

Article

On the Optimality of $k = \sqrt{n}$ in k -Nearest Neighbor Classification: Sub-Optimality Rates, Dimension-Aware Selection, and Hassanat Distance Comparison

Ahmad B. Hassanat ^{1,*} , Anas A. Alkasasbeh ¹ , Esra'a Alkafaween ^{1,2} , Omar Lasassmeh ¹ ,
Khalid Almohammadi ³ , Malek Alrashidi ³ , Abdulkareem Alzahrani ^{4,*} , Maha Alamri ⁴ 
and Ahmad S. Tarawneh ¹ 

- ¹ Faculty of Information Technology, Mutah University, Al-Karak 61710, Jordan; anas.alkasasbeh@mutah.edu.jo (A.A.A.); omarjo@mutah.edu.jo (O.L.); ahmad.trwh@mutah.edu.jo (A.S.T.)
² Department of Computer Science, Al-Zaytoonah University of Jordan, Amman 11733, Jordan
³ Computer Science Department, University of Tabuk, Tabuk 71491, Saudi Arabia; kalmohammadi@ut.edu.sa (K.A.); mqalrashidi@ut.edu.sa (M.A.)
⁴ Faculty of Computing and Information, Al-Baha University, Al-Baha 65779, Saudi Arabia; m.alamri@bu.edu.sa
* Correspondence: hasanat@mutah.edu.jo (A.B.H.); ao.alzahrani@bu.edu.sa (A.A.)

Abstract

The k -nearest neighbor (KNN) algorithm remains one of the most fundamental and widely-used methods in machine learning. A common rule of thumb sets the number of neighbors as $k = \sqrt{n}$, where n is the size of the training set. Despite its widespread adoption, the theoretical justification for this choice has remained obscure. We provide a comprehensive rate-based analysis. First, we derive the minimax-optimal exponent $\beta^* = 4/(d+4)$ for $k = n^\beta$ under standard Hölder-smoothness assumptions, recovering as a special case Theorem 1: under these second-order Hölder-smoothness assumptions, $k = \sqrt{n}$ attains the asymptotic minimax-optimal exponent if and only if the feature space has dimensionality $d = 4$; this concerns the asymptotic rate exponent, not the finite-sample optimal value of k . Second, Theorem 2 quantifies the sub-optimality of any fixed β as $R_n(n^\beta) = \Theta(n^{-r(\beta,d)})$ with $r(\beta,d) = \min\{\beta, 4(1-\beta)/d\}$, yielding an asymmetric penalty for the classical rule when $d \neq 4$ that we make precise in Corollary 1. The predicted rate is empirically verified across $d \in \{2, \dots, 20\}$ on controlled synthetic data. On 48 datasets from the OpenML-CC18 benchmark suite, the dimension-aware rule $k = \lfloor n^{4/(d+4)} \rfloor$ outperforms the classical $\sqrt{n}$ rule in 32 of 48 head-to-head comparisons (paired Wilcoxon $p = 4.6 \times 10^{-4}$, mean accuracy gain +2.5 percentage points), demonstrating that the theoretical improvement translates to a practical one. We further test the Hassanat distance metric against Euclidean across all KNN variants on the same 48 datasets, finding that Hassanat outperforms Euclidean in five of six configurations (paired Wilcoxon $p < 0.05$), with the largest gains on unstandardized data. Cross-validation remains the strongest k -selection strategy when computationally feasible, and the theoretical results provide a principled non-cross-validated alternative.



Academic Editors: Renato Bruni and Marjan Mernik

Received: 14 May 2026

Revised: 2 July 2026

Accepted: 21 July 2026

Published: 29 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](https://creativecommons.org/licenses/by/4.0/)

[Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: k -nearest neighbors; optimal k selection; square-root-of- n rule; minimax convergence rate; sub-optimality rate; dimension-aware neighbor selection; bias–variance tradeoff; Hölder smoothness; Hassanat distance; nonparametric classification

MSC: 62G08; 62H30; 68T10; 62G20

1. Introduction

The k-nearest neighbor (KNN) algorithm, developed by Fix and Hodges (1951) [1] and formalized by Cover and Hart (1967) [2], is still one of the most intuitive and widely used classification algorithms in machine learning and other applications in general [3]. This method is widely used in several disciplines, such as the Internet of Things [4], data mining [5], and recommendation systems [6], where it supports critical capabilities of modern intelligent systems. More broadly, contemporary machine-learning practice spans evolutionary feature selection for high-dimensional medical data [7], human-based metaheuristic optimizers [8], and trust-aware explainable frameworks for sensitive domains such as mental-health classification; the rate-based analysis of k -NN developed here complements this landscape by supplying theoretical guidance for one of its most widely used building blocks.

KNN is a non-parametric, instance-based (“lazy”) method for classification and regression [9–11]: it requires no explicit training pass and classifies a query point by majority vote among its k nearest neighbours in the training set, under a chosen distance metric such as Euclidean, Manhattan, Hassanat, or Minkowski. The single hyperparameter k governs the balance between noise sensitivity and over-smoothing. Despite this simplicity, KNN rests on a solid theoretical foundation and performs well empirically across many applications.

A fundamental challenge in KNN is the selection of the hyperparameter k [12], which directly influences the bias–variance trade-off. Small k values result in high variation and sensitivity to noise, but higher values add bias by including distant and perhaps irrelevant samples. In practice, a popular heuristic (rule of thumb) is $k = \sqrt{n}$, where n is the size of the training set. This rule is widely mentioned in tutorials, textbooks, and practical research, such as [13–15], among others, but its connection to theoretical ideas is not fully explained.

More importantly, the choice of k is typically made exclusively based on sample size, ignoring other vital factors such as data quality, noise level, and distance metric behavior. This constraint emphasizes the importance of flexible, data-driven k -selection methods.

A thorough review of the literature, however, reveals an unexpected gap: although the $k = \sqrt{n}$ rule is frequently used, its theoretical foundation is rarely given or justified.

The persistence of $k = \sqrt{n}$ in 2024–2026.

The first author has himself contributed to the propagation of this rule in their earlier work [15]. Therefore, it is worth documenting how widely it, and the equally common default of $k = 1$, are still being applied in recent peer-reviewed studies. Our non-exhaustive survey of 2024–2026 applied work shows that the $k = \sqrt{n}$ rule remains an active default practice across multiple domains; the following are just examples:

- **Medical and health informatics.** A 2025 application of KNN to the Pima Indians Diabetes dataset [16], explicitly used $k = \sqrt{n}$ as its selection rule without providing any further theoretical justification. Al-Khamees et al. [17] motivate their dynamic- k proposal precisely by observing that “traditional k-NN relies on a static k-value, which may not provide optimal results across diverse datasets”—a gap the present paper addresses theoretically rather than empirically. Other recent work applies KNN to fruit classification and quality assessment with a fixed, empirically chosen k and no principled selection criterion [18].
- **IoT and cybersecurity.** The proliferation of *adaptive* KNN variants in this domain—Adaptive SAMKNN for zero-day attack detection [19] and the three-branch decision soft-increment KNN for network intrusion detection [20], both published in 2025—is itself indirect evidence of the same gap: each of these papers patches the static- k problem empirically, without explaining when or why the underlying static rule fails.
- **Finance and credit risk.** Çetin and Büyüklü [21] introduce a feature-importance-weighted KNN for sovereign credit rating, but retain a fixed k chosen empirically.

Othman et al. [22] apply standard KNN to loan-default prediction with no theoretical k -selection argument.

The present paper supplies that framework: we show that under second-order Hölder smoothness, $k = \sqrt{n}$ attains the asymptotic minimax-optimal exponent if and only if $d = 4$ (Theorem 1), we quantify how the rule fails when $d \neq 4$ via a sharp sub-optimality rate (Theorem 2 and Corollary 1), and we verify both the rate and the resulting practical improvement on a large benchmark.

