SSRG International Journal of Computer Science and Engineering
ISSN: 2348–8387 / https://doi.org/10.14445/23488387/IJCSE-V11I12P102

Volume 11 Issue 12, 7-15, December 2024

Original Article

Optimizing Initial Guesses for Nonlinear System Solvers Using Machine Learning: A Comparative Study of Classification Algorithms

Iven Aabaah[1], Japheth Kodua Wiredu[2], Bakaweri Emmanuel Batowise[3]

[1]*Department of Information Systems & Technology, C. K. Tedam University of Technology & Applied Sciences, Navrongo, Ghana.*
[2]*Department of Computer Science, Regentropfen University College, Upper East, Ghana.*
[3]*Department of Cyber Security and Computer Engineering Technology, C. K. Tedam University of Technology & Applied Sciences, Navrongo, Ghana.*

[2]*Corresponding Author : wiredujapheth130@gmail.com*

Received: 11 October 2024 Revised: 16 November 2024 Accepted: 10 December 2024 Published: 29 December 2024

Abstract - *This paper focuses on the problem of improving initial guesses provided to solvers of nonlinear systems in terms of enhancing both convergence efficiency and reliability. A novel approach for constructing confidence models of initial guesses is proposed based on a Logistic Regression, Support Vector Machines (SVM), Random Forests, and K-Nearest Neighbors (KNN) classification schemes. Experimental evaluation across diverse nonlinear systems highlights Random Forests as the most effective model with an average accuracy of 81.69%, average precision – of 83.23%, average recall – of 82.16%, average F1 score of 82.69% and the highest AUC score equal to 0.90. Backed up by broad evaluation metrics, the above research inquiries mark the ideal potential of machine learning to revolutionize data processing by increasing solver adaptability, enhancing convergence patterns and economizing computations in scientific and engineering modalities.*

Keywords - *Machine Learning algorithms, Nonlinear system solvers, Data pre-processing, Model Evaluation, Predictive modeling.*

1. Introduction

The resolution of nonlinear systems of equations is a fundamental problem encountered across a broad spectrum of scientific and engineering disciplines, including physics, economics, biology, and control systems [1][2]. Nonlinear systems often lack closed-form solutions, necessitating the use of iterative numerical methods. The efficiency and success of these iterative methods, such as Newton-Raphson or Levenberg-Marquardt, heavily depend on the quality of the initial guesses provided [3]. Poor initial guesses can lead to slow convergence, convergence to incorrect solutions, or even divergence of the algorithm [4].

For instance, in power grid stability analysis, poorly chosen initial guesses in power flow equations can lead to extended computation times or complete failure to converge, potentially causing delays in real-time system monitoring [5]. Similarly, in large-scale chemical equilibrium calculations, inappropriate starting points may result in convergence to metastable states, undermining the reliability of the simulations [6]. These examples highlight that despite significant advances in numerical methods, the process of selecting effective initial guesses remains largely heuristic, highly problem-specific, and prone to inefficiencies.

Traditional approaches for selecting initial guesses rely on heuristic methods, domain-specific knowledge, or random sampling within predefined ranges [7]. While these methods may perform adequately for simple systems, they often fail in complex, high-dimensional, or highly nonlinear systems where multiple solution branches exist [8]. Furthermore, these approaches lack adaptability across different problem domains, limiting their effectiveness and scalability. As a result, poorly initialized solvers may require excessive computational resources or fail to produce a valid solution. This ongoing challenge highlights a significant research gap: there is no universally reliable, data-driven method for efficiently predicting high-quality initial guesses across diverse nonlinear systems.

Recent advances in machine learning have opened new avenues for addressing this challenge [9]. By leveraging large datasets and powerful algorithms, machine learning models can identify complex patterns and relationships that are not

immediately apparent through traditional methods [10][11]. For example, a trained model can efficiently classify whether a given initial guess is likely to lead to convergence, reducing computational costs associated with repeated trial-and-error attempts [12]. This potential suggests that machine learning could revolutionize the initialization phase of nonlinear system solvers by predicting high-quality initial guesses with greater accuracy and efficiency.

Logistic Regression, Support Vector Machines (SVM), Random Forests and K-Nearest Neighbors (KNN). Each model is fitted with data created from the random set of dissimilar nonlinear systems and used as a classifier to evaluate the success or failure of the starting guesses.

These models are assessed based on their performance in improving convergence rate and cost in terms of time, thus solving the problems associated with the conventional initialization schemes of algorithms.

The main contributions of this research can be summarized in three points. First, it presents a new methodology for the use of machine learning in the pre-solving period of nonlinear solvers, which has been shown to be beneficial for the improvement of solvers' performance. Second, it presents a comparative evaluation of various machine learning models and discusses their merits and demerits in this application.

Lastly, it demonstrates how the ideas and techniques proposed can be implemented in real life by performing many experiments on different types of nonlinear systems, showing improvement in the efficiency of the solver used and reduction in the number of computations needed.

