LEX LOCALIS-JOURNAL OF LOCAL SELF-GOVERNMENT
ISSN:1581-5374 E-ISSN:1855-363X
VOL. 23, NO. S5(2025)

OPTIMISING MACHINE LEARNING TECHNIQUES FOR IRREGULAR SAMPLING

Zhenyu Xu[1], Negar Riazifar[2]

[1]WMG Department,University of Warwick,Coventry,United Kingdom
[2]WMG Department,University of Warwick,Coventry,United Kingdom

*CorrespondingAuthor:zhenyu.xu@warwick.ac.uk

Abstract

This study examines how simple linear interpolation (SLI) and natural-neighbourinterpolation (NNI) affect machine learning model performance on irregularly sampled commercial data. Seoul bike-sharing rental datasetispre-processed with SLI and NNI to manage missing values and inconsistencies. The performance of SLI and NNI isthen evaluated by constructing various machine learning models, including XGBoost, Random Forest, k-nearest neighbors(KNN) and Stacking model. Results show that SLI consistently improved the accuracy, particularly in the stacking model, as demonstrated by the area under the receiver operating characteristic(AUC) and kolmogorov-smirnow(KS) statistics. Conversely, NNI had more variable outcomes, occasionally reducing performance. The findings underscore the critical role of data pre-processing throughout machine learning, particularly in domains where data irregularities are prevalent, thereby providing empirical support for employing interpolation methods to improve both model reliability and accuracy. Eventually, findings uncovered by this study empirically support data pre-processing for business data modelling, highlighting the critical role of data pre-processing in optimising the performance of machine learning models.

Keywords: Machine Learning, Random Forest, Simple Linear Interpolation, XGBoost

INTRODUCTION

1 Irregular Data

Irregular data, markedby uneven time intervals or non-uniform value distributions [1], results from inconsistent sampling, missing observations, or unpredictable patterns [1], [2].It is common in finance, healthcare, and environmental monitoring [3], [4].For example, intensive care data can exhibit up to 80% sparsity, compared to ~13.3% in particulate matter 2.5 records [1],[5].

Irregularity stems from sensor variability, operational differences, or manual recording [1]. Univariate irregularity involves inconsistent observation intervals, while multivariate irregularity results from mismatched sampling rates across variables [1]. Such data challenges traditional machine learning models, which assume fixed-size, fully observed inputs. Sensor failures or environmental factors further distort patterns, complicating analysis—e.g., seismic irregularities cause spectral leakage, addressed via techniques like non-equispacedfast antileakage fouriertransform (NFALFT)[6]. Modern approaches like multi-time attention networks (mTAN) improve interpolation and classification accuracy [5].

Outliers, equipment failures, or non-stationarity exacerbate irregularities[7]. While convolutional neural network – gated recurrent unit(CNN-GRU) networks enhance financial data handling[8], they struggle with abrupt market shifts[9]. Generative AI (e.g., financial time series-diffusion) effectively models scale-invariant patterns, reducing prediction errors[7].

2 Interpolation to Irregular Data

The presence of irregular sampling causes analytical challenges and diminishes the performance of

standard data-driven approaches. To mitigate these issues, interpolation methods are used to reconstruct regular, evenly sampled datasets from irregularly spaced observations.Interpolation methods for irregular data are typically based upon either move-out or partial prestack migration, which commonly demands unaliased data [10]. That is, the particular approach can also be summarised as estimating the variable value according to known data points [11], such as geospatial information. Sambridge, Braun, and McQueen proposed natural-neighbor interpolation (NNI) for irregular data, particularly in geophysical contexts[11]. This interpolation is known as Spatial interpolation, as demonstrated in terrain meshing for irregular datasets[11]. In contrast, extrapolation estimates values outside the observed range. A key consideration is smoothness—ensuring continuity between data points. Most interpolation methods rely on partial differential equations (PDEs) with first-derivative continuity [11]. Linear interpolation can cause first-derivative discontinuities at boundaries, while NNI ensures continuity in first and second derivatives (except at nodes) [11]. However, NNI may still exhibit discontinuities at nodes.While interpolation methods like NNI offer efficient solutions for irregular data, challenges such as node discontinuities persist, requiring careful selection based on application needs.

2.1 Simple Linear Interpolation

The most conventional interpolation method is simple linear interpolation (SLI), which is a fundamental numerical analysis method for estimating an unknown point's value between two or more known data points. To elaborate, the central idea is to assume a linear relationship between the data points and compute a value of the intermediate point by constructing a linear equation across the identified points. Indeed, Costabile[12]defines SLI as a method of approximating a function by means of a polynomial[12].Specifically, linear interpolation involves constructing a polynomial at a given point such that the polynomial passes through those points, thus processing precise interpolation. Also, Costabile mentions basic concepts and theorems about linear interpolation, including how to apply Sheffer polynomials for interpolation and how to extend these models to more general cases[12].As a fundamental method, SLI has applications in many fields. Firstly, it has been shown that SLI can significantly reduce the number of nonlinear analyses needed to optimise analysis in the successive response surface method (SRSM), which is a metamodel-based and generalised approach to facilitate nonlinear optimisation[13]. This is because SLI is capable of reducing unnecessary oversampling by integrating interpolated linear polynomials rather than regression models. In this way, total number of nonlinear analyses can be controlled without sacrificing accuracy. Meanwhile, a study has proposed the implementation of adaptive multipath linear interpolation (AMLI) based on SLI[13]. This method could effectively expand the sampled data processing and reduce the overall mean squared error(MSE)of the samples, thus controlling for the proportion of large error samples. Also, interpolation plays an essential role in finance and is frequently integrated into more complex algorithms. The speed of linear interpolation, one of the most straightforward interpolation techniques, explains why it is one of the most common forms of interpolation for difficult-to-compute functions.

2.2 Natural-Neighbour Interpolation

Natural-neighbourinterpolation (NNI) is a spatial interpolation approach whose core idea is to determine the natural neighbour of an interpolated point based on a Voronoi diagram and to develop an estimate of the point based on the relative weights of these neighbours[11].Unlike simple nearest-neighbour interpolation, the natural neighbour interpolation method provides smoother interpolation results derived from multiple points in the neighbour and is particularly suitable for irregularly

distributed data points. The main advantage of the natural neighbour interpolation method is its ability to dynamically adjust the number of data points used according to both the configuration of data points and target points[14]. To be precise, this method initially produces a Voronoi diagram of the data points and the target point, from which data points neighbouring the target point can be identified [14]. The data values of these natural neighbours and the weights calculated from the Voronoi diagram are then employed for further interpolation[14]. This strategy considers the spatial distribution of the data points, enabling the interpolation results to be more flexible and accurate. Due to these properties, NNI has been widely adopted in the fields such as geographic information systems (GIS), geology, and environmental modelling[15].