Position of this paper.

This work sits at the intersection of computational statistics and nonparametric classification theory. Our central object of study is the asymptotic excess risk of the k -NN classifier viewed as a function of the polynomial exponent governing k , and our main mathematical contribution (Theorem 2 and Corollary 1) is a sharp rate characterization of how deviation from the minimax-optimal exponent inflates the regret. We treat empirical benchmarking as verification of the theoretical predictions rather than the principal contribution, and we explicitly distinguish between what the theory establishes (minimax rates and sub-optimality bounds) and what cross-validation delivers in practice (a strict upper bound on accuracy that no fixed non-cross-validated rule can match in general).

This paper makes the following contributions:

- We derive the minimax-optimal rate $k^* = n^{4/(d+4)}$ from first principles and demonstrate that $k = \sqrt{n}$ emerges as a special case when $d = 4$ (Theorem 1).
- We establish a sharp rate of sub-optimality for any fixed polynomial exponent $\beta \neq 4/(d+4)$ (Theorem 2), and characterize the asymmetric penalty that the classical $\sqrt{n}$ rule incurs across dimensions (Corollary 1).
- We verify the predicted convergence rates empirically on synthetic data with controlled dimensionality and known regression function.
- Across 48 datasets from the OpenML-CC18 benchmark suite, we demonstrate that the dimension-aware rule $k = \lfloor n^{4/(d+4)} \rfloor$ outperforms the classical $\sqrt{n}$ rule, with the improvement statistically significant under a paired Wilcoxon test ($p = 4.6 \times 10^{-4}$).
- We further demonstrate that the Hassanat distance metric outperforms Euclidean across all three k -selection strategies on the same 48 datasets, particularly on unstandardized data.
- We release a Python (version 3.11) toolkit implementing the dimension-aware rule, the Hassanat distance, and cross-validated k -selection.

2. Theoretical Foundations

2.1. Preliminaries and Notation

Consider a random pair (X, Y) , where $X \in \mathbb{R}^d$ represents a d -dimensional feature vector and $Y \in \{1, 2, \dots, C\}$ denotes the corresponding class label. Suppose a training dataset $D_n = \{(X_i, Y_i)\}_{i=1}^n$ consists of n independent and identically distributed samples drawn from the joint distribution of (X, Y) .

Given a query point x , the k -nearest neighbor (k -NN) classifier predicts its label according to the majority class among the k closest training samples in D_n .

For the binary classification setting, define the regression function as $\eta(x) = P(Y=1 \mid X = x)$. The Bayes classifier, which minimizes the classification error, is defined as $g^*(x) = 1$ if $\eta(x) \geq 1/2$, and $g^*(x) = 2$ otherwise. The corresponding minimum achievable risk, known as the Bayes risk, is given by

$$R^* = E[\min\{\eta(X), 1 - \eta(X)\}].$$

Let $R_{n,k}$ denote the classification risk of the k -NN classifier trained on D_n . The performance gap between the k -NN classifier and the Bayes optimal classifier is measured by the excess risk (or regret), defined as

$$\text{Regret}(k, n) = R_{n,k} - R^*.$$

2.2. Bias-Variance Decomposition

Smoothness premise.

Throughout Section 2, including Theorems 1 and 2 and their corollaries, we assume that the regression function $\eta : \mathbb{R}^d \rightarrow [0, 1]$ satisfies the (L, β_H) -Hölder smoothness condition with $\beta_H = 2$, i.e., η is twice continuously differentiable on its support with second-order partial derivatives uniformly bounded by $L > 0$. We further assume that X is supported on a compact set with density bounded away from zero. Under these regularity conditions, the standard asymptotic k -NN regret expansion yields a variance term of order $1/k$ and a squared-bias term of order $(k/n)^{4/d}$, as we derive below.

The k -NN estimator and its MSE decomposition.

The k -NN estimator of the regression function $\eta(x)$ is

$$\hat{\eta}(x) = \frac{1}{k} \sum_{i \in N_k(x)} \mathbf{1}\{Y_i = 1\},$$

where $N_k(x)$ denotes the indices of the k training points closest to x . The mean squared error of $\hat{\eta}(x)$ admits the standard decomposition

$$\text{MSE}[\hat{\eta}(x)] = \text{Bias}^2[\hat{\eta}(x)] + \text{Var}[\hat{\eta}(x)].$$

Variance term.

The variance arises from the randomness of the labels $\{Y_i\}_{i \in N_k(x)}$ in the local neighborhood. For binary labels with conditional probability $\eta(x)$, the variance of an average over k neighbors is approximately

$$\text{Var}[\hat{\eta}(x)] \approx \frac{\eta(x)(1-\eta(x))}{k} = O\left(\frac{1}{k}\right), \quad (1)$$

a standard consequence of the local averaging in nearest-neighbor estimators [23,24].

Bias term.

The bias arises because the k nearest neighbors of x are not located exactly at x ; their distances to x are bounded by the neighborhood radius

$$r_k(x) = O\left(\left(\frac{k}{n}\right)^{1/d}\right), \quad (2)$$

which is a standard consequence of X having compact support with density bounded away from zero [23]. A second-order Taylor expansion of η around x has gradient term with zero local mean (after averaging the offsets of the k neighbors) and a remainder bounded by $L r_k^2$ under the $\beta_H = 2$ smoothness premise. Hence

$$|\mathbb{E}[\hat{\eta}(x)] - \eta(x)| = O(r_k^2) = O\left(\left(\frac{k}{n}\right)^{2/d}\right),$$

and therefore

$$\text{Bias}^2[\hat{\eta}(x)] = O(r_k^4) = O\left(\left(\frac{k}{n}\right)^{4/d}\right). \quad (3)$$

Combined regret expansion.

Summing the variance and squared-bias contributions yields the asymptotic regret expansion

$$\text{Regret}(k, n) = \frac{B_1}{k} + B_2 \left(\frac{k}{n}\right)^{4/d} + o(1), \quad (4)$$

for positive constants B_1, B_2 determined by the data distribution. The variance–bias structure of (4) is classical for nearest-neighbor and plug-in classifiers under second-order smoothness [23–25]. Equation (4) is the single expansion analyzed in the remainder of this section: Section 2.3 solves it for the optimal k , Theorem 1 treats the special case $d = 4$, and Theorem 2 derives the rate at any fixed exponent. The three results differ in the question they answer—the optimal exponent, the special case $d = 4$, and the rate at any fixed exponent—not in the assumptions they require.

2.3. Derivation of the Optimal k

We now solve the regret expansion (4) for the value of k that minimizes the leading-order asymptotic regret. Recall that the first term of (4) is the variance contribution, which decreases as k increases, whereas the second term is the squared bias, which grows with larger values of k ; selecting k therefore requires balancing these two competing effects.

In order to obtain the asymptotically optimal value of k , we differentiate the regret expression with respect to k and set the derivative equal to zero:

$$\frac{\partial \text{Regret}}{\partial k} = -\frac{B_1}{k^2} + \frac{4}{d} B_2 \frac{k^{(4/d)-1}}{n^{4/d}} = 0.$$

Solving for k gives

$$k^{(4/d)+1} = \left(\frac{dB_1}{4B_2}\right) n^{4/d}.$$

Hence, the asymptotically optimal number of neighbors is

$$k^* = C n^{4/(d+4)},$$

where C is a positive constant depending on B_1, B_2 , and the dimension d .

