International Journal on Advanced Science Engineering Information Technology

Vol.13 (2023) No. 1
ISSN: 2088-5334

A Dataset-Driven Parameter Tuning Approach for Enhanced K-Nearest Neighbour Algorithm Performance

Udoinyang G. Inyang [a,b,*], Funebi F. Ijebu [c], Francis B. Osang [d], Adenrele A. Afolorunso [d], Samuel S. Udoh [a,b], Imo J. Eyoh [a,b]

[a] *Department of Computer Science, Faculty of Science, University of Uyo, Nigeria*
[b] *TETFund Centre of Excellence in Computational Intelligence Research, University of Uyo, Nigeria*
[c] *School of Computer Science and Technology, Harbin Institute of Technology, China*
[d] *Department of Computer Science, National Open University of Nigeria, Abuja, Nigeria*
*Corresponding author: *udoinyanginyang@uniuyo.edu.ng*

***Abstract*—The number of Neighbours (k) and distance measure (DM) are widely modified for improved kNN performance. This work investigates the joint effect of these parameters in conjunction with dataset characteristics (DC) on kNN performance. Euclidean; Chebychev; Manhattan; Minkowski; and Filtered distances, eleven k values, and four DC, were systematically selected for the parameter tuning experiments. Each experiment had 20 iterations, 10-fold cross-validation method and thirty-three randomly selected datasets from the UCI repository. From the results, the average root mean squared error of kNN is significantly affected by the type of task ($p<0.05$, 14.53% variability effect), while DC collectively caused 74.54% change in mean RMSE values, k and DM accumulated the least effect of 25.4%. The interaction effect of tuning k, DC, and DM resulted in DM='Minkowski', $3 \leq k \leq 20$, $7 \leq$ target dimension ≤ 9, and sample size (SS) >9000, as optimal performance pattern for classification tasks. For regression problems, the experimental configuration should be $7000 \leq SS \leq 9000$; $4 \leq$ number of attributes ≤ 6, and DM = 'Filtered'. The type of task performed is the most influential kNN performance determinant, followed by DM. The variation in kNN accuracy resulting from changes in k values only occurs by chance, as it does not depict any consistent pattern, while its joint effect of k value with other parameters yielded a statistically insignificant change in mean accuracy ($p>0.5$). As further work, the discovered patterns would serve as the standard reference for comparative analytics of kNN performance with other classification and regression algorithms.**

***Keywords*—kNN; kNN performance; k-Neighbours; parallel analysis; principal component analysis; kNN parameter tuning.**

Manuscript received 21 Nov. 2021; revised 3 Jun. 2022; accepted 8 Nov. 2022. Date of publication 28 Feb. 2023.
IJASEIT is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.

I. INTRODUCTION

K-Nearest Neighbor (kNN) is known to be an extensively applied algorithm in diverse areas, including data science, data mining, and machine learning (ML) research [1]–[4]. Its wide acceptability is largely due to simplicity [5], [6], usability, intuitiveness, flexibility, and applicability [1], [7]. It accepts continuous, discrete, ordinal, and categorical data and is exceptional for handling all types of missing data [5]. kNN models are remarkably successful in many fields, including pattern recognition [8], biometrics [9], text categorization, outlier detection, and collaborative filtering [6], [10]. Since a linearly separable target is not a requirement, it is very suitable for classification [11], [12], and regression problems [13]–[15].

It is fast during the training phase but computationally expensive while estimating the optimal number of nearest neighbors. Most worrisome is the relatively low accuracy when compared to other ML approaches [1], [7], [16]. This challenge is mainly linked to some critical issues—parameter configuration (choice of k value and distance measures), and pre-processing data requirements [11], and has caused the gradual decline in its popularity. Another key determinant of kNN performance is the nature of the dataset [17]. Sample size, input feature dimensionality, target class dimensionality, missing values, and ML problem types generate significant prediction errors [17], [18].

A widely adopted performance improvement methodology for kNN is parameter tuning. This involves running multiple times with varying parameter values (k or distance), and the parameter-value producing the best performance accuracy is chosen. This approach is computationally expensive and

produces results that are not reliable due to the exclusion of dataset structure. Some previous studies investigated the effect of parameters on kNN performance individually, and optimal parameters were estimated for the applied domain[2], [19]–[21]. These optimal parameters differ from one problem to another, even for solutions in the same domain.

Considering that kNN performance depends on the model parameters and dataset characteristics, there ought to be a viable reference for an optimal decision on the configuration of these determinants for optimal results. However, the joint effect of both metrics of kNN vis-à-vis the dataset's characteristics is not reported in the literature. To fill this gap, this work investigates the combined effect of kNN parameters on performance. It estimates the desirable kNN parameter configuration under varying dataset characteristics (feature dimensionality, class dimensionality, type of ML tasks, sample size), using some selected research datasets published in the University of California, Irvine (UCI) ML repository [22]. Moreso, the influence of the interactions among the dataset properties on kNN accuracy will also be investigated by learning the optimal parameters for each dataset sample. This work would therefore make the optimal parameters of kNN handy for adoption in any given structure of the datasets as its contributions to knowledge.

II. Materials and Methods

This section is divided into five sub-sections. The related works are summarized in sub-section A, while the data-driven analytic framework is described in sub-section B. The series of the experimental setup is presented in sub-section C. Sub-section D presents the effect of patterns produced from parameters interaction.

A. Related Works

This section discusses related works on kNN algorithm, the effect of each parameter, and dataset characteristics on kNN Performance.

1) kNN Algorithm: KNN is a non-parametric supervised ML algorithm often regarded as a lazy learner because it does not learn any discriminative function from the training data; rather, it memorizes the training dataset and utilizes k number of related instances to predict the class of a new data [11], [23]–[26]. Initially proposed for the classification of numeric data, research over the years has led to its further successful application in regression tasks and in predicting other non-numeric data types. Although considered a simple algorithm because of its underlying operational principles, its performance in most classification and regression problems has been seen to surpass other supervised ML algorithms [18], [27], [28].

To improve the performance of the kNN algorithm, several variations have been proposed and applied in different domains. From the kNN graph neural network (kNNGNN) [28]; to the PK Means++ (kNNPK++) [27] and the Particle Optimized scored kNN (POSkNN) [29], or the neuro-fuzzy kNN method and the K-dimensional tree (kNN-KD-tree) by [25] the algorithm has continued to use hyper-parameters (k Neighbours, distance measure, and weighting function) for performance determination and enhancement. Enhanced adaptive kNN using simulated *AneaAnneais lingposed* [18], [30] and a modified model [30]. More description of kNN parameters and applications [31]–[35]

2) Effect of Distance Metric on Performance: The performance of the kNN is dependent on the adopted distance measure [36]. kNN was only initially utilized with traditional metrics like Euclidean and Manhattan distance measures. Recently, advances in research have led to its adaptation for the classification of non-numeric data [37]. Some studies [38]–[40] proposed a time-saving memory-efficient variant of kNN algorithm using heterogeneous Euclidean-overlap metric – a modified version of the conventional Euclidean distance measure. The algorithm was trained and tested using categorical data to reduce execution time through dataset compression. The results show that the accuracy of data classification in the compressed and uncompressed datasets was almost the same, while classification time was noticeably reduced with the former.

Furthermore, several reports proposed and selected different distance measures with single or multiple datasets and presented varying results in terms of the most efficient distance measure with kNN. Consequently, the question of which distance measure to adopt for the kNN classier has been asked and investigated [20], leading to the submission that the performance of kNN significantly depends on the distance measure. In the study, fifty-four (54) distance measures were selectively reviewed, out of which the Hassanat distance metric was observed to have the best average performance.

An investigation into the performance of kNN on six heterogeneous datasets of non-numeric binary data types using Euclidean, Manhattan, Cosine, Jaccard, and Canberra DMs [21]. K values comprised odd numbers between 1 and 9; the feature weights of 0.8 and 0.6 were assigned to the most important and least important feature sets, respectively, while the feature set with equal importance weighted 0.5. Their results showcased Euclidean distance as the best. Although their results showed that the optimal value of k for most datasets under the different weight assumptions was one, the reason for the enhanced performance is yet to be established.

Similarly, kNN performance evaluation with electroencephalogram data comparatively studied Manhattan, Euclidean, Minkowski, Chebychev, and Hamming [41] distance measures. Accuracy is one of the key performance metrics in ML; it was used for evaluation and returned a 70.08% score for the Minkowski distance measure, which was observed to be the highest. With kNN, the value of k is dependent on the data under investigation, while the distance measure affects the classification result. In addition to a large variation of k-values, the values at which the highest accuracy scores are returned are large compared to the seemingly moderate accuracy scores observed [41].

In another comparative study to examine the distance measure effect on kNN performance; the Euclidean, Cosine, Chi-square, and Minkowski distance measures, among others; were investigated [42], utilizing heterogeneous datasets for the kNN performance evaluation. From their results, the Chi-Square measure produced the highest accuracy in all three medical-related datasets against the expected Euclidian distance. Another study [7] proposed a combination of the low mean-based kNN and distance weight kNN technique for improved classification accuracy using Euclidean distance.

Earlier, Parvin, Alizadeh, and Minati [43] proposed a modified kNN algorithm with Euclidian distance as the preferred distance measure.