Delaunay triangulations and Voronoi diagrams are key geometric tools for handling irregular data interpolation [11]. Delaunay triangulations split data into nearly equilateral triangles using the empty circle property, which minimises errors by preventing elongated or distorted shapes—making them valuable in computational geometry and GIS [11], [16].In parallel, Voronoi diagrams partition space into convex cells that capture the region of influence around each data point, a feature that underpins spatial interpolation methods like natural-neighbourinterpolation [17]. Together, these structures offer a robust foundation for accurate and smooth interpolation across diverse fields.

3 Machine Learning Model Building

Developing machine learning models for irregular data involves methods that enhance performance while addressing common challenges. Key approaches include feature selection, algorithm choice, model tuning, and performance validation. For example, stepwise feature selection reduces noise by identifying predictive variables, while synthetic minority over-sampling technique (SMOTE) addresses data imbalance. Hyperparameter tuning improves accuracy, and hold-out set testing verifies generalisation to unseen data, mitigating overfitting. Each method offers benefits and trade-offs, underscoring the need for a balanced, validated approach.

3.1 Stepwise Feature Selection

Feature selection is a crucial pre-processing step in machine learning that improves accuracy and efficiency by reducing dimensionality and isolating the most relevant features[18]. It has been a research focus, enhancing interpretability and generalisation across domains including finance, healthcare, marketing, and bioinformatics [19].

Stepwise feature selection, originally designed for linear regression, systematically adds or removes features based on statistical significance, typically assessed through p-values [19]. As a wrapper method, feature selection is treated as an optimisation problem that iteratively evaluates feature subsets using a predictive model, directly enhancing performance by identifying those that most improve predictive accuracy for the targeted algorithm [19]. Despite its benefits, stepwise selection can lead to over-selection, incorporating irrelevant features that increase complexity, bias, and reduce generalisation. The iterative nature of stepwise methods can lead to instability, with dataset variations causing inconsistent feature selection that undermines model reliability and interpretability [19]. Moreover, relying solely on p-values may omit key features or select noise-driven variables, ultimately reducing model robustness and accuracy [20].

This study used Python's 'mlxtend' library for stepwise feature selection, with data scaled using StandardScaler and classification performed via LGBMClassifier. Accuracy improvements and feature importance were visualised at each step, offering clear insights into variable significance.

3.2 Machine Learning Algorithm and SMOTE

This study builds baseline models using popular machine learning algorithms—Logistic Regression, Decision Tree, K-Nearest Neighbours, Random Forest, Gaussian Naive Bayes, Light GBM, XGBoost, and Gradient Boosting. After evaluating their initial accuracy across the dataset, the top three performers are selected for hyperparameter optimisation.

Additionally, the synthetic minority over-sampling technique (SMOTE) is employed to address class imbalance, a common challenge in practical dataset. SMOTE creates new synthetic minority class samples by interpolating between neighbouring minority data points within the feature space, thereby balancing the dataset without simply duplicating existing samples [21]. This interpolation-based approach helps reduce the risk of overfitting caused by exact replication, improving model robustness and predictive accuracy on imbalanced dataset [21]. Despite its effectiveness, SMOTE also has limitations; synthetic samples may not always precisely reflect the true distribution of the original minority class, occasionally resulting in fluctuations or inaccuracies in model performance[22].

Several machine learning algorithms are commonly employed for classification and regression tasks. Logistic Regression applies a logistic function to map input features to probabilities in a linear model for binary classification [22].LightGBM is a fast and efficient gradient boosting framework that uses histogram-based methods, making it suitable for high-dimensional and large-scale data [23]. Decision Trees partition the feature space into regions through conditional splits at each node [22], while XGBoost offers a performance-optimised form of gradient boosting with regularisation to prevent overfitting [24]. K-Nearest Neighbors is a non-parametric method that classifies or predicts based on the majority vote or average of the K-nearest training samples [22]. Gradient Boosting builds an ensemble of weak learners sequentially, correcting errors from previous models [25]. Random Forests aggregate the outputs of multiple decision trees to improve accuracy and reduce overfitting risks [22]. Neural Networks, inspired by biological neural systems, model complex non-linear relationships through interconnected layers of neurons [22]. Lastly, Gaussian Naive Bayes is a probabilistic classifier based on Bayes' theorem, assuming feature independence and Gaussian distribution for continuous data [26].

3.3 Model Tuning

After identifying the three best-performing algorithms from each dataset, hyperparameter optimisation (HPO) is applied to further enhance model performance. Hyperparameters, which are distinct from model parameters, play a crucial role in influencing algorithm behaviour and need to be fine-tuned based on the specific dataset to achieve optimal results [27].Given that optimal hyperparameters vary depending on dataset characteristics, systematic tuning is essential to maximise the predictive capability of each model.

In our study, hyperparameter tuning involved generating an exhaustive list of parameter combinations using a defined search function. Each combination was assessed through a brute-force search approach, systematically evaluating performance across all possibilities. Ultimately, the search identified the best-performing hyperparameter combination, along with the corresponding optimal model score and SMOTE settings.

3.4 Stacking Model Building and Tuning

To attain a comprehensive evaluation of the predictive performance, accounting for the interpolation methods' effects, our study employs a stacking model, also known as stacked generalisation. Stacking is

an ensemble learning technique that integrates multiple base models to enhance overall predictive accuracy[24]. Specifically, predictions from selected base models serve as input features for a higher-level "meta-learner," which generates the final prediction by leveraging the strengths of each contributing model[24]. This approach provides a robust and accurate ensemble by effectively combining the individual predictive capabilities of the three previously selected algorithms.
In this research, the 'XGBClassifier' is used as the meta-learner within the stacking framework. To manage model complexity and prevent overfitting, the maximum depth of this classifier is constrained to three.