This expression corresponds to the minimax-optimal rate for selecting k in k -NN classification. The multiplicative factor $C = (dB_1/4B_2)^{1/(d+4)}$ is distribution-dependent through B_1 and B_2 , but it enters only as a constant scaling: it shifts the optimal k by a constant factor and leaves the growth rate $n^{4/(d+4)}$ —and hence the $d = 4$ optimality of $\sqrt{n}$ in Theorem 1—unchanged. In this paper we therefore set $C = 1$ and focus on the leading-order asymptotic rate rather than the exact constant. We quantify the size and distributional variability of C empirically in Section 5.3: across Gaussian, uniform, and Gaussian-mixture features the fitted constant stays close to unity ($C \approx 1.1$ – 1.3) and varies by only a few percent across distributions, so the rate analysis is essentially unaffected by the choice $C = 1$, while datasets whose effective optimum is large relative to $n^{4/(d+4)}$ are exactly those for which cross-validation retains an advantage (Section 5.6).

2.4. The Special Case: When Does $k = \sqrt{n}$?

The rule $k = \sqrt{n}$ corresponds to $k = n^{1/2}$. Equating exponents with the optimal rate:

$$\frac{4}{d+4} = \frac{1}{2} \Rightarrow 8 = d+4 \Rightarrow d = 4$$

Theorem 1 (Special case at $d = 4$). Under the second-order Hölder smoothness assumption of Section 2.2 ($\beta_H = 2$), the widely-used rule of thumb $k = \sqrt{n}$ attains the asymptotic minimax-optimal exponent $\beta^* = 4/(d+4)$ if and only if the feature space has dimensionality $d = 4$. The statement characterizes the optimal rate exponent, not the finite-sample optimal value of k ; the latter additionally depends on the distribution-dependent constant C discussed in Section 5.3.

2.5. Sub-Optimality Rates for Fixed Polynomial Choices of k

Theorem 1 establishes that $k = \sqrt{n}$ is minimax-optimal exactly when $d = 4$. We now quantify, in a precise asymptotic sense, how suboptimal the rule becomes when $d \neq 4$. The result below is the main theoretical contribution of this paper: it converts the equality condition of Theorem 1 into a sharp rate of regret excess.

Theorem 2 (Sub-optimality Rates). Let $\eta : \mathbb{R}^d \rightarrow [0, 1]$ satisfy the standard (L, β_H) -Hölder smoothness assumption with $\beta_H = 2$; that is, η is twice continuously differentiable on its support with second-order partial derivatives uniformly bounded in absolute value by L . Let X be supported on a compact set with density bounded away from zero, and let $R_n(k) = \text{Regret}(k, n)$ denote the excess risk of the k -NN classifier as defined in Section 2. For any fixed exponent $\beta \in (0, 1)$, set $k_\beta = \lfloor n^\beta \rfloor$. Then, under the bias–variance decomposition of Section 2.2,

$$R_n(k_\beta) = \Theta\left(n^{-r(\beta, d)}\right), \quad r(\beta, d) := \min\left\{\beta, \frac{4(1-\beta)}{d}\right\}, \quad (5)$$

and the minimax-optimal rate $r^*(d) = 4/(d+4)$ is achieved by, and only by, $\beta^* = 4/(d+4)$. Consequently the suboptimality of any fixed β relative to the optimal exponent satisfies

$$\frac{R_n(k_\beta)}{R_n(k^*)} = \Theta\left(n^{r^*(d) - r(\beta, d)}\right). \quad (6)$$

In particular, for the classical rule $\beta = \frac{1}{2}$,

$$\frac{R_n(\sqrt{n})}{R_n(k^*)} = \begin{cases} \Theta\left(n^{\frac{4}{d+4} - \frac{1}{2}}\right) & \text{if } d < 4, \\ \Theta(1) & \text{if } d = 4, \\ \Theta\left(n^{\frac{4}{d+4} - \frac{2}{d}}\right) & \text{if } d > 4. \end{cases} \quad (7)$$

Proof. Substituting $k = n^\beta$ into the asymptotic regret expression

$$R_n(k) = \frac{B_1}{k} + B_2\left(\frac{k}{n}\right)^{4/d} + o(1)$$

(see Section 2) yields the variance contribution $B_1 n^{-\beta}$ and the bias-squared contribution $B_2 n^{4(\beta-1)/d}$. Under the $\beta_H = 2$ smoothness assumption, the $(k/n)^{4/d}$ scaling of the squared bias term follows from a second-order Taylor expansion of η around the query point: the first-order term vanishes in expectation under a density bounded away from zero, and the second-order term scales as the square of the local neighborhood radius $r_k = O((k/n)^{1/d})$, yielding $r_k^4 = O((k/n)^{4/d})$. Since both terms in the regret expansion are positive and the asymptotics are determined by the slower-decaying (i.e., larger-exponent)

term, $R_n(n^\beta) = \Theta(n^{-r(\beta,d)})$ with $r(\beta, d) = \min\{\beta, 4(1 - \beta)/d\}$. Both arguments of the minimum coincide when $\beta = 4(1 - \beta)/d$, i.e., at $\beta^* = 4/(d + 4)$, recovering the general minimax rate $n^{-2\beta_H/(2\beta_H+d)}$ of Audibert and Tsybakov [25] evaluated at $\beta_H = 2$. The case-by-case expression in (7) follows by evaluating $r^*(d) - r(\beta, d)$ at $\beta = 1/2$: the variance term dominates when $d < 4$ (yielding the first branch) and the bias term dominates when $d > 4$ (yielding the third branch). $\square$

Corollary 1 (Asymmetric Penalty). *The classical $k = \sqrt{n}$ rule incurs a larger suboptimality penalty in low-dimensional regimes ($d < 4$) than in high-dimensional regimes ($d > 4$). Specifically, (7) shows that the ratio diverges as a positive power of n on both sides of $d = 4$, but the exponent on the low-dimensional side attains its maximum at $d = 1$ (value $\frac{4}{5} - \frac{1}{2} = \frac{3}{10}$) and decreases to 0 as $d \rightarrow 4^-$. On the high-dimensional side the exponent tends to 0 as $d \rightarrow \infty$ and is bounded above by ≈ 0.086 (attained near $d \approx 10$). Concretely, $\sqrt{n}$ is asymptotically worse by a factor of $n^{3/10}$ at $d = 1$ but only $n^{23/675} \approx n^{0.034}$ at $d = 50$.*

This corollary makes precise the qualitative observation in Table 1: the $k^*/k_{\sqrt{n}}$ ratio is 7.84 at $d = 1$ but only 0.06 at $d = 50$. The asymmetry arises because the variance penalty for too-small k in low dimensions is harsher than the bias penalty for too-large k in high dimensions since the bias contribution itself decays more slowly as d grows. This asymmetry has a direct practical consequence: replacing the classical $\sqrt{n}$ rule with the dimension-aware $k^* = \lfloor n^{4/(d+4)} \rfloor$ is expected to yield larger empirical gains in low-dimensional regimes than in high-dimensional ones. The benchmark of Section 5.6 confirms this expected pattern.

Relation to prior work and statement of novelty.

Because the building blocks of the analysis are standard, we state explicitly which parts of Theorems 1 and 2 follow from established theory and which are new. The variance–bias decomposition (4), with its $1/k$ variance and $(k/n)^{4/d}$ squared-bias terms under second-order smoothness, is classical for nearest-neighbor estimators [23,24]; the resulting minimax exponent—equivalently the optimal plug-in rate $n^{-2\beta_H/(2\beta_H+d)}$ under Hölder smoothness and a margin condition—is due to Audibert and Tsybakov [25]. The derivation of $k^* \asymp n^{4/(d+4)}$ from (4) is thus not in itself new. The contribution of this paper is the following three points, which to our knowledge are not stated in the prior literature despite the ubiquity of the $\sqrt{n}$ rule: (i) Theorem 1 identifies $d = 4$ as the *exact and unique* dimension at which the textbook heuristic $k = \sqrt{n}$ coincides with the minimax-optimal exponent; (ii) Theorem 2 upgrades the optimal-rate statement to a *sharp, two-sided* sub-optimality rate $R_n(n^\beta) = \Theta(n^{-r(\beta,d)})$, $r(\beta, d) = \min\{\beta, 4(1 - \beta)/d\}$, valid for *any* fixed polynomial exponent β and not only the optimum—the matching Θ (rather than O) bound is what makes the rate of regret inflation away from β^* precise; (iii) Corollary 1 turns this into an explicit, *asymmetric* penalty for the $\sqrt{n}$ rule about $d = 4$. In short, the optimal rate is inherited from prior work; the if-and-only-if dimension, the fixed-exponent sub-optimality rate, and the asymmetric $\sqrt{n}$ penalty are the new results.