The Euclidean distance appears to be the most patronized distance measure in the literature [42]–[44]. However, it is less effective when high classification accuracies are a priority, as observable in some studies [20], [42], [44]. Ehsani and Drabløs [24] found out of the twelve distance functions adopted for kNN performance evaluation, Sobolev and Fisher were outstanding in several cancer datasets in the study. A similar report is tenable; the Manhattan distance measure returned the highest accuracy ahead of the Euclidean and Chebyshev measures for classification tasks. In Chomboon et al. [45], eleven distance functions were considered; Minkowski, Mahalanobis, Euclidian, Cosine Similarity, Manhattan, Chebyshev, Correlation, Hamming, Jaccard, Standardized Euclidian, and Spearman. Although no single pole performing distance measure on eight binary synthetic datasets was given, their investigations showed that the Manhattan, Minkowski, Chebyshev, Euclidian, Mahanathan, and Standardized Euclidean distance measures outperformed others with relatively similar accuracy scores.

Furthermore, using the Manhattan, Euclidian, Soergel, Lance-Williams, contracted Jaccard-Tanimoto, Bhattacharyya, Lagrange, Mahalanobis and four of its variants, Canberra, Wave-Edge, Clark, Cosine, and Correlation distance measures with kNN on eight different datasets; Manhattan, Euclidian, Soergel, Lance-Williams and contracted Jaccard-Tanimoto returned the highest accuracy score. Some studies [46]–[49] have insights into the relationship between distance measures and kNN performance

3) Effect of K-Neighbors on kNN Performance: The performance of several proposed kNN models using various domain-specific datasets depends on the distance measure and k values [15] adopted. Nevertheless, no known domain-specific or generally acceptable range of k values can be used with specific distance measures or particular study scenarios to obtain optimal results when the underlying algorithm in the study is known. As such, the literature has a huge variation of k values. In most cases, specific k values have been observed to produce the best performance. There is no justifiable explanation of why one k value performs better or less than the other factors that may be involved. However, without proper investigations, there will be no standardized criteria for selecting k values in kNN kNN-based investigations.

The sensitivity of kNN to chosen k values is a challenging characteristic of the algorithm [50], which is an area of interest in this work. In k-NN, the k value represents the number of nearest Neighbors; hence the value is a core deciding factor for the classifier's performance. In several works, k-neighbors varied over different ranges also forces a variation in the accuracy of the kNN model. In Syaliman, Nababan, and Sitompul [7], optimal classification was obtained with k=10 in an experiment involving k in the range $1 \leq k \leq 10$. In a similar experiment, for k-values ranging from $1 \leq k \leq 20$, k=8 produced the best result.

Investigating the efficiency of kNN in classifying electroencephalogram data, Md Isa and collaborators [41] iterated k in the range $1 \leq k \leq 15$. This k value range is not very different from that adopted in Parvin, Alizadeh, and Minati [43], where a modified kNN algorithm is proposed with k values is $3 \leq k \leq 15$. Considering the range of the chosen k values in these studies, we note that the value at which the models performed best was a relatively high k value. A distinctively large range of k values is seen in [51], where $1 \leq k \leq 133$. Granted that large k values with different datasets returned high accuracies, this is not always the case because; in Hu et al. [42], even when k values ranged $1 \leq k \leq 15$, the highest accuracy of 78.8% was returned when k = 4, followed by accuracies of 71.9% and 76.5% for two different datasets respectively when k = 1.

Ali, Neagu, and Trundle [21] chose a relatively smaller range of k values consisting of only the odd numbers between 1 and 9, and had the best performance when k =1. Consequently, these results show conditions of high accuracy with small/low k values and correlate with a special case of kNN known as the nearest Neighbor.

K values in the range $k = \{3,5,7\}$ were chosen to compare the efficiency of the traditional kNN model using Euclidian distance with variant ensemble clustering kNN algorithm. Even though the performance of the modified kNN was better than the traditional algorithm, the reason for the choice of k and optimal runtime at k = 5 concerning the other k values is not reported. Furthermore, after experimenting with eight different k values ($k = \{1,3,5,11,21,31,41,51\}$), to arrive at the conclusion that near-zero k values do not always suit small datasets, while large k values are also unsuitable for the huge dataset. Concerning large k values, [51] thinks that they are less sensitive to noise and make the boundary between different classes of a dataset smoother.

Moreso, even for datasets from the same domain with similar characteristics, the best-performing k-values are not uniform. The submission of Isa et al. [41] and Hassan et al. [52] contains non-uniform classification results even though a similar dataset was used in both studies. Additionally, considering the Wisconsin Breast Cancer (WBC) dataset adopted by Ehsani and Drabløs [24] and Iswanto, Tulus, and Sihombing [39], the selected k values ranged from 1 to 20 in the former, while the latter chose four k values between 5 and 20 in steps of 5. While previous studies reported the highest accuracies without pointing to a specific value of k; the same dataset yielded the best performance [24] when k = 5. Even though both studies adopted the same dataset, there is a disparity in the k values at which kNN performed best. Even though this disparity is associated with the choice of distance measures, in both investigations [51], there is still the dilemma of what distance measure and k value to combine with a dataset to obtain optimal kNN performance.

The simplest way to choose a k value is to iteratively run the kNN model a number of times and choose the k value associated with the best performance. This has been the take of many researchers, as evident in the literature. This position, however, compounds and increases the time cost for an investigation and model resolution since the range of possible k values that can be tested for optimality determination is infinite.

To the best of our knowledge, the lack of standardized benchmarks for k value selection has left researchers with no criteria for selecting the smallest and largest values of k. Being that static k values are discouraged [53], [54], k-values

that make the range of a study are an exclusive choice of the author, and in many cases, the individual guiding benchmarks for k value selection are not reported. Thus, one can opine that a trial-by-error method is adopted in most studies for the set of k values to be used in the model testing.

Although another study [55] presented experimental results on what data properties affect the choice of k-value; it did not incorporate what data properties or features should be considered in selecting an appropriate distance measure. Since the k value and the distance measure are core determinants of the algorithm's classification efficiency, we identify this subject as requiring research attention.

4) Effect of Dataset Characteristics on kNN Performance: Several empirical ML and data mining studies have revealed that the performance of classifiers depends on the dataset employed and the parameters of individual classifiers. There is no k value or distance measure of kNN that is the best in all dataset situations [51], [55], [56]. In two other studies [51], [53], the significance of the impact of dataset characteristics on the classifier's performance is acknowledged and exploited to study the performance of kNN algorithm. The study confirmed that higher k values are suitable for 2-class datasets with relatively high sample sizes while a different relationship is depicted with $n > 2$ −class datasets.

An investigation of how dataset characteristics and other performance metrics collectively or individually affect the efficiency of an algorithm and the impact level of each criterion [56]. Results of experiments confirmed differing ranks of five (5) distance measures using lung, prostate, and breast cancer datasets. Therefore, understanding the relationship between dataset characteristics and kNN parameter is crucial to the performance of kNN. Sample size or number of instances, sparsity, uneven density, missing values, data format, class dimensionality or distribution, and domain area are some dataset particulars that affect model performance [57].

Therefore, there is no best algorithm for all dataset situations using performance metrics [17], [58], but there should be best algorithms for specific datasets. While most kNN research focuses on finding the optimal k values, others investigate the best distance metric or the effect of dataset parameters. No previous research attempts to simultaneously investigate the optimal k value and best distance metric in specific data situations. Therefore, to fill this gap, this work investigates the optimal k value, the best distance metric concerning specific data scenarios using thirty-three UCL datasets.

B. Dataset Driven Analytic Methodology

The dataset-driven analytic approach (Fig. 1) phases through the following series of activities:

- dataset collection and characterization;
- 10-fold cross-validation-based experiments;
- Input rank analysis;
- Interaction effect computations and tests;
- Rule-set formulation.

Datasets published in the UCI Repository were randomly selected for the experiments [59]. The UCI repository manages several benchmarking datasets organized for empirical performance analysis of algorithms and gives descriptions and papers associated with each dataset [11], [22], [55]. Thirty-three (33) datasets were retrieved and used for the analytic experiments, out of which thirteen (13) are regression and twenty (20) classification tasks.