3.5 Hold-Out Set Test

This study employs the hold-out set test to assess model generalisability and prevent overfitting. Overfitting occurs when a model learns not only underlying patterns but also captures noise and irregularities from the training dataset, resulting in high performance on training data but poor performance on new, unseen data[28]. Such models often fail to generalise to scenarios or novel datasets that werenot seen, thus are of little practical value.

To mitigate this issue, the dataset is split into two subsets: a training set, used exclusively for model training, and a test set, which evaluates model performance on previously unseen data. This method assesses the model's capability to adapt to new scenarios, thereby determining its reliability and robustness [28]. However, the accuracy of this evaluation heavily depends on how effectively the data is split into training and testing subsets. This study systematically applies the hold-out method to evaluate the tuned models and calculates relevant performance metrics, ultimately reporting each model's generalisation performance.

Due to the complexities and constraints associated with irregular data, conventional machine learning methods need preprocessing steps to enable stable and precise analysis. Interpolation techniques are commonly used as an effective step for handling irregularities, imputing missing observations, and normalising the format of the data for modelling. The mothods section discusses the theoretical aspects and practical implementation of interpolation techniques for application in irregular data sets.

OBJECTIVES

In data science and predictive analytics, the treatment of irregularly sampled data faces significant challenges, especially when such data are utilised in machine learning models for business and financial applications. To be precise, irregular data resulting from uneven sampling intervals, missing values, or outliers could obscure underlying patterns and lead to sub-optimal model performance. Therefore, effective pre-processing techniques, such as interpolation methods, are essential for reconstructing more consistent and continuous datasets to enhance the predictive power of these models. Standard techniques entail linear interpolation, spline interpolation, and kriging interpolation, which are particularly efficient at handling seismic and meteorological data[1].

This study aims to investigate the impact of two interpolation methods—simple linear interpolation (SLI) and natural-neighbourinterpolation (NNI)—on the accuracy and effectiveness of predictive models. For this study, we used Seoul bike-sharing rentals datasets, and the research examines how these interpolation techniques affect the performance of various machine learning algorithms, involving XGBoost, Random Forest, and stacking models.

METHODS

1 Data Collection

Following the theoretical discussion, the empirical methodology is presented as follows. Our research strategy is outlined in the next section, beginning with the comprehensive explanation of urban transport dataset, exemplifying these irregularities. These datasets serve as the empirical basis for comparing and evaluating the interpolation and machine learning techniques outlined above.

The dataset comprises Seoul bike-sharing records from the UCI Repository[29], featuring daily rentals alongside weather variables (temperature, humidity, wind speed, etc.).The irregularities in this dataset are primarily due to incomplete observations or anomalies in sensor recordings and weather-induced variations, reflecting realistic data quality issues encountered in transportation and environmental domains. The spatial representation of the number of bicycles rented is displayed in Figure 1.

Analysis of 6,570 observations per parameter shows a mean of around 700 daily rentals, with significant variability in weather metrics (e.g., temperature, rainfall). Descriptive statistics highlight these fluctuations, underscoring the dataset's suitability for testing irregular data methods.

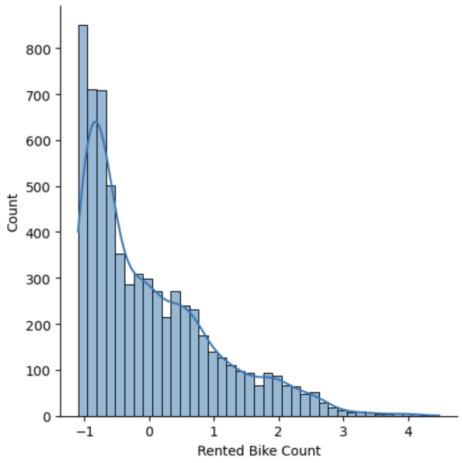


Figure 1. Volume distribution map of the Seoul Bike Data

Interpolation is most effective when data points are moderately dispersed, but preprocessing is essential for highly irregular datasets. First, missing values and outliers are identified using an outlier detection algorithm. Once anomalies are processed, the dataset is normalised using Min-Max Normalisation or Z-score Standardisation to standardise feature distributions. Min-Max Normalisation, which scales values within the range [0,1], is applied using the formula:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1)$$

Alternatively, Z-score Standardisation is used to adjust data so that it has a mean of 0 and a standard deviation of1, ensuring consistency across features:

$$x' = \frac{x - \mu}{\sigma} \quad (2)$$

After applying the appropriate normalisation method, the dataset is re-examined to confirm that no bias or loss of crucial information has been introduced.

2 Linear Interpolation Method

Simple Linear Interpolation is a widely used method due to its efficiency and applicability, particularly

for estimating missing values in irregular datasets. It approximates unknown values by assuming a linear relationship between adjacent known data points. The interpolation formula is given as:

$$y = y_1 + \frac{(y_2 - y_1)}{(x_2 - x_1)} \times (x - x_1) \quad (3)$$

For multivariate datasets, interpolation is performed independently across each dimension. Given a dataset $\{x_i, y_i\}$,where $x_i = (x_i^{(1)}, x_i^{(2)}, x_i^{(3)}, \ldots, x_i^{(n)})$is an n-dimensional feature vector, interpolation is conducted for each dimension j(j=1,2,3,…,n):

$$y = y_1 + \frac{y_2 - y_1}{x_2^{(j)} - x_1^{(j)}} \times (x^{(j)} - x_1^{(j)}) \quad (4)$$

To determine the final interpolated value, a weighted average of individual dimension estimates is computed:

$$y_{final} = \frac{\sum_{j=1}^{n} w_j y_j}{\sum_{j=1}^{n} w_j} \quad (5)$$

wherew_j represents the weight assigned to each dimension. This approach ensures robust interpolation for multidimensional datasets. The entire SLI process is illustrated in Figure 2 (a).

3 Natural-Neighbour Interpolation Method

Natural-neighbour interpolation is a spatial interpolation technique based on Voronoi diagrams, effectively handling irregularly distributed data. This method determines the interpolated value by assessing how a new point affects surrounding Voronoi cells. By inserting the new point into an existing Voronoi structure, NNI calculates the area changes in neighbouring cells to assign interpolation weights. Algorithms such as Fortune's algorithm enable the efficient construction of Voronoi diagrams for this process.

