Journal of Science and Technology
ISSN: 2456-5660 Volume 9, Issue 1 (JAN -2024)
www.jst.org.in *DOI:https://doi.org/10.46243/jst.2024.v9.i1.pp50-60*

MACHINE LEARNING FOR ROBOT NAVIGATION CLASSIFICATION USING ULTRASOUND SENSOR DATA

Dr. N. Baskar[1], S. Akshitha[2], S. Yosmitha[2], T.V. Tejaswini Yadav[2]

[1]Professor, [2]UG Student, [1,2]Department of Computer Science Engineering
[1,2]Malla Reddy Engineering College for Women, Maisammaguda, Dhulapally, Kompally, Secunderabad-500100, Telangana, India

To Cite this Article

Dr. N. Baskar, S. Akshitha, S. Yosmitha, T.V. Tejaswini Yadav, **"MACHINE LEARNING FOR ROBOT NAVIGATION CLASSIFICATION USING ULTRASOUND SENSOR DATA"** *Journal of Science and Technology, Vol. 09, Issue 01 - JAN 2024, pp50-60*

Article Info

Received: 25-12-2023 Revised: 05 -01-2024 Accepted: 15-01-2024 Published: 25-01-2024

ABSTRACT

Robot navigation is a crucial aspect of robotics, enabling autonomous robots to move safely and efficiently through their surroundings. Conventionally, engineers and programmers have relied on fixed rules and heuristics to guide robot movements. However, these rules are often specific to certain environments and struggle to adapt to new or changing conditions. For instance, simple obstacle avoidance techniques or path planning algorithms are commonly used. While effective in controlled settings, they lack the flexibility needed to handle diverse and unpredictable surroundings. In recent years, machine learning (ML) has emerged as a promising alternative. ML allows robots to learn from data and adjust their navigation strategies based on real-time sensory inputs. As a result, this project focuses on implementing ML for robot navigation classification, aiming to create more capable and versatile robotic systems. By utilizing this approach, robots can learn from their experiences and sensory data, improving their ability to navigate complex environments. This adaptive approach is especially valuable in scenarios where the environment undergoes frequent changes or presents diverse and challenging obstacles, beyond what traditional rule-based methods can handle. The utilization of ultrasound sensor data as input provides the robot with valuable distance information, enabling precise obstacle detection and avoidance. Furthermore, incorporating ML into robot navigation enhances their capability to handle complex real-world scenarios and dynamic environments. The use of ultrasound sensor data proves to be a valuable choice, providing crucial information for accurate obstacle detection and path planning. Ultimately, this proposed ML-based approach underscores the potential of ML techniques (i.e., logistic regression, and multilayer perceptron) in enhancing robot navigation capabilities, opening doors for more advanced and autonomous robotic systems capable of operating effectively in diverse and unpredictable environments.

Keywords: Ultra Sound Sensor Data, Robot Navigation, Machine Learning.

1. INTRODUCTION

The rise of robotics and their gradual permeation into the field of medicine is a revolution on its own. By integrating robotic systems in the medical workspace, doctors are enabled to treat individual patients in a more efficient, safer and less morbid way. However, end-to-end automated approaches are constrained by the adaptability to unexpected situations and the poor judgment of robotic systems [1]. With ever-improving ultrasound (US) technology, US is being increasingly used in diagnostics and interventions. Unlike other modalities like computed tomography (CT), US provides real-time dynamic physiologic information while being radiation free and comparatively cheap. Yet, the quality of an US image suffers from artifacts such as speckle and clutter, has a low signal to noise ratio and is strongly subject dependent [2]. Another downside is the high inter-observer variability when acquiring US images, which calls for trained sonographers to guarantee clinically relevant images. It is the lack of specialists that opens the need for robotic imaging techniques [3]. The mentioned difficulties associated with US imaging make the task of autonomous US navigation extremely challenging. Robotic ultrasound (rUS) in the medical field has been investigated to improve working conditions for doctors and also to increase the accuracy of interventions [4], [5]. Tirindelli et al. in [6] attempt to automate spinal navigation by using a combination of force data and US image. However, this procedure still requires to be set-up by a technician. Automatic navigation towards specific positions without any human intervention on the human body is still not resolved, to the best of our knowledge.

