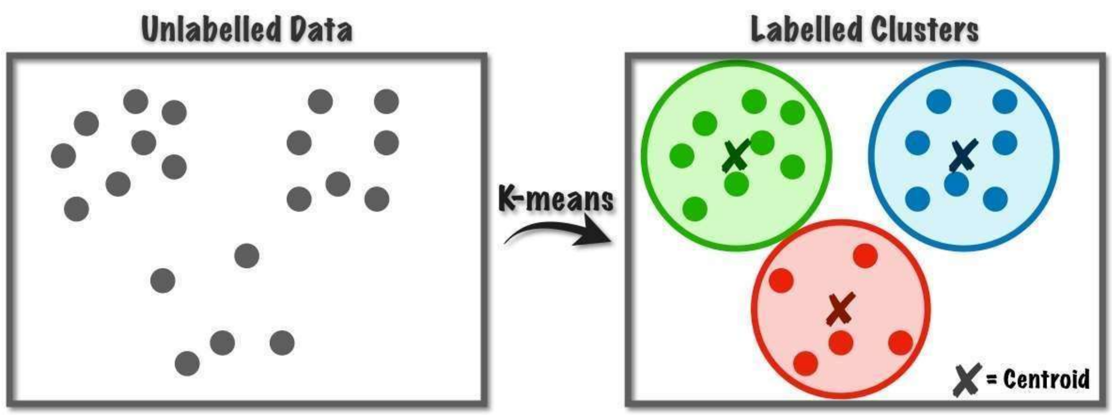
Improved safety: By identifying areas with high potential for mineral deposits, rock mine prediction can help mining companies avoid areas with high risk of geological hazards such as landslides, rockfalls, and earthquakes. This can improve safety for workers and reduce the risk of accidents and injuries.

Cost savings: Rock mine prediction can help mining companies save significant amounts of money by reducing the amount of time and resources required for exploration and drilling. By focusing on the most promising areas, mining companies can reduce the amount of money spent on exploration and increase the return on investment for mining operations.

4.4 K-Means Clustering

4.4.1 K-Means Algorithm

K-Means Clustering is an unsupervised learning algorithm that is used to solve the clustering problems in machine learning or data science. In this topic, we will learn what is K-means clustering algorithm, how the algorithm works, along with the Python implementation of k-means clustering. K   Means Clustering is an Unsupervised Learning algorithm, which groups the unlabeled dataset into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on. It is an iterative algorithm that divides the unlabeled dataset into k different clusters in such a way thateach dataset belongs only one group that has similar properties. It allows us to cluster the data into different groups and a convenient way to discover the categories of groups in the unlabeled dataset on its own without the need for any training. It is a centroid-based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters.



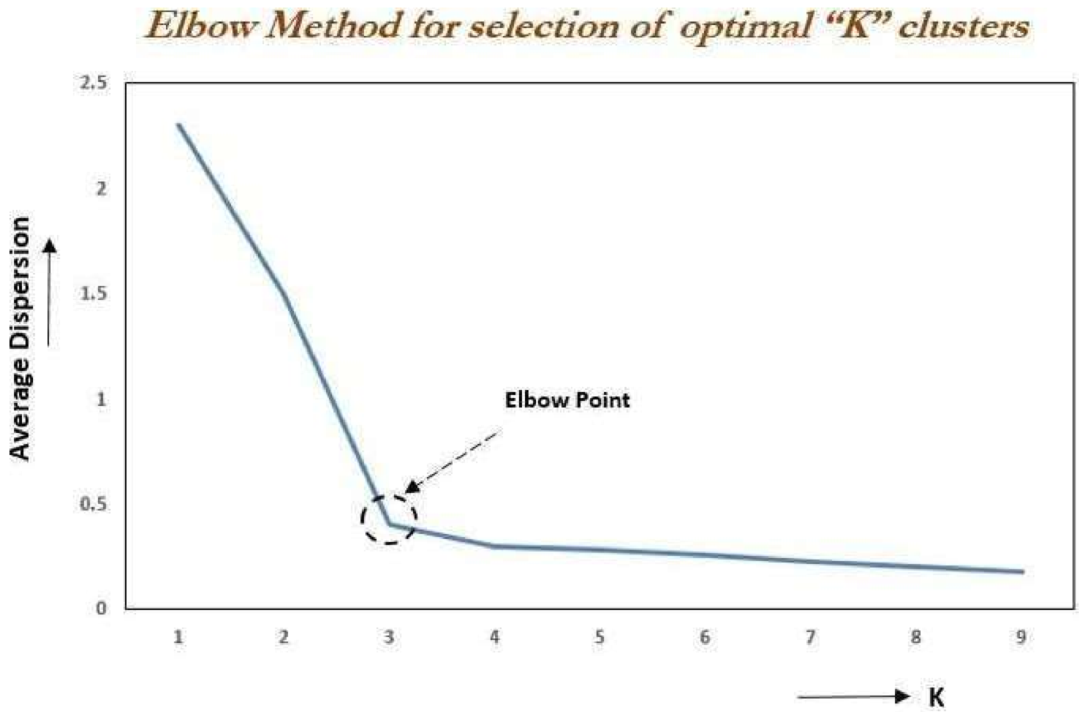
The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm. k-means clustering algorithm mainly performs two tasks:

Determines the best value for K center points or centroids by an iterative process.

Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create cluster.

Elbow Method

It is the simplest and most commonly used iterative type of unsupervised learning algorithm. Unlike supervised learning, we don't have labeled data in K-Means. Some other unsupervised learning algorithms arePCA (Principle Component analysis), K-Medoid, etc. In K-Means, we randomly initialize the K number of cluster centroids in the data (the number of k found using the Elbow Method will be discussed later in this tutorial) and iterates these centroids until no change happens to the position of the centroid. Let's go through the steps involved in K-means clustering for a better understanding. Select the number of clusters for the dataset (K) Select the K number of centroids randomly from the dataset. Now we will use Euclidean distance or Manhattan distance as the metric to calculate the distance of the points from the nearest centroid and assign the points to that nearest cluster centroid, thus creating K clusters.Now we find the new centroid of the clusters thus formed. Again reassign the whole data point based on this new centroid, then repeat step 4. We will continue this for a given number of iterations until the position of the centroiddoesn't change, i.e., there is no more convergence. Finding the optimal number of clusters is an important part of this algorithm. A commonly used method for finding the optimum K value is Elbow Method.



So from the above methods we can say that choosing of k value would be more significant and more convenient in performing the clustering algorithm and to get better results, accuracy and precision.

5. Coding and Testing

5.1 Load the Dataset

import dependencies

In [65]:
```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

data collection and data processing

In [66]:
```
#loading the dataset into pandas dataframe
sonar_data = pd.read_csv('sonar.csv',header = None)
```

5.2 Dataset For Training and Testing