2. Related Works

2.1. Existing Methods for Predicting Initial Guesses

Initial guesses for solving nonlinear systems of equations are key to efficient and successful convergence of solvers. Traditional methods are based on empirical rules, heuristic methods and optimization-based methods.

2.2. Empirical and Heuristic Methods

The foundation of initial guess prediction has been built on empirical and heuristic approaches. Domain experts often rely on empirical rules derived from past experiences and a deep understanding of the problem domain to generate initial guesses [13][14].

Common heuristic methods are trial and error, perturbation techniques and simple function approximations, which are quick and easy to implement [15]. However, these methods typically lack the precision and adaptability necessary for handling complex, high-dimensional nonlinear systems.

2.3. Optimization-Based Methods

Optimization techniques provide a more structured approach to generating initial guesses. Gradient-based methods, which iteratively refine initial guesses using the gradient of the objective function, are commonly used but can be computationally expensive and require close-to-optimal initial estimates to avoid converging to local minima [16][17]. Methods of global optimization, such as genetic algorithms, simulated annealing, and particle swarm optimization, are more resilient due to their comprehensive exploration of the solution space. Nevertheless, these techniques are often slower and demand significant computational resources compared to heuristic methods [18][19].

2.4. Machine Learning Approaches

The integration of machine learning into nonlinear system solving has created new opportunities to enhance the accuracy and efficiency of initial guess predictions. Various machine learning methods have been investigated, each offering distinct advantages and challenges.

2.4.1. Supervised Learning

Supervised learning has been highly encouraged when predicting initial guesses. For example, the regression models that include linear and support vector regression make some initial guesses using historical data. These models can be of great help, but since they make some linear assumptions, their utility may be limited when tackling highly nonlinear systems [16]. However, as already mentioned, neural networks, and specifically deep learning models, have demonstrated remarkable effectiveness in capturing the complex links between the input parameters and the optimal starting point. The ability of deep learning models to be trained on vast amounts of data and to adapt to new unseen problems makes them great for initial guess predictions [20][21]. Some traditional techniques have been used.

On the other hand, random initialization, grid search, and genetic algorithms have been used in the past to estimate initial guesses. Random selection is easy to perform, but for high-dimensional or complicated systems, it is not very reliable. There are sets of parameter guesses that grids systematically search through a vast array of possibilities, but they have scaling troubles. More complex techniques, such as genetic algorithms, are very computationally intensive and generally slow to reach an optimum. All these drawbacks have led to a movement towards machine learning techniques that are more efficient and adaptable.

2.4.2. Reinforcement Learning

This paper considers reinforcement learning, which is also a relatively new video game. In this approach, an agent using policy-based methods is trained through many trials and errors to attempt and maximize an initial guess strategy by making it optimize according to the rewards expected

from the performance of the solver. Even though these algorithms are able to progressively better their guesses, they still have a very long learning curve and high computing costs. The most appealing capability of reinforcement learning, in comparison with pre-specified strategies (or genetic algorithms) that make use of grid search and many other methods, is that it allows for the incorporation of feedback from the solver into its guessing procedures. Still, issues such as the need for considerable computation and uncertainty of convergence are serious problems. [22][23].

2.4.3. Hybrid Approaches

It has been done in previous studies to improve the prediction performance of prediction engines via hybrid methods, which utilize computational intelligence approaches augmented with relevant domain knowledge. These techniques, by including physical intuition or simplified models in the learning phase, can direct machine learning models to more appropriate starting points [24]. The integration of past experience with advanced algorithms is expected to significantly enhance the accuracy and reliability of algorithms designed to improve initial guesses. From the statistical tests, we conclude that hybrid strategies usually outperform normal heuristics and single stand alone machine learning models by balancing computational cost with prediction accuracy [25].

2.5. Key Research Gaps and Challenges

There are still further issues and gaps in research, such as challenges, as machine learning is still working on adding more value to estimating first guesses.

2.5.1. Generalizability

One of the biggest problems in developing machine learning algorithms is their transferability, particularly generalizing across parameters such as different nonlinear systems or varying problem scales.

This is very limiting; being able to build models that can fit different types of problems is a major challenge in this area. It is essential to highlight that comparison to classical approaches has shown that machine learning techniques stand out in some areas of application but struggle when applied to new and unfamiliar problems [26][27].

2.5.2. Adaptive Methods

Adaptive techniques require less integration and can improve an initial guess without human intervention based on the solver's feedback and the specific features of the problem at hand. Such algorithms are critical in the efficient and robust solution of nonlinear systems.

Within reinforcement algorithms, this has been more prominent, especially due to the capabilities that allow several plans to be modified according to what the solver returns, unlike heuristics that are almost always constant [28].