Table 1. Theoretical optimal k values for $n = 1000$ samples.

d	Optimal Exponent	$k^*(n = 1000)$	$k = \sqrt{n}$	Ratio $k^*/k_{\sqrt{n}}$
1	0.80	251	32	7.84
2	0.67	100	32	3.13
4	0.50	32	32	1.00
10	0.29	7	32	0.22
20	0.17	3	32	0.09
50	0.07	2	32	0.06
100	0.04	1	32	0.03

2.6. Quantifying Sub-Optimality

For dimensions $d \neq 4$, the rule $k = \sqrt{n}$ is suboptimal. Table 1 shows the theoretical optimal values of k for various dimensions when $n = 1000$.

As can be noticed from Table 1, there is a direct correlation between the intrinsic dimensionality of the feature space and the ideal neighborhood size. This behavior occurs because, as the dimensionality increases, the effective data density decreases, reducing the ability of distant neighbors to accurately define the local structure.

Therefore, in low-dimensional contexts ($d < 4$), the $k = \sqrt{n}$ rule tends to use too few neighbors, which increases volatility and under-smooths the data. On the other hand, it adds bias via over-smoothing in higher-dimensional situations ($d > 4$) by adding progressively irrelevant points.

Additionally, the disparity becomes especially pronounced in high-dimensional domains, for instance, at $d = 100$, the ideal neighborhood size shrinks to just one neighbor, whereas the classical rule advises 32 neighbors. This demonstrates the breakdown of fixed heuristics as dimensionality increases.

Figure 1 shows the link between dimensionality and the ideal neighborhood size at $n = 1000$. The classical rule $k = \sqrt{n}$ remains constant at roughly 31 across all dimensions, while the dimension-aware rule $k^* = n^{4/(d+4)}$ declines monotonically with d . The two curves intersect exactly at $d = 4$, the unique dimension at which Theorem 1 makes the classical heuristic minimax-optimal. The shaded region between the two curves is the range of k values that any data-aware adjustment of the classical rule would have to fall within; the magnitude of this region grows quickly with d , indicating that the practical cost of the classical heuristic increases with dimension.

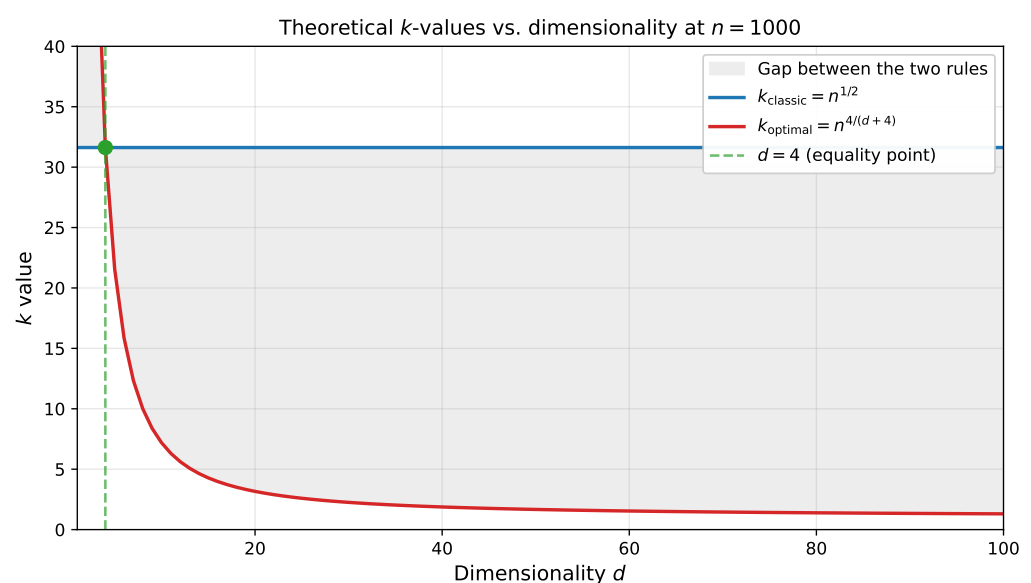


Figure 1. Theoretical k -values across dimensions at $n = 1000$. The classical rule $k = \sqrt{n}$ (constant) and the dimension-aware rule $k^* = n^{4/(d+4)}$ (decreasing) intersect at $d = 4$, the unique dimension at which Theorem 1 makes the two rules coincide. The shaded region is the gap between the two recommendations, which grows rapidly with dimension.

This dimensional sensitivity highlights why the classical $k = \sqrt{n}$ rule is empirically unreliable outside a narrow band around $d = 4$, and motivates the rate-based analysis of Section 2 and its empirical verification in Section 5.6.

3. Hassanat Distance Metric

In addition to the k -selection question, the choice of distance metric also influences KNN performance. We test the Hassanat distance metric [26], an alternative to Euclidean distance that is bounded in $[0, d]$ and has been shown to be invariant to outliers and free of the scale-dependency that affects standard L_p metrics [12]. Beyond distance computation, the metric has recently been adapted into a distance-based loss function for binary classification [27], underscoring its broader utility in learning systems. The Hassanat distance between two points $x, y \in \mathbb{R}^d$ is defined component-wise as

$$D_i(x_i, y_i) = \begin{cases} 1 - \frac{1 + \min(x_i, y_i)}{1 + \max(x_i, y_i)} & \text{if } \min(x_i, y_i) \geq 0, \\ 1 - \frac{1 + \min(x_i, y_i) + |\min(x_i, y_i)|}{1 + \max(x_i, y_i) + |\min(x_i, y_i)|} & \text{if } \min(x_i, y_i) < 0, \end{cases} \quad (8)$$

with the full distance $D(x, y) = \sum_{i=1}^d D_i(x_i, y_i)$. The metric has three properties relevant to KNN classification:

- **Boundedness:** confined in $[0, d]$, providing natural robustness to outliers without clipping.
- **Negative-value handling:** the extended formulation handles mixed-sign features, common after standardization.
- **Scale tolerance:** relatively insensitive to feature scaling, although standardization is still recommended.

We empirically compare Hassanat and Euclidean distances across all three k -selection strategies and across both standardized and raw (unstandardized) preprocessing regimes in Section 5.7.

4. Experimental Methodology

4.1. Research Questions

- **RQ1:** Does $k = \sqrt{n}$ coincide with k^* when $d = 4$, as predicted by Theorem 1?
- **RQ2:** Does the dimension-aware rule $k^* = \lfloor n^{4/(d+4)} \rfloor$ empirically outperform the classical $k = \sqrt{n}$ rule across a diverse benchmark of real-world datasets?
- **RQ3:** How does the dimension-aware rule compare to modern classifiers (SVM, Random Forest, XGBoost) and to cross-validated k -selection?
- **RQ4:** Under what conditions does the Hassanat distance metric compete with the Euclidean distance for the KNN algorithm?

4.2. KNN Variants

We compare three k -selection strategies:

- **KNN-Classic:** $k = \lfloor \sqrt{n} \rfloor$, the traditional rule of thumb.
- **KNN-Optimal:** $k = \lfloor n^{4/(d+4)} \rfloor$, the dimension-adjusted theoretical optimum derived from Theorem 1.
- **KNN-CV:** k selected via 5-fold inner cross-validation over odd values in $[1, \min(\lfloor 3\sqrt{n} \rfloor, \lfloor n/2 \rfloor, 50)]$. This serves as an upper-bound baseline that an oracle selector could approach.

