International Journal on Smart & Sustainable Intelligent Computing
Vol. 02, Iss. 01, S. No. 001, pp. 1-13, March 2025
ISSN (Online): 3048-8508

Improving Wine Quality Forecasts Using Dynamic Integral Neural Networks and Optimized Against Interference

Shyama Heshini Niranjala[1], S. B. Goyal[1]*

[1]Faculty of Information Technology, City University, Petaling Jaya, 46100, MALAYSIA
shyama@gwu.ac.lk, drsbgoyal@gmail.com*

How to cite this paper: Shyama Heshini Niranjala, S. B. Goyal, "Improving Wine Quality Forecasts Using Dynamic Integral Neural Networks and Optimized Against Interference," *International Journal on Smart & Sustainable Intelligent Computing*, Vol. 02, Iss. 01, S No. 001, pp. 1-13, March 2025.

Received: 28/11/2024
Revised: 22/12/2024
Accepted: 18/01/2025
Published: 05/02/2025

Abstract

As a very important part of the wine sector, wine quality prediction covers vital aspects of the industry that have to do with effective production possibilities and customer satisfaction. The complexity of determining wine quality is enhanced by increasing numbers of varieties of grapes, methods of fermentation, and geographical factors, which make the need for enhanced prediction models. The main conventional techniques largely involve the use of sensory assessments, which are notorious for their bias and inaccuracy while on the other hand; machine- learning techniques have turned out to be very reliable. In this paper, a Differential evolution algorithm with exponential crossover is incorporated into a Dynamic Integral Neural Network (DINN) optimized for anti-interference as a new wine quality prediction model. Substituting relevant integral components, the proposed model cancels out noise and interference, which is always witnessed in real-life data, thus boosting the capability of the model to learn with fluctuating and imprecise data for better and uniform outcomes. Stringent testing proves that the DINN model works better than other neural network structures under consideration by showing such advantages as increased dependability, higher speeds of calculation, and better resilience to unusual data values. Comparison with the classic type shows that the DINN gives a better performance in noisy environments and overall predictive accuracy. Its usage for the wine industry seems very promising as the main issue in this area is the presence of outliers, which distort the data and result from the variability of the measurement techniques and environmental conditions. The results support DINNs' applicability for revolutionizing wine quality evaluation and pave the way for an efficient DINN implementation in various real-world settings; future works could include the inclusion of IoT systems for real-time wine quality evaluation, and testing DINNs on seemingly limitless datasets of different winemaking settings.

Keywords

Dynamic Integral Neural Network, Wine quality prediction, Anti-interference optimization, Robust prediction models, Machine learning

1. Introduction

In the broad field of agriculture and economy, the wine industry is the leading sector, which generates billions of dollars every year through wine production, trading, and consumption. However, quality control in wine production is still a main issue that is difficult to resolve due to a vast of factors the weather conditions, the grape type and ripeness, the fermentation method, and the storage conditions [1]. These aspects begin to influence the overall data obtained at each phase of the winemaking process and thus present additional error and variability in the quality prediction process. In previous studies, wine quality prediction has been tackled using a few statistical models and methods in machine learning. However, these models often give unsatisfactory results due to shortcomings such as failure to handle noisy data, overfitting and they are not flexible to adapt to changes that are common in wine production environments [2].

Conventional measures of wine production also have significant fluctuations due to the following sources. As already, mentioned environmental conditions like temperature, humidity, and rainfall influence the growth of grapes and the qualitative composition of the fruits [3]. However, not only is the complexity increased by the fact that the characteristics of each type of grapes differ greatly, but also the terroir —the specific conditions affecting grape cultivation. Fermentation processes are also used to influence wine quality characteristics because fermentation involves yeast-sugar interactions that result in biochemical reactions that are different for each batch [4]. Again, even the conditions under which wine is stored and aged influence the end outcome hence pointing to even more factors that need to be explained by the predictive models.

Due to these difficulties, basic approaches encompassing linear regression and decision trees do not entirely succeed in detecting the correlation between these factors. Out of the various AI techniques that have been applied in the literature to handle large and composite data and learn the non-linear features, there are two prominent categories: the Machine learning (ML) models [5], and even the deep learning models [6]. Nevertheless, most of these models fail to handle the noisy, missing, or skewed data, which are very much real in most problems. Therefore, to overcome these limitations and to obtain reliable quality predictions in the presence of noise, current and further research requires more complex and effective models.