Interpolation begins by inserting the unknown target points into the existing Voronoi diagram. The change in area $\Delta A_i$for each neighbouring Voronoi cell is calculated, determining its influence through a weight formula:

$$w_i = \frac{\Delta A_i}{\sum_{j=1}^{n} \Delta A_j} \quad (6)$$

where $w_i$represents the weight of the i_{th}neighbouring point. The interpolated value is then determined by evaluating the weighted average:

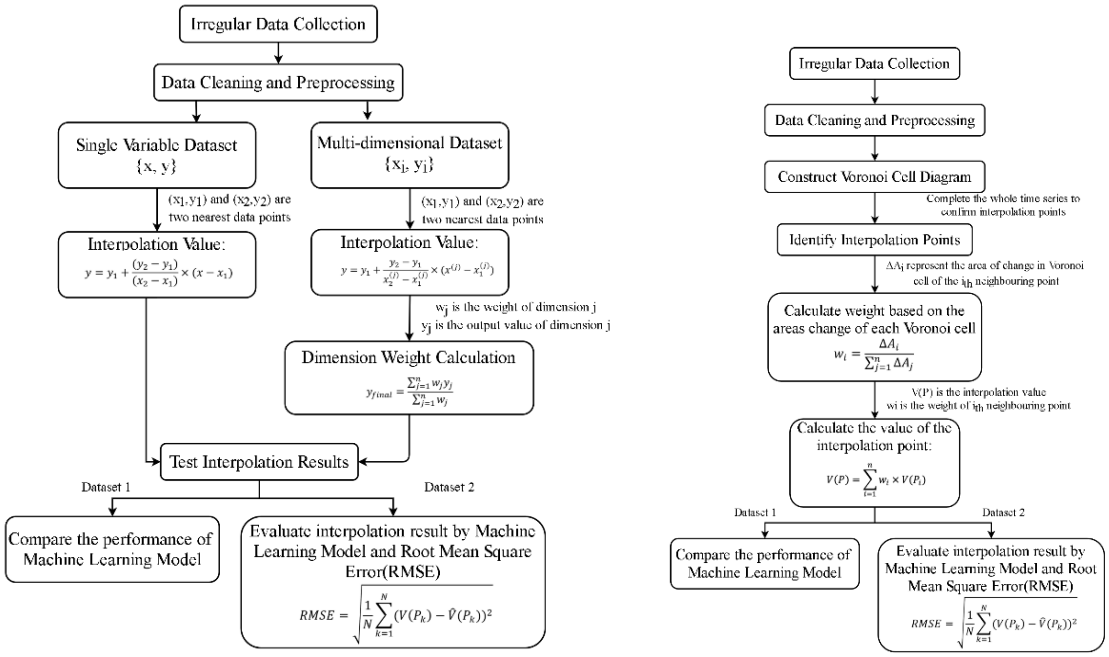
$$V(P) = \sum_{i=1}^{n} w_i \times V(P_i) \quad (7)$$

where, V(P) is the value of the interpolation points.

To validate the interpolation results, cross-validation is performed by dividing the dataset into training and validation sets. The accuracy of NNI is measured using the root mean square error (RMSE), a widely used metric in regression analysis and predictive modelling [47]:

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(V(P_k) - \hat{V}(P_k))^2} \quad (8)$$

where $V(P_k)$ is the true value, and $\hat{V}(P_k)$is the predicted value of interpolation. RMSE quantifies the average deviation between interpolated and actual values, ensuring the effectiveness of NNI. The entire processes of NNI method are shown in Figure 2 (b).



(a) (b)

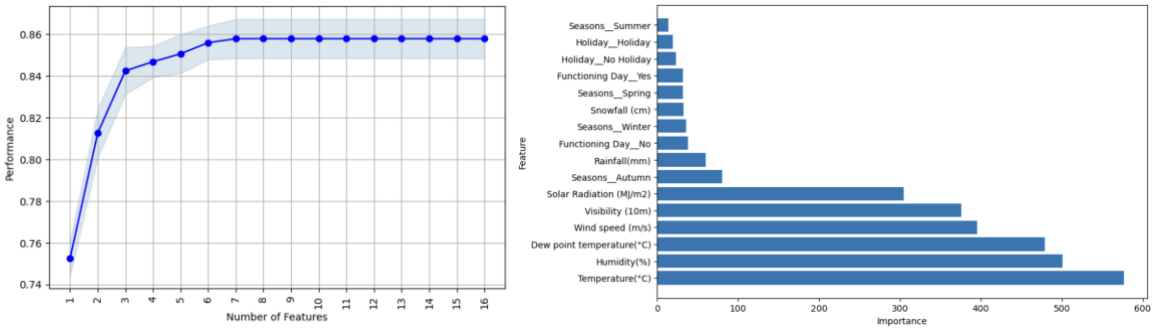
Figure 2. A block diagram of (a)Simple Linear Interpolation Method and (b)Natural-Neighbour Interpolation method

RESULTS AND DISCUSSION

1 Feature Selection for the Dataset

Feature selection plays a crucial role in optimising model performance by identifying the most relevant predictors and eliminating redundant variables. The stepwise feature selection method was applied to the dataset to assess the impact of different features on model accuracy. The results are illustrated in Figure 3, which presents (a) the model's performance trend as features are added and (b) the relative importance of each feature.

With results shown in Figure3, depicting the model's performance trend and feature importance rankings. Panel demonstrates that model performance improves considerably when the first three features are selected, but gains become marginal after the fourth feature. This suggests that additional features either contribute less meaningful information or introduce redundancy. The confidence interval in the early stages is wide, indicating variability, but narrows as more significant features are chosen. However, as more features are added, the confidence interval broadens again, reflecting increased uncertainty in performance stability.



(a) (b)

Figure 3.(a) Performance of stepwise feature selection on the dataset as the number of selected features increases. (b)Feature importance rankings for the dataset

Feature importance rankings in Figure 3highlight 'Temperature' as the most influential predictor, followed by 'Humidity' and 'Dew Point Temperature'. Other environmental factors, such as 'Wind Speed,' 'Visibility,' and 'Solar Radiation,' also contribute to the outcome to a lesser extent. Meanwhile, 'Seasons' and 'Holiday' are the least significant, likely due to their low correlation with target variables or the presence of collinearity with other features. If model performance deteriorates, these low-impact features may be removed in future iterations to enhance stability and efficiency.