```
0.0025,0.0309,0.0171,0.0228,0.0434,0.1224,0.1947,0.1661,0.1368,0.1430,0.0994,0.2250,0.2444,0.3239,0.3039,0.2410,0.0367,0.1672,0.3038,0.4069,
0.3613,0.1994,0.4611,0.6849,0.7272,0.7152,0.7102,0.8516,1.0000,0.7690,0.4841,0.3717,0.6096,0.5110,0.2586,0.0916,0.0947,0.2287,0.3480,0.2095,
0.1901,0.2941,0.2211,0.1524,0.0746,0.0606,0.0692,0.0446,0.0344,0.0082,0.0108,0.0149,0.0077,0.0036,0.0114,0.0085,0.0101,0.0016,0.0028,0.0014,
R
0.0291,0.0400,0.0771,0.0809,0.0521,0.1051,0.0145,0.0674,0.1294,0.1146,0.0942,0.0794,0.0252,0.1191,0.1045,0.2050,0.1556,0.2690,0.3784,0.4024,
0.3470,0.1395,0.1208,0.2827,0.1500,0.2626,0.4468,0.7520,0.9036,0.7812,0.4766,0.2483,0.5372,0.6279,0.3647,0.4572,0.6359,0.6474,0.5520,0.3253,
0.2292,0.0653,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0000,0.0056,0.0237,0.0204,0.0050,0.0137,0.0164,0.0081,0.0139,0.0111,
R
0.0181,0.0146,0.0026,0.0141,0.0421,0.0473,0.0361,0.0741,0.1398,0.1045,0.0904,0.0671,0.0997,0.1056,0.0346,0.1231,0.1626,0.3652,0.3262,0.2995,
0.2109,0.2104,0.2085,0.2282,0.0747,0.1969,0.4086,0.6385,0.7970,0.7508,0.5517,0.2214,0.4672,0.4479,0.2297,0.3235,0.4480,0.5581,0.6520,0.5354,
0.2478,0.2268,0.1788,0.0898,0.0536,0.0374,0.0990,0.0956,0.0317,0.0142,0.0076,0.0223,0.0255,0.0145,0.0233,0.0041,0.0018,0.0048,0.0089,0.0085,
R
0.0491,0.0279,0.0592,0.1270,0.1772,0.1908,0.2217,0.0768,0.1246,0.2028,0.0947,0.2497,0.2209,0.3195,0.3340,0.3323,0.2780,0.2975,0.2948,0.1729,
0.3264,0.3834,0.3523,0.5410,0.5228,0.4475,0.5340,0.5323,0.3907,0.3456,0.4091,0.4639,0.5580,0.5727,0.6355,0.7563,0.6903,0.6176,0.5379,0.5622,
0.6508,0.4797,0.3736,0.2804,0.1982,0.2438,0.1789,0.1706,0.0762,0.0238,0.0268,0.0081,0.0129,0.0161,0.0063,0.0119,0.0194,0.0140,0.0332,0.0439,
M
0.1313,0.2339,0.3059,0.4264,0.4010,0.1791,0.1853,0.0055,0.1929,0.2231,0.2907,0.2259,0.3136,0.3302,0.3660,0.3956,0.4386,0.4670,0.5255,0.3735,
0.2243,0.1973,0.4337,0.6532,0.5070,0.2796,0.4163,0.5950,0.5242,0.4178,0.3714,0.2375,0.0863,0.1437,0.2896,0.4577,0.3725,0.3372,0.3803,0.4181,
0.3603,0.2711,0.1653,0.1951,0.2811,0.2246,0.1921,0.1500,0.0665,0.0193,0.0156,0.0362,0.0210,0.0154,0.0180,0.0013,0.0106,0.0127,0.0178,0.0231,
M
0.0201,0.0423,0.0554,0.0783,0.0620,0.0871,0.1201,0.2707,0.1206,0.0279,0.2251,0.2615,0.1770,0.3709,0.4533,0.5553,0.4616,0.3797,0.3450,0.2665,
0.2395,0.1127,0.2556,0.5169,0.3779,0.4082,0.5353,0.5116,0.4544,0.4258,0.3869,0.3939,0.4661,0.3974,0.2194,0.1816,0.1023,0.2108,0.3253,0.3697,
0.2912,0.3010,0.2563,0.1927,0.2062,0.1751,0.0841,0.1035,0.0641,0.0153,0.0081,0.0191,0.0182,0.0160,0.0290,0.0090,0.0242,0.0224,0.0190,0.0096,
M
```

5.3 Training and Testing

training and test data

In [74]:
```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.1, stratify=Y, random_state=1)
```

In [75]:
```
print(X.shape, X_train.shape, X_test.shape)
```

```
(208, 60) (187, 60) (21, 60)
```

Model Training! ---> logistic regression model!

In [76]:
```
model  = LogisticRegression()
```

Training the logistic regression model using our training data

In [77]:
```
model.fit(X_train, Y_train)
```

Out[77]:
```
LogisticRegression()
```

5.4 Observing accuracy of Model

count	208.000000	208.000000	208.000000	208.000000	208.000000	208.000000	208.000000	208.000000	208.000000	208.000000	...	208.000000	208.000000	20
mean	0.029164	0.038437	0.043832	0.053892	0.075202	0.104570	0.121747	0.134799	0.178003	0.208259	...	0.016069	0.013420	
std	0.022991	0.032960	0.038428	0.046528	0.055552	0.059105	0.061788	0.085152	0.118387	0.134416	...	0.012008	0.009634	
min	0.001500	0.000600	0.001500	0.005800	0.006700	0.010200	0.003300	0.005500	0.007500	0.011300	...	0.000000	0.000800	
25%	0.013350	0.016450	0.018950	0.024375	0.038050	0.067025	0.080900	0.080425	0.097025	0.111275	...	0.008425	0.007275	
50%	0.022800	0.030800	0.034300	0.044050	0.062500	0.092150	0.106950	0.112100	0.152250	0.182400	...	0.013900	0.011400	
75%	0.035550	0.047950	0.057950	0.064500	0.100275	0.134125	0.154000	0.169600	0.233425	0.268700	...	0.020825	0.016725	
max	0.137100	0.233900	0.305900	0.426400	0.401000	0.382300	0.372900	0.459000	0.682800	0.710600	...	0.100400	0.070900	

8 rows × 60 columns

In [71]:
```
sonar_data[60].value_counts()
```

Out[71]:
```
M    111
R     97
Name: 60, dtype: int64
```

M -> Mines

R -> Rocks

5.5 Model Evaluation

Model Evaluation

In [78]:
```
#accuracy evaluation for train data
X_train_prediction = model.predict(X_train)
training_data_accuracy = accuracy_score(X_train_prediction, Y_train)
print(training_data_accuracy)
```