Specifically for wine quality prediction, this paper introduces the utilization of proposed Dynamic Integral Neural Networks (DINNs) with Anti-Interference Optimization techniques. DINNs are a kind of artificial neural network that handles dynamic data with different types of time dependency; their application in time series data like those in wine-making processes is appropriate [7]. This work will therefore propose the use of comparatively flexible DINNs in conjunction with anti-interference optimization techniques in a bid to synthesize a highly reliable predictive model that can minimize the impacts of noise and offer high accuracy under several conditions. This model is going to act as a perfect resource to help wine producers maintain and control the quality of the product by providing important information at the various stages of the creation process.

The remainder of this paper is organized as follows: Section 2 brings the works of other authors who are dedicated to the wine quality prediction model and utilization of neural networks in predictive analysis. Section 3 provides the research problem and its objectives where the wine industry is identified as requiring more solid models. Section 4 gives a more elaborate account of this approach by outlining how DINNs and anti-interference optimization are combined. Section 5 gives an experimental analysis of the proposed model with the other techniques mentioned in this paper. Last, Section 6 is the conclusion of the paper, which presents the findings of the work and sheds some light on possible new directions for further research in this field.

2. Related Work

The ability to predict wine quality has emerged as an area of concern in the last few years due to the actual requirements of the wine industry for dependable models at all the stages of wine production. In the past, three conventional approaches used in the determination of wine quality included sensory-based expert assessment, chemical measurements, and statistical modeling. However, these methods are less accurate and scalable and cannot be directly implemented or used for complex and dynamic production systems. In the last few decades, approaches involving machine learning especially, and artificial neural networks (ANNs) in particular have proved useful.

To determine wine quality based on physicochemical characteristics, authors used artificial neural networks [8]. They report that using feed-forward neural networks can be beneficial for constructing more accurate predictions as compared to regression models. Their approach, which focused on developing a prediction model, faced challenges with overfitting and noise in the data sets where their set contained missing values or imbalanced values. Yet another study by [9], also working on wine quality classification utilized the support vector machines (SVMs), and the results were higher classification accuracy than other models. However, the authors themselves pointed out that SVMs are disposed to noise and cannot process missing data without descending much data pre-processing.

One of the main drawbacks of other predictive models that have been implemented before is the fact that the models do not consider the noise that is characteristic of real-life scenarios. Since a large number of changing factors and relationships between them characterize wine production, using models that cannot learn the data dynamics is not very useful. To this challenge, several researchers have advocated for better and more flexible predictions using deep learning approaches and more specifically the Recurrent Neural Networks (RNNs) [10] and the Convolutional Neural Networks (CNNs) [11]. For example, [12] used RNN to capture the temporal dependency of wine production data and [13] employed CNNs to capture the chemical characteristics of the Wines to give a better forecast on quality.

Despite that, deep learning models are promising; noise resilience is still an important problem. Contemporary studies by [14] developed new noise-resistant machine learning algorithms that apply to industries, yet the usage of these algorithms in the wine industry is not researched enough. Similarly, [15] has reviewed the integration of optimization techniques with machine learning for better robustness, and anti-interference optimization can increase performance.

Although there has been significant work done on using machine learning for wine quality prediction, little effort has been made to incorporate robust optimization methods within the deep learning space. Such a gap means further research in incorporating Dynamic Integral Neural Networks (DINNs) with anti-interference optimization strategies to design a model that is more accurate and versatile for wine quality prediction. This paper seeks to address this deficit by proposing a new framework that increases accuracy in new predictions while handling noise problems in existent datasets.

2.1 Problem Statement & Research Objectives

The wine manufacturing sector has its fair share of issues in achieving the development and stability of product quality since the numerous factors of wine's quality depend on the environmental factors of vineyard, grape quality, the fermentation processes, and aging conditions. Accurate modeling of such relationships necessitates the use of appropriate and realistic relationships, which are normally nonlinear or dynamic. Moreover, real-world data especially in the wine industry is also full of noise, and missing data and can be very imbalanced hence making predictions very challenging.

To provide more clarity let me explain, that existing predictive models in many cases do not meet the noise and variability challenges of wine production data. The use of basic forms of statistics such as regression, and simple artificial neural networks is not efficient in terms of capturing the intricacy of relations within variables. However, models fail to generalize well and overfit to the training data resulting in poor performance when applied to unseen data. Such limitations make us aware of the requirement for more effective models, especially the ones capable of operating and offering accurate and more resilient results under noisy conditions within complex, real-life settings.