In summary, the dataset demonstrated that model accuracy improved significantly with the first three selected features, with diminishing returns. The most impactful predictors were environmental variables such as Temperature and Humidity, while Seasonality and Holiday information were less relevant and could be considered for removal in subsequent model refinements.

2 Baseline Model Comparison

Following the construction of baseline models, three key performance metrics—Accuracy, area under the curve (AUC), and kolmogorov-smirnovstatistic(KS)—are employed in this study to evaluate their effectiveness. Accuracy is a widely used metric representing the percentage of correctly classified samples [30]. However, it can be misleading for imbalanced datasets, as high accuracy does not necessarily indicate strong predictive power. For instance, in medical applications, a model predicting only the majority class (e.g., 'healthy') may still achieve high accuracy while failing to detect minority cases (e.g., 'diseased'), rendering it ineffective in critical real-world scenarios[31].

Accuracy assesses the model's capacity to distinguish between positive and negative categories, tends to be less stable than AUC[30]. AUC, ranging from 0.5 to 1, evaluates the model's ranking ability rather than absolute correctness. It is more reliable than accuracy, particularly for imbalanced datasets, as it measures the likelihood that a randomly chosen positive sample ranks higher than a randomly chosen negative sample [32].Meanwhile, KS Statistic, originally introduced by Kolmogorov and Smirnov in the 1930s [33],measures the maximum difference between the cumulative distribution functions of the positive and negative class predictions. A higher KS value indicates better model discrimination, making it particularly useful for evaluating classification models[34].

To optimise model performance, this study selects the three algorithms with the highest KS values from the baseline models for further tuning, as KS directly reflects class separability. While accuracy is intuitive and AUC provides an overall ranking measure, KS uniquely captures class differentiation. Based on the baseline model results (Table1), the top-performing models selected for the dataset were XGBoost, Gradient Boosting, and Random Forest.

Table 1. Dataset Baseline Model Result

Model	Accuracy	AUC	KS	Model	Accuracy	AUC	KS
Random Forest	0.872	0.949	0.752	Decision Tree	0.857	0.933	0.721
XGBoost	0.872	0.948	0.755	Neural Network	0.864	0.940	0.736
K Nearest Neighbors	0.840	0.928	0.692	Logistic Regression	0.853	0.930	0.714
Light GBM	0.871	0.944	0.752	Gaussian Naive Bayes	0.767	0.892	0.644
Gradient Boosting	0.871	0.945	0.753				

3Root Mean Square Error Evaluation

To assess the quality of interpolation, we adopted root mean square error (RMSE) as a metric to measure the difference between interpolated values and actual values. A lower RMSE value indicates a more accurate interpolation, meaning that the interpolated data closely matches the real values. This study evaluates RMSE scores for both simple linear interpolation (SLI) and natural-neighbour interpolation (NNI) across various features in the dataset. The results are summarised in Table 2 (SLI) and Table 3 (NNI), while Figures 4 and 5 visualise RMSE scores for each feature.

3.1 Simple Linear Interpolation RMSE Result

The SLI RMSE results (Table 2 and Figure 4) show that different features exhibit varying levels of interpolation accuracy. Notably, Temperature (RMSE = 0.0186) and Dew Point Temperature (RMSE = 0.0337) have very low RMSE values. This indicates high interpolation accuracy. This is expected since temperature is relatively stable over time, making it easier for linear interpolation to estimate missing values effectively. Similarly, categorical variables such as Seasons (Autumn, Spring, Summer, Winter) and Holidays show near-zero RMSE values, meaning they were well-interpolated with minimal error. This is likely because these features have clear categorical boundaries, making them easier to predict.

On the other hand, features with high variability by nature, such as Wind Speed (RMSE = 0.2756), Visibility (RMSE = 0.1440), and Rainfall (RMSE = 0.5982), have significantly larger errors. This suggests that SLI struggles to handle highly volatile and irregularly distributed data. The large RMSE for Rainfall in particular highlights a challenge—rainfall is inherently sporadic and varies widely across different time periods, making it difficult for a simple linear approach to capture accurate patterns. The Functioning Day feature has a slightly higher RMSE (0.1021), likely due to variations in human activity across different weekdays.

Table 2. Simple Linear Interpolation Root Mean Squared Error Test Result for Dataset

Feature	RMSE	Feature	RMSE
DateTime	0.000	Dew Point Temp.(°C)	0.034
Rented Bike Count	0.164	Solar Radiation	0.111
Temperature(°C)	0.019	Rainfall(mm)	0.598
Humidity(%)	0.087	Snowfall(cm)	0.034
Wind speed(m/s)	0.276	Seasons (Autumn/Summer)	0.000
Visibility(10m)	0.144	Seasons (Spring/Winter)	0.012
Holiday (Yes/No)	0.079	Functioning Day (Yes/No)	0.102

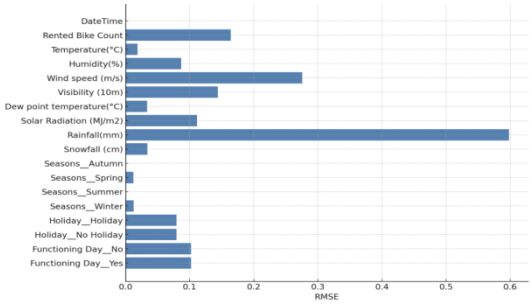


Figure 4. Simple Linear Interpolation Features Root Mean Squared Error Scores

3.2 Natural-Neighbour Interpolation Method

Overall, we may conclude that SLI performs well for stable environmental features but struggles with highly variable and sparse features like wind speed and rainfall. This highlights the limitations of simple linear interpolation for datasets with high volatility.

The NNI RMSE results (Table 3 and Figure 5) provide insights into how this method compares to SLI. Like SLI, NNI achieves very low RMSE values for Temperature (RMSE = 0.0416) and Dew Point Temperature (RMSE = 0.0456), demonstrating strong interpolation accuracy for stable environmental features. Additionally, categorical features like Seasons and Holidays remain well-interpolated, further supporting the method's effectiveness for well-defined categorical data.

However, NNI outperforms SLI for more complex features, particularly for Rented Bike Count (RMSE = 0.2281). This suggests that NNI better captures underlying trends in human activity-based data, likely due to its ability to adapt to non-linear relationships. Similarly, Humidity (RMSE = 0.1380) was interpolated with reasonable accuracy, though with slightly higher error than temperature-based features.