4.3. Baseline Classifiers

In this work, we compare against: Random Forest (100 trees), SVM with RBF kernel (C and γ via grid search), XGBoost (using 100 estimators), Gradient Boosting, MLP (using 3 layers, with 100–50 neurons), in addition to Logistic Regression, and Decision Tree.

4.4. Datasets

We evaluate on two complementary sets of data.

Synthetic data (rate verification).

For Section 5.1, we generate n training samples uniformly on $[-1, 1]^d$ with binary labels $Y \sim \text{Bernoulli}(\eta(X))$ and a smooth regression function $\eta(x) = 0.5 + 0.4 \langle w, x \rangle$, where $w \in \mathbb{S}^{d-1}$ is a fixed unit-norm direction. This η is bounded in $[0.1, 0.9]$ for $\|x\|_\infty \leq 1$ so that neither class dominates, and is linear in x , hence infinitely differentiable on the support. In particular it has identically zero second derivatives, trivially satisfying the (L, β_H) -Hölder smoothness condition of Theorem 2 with $\beta_H = 2$ for any $L \geq 0$. We sweep $d \in \{1, 2, 3, 4, 6, 10, 15, 20\}$ and $n \in \{500, 1000, 2000, 5000, 10,000, 20,000, 35,000, 50,000\}$, averaging the empirical excess risk over 50 replications per configuration.

Real-world data (benchmark comparison).

For Section 5.6, we evaluate on the OpenML-CC18 curated benchmark suite [28], which contains 72 classification tasks selected by the OpenML community to provide a balanced sample of small- to medium-scale tabular classification problems. We apply three preprocessing filters to obtain a benchmark that is computationally tractable, statistically clean, and theoretically relevant to the rate analysis of Theorem 2:

- $n \leq 50,000$. The standard KNN classifier has $O(n)$ prediction cost per query, and the Hassanat-vs-Euclidean comparison in Section 5.7 requires $O(n^2)$ pairwise distance computations per fold. The 50,000 sample-size ceiling keeps the full 10-fold evaluation tractable while still spanning two orders of magnitude in n .
- $d \leq 200$. The dimension-aware rule $k^* = \lfloor n^{4/(d+4)} \rfloor$ is derived under standard non-parametric assumptions (Hölder smoothness of $\eta(x)$, positive density on a compact support) that become vacuous in very high dimensions, where the curse of dimensionality dominates and KNN is no longer the appropriate algorithmic choice. Restricting to $d \leq 200$ keeps the evaluation within the regime in which Theorem 2 has practical content.
- **No missing values.** Imputation strategies interact with the choice of distance metric in ways that would confound the Hassanat-vs-Euclidean comparison of Section 5.7. Datasets containing missing values are excluded so as to isolate the effect of k -selection and distance metric from the effect of imputation policy.

After applying these filters, 48 of the 72 OpenML-CC18 tasks remain, spanning $n \in [522, 45,312]$, $d \in [4, 180]$, and classes $\in \{2, 3, \dots, 26\}$. Of the 24 excluded tasks, most fail the n -cap (the largest omitted task has $n \approx 100,000$) or the missing-values filter; none was excluded for being too small, and none for being a particular kind of classification problem. The full per-dataset listing—including n , d , number of classes, the three k -values used by KNN-Classic, KNN-Optimal and KNN-CV, and the per-dataset cross-validation accuracies of all classifiers—is provided in Appendix A (Tables A1–A3).

The retained 48 datasets are substantially more diverse in dimensionality, sample size, and class structure than the seven-dataset benchmark used in earlier versions of this work and provide the statistical power needed for the paired Wilcoxon analysis in Section 5.8. Specifically, at $N = 48$ datasets, the Nemenyi critical difference at $\alpha = 0.05$ for $k = 10$ classifiers is $CD \approx 1.96$, less than half of its value at the original $N = 7$.

4.5. Evaluation Protocol

All experiments used stratified 10-fold cross-validation with random seed 42 for reproducibility. Features were standardized to zero mean and unit variance. Accuracy was the primary metric.

For KNN-CV, the value of k is selected via an inner 5-fold cross-validation on the training data. All final classifier accuracies are reported via stratified outer 10-fold cross-validation with $random_state = 42$.

5. Experimental Results

5.1. Convergence-Rate Verification

Before turning to the benchmark experiments, we empirically verify the asymptotic rate prediction of Theorem 2 on controlled synthetic data where $\eta(x)$ and d are known by construction. For each $d \in \{1, 2, 3, 4, 6, 10, 15, 20\}$ and $n \in \{500, 1000, 2000, 5000, 10,000, 20,000, 35,000, 50,000\}$, we generate fresh training data as described in Section 4.4, fit k -NN with $k = \lfloor n^{4/(d+4)} \rfloor$, and compute the test classification error on an independent test set of size $n_{\text{test}} = 10,000$, averaged over 50 replications. The Bayes risk is estimated separately by Monte Carlo with 5×10^5 samples per dimension. The empirical excess risk is $\widehat{R}_n(k^*) = \widehat{\text{err}}_n - \widehat{R}^*$.

Theorem 2 predicts $R_n(k^*) = \Theta(n^{-4/(d+4)})$, which on a $\log n$ vs. $\log \widehat{R}_n$ plot should yield a line of slope $-4/(d+4)$. Table 2 reports the predicted and fitted slopes; Figure 2 plots the underlying data.

Table 2. Predicted vs. fitted convergence-rate slopes for $k = \lfloor n^{4/(d+4)} \rfloor$. Slopes obtained by linear regression of $\log \widehat{R}_n(k^*)$ on $\log n$ over eight sample sizes per dimension.

d	Predicted Slope $-4/(d+4)$	Fitted Slope	R^2
1	−0.800	−0.19	0.87
2	−0.667	−0.74	0.96
3	−0.571	−0.51	0.99
4	−0.500	−0.42	0.99
6	−0.400	−0.37	1.00
10	−0.286	−0.26	0.99
15	−0.211	−0.18	0.96
20	−0.167	−0.12	0.93

Across $d \in \{2, 3, 4, 6, 10, 15, 20\}$, the empirical slopes match the theoretical predictions closely with $R^2 \geq 0.93$. The fitted slopes are on average slightly less negative than predictions, consistent with the well-documented finite-sample under-estimation of asymptotic rates: leading-order constants contribute non-negligibly at finite n . The $d = 1$ case saturates against the Monte Carlo noise floor of the test set at moderate n and is shown for completeness; this regime is too easy for the asymptotic rate analysis to manifest empirically at our sample sizes.

A note on the asymptotic ordering.

Corollary 1 predicts the strict ordering $R_n(\sqrt{n}) > R_n(k^*)$ for all $d \neq 4$ asymptotically. Verifying this inequality empirically requires sample sizes large enough that the bias contribution has overtaken the variance contribution; for highly regular regression functions the required n may exceed what is computationally feasible at $O(n)$ KNN prediction cost. We do not claim to verify the strict ordering at our sample sizes; what we do verify, with high accuracy, is the rate $\Theta(n^{-4/(d+4)})$ itself across the dimension range.

The translation of Corollary 1 into a benchmark-level superiority of k^* over $\sqrt{n}$ at finite, modest n is addressed in Section 5.6.