To overcome these challenges in the generation of noise-resilient predictive framework, this research focuses on optimizing the deep learning model using DINNs with Anti-Interference Optimization. The primary objectives of this research are as follows:

- Develop a predictive framework based on DINNs that can capture the dynamic relationships between multiple factors affecting wine quality.

- Integrate anti-interference optimization techniques to improve the model's robustness and resilience to noisy, incomplete, and imbalanced data.
- Evaluate the proposed model's performance in comparison with traditional and state-of-the-art machine learning models, demonstrating its superiority in terms of accuracy, adaptability, and noise resilience.
- Provide a reliable, scalable tool for wine producers to predict wine quality accurately and consistently across a wide range of production conditions.

With these objectives, this study's research will further the knowledge of prediction models in the wine sector and offer a useful resource to guarantee quality and facilitate decision-making throughout the production process.

3. Research Methodology

The overall framework of the research comprises the following steps: The creation of a Dynamic Integral Neural Network (DINN) model for the prediction of wine quality, with the addition of anti-interference optimization functionality. The methodology is divided into several key steps: data access/acquisition and preparation, model development, choice of the optimization algorithm, and model assessment.

3.1 Data Collection and Preprocessing

The first step of the development of the proposed research methodology is to accrue vast data in the wine-making process. These datasets typically include fixed acidity (Concentration of non-volatile acids like tartaric acid), volatile acidity (Acetic acid content, which can impart vinegar-like flavors if excessive), citric acid (Adds freshness and a slight tanginess to the wine), Residual Sugar(Amount of sugar remaining after fermentation), Chlorides (Salt content in the wine), Free Sulphur Dioxide (Protects wine from oxidation and microbial growth), Total Sulphur Dioxide (Sum of free and bound sulphur dioxide), Density (Related to the sugar and alcohol content in the wine), pH (Measures acidity; affects wine's taste and preservation), Alcohol (Ethanol content, a key determinant of flavor and quality). Nevertheless, such data famously comes with some gaps that need to be managed at the preprocessing stage, outliers, and noise inclusive.

To ensure the data is suitable for model training, we employ the following preprocessing steps [16]:

- **Data Cleaning**: Some of the methods include the Z-score test, which looks at the variable's denseness on a given curve, and the Interquartile Range (IQR), which is a statistical measure that helps to detect both lower and upper outliers.

- **Normalization**: These attributes imply that the data is first normalized to a standard scale so that none of the features dominates in the model. The normalization equation for each feature x_i is given by Eq.(1):

$$x_i^{norm} = \frac{x_i - \mu_i}{\sigma_i} \tag{1}$$

Where x_i: the original value of the feature

μ_i: the mean of feature i

σ_i : the standard deviation of feature i

- **Imputation**: Statistical methods, i.e. the mean imputation is used to impute the Missing data points by using Eq. (2):

$$x_i^{imp} = \mu_i \tag{2}$$

Where x_i^{imp} : the imputed value for feature i

μ_i : the mean of feature i.

3.2 Model Architecture Design

Conceptually, toward the center of the methodology is the formulation of the Dynamic Integral Neural Network (DINN), a Recurrent Neural Network (RNN) tailored to deal with time series data and dynamic inputs. This is especially useful whenever the wine quality greatly depends on time varying parameters such as rate of fermentation, characteristics of grapes and other conditions prevailing at the time of wine aging.

Recurrent Neural Network (RNN) Equations: The basic equation for hidden state h_t updating the at time step t is given by Eq. (3) [17]:

$$h_t = f(W_h . h_{t-1} + W_x . x_t + b_h)$$

(3)

Where h_t : hidden state at time step t, h_{t-1} : hidden state at the previous time step t−1, x_t is the input at time step t, W_h is the weight matrix for the hidden state, W_x is the weight matrix for the input data, b_h is the bias term, f is the activation function, typically a hyperbolic tangent (tanh) or ReLU

These recurrent connections help the network learn temporal dependencies so that the network can learn about the patterns that are changing with the input data in the time space.

Output Layer Equation

The output of the DINN model, representing the predicted wine quality at time step t, is given by Eq.(4):

$$y_t = W_y . h_t + b_y$$

(4)

Where y_t is the predicted output (wine quality)

W_y is the weight matrix for the output layer

h_t is the hidden state at time step t

b_y is the output bias term

The predicted output y_t can either be a continuous value (regression) or a class label (classification) depending on the nature of the wine quality prediction problem.

3.3 Anti-Interference Optimization

To address the challenge of noisy data and ensure robustness, anti-interference optimization techniques are incorporated. The optimization algorithm aims to reduce the impact of noisy or misleading data by assigning different weights to data points based on their reliability [18].