Despite these strengths, NNI still struggles with highly volatile variables, such as Wind Speed (RMSE = 0.3407), Visibility (RMSE = 0.1848), and Rainfall (RMSE = 0.6085). The high RMSE for Rainfall suggests that even advanced interpolation techniques struggle to predict highly sporadic events. However, NNI generally produces slightly better results than SLI for most numerical features, indicating its superiority for interpolating more complex patterns.

Table 3. Natural-Neighbour Interpolation Root Mean Squared Error Test Result

Feature	**RMSE**	**Feature**	**RMSE**
DateTime	0.000	Dew Point Temp.(°C)	0.046
Rented Bike Count	0.228	Solar Radiation	0.199
Temperature(°C)	0.042	Rainfall(mm)	0.609
Humidity(%)	0.138	Snowfall(cm)	0.106
Wind speed(m/s)	0.341	Seasons (Autumn/Summer)	0.000
Visibility(10m)	0.185	Seasons (Spring/Winter)	0.025
Holiday (Yes/No)	0.099	Functioning Day (Yes/No)	0.133

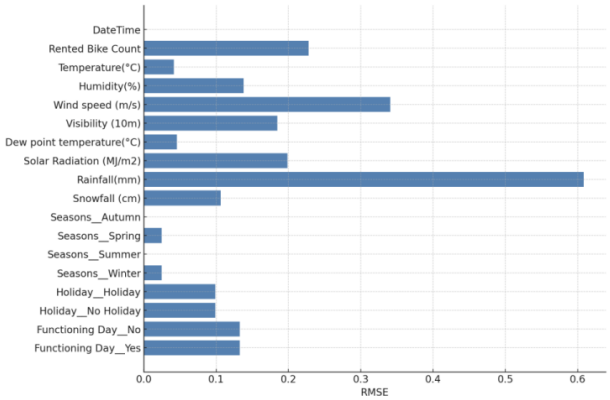


Figure 5. Root Mean Squared Error Scores for Natural-Neighbour Interpolation

3.3 Comparing SLI and NNI Interpolation

Based on the results from the abovementioned evaluation metrics, a direct comparison of SLI and NNI reveals some key takeaways: First, for stable features (e.g., Temperature, Dew Point Temperature, Seasons, Holidays), both methods perform well, with RMSE values close to zero. Second, for moderately volatile features (e.g., Humidity, Solar Radiation, Visibility), NNI slightly outperforms SLI, suggesting it better captures variations in environmental conditions. Third, for highly volatile features (e.g., Wind Speed, Rainfall), both methods exhibit high RMSE values, but NNI generally performs slightly better. Fourth, SLI is computationally simpler but may introduce errors when applied to complex, non-linear features, whereas NNI is better suited for handling non-linear spatial relationships.In conclusion, while SLI is a viable option for simpler datasets, NNI offers better performance for capturing complex patterns in data.

4 Model Performing Evaluation

4.1 Dataset Evaluation

Simple linear interpolation yields a more substantial enhancement in model performance, particularly during applications in the XGBoost and Random Forest models. The performance detail are shown in Table 4. Conversely, the impact of NNI is more intricate and might occasionally result in a decline in model performance. Applying basic linear interpolation to the XGBoost model led to a 1.06% increase in accuracy, a 0.44% increase in AUC, and a 1.37% increase in KS value. The results suggest that employing basic Generator and interpolation effectively enhances data consistency, thereby enabling the XGBoost model to capture patterns in the data more accurately. Nevertheless, the utilisation of NNI exerted little effect on the XGBoost model and occasionally resulted in a minor decline in performance, denoted by a 0.09% fall in accuracy.

In the Gradient Boosting model, the effect of simple linear interpolation is smoother, with a slight improvement in all metrics. However, the effect of NNI in this model is more negative, which not only fails to significantly improve the model performance but even leads to a marginal decrease in performance in some cases, where the AUC is declined by 0.09%.

The Random Forest model has revealed a relatively large performance improvement after applying simple linear interpolation. The accuracy is improved by 0.69%, the AUC by 0.84% and the KS value by 1.26%. This suggests that Random Forest requires data consistency, and simple linear interpolation can effectively improve its performance. In contrast, NNI is less effective in this model, with a marked decrease in both accuracy and AUC.

Stacking Model shows an improvement of 0.82% in accuracy and 1.62% in KS after applying simple linear interpolation, notwithstanding the fact that AUC decreased slightly by 0.56%. After applying NNI, the Stacking Model performs even worse, with a significant decrease of 1.66% in AUC, indicating the negative impact of NNI on the Stacking Model.

Table 4. Dataset Model Performance

ModelPerformanceinSharingBikeBusinessData									
	BaseData			**SimpleLinearInterpolation**			**Natural-neighbourInterpolation**		
Model	**Accuracy**	**AUC**	**KS**	**Accuracy**	**AUC**	**KS**	**Accuracy**	**AUC**	**KS**
XGBoost	0.8719	0.9501	0.7618	0.8811	0.9543	0.7722	0.8738	0.9503	0.7585
Gradient Boosting	0.8668	0.9439	0.7426	0.8706	0.9458	0.7470	0.8666	0.9431	0.7447
Random Forest	0.8759	0.9496	0.7613	0.8810	0.9547	0.7708	0.8679	0.9478	0.7551
Stacking Model (XGB)	0.8704	0.9436	0.7480	0.8728	0.9494	0.7603	0.8691	0.9276	0.7480

The findings uncovered by NNI as a complex model can even negatively affect model performance under certain environments. From a generalisability point of view, SLI has a more consistent effect on model enhancement compared with NNI. In the field of predictive modelling, particularly for applications with commercial data, model robustness and accuracy are critical determinants. When these metrics decline, the impact can be significant and multifaceted. To be specific, a 1.66% decline in AUC, while seemingly minor, implies a substantial reduction in the model's ability to differentiate between categories, for example, predicting whether or not there would be available bikes at a given site. This reduced predictive power can plausibly lead to more misclassifications, which, under a business context, translates into reduced operational efficiency. For example, with respect to a bike-sharing system, if the model incorrectly predicts the availability of bikes, it may result in too many or too few bikes at a site, thereby reducing user satisfaction and may lead to financial losses due to misallocation of resources. In addition, performance degradation may also erode user trust in the model, especially in situations where decisions are made based on the model itself. Under high-risk environments where models perform inconsistently after the data pre-processing phrase, it may be necessary to re-evaluate the data processing process or even reconsider the deployment of the model during production. Besides, a decrease in the KS value indicates a weakening of the model's ability to effectively identify the optimal thresholds for decision-making. This may exacerbate the problem of false positives or omissions, that in turn may render further inefficiencies or missed opportunities within the shared-vehicle network.