2.5.3. Uncertainty Quantification

Controlling the uncertainty in first guesses is a fundamental practice that should also guarantee robust solver performance, especially in unresolved or uncertain cases. Thus, the creation of methods that can adequately estimate uncertainty and treat it improves the quality of the predicted first guess [29]. One of the reasons why traditional heuristic methods are successful is the lack of applied techniques to methodologically cope with uncertainty. In contrast, state-of-the-art machine learning techniques, especially hybrid models, provide a better approach to measuring and mitigating these uncertainties.

2.5.4. Visualization Tools

The number of tools for visualization in high-dimensional space is rather small, which affects the comprehension and further improvement of the machine learning models. There is a need for better tools that visualize the relationship between initial guesses and the solution of the problems in high-dimensional spheres [30] in order to optimize and make the relationship clearer. This deficiency is not only a machine learning feature. Conventional methods also have difficulties with high-dimensional visualization.

2.6. Summary

The application of machine learning techniques for the prediction of initial guesses for nonlinear systems is new to computational numerical analysis. This promises to increase sophisticated future developments because it is well known that the limitations in this area have much space to be exploited by researchers and further enhance the efficiency and reliability of the solutions of nonlinear systems in a variety of scientific and engineering applications. A number of comparative studies with classical heuristic methods have shown that a key benefit of utilizing machine learning is the very high accuracy, speed of computation, and reliability of the results that are obtained. It can be confidently stated that as machine learning models and techniques continue to evolve, the field of nonlinear system solving will undergo significant transformation, providing solutions that were once difficult to envision yet are now precise and highly effective.

3. Methodology

3.1. Overview

This paper examines the effectiveness and precision of various machine learning algorithms in generating initial guesses for nonlinear solvers. The process comprises various phases, including data generation, data pre-processing, data splitting into training and test sets, model training, and model evaluation. It aims to find the most accurate and reliable algorithm that classifies the data, allowing the solver to converge more often. Furthermore, other classical heuristic methods, such as grid search and genetic algorithms, have also been implemented.

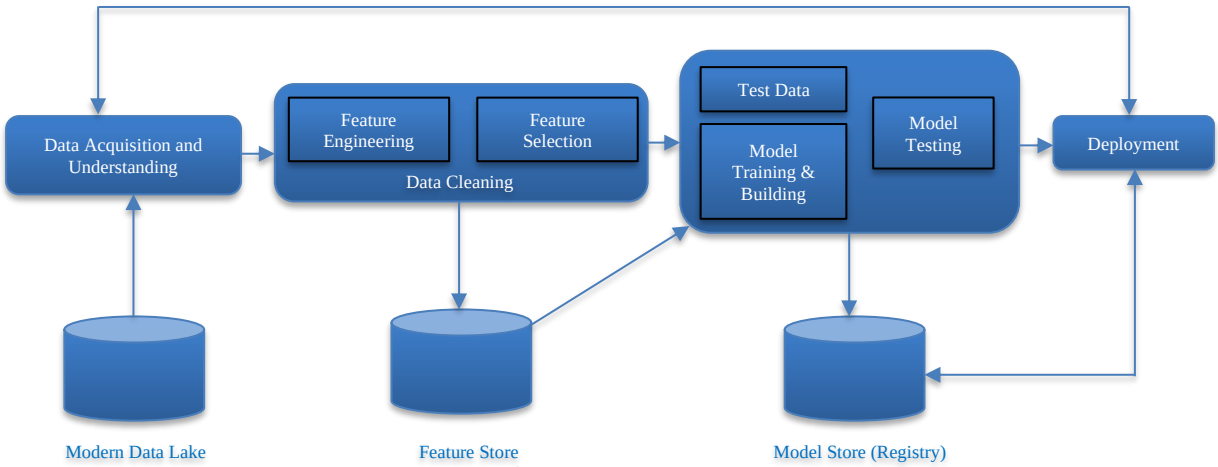

Fig. 1 Phases of the Model Development Cycle [31]

3.2. Data Generation

3.2.1. Nonlinear Systems of Equations

Three distinct systems of nonlinear equations, denoted as Problems 1, 2, and 3, were employed to generate training data. These problems represent varied mathematical structures designed to test the robustness of the machine learning models [32]. The systems are defined as follows:

Problem 1:

$$e^{x_1} + \sin(x_2) - x_3^2 + \cos(x_4) + x_5 = 2 \quad (1)$$
$$x_1 + \ln(x_2) + x_3 + \tan(x_4) - x_5^2 = 1 \quad (2)$$
$$x_1^2 - e^{x_2} + x_3 + \sin(x_4) + \cos(x_5) = 3 \quad (3)$$
$$\cos(x_1) - x_2 + x_3^2 + \ln(x_4) + \tan(x_5) = 0 \quad (4)$$
$$\tan(x_1) + x_2 + \sin(x_4) + \cos(x_5) - e^{x_5} = 1 \quad (5)$$