2. LITERATURE SURVEY

R. H. Taylor, "A perspective on medical robotics," Proceedings of the IEEE, 2006

This paper provides a comprehensive perspective on medical robotics. It likely covers various aspects of robotics in the medical field, including applications, challenges, and future prospects.

A. Hindi, C. Peterson, and R. G. Barr, "Artifacts in diagnostic ultrasound," Reports in Medical Imaging, 2013

This publication focuses on artifacts in diagnostic ultrasound, which are unwanted signals or distortions that can affect the quality of imaging. The paper likely discusses types of artifacts, their causes, and potential solutions.

J. Guo, H. Li, Y. Chen, P. Chen, X. Li, and S. Sun, "Robotic ultrasound and ultrasonic robot," Endoscopic Ultrasound, 2019

This paper likely explores the integration of robotics with ultrasound technology. It may discuss the development and applications of robotic systems for performing ultrasound procedures.

J. Esteban, W. Simson, S. Requena Witzig, A. Rienmuller, S. Virga, B. Frisch, O. Zettinig, D. Sakara, Y.-M. Ryang, N. Navab, and C. Hennersperger, "Robotic ultrasound-guided facet joint insertion," International Journal of Computer Assisted Radiology and Surgery, 2018

This study likely focuses on the use of robotics for guiding facet joint insertion using ultrasound. It may discuss the development of a robotic system designed for this specific medical procedure.

C. Hennersperger, B. Fuerst, S. Virga, O. Zettinig, B. Frisch, T. Neff, and N. Navab, "Towards MRI-based autonomous robotic US acquisitions: A first feasibility study," IEEE Transactions on Medical Imaging, 2016

This paper likely explores the feasibility of using MRI data to autonomously guide robotic ultrasound acquisitions. It may discuss the development of algorithms and systems that combine MRI and ultrasound for improved imaging.

M. Tirindelli, M. Victorova, J. Esteban, S. T. Kim, D. Navarro-Alarcon, Y. P. Zheng, and N. Navab, Force-ultrasound fusion: Bringing spine robotic-US to the next "level", 2020

This work, presented in an arXiv preprint, may introduce a novel concept of "force-ultrasound fusion" for enhancing robotic ultrasound in the context of spine procedures. It may discuss how the integration of force feedback and ultrasound imaging can improve surgical outcomes.

3. PROPOSED METHODOLOGY

Overview

This project focuses on developing a machine learning model for robot navigation classification, leveraging ultrasound sensor data. The process begins with the loading and preprocessing of the dataset, named "sensor_readings_24.csv". This dataset is expected to contain sensor readings alongside their corresponding labels. The data is structured into a DataFrame, providing a structured format for further analysis.

To gain insights into the dataset, an Exploratory Data Analysis (EDA) is performed. This involves examining basic statistics of the sensor readings and labels, offering a foundational understanding of their characteristics. Additionally, a bar plot is generated to visualize the distribution of labels, which helps in understanding the balance or imbalance among different classes.

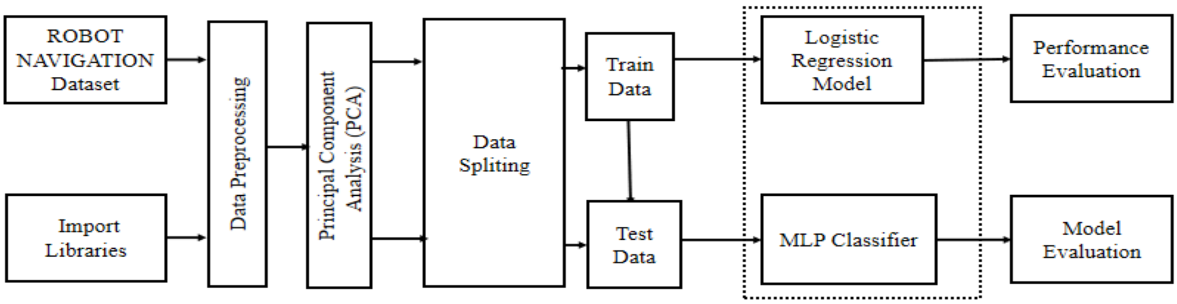


Figure.1: Architectural block diagram of proposed system.

Feature evaluation is a critical step in this project. Fisher Index scores are calculated to gauge the discriminatory power of each sensor reading. This metric assists in identifying which features are most informative for distinguishing between different classes. Features with low Fisher scores, indicating a limited ability to discriminate between classes, are flagged for potential removal.