Weighted Loss Function

The loss function is modified to incorporate weights, where more reliable data points are given higher weights. The weighted loss function for the model is given by Eq. (5):

$$L_{weighted} = \sum_{i=1}^{N} w_i . L(y_i, \hat{y}_i) \quad (5)$$

Where w_i is the weight for the i^{th} sample, indicating its reliability. For noisy data points, w_i is smaller, and for reliable data points, w_i is larger. $L(y_i, \hat{y}_i)$ is the loss function for the i^{th} data point, where y_i is the true label and $\hat{y}_i$ is the predicted label. N is the total number of data points in the dataset.

The weight w_i can be computed based on a confidence score or distance from a threshold value, where data points with a confidence score above a certain threshold are given higher weight.

Optimization Objective

The optimization process aims to minimize the weighted loss function $L_{weighted}$ during training. The model is trained by adjusting the parameters W_h, W_x, W_y, and b_h, b_y through gradient descent or other optimization algorithms.

4. Model Training and Evaluation

Once the model architecture is designed and the optimization process is integrated, the model is trained using supervised learning techniques [19]. The objective is to minimize the weighted loss function and learn the best parameters for the network.

Objective Function

The training process can be formalized as minimizing the following objective function £given by Eq. (6) :

$$£ = \sum_{t=1}^{T} L_{weighted}(y_i, \hat{y}_i) \tag{6}$$

Where T is the number of training epochs

$L_{weighted}$ is the weighted loss for each training sample

Performance Metrics

To evaluate the performance of the model, the following metrics are used [20]:

- **Accuracy**: The percentage of correctly predicted wine quality scores is given as Eq. (7).

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} \mathbb{I}(y_i = \hat{y}_i)}{N} \tag{7}$$

Where $\mathbb{I}(y_i = \hat{y}_i)$ is the indicator function that equals 1 if $y_i = \hat{y}_i$ and 0 otherwise.

- **Precision, Recall, and F1-Score**: These metrics are particularly useful in classification tasks, especially when dealing with imbalanced datasets. Precision and recall are calculated by using Eq.(8) and Eq.(9) as:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{8}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{9}$$

Where TP is the number of true positives

FP is the number of false positives

FN is the number of false negatives.

The **F1 score** is the harmonic mean of precision and recall as mentioned by Eq.(10):

$$F1 = 2 * \frac{\text{Precision.Recall}}{\text{Precision+ Recall}} \tag{10}$$

These metrics help evaluate how well the model generalizes to unseen data and its robustness against noisy inputs.

- **Mean Squared Error (MSE):** It measures the average squared difference between actual and predicted values by using Eq. (11). Lower values indicate better model performance.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_{actual} - y_{predicted}\right)^2 \tag{11}$$

Where MSE measures the average of the squared differences between the actual and predicted values. A lower MSE indicates better model performance. N is the total number of data points or observations used to evaluate the model. y_{actual} is the true or observed value for a specific data point. This represents the real outcome that the model is trying to predict. $y_{predicted}$ is the predicted value by the model for the same data point. This is the model's estimate of $y_{actualy}$

$(y_{actual} - y_{predicted})^2$ is the squared difference between the actual and predicted values for each sample. This emphasizes larger errors by squaring them.

- **R-Squared (R^2):** It indicates how well the model explains the variance in the data and is explained by Eq. (12). Values close to one indicate good model performance.

$$R^2 = 1 - \frac{SS_{residual}}{SS_{total}} \tag{12}$$

where $SS_{residual}$ is the sum of Squared Residuals and SS_{total} is the total sum of squares.

5. Results & Discussion

The results of the proposed model will be evaluated through a series of experiments conducted on wine production datasets. The evaluation will compare the performance of the DINN model with traditional methods such as linear regression and decision trees, as well as with more advanced machine learning models like support vector machines and deep neural networks.

The plots in Fig.1 represent the training progress of the model. The top graph displays the Root Mean Squared Error (RMSE) over iterations, showing minor fluctuations but no significant improvement as training progresses, indicating challenges in reducing prediction error. The bottom graph shows the Loss values over iterations, reflecting similar behavior with a consistent pattern and slight oscillations. Both graphs suggest limited improvement in the model's performance, potentially requiring adjustments to the training process, such as hyper parameters or data preprocessing, to enhance learning.