To sum up, the degradation of model performance after applying NNI demonstrates the need to evaluate data with preprocessing approaches with a critical development of a model. Minor adjustments might lead to tangible consequences for business operations and customer satisfaction. Therefore, it is essential to choose preprocessing techniques carefully and examine the specific needs of the model and the complexity of the data in order to accomplish optimal performance.

4.2 Model Comparison

SLI generally demonstrated a positive impact on model performance, especially in terms of accuracy,

AUC and KS values. This indicates the effectiveness of SLI in enhancing data consistency, which in turn allows the model to capture the underlying patterns in the data better. In addition, SLI is also effective in improving data consistency, which would, therefore, improve the predictive performance of the model. In contrast, the effectiveness of NNI is more variable. In some cases, it managed to bring about a slight improvement in metrics. However, in other cases, especially for more complex models such as stacked models, interpolation instead led to a decrease in performance. This phenomenon suggests that while NNI may capture local patterns in the data with precision, it may also introduce noise or overfit local data structures, thus degrading the overall model performance.

The differences in the performance of interpolation methods in distinct models may stem from several reasons. Firstly, different machine learning models demonstrate various levels of data dependency. For example, the XGBoost model has a built-in mechanism to handle missing data[24], so interpolation tends to have little impact on its performance. Other models, such as Random Forest and K Nearest Neighbors, rely more on data integrity and consistency. Thus, interpolation significantly improves subsequent performance. Secondly, the characteristics of the interpolation method itself also affect its effectiveness. SLI estimates the missing values through a linear function, which is suitable for cases where the data features change more smoothly and hence perform better on most occasions. While NNI considers the distribution of surrounding data points, its effect depends on the local structural characteristics of the data. Under conditions of complex data structure or high heterogeneity, NNI may introduce local errors and reduce the model performance instead.

On the other hand, the stacking model is inclined to be more sensitive to the choice of interpolation method due to its complex model structure, inducing errors introduced by NNI, further undermining the model performance. Secondly, the limitations of the interpolation method itself may also be one of the reasons negatively affecting the model performance. SLI is adopted to estimate the missing values through a linear function, which is suitable for data with smooth feature changes and thus performs better in most cases. Such results have been confirmed in several related studies published recently. It has been demonstrated that SLI is also capable of avoiding the overfitting problem that may be introduced by complex interpolation methods, especially when the selected sample size is limited[35]. In contrast, although NNI considers the distribution of surrounding data points, its effectiveness depends on the local structural characteristics of individual data. Overall, the experimental results show that interpolation methods do have a role in improving the performance of machine learning models; however, their effects vary depending on the model type and the choice of interpolation method. SLI is more stable in improving model performance, especially on XGBoost, Random Forest and stacked models. The effect of NNI, on the other hand, tends to be more complicated, and in some cases, may instead lead to a decrease in model performance. Therefore, in practical applications, interpolation approaches must be carefully chosen according to specific model characteristics and particular data properties.

CONCLUSION

In conclusion, this study has explored the effects of simple linear interpolation and natural-neighbourinterpolation on the performance of various machine learning models incorporating Seoul bike-sharing rental data. The results reveal that different machine learning models illustrate varying requirements for data consistency and integrity, with XGBoost and Random Forest models being more sensitive to data completeness. For complex stacking models, the impact of interpolation methods is more pronounced. In particular, SLI generally improves model accuracy, AUC values, and KS statistics

in most cases, with significant performance enhancements observed in stacking models and ensemble methods. This suggests that SLI, as a data pre-processing technique, could efficiently enhance the predictive power of models by improving data consistency and completeness.

However, the impact of NNI on model performance is more complex and may even lead to performance degradation in certain cases. This phenomenon indicates that while NNI can capture local patterns in the data, it may introduce noise or cause local overfitting when dealing with complicated data structures or highly heterogeneous data, thus reducing overall model performance. Therefore, in practical applications, the specific characteristics of the models and the individual properties of the data should be considered when selecting interpolation methods. Future research couldexplore additional interpolation techniques, such as spline interpolation or machine learning-based imputation methods, to further reduce RMSE for highly volatile features like rainfall.

Overall, the experiments in this research demonstrate the important role of interpolation methods in enhancing the performance of machine learning models, though their effects vary depending on the model type and the choice of interpolation method. SLI shows more stable performance improvements, especially in XGBoost, Random Forest, and stacking models, whereas NNI's impact is more complex and may undermine model performance in some cases. Future research could focus on evaluating additional pre-processing techniques to improve the generalisability and performance of models across different domains.

REFERENCES

[1] P. B. Weerakody, K. W. Wong, G. Wang, and W. Ela, "A review of irregular time series data handling with gated recurrent neural networks," *Neurocomputing*, vol. 441, pp. 161–178, Jun. 2021, doi: 10.1016/j.neucom.2021.02.046.

[2] J. Gao, Z. An, and X. Bai, "A new representation method for probability distributions of multimodal and irregular data based on uniform mixture model," *Ann Oper Res*, vol. 311, no. 1, pp. 81–97, Apr. 2022, doi: 10.1007/s10479-019-03236-9.

[3] P. Manousopoulos, V. Drakopoulos, and E. Polyzos, "Financial Time Series Modelling Using Fractal Interpolation Functions," *AppliedMath*, vol. 3, no. 3, pp. 510–524, Jun. 2023, doi: 10.3390/appliedmath3030027.

[4] X. Gong, S. Chen, and C. Jin, "Intelligent reconstruction for spatially irregular seismic data by combining compressed sensing with deep learning," *Front Earth Sci (Lausanne)*, vol. 11, 2023, doi: 10.3389/feart.2023.1299070.

[5] S. N. Shukla and B. M. Marlin, "Multi-Time Attention Networks for Irregularly Sampled Time Series," Jan. 2021, [Online]. Available: http://arxiv.org/abs/2101.10318

[6] A. Cao, B. Stump, and H. DeShon, "High-resolution seismic data regularization and wavefield separation," *Geophys J Int*, vol. 213, no. 1, pp. 684–694, Apr. 2018, doi: 10.1093/gji/ggy009.