Problem 2:

$$\ln(x_1) + \sin(x_2) - x_3^2 + \cos(x_4) - x_5 = 1.5 \quad (1)$$
$$x_1^2 + \ln(x_2) + x_3 + \tan(x_4) - x_5^2 = 2 \quad (2)$$
$$e^{x_1} - e^{x_2} + x_3 + \sin(x_4) + \cos(x_5) = 3 \quad (3)$$
$$\cos(x_1) - x_2 + x_3^2 + \ln(x_4) + \tan(x_5) = 0 \quad (4)$$
$$\tan(x_1) - x_2 + \sin(x_3) + \cos(x_4) + e^{x_5} = 1 \quad (5)$$

Problem 3:

$$e^{x_1} + \sin(x_2) - x_3^2 + \cos(x_4) - x_5 = 2 \quad (1)$$
$$x_1 + x_2^2 + x_3 + \tan(x_4) - \ln(x_5) = 1 \quad (2)$$
$$x_1^2 - e^{x_2} + x_3 + \sin(x_4) + \cos(x_5) = 3 \quad (3)$$
$$\cos(x_1) - x_2 + x_3^2 + \ln(x_4) + \tan(x_5) = 0 \quad (4)$$
$$\tan(x_1) - x_2 + \sin(x_3) + \cos(x_4) + e^{x_5} = 1 \quad (5)$$

3.2.2. Data Generation Process

The dataset used in this study comprises initial guesses and corresponding convergence outcomes from simulations across multiple nonlinear systems, including power systems, chemical equilibria, and biological networks. The dataset includes over 10,000 samples, ensuring a diverse representation of initial guess scenarios. Each sample consists of input features representing key parameters of the system and labels indicating convergence success or failure. Dataset diversity was ensured by incorporating scenarios from varied system dynamics, including cases with strong nonlinearity, multiple equilibrium points, and differing parameter sensitivities.

- Sample Generation: Random samples for x_1 through x_5 were drawn from the range [−10, 10].
- Solver Application: A numerical solver, such as Newton-Raphson, was employed to determine convergence outcomes.
- Labeling: Data was labeled as 1 for successful convergence and 0 for failure, based on solver results.

3.3. Data Pre-processing

Data pre-processing ensured the dataset's suitability for machine learning:

1. Cleaning: Missing values were either imputed or removed.
2. Normalization: All variables were scaled to the [0, 1] range to enhance model performance.
3. Feature and Target Variables:
   - Features (X): x_1, x_2 , x_3, x_4, x_5 (independent variables).
   - Target (y): Binary labels indicating solver success (1) or failure (0).

3.4. Model Training and Evaluation

3.4.1. Model Selection and Training

The following models were selected for this study: Logistic Regression (LR), Support Vector Machine (SVM), Random Forest Classifier (RFC), and K-Nearest Neighbors (KNN). These models were easily selected to interpret; on SVM, the following excellent properties: in LR, high is dimensions and the baseline can linear handle model more that complex can decision produce boundaries; results from RFC which reduces with overfitting well-separated by data ensemble sets. Learning: Due to KNN is a suitable fact for

that nonlinear, each decision of boundary the and models works has well its strengths and weaknesses. The following hyperparameters for each algorithm were tuned using Grid Search with 5-fold cross-validation in order to enhance performance. In the case of LR, the changes have been made in the cost coefficient (C) and solver, namely liblinear or lbfgs. For SVM, the kernel function chosen was linear, and RBF with parameter C was tuned.

The tree parameters for the RFC forest included the is number of n_estimators of the maximum depth of a tree that is max_depth and the splitting criterion, which can either be gini or entropy. For KNN, the optimization was done on the number of neighbors (n_neighbors) and the distance metric, which could either be Euclidean or Manhattan. The data set was split into a training set containing 80% of the data and a testing set with 20% of the data. To enhance result reliability, prevent overfitting, and obtain a more accurate performance estimate on various data partitions, 5-fold cross-validation was utilized.

3.4.2. Evaluation Metrics

Model performance was assessed using:

- Accuracy: Proportion of correct predictions.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions}$$

- Confusion Matrix: Provides insight into performance in terms of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN).
- F1 Score: Harmonic mean of precision and recall.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

- Confusion Matrix: To evaluate true/false positives and negatives.
- ROC-AUC: Area under the Receiver Operating Characteristic curve to measure class distinction.

3.5. Visualization and Comparative Analysis

- Visualization: Confusion matrices (heatmaps) and ROC curves were plotted for all models to provide a visual performance comparison.
- Comparative Analysis: Models were compared based on accuracy, F1 score, and ROC-AUC. The results were interpreted to determine the algorithm yielding the most reliable initial guesses.

3.6. Discussion

The findings were discussed with a focus on their implications for improving the efficiency of nonlinear system solvers, including practical recommendations for integrating predictive models into solver workflows.

