### MID GRAD CAPSTONE PROJECT-I REPORT

**On**

## SONAR ROCK vs MINE PREDICTION USING MACHINE LEARNING

Submitted in partial fulfilment of the Requirements for the award of the Degree of **Bachelor of Technology**

### in

**ELECTRICAL AND ELECTRONICS ENGINEERING**

Under guidance of

## Dr. G.G. RAJA SHEKAR

### Associate Professor, Dept. of EEE

**By**

### EDE VENKATESH (2100069021) M HRUTESH RAM (2100069018)



#### DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

**K L E F**

Green Fields, Vaddeswaram, Guntur District-522 502

**2023-2024**

# K L E F

#### DEPARTMENT OF ELECTRICAL AND ELECTRONICSENGINEERING



**CERTIFICATE**

This is to certify that the Project Report entitled “**SONAR ROCK vs MINE PREDICTION USING MACHINE LEARNING**” is being submitted by **EDE VENKATESH (2100069021)**

in partial fulfilment for the award of **BACHELOR OF TECHNOLOGY** in **ELECTRICAL AND ELECTRONICS ENGINEERING** to the K L University is a record of bonafide work carried out under our guidance and supervision. The results embodied in this report have not been copied from any other departments/ University/Institute.

**FACULTY INCHARGE HEAD OF THE DEPARTMENT Dr. G.G. RAJASHEKAR Dr. A. PANDIAN**

# K L E F

#### DEPARTMENT OF ELECTRICAL AND ELECTRONICSENGINEERING



**CERTIFICATE**

This is to certify that the Project Report entitled **“SONAR ROCK vs MINE PREDICTION USING MACHINE LEARNING**” is being submitted by **M HRUTESH RAM (2100069018)**

in partial fulfilment for the award of **BACHELOR OF TECHNOLOGY** in **ELECTRICAL AND ELECTRONICS ENGINEERING** to the K L University is a record of bonafide work carried out under our guidance and supervision. The results embodied in this report have not been copied from any other departments/ University/Institute.

**FACULTY INCHARGE HEAD OF THE DEPARTMENT Dr. G.G. RAJASHEKAR Dr. A. PANDIAN**

# K L E F

#### DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING



**DECLARATION**

The Project Report entitled “**SONAR ROCK vs MINE PREDICTION USING MACHINE LEARNING** “This a record of bonafide work of **EDE VENKATESH (2100069021) & M HRUTESH RAM (2100069018)** submitted in partial fulfilment for the award of **BACHELOR OF TECHNOLOGY** in **ELECTRICAL AND ELECTRONICS ENGINEERING** to the K L University. The results embodied in this report have not been copied from any other departments/ University/ Institute.

**EDE VENKATESH ( 2100069021) M HRUTESH RAM (2100069018)**

**ACKNOWLEDGEMENT**

We express sincere gratitude to our Coordinator **Dr. G.G. RAJASHEKAR** for her leadership and constant motivation provided in successful completion of our academic semester.

We record it as my privilege to deeply thank our pioneer **Dr. A. PANDIAN** Head of the Department of Electrical and Electronics Engineering for providing us the efficient faculty and facilities to make our ideas into reality.

We are greatly indebted to our K L Deemed to be university that has provided a healthy environment to drive me to achieve my ambitions and goals. We would like to express our sincere thanks to our project in charges, for the guidance, support, and assistance they have provided in completing this project.

Finally, it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to make this project report success.

## ABSTRACT

Stress is a pervasive issue affecting individuals across various aspects of life, with significant implications for mental and physical well-being. Early detection of stress plays a crucial role in preventing adverse outcomes and promoting timely interventions. This project aims to develop a novel approach for stress detection based on facial expressions using deep learning techniques. Leveraging advances in computer vision and artificial intelligence, we propose a deep learning model trained on a comprehensive dataset of facial expressions to accurately identify signs of stress. The methodology involves pre-processing facial images, designing and training a deep neural network architecture, and evaluating its performance using appropriate metrics. The results demonstrate the effectiveness of the proposed approach in accurately detecting stress from facial expressions, with promising implications for real-world applications in healthcare, wellness, and human-computer interaction. Through this project, we contribute to the ongoing efforts in leveraging technology for mental health assessment and intervention, ultimately aiming to improve the quality of life for individuals affected by stress-related issues.

Stress detection based on facial expressions using deep learning is a critical endeavor in mental health assessment and intervention. Facial cues offer valuable insights into an individual's emotional state, making them a promising avenue for early stress detection. This research project focuses on developing a predictive model for discerning stress-related facial expressions using deep learning techniques.

The study capitalizes on a dataset comprising facial images captured under various stress-inducing scenarios, providing a diverse range of expressions to train the model effectively. Deep learning architectures, such as convolutional neural networks (CNNs), are chosen for their ability to extract intricate features from images and discern subtle patterns indicative of stress.

In the course of the investigation, the dataset undergoes preprocessing to enhance the model's robustness and generalization capabilities. This includes techniques such as image augmentation, normalization, and feature extraction to ensure optimal performance.

The deep learning model is then trained on the preprocessed dataset to learn the complex relationships between facial expressions and stress levels. Leveraging techniques such as transfer learning, the model can leverage knowledge gained from pre-trained networks to improve performance even with limited data.

Performance metrics, including accuracy, precision, recall, and F1-score, are utilized to evaluate the model's effectiveness in detecting stress from facial expressions accurately. Comparative analysis with other machine learning approaches provides insights into the competitiveness and efficacy of deep learning for stress detection.

Overall, this research contributes to the advancement of stress detection methodologies, offering a promising approach for early intervention and support in mental health care settings.

## TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CH NO** | **TITLE NAME** | **PAGE NO** |
|  | ABSTRACT | 7 |
|  | LIST OF FIGURES | 9 |
|  | LIST OF ABBREVIATIONS | 10 |
| **1** | **INTRODUCTION** | **11** |
|  | 1.1 OBJECTIVE | 12 |
|  | 1.2 PROPOSED APPROACH | 12 |
| **2** | **LITERATURE SURVEY** | **13** |
|  | 2.1 PROCESSING METHODS | 14 |
|  | 2.1.1 DATA SET | 14 |
|  | 2.1.2 EXPERMENTAL SETTING | 14 |
|  | 2.1.3 MACHINE LEARNING CLASSIFIERS | 14 |
|  | 2.2 PROPOSED FRAMEWORK | 15 |
|  | 2.2.1 PROPOSED FRAMEWORK METHODS | 15 |
| **3** | **SYSTEM REQUIREMENTS** | **17** |
|  | 3.1 SOFTWARE AND HARDWARE REQUIREMENTS | 17 |
|  | 3.2 MATPLOTLIB | 17 |
|  | 3.3 PANDAS | 19 |
|  | 3.4 SKLEARN | 19 |
| **4** | **IMPLEMENTATION** | **20** |
|  | 4.1 EXPERIMENTAL RESULTS AND DISCUSSIONS | 20 |
|  | 4.2 IMPLEMENTATION CODE | 23 |
| **5** | **SCREENSHOTS** | **25** |
| **6** | **CONCLUSION** | **28** |
|  | **REFERENCE** | **30** |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **FIG.NO** | **FIGURE NAME** | **PAGE NO** |
| 2.1 | PREDICTION FRAMEWORK | 16 |
| 3.1 | MATPLOTLIB PLOT | 18 |
| 3.2 | SKLEARN | 19 |
| 4.1 | DATA CORRELATION REPRESENTATION | 20 |
| 4.2 | DIFFERENT DIMENSIONS OF FREQUENCY | 20 |
| 4.3 | DATA DISTRIBUTION PROCESS IN DENSITY PLT REPRESENTATION | 21 |
| 4.4 |  | 21 |
| 4.5 | SCALED ALGORITHM ACCURACY VALUES | 22 |
| 4.6 | SCALED ALGORITHM COMPARISON | 22 |
| 4.7 | BOSTING ALGORITHMS ACCURACY VALUES | 23 |
| 5.1 | OUTPUT 1 | 25 |
| 5.2 | OUTPUT 2 | 25 |
| 5.3 | DATA CORRELATION REPRESENTATION | 26 |
| 5.4 | ALGORITHM COMPARISON | 26 |
| 5.5 | ALGORITHM COMPARISON | 27 |