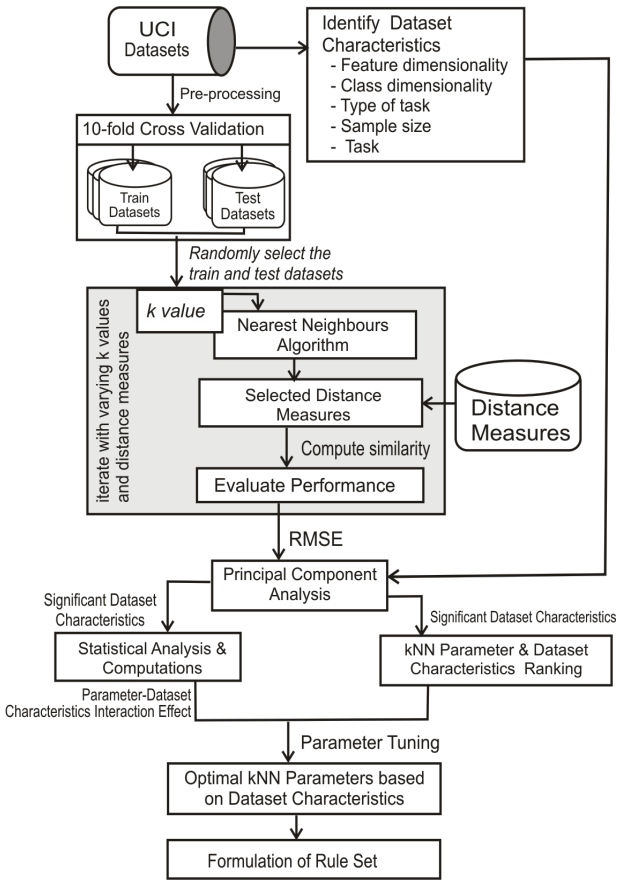


Fig. 1 Dataset-Driven Analytic Workflow

A parametric description of the datasets (Table 1) has the number of attributes (NOA), associated ML task (TA), sample size (SS), Domain Area (DA), and Attribute Format (AF) of each dataset. NOA ranges from 4 - 38; SS ranges from 23-45730; TD has values between 0-29. SA includes life, financial, computer, physical and social, while AF was grouped into real, integer, categorical, and mixed formats. AF is "mixed" when more than one data format is involved. Categorical attributes remained unchanged, while the continuous ones (input dimensionality, sample size, and target class dimensionality) were broken down into subgroups to allow for groupings of specific data situations. This will provide a standardized criterion for adoption when referencing datasets; and eases the interpretation of results

C. Experimental Analytics

A series of experiments were set up for each dataset in the Waikato Environment for Knowledge Analysis (WEKA) experiment environment [1]. The default parameters of kNN algorithm were used apart from varying the number of neighbors (k) and distance measures. Before running the experiments, all datasets were pre-processed by removing irrelevant features (for example, ID, dates etc.) and records with missing values.

The experiment was performed with eleven k values; k = {1,3,5,7,10,13,15,17,20,25,30} neighbors and five distance measures; Chebyshev (CH), Euclidean (EU), Manhattan (MA), Minikowski (MI) and filtered (FI) distance metrics. Each experiment was repeated 20 times with 10-fold cross-validation (10-FCV). Extensive tests on different datasets with different classifier schemes confirmed the suitability of ten (10) as the number of folds to get the best estimates of accuracy. The 10-FCV results were averaged (with standard deviation), producing a single result for each k and distance measure combination.

Root Mean Squared Error (RMSE) was the performance metric used for the analytic evaluation. The justification for the adoption of RMSE is its wide acceptability as a measure of accuracy in comparing the predictive errors of different estimation models/model configurations [60]. Each dataset produced 55 results (average RMSE ± standard derivation), giving a total of 1,815 data points

The k values, distance measure, and the resultant RMSE, were columns added to the corresponding dataset properties described in Table 1. The resultant performance dataset had eight (8) input features and one target variable. Table 2 shows the structure of the dataset for *k=1* and Euclidean distance metric for the Solar Flare dataset.

Dimension reduction maps data to a lower-dimensional space to discover and extract explanatory variance in the dataset, or such that a sub-space that comprises the data is known [61], [62]. An interesting projection method for dimension reduction that is more adept at preserving the global structure of data is the Principal Component Analysis (PCA) [63], [64].

TABLE I
SUMMARY OF DATASET PROPERTIES

S/N	Dataset Name	No. of Attributes (NOA)	Tasks (TA)	Sample Size (SS)	No. of Target Class (TD)	Domain Area (DA)	Attribute Format (AF)
1	Abalone	8	1	4177	29	Life	Mixed
2	Audit Data	18	1	77	2	Financial	Real
3	Contraceptive Method Choice	9	1	1473	3	life	Categorical
4	Glass identification	10	1	214	7	Physical	Real
5	Letter recognition	16	1	20000	26	Computer	Integer
6	Mushroom	22	1	8124	2	Life	Categorical
7	Nursery	8	1	12960	5	Social	Categorical
8	Processed Cleveland	13	1	303	5	Life	Mixed
9	Teaching assistant	5	1	151	3	None	Categorical
10	Water treatment	38	0	527	0	Physical	Real
11	Wine	13	1	178	3	Physical	mixed
12	Breast Cancer	32	1	569	2	life	Real
13	Credit Approval	14	1	690	2	Financial	Mixed
14	Computer Hardware	9	0	209	0	Computer	Integer
15	Dermatology	33	1	366	6	Life	Mixed
16	Ecoli	8	1	336	8	Life	Real
17	Flags	29	1	194	8	None	mixed
18	Haberman's Survival	3	1	306	2	Life	Integer
19	Hepatitis	19	1	155	2	Life	mixed
20	Energy Efficiency	8	0	768	0	Computer	mixed
21	Solar Flare	10	0	1389	0	Physical	Categorical
22	Seoul Bike Sharing Demand	12	0	8760	0	Computer	Mixed
23	Monks Problem	7	1	432	2	none	Categorical
24	Challenger USA	4	0	23	0	Physical	Integer
25	Forest fires	13	0	517	0	Life	Real
26	Algerian Forest Fires	11	1	244	2	Life	Real
27	Dry Bean Dataset	16	1	13611	7	Computer	mixed
28	Servo	4	0	167	0	Computer	Mixed
29	Yacht Hydrodynamics	6	0	308	0	Physical	Real
30	Concrete Compressive Strength	8	0	1030	0	Physical	Real
31	Physiochemical	9	0	45730	0	Life	Real
32	Frogs MFCC	22	0	7195	0	Life	Real
33	Bike Sharing Count	13	0	732	0	Social	Mixed

TABLE II
RMSE VALUES FOR K=1 AND EUCLIDEAN DISTANCE COMBINATION FOR SOLAR FLARE DATASET

Dataset Name	NOA	TA	SS	DA	AF	k	DM	RMSE
Solar Flare	≥10	Reg	1001-1500	Phy	Cat	1	CH	0.15±0.09
Solar Flare	≥10	Reg	1001-1500	Phy	Cat	1	EU	0.17±0.10
Solar Flare	≥10	Reg	1001-1500	Phy	Cat	1	MA	0.17±0.11
Solar Flare	≥10	Reg	1001-1500	Phy	Cat	1	MI	0.17±0.09
Solar Flare	≥10	Reg	1001-1500	Phy	Cat	1	FI	0.16±0.12

It minimizes the dimensionality of sizable datasets; by transforming a large set of variables into a smaller one that still retain a significant proportion of information in the original dataset[63], [65]. PCA was carried out in the following steps: i) Standardization of continuous variables to fall between 0 and 1, to ensure equal contribution to the

analysis. ii) computation of the covariance matrix for highly correlated variables identification. iii) Acquisition of eigenvalues and computation of the eigenvectors. iv) ordering of eigenvalues in descending order for principal components in order of significance. v) choice of eigenvectors that correspond to the largest eigenvalues. vi) selection of the total number of relevant factors.

Meaningful principal factors were selected based on the identification of factors earning eigenvalues greater or equal to unity (1) [61], [65], [66] and were retained through the Parallel Analysis (PA) approach [67] via the Monte Carlo Protocol [67] in 1000 repetitions. The percentile intervals (at 95% confidence level) for each artificial eigenvalue were estimated and used as standard critical values for assessing actual eigenvalues. The results (Table 3) revealed eigenvalues of the dataset properties as individually greater than those of the kNN parameters except for SS, which earned the least eigenvalue of 0.0727.

TABLE III
EIGENVALUES AND FACTOR LOADINGS OF FEATURES

Components	Actual Eigenvalues	PA Eigenvalues	95% PA Eigenvalues	Prop. of Variance (Actual)	Cumulative Variance (Actual)
TD	2.13	1.088	1.118	0.304	0.3040
TA	1.02	1.053	1.074	0.145	0.4493
NOA	1.00	1.025	1.004	0.1429	0.5922
AF	0.99	0.999	1.016	0.1428	0.7350
KV	0.93	0.974	0.991	0.1338	0.8689
DM	0.85	0.947	0.977	0.1207	0.9896
SS	0.07	0.914	0.941	0.0104	1.00

The dataset parameters collectively caused 74.54% variability of the RMSE values, while KV and DM accumulated 25.4% variance (Fig. 2). The first two actual eigenvalues are greater than their PA counterparts (for both the mean and 95th percentile criteria) and thus would be retained. As evidenced in the PA values, there is a seeming indication of a third factor because the eigenvalue is almost the same as the randomized average and 95th quartile eigenvalues, but only two factors show up in the graphical representation (Fig. 2) — the PA values plot crosses the actual eigenvalues line right at two factors.

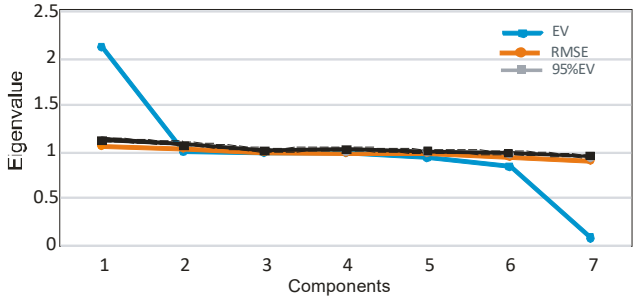


Fig. 2 Plot of actual versus average randomly generated eigenvalues

The results confirmed two significant components – TD and TA. Since a single TA is performed at a time, the dataset was partitioned into two based on TA, regression, and classification datasets. The PCA experiment was repeated for each class of dataset; the results are presented in Tables 4 and 5, respectively. The proportion of variance explained by individual components showed that the KNN parameters together contributed 20.76% variability, while the dataset properties took 79.24% for the classification task.