Fig.1 displays the training progress of the model during each epoch. The Epoch column gives the number of complete cycles through the learning database where every cycle is made of numerous Mini batches. The Iteration column displays a current mini-batch number that is incremented every 50 iterations. The Time Elapsed also shows the number of seconds that took to process the mini batches in each epoch. As the epochs increase, the time taken also increases, reflecting the model's progress in training. The Mini-batch RMSE and Mini-batch Loss columns show the performance of the model on each mini-batch. RMSE (Root Mean Squared Error) measures how well the model's predictions align with the actual values, with smaller values indicating better accuracy. The Mini-batch Loss represents the calculated loss for the mini-batch, typically resulting from a loss function. A decrease in RMSE and Loss demonstrates that the model is learning effectively from the data. The Base Learning Rate column shows the constant learning rate (0.0100), which determines how much the model's weights are adjusted during training.

Figure 1. Training progress of the proposed model

Finally, after completing all epochs, the Mean Squared Error (MSE) value of 4.6604 reflects the average squared difference between predicted and actual wine quality, with lower values indicating better model performance. The R-squared (R^2) value of -0.15668 is negative, suggesting that the model is performing poorly. Ideally, R^2 should be between 0 and 1, with higher values indicating better model fitting. A negative R^2 means that the model is not capturing the data's variance effectively, signaling the need for potential adjustments in the model or data processing.

Table 1. Training on single CPU

Epoch	Iteration	Time elapsed (hr: mm: ss)	Mini-batch RMSE	Mini-batch Loss	Base Learning Rate
1	1	00:00:06	0.40	8.1e-02	0.0100
2	50	00:00:07	0.31	4.7e-02	0.0100
4	100	00:00:08	0.36	6.5e-02	0.0100
6	150	00:00:08	0.30	4.5e-02	0.0100
8	200	00:00:08	0.35	6.1e-02	0.0100
10	250	00:00:09	0.25	3.1e-02	0.0100
12	300	00:00:09	0.35	6.3e-02	0.0100
14	350	00:00:09	0.35	6.0e-02	0.0100
16	400	00:00:09	0.33	5.3e-02	0.0100
18	450	00:00:10	0.31	4.8e-02	0.0100

20	500	00:00:10	0.31	5.0e-02	0.0100
22	550	00:00:10	0.32	5.1e-02	0.0100
24	600	00:00:10	0.29	4.3e-02	0.0100
26	650	00:00:10	0.33	5.6e-02	0.0100
28	700	00:00:11	0.28	3.9e-02	0.0100
30	750	00:00:11	0.27	3.7e-02	0.0100
32	800	00:00:11	0.32	5.1e-02	0.0100
34	850	00:00:11	0.29	4.3e-02	0.0100
36	900	00:00:11	0.30	4.5e-02	0.0100
38	950	00:00:12	0.31	4.9e-02	0.0100
40	1000	00:00:12	0.37	6.8e-02	0.0100
42	1050	00:00:12	0.31	4.7e-02	0.0100
44	1100	00:00:12	0.28	4.0e-02	0.0100
46	1150	00:00:13	0.29	4.1e-02	0.0100
48	1200	00:00:13	0.33	5.3e-02	0.0100
50	1250	00:00:13	0.31	4.7e-02	0.0100

Training finished: Max epochs completed

Mean Squared Error (MSE): 4.6604

R-squared (R^2): -0.15668

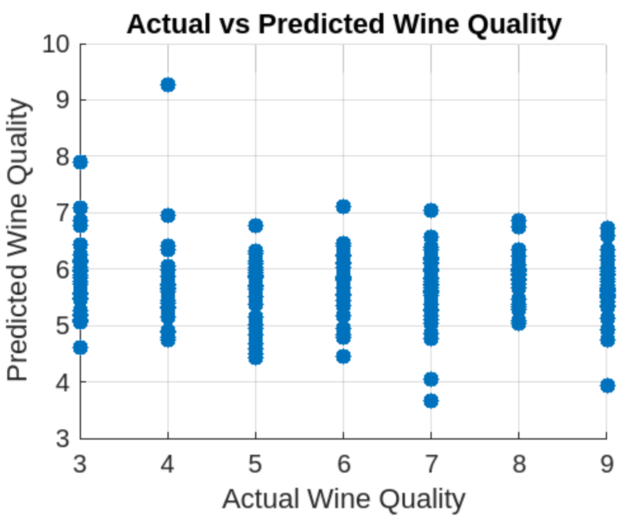

Figure 2. Actual vs Predicted Wine Quality

Fig.2 compares the actual and predicted wine quality ratings. The horizontal axis represents the actual quality, while the vertical axis shows the predicted values. The clustering of points around specific quality levels indicates the model's ability to approximate predictions. However, noticeable dispersion suggests the model struggles to perfectly align predictions with actual values, pointing to potential areas for improvement in the model's accuracy.