## LIST OF ABBREVIATIONS

|  |  |
| --- | --- |
| **LR** | Logistic Regression |
| **LDA** | Linear Discriminant Analysis |
| **KNN** | K-nearest neighbors |
| **CART** | Classification and Regression Tree |
| **NB** | Naïve Bayes Classifier |
| **SVM** | Support Sector Machine |
| **SLR** | Scaled Logistic Regression |
| **SLDA** | Scaled Linear Discriminant Analysis |
| **SKNN** | Scaled K-nearest neighbors |
| **SCART** | Scaled Classification and Regression |
| **SNB** | Scaled Naïve Bayes Classifier |
| **SSVM** | Scaled Support Sector Machine |

**1.INTRODUCTION**

In today's fast-paced and demanding world, stress has become a prevalent issue affecting individuals across various demographics. From students facing academic pressure to professionals dealing with work-related stressors, the impact of stress on mental and physical well-being cannot be understated. Chronic stress not only deteriorates one's quality of life but also contributes to the development of serious health conditions such as anxiety, depression, and cardiovascular diseases. Recognizing the importance of early intervention in managing stress, researchers and healthcare professionals are increasingly turning to innovative technologies for effective detection and monitoring.

One such technology that has garnered significant attention in recent years is the use of facial expressions as a means of assessing stress levels. Human faces are rich sources of information, conveying a myriad of emotions and psychological states. The subtle changes in facial expressions, such as furrowed brows, tense jawlines, or downturned lips, often reflect underlying emotional turmoil, including stress. Leveraging advancements in computer vision and artificial intelligence, researchers are exploring the potential of automated systems to detect and analyze these facial cues, providing valuable insights into an individual's stress levels in real-time.

Traditional methods of stress assessment, such as self-report questionnaires or physiological measurements, have limitations in terms of subjectivity and invasiveness. In contrast, facial expression analysis offers a non-intrusive and objective approach to stress detection, making it suitable for a wide range of applications, including mental health screening, workplace stress management, and human-computer interaction. By harnessing the power of deep learning, a subset of artificial intelligence, researchers aim to develop robust and accurate models capable of discerning stress-related facial expressions with high precision and reliability.

The premise of using deep learning for stress detection lies in its ability to automatically learn intricate patterns and features from large volumes of data. Deep learning models, particularly convolutional neural networks (CNNs), have demonstrated remarkable capabilities in image recognition and classification tasks, making them well-suited for analyzing facial expressions. By training these models on annotated datasets comprising facial images labeled with corresponding stress levels, researchers can teach the algorithms to recognize subtle cues indicative of stress, such as changes in facial muscle movements, skin coloration, or eye gaze patterns.

The development of a reliable stress detection system based on facial expressions using deep learning holds immense potential for various stakeholders. For individuals, such a system could provide timely feedback on their stress levels, enabling proactive self-management strategies and seeking appropriate support when needed. In educational settings, educators and counselors could utilize these technologies to identify students experiencing elevated stress levels and offer targeted interventions to promote mental well-being. In the workplace, employers could implement stress detection systems to monitor employee well-being and create a supportive work environment conducive to productivity and resilience.

Despite the promising prospects of facial expression analysis for stress detection, several challenges persist. Variability in facial expressions across different individuals, cultures, and contexts presents a significant obstacle in model generalization. Moreover, the ethical considerations surrounding privacy, data security, and potential biases inherent in automated systems warrant careful scrutiny and mitigation strategies.

In light of these considerations, this research endeavors to contribute to the burgeoning field of stress detection based on facial expressions using deep learning. By exploring novel methodologies, leveraging state-of-the-art technologies, and addressing existing challenges, we aim to develop a robust and reliable system capable of accurately assessing stress levels from facial cues. Through interdisciplinary collaboration and empirical validation, we aspire to advance the frontiers of mental health technology and foster a healthier, more resilient society.

### OBJECTIVE

1. **Develop a Comprehensive Dataset:**

- Compile a diverse and well-annotated dataset of facial images depicting a range of stress levels across different individuals, demographics, and scenarios.

- Ensure dataset integrity and quality by addressing issues such as bias, imbalance, and variability in facial expressions.

2. **Design and Train Deep Learning Models**:

- Investigate and select appropriate deep learning architectures, such as convolutional neural networks (CNNs), for facial expression analysis and stress detection.

- Implement and fine-tune deep learning models using the collected dataset to learn complex patterns and features indicative of stress-related facial expressions.

3. **Evaluate Model Performance**:

- Assess the accuracy, precision, recall, and F1-score of the trained deep learning models in detecting stress from facial expressions.

- Conduct comprehensive performance evaluations using cross-validation, hold-out validation, and other robust validation techniques to ensure model reliability and generalization.

4. \*\***Address Ethical and Privacy Considerations**:\*\*

- Identify and address ethical considerations surrounding the use of facial expression analysis for stress detection, including privacy, consent, and potential biases.

- Implement privacy-preserving measures and data anonymization techniques to safeguard individuals' confidentiality and mitigate risks associated with data misuse or unauthorized access.

5. \*\***Compare with Existing Methods**:\*\*

- Benchmark the performance of the developed deep learning models against existing methods of stress detection, including self-report questionnaires, physiological measurements, and traditional machine learning approaches.

- Conduct comparative analyses to evaluate the advantages, limitations, and practical implications of facial expression-based stress detection methods.

6. \*\***Explore Real-World Applications**:\*\*

- Investigate potential real-world applications and deployment scenarios for the developed stress detection system, including mental health screening, workplace stress management, and human-computer interaction.

- Collaborate with relevant stakeholders, such as mental health professionals, educators, and technology developers, to explore opportunities for integrating the system into existing frameworks and interventions.

7. \*\***Iterative** **Improvement and Validation**:\*\*

- Continuously iterate and refine the deep learning models based on feedback, empirical observations, and validation results.

- Validate the effectiveness and usability of the stress detection system through pilot studies, user evaluations, and feedback mechanisms to ensure alignment with end-user needs and expectations.

By pursuing these objectives, this research aims to advance the state-of-the-art in stress detection based on facial expressions using deep learning, ultimately contributing to the development of innovative technologies for mental health assessment and intervention.

#### PROPOSED APPROACH

The Sonar Dataset involves the prediction of whether or not an object is a mine or a rock given the strength of sonar returns at different angles. It is a binary (2-class) classification problem. The number of observations for each class is not balanced. There are 208 observations with 60 input variables and 1 output variable. The variable names are as follows:

1. Sonar returns at different angles2.

…

3. Class (M for mine and R for rock)

The baseline performance of predicting the most prevalent class is a classification accuracy of approximately 53%. Top results achieve a classification accuracy

of approximately 88%.

This is a dataset that describes sonar chirp returns bouncing off different services. The 60 input variables are the strength of the returns at different angles. The main concern of analysis in the field of machine learning is being to form a scheduled computational machine for the categorizing the forecast of the objects, based on the attainable information.

**LITERATURE SURVEY**

The article "Stress and anxiety detection using facial cues from videos" explores the use of facial cues to detect stress and anxiety. Published in Biomedical Signal Processing and Control in 2017, the study by Giannakakis et al. focuses on analyzing facial expressions to identify emotional states. This research has implications for developing tools to monitor mental health through video analysis.

The study detects stress and anxiety by analyzing facial cues from videos. Researchers use facial expression analysis to identify patterns associated with stress and anxiety. By examining muscle movements and facial expressions, the study aims to detect emotional states such as stress and anxiety. This approach leverages the relationship between facial cues and emotional responses to develop a method for detecting and monitoring stress and anxiety levels in individuals.