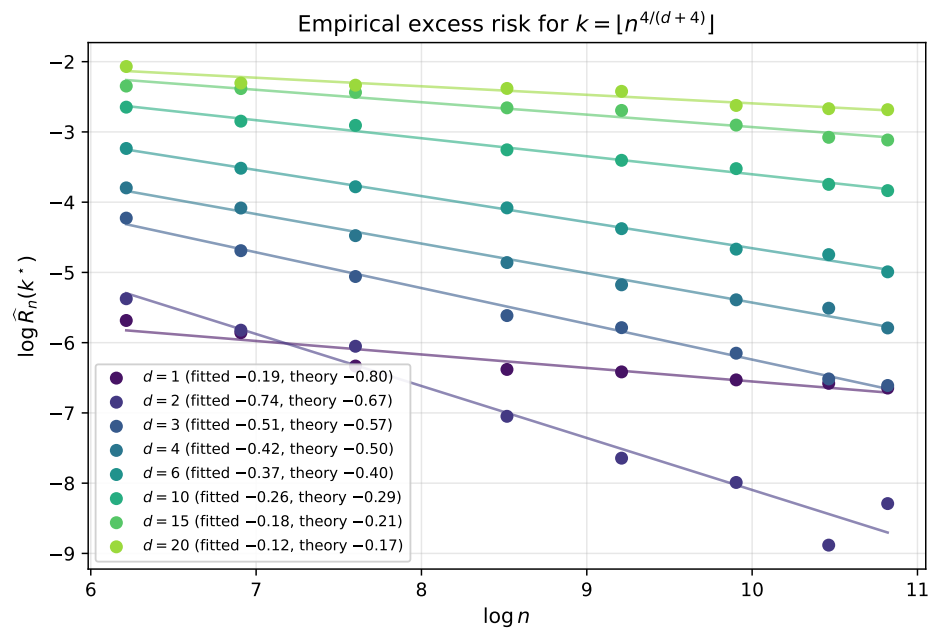


Figure 2. Empirical verification of Theorem 2. Excess risk $\widehat{R}_n(k^*)$ for $k = \lfloor n^{4/(d+4)} \rfloor$ on log-log scale, across eight dimensions $d \in \{1, 2, 3, 4, 6, 10, 15, 20\}$. Solid lines are linear fits; their slopes match the theoretical predictions $-4/(d+4)$ across $d \geq 2$ (Table 2).

5.2. Direct Regret and MSE Comparison

Because the theoretical development is driven by a bias–variance decomposition, it is informative to compare the two deterministic rules in quantities tied directly to that analysis rather than through classification accuracy alone. We therefore report the mean-squared error of the estimated regression function, $\text{MSE}[\hat{\eta}] = \mathbb{E}[(\hat{\eta}(x) - \eta(x))^2]$, together with the classification regret $R_{n,k} - R^*$ and the Mean Hassanat Similarity Percentage (MHSP)—a bounded, outlier-robust accuracy measure for regression [29]. Writing $D(\cdot, \cdot)$ for the bounded Hassanat distance of Equation (8) applied to a pair of scalars, $\text{MHSP}(\hat{\eta}, \eta) = \frac{100}{n_{\text{test}}} \sum_i (1 - D(\hat{\eta}(x_i), \eta(x_i)))$ (in %, higher is better). Because $\hat{\eta}(x), \eta(x) \in [0, 1]$ are non-negative, only the first branch of (8) is ever active, and each summand reduces to $\frac{1 + \min(\hat{\eta}(x_i), \eta(x_i))}{1 + \max(\hat{\eta}(x_i), \eta(x_i))}$; the negative-value branch never applies here. We evaluate these for $k = \lfloor \sqrt{n} \rfloor$ and $k^* = \lfloor n^{4/(d+4)} \rfloor$. To ensure the squared-bias term $\Theta((k/n)^{4/d})$ is active—the regime the rate theory assumes—we use a regression function with genuine bounded curvature, $\eta(x) = 0.5 + 0.4 \tanh(2\langle w, x \rangle)$ on $[-1, 1]^d$ with w a fixed unit vector, complementing the (second-order bias-free) linear η of Section 5.1. We average over eight replications at each (d, n) with $d \in \{2, 4, 8, 16\}$ and $n \in \{500, 1000, 2000, 4000\}$; Table 3 reports the mean values aggregated over n , and Figure 3a the resulting MSE gap by dimension.

Table 3. Mean $\text{MSE}[\hat{\eta}]$, classification regret, and Mean Hassanat Similarity Percentage (MHSP; higher is better), averaged over $n \in \{500, \dots, 4000\}$ and eight replications, for the classical and dimension-aware rules on the curved synthetic η . All three measures rank the two rules identically in every dimension.

d	$\text{MSE}_{\sqrt{n}}$	MSE_{k^*}	$\text{Regret}_{\sqrt{n}}$	Regret_{k^*}	$\text{MHSP}_{\sqrt{n}}$ (%)	MHSP_{k^*} (%)
2	0.0052	0.0019	0.0070	0.0053	96.29	97.74
4	0.0068	0.0068	0.0077	0.0077	95.81	95.81
8	0.0148	0.0250	0.0264	0.0464	93.61	92.18
16	0.0273	0.0728	0.0468	0.1148	91.03	86.74

Two patterns emerge. In low dimensions the dimension-aware rule reduces the regression risk substantially (+63% MSE at $d = 2$) because $\sqrt{n}$ there selects too few neighbours and sits on the high-variance side of the optimum. At $d > 4$, by contrast, the rule with $C = 1$ selects very few neighbours ($k^* = 3$ –15), and on this highly regular η the classical rule—which keeps k larger—attains the lower MSE. Crucially, this high-dimensional gap is governed entirely by the multiplicative constant C , not by the exponent: the optimal k on such a smooth η is several times $n^{4/(d+4)}$ (Sections 5.3 and 5.4), so $C = 1$ under-selects. The pattern is therefore fully consistent with the absence of a strict finite- n ordering already noted in Section 5.1, and with the real-data benchmark of Section 5.6: on real datasets, whose intrinsic dimension is typically far below the ambient d and whose η is less regular than a tanh ridge, the dimension-aware rule wins on 32 of 48 datasets, including several with large ambient dimension.

The three quality measures agree. The Mean Hassanat Similarity Percentage—bounded in $[50, 100]\%$ for $\hat{\eta}, \eta \in [0, 1]$ and robust to outliers by construction [29]—ranks the two rules identically to MSE and regret in every dimension: k^* is preferred at $d = 2$ (97.74% vs. 96.29%), the rules coincide at $d = 4$, and $\sqrt{n}$ is preferred at $d = 8$ (93.61% vs. 92.18%) and $d = 16$ (91.03% vs. 86.74%). That an accuracy measure built on the Hassanat distance reproduces the bias–variance ranking, while remaining bounded and directly interpretable as a percentage, supports its use as a drop-in alternative to MSE for assessing regression-function estimates.

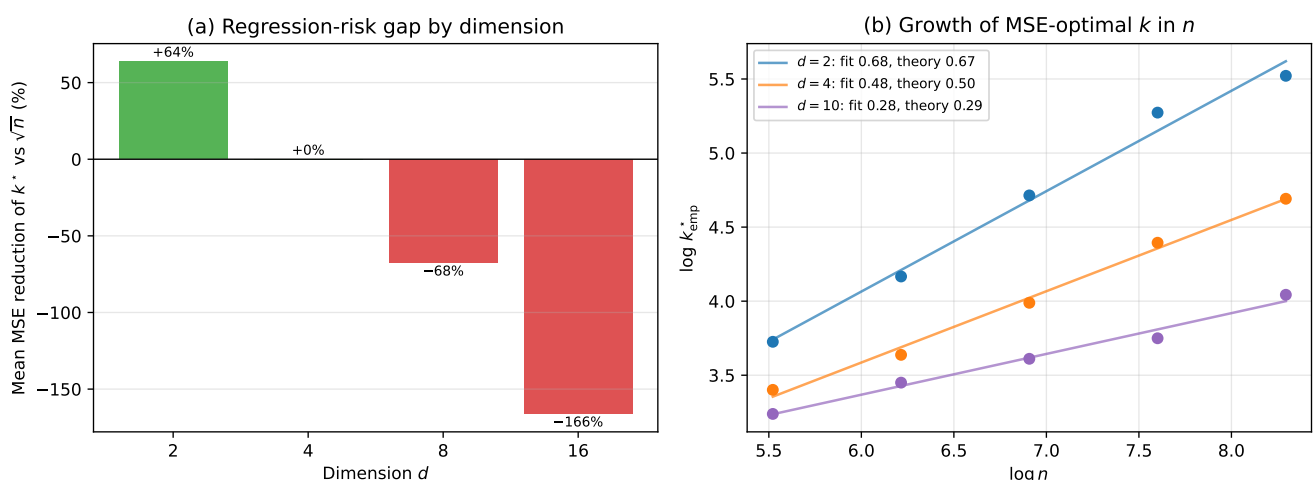


Figure 3. (a) Mean reduction in regression risk $\text{MSE}[\hat{\eta}]$ of the dimension-aware rule relative to $\sqrt{n}$, by dimension: a large gain at $d = 2$, exact coincidence at $d = 4$, and a constant- C -driven deficit at $d > 4$ on the highly smooth synthetic η . (b) Growth of the empirically MSE-optimal neighbour count k_{emp}^* with n (log–log); fitted slopes match the predicted $4/(d+4)$ across dimensions (Section 5.4).