TABLE IV
EIGENVALUES AND FACTOR LOADINGS FOR REGRESSION TASK

Components	Actual EV	PA EV (Average)	95% PA EV	Prop. of Variance (Actual)	Cumm. Variance (Actua)
AF	1.128	1.103	1.156	0.2256	0.2256
NOA	1.030	1.045	1.081	0.2061	0.4317
SS	1.002	0.999	1.028	0.2006	0.6322
KV	0.997	0.953	0.983	0.1994	0.8317
DM	0.841	0.898	0.935	0.1683	1.00

TABLE V
EIGENVALUES AND FACTOR LOADINGS FOR CLASSIFICATION TASK

Components	Actual EV	PA EV (mean)	95% PA EV	Prop. of Vari. (Actual)	Cumm. Var. (Actual)
SS	1.46	1.0994	1.137	0.2439	0.243
TD	1.29	1.0528	1.079	0.2151	0.459
AF	1.00	1.0160	1.038	0.1670	0.626
NOA	0.998	0.9816	1.003	0.1664	0.792
DM	0.721	0.9471	0.969	0.1202	0.912
KV	0.524	0.9028	0.934	0.0874	1.000

In the regression task (table 4), the situation improved for the KNN parameters (36.88%), with KV having a proportion of 19.94%. Figs. 3-4 visualized the individual and cumulative explained variance for each component.

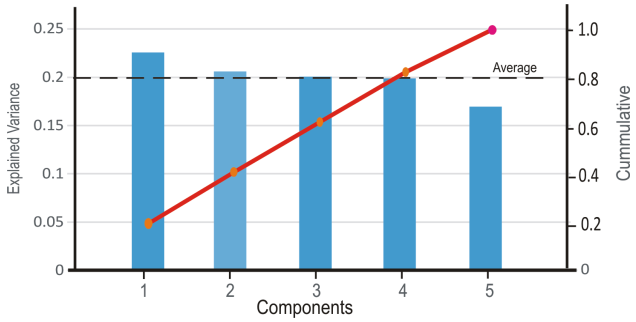


Fig. 3 Regression Scree plot of explained variance across components

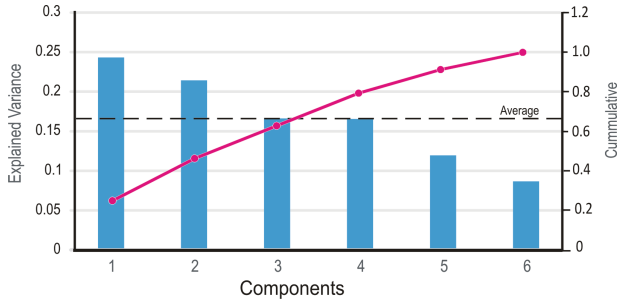


Fig. 4 Classification Scree plot of explained variance across components

The explained variance depicts different patterns; — the second, third and fourth factors explained virtually the same variance in the regression dataset, while a majority of the factors in the classification task were disproportionately distributed. The variation in the factor-distribution pattern in both tasks confirmed the significant effect of ML task on the performance of KNN.

On the decision of the principal factors to retain, three components have actual eigenvalues greater or equal to 1 in both classification and regression tasks. However, the PA scree plot for regression datasets (Fig. 5) approves three significant factors (AF, NOA, SS) for retention. In Fig. 6, the actual eigenvalue plot meets the PA eigenvalue plot before getting to the third factor, thereby further affirming the presence of two principal factors of SS and TD.

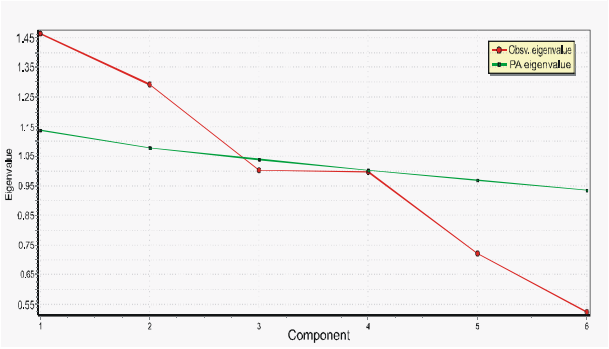


Fig. 5 Interaction of actual and PA eigenvalues for classification Task

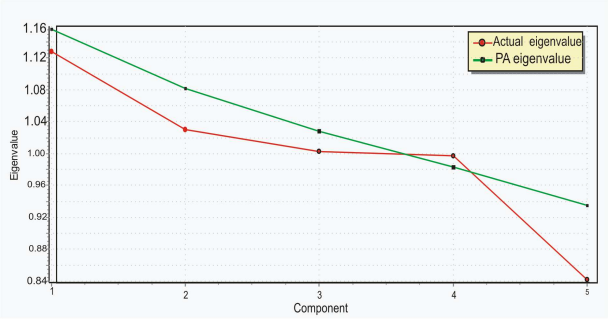


Fig. 6 Interaction of actual and PA eigenvalues for Regression Task

The biplot visual (Fig. 7) shows the original features as vectors in the plane formed by two principal components. They commenced at the origin [0,0] [0,0] and extended to coordinates based on the factor loadings. The angles between vectors of the different variables showed that SS and TD had a high positive correlation, while NOA and AF depicted a negative correlation for the classification task.

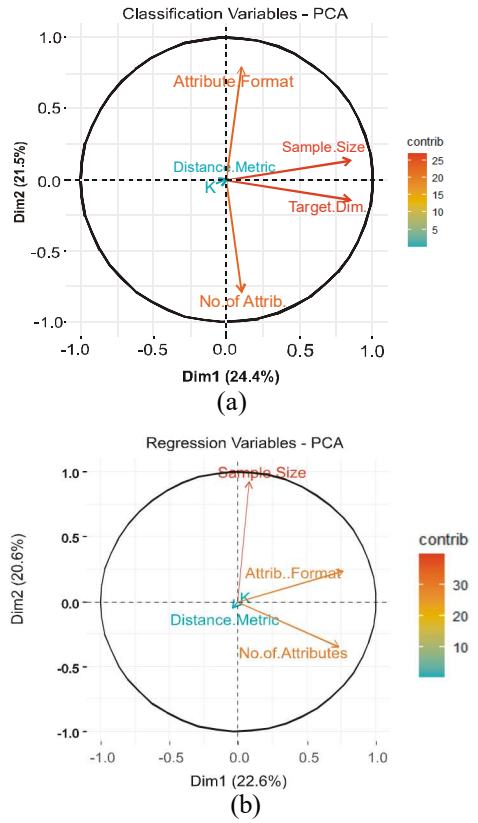


Fig. 7 (a-b) BIPLOT showing the relationship of features along two principal components

For regression datasets, NOA and AF were highly positively correlated, although a near lack of correlation was noticed between AF and SS. In both regression and classification tasks, k and DM correlated negatively with other features as they lied in the opposite direction with an insignificant proportion of explained variance, as shown by their length.

D. Parameter Interaction Effect Pattern

The relationship between the kNN performance and the principal factors in classification and regression tasks was modeled by fitting the generalized linear models to examine the factors' combined effects on RMSE. As shown in Table 6 and Fig. 8, the results indicate a statistically significant difference in the mean performance of kNN across TD, SS, DM, and k-neighbors for the classification task and across SS, NOA, AF, and DM for regression tasks ($p<0.0000$), at 95% confidence level. This implies that changing each factor level causes a significant variation in the RMSE value.

Whereas there exists a statistically significant direct impact on k-neighbors on classification performance ($F=2.89$, $p=0.002$), the reverse is the case regarding regression task ($F=0.14$, $p=0.999$).

TABLE VI
INTERACTION EFFECT OF PRINCIPAL FACTORS AND KNN PARAMETERS FOR CLASSIFICATION TASK

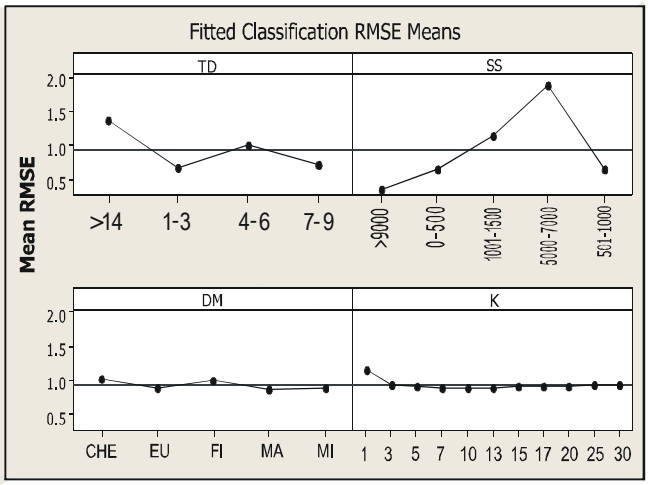
Factors	DF	Adj MS	F	P
TD	3	9.0372	117.42	**0.000**
SS	4	22.5295	292.73	**0.000**
DM	4	0.4576	5.95	**0.000**
K	10	0.2226	2.89	**0.002**
DM*K	40	0.0050	0.07	1.000
SS*K	40	0.0447	0.58	0.983
TD*K	30	0.0121	0.16	1.000
SS*DM	16	0.0486	0.63	0.859
TD*DM	**12**	**0.2483**	**3.23**	**0.000**
SS*DM*K	160	0.0170	0.22	1.000
TD*DM*K	120	0.0013	0.02	1.000
Error	**661**	**0.0770**		
Total	**1100**			