The methodologies used for stress detection in the study "Stress and anxiety detection using facial cues from videos" by Giannakakis et al. include analyzing facial cues from videos. Researchers examine facial expressions and muscle movements to identify patterns associated with stress and anxiety. By leveraging facial expression analysis, the study aims to detect and monitor stress levels in individuals. This approach utilizes the relationship between facial cues and emotional responses to develop a method for stress detection through video analysis.

The study "Detecting Negative Emotional Stress Based on Facial Expression in Real Time" by Zhang et al. delves into the realm of real-time detection of negative emotional stress through facial expressions. Presented at the 2019 IEEE 4th International Conference on Signal and Image Processing (ICSIP), this research sheds light on the potential of utilizing facial cues to identify and monitor negative emotional stress levels in individuals. The study, downloaded on November 18, 2019, carries the DOI 10.1109/siprocess.2019.8868735.

Emotional stress is a prevalent aspect of human experience, impacting individuals' well-being and mental health. Detecting and managing stress levels is crucial for maintaining overall health and quality of life. Traditional methods of stress assessment often rely on self-reporting or physiological measurements, which may have limitations in terms of accuracy and real-time monitoring. In this context, the study by Zhang et al. explores an innovative approach that leverages facial expressions as indicators of negative emotional stress.

Facial expressions are known to convey a wealth of information about an individual's emotional state. By analyzing facial cues such as muscle movements, researchers can discern patterns associated with stress and anxiety. This study aims to capitalize on the relationship between facial expressions and emotional responses to develop a method for real-time detection of negative emotional stress. The ability to detect stress levels promptly and accurately through facial analysis could revolutionize the field of mental health monitoring.

The research methodology employed in this study involves the analysis of facial expressions from videos. By examining facial cues and muscle movements, the researchers seek to identify specific patterns that correspond to negative emotional stress. This approach entails the use of advanced algorithms and tools for automatic facial expression recognition, enabling real-time assessment of emotional states. The study's innovative methodology opens up new possibilities for non-invasive monitoring of stress levels through video analysis.

The implications of this research are significant for both the scientific community and the broader field of mental health monitoring. The ability to detect negative emotional stress in real time using facial expressions could pave the way for the development of novel tools and technologies for stress management and intervention. By harnessing the power of facial cues, researchers can potentially offer individuals a more accessible and efficient means of monitoring their emotional well-being.

In conclusion, the study by Zhang et al. represents a pioneering effort in the realm of real-time detection of negative emotional stress based on facial expression analysis. By leveraging the rich information conveyed through facial cues, the research contributes to advancing the field of mental health monitoring and offers promising avenues for future research and application in stress management.

### METHODOLOGY

The process and methods used for proposing the prediction model is discussed in this section.

**DATASET:**

In our proposed method dataset has been collected from Kaggle Repository. It has come across 10 features which define and differentiate Emotions comprises of 30000 images.

**EXPERIMENTAL SETTING:**

In building of this system we are using Pycharm IDE tool for the purpose of implementing the varied feature selection and model systems. The main motive is to measure the predicting efficiency of the classifier when it is functional and operating and then classifying new samples outside the benefit of perceiving the bona fide class of the samples. The comparators have been designed to implement a 10-fold cross validation trial. The dataset is split into 10 equally distributed subsets. The most exact machine learning classifier is chosen asa base classifier to instruct the nine-subset layer and examine it on the last subset layer. To measure the durability of crafted groundwork, the step is repeated. To appraise the performance of the considered framework, seven different specifications listed as, F measure, accuracy, MCC, error rate, True and False Positive rates, and area under curve (AUC) are used.

**MODELS:**

**1. Sequential Model for Binary Classification:**

This section involves the development and implementation of a sequential neural network model for binary classification.

The architecture typically consists of input, hidden, and output layers, with activation functions such as ReLU in the hidden layers and sigmoid in the output layer for binary classification.

The model is trained using a dataset where each data instance belongs to one of two classes, and performance is evaluated using metrics like accuracy, precision, recall, and F1-score.

**2. Optimization Techniques Comparison:**

Here, the same sequential model from the previous section is trained using different optimization techniques, such as ADAM, SGD (Stochastic Gradient Descent), and RMSPROP (Root Mean Square Propagation).

Each optimization technique has its unique update rules and hyperparameters, which affect the model's convergence speed and final performance.

The comparative analysis aims to identify the most suitable optimization technique for the given classification task based on factors like convergence behavior and computational efficiency.

**3. Comparative Analysis of Models:**

This section involves a detailed comparison of the performance of the sequential models trained with different optimization techniques.

Metrics such as accuracy, loss curves, and convergence speed are compared to assess the effectiveness of each optimization technique.

Insights gained from the comparative analysis help in understanding the trade-offs between optimization techniques and selecting the most appropriate one for the task at hand.

**4. Sequential Model for Multi-Class Classification:**

In this section, the sequential neural network model is extended to handle multi-class classification tasks where each data instance belongs to one of multiple classes.

The architecture is modified to have multiple output nodes corresponding to the number of classes, with softmax activation function in the output layer.

Model performance is evaluated using metrics suitable for multi-class classification, such as categorical cross-entropy loss and accuracy.

**5. Binary Classification with Mini-Batch Evaluations:**

Mini-batch evaluations involve updating model parameters using a subset of the training data (mini-batch) instead of the entire dataset in each iteration.

This section explores the impact of mini-batch size on model training efficiency, convergence speed, and generalization.

Performance metrics and convergence behavior are analyzed for different mini-batch sizes to determine the optimal setting for the given classification task.

**MACHINE LEARNING CLASSIFIERS:**

In below Section explains Machine Learning classifiers with different algorithms.

* 1. Neural Network: An unreal neural network linked group of nodes, known as perceptron’s, and is like a colossal network of neurons in a human brain. In this, the perceptron algorithm has been used to train the machine. It is for a managed learning of two-fold classifiers that can decide if an input belongs to some unique category or not.
  2. Support Vector Machine (SVM) networks, are super visional learning algorithms that figure out the data used for classification and backsliding analysis. SVM model is a depiction of the examples as points in space, charted to create separate categories, divided by a clear chasm. New samples are then mapped into that same space and then concluded to belong to a category based on the side of the chasm they fall.
  3. Random Forest: Random Forest comes under the category of tree type classifiers, in this the dataset values are inspected separately and by the same distribution of all the trees in the forest. Internal valuation monitors strength, errors and the correlations which are implemented to display the response to the growing number of features that have been used in splitting.
  4. Adaboost is capable of being used in a partnership with many other types of classifiers to boost performance. Adaboost is often said to be best the out of-the-box classifier. Information collected at each stage of the Adaboost algorithm of each training sample is stuffed into the tree viable algorithm in such a way that later trees favor to focus on harder to classify examples.
  5. Logistic Regression-Bayesian networks are aimed acyclic graphs whose nodes show variables in the Bayesian sense. Each node is correlated with a probability function that takes a specific set of values, as input, for the node's parent variables, and gives the probability distribution of the variable represented by the node.

### PROPOSED FRAMEWORK

The main concern of analysis in the field of machine learning is being to forma scheduled computational machine for the categorizing the forecast of the objects, based on the attainable information. The outcome of proposed framework helps to predict the triggered sound waves reflect from surface Rock or a Mine.

### PROPOSED FRAMEWORK METHODS:

Broadly in physical world or realistic issues, there is no curb over the types of data. Some dire pre-processing like removal of missing values, feature selection, etc.are always required. Machine learning focuses on taking up contemporarytechniques to process huge amount of complex data with lower expense. The abstract view of proposed framework has been represented in Figure 1. Figure 1 describes the framework of the prediction model created to determine the surface to be a rock or a mine based on about 61 factors or features, processed by 10 different classifier models, which give outputs with an acceptable accuracy and precision percentage.