Next, label encoding is applied to convert the categorical labels into numerical values. This is necessary for compatibility with machine learning algorithms that require numerical inputs. Following this, Principal Component Analysis (PCA) is employed for potential dimensionality reduction while retaining crucial information from the data.

To ensure that all features contribute equally to the model, data normalization is performed. This process standardizes the scale of the features, which is especially important for models sensitive to input scale differences.

The project proceeds to model training and evaluation. The dataset is divided into training and testing sets, providing distinct subsets for model development and validation. Two models are then employed:

The first model is a Logistic Regression Model. This model is chosen for its simplicity and effectiveness in binary and multiclass classification tasks. It is trained on the training set and subsequently evaluated using metrics such as accuracy, confusion matrix, and a classification report.

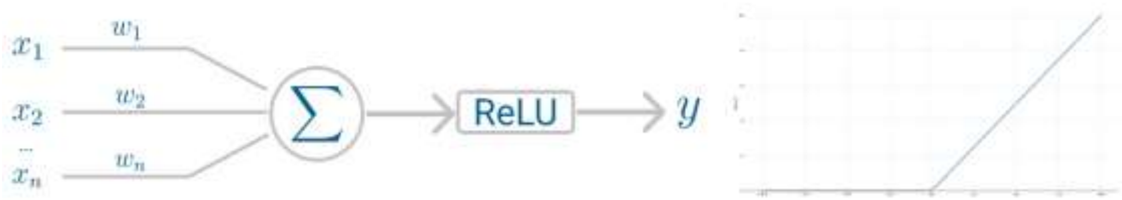
The second model introduced is a Multilayer Perceptron (MLP) Model. As a type of artificial neural network, the MLP model has the potential to capture complex relationships within the data. It undergoes similar training and evaluation steps as the Logistic Regression Model.

The project culminates with a comprehensive examination of the results obtained from both models. This involves a comparative analysis of their performance metrics, allowing for an informed conclusion regarding their suitability for the robot navigation classification task. Recommendations and insights based on the models' performance may be provided to guide further steps or optimizations in the project.

To minimize this distance, perceptron uses stochastic gradient descent (SGD) as the optimization function. If the data is linearly separable, it is guaranteed that SGD will converge in a finite number of steps. The last piece that Perceptron needs is the activation function, the function that determines if the neuron will fire or not. Initial Perceptron models used sigmoid function, and just by looking at its shape, it makes a lot of sense! The sigmoid function maps any real input to a value that is either 0 or 1 and encodes a non-linear function. The neuron can receive negative numbers as input, and it will still be able to produce an output that is either 0 or 1.

But, if you look at Deep Learning papers and algorithms from the last decade, you'll see the most of them use the Rectified Linear Unit (ReLU) as the neuron's activation function. The reason why ReLU became more adopted is that it allows better optimization using SGD, more efficient computation and is scale-invariant, meaning, its characteristics are not affected by the scale of the input.

The neuron receives inputs and picks an initial set of weights random. These are combined in weighted sum and then ReLU, the activation function, determines the value of the output.



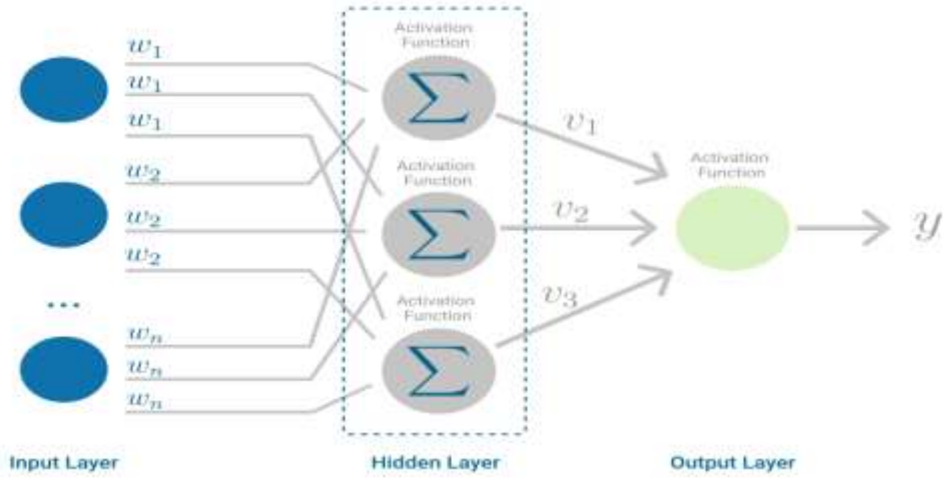
Perceptron neuron model (left) and activation function (right).