TABLE VII
INTERACTION EFFECT OF PRINCIPAL FACTORS AND KNN PARAMETERS FOR REGRESSION TASK

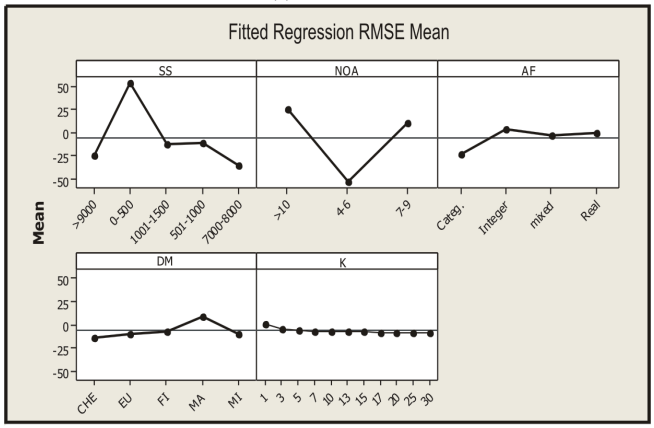
Factors	DF	Adj MS	F	P
SS	**4**	**31061.6**	**57.07**	**0.000**
NOA	**2**	**63072.6**	**115.88**	**0.000**
AF	**3**	**4698.0**	**8.63**	**0.000**
DM	**4**	**2230.2**	**4.10**	**0.003**
K	10	74.8	0.14	0.999
DM*K	40	8.2	0.02	1.000
AF*K	30	4.5	0.01	1.000
NOA*K	20	329.9	0.61	0.904
SS*K	40	143.2	0.26	1.000
AF*DM	**12**	**1135.3**	**2.09**	**0.020**
NOA*DM	**8**	**3352.0**	**6.16**	**0.000**
SS*DM	**16**	**1093.6**	**2.01**	**0.015**
AF*DM*K	120	3.7	0.01	1.000
NOA*DM*K	80	19.1	0.04	1.000
SS*DM*K	160	7.8	0.01	1.000
Error	**164**	**544.3**		
Total	**713**			

Fig. 8a reveals that TD>14 (mean=1.37), SS of 5000 -7000 (mean=1.89), CHE (mean=1.03), and k=1 (mean =1.03) are associated with worse kNN performance in their respective categories for classification dataset. However, regression task results depict k=30 (mean=-7.71), CHE (mean=-13.67), SS between 7000-8000 (mean=-34.43), NOA in the range 4-6 (mean=-53.57), and Categorical AF (mean=-23.88) are associated with the best kNN performance.

1) DM Relationship Pattern: Multiple comparisons using Bonferroni test on the RMSE reveal that TD of "1-3" and "7-9" produces statistically insignificant mean difference and produces a significantly least mean RMSE values (Mean=0.7) while the class ">14" (Mean= 1.4) depicts the highest significant difference in RMSE means. As shown in Table 7, the relationship between RMSE and the other variables does not make statistically dependent on the value of k ($p>0.05$) in both tasks. However, for classification, the impact of TD on kNN performance significantly relies on the level of DM ($F=3.23$, $p=0.000$), while the combined effect between SS and other factors displayed no statistically significant evidence of variation. A statistically significant joint effect was noticed with AF*DM ($F=2.09$, $p=0.020$), NOA*DM ($F=6.16$, $p=0.000$), and SS*DM ($F=2.01$, $p=0.015$) regarding regression task.



(a) Classification



(b) Regression

Fig. 8 Direct Effect of Factors of kNN performance a) Classification b) Regression

Fig. 9a shows a statistically significant difference in the means between levels of both DM and TD. The highest mean difference of RMSE is noticed with TD ">14" across all DMs followed by "4-6". Although the mean difference for TD of 7-9 and 1-3, respectively, are statistically insignificant across DM levels, 7-9 class earned the least RMSE in all DMs followed by 1-3. It follows that when the TD is between 1-3 or 7-9, DMs produce statistically equivalent RMSE values. That is, a change in the distance measure does not produce a significantly different RMSE value.

However, for TD between 4-6, RMSE values associated with EU, MA, and MI metrics differ insignificantly among themselves but depict a significant variation from RMSE values related to CHE ($p=0.001$) and FI ($p=0.000$) metrics and produce relatively low RMSE values. TD > 14 produces an average RMSE that differs significantly from CHE and FI distance measures, while the performances are relatively better and similar for EU, MA, and MI DMs.

Factor interaction for regression task (fig. 9b-d), shows significant variation in the means of RMSE across DMs. Fig. 9b describes the interaction of NOA and DM in the regression scenario. The three levels of NOA (4-6, 7-9, >10) exhibited diverse patterns across DMs. Worse performance was noticed with CHE in conjunction with "7-9" class, followed by MA with "4-6". The RMSE values resulting from DMs for NOA>10 showed no statistically significant difference, even when they expressed relatively mid-range performance. MA produced the worse performance for datasets with 4-6 NOA, while other DMs exhibited statistically equivalent performance.

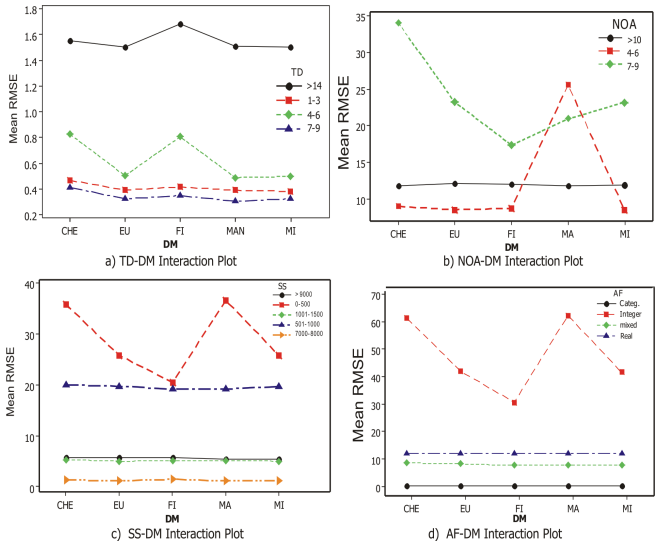


Fig. 9 Visualized feature interaction Pattern. a) TD-DM, b) NOA-DM, c) SS-DM, d) AF-DM

On the strength of the relationship of DM on kNN performance, Fig. 9(c) reveals a significant variation dependency on DM ($F=16$, $p=0.015$). This was noticed in the 0-500 dataset group, with CHE and MA producing the highest mean difference while FI metric earned the least mean RMSE in this category. In the other SS groups, the differences do not change significantly across DMs, although higher values are exhibited by $501 \leq SS \leq 1000$ followed by $1000 \leq SS \leq 1500$, and levels >7000, respectively.

Fig. 9(d) and Table 6 show that DM–RMSE relationship significantly depends on the diversification of AF levels ($F=2.09$, $p=0.020$). Although results from categorical, mixed

and real AF individually do not deviate significantly across DMs, categorical attributes performed best, followed by mixed and real attribute formats. For integer AF, there is a substantial variation of performances, with CHE giving the worse RMSE while FI yielded the lowest RMSE values.

(c) K-Relationship Pattern Analysis: The pattern of the relationship between k and other principal factors in both classification and regression problems exhibits no meaningful variance ($p>0.05$) at 95% level (Table 7). The behavior of RMSE, due to varying levels of k in conjunction with other features for classification datasets (Figure 10), shows relatively high RMSE for k=1 in all dataset properties. For K-DM interaction, there is a steady marginal rise in RMSE for k neighbors>3 in all DMs. A similar pattern is displayed by k-TD effect, except TD between "1-3" where there is a slight improvement in performance up to *k=30*. This implies that *k=3* produces the best results when the DMs levels vary. An opposite trend was observed for the k-SS relationship, where RMSE is almost the same as k increases in all SS categories. As shown in figure 10a-c, although, no reasonable difference in RMSE is noticed with the variation of k levels, any value of $10\leq k\geq 3$ would be optimally suitable for classification problems.

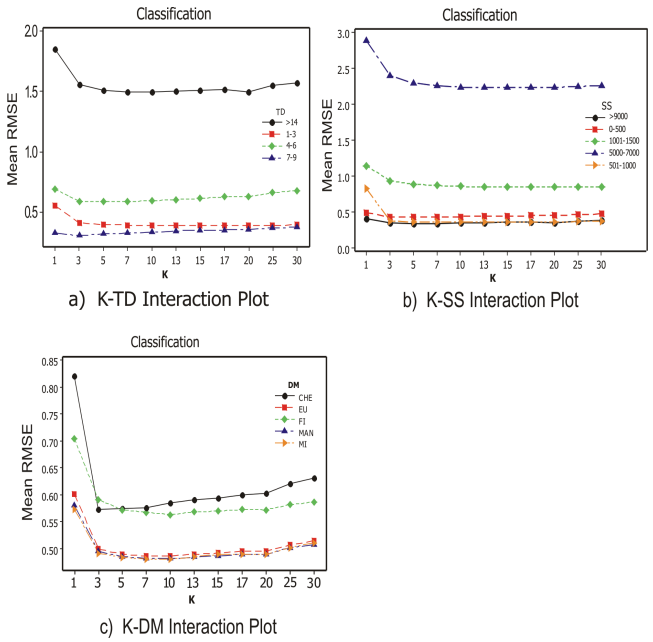


Fig. 10 Plot of k interaction pattern in the classification task

Regarding regression problems (Fig.11), the impact of k-neighbors on the mean performance produces a relatively lowest RMSE at k=1 for all DMs except CHE. The trend also depicts a gradual insignificant rise in mean RMSE for all DMs except MA, which is near uniform RMSE values for levels of K>7. An SS ≤500 produces significantly the highest mean RMSE values for levels of K>1, followed by $501\leq SS\leq 1000$ while SS in the range $7000\leq SS\leq 8000$ is the best-performing class. Since the evaluation of statistical significance produced $p>0.05$ for all categories, the difference in means induced by the factor levels may only occur by chance.