4. Experimental Results

4.1. Overview

The experiments demonstrated the capability of machine learning algorithms to predict successful initial guesses for solving nonlinear systems of equations. Three nonlinear problems were used to generate labeled datasets. The performance of four machine-learning models was assessed using accuracy, precision, recall, F1 score, and ROC-AUC metrics.

4.2. Data Generation and Pre-processing

4.2.1. Data Generation Results

Data was generated by sampling variables (x_1, x_2 , x_3, x_4, x_5) randomly within the range [-10, 10]. Using the Newton-Raphson solver:

- Total samples (N): 1,471 (741 successes, 730 failures).
- Success rate (convergence): 50.37%.
- Failure rate (non-convergence): 49.63%.

4.2.2. Pre-processing Outcomes

- Missing Values: Negligible (<3%), resolved through imputation.
- Normalization: Features scaled to [0, 1] to ensure uniformity.
- Data Split: The dataset is divided into training (80%) and testing (20%) subsets.

Table 1. Summarizes the evaluation metrics for all models

Model	Accuracy	Precision	Recall	F1 Score	ROC-AUC
Logistic Regression	0.789831	0.784431	0.834395	0.808642	0.87
Support Vector Machine	0.813559	0.807229	0.853503	0.829721	0.89
Random Forest	0.816949	0.832258	0.821656	0.826923	0.90
K-Nearest Neighbors	0.793220	0.789157	0.834395	0.811146	0.87

4.3. Model Training and Evaluation

Table 1 presents a comprehensive overview of the performance of each machine-learning model, utilizing essential evaluation metrics such as accuracy, precision, recall, F1 score, and ROC-AUC.

These metrics serve as benchmarks for assessing and comparing the models, offering valuable insights into their effectiveness in predicting solver convergence within nonlinear systems of equations.

4.4. Visualization and Confusion Matrix Analysis

Confusion matrices for each model are shown in the Figures below. These matrices illustrate the distribution of:

- True Positives (TP): Instances where the model accurately predicted successful solver convergence.
- True Negatives (TN): Cases where failures were correctly identified.
- False Positives (FP): Situations where the model incorrectly predicted success despite solver failure.
- False Negatives (FN): Incorrectly predicted failure when the solver succeeded.

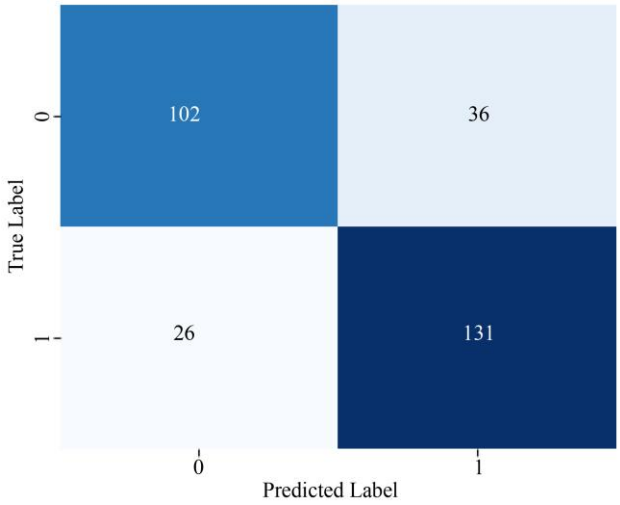

Fig. 2 Logistic regression confusion matrix

Figure 2 reveals that the model successfully classified 74.4% of non-convergence cases as True Negatives and 83.5% of convergence cases as True Positives. However, it also misclassified 25.6% of non-convergence cases as convergence (False Positives) and 6.5% of convergence cases as non-convergence (False Negatives).

Figure 3 shows that it correctly predicted 76.8% of non-convergence cases (True Negatives) and 85.4% of convergence cases (True Positives). However, 23.2% of non-convergence cases were misclassified as convergence (False Positives), while 14.6% of convergence cases were incorrectly predicted as non-convergence (False Negatives).