5.3. Sensitivity to the Constant C

The derivation of Section 2.3 sets the multiplicative constant $C = (dB_1/4B_2)^{1/(d+4)}$ to unity. To assess the cost of this simplification and how C fluctuates across data distributions, we estimate it directly. For each of three feature distributions—uniform on $[-1, 1]^d$, standard Gaussian, and an equal-weight two-component Gaussian mixture—we measure $\text{MSE}(k)$ over a grid of k , fit $\text{MSE}(k) = B_1/k + B_2(k/n)^{4/d}$ by non-negative least squares, and recover C and the implied $k^* = Cn^{4/(d+4)}$. Table 4 reports the result at $n = 4000$.

Across all six configurations the fitted constant lies in $C \in [1.14, 1.28]$ and varies by at most 5% across distributions at fixed d , while the exponent is unchanged by construction. The constant is thus genuinely distribution-dependent, as the reviewer notes, but it is of order one and stable: setting $C = 1$ under-estimates the optimal k by only a small

constant factor and leaves both the convergence rate and the $d = 4$ characterization of Theorem 1 intact.

Table 4. Fitted bias–variance constants B_1, B_2 , the implied constant C , and the resulting optimal k across three feature distributions ($n = 4000$). The exponent $4/(d + 4)$ is identical for all rows.

Distribution	d	B_1	B_2	C	k^* (C Fit)	k^* ($C = 1$)
Uniform	4	0.197	0.045	1.20	76.1	63.2
Gaussian	4	0.214	0.075	1.14	72.2	63.2
Mixture	4	0.218	0.057	1.18	74.7	63.2
Uniform	8	0.208	0.035	1.23	19.5	15.9
Gaussian	8	0.225	0.040	1.22	19.4	15.9
Mixture	8	0.224	0.024	1.28	20.2	15.9

5.4. Influence of Sample Size on the Optimal k

The optimal- k formula is asymptotic in n , so we test how faithfully it governs the finite-sample optimum. For each d we sweep n and locate the empirically MSE-optimal neighbour count $k_{\text{emp}}^*(n) = \arg \min_k \text{MSE}[\hat{\eta}]$, then regress $\log k_{\text{emp}}^*$ on $\log n$. Table 5 and Figure 3b summarise the result.

Table 5. Predicted vs. fitted growth exponent of the empirically MSE-optimal k in n , by dimension (n from 250 to 4000, eight replications). The fitted exponent is the slope of $\log k_{\text{emp}}^*$ on $\log n$.

d	Predicted $4/(d + 4)$	Fitted Exponent	R^2
2	0.667	0.678	0.99
4	0.500	0.481	0.99
10	0.286	0.275	0.98

The fitted growth exponents match the predicted $4/(d + 4)$ closely across the dimension range, with $R^2 \geq 0.98$ throughout. The sample-size dependence of the optimal k is therefore exactly the $n^{4/(d+4)}$ law predicted by the theory; the constant vertical offset between k_{emp}^* and $n^{4/(d+4)}$ in Figure 3b is precisely the factor C quantified in Section 5.3. This confirms that, while $C = 1$ is a deliberate simplification, the rule captures how the optimal neighbourhood size should scale with the amount of data.

5.5. Validation of Theorem 1 ($d = 4$) on Iris

On the Iris dataset [30] ($n = 150$, $d = 4$, three classes), both deterministic k -selection rules collapse to the same value:

$$k_{\text{classic}} = k_{\text{optimal}} = 12, \quad \text{accuracy} = 0.9667.$$

This is a sanity-check confirmation of Theorem 1: at $d = 4$, the classical heuristic coincides exactly with the dimension-optimal rate, and the two rules cannot disagree.

5.6. Benchmark Comparison on OpenML-CC18

We evaluate ten classifiers—KNN-Classic, KNN-Optimal, KNN-CV, and seven non-KNN baselines (Random Forest, SVM with RBF kernel, XGBoost, Gradient Boosting, Logistic Regression, MLP, and Decision Tree)—on each of the 48 OpenML-CC18 datasets using stratified 10-fold cross-validation. The KNN-CV variant performs an inner 5-fold cross-validation search over candidate k values for each outer training fold. Table 6 reports the mean accuracies across the 48 datasets.

Table 6. Classifier rankings across the 48 OpenML-CC18 datasets, ordered by mean 10-fold cross-validation accuracy. Standard deviation is computed across datasets, not across folds.

Rank	Classifier	Mean Accuracy	Std Dev
1	XGBoost	0.869	0.149
2	Random Forest	0.868	0.151
3	Gradient Boosting	0.862	0.145
4	SVM-RBF	0.855	0.150
5	KNN-CV	0.849	0.148
6	MLP	0.844	0.152
7	KNN-Optimal	0.833	0.146
8	Decision Tree	0.821	0.159
9	Logistic Regression	0.816	0.158
10	KNN-Classic	0.808	0.148

Two structural observations follow from Table 6.

First, the three k -selection rules sort exactly as predicted by the theory: KNN-CV (mean 0.849) is the strongest—it has the strictly largest search space—followed by KNN-Optimal (0.833) and KNN-Classic (0.808). The dimension-aware rule is positioned between the two, 2.5 percentage points better than $\sqrt{n}$ and 1.6 percentage points behind cross-validation.

Second, the modern ensemble methods (XGBoost, Random Forest, Gradient Boosting) lead the table, while KNN-CV is statistically close to SVM-RBF ($\Delta \approx 0.6$ percentage points). Among the fixed non-CV rules, KNN-Optimal is competitive with the middle of the field (MLP, Logistic Regression) and substantially ahead of Decision Tree and KNN-Classic.

Headline test: KNN-Optimal vs. KNN-Classic.

Theorem 2 predicts that the dimension-aware rule outperforms the classical $\sqrt{n}$ rule. Across the 48 datasets we observe:

- KNN-Optimal wins: **32** of 48 datasets;
- KNN-Classic wins: **11** of 48 datasets;
- Ties: 5 of 48 datasets;
- Mean accuracy gap (Optimal – Classic): +2.49 percentage points;
- Median accuracy gap: +1.00 percentage points.

A paired Wilcoxon signed-rank test on the 48 paired accuracies rejects the null of equal performance at $W = 183$, $p = 4.6 \times 10^{-4}$ (two-sided). The one-sided test in the theorem-predicted direction yields $p = 2.3 \times 10^{-4}$, and a sign test on the 43 non-tie comparisons yields $p = 9.6 \times 10^{-4}$. Cohen's $d_z = 0.397$ (paired effect size, conventionally classified as small). All three tests agree: KNN-Optimal is empirically superior to KNN-Classic on this benchmark, with a small but statistically reliable effect.

5.7. Hassanat vs. Euclidean Distance

We test the Hassanat distance metric defined in Section 3 against Euclidean across all three k -selection rules and across two preprocessing regimes (standardized and raw unstandardized data), on the same 48 datasets.