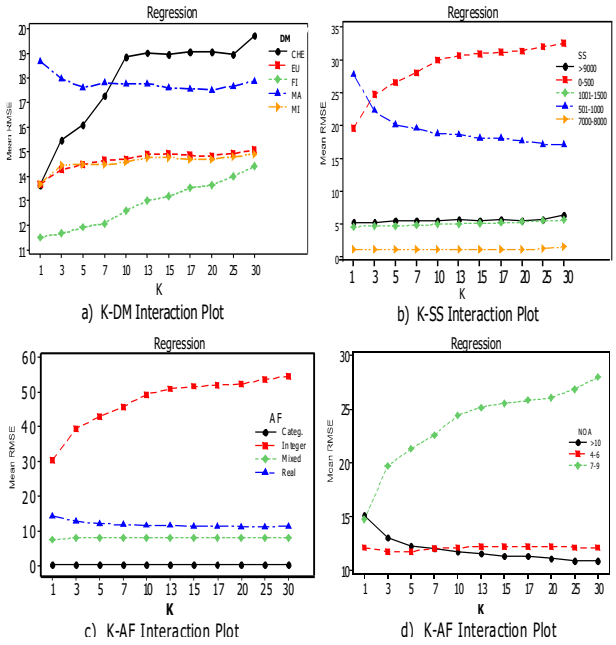


Fig. 11 Plot of k interaction pattern in the regression task

III. Results and Discussion

In some previous studies highlighted in this work, the range of k neighbors selected is arbitrary and large, with the difference between the k values hardly well defined. However, since the choice of k values are an exclusive decision of Ma and Zhou [37]; the k Neighbors adopted in this study follow a simple addition of 2, 3 or 5 to the preceding value to get the successive k value. In the state-of-the-art submission by Kumbure and Luukka [16], the performance of their proposed method was tested with eight datasets from different fields and benchmarked to kNN and three other regression methods using the RMSE metric for performance evaluation.

On the other hand, our work performs classification experiments with twenty real-world datasets and further undertook regression experiments with twelve other publicly available datasets. The relationship between the kNN performance and the principal factors in classification and regression tasks in our study is modeled by fitting the generalized linear models to examine the combined effects of factors on RMSE, a direction not considered in Kumbure and Luukka [16].

The variation of k while holding other performance determining factors constant; produces an RMSE behavior that posits low accuracies for k=1 in most investigated datasets in the classification task (fig. 10). Considering the individual impacts of the distance measures and k values in the classification task (fig. 8a), the most effect from the distance measure was observed with EU distance, followed by MA, MI, CH and FI distance respectively. A mean RMSE of 0.8 is returned when k =7, 10 or 13 while the mean RMSE obtained is 0.9 when k = 5, 17 and 20; and the worst performance is observed at k=1.

For the regression task, a similar result is obtained when k=1, and had the least RMSE when k=30. These results are in agreement with the position that large k values returns high accuracies with different dataset, and is in tandem with the submissions in some previous studies [3], [19], [68]. In terms

of the DM, MA produced the least effect, while CH had the best impact with 10 neighbors. Looking at variation in RMSE values; the dataset parameters collectively caused 74.54% variability of the RMSE values, while no reasonable difference in RMSE was observed with the variation of k levels, meaning that any value of k greater than or equal to 3 and less than or equal to 10 would be optimally suitable for classification problems (Fig. 10). Above all, these results imply that distance metrics affects the performance of kNN more than the k values and that the k neighbors only affect the performance of the algorithm individually.

Dataset characteristics identified as performance-affecting variables analyzed for classification, and regression tasks showed similar response patterns, especially where a rapid increase is observed for 9000<SS≥500 (Fig. 8). This response peaked at SS of 500 – 700 in the classification experiment and at 0 – 500 for the regression experiment. In both tasks, the most significant performance is recorded by kNN when SS is 7000 – 9000. In terms of the direct effect of the number of TD on kNN performance, the classification task showed optimal performance when the TD was $1 \leq TD \leq 3$ and $7 \leq TD \leq 9$. With an RMSE =1.4; the least effect of TD on kNN performance is observed when TD >14, signifying that kNN performs better with a smaller TD even though the best performance was not returned when 1≤ TD≤3. In addition, the individual interactions of dataset performance attribute with DM illustrated in Fig. 9 show that distance metrics are more interactive with integer datasets.

The interactions of the other dataset properties and, ultimately, the mean RMSE does not change exponentially with a change in the distance measure. The results indicate that the two dataset components that contributed the most significantly to the variance in RMSE are TD and SS. For classification, the degree of impact of TD on kNN performance depends significantly on the level of the DM, while the combined effect between SS and k neighbors does not depend on the DM used; except when the dataset SS is between 0≤SS≤500. The results in Fig. 9 (a-d) show that FI distance is a metric of choice for 0≤SS≤500 groups considering its corresponding low RMSE values across all dataset properties investigated, while other distance measures would produce impressive results for other categories.

In Fig. 10, the sensitivity and response of dataset properties (TD and SS) to change in k values during the classification experiment; and the interaction of distance metrics with varying k Neighbors is shown. Figure 10a shows more interaction when TD was 4-6. This interaction is comparable to the dynamics shown when TD>14. Nevertheless, the performance of kNN algorithm would be better when TD is of the former range. In each of the TD ranges considered, the greatest changes in performance are observed when k changes from 1 to 3 and when k > 20. In other instances of k, the performance change is relatively insignificant. In figure 10b, kNN performed best with an RMSE of 0.4; when the SS was greater than 9000. Although the SS between 501 – 1000 also produced a seemingly equivalent interaction with k variation; its sensitivity to k increase from 1 to 3 is notably large; compared to the sensitivity noticed in SS >9000. This implies that if SS of $501 \leq TD \leq 3$ is to be used in data classification experiment; then k ≥ 3 is desired for optimal accuracy to be obtained.

A similar performance change to that expressed by TD is shown in Fig. 10c where DMs were observed during k variation. From the results, the best performance in the classification task could be seen in MI and MA distances with slight overlaps at $k = 1, 3, 15 \; and \; k = 30$, where MI performed better when $k = 1 \; and \; k=3$, as against when k was 15 and 30. The performance of EU is better and less sensitive to the change in k compared to the FI distance metric. The only instance CH was observed to perform best was when k =3, at that point; it produced a lesser RMSE than the FI metric in the classification experiment. With these results, a combination of MI with TD between 7–9 and SS > 9000 would produce optimal classification results.

In the regression experiment, however, the least mean RMSE for all values of k are obtained with FI DM (Fig. 11a). The result obtained in the regression task presents CH as the most sensitive to k variation, such that optimal performance depends greatly on the use of small k values. This corroborates the submission of [18] about choosing DM being able to affect the classification accuracy of kNN algorithm. In this study, CH negatively influenced the kNN algorithm's accuracy when the k value was varied upwards. For SS between 0≤SS≤500 in Fig. 11b; the increasing mean RMSE indicates a reduction in accuracy while the mean RMSE dynamics of the 501≤SS≤1000 indicate improved performance.

Apart from these two sample sizes, others showed minimal or no interaction with the increase in k-neighbors. Although all considered DM showed variation in performance with increasing k values, Chebyshev had the worst performance both in the classification (Fig. 10c) and regression (Fig. 11a) experiments. Consequently, the almost parallel interaction pattern between the MI and MA distances observed in the classification experiment does not repeat in the regression task. Instead, EU distance maintained almost similar dynamics in its mean RMSE closeness to those of the MI metric. But at k=3, Euclidean distance performed better than MI distance. This means that the tendency of enhancing kNN performance is higher with MI distance as long as the value of $k \neq 3$.

IV. Conclusion

This work has presented a data-driven approach to discover desirable parameters for enhanced performance of kNN using eleven k-Neighbors, five DMs, and four dataset properties. From the results, the type of task (regression or classification) is the main determinant of the accuracy of kNN followed by DM. Changes in the number of k neighbors affected the kNN performance arbitrarily in both regression and classification tasks. However, the combination of parameters yielded a significant pattern-driven effect on accuracy. From the results, optimal interaction was noticed when the TD range was in the range $1 \leq TD \leq 3$ and $7 \leq TD \leq 9$; leading to the conclusion that kNN performs better with a smaller TD, preferably $3 \leq k \leq 14$. Regarding SS interaction with k-Neighbors for optimal kNN performance, we obtained the best RMSE of 0.4 when SS > 9000 during classification and ranges from $7000 \leq SS \leq 8000$ in the regression task. The DM tuning experiment showed that MI distance had the least RMSE in the classification task, even though it had almost parallel readings with the MA distance metric. FI distance produced the best

performance for the regression experiments, while CH had the worst performance in the classification and regression experiments.