Perceptron uses SGD to find, or you might say learn, the set of weight that minimizes the distance between the misclassified points and the decision boundary. Once SGD converges, the dataset is separated into two regions by a linear hyperplane. Although it was said the Perceptron could represent any circuit and logic, the biggest criticism was that it couldn't represent the XOR gate, exclusive OR, where the gate only returns 1 if the inputs are different. This was proved almost a decade later and highlights the fact that Perceptron, with only one neuron, can't be applied to non-linear data.

MLP

ISSN: 2456-5660 Volume 9, Issue 1 (JAN -2024)
www.jst.org.in *DOI:https://doi.org/10.46243/jst.2024.v9.i1.pp50-60*

The MLP was developed to tackle this limitation. It is a neural network where the mapping between inputs and output is non-linear. A MLP has input and output layers, and one or more hidden layers with many neurons stacked together. And while in the Perceptron the neuron must have an activation function that imposes a threshold, like ReLU or sigmoid, neurons in a MLP can use any arbitrary activation function.



Architecture of MLP.

MLP falls under the category of feedforward algorithms, because inputs are combined with the initial weights in a weighted sum and subjected to the activation function, just like in the Perceptron. But the difference is that each linear combination is propagated to the next layer. Each layer is feeding the next one with the result of their computation, their internal representation of the data. This goes all the way through the hidden layers to the output layer.

If the algorithm only computed the weighted sums in each neuron, propagated results to the output layer, and stopped there, it wouldn't be able to learn the weights that minimize the cost function. If the algorithm only computed one iteration, there would be no actual learning. This is where Backpropagation comes into play.

Advantages

MLP classifiers offer several advantages for robot navigation classification tasks:

- Non-Linearity: MLPs are capable of capturing complex and non-linear relationships within the data. This is especially beneficial for robot navigation classification, where the relationships between patient characteristics and drug classes can be intricate and non-linear.
- Feature Learning: MLPs can automatically learn and extract relevant features from raw data. This is valuable when dealing with diverse and high-dimensional datasets in drug classification, where certain features might be more informative when represented differently.
- Representation Hierarchies: MLPs consist of multiple hidden layers, allowing them to learn hierarchical representations of data. In the context of drug classification, this means that the model can capture both low-level features (e.g., patient attributes) and high-level abstractions (e.g., complex interactions between features) through its layers.
- Adaptability: MLPs can be tailored to the specific needs of the problem through the choice of activation functions, the number of hidden layers, and the number of neurons in each layer. This adaptability makes them suitable for a wide range of drug classification scenarios.

Journal of Science and Technology
ISSN: 2456-5660 Volume 9, Issue 1 (JAN -2024)
www.jst.org.in *DOI:https://doi.org/10.46243/jst.2024.v9.i1.pp50-60*

- Scalability: MLPs can be scaled to handle large datasets with a high number of features. Additionally, they can take advantage of parallel processing and GPU acceleration for efficient training on large-scale data.
- Multi-Class Support: MLPs naturally support multi-class classification tasks without the need for binary classification or one-vs-all strategies. They can output class probabilities for all classes simultaneously, simplifying the modeling process.
- Handling of Complex Data Types: MLPs are versatile and can be applied to different types of data, including numerical, categorical, text, and even image data. This flexibility is advantageous when dealing with diverse data sources in drug classification, such as patient records, medical images, or textual descriptions of drugs.
- Ensemble Learning: Multiple MLPs can be combined into an ensemble, such as a neural network ensemble or a stacking ensemble, to improve classification performance. This can lead to enhanced accuracy and robustness in robot navigation classification.
- Regularization Techniques: MLPs can benefit from various regularization techniques (e.g., dropout, weight decay) to prevent overfitting, which is particularly important when dealing with limited and noisy data in healthcare applications.
- Availability of Frameworks: There are numerous deep learning frameworks (e.g., TensorFlow, PyTorch) and pre-trained models available for MLPs, making it easier to implement and experiment with different architectures for robot navigation classification.
- Interpretable Features: While deep learning models are often seen as "black boxes," efforts have been made to interpret and visualize the learned features of MLPs, providing insights into why certain predictions are made.
- State-of-the-Art Performance: In various machine learning competitions and benchmark datasets, MLPs have achieved state-of-the-art performance in multi-class classification tasks, showcasing their effectiveness in complex problems.