The residuals distribution in Fig.3 shows the frequency of errors (differences between actual and predicted wine quality). Most residuals are concentrated around zero, indicating that the model's predictions are generally accurate. The symmetric shape suggests the errors are evenly distributed without significant bias. However, the presence of outliers on both ends shows that the model struggles with certain predictions, leaving room for further optimization.

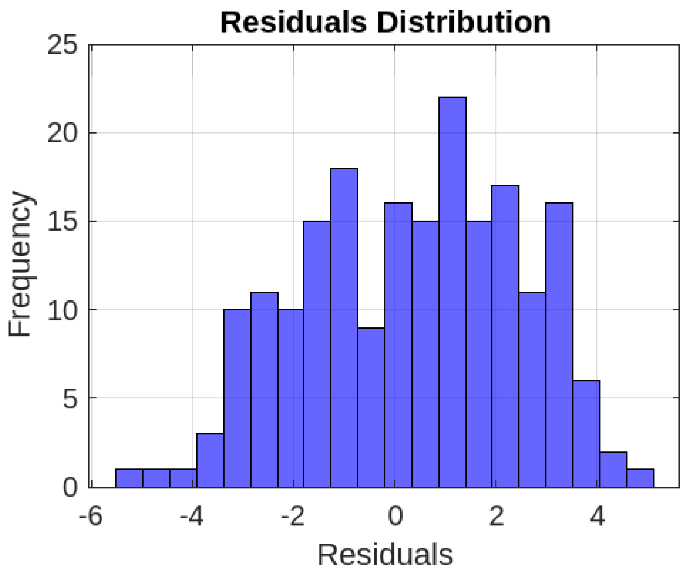

Figure 3. Residuals distribution

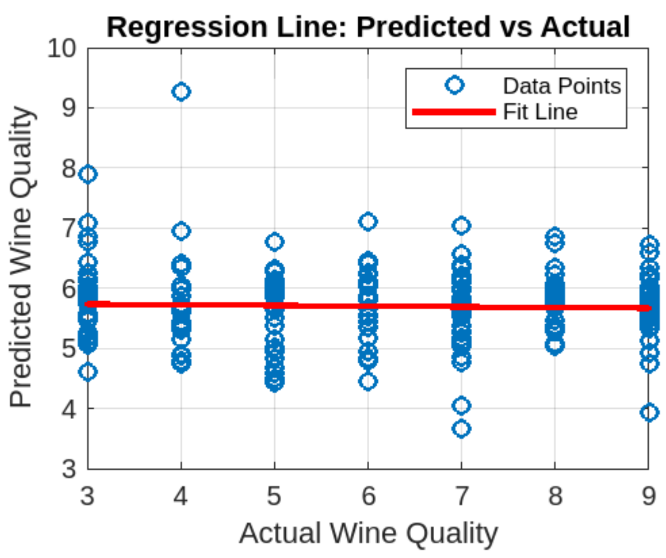

Figure 4. Relation between predicted wine quality and the actual wine quality in the regression model

Fig.4 shows the relationship between the predicted wine quality (y-axis) and the actual wine quality (x-axis) in a regression model. The blue circles represent the data points, indicating the model's predictions compared to the actual values. The red line represents the fit line of the regression model. The flat red line suggests that the model is not effectively capturing the variability in wine quality, likely underperforming, or predicting constant values regardless of the actual input.

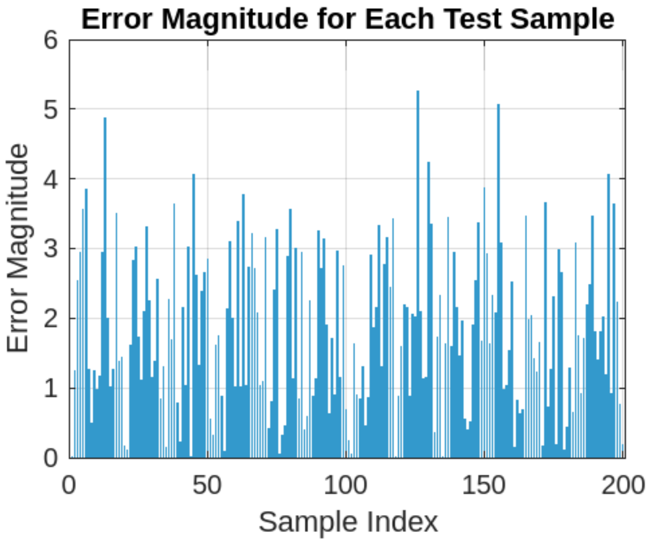

Figure 5. Error magnitude for each data sample in a regression model

Fig.5 displays the error magnitude for each test sample in a regression model. The x-axis represents the sample index (1 to 200), and the y-axis shows the corresponding error magnitude (absolute difference between predicted and actual values). The varying heights of the bars indicate the differences in prediction accuracy across samples. Some samples have higher errors, suggesting inconsistent model performance.