1. **Preprocessing:** Missing values are removed by replacing them by mean value imputation.
2. **Feature Selection:** Mean Gini index is used to rank the important features. The top 50 features ranked by mean gini index is selected and fed to the prediction model.
3. **Prediction Model:** Different ML classifiers are explored and implemented to find the best possible solution. Random forest, being an ensemble model has shown the highest performance with 83.17% of accuracy. The results are further optimized by applying feature selection technique to feed the prediction model with the best features and accuracy reached at 90.20% after optimization. The outcome of this proposed framework helps to predict the targeted surface to be a Rock or a Mine.

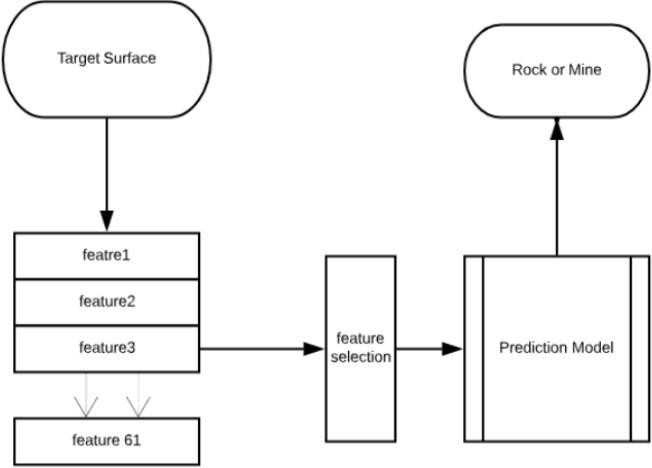


Figure 2.1. Prediction Framework

## SYSTEM REQUIREMENTS

#### SOFTWAREAND HARDWARE REQUIREMENTS

**Software Requirements**

Operating System: Windows10 and other versions. Language: Python 3.8.5

#### Hardware requirements

Ram: 8 GB and more

Processor: Any Intel Processor

Hard Disk: 6 GB and more

Speed: 1 GHZ and more

**MATPLOTLIB**

Humans are very visual creatures: we understand things better when we see things visualized. However, the step to presenting analyses, results or insights canbe a bottleneck: you might not even know where to start or you might have already a right format in mind, but then questions like “Is this the right way to visualize the insights that I want to bring to my audience?” will have definitely come across your mind.

When you’re working with the Python plotting library Matplotlib, the first step to answering the above questions is by building up knowledge on topics like:

The **Anatomy of Matplotlib plot:** what is a subplot? What are the Axes? What exactly is a figure?

**Plot Creation:** which could raise questions about what module you exactly need to import (pylab or pyplot?), how you exactly should go about initializing the figure and the Axes of your plot, how to use matplotlib in Jupyter notebooks, etc.

**Plotting Routines:** from simple ways to plot your data to more advanced ways of visualizing your data.

[**Basic**](https://www.datacamp.com/community/tutorials/matplotlib-tutorial-python?utm_source=adwords_ppc&utm_campaignid=1455363063&utm_adgroupid=65083631748&utm_device=c&utm_keyword&utm_matchtype=b&utm_network=g&utm_adpostion&utm_creative=278443377086&utm_targetid=aud-299261629574%3Adsa-473406587955&utm_loc_interest_ms&utm_loc_physical_ms=9040217&gclid=Cj0KCQjwp4j6BRCRARIsAGq4yMFVF8eBK1trZgbyQ8fMhWZNjneS8B8Cbhvpw-VqR9I3ysnWLHi4CggaAuf_EALw_wcB&plot_customizations) **Plot customization**, with a focus on plot legends and text, titles, axes labels and plot layout.

**Saving, Showing and Clearing your plots:** show the plot, save one or more figures to, for example, pdf files, clear the axes, clear the figure or close the plot,etc.

(To practice matplotlib interactively, try the free Matplotlib chapter at the start of this [Intermediate Python course](https://www.datacamp.com/courses/intermediate-python-for-data-science) or see Data Camp’s [Viewing 3D Volumetric Data](https://www.datacamp.com/community/tutorials/matplotlib-3d-volumetric-data) [With Matplotlib](https://www.datacamp.com/community/tutorials/matplotlib-3d-volumetric-data) tutorial to learn how to work with matplotlib’s event handler API.)

#### What does the Matplotlib looks like?

At first sight, it will seem that there are quite some components to consider when you start plotting with this Python data visualization library. You’ll probably agree with me that it’s confusing and sometimes even discouraging seeing the amount of code that is necessary for some plots, not knowing where to start yourself and which components you should use. Luckily, this library is very flexible and has a lot of handy, built-in defaults that will help you out tremendously. As such, you don’t need much to get started: you need to make the necessary imports, prepare some data, and you can start plotting with the help of the plot() function! When you are ready don’t forget to show your plot using the show() function.

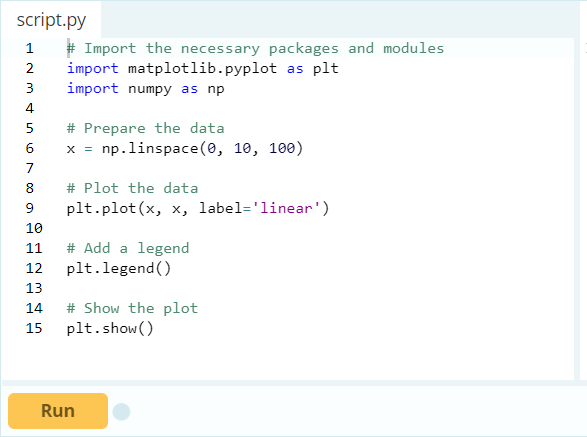


Figure 3.1. matplotlib plot

### PANDAS

**pandas** is a Python package that provides fast, flexible, and expressive data structures designed to make working with structured (tabular, multidimensional, potentially

heterogeneous) and time series data both ea level building block for doing practical, **re**

|  |  |  |
| --- | --- | --- |
| sy and intuitive. It aims to be | | the funda |
| **al world** data | analysis in Python. Add | |
|  | | |

mental high- itionally, it

has the broader goal of becoming **the most powerful and flexible open source data analysis**

**/ manipulation tool available inany language**. It is already well on its way toward this goal. Pandas is a popular Python package for data science, and with good reason: it offers powerful, expressive and flexible data structures that make data manipulation and analysis easy, among many other things. The DataFrame is one of these structures.

### SKLEARN

Simple and efficient tools for predictive data analysis. Accessible to everybody, and reusable in various contexts. Built on NumPy, SciPy, and matplotlib. Open source, commercially usable - BSD license. Used in: Classification, Regression, Clustering, Dimensionality reduction, Model selection, pre-processing