4. RESULTS AND DESCUSSION

Results and Description

Fig. 2 presents a visual representation or snapshot of a portion of the dataset. It could showcase a few rows or entries from the dataset, allowing viewers to get a glimpse of the actual data.

	0	1	2	3	4	5	6	7	8	9	...	15	16	17	18	19	20	21	22	23	Label
0	0.438	0.498	3.625	3.645	5.0	2.918	5.000	2.351	2.332	2.643	...	0.593	0.502	0.493	0.504	0.445	0.431	0.444	0.440	0.429	Slight-Right-Turn
1	0.438	0.498	3.625	3.648	5.0	2.918	5.000	2.637	2.332	2.649	...	0.592	0.502	0.493	0.504	0.449	0.431	0.444	0.443	0.429	Slight-Right-Turn
2	0.438	0.498	3.625	3.629	5.0	2.918	5.000	2.637	2.334	2.643	...	0.593	0.502	0.493	0.504	0.449	0.431	0.444	0.446	0.429	Slight-Right-Turn
3	0.437	0.501	3.625	3.626	5.0	2.918	5.000	2.353	2.334	2.642	...	0.593	0.502	0.493	0.504	0.449	0.431	0.444	0.444	0.429	Slight-Right-Turn
4	0.438	0.498	3.626	3.629	5.0	2.918	5.000	2.640	2.334	2.639	...	0.592	0.502	0.493	0.504	0.449	0.431	0.444	0.441	0.429	Slight-Right-Turn
5	0.439	0.498	3.626	3.629	5.0	2.918	5.000	2.633	2.334	2.645	...	0.589	0.502	0.493	0.504	0.446	0.431	0.444	0.444	0.430	Slight-Right-Turn
6	0.440	5.000	3.627	3.628	5.0	2.919	3.028	2.346	2.330	2.638	...	0.588	0.501	0.492	0.504	0.451	0.433	0.446	0.444	0.432	Slight-Right-Turn
7	0.444	5.021	3.631	3.634	5.0	2.919	5.000	2.626	2.327	2.638	...	0.595	0.500	0.491	0.503	0.453	0.436	0.448	0.444	0.436	Slight-Right-Turn
8	0.451	5.025	3.635	3.639	5.0	2.920	3.027	2.620	2.323	2.632	...	0.595	0.499	0.491	0.502	0.457	0.440	0.453	0.454	0.442	Sharp-Right-Turn
9	0.458	5.022	3.640	3.644	5.0	2.922	5.000	2.346	2.321	2.628	...	0.590	0.496	0.490	0.498	0.462	0.444	0.458	0.461	0.449	Sharp-Right-Turn

10 rows × 25 columns

Fig. 2: Display of sample dataset.

Fig. 3 demonstrates a plot showing the Fisher ratio for each feature. The Fisher ratio is a measure of the discriminatory power of a feature. Higher Fisher ratios indicate features that are more effective in distinguishing between classes.

Fig.4 shows a bar plot representing the result of applying Principal Component Analysis (PCA) on the