The large gaps between the performances of kNN upon using different DM for the classification and regression experiments confirmed its importance to kNN's performance. However, combining DM and dataset characteristics produces interestingly significant patterns for achieving optimal kNN performance. These patterns would form the basis for weighing the performance of kNN against other notable classification and regression algorithms, including support vector machines, random forests, logistic regression, decision trees, and deep neural networks, as future works.

ACKNOWLEDGMENTS

We thank the Tertiary Education Trust Fund (TETFund) for their support and funding of this research under the TETFund Centre of Excellence in the Computational Intelligence Research University of Uyo and the University of Uyo management for the enabling environment.

REFERENCES

[1] U. G. Inyang, I. J. Eyoh, C. O. Nwokoro, F. B. Osang, and A. A. Afolorunso, "Comparative analytics of classifiers on resampled datasets for pregnancy outcome prediction", *Int. J. Adv. Comput. Sci. Appl.*, roč. 11, č. 6, s. 494–504, 2020, doi: 10.14569/IJACSA.2020.0110662.

[2] V. B. S. Prasath *et al.*, "Effects of Distance Measure Choice on KNN Classifier Performance - A Review", *arXiv:1708.04321v3*, 2019.

[3] O. Dervisevic, E. Zunic, D. Donko, and E. Buza, "Application of KNN and Decision Tree Classification Algorithms in the Prediction of Education Success from the Edu720 Platform", 2019. doi: 10.23919/SpliTech.2019.8783102.

[4] S. Zhang, „Cost-sensitive KNN classification", *Neurocomputing*, roč. 391, 2020, doi: 10.1016/j.neucom.2018.11.101.

[5] A. Hamed, A. Sobhy, and H. Nassar, "Accurate Classification of COVID-19 Based on Incomplete Heterogeneous Data using a KNN Variant Algorithm", *Arab. J. Sci. Eng.*, roč. 46, č. 9, 2021, doi: 10.1007/s13369-020-05212-z.

[6] S. Uddin, I. Haque, H. Lu, M. A. Moni, and E. Gide, "Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction", *Sci. Rep.*, roč. 12, č. 1, s. 1–11, 2022, doi: 10.1038/s41598-022-10358-x.

[7] K. U. Syaliman, E. B. Nababan, and O. S. Sitompul, "Improving the accuracy of k-nearest neighbor using local mean based and distance weight", in *Journal of Physics: Conference Series*, 2018, roč. 978, č. 1. doi: 10.1088/1742-6596/978/1/012047.

[8] J. Sun, Y. Bo, J. Luo, and J. Yang, "Application of the K nearest neighbor algorithm based on scaling weight in intelligent attendance system", 2019. doi: 10.1109/ICMTMA.2019.00142.

[9] H. Ohmaid, S. Eddarouich, A. Bourouhou, and M. Timouyas, "Comparison between svm and knn classifiers for iris recognition using a new unsupervised neural approach in segmentation", *IAES Int. J. Artif. Intell.*, roč. 9, č. 3, 2020, doi: 10.11591/ijai.v9.i3.pp429-438.

[10] P. Wang, Y. Zhang, and W. Jiang, "Application of K-Nearest Neighbor (KNN) Algorithm for Human Action Recognition", 2021. doi: 10.1109/IMCEC51613.2021.9482165.

[11] S. Zhang, Challenges in KNN Classification", *IEEE Trans. Knowl. Data Eng.*, 2021, doi: 10.1109/TKDE.2021.3049250.

[12] G. M. Bellino, L. Schiaffino, M. Battisti, J. Guerrero, and A. Rosado-Muñoz, "Optimization of the KNN supervised classification algorithm as a support tool for the implantation of deep brain stimulators in patients with Parkinson'S Disease", *Entropy*, roč. 21, č. 4, 2019, doi: 10.3390/e21040346.

[13] Y. Pan, Z. Pan, Y. Wang, and W. Wang, "A new fast search algorithm for exact k-nearest neighbors based on optimal triangle-inequality-based check strategy", *Knowledge-Based Syst.*, roč. 189, 2020, doi: 10.1016/j.knosys.2019.105088.

[14] V. Kumar and M. Sahu, "Evaluation of nine machine learning regression algorithms for calibration of low-cost PM2.5 sensor", *J. Aerosol Sci.*, roč. 157, 2021, doi: 10.1016/j.jaerosci.2021.105809.

[15] W. T. Ho and F. W. Yu, "Chiller system optimization using k nearest neighbour regression", *J. Clean. Prod.*, roč. 303, 2021, doi: 10.1016/j.jclepro.2021.127050.

[16] M. Mailagaha Kumbure and P. Luukka, "A generalized fuzzy k-nearest neighbor regression model based on Minkowski distance", *Granul. Comput.*, 2021, doi: 10.1007/s41066-021-00288-w.

[17] Y. Zhang *et al.*, "Empirical study of seven data mining algorithms on different characteristics of datasets for biomedical classification applications", *Biomed. Eng. Online*, roč. 16, č. 1, 2017, doi: 10.1186/s12938-017-0416-x.

[18] T. Sanlı, Ç. Sıcakyüz, and O. H. Yüregir, "Comparison of the accuracy of classification algorithms on three datasets in data mining: Example of 20 classes", *Int. J. Eng. Sci. Technol.*, roč. 12, č. 3, 2020, doi: 10.4314/ijest.v12i3.8.

[19] P. Kalaiyarasi and J. Suguna, "The significance of fine tuning parameters in supervised machine learning techniques for diabetic disease prediction", *Int. J. Adv. Sci. Technol.*, roč. 28, č. 17, 2019.

[20] H. A. Abu Alfeilat *et al.*, "Effects of Distance Measure Choice on K-Nearest Neighbor Classifier Performance: A Review", *Big Data*, roč. 7, č. 4. 2019. doi: 10.1089/big.2018.0175.

[21] N. Ali, D. Neagu, and P. Trundle, "Evaluation of k-nearest neighbour classifier performance for heterogeneous data sets", *SN Appl. Sci.*, roč. 1, č. 12, 2019, doi: 10.1007/s42452-019-1356-9.

[22] D. Dua and C. Graff, "UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]", 2019. http://archive.ics.uci.edu/ml (viděno 21. duben 2022).

[23] P. Tamrakar and S. P. Syed Ibrahim, "Lazy Learning Associative Classification with WkNN and DWkNN Algorithm", *ITM Web Conf.*, roč. 37, 2021, doi: 10.1051/itmconf/20213701023.

[24] R. Ehsani and F. Drabløs, "Robust Distance Measures for kNN Classification of Cancer Data", *Cancer Inform.*, roč. 19, 2020, doi: 10.1177/1176935120965542.

[25] W. Hou, D. Li, C. Xu, H. Zhang, a T. Li, "An Advanced k Nearest Neighbor Classification Algorithm Based on KD-tree", 2019. doi: 10.1109/IICSPI.2018.8690508.

[26] G. Bhattacharya, K. Ghosh, and A. S. Chowdhury, "Granger Causality Driven AHP for Feature Weighted kNN", *Pattern Recognit.*, roč. 66, 2017, doi: 10.1016/j.patcog.2017.01.018.

[27] H. Wang, P. Xu, and J. Zhao, "Improved KNN Algorithm Based on Pre-processing of Center in Smart Cities", *Complexity*, roč. 2021, 2021, doi: 10.1155/2021/5524388.

[28] S. Kang, "K-nearest neighbor learning with graph neural networks", *Mathematics*, roč. 9, č. 8, 2021, doi: 10.3390/math9080830.

[29] R. Kadry and O. Ismael, "A New Hybrid KNN Classification Approach based on Particle Swarm Optimization", *Int. J. Adv. Comput. Sci. Appl.*, roč. 11, č. 11, 2020, doi: 10.14569/IJACSA.2020.0111137.

[30] M. Jawthari and V. Stoffová, "Predicting students' academic performance using a modified kNN algorithm", *Pollack Period.*, roč. 16, č. 3, 2021, doi: 10.1556/606.2021.00374.

[31] N. N. A. Sjarif, Y. Yahya, S. Chuprat, and N. H. F. M. Azmi, "Support vector machine algorithm for SMS spam classification in the telecommunication industry", *Int. J. Adv. Sci. Eng. Inf. Technol.*, roč. 10, č. 2, 2020, doi: 10.18517/ijaseit.10.2.10175.

[32] H. Sujaini, „Image Classification of Tourist Attractions with K-Nearest Neighbor, Logistic Regression, Random Forest, and Support Vector Machine", *Int. J. Adv. Sci. Eng. Inf. Technol.*, roč. 10, č. 6, 2020, doi: 10.18517/ijaseit.10.6.9098.

[33] A. Bustamam, D. Sarwinda, B. Abdillah, and T. P. Kaloka, "Detecting lesion characteristics of diabetic retinopathy using machine learning and computer vision", *Int. J. Adv. Sci. Eng. Inf. Technol.*, roč. 10, č. 4, 2020, doi: 10.18517/ijaseit.10.4.8876.

[34] N. P. T. Prakisya, F. Liantoni, Y. H. Aristyagama, and P. Hatta, „Classification of Acute Myeloid Leukemia Subtypes M1, M2 and M3 Using K-Nearest Neighbor", *Int. J. Adv. Sci. Eng. Inf. Technol.*, roč. 11, č. 5, 2021, doi: 10.18517/ijaseit.11.5.9585.