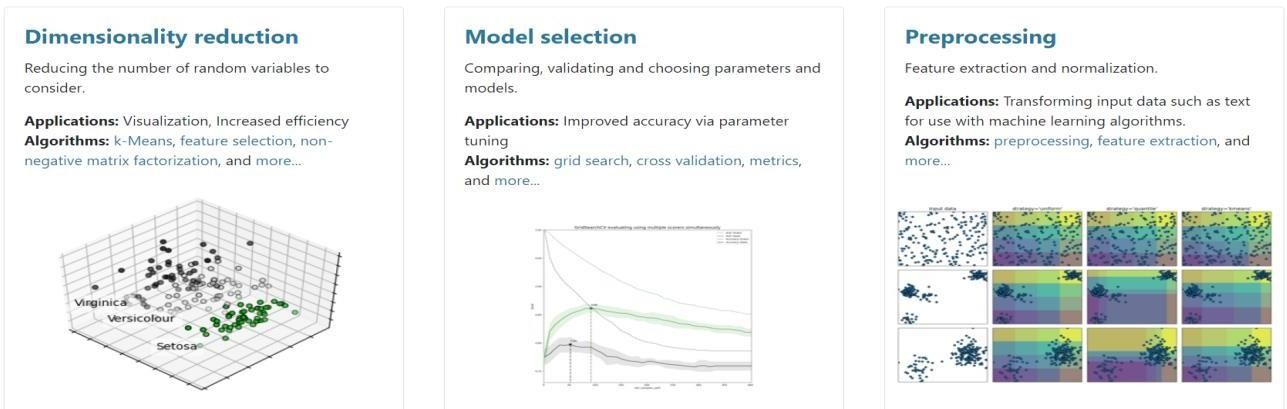
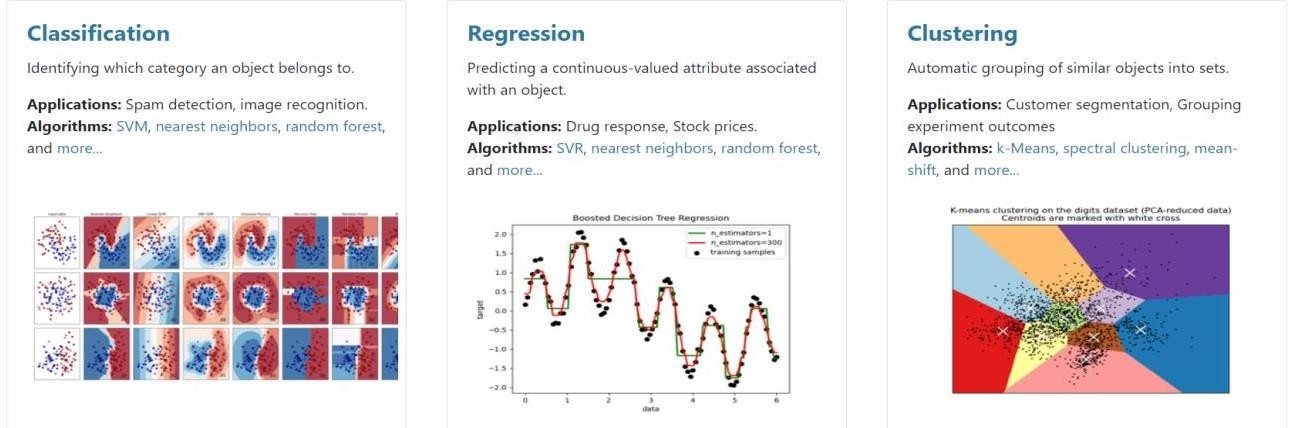


Figure 3.2. sklearn

## IMPLEMENTATION

### EXPERIMENTAL RESULTS AND DISCUSSION:

This section discusses parameter evaluation metrics to measure the performance of various machine learning algorithms. The results of 10-fold cross validation method are presented graphically and discussed much in detail.

### DATA CORRELATION REPRESENTATION

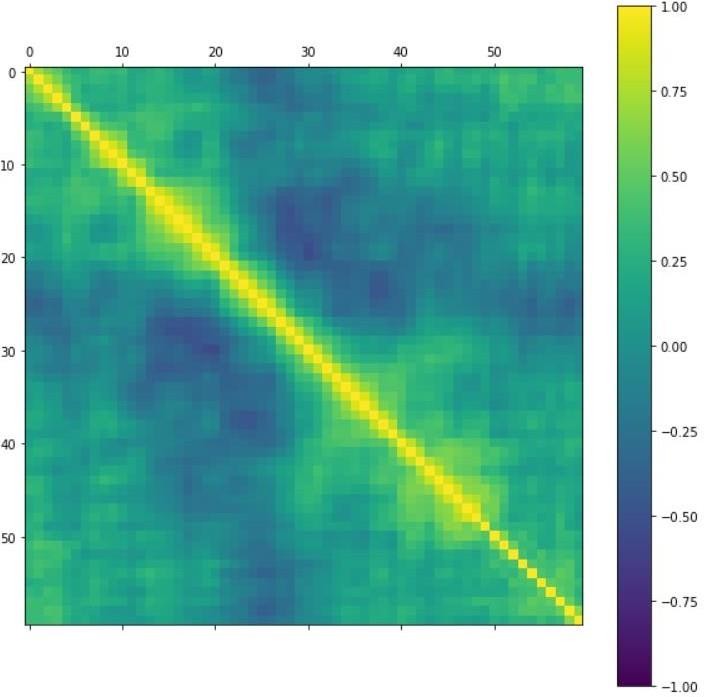


Figure 4.1. Data correlation representation

#### DIFFERENT DIMENSIONS OF FREQUENCY IN VERTICAL AXIS AND HORIZONTAL AXIS

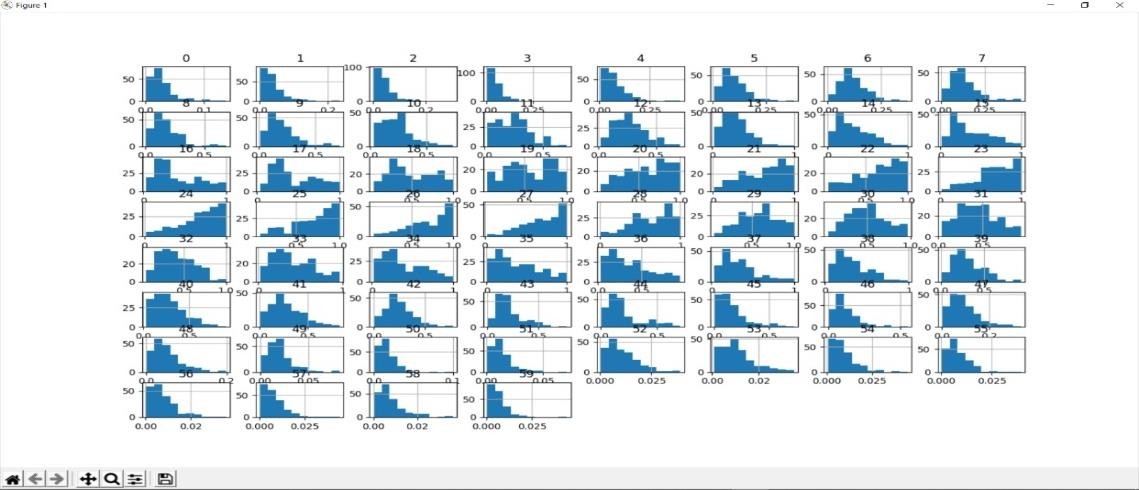


Figure 4.2. Different dimensions of frequency

#### DATA DISTRIBUTION PROCESS IN DENSITY PLOTS REPRESENTATION:

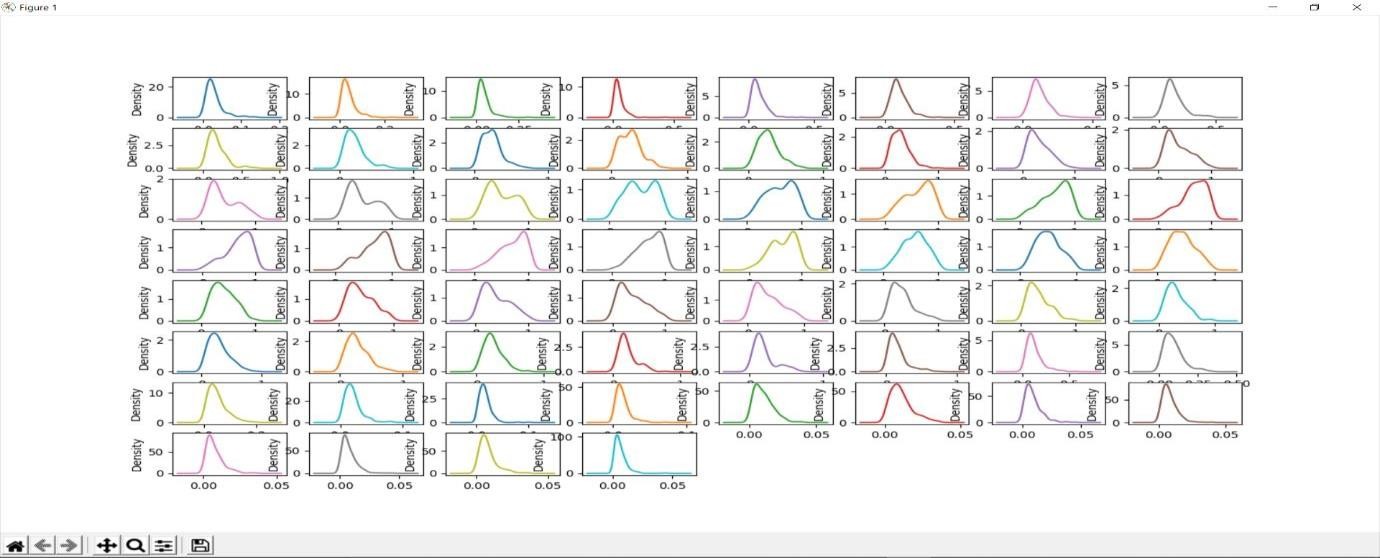


Figure 4.3. Data distribution process in density plot representation

#### ALGORITHMS COMPARISON:

In this module, we compare Numerical Data based on Quartile Values withdifferent algorithms.