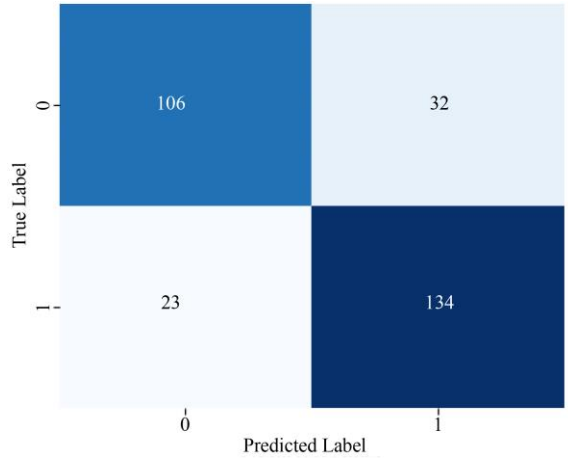

Fig. 3 Support vector confusion matrix

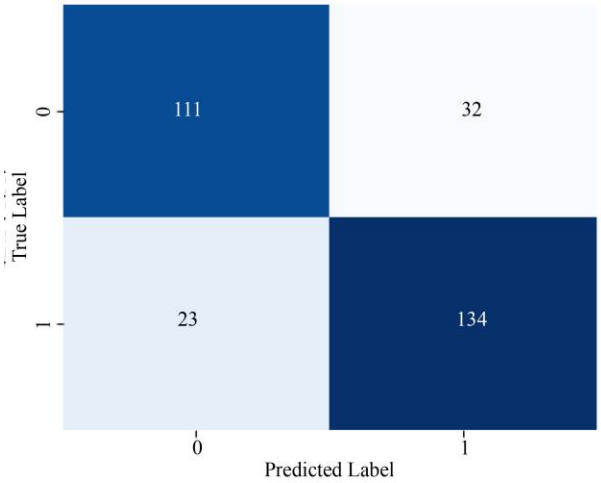

Fig. 4 Random forest confusion matrix

Figure 4 illustrates that the model accurately identified 82.6% of non-convergence cases (True Negatives) and 77.7% of convergence cases (True Positives). However, it misclassified 17.4% of non-convergence cases as convergence (False Positives) and 22.3% of convergence cases as non-convergence (False Negatives).

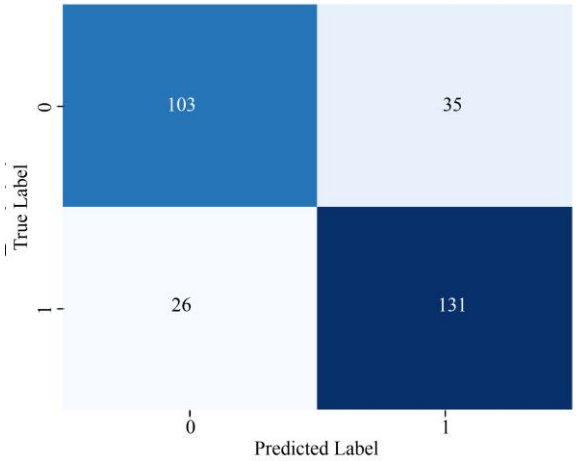

Fig. 5 K-Nearest Neighbor Confusion Matrix

Figure 5 shows the following results in percentages: 74.6% of the actual negatives were correctly predicted as negatives and 83.5% of the actual positives were correctly predicted as positives. The classifier, however, showed errors in classification: 25.4% of true negative cases were mistakenly classified as positives, and 16.5% of true positive cases were mislabeled as negatives. The confusion matrix provides a visual summary of the classifier's performance, illustrating the percentages of correct and incorrect predictions.

4.4.1. ROC Curves

In their research [33], they describe the results obtained from four classifiers, i.e., LR, SVM, RFC, and KNN, based on Receiver Operating Characteristic (ROC) curves, as shown in Figure 6. The area under the curve (AUC) values for these classifiers are as follows: logistic regression estimates curves' area of AUC=0.87, for SVM AUC=0.89, for random forest AUC=0.90 and KNN 0.87, respectively. The horizontal axis of the Figures presents various threshold values, while the vertical axes show the sensitivity (TPR) and (1-specificity) of the respective curves as plotted on the ROC Curve. The higher the value of the AUC, the greater the capability of the model in distinguishing, classifying and identifying positive cases from negative cases. In this case, it can be said that the Random Forest classifier is the most competent since it has the greatest AUC compared to the other three models.