[35] T. D. Nguyen and T. D. Nguyen, "Application of the KNN algorithm in determining the orientation of the probability area containing the ship position by GPS systems on Hai Phong coastal area", *Int. J. Adv. Sci. Eng. Inf. Technol.*, roč. 9, č. 3, 2019, doi: 10.18517/ijaseit.9.3.8869.

[36] H. Arslan and H. Arslan, "A new COVID-19 detection method from human genome sequences using CpG island features and KNN classifier", *Eng. Sci. Technol. an Int. J.*, roč. 24, č. 4, 2021, doi: 10.1016/j.jestch.2020.12.026.

[37] J. Ma and S. Zhou, "Metric learning-guided k nearest neighbor multilabel classifier", *Neural Comput. Appl.*, roč. 33, č. 7, 2021, doi: 10.1007/s00521-020-05134-9.

[38] D. Ö. Şahin, S. Akleylek, and E. Kılıç,, On the Effect of k Values and Distance Metrics in KNN Algorithm for Android Malware Detection ", *Adv. Data Sci. Adapt. Anal.*, roč. 13, č. 03n04, 2021, doi: 10.1142/s2424922x21410011.

[39] I. Iswanto, T. Tulus, and P. Sihombing, „Comparison of Distance Models on K-Nearest Neighbor Algorithm in Stroke Disease Detection", *Appl. Technol. Comput. Sci. J.*, roč. 4, č. 1, 2021, doi: 10.33086/atcsj.v4i1.2097.

[40] J. Salvador-Meneses, Z. Ruiz-Chavez, and J. Garcia-Rodriguez, "Compressed kNN: K-nearest neighbors with data compression", *Entropy*, roč. 21, č. 3, 2019, doi: 10.3390/e21030234.

[41] N. E. Z. Md Isa, A. Amir, M. Z. Ilyas, and M. S. Razalli, The Performance Analysis of K-Nearest Neighbors (K-NN) Algorithm for Motor Imagery Classification Based on EEG Signal", in *MATEC Web of Conferences*, 2017, roč. 140. doi: 10.1051/matecconf/201714001024.

[42] L. Y. Hu, M. W. Huang, S. W. Ke, and C. F. Tsai, "The distance function effect on k-nearest neighbor classification for medical datasets", *Springerplus*, roč. 5, č. 1, 2016, doi: 10.1186/s40064-016-2941-7.

[43] H. Parvin, H. Alizadeh, and B. Minati, "A Modification on K-Nearest Neighbor Classifier", *Glob. J. Comput. Sci. Technol.*, roč. 10, č. 14, 2010.

[44] A. F. Pulungan, M. Zarlis, and S. Suwilo, „Analysis of Braycurtis, Canberra and Euclidean Distance in KNN Algorithm", *SinkrOn*, roč. 4, č. 1, 2019, doi: 10.33395/sinkron.v4i1.10207.

[45] K. Chomboon, P. Chujai, P. Teerarassammee, K. Kerdprasop, and N. Kerdprasop, "An Empirical Study of Distance Metrics for k-Nearest Neighbor Algorithm", 2015. doi: 10.12792/iciae2015.051.

[46] G. Baldini and D. Geneiatakis, "A performance evaluation on distance measures in KNN for mobile malware detection", 2019. doi: 10.1109/CoDIT.2019.8820510.

[47] N. Wang *et al.*, "Study on the semi-supervised learning-based patient similarity from heterogeneous electronic medical records", *BMC Med. Inform. Decis. Mak.*, roč. 21, 2021, doi: 10.1186/s12911-021-01432-x.

[48] D. N. Cosenza *et al.*, "Comparison of linear regression, k-nearest neighbour and random forest methods in airborne laser-scanning-based prediction of growing stock", *Forestry*, roč. 94, č. 2, 2021, doi: 10.1093/forestry/cpaa034.

[49] N. Rastin, M. Z. Jahromi, and M. Taheri, "A generalized weighted distance k-Nearest Neighbor for multi-label problems", *Pattern Recognit.*, roč. 114, 2021, doi: 10.1016/j.patcog.2020.107526.

[50] S. Zhang, D. Cheng, Z. Deng, M. Zong, and X. Deng, "A novel kNN algorithm with data-driven k parameter computation", *Pattern Recognit. Lett.*, roč. 109, 2018, doi: 10.1016/j.patrec.2017.09.036.

[51] C.-M. Ma, W.-S. Yang, and . B.-W. C., "How the Parameters of K-nearest Neighbor Algorithm Impact on the Best Classification Accuracy: In Case of Parkinson Dataset", *J. Appl. Sci.*, roč. 14, č. 2, 2014, doi: 10.3923/jas.2014.171.176.

[52] M. R. Hassan, M. M. Hossain, J. Bailey, and K. Ramamohanarao, "Improving k-nearest neighbour classification with distance functions based on receiver operating characteristics", in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2008, roč. 5211 LNAI, č. PART 1. doi: 10.1007/978-3-540-87479-9_50.

[53] Z. R. Tembusai, H. Mawengkang, and M. Zarlis, „K-Nearest Neighbor with K-Fold Cross Validation and Analytic Hierarchy Process on Data Classification", *Int. J. Adv. Data Inf. Syst.*, roč. 2, č. 1, 2021, doi: 10.25008/ijadis.v2i1.1204.

[54] A. G. Jivani, "The Novel k Nearest Neighbor Algorithm", 2013. doi: 10.1109/ICCCI.2013.6466287.

[55] I. Paryudi, "What Affects K Value Selection In K-Nearest Neighbor", *Int. J. Sci. Technol. Res.*, roč. 8, č. 7, 2019.

[56] O. Kwon and J. M. Sim, "Effects of data set features on the performances of classification algorithms", *Expert Syst. Appl.*, roč. 40, č. 5, 2013, doi: 10.1016/j.eswa.2012.09.017.

[57] A. Onyezewe, A. F. Kana, F. B. Abdullahi, and A. O. Abdulsalami, "An Enhanced Adaptive k-Nearest Neighbor Classifier Using Simulated Annealing", *Int. J. Intell. Syst. Appl.*, roč. 13, č. 1, s. 34–44, 2021, doi: 10.5815/ijisa.2021.01.03.

[58] H. Saadatfar, S. Khosravi, J. H. Joloudari, A. Mosavi, and S. Shamshirband, "A new k-nearest neighbors classifier for big data based on efficient data pruning", *Mathematics*, roč. 8, č. 2, 2020, doi: 10.3390/math8020286.

[59] S. Thongsuwan, S. Jaiyen, A. Padcharoen, and P. Agarwal, "ConvXGB: A new deep learning model for classification problems based on CNN and XGBoost", *Nucl. Eng. Technol.*, roč. 53, č. 2, 2021, doi: 10.1016/j.net.2020.04.008.

[60] P. Prihandoko, B. Bertalya, and L. Setyowati, "City health prediction model using random forest classification method", 2020. doi: 10.1109/ICIC50835.2020.9288542.

[61] M. E. Ekpenyong and U. G. Inyang, "Unsupervised mining of under-resourced speech corpora for tone features classification", in *Proceedings of the International Joint Conference on Neural Networks*, 2016, roč. 2016-Octob. doi: 10.1109/IJCNN.2016.7727494.

[62] U. G. Inyang and O. C. Akinyokun, "A hybrid knowledge discovery system for oil spillage risks pattern classification", *Artif. Intell. Res.*, roč. 3, č. 4, 2014, doi: 10.5430/air.v3n4p77.

[63] M. A. Marjan, M. R. Islam, M. P. Uddin, M. I. Afjal, and M. Al Mamun, "PCA-based dimensionality reduction for face recognition", *Telkomnika (Telecommunication Comput. Electron. Control.*, roč. 19, č. 5, 2021, doi: 10.12928/TELKOMNIKA.v19i5.19566.

[64] T. Ehidiamen Oamen, "An Exploratory Factor Analysis of Work-Attributes of Pharmaceutical Sales Workforce during COVID-19 Lockdown", *J. Contemp. Res. Soc. Sci.*, roč. 3, č. 1, 2021, doi: 10.33094/26410249.2021.31.11.27.

[65] U. G. Inyang, E. E. Akpan, and O. C. Akinyokun, "A Hybrid Machine Learning Approach for Flood Risk Assessment and Classification", *Int. J. Comput. Intell. Appl.*, roč. 19, č. 2, 2020, doi: 10.1142/S1469026820500121.

[66] U. G. Inyang, S. A. Robinson, F. F. Ijebu, I. J. Udo, and C. O. Nwokoro, "Optimality Assessments of Classifiers on Single and Multi-labelled Obstetrics Outcome Classification Problems", *Int. J. Adv. Comput. Sci. Appl.*, roč. 12, č. 2, 2021, doi: 10.14569/IJACSA.2021.0120260.

[67] Y. Xia, "Determining the Number of Factors When Population Models Can Be Closely Approximated by Parsimonious Models", *Educ. Psychol. Meas.*, roč. 81, č. 6, 2021, doi: 10.1177/0013164421992836.

[68] Y. Mehmood, S. Khadam, K. Hameed, F. Riaz, and A. Ghafoor, "Effects of different data characteristics on classifier's performance", 2010. doi: 10.1109/ICET.2010.5638383.