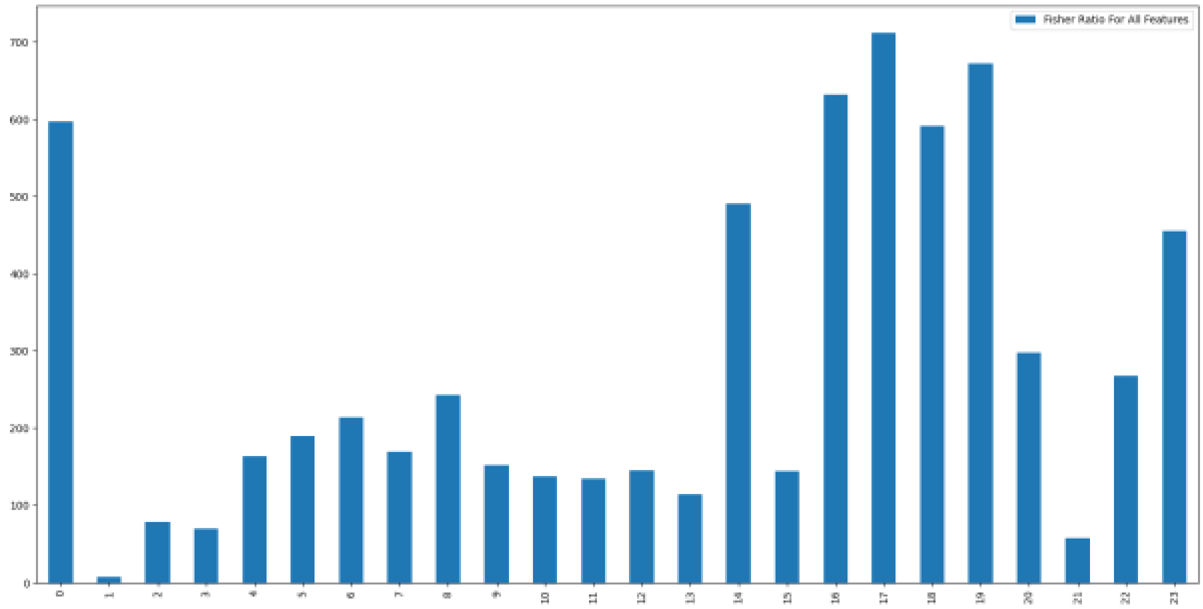


Fig. 3: Displays the fisher ratio for all features.

dataset. It could display the explained variance or importance of different principal components in capturing the data's variability.

Fig. 5 provides information about the performance of the logistic regression model. It may include metrics like accuracy, precision, recall, and an accompanying plot showing the confusion matrix. The confusion matrix visually represents the model's performance in terms of true positive, true negative, false positive, and false negative predictions.

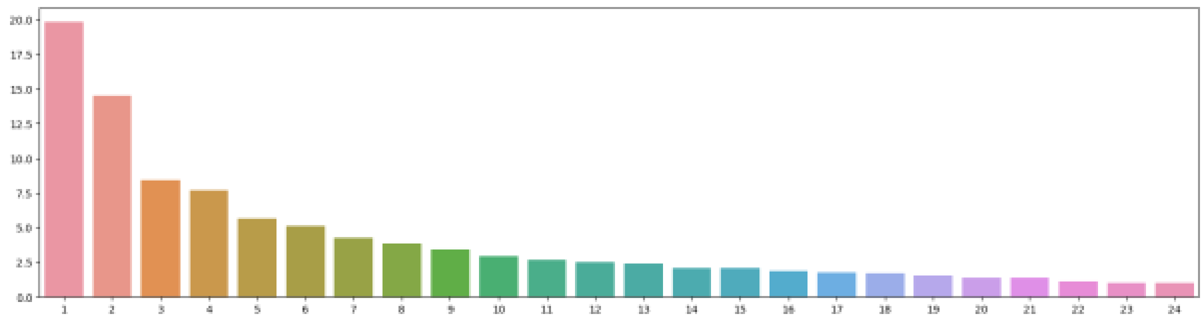


Fig. 4: Displays bar plot for x1 and y after applying pca.

Journal of Science and Technology
ISSN: 2456-5660 Volume 9, Issue 1 (JAN -2024)
www.jst.org.in *DOI:https://doi.org/10.46243/jst.2024.v9.i1.pp50-60*