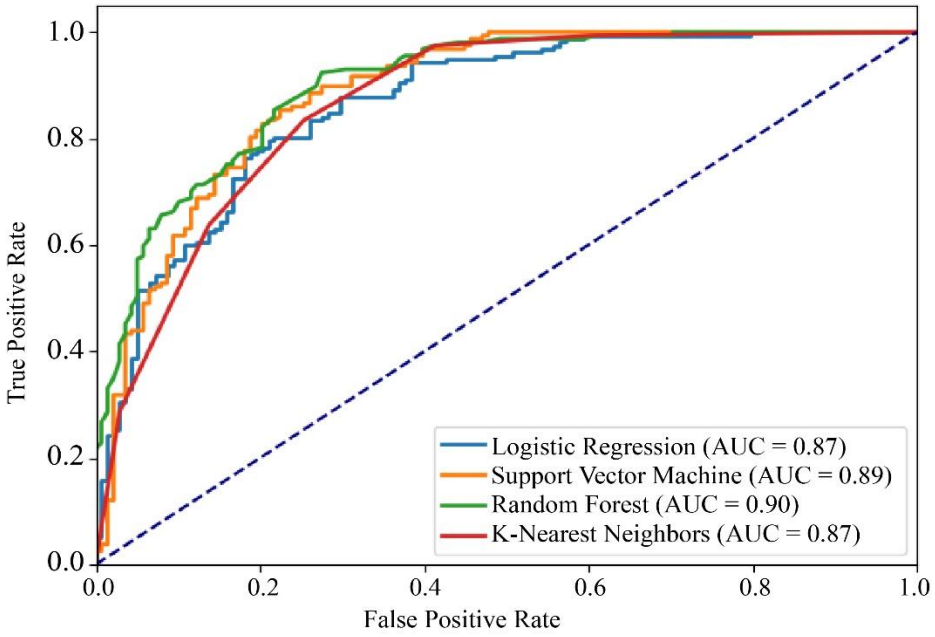

Fig. 6 Receiver Operating Characteristic (ROU) Curves

4.5. Comparative Analysis

Performance comparisons revealed:

- Random Forest Classifier: It is the best model, for it was used to get the best accuracy, F1 score, and AUC.
- Logistic Regression: it is an interesting and interpretable model with good performance in both precision and recall.
- Logistic Regression: it is an interesting and interpretable model with good performance in both precision and recall.

4.6. Discussion

The experimental results indicate that Random Forest yields the most accurate result in predicting the successful first guesses in applications of the nonlinear solver. With that said, there are other more interpretable models, such as the logistic regression, which yield decent results. Future work may involve the investigation of ensemble methods or tuning of hyper-parameters to improve the results further.

5. Conclusion

This paper demonstrated the potential of machine learning techniques in improving the efficiency of nonlinear system solvers concerning initial guesses. Among all the models tested, Random Forest emerged as the most accurate and effective, making it the recommended choice for practical applications. Other models, including ensemble models and hybrid models that integrate knowledge from some domain, could be future research directions. These advancements could further enhance solver efficiency and broaden the applicability of machine learning in computational science.

Acknowledgements

The authors wish to express their gratitude for the support given by Regentropfen University College (RUC) and C. K. Tedam University of Technology and Applied Sciences (CKT-UTAS) throughout the research and

development of this manuscript. Their contributions to creating a conducive environment for this study are deeply appreciated.

Author 2, the principal contributor and lead researcher, conceptualized the study and wrote the manuscript. Author 1 and Author 3 contributed by providing supplementary analysis and critical feedback on the manuscript.

References

[1] Richard H. Enns, and George C. McGuire, *Nonlinear Physics with Mathematica for Scientists and Engineers*, Springer Science & Business Media, pp. 1-695, 2012. [Google Scholar] [Publisher Link]

[2] Aarav Chhabra, "Applied Mathematics: Building Theory and Practice," *International Journal for Research Publication and Seminar*, vol. 15, no. 2, pp. 137-149, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[3] Nasima Sawlat, Yalda Qani, and Naqibullah Sadeqi, "Numerical and Symbolic Analysis for Mathematical Problem-Solving with Maple," *Journal of Natural Science Review*, vol. 2, no. 3, pp. 29-46, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[4] Dipjyoti Nath et al., "Application of Machine Learning and Deep Learning in Finite Element Analysis: A Comprehensive Review," *Archives of Computational Methods in Engineering*, vol. 31, pp. 2945-2984, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[5] Zeynab Kaseb et al., "Power Flow Analysis using Quantum and Digital Annealers: A Discrete Combinatorial Optimization Approach," *Scientific Reports*, vol. 14, no. 1, pp. 1-14, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[6] Sergio Decherchi, and Andrea Cavalli, "Thermodynamics and Kinetics of Drug-Target Binding by Molecular Simulation," *Chemical Reviews*, vol. 120, no. 23, pp. 12788-12833, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[7] S. Passone, P.W.H. Chung, and V. Nassehi, "Incorporating Domain-Specific Knowledge into a Genetic Algorithm to Implement Case-Based Reasoning Adaptation," *Knowledge-Based Systems*, vol. 19, no. 3, pp. 192-201, 2006. [CrossRef] [Google Scholar] [Publisher Link]

[8] Ali Al-Aradi et al., "Solving Nonlinear and High-Dimensional Partial Differential Equations Via Deep Learning," *arXiv*, pp. 1-76, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[9] Hayssam Dahrouj et al., "An Overview of Machine Learning-Based Techniques for Solving Optimization Problems in Communications and Signal Processing," *IEEE Access*, vol. 9, pp. 74908-74938, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10] Maryam M Najafabadi et al., "Deep Learning Applications and Challenges in Big Data Analytics," *Journal of Big Data*, vol. 2, pp. 1-21, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[11] William Grant Hatcher, and Wei Yu, "A Survey of Deep Learning: Platforms, Applications and Emerging Research Trends," *IEEE Access*, vol. 6, pp. 24411-24432, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[12] Olexandr Isayev et al., "Universal Fragment Descriptors for Predicting Properties of Inorganic Crystals," *Nature Communications*, vol. 8, no. 1, pp. 1-12, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[13] Jonathan Calver, Jienan Yao, and Wayne Enright, "Using Shooting Approaches to Generate Initial Guesses for ODE Parameter Estimation," *Recent Developments in Mathematical, Statistical and Computational Sciences*, Waterloo, Canada, pp. 267-276, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[14] Federico Toso, and Christie Maddock, "Initial Guess Generation Strategies for Spaceplane Trajectory Optimization," *Transactions of the Japan Society for Aeronautical and Space Sciences*, pp. 1-8, 2017. [Google Scholar] [Publisher Link]