```
0.8342245989304813
```

In [79]:
```
X_test_prediction = model.predict(X_test)
testing_data_accuracy = accuracy_score(X_test_prediction, Y_test)
print(testing_data_accuracy)
```

```
0.7619047619047619
```

5.6 Predicting System

Making a Predective system

In [81]:
```
input_data = (0.0790,0.0707,0.0352,0.1660,0.1330,0.0226,0.0771,0.2678,0.5664,0.6609,0.5002,0.2583,0.1650,0.4347,0.4515,0.4579,0.3
input_data_as_numpy_array = np.asarray(input_data)

input_data_reshape = input_data_as_numpy_array.reshape(1,-1)

prediction = model.predict(input_data_reshape)

print(prediction)

if(prediction[0] == 'R'):
  print("It was a Rock, The ship is safe!")
else:
  print("Red Alert we have found a mine!")
```

```
['M']
Red Alert we have found a mine!
```

6. Screenshot and Results

In [7]:
```
sonar_data[60].value_counts()
```

Out[7]:
```
M    111
R     97
Name: 60, dtype: int64
```

M -> Mines

R -> Rocks

In [8]:
```
sonar_data.groupby(60).mean()
```

Out[8]:

60	0	1	2	3	4	5	6	7	8	9	...	50	51	52	53	54
M	0.034989	0.045544	0.050720	0.064768	0.086715	0.111864	0.128359	0.149832	0.213492	0.251022	...	0.019352	0.016014	0.011643	0.012185	0.009923
R	0.022498	0.030303	0.035951	0.041447	0.062028	0.096224	0.114180	0.117596	0.137392	0.159325	...	0.012311	0.010453	0.009640	0.009518	0.008567

2 rows × 60 columns

Making a Predective system

In [16]:
```
input_data = (0.0790,0.0707,0.0352,0.1660,0.1330,0.0226,0.0771,0.2678,0.5664,0.6609,0.5002,0.2583,0.1650,0.4347,0.4515,0.4579,0.3
input_data_as_numpy_array = np.asarray(input_data)

input_data_reshape = input_data_as_numpy_array.reshape(1,-1)

prediction = model.predict(input_data_reshape)

print(prediction)

if(prediction[0] == 'R'):
  print("It was a Rock, The ship is safe!")
else:
  print("Red Alert we have found a mine!")
```

```
['M']
Red Alert we have found a mine!
```

7. Conclusion and Future Enhancement

In conclusion, rock mine prediction is an important tool for the mining industry, offering a range of benefits including increased efficiency, improved accuracy, reduced environmental impact, improved safety, and cost savings. Existing systems for rock mine prediction have been developed and tested over many years, and they continue to evolve and improve as new technologies and techniques are developed.

However, there is always room for future enhancement and improvement in the field of rock mine prediction. Some possible future directions for enhancement include:

Integration of multiple data sources: Combining data from different sources such as satellite imagery, geophysical surveys, and geological maps can improve the accuracy and reliability of rock mine prediction.

Incorporation of artificial intelligence: The use of artificial intelligence techniques such as deep learning and neural networks can help mining companies identify patterns in large datasets, leading to more accurate predictions and faster decision-making.

Improved automation: The automation of data processing and analysis can help mining companies to quickly identify areas with high potential for mineral deposits, reducing the time and resources required for exploration.

Integration of sustainability considerations: Incorporating sustainability considerations such as environmental impact and community engagement into rock mine prediction can help mining companies to operate more responsibly and sustainably.

8. References

- "Underwater Mine Detection Using Symbolic Pattern Analysis of Sidescan Sonar Images." Underwater Mine Detection Using Symbolic Pattern Analysis of Sidescan Sonar Images, ieeexplore.ieee.org, https://ieeexplore.ieee.org/document/5160102.

- "Connectionist Bench (Sonar, Mines vs. Rocks)." Connectionist Bench (Sonar, Mines vs. Rocks) | Kaggle, www.kaggle.com, https://www.kaggle.com/datasets/armanakbari/connectionist-bench-sonar-minesvs-rocks.