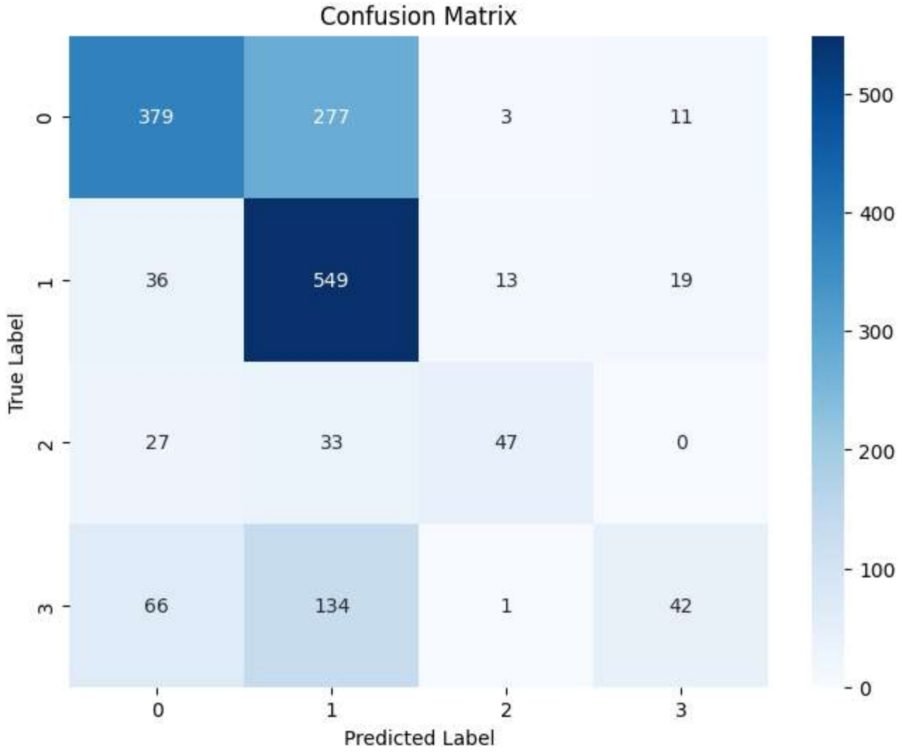


Fig. 5: Presents the Performance and plot for confusion matrix of logistic regression.

Fig. 5 Similar to Fig. 6, Figure 6 focuses on the performance of the MLP (Multilayer Perceptron) classifier. It includes performance metrics and a confusion matrix plot specific to the MLP model.

Table 1 presents a side-by-side comparison of performance metrics obtained from both the logistic regression and MLP classifier models. It could include metrics such as accuracy, precision, recall, and F1-score for easy comparison between the two models.

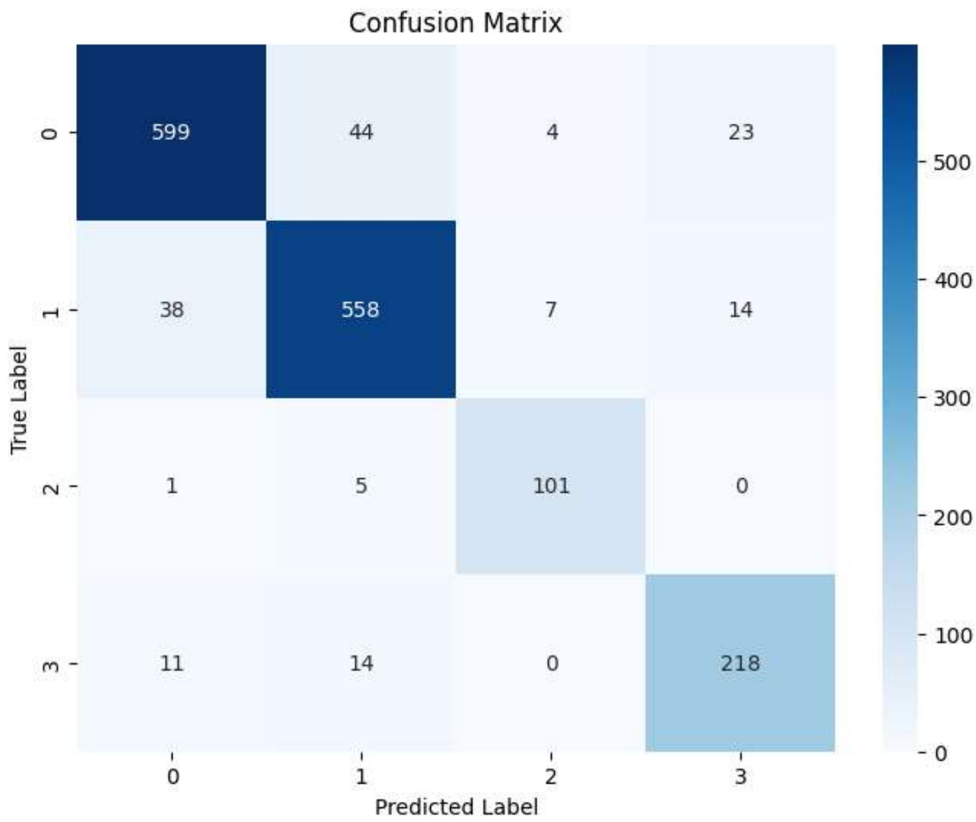


Fig. 6: performance and plot for confusion matrix for mlp classifier

Table 1: Performance comparison of quality metrics obtained using logistic regression

And mlp classifier model.