[15] Haksun Li, "Heuristics," *Numerical Methods Using Java*, pp. 625-654, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[16] Ilias Theodorakos, and Michael Skipper Andersen, "Influence of the Initial Guess on the Estimation of Knee Ligament Parameters via Optimization Procedures," *Bioengineering*, vol. 11, no. 12, pp. 1-12, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] Manouchehr Sanei et al., "An Innovative Scheme to Make an Initial Guess for Iterative Optimization Methods to Calibrate Material Parameters of Strain-Hardening Elastoplastic Models," *Rock Mechanics and Rock Engineering*, vol. 55, pp. 399-421, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[18] Shitharth Selvarajan, "A Comprehensive Study on Modern Optimization Techniques for Engineering Applications," *Artificial Intelligence Review*, vol. 57, no. 8, pp. 1-52, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[19] Hajime Igarashi, *Topology Optimization and AI-Based Design of Power Electronic and Electrical Devices*, Academic Press, pp. 127-179, 2024. [Google Scholar] [Publisher Link]

[20] Ramamani Tripathy et al., "Early Diabetes Prediction Using Supervised Machine Learning Techniques," *International Conference on Advances in Computing Research on Science Engineering and Technology*, Indore, India, pp. 1-4, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[21] Xiang Huang, "Predictive Models: Regression, Decision Trees, and Clustering," *Applied and Computational Engineering*, vol. 79, pp. 124-133, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[22] K. Kalaiselvi et al., "Innovations in Natural Language Processing through Enhanced Linguistic Model Accuracy and Efficiency Using Advanced Reinforcement Learning Techniques," *Second International Conference on Advances in Information Technology*, Chikkamagaluru, Karnataka, India, pp. 1-5, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[23] Benjamin Ribba, "Reinforcement Learning as an Innovative Model-Based Approach: Examples from Precision Dosing, Digital Health and Computational Psychiatry," *Frontiers in Pharmacology*, vol. 13, pp. 1-7, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[24] Ilknur Gur Nalcaci et al., "Enhancing Sales Prediction Accuracy Through Incorporation of Expert Knowledge as Constraints Into the Model," *IEEE 18th International Conference on Application of Information and Communication Technologies*, Turin, Italy, pp. 1-5, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[25] A. Sheik Abdullah et al., "Enhancing Predictive Analytics Through Hybrid Machine Learning Models: A Comparative Analysis Of SVM-NB And Lasso-Ridge Techniques," *Second International Conference on Networks, Multimedia and Information Technology*, Bengaluru, India, pp. 1-10, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[26] Huaxiu Yao, "Towards Reliable Learning in the Wild: Generalization and Adaptation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 20, pp. 22683-22683, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[27] Federico Matteucci et al., "Generalizability of Experimental Studies," *arXiv*, pp. 1-16, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[28] Watchara Wongsa, Pikul Puphasuk, and Jeerayut Wetweerapong, "An Adaptive Differential Evolution Algorithm with a Bound Adjustment Strategy for Solving Nonlinear Parameter Identification Problems," *IT, Automation, Measurements in Economy and Environmental Protection*, vol. 14, no. 2, pp. 119-126, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[29] Ziyi Huang, Henry Lam, and Haofeng Zhang, "Efficient Uncertainty Quantification and Reduction for Over-Parameterized Neural Networks," *Advances in Neural Information Processing Systems*, vol. 36, 2023. [Google Scholar] [Publisher Link]

[30] Alexey Orlov et al., "From High Dimensions to Human Comprehension: Exploring Dimensionality Reduction for Chemical Space Visualization," *Theoretical and Computational Chemistry*, pp. 1-43, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[31] Jules S. Damji, and Michael Galarnyk, Considerations for Deploying Machine Learning Models in Production, Anyscale, 2021. [Online]. Available: https://www.anyscale.com/blog/considerations-for-deploying-machine-learning-models-in-production

[32] Isaac Azure et al., "AI-Enhanced Performance Evaluation of Python, MATLAB, and Scilab for Solving Nonlinear Systems of Equations: A Comparative Study Using the Broyden Method," *American Journal of Computational Mathematics*, vol. 13, no. 4, pp. 644-677, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[33] Charu C. Aggarwal, *Neural Networks and Deep Learning*, Cham: Springer, 2023. [CrossRef] [Google Scholar] [Publisher Link]