[7] H. Huang, M. Chen, and X. Qiao, "GENERATIVE LEARNING FOR FINANCIAL TIME SERIES WITH IRREGULAR AND SCALE-INVARIANT PATTERNS IN MEMORY OF PROF. DUAN LI," 2024.

[8] Zhenyang Guo, "Integrated Prediction of Financial Time Series Data Based on Deep Learning," in *Application of Intelligent Systems in Multi-modal Information Analytics*, vol. 1, 2022, pp. 235–242. doi: https://doi.org/10.1007/978-3-031-05237-8_29.

[9] Richard James and Povinelli, "TIME SERIES DATA MINING: IDENTIFYING TEMPORAL PATTERNS FOR CHARACTERIZATION AND PREDICTION OF TIME SERIES EVENTS," Dec. 1999.

[10] W. Curry, "INTERPOLATION WITH PREDICTION-ERROR FILTERS AND TRAINING DATA," Stanford, CA, USA, 2008.

[11] M. Sambridge, J. Braun, and H. McQueen, "Geophysical parametrization and interpolation of irregular data using natural neighbours," *Geophys J Int*, vol. 122, no. 3, pp. 837–857, 1995, doi: 10.1111/j.1365-246X.1995.tb06841.x.

[12] F. Aldo. Costabile, *Modern umbral calculus : an elementary introduction with applications to linear interpolation and operator approximation theory*. De Gruyter, 2019.

[13] A. Timmer, R. Immel, and L. Harzheim, "The Linear Interpolation Approach (LInA), an approach to speed up the Successive Response Surface Method," *Structural and Multidisciplinary Optimization*, vol. 62, no. 6, pp. 3287–3300, Dec. 2020, doi: 10.1007/s00158-020-02687-0.

[14] G. Goos *et al.*, "Computational Science and Its Applications - ICCSA 2004," 2004.

[15] M. Sambridge, J. Braun, and H. McQueen, "Geophysical parametrization and interpolation of irregular data using natural neighbours," *Geophys J Int*, vol. 122, no. 3, pp. 837–857, 1995, doi: 10.1111/j.1365-246X.1995.tb06841.x.

[16] D. F. Watson, "Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes," Department of Geology and Geophysics, Edgeworth-David Building, 2006.

[17] Mark L. Burkey, Joy Bhadury, and H. A. Eiselt, "Foundations of Location Analysis," vol. 155, Springer, New York, NY, 2011, pp. 445–470. doi: https://doi.org/10.1007/978-1-4419-7572-0_19.

[18] M. Virvou, G. A. Tsihrintzis, and L. C. Jain, "Learning and Analytics in Intelligent Systems 24 Advances in Selected Artificial Intelligence Areas World Outstanding Women in Artificial Intelligence," 2022. [Online]. Available: https://link.springer.com/bookseries/16172

[19] D. Theng and K. K. Bhoyar, "Feature selection techniques for machine learning: a survey of more than two decades of research," Mar. 01, 2024, *Springer Science and Business Media Deutschland GmbH*. doi: 10.1007/s10115-023-02010-5.

[20] B. Engelmann, "Comprehensive Stepwise Selection for Logistic Regression," Jun. 2023, [Online]. Available: http://arxiv.org/abs/2306.04876

[21] D. Elreedy, A. F. Atiya, and F. Kamalov, "A theoretical distribution analysis of synthetic minority oversampling technique (SMOTE) for imbalanced learning," *Mach Learn*, vol. 113, no. 7, pp. 4903–4923, Jul. 2024, doi: 10.1007/s10994-022-06296-4.

[22] I. H. Sarker, "Machine Learning: Algorithms, Real-World Applications and Research Directions," May 01, 2021, *Springer*. doi: 10.1007/s42979-021-00592-x.

[23] G. Ke *et al.*, "LightGBM: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, 2017, pp. 3146–3154.

[24] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.

[25] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann Stat*, vol. 29, no. 5, pp. 1189–1232, 2001.

[26] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. Wiley, 1973.

[27] T. Yu and H. Zhu, "Hyper-Parameter Optimization: A Review of Algorithms and Applications," Mar. 2020, [Online]. Available: http://arxiv.org/abs/2003.05689

[28] O. A. Montesinos López, A. Montesinos López, and J. Crossa, *Multivariate Statistical Machine Learning Methods for Genomic Prediction*. Springer International Publishing, 2022. doi: 10.1007/978-3-030-89010-0.

[29] UC Irvine, "Seoul Bike Sharing Demand," UC Irvine Machine Learning Repository.

[30] R. Silhavy and P. Silhavy, "A Review of Evaluation Metrics in Machine Learning Algorithms.," in *Artificial Intelligence Application in Networks and Systems*, vol. 724, 2023, pp. 15–25. doi: https://doi.org/10.1007/978-3-031-35314-7_2.

[31] M. Owusu-Adjei, J. Ben Hayfron-Acquah, T. Frimpong, and G. Abdul-Salaam, "Imbalanced class distribution and performance evaluation metrics: A systematic review of prediction accuracy for determining model performance in healthcare systems," *PLOS Digital Health*, vol. 2, no. 11, p. e0000290, Nov. 2023, doi: 10.1371/journal.pdig.0000290.

[32] T. Yang and Y. Ying, "AUC Maximization in the Era of Big Data and AI: A Survey," *ACM ComputSurv*, vol. 55, no. 8, Dec. 2022, doi: 10.1145/3554729.

[33] Yadolah Dodge, "Kolmogorov–Smirnov Test," in *The Concise Encyclopedia of Statistics*, 2008, pp. 283–287. doi: https://doi.org/10.1007/978-0-387-32833-1_214.

[34] Z. Cong, L. Chu, Y. Yang, and J. Pei, "Comprehensible counterfactual explanation on Kolmogorov-Smirnov test," in *Proceedings of the VLDB Endowment*, VLDB Endowment, 2021, pp. 1583–1596. doi: 10.14778/3461535.3461546.

[35] W. C. Lin and C. F. Tsai, "Missing value imputation: a review and analysis of the literature (2006–2017)," *ArtifIntell Rev*, vol. 53, no. 2, pp. 1487–1509, Feb. 2020, doi: 10.1007/s10462-019-09709-4.