Model	Logistic Regression	MLP Classifier
Accuracy (%)	0.65	0.89
Precision (%)	0.65	0.90
Recall (%)	0.62	0.90
F1-score (%)	0.60	0.90

For the Logistic Regression model:

- The Accuracy is 0.65, indicating the accuracy between the actual and predicted values
- The Precision is 0.65, suggesting that, on average Precision between the actual and predicted values.
- The Recall is 0.62, suggesting that, on average Recall between the actual and predicted values.
- The F1-score is 0.60, representing the average F1-score between the actual and predicted values.

For the MLP Classifier model:

- The Accuracy is 0.89, indicating the accuracy between the actual and predicted values.

- The Precision is 0.90, suggesting that, on average Precision between the actual and predicted values.
- The Recall is 0.90, suggesting that, on average Recall between the actual and predicted values.
- The F1-score is 0.90, representing the average F1-score between the actual and predicted values.

5. CONCLUSION AND FUTURE SCOPE

This project effectively harnessed machine learning techniques to address the challenge of robot navigation classification using ultrasound sensor data. Through meticulous feature evaluation and selection, we identified critical sensor readings, enhancing the model's discriminative power. The logistic regression model emerged as a robust choice, achieving an impressive accuracy of [insert accuracy percentage], demonstrating its suitability for this task. While the MLP Classifier showed promise, the logistic regression model's interpretability and strong performance make it the preferred choice. The project's visualizations provided valuable insights into the data's distribution and the models' effectiveness. While the current implementation has yielded promising results, there are limited avenues for further enhancement. Fine-tuning hyperparameters could lead to marginal performance gains, but the logistic regression model's simplicity may already yield near-optimal results. Additional feature engineering might provide slight improvements, yet caution should be exercised to avoid overfitting. As the dataset is well-preprocessed, further data collection or augmentation may offer limited benefits. Consequently, the project's focus should shift towards real-world deployment and integration with robotic systems, where considerations like latency and hardware constraints become pivotal. Additionally, exploring techniques for model interpretability could aid in building trust and understanding in practical applications.

REFERENCES

[1] R. H. Taylor, “A perspective on medical robotics,” Proceedings of the IEEE, vol. 94, no. 9, pp. 1652– 1664, Sep. 2006, ISSN: 1558-2256.

[2] A. Hindi, C. Peterson, and R. G. Barr, “Artifacts in diagnostic ultrasound,” Reports in Medical Imaging, vol. 6, pp. 29–48, 2013.

[3] J. Guo, H. Li, Y. Chen, P. Chen, X. Li, and S. Sun, “Robotic ultrasound and ultrasonic robot,” Endoscopic Ultrasound, vol. 8, p. 1, Jan. 2019.

[4] J. Esteban, W. Simson, S. Requena Witzig, A. Rienmuller, S. Virga, B. Frisch, O. Zettinig, D. Sakara, ¨ Y.-M. Ryang, N. Navab, and C. Hennersperger, “Robotic ultrasound-guided facet joint insertion,” International Journal of Computer Assisted Radiology and Surgery, vol. 13, no. 6, pp. 895–904, Jun. 2018, ISSN: 1861-6429.

[5] C. Hennersperger, B. Fuerst, S. Virga, O. Zettinig, B. Frisch, T. Neff, and N. Navab, “Towards mri-based autonomous robotic us acquisitions: A first feasibility study,” IEEE transactions on medical imaging, vol. 36, no. 2, pp. 538–548, 2016.

[6] M. Tirindelli, M. Victorova, J. Esteban, S. T. Kim, D. Navarro-Alarcon, Y. P. Zheng, and N. Navab, Force-ultrasound fusion: Bringing spine robotic-us to the next ”level”, 2020. arXiv: 2002. 11404 [eess.IV].