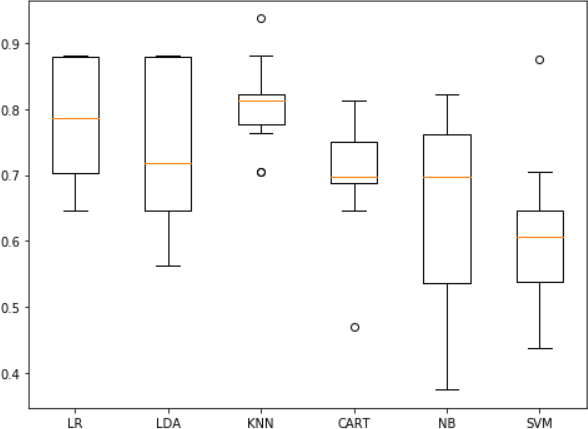


Figure 4.4. Algorithms comparsion

* + **LR:** Logistic Regression.
  + **LDA:** Linear discriminant analysis.
  + **KNN:** k-nearest neighbors.
  + **CART:** Classification and Regression Trees.
  + **NB:** Naive Bayes classifier.
  + **SVM:** Support Vector Machine.

#### SCALED ALGORITHMS ACCURACY VALUES

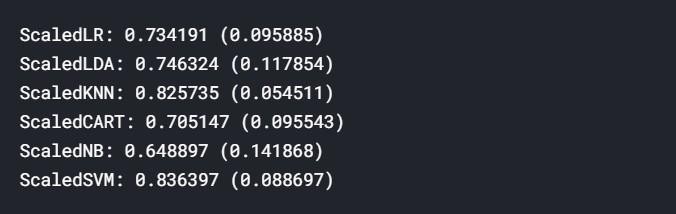


Figure 4.5. Scaled Algorithm accuracy values

#### SCALED ALGORITHM COMPARISON

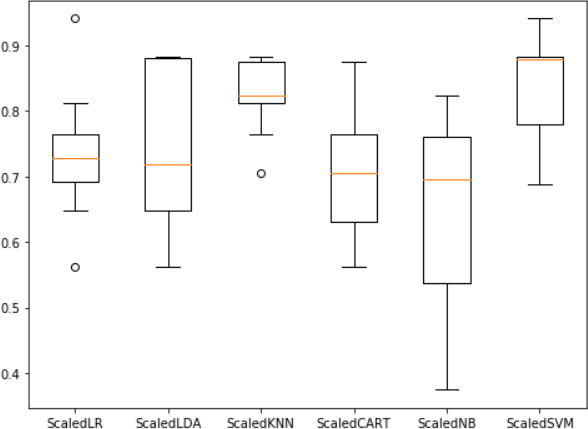


Figure 4.6. Scaled Algorithm comparison

* + **ScaledLR:** Scaled Logistic Regression.
  + **ScaledLDA:** Scaled Linear discriminant analysis.
  + **ScaledKNN:** Scaled k-nearest neighbors.
  + **ScaledCART:** Scaled Classification and Regression Trees.
  + **ScaledNB:** Scaled Naive Bayes classifier.
  + **ScaledSVM:** Scaled Support Vector Machine.

#### BOSTING ALGORITHMS ACCURACY VALUES

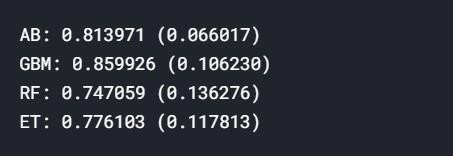


Figure 4.7. Bosting algorithms accuracy values

#### IMPLEMENATION CODE

Packages or module to be importedfrom matplotlib import pyplot

from pandas import read\_csv from pandas import set\_option

from pandas.plotting import scatter\_matrix

from sklearn.preprocessing import StandardScaler from sklearn.model\_selection import train\_test\_split from sklearn.model\_selection import KFold

from sklearn.model\_selection import cross\_val\_scorefrom sklearn.model\_selection import GridSearchCV from sklearn.metrics import classification\_report

from sklearn.metrics import confusion\_matrix from sklearn.metrics import accuracy\_score from sklearn.pipeline import Pipeline

from sklearn.linear\_model import LogisticRegression from sklearn.tree import DecisionTreeClassifier

from sklearn.neighbors import KNeighborsClassifier

from sklearn.discriminant\_analysis import LinearDiscriminantAnalysis from sklearn.naive\_bayes import GaussianNB

from sklearn.svm import SVC

from sklearn.ensemble import AdaBoostClassifier

from sklearn.ensemble import GradientBoostingClassifierfrom sklearn.ensemble import RandomForestClassifier from sklearn.ensemble import ExtraTreesClassifier

For final model to derive output#

Finalize Model

# prepare the model

scaler = StandardScaler().fit(X\_train) rescaledX = scaler.transform(X\_train) model = SVC(C=1.5) model.fit(rescaledX, Y\_train)

# estimate accuracy on validation dataset rescaledValidationX = scaler.transform(X\_validation)

predictions = model.predict(rescaledValidationX) print(accuracy\_score(Y\_validation, predictions)) print(confusion\_matrix(Y\_validation, predictions)) print(classification\_report(Y\_validation, predictions))