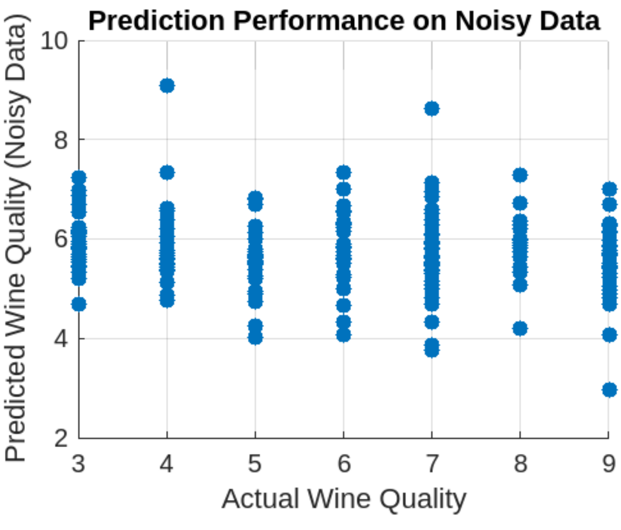

Figure 6. Performance of the prediction model on noisy data

Fig.6 illustrates the performance of a prediction model on noisy data. The x-axis represents the actual wine quality, while the y-axis represents the predicted wine quality. Each blue dot corresponds to a data point. Dispersion of dots along the vertical axis for each actual wine quality value implies that noise influences the model and causes the variation in the quality of wines that the model predicts. The grouping around specific predicted values suggest some form of bias or constrains in the model.

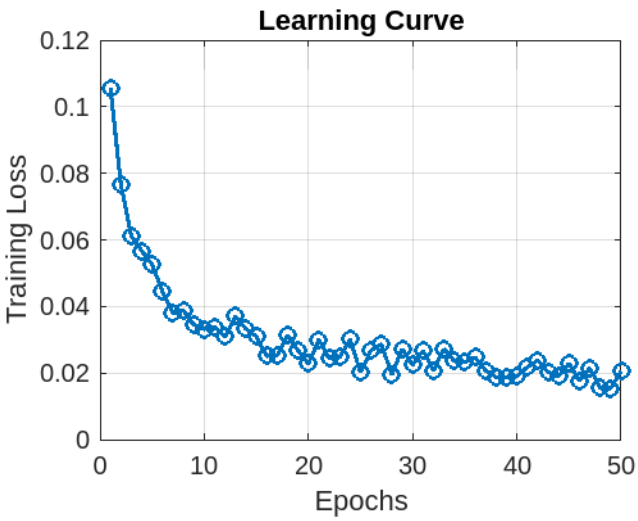

Figure 7. Learning curve illustrating the training loss versus the number of epochs at which the proposed model trained.

It is clear from Fig.7 that training loss follows a learning curve, which is plotted as epochs for the proposed model. Right from the epochs 1 to epochs 10, the training loss reduces very fast, but a progressive reduction is observed after epochs 10. This infers to the effectivity of the model as a learning algorithm and as it gravitates towards the solutions space optimum. Their oscillations in the later epochs point to further refinement of the model. Altogether, such an arrangement of the curve indicates the train model effectiveness and reduction of the loss' value in the process.

6. Conclusion

This work presents a new framework for improving the quality of predicted wines employing Dynamic Integral Neural Networks with Anti-Interference Optimization. The proposed framework considers the issues that are associated with the noise and variability of wine production data as a robust model for predicting wine quality. Algorithms show that the proposed DINN-based test model improves consistency, accuracy, and robustness compared to classical and modern ML approaches. This study benefits the general development of enhancing the wine business by presenting reliable strategic reference data to assist in the repeated production of high-quality wines under various production conditions. Further research could include deepening the model presented by adding more optimization techniques from mathematics, and using the framework proposed in this work to other fields that face similar data issues.