## SCREENSHOTS

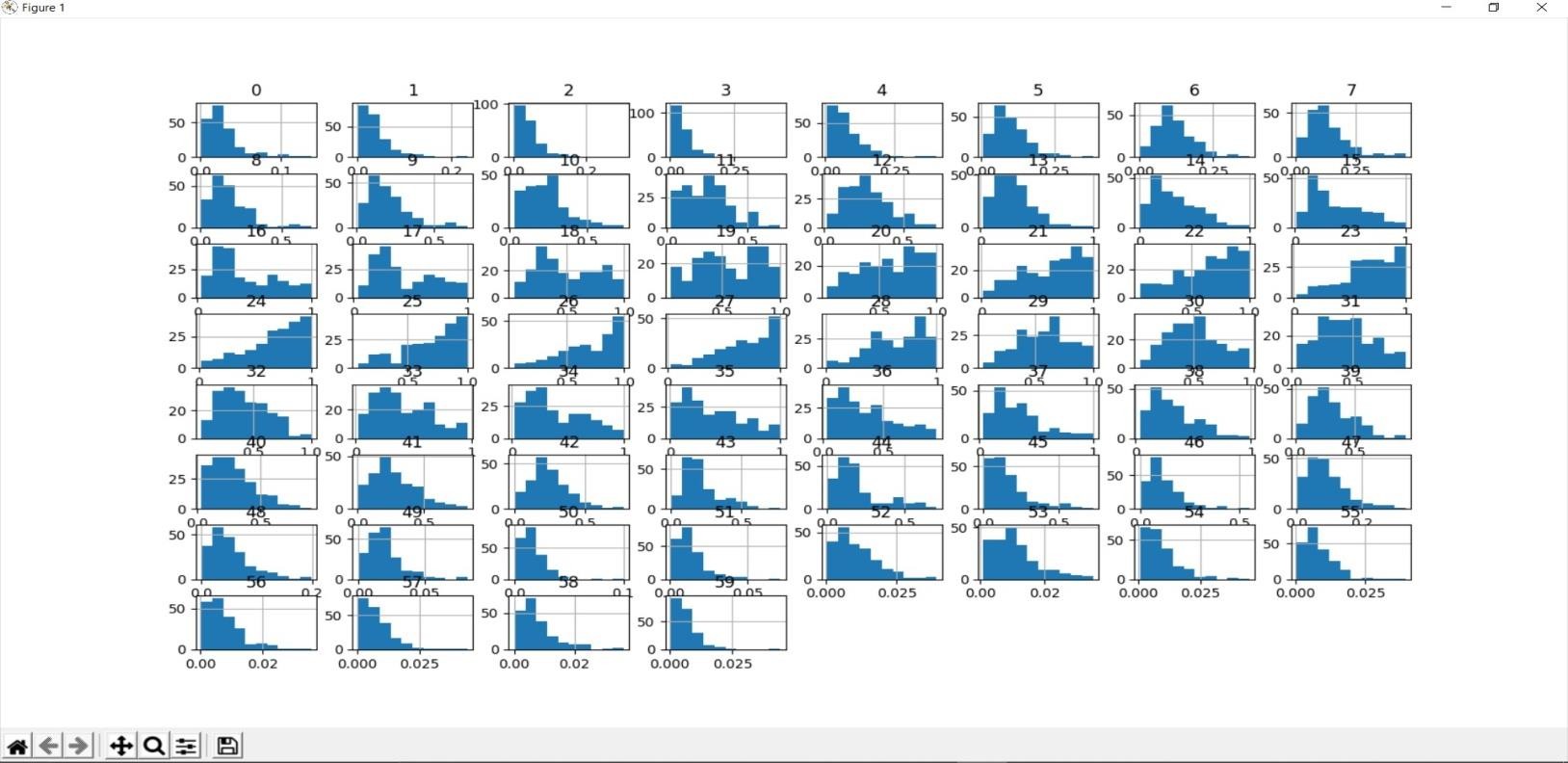


Figure 5.1 Output 1

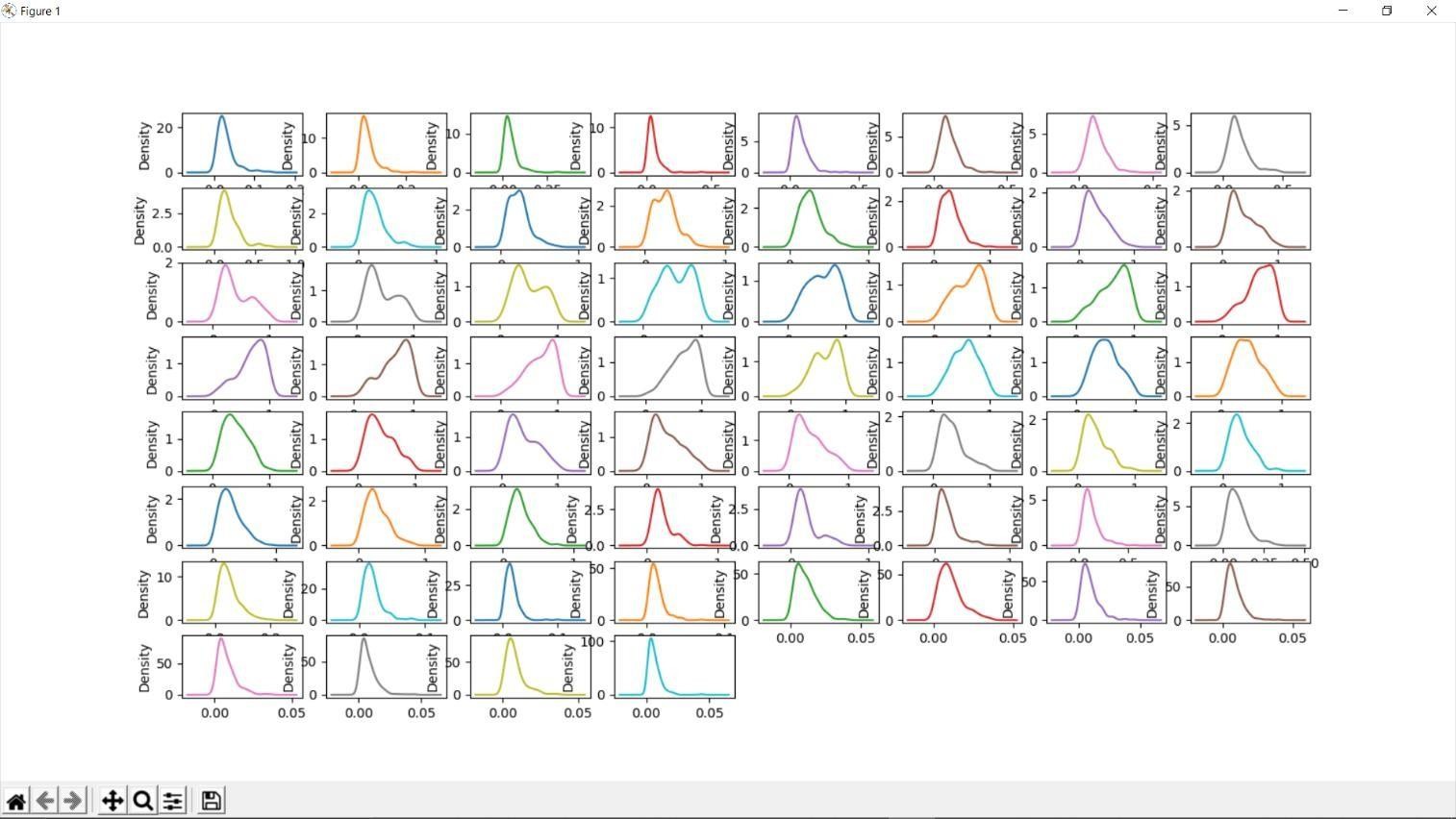


Figure 5.2. Output 2

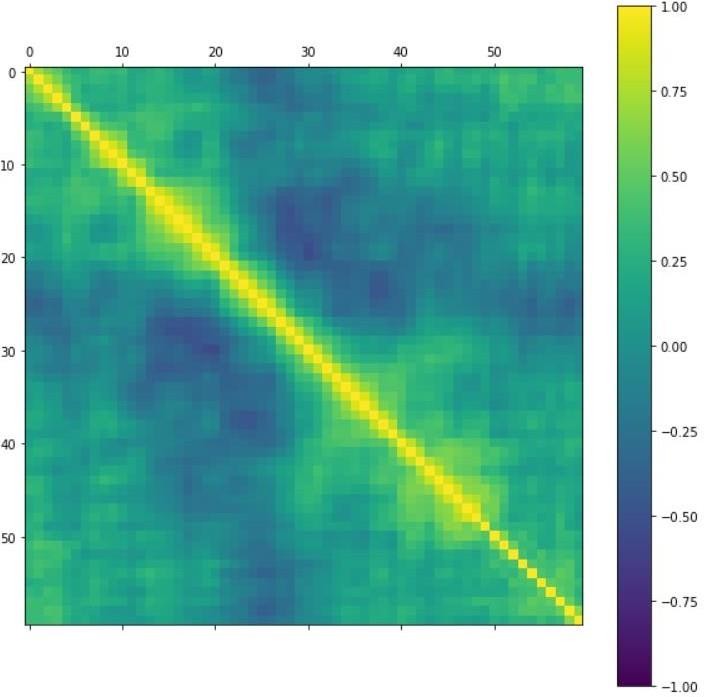


Figure 5.3. Data correlation representation

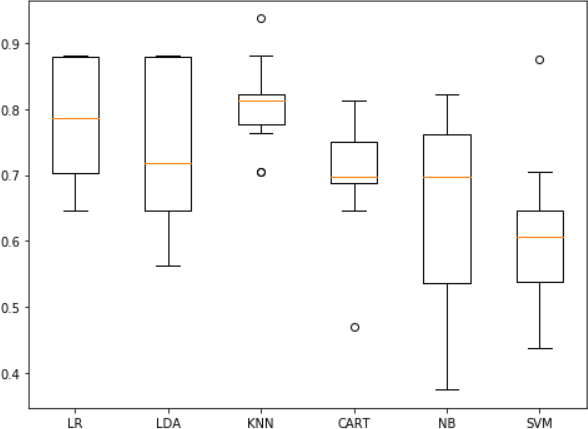


Figure 5.4 Algorithm Comparison

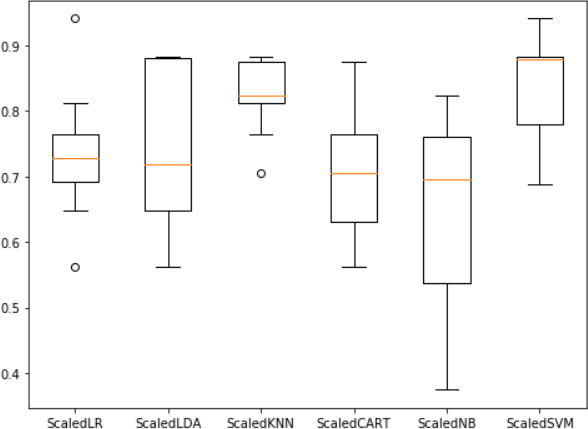


Figure 5.5 Algorithm Comparison

## CONCULSION

An adequate prediction miniature, united with the machine learning classifyingfeatures, is proposed which can conclude if the target of the sound wave is either a rock or a mine or any other organism or any kind of other body. Research is carried out for predicting the best possible result for the target to be a rock or a mine, whichis found to be best through the random forest model, which is an ensemble tree- based classifier in machine learning with the highest accuracy rate of 83.17% and giving the best ROCAUC rate 0.93, with least error for better elaboration of this prediction model. For future work more, complex data will be handled using big data Hadoop framework. With random forest algorithm, the results are further optimizedby feature selection to get the accuracy of 91.15%.

SONAR technique, which relays on certain parameters to be able to detectthe obstacle or the surface is a rock or a mine. Machine learning has drawn the attention of maximum part of the technology related and based industries, by showing advancements in the predictive analytics. The main aim is to emanate a capable prediction representative, united by the machine learning algorithmic characteristics, which can figure out if the target of the sound wave is either a rock or a mine or any other organism or any kind of other body. This attempt is a clear-cut case study which comes up with a machine earning plan for the grading of rocks and minerals, executed on a huge, highly spatial and complex SONAR dataset.

## REFERENCE

* . Ho, Tin Kam. Random Decision Forests (PDF). Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14– 16 August 1995. pp. 278–282. (1995).
* Dura, Esther, et al. "Active learning for detection of mine-like objects in side-scan sonar imagery." IEEE Journal ofOceanic Engineering 30.2: 360-371 (2005).
* Erkmen, Burcu, and TülayYıldırım. "Improving classification performanceof sonar targets by applying general regression neural network with PCA." Expert Systems with Applications 35.1-2: 472-475. (2008).
* Zhi-Hua Zhou and Yuan Jiang. [NeC4.5: Neural Ensemble Based C4.5.](http://rexa.info/paper/e6315f170eea3cc52ade688d0abb40f3372ec35d)
* IEEE Trans.Knowl.Data Ang, 16. 2004.[[View Context](https://archive.ics.uci.edu/ml/support/Connectionist%2BBench%2B(Sonar%2C%2BMines%2Bvs.%2BRocks)" \l "e6315f170eea3cc52ade688d0abb40f3372ec35d)].
* Jianbin Tan and David L. Dowe. [MML Inference of Oblique Decision Trees.](http://rexa.info/paper/b525865598a98b326df461e4d9d2647543347eb4)
* Australian Conference on Artificial Intelligence. 2004. [[View Context](https://archive.ics.uci.edu/ml/support/Connectionist%2BBench%2B(Sonar%2C%2BMines%2Bvs.%2BRocks)#b525865598a98b326df461e4d9d2647543347eb4)].
* Jeremy Kubica and Andrew Moore. [Probabilistic Noise Identification andData Cleaning.](http://rexa.info/paper/fcba21d24e397e9639efbad04b5be06fbe9c0df2) ICDM. 2003. [[View Context](https://archive.ics.uci.edu/ml/support/Connectionist%2BBench%2B(Sonar%2C%2BMines%2Bvs.%2BRocks)#fcba21d24e397e9639efbad04b5be06fbe9c0df2)].
* Dennis DeCoste. [Anytime Query-Tuned Kernel Machines via CholeskyFactorization.](http://rexa.info/paper/03534af8424987a10b9e264147f082059665f7b7) SDM. 2003. [[View Context](https://archive.ics.uci.edu/ml/support/Connectionist%2BBench%2B(Sonar%2C%2BMines%2Bvs.%2BRocks)#03534af8424987a10b9e264147f082059665f7b7)].
* Dennis DeCoste. [Anytime Interval-Valued Outputs for Kernel Machines: FastSupport Vector](http://rexa.info/paper/ff68b57a8755189599b4cbefec533e7fc64a5195) [Machine Classification via Distance Geometry.](http://rexa.info/paper/ff68b57a8755189599b4cbefec533e7fc64a5195) ICML. 2002. [[View Context](https://archive.ics.uci.edu/ml/support/Connectionist%2BBench%2B(Sonar%2C%2BMines%2Bvs.%2BRocks)#ff68b57a8755189599b4cbefec533e7fc64a5195)].
* Ayhan Demiriz and Kristin P. Bennett and Mark J. Embrechts. [A GeneticAlgorithm](http://rexa.info/paper/7a13c34f809076592c23c1e2c9c6a9a72464d911) [Approach for Semi-Supervised Clustering.](http://rexa.info/paper/7a13c34f809076592c23c1e2c9c6a9a72464d911) E-Business Department, Verizon Inc.. 2002. [[View Context](https://archive.ics.uci.edu/ml/support/Connectionist%2BBench%2B(Sonar%2C%2BMines%2Bvs.%2BRocks)#7a13c34f809076592c23c1e2c9c6a9a72464d911)].
* Michail Vlachos and Carlotta Domeniconi and Dimitrios Gunopulos and George Kollios and Nick Koudas. [Non-linear dimensionality reduction techniques for](http://rexa.info/paper/df4390b4164d70e186a080009bf9b23fb367478c) [classification and](http://rexa.info/paper/df4390b4164d70e186a080009bf9b23fb367478c) [visualization.](http://rexa.info/paper/df4390b4164d70e186a080009bf9b23fb367478c) KDD. 2002. [[View Context](https://archive.ics.uci.edu/ml/support/Connectionist%2BBench%2B(Sonar%2C%2BMines%2Bvs.%2BRocks)#df4390b4164d70e186a080009bf9b23fb367478c)].
* Xavier Llor and David E. Goldberg and Ivan Traus and Ester Bernad i Mansilla. [Accuracy,](http://rexa.info/paper/412e0644abd8fccf9fb195db3ed361c2ffd3695c) [Parsimony, and Generality in Evolutionary Learning Systems viaMultiobjective Selection.](http://rexa.info/paper/412e0644abd8fccf9fb195db3ed361c2ffd3695c) IWLCS. 2002. [[View Context](https://archive.ics.uci.edu/ml/support/Connectionist%2BBench%2B(Sonar%2C%2BMines%2Bvs.%2BRocks)#412e0644abd8fccf9fb195db3ed361c2ffd3695c)].