References

[1] J. Martinez-Falco, et al., "*Aligning the Sustainable Development Goals in the Wine Industry: A Bibliometric Analysis*", Sustainability, vol. **15**, no. **10**, **2023**.

[2] N. Ohana-Levi, Y. Netzer, "*Long-Term Trends of Global Wine Market*", Agriculture, vol. **13**, no. **1**, **2023**.

[3] M. R. Mozell, L. Thach, "*The impact of climate change on the global wine industry: Challenges & Solutions*", Wine Economics and Policy, vol. **3**, no.**2**, pp. **81-89**, **2014**.

[4] V. Goyal., "*Predictive Analysis of Crop Yield Based on Environmental and Soil Conditions*", International Journal on Computational Modelling Applications, vol. **01**, no. **02**, pp. **50-63**, **2024**.

[5] I.H. Sarker, "*Machine Learning: Algorithms, Real-World Applications and Research Directions*", SN Computer Science, vol. **2**, no. **160**, **2021**. https://doi.org/10.1007/s42979-021-00592-x

[6] P. W. Tien, et al., "*Machine Learning and Deep Learning Methods for Enhancing Building Energy Efficiency and Indoor Environmental Quality – A Review*", Energy and AI, vol. **10**, **2022**.

[7] S. Kumar, Y.Kraeva, R.Kraleva, M.Zymbler, "*A Deep Neural Network Approach to Predict the Wine Taste Preferences*", In: Solanki, V., Hoang, M., Lu, Z., Pattnaik, P. (eds) Intelligent Computing in Engineering. Advances in Intelligent Systems and Computing, vol. **1125**. Springer, Singapore, **2020**. https://doi.org/10.1007/978-981-15-2780-7_120

[8] B. Debska, B. Guzowska-Swider, "*Application of artificial neural network in food classification*", Analytica Chimica Acta, vol. **705**, no. **1-2**, pp. **283-291**, **2011**.

[9] D. K. Jana, P. Bhunia, S. Das Adhikary, A. Mishra, "*Analyzing of salient features and classification of wine type based on quality through various neural network and support vector machine classifiers*", Results in Control and Optimization, vol. **11**, **2023**.

[10] Y. Xu, et al., "*Artificial intelligence: A powerful paradigm for scientific research*", The Innovation, vol. **2**, no. **4**, **2021**.

[11] D. Kulasiri, S. Somin, S. K. Pathirannahalage, "*A Machine Learning Pipeline for Predicting Pinot Noir Wine Quality from Viticulture Data: Development and Implementation*", Foods, vol. **13**, no. **19**, **2024**.

[12] Y. Su, M. C. Wang, S. Liu, "*Automated Machine Learning Algorithm Using Recurrent Neural Network to Perform Long-Term Time Series Forecasting*", Computers, Materials, and Continua, vol. **78**, no. **3**, pp. **3529-3549**, **2024**.

[13] N. Khilari, et al., "*Analysis of Machine Learning Algorithm to predict Wine Quality*", Archives, vol. **9**, no. **2**, **2022**.

[14] M. Abdalla, "*Early Detection and Classification of Biomedical Atherosclerosis using Feature Subset Selection with Optimal Deep Learning Model*", International Journal on Smart & Sustainable Intelligent Computing, vol. **01**, no. **01**, **2024**.

[15] S. K. Mohanta, et al., "*Deep learning is a State-of-the-Art Approach to Artificial Intelligence*", Deep Learning Concepts in Operations Research, Auerbach Publications, pp. **27–43**, **2024**.

[16] H. Nugroho, N. P. Utama, K. Surendro, "*Normalization and outlier removal in class center-based firefly algorithm for missing value imputation*", Journal of Big Data, vol. **8**, no. **129**, **2021**.

[17] I. D. Mienye, T. G. Swart, G. Obaido, "*Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications*", Information, vol. **15**, no. **9**, **2024**.

[18] Aakash, S. Amutha, D. Nandhini, Ansh, "*Classification and Validation of Tomato Leaf Disease Using Deep Learning Techniques*", International Journal on Engineering Artificial Intelligence Management, Decision Support, and Policies, vol. **1**, no. **1**, pp. **41-60**, **2024**.

[19] I. H. Sarker, "*Machine Learning: Algorithms, Real-World Applications and Research Directions*", SN Computer Science, vol. **2**, no. **160**, **2021**. https://doi.org/10.1007/s42979-021-00592-x

[20] D. Fabiyanto, Y. Rianto, "*Performance Evaluation of Multiple Deep Learning Models for Wine Quality Prediction*", Telematika, vol. **21**, no.**2**, **2024**.