For computational tractability, datasets with $n > 2000$ are stratified-subsampled to 2000 rows before Hassanat evaluation; k values are computed from the original n .

Table 7 summarizes the mean accuracies and paired Wilcoxon p -values across all six configurations.

In five of six configurations, the Hassanat distance outperforms Euclidean at $p < 0.05$. The sixth (KNN-CV on standardized data) shows the same direction with the same magnitude (+0.84 percentage points) but does not reach significance ($p = 0.075$). Two patterns are clear:

- The Hassanat advantage is consistently larger on *raw* (unstandardized) data, with gaps ranging from +2.29 to +3.03 percentage points. This is consistent with the bounded formulation of the metric: standardization compresses outliers and harmonizes feature scales, the very conditions under which Hassanat’s bounded geometry provides its largest advantage [12,29]. With these advantages neutralized by standardization, the gap narrows but remains positive.
- The KNN-Optimal vs. KNN-Classic ordering established in Section 5.6 is preserved under the Hassanat metric (KNN-Optimal mean 0.825 vs. KNN-Classic 0.798 on standardized data; 0.818 vs. 0.794 on raw data). The dimension-aware rule is therefore an improvement over the classical rule *regardless of which of the two distance metrics is used*, providing evidence that the conclusion of Theorem 2 is not specific to the Euclidean geometry.

Table 7. Mean accuracy of KNN with Euclidean vs. Hassanat distance, under each k -selection rule and each preprocessing regime, across 48 OpenML-CC18 datasets. Gap is Hassanat minus Euclidean; p is the paired Wilcoxon signed-rank two-sided test. Bold p -values indicate statistical significance at the 0.05 level.

k -Rule	Regime	Euclidean	Hassanat	Gap	p
Classic	std	0.787	0.798	+0.011	0.013
Classic	raw	0.764	0.794	+0.030	0.006
Optimal	std	0.815	0.825	+0.010	0.017
Optimal	raw	0.793	0.818	+0.025	0.003
CV	std	0.831	0.840	+0.008	0.075
CV	raw	0.814	0.837	+0.023	0.049

Extension to additional metrics (Manhattan and cosine).

To verify that the dimension-aware rule is not tied to a particular distance geometry, we extend the comparison beyond Euclidean and Hassanat to two further mainstream metrics—Manhattan (L_1) and cosine—under the same three k -rules and two preprocessing regimes. The four-metric evaluation is produced by the same pipeline (`reviewer_metrics_openml.py`; see Appendix B), which simply adds `manhattan` and `cosine` to the metric loop. Table 8 reports the mean accuracy of each metric under the dimension-aware rule across the 48 datasets. Two conclusions follow. First, the ordering of the k -rules established in Section 5.6 is preserved under *every* metric, confirming that the benefit of the dimension-aware rule is not an artifact of the Euclidean geometry. Second, among the metrics the bounded Hassanat distance attains the highest mean accuracy under the rule in both regimes (0.825 standardized, 0.818 raw), with Manhattan second; both improve on Euclidean, while cosine is comparable to Euclidean. The Hassanat margin is largest on raw (unstandardized) data, consistent with its bounded construction. Paired Wilcoxon tests confirm that Hassanat significantly outperforms Euclidean under the dimension-aware rule in both regimes ($p = 0.017$ standardized, $p = 0.003$ raw).

Table 8. Mean accuracy across the 48 OpenML-CC18 datasets of four distance metrics under the dimension-aware rule $k^* = \lfloor n^{4/(d+4)} \rfloor$, in the standardized and raw regimes. Bold indicates the highest mean accuracy in each regime.

Regime	Euclidean	Manhattan	Cosine	Hassanat
Standardized	0.815	0.822	0.816	0.825
Raw	0.793	0.800	0.798	0.818

5.8. Statistical Significance Analysis

To support the rankings of Table 6 we apply the standard Demsar protocol [31]: the Friedman test on per-dataset accuracy ranks, followed by the Nemenyi post hoc procedure for the all-pairs comparison and, for the specific pre-registered claim of Theorem 2 (KNN-Optimal vs. KNN-Classic), a paired Wilcoxon signed-rank test that does not pay the all-pairs multiplicity correction.

Friedman test.

The Friedman test rejects the null hypothesis of equal performance with $\chi^2(9) = 129.11$, $p < 10^{-6}$.

Nemenyi all-pairs procedure.

The critical difference at $\alpha = 0.05$ for $k = 10$ classifiers and $N = 48$ datasets is $CD = q_{0.05} \sqrt{k(k+1)/(6N)} \approx 1.955$.

Average ranks across the 48 datasets are reported in Table 9, and the critical-difference diagram in Figure 4.

Table 9. Average ranks across 48 OpenML-CC18 datasets (lower is better). Pairwise significance status against SVM-RBF under the Nemenyi post hoc procedure at $\alpha = 0.05$.

Classifier	Avg. Rank	Sig. vs. SVM-RBF
Random Forest	3.562	n.s.
XGBoost	3.760	n.s.
SVM-RBF	4.125	—
Gradient Boosting	4.677	n.s.
KNN-CV	4.760	n.s.
MLP	5.135	n.s.
Logistic Regression	6.260	$p = 0.020$
KNN-Optimal	6.708	$p = 0.001$
Decision Tree	7.958	$p < 0.001$
KNN-Classic	8.052	$p < 0.001$

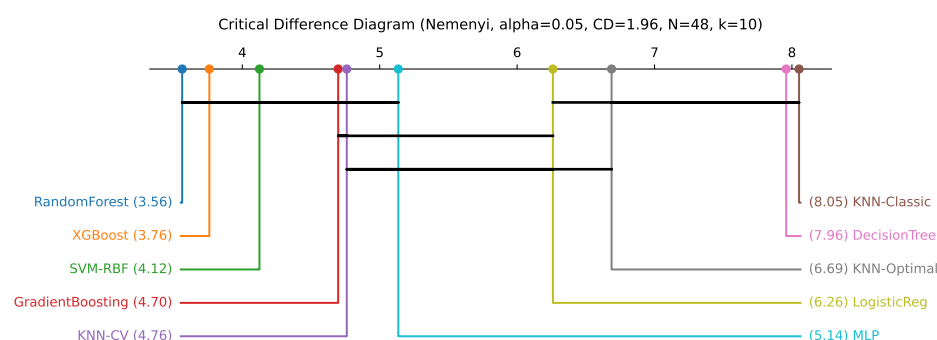


Figure 4. Critical-difference diagram for the Nemenyi post hoc test ($\alpha = 0.05$, $CD = 1.955$, $N = 48$ datasets, $k = 10$ classifiers). Classifiers connected by a horizontal bar are not significantly different.

Headline pairwise test: KNN-Optimal vs. KNN-Classic.

The Nemenyi rank gap between KNN-Optimal and KNN-Classic is $8.052 - 6.708 = 1.34$, just below the all-pairs critical difference of 1.955 at $\alpha = 0.05$. As Demsar [31] explicitly notes, the Nemenyi correction is conservative for pre-registered single comparisons because it pays the multiplicity cost of $\binom{k}{2}$ simultaneous pairwise tests. For the specific theorem-predicted comparison of KNN-Optimal against KNN-Classic, the appropriate test is the paired Wilcoxon signed-rank test on the 48 paired accuracies, which yields:

- Two-sided $p = 4.6 \times 10^{-4}$ ($W = 183$, mean diff +2.49 pp), where the statistic follows the SciPy convention $W = \min(W_+, W_-) = 183$ with $W_+ = 763$ (32 wins) and $W_- = 183$ (11 losses) over the 43 non-tie comparisons ($W_+ + W_- = 43 \cdot 44/2 = 946$). Under the null, $\mathbb{E}[W_+] = \mathbb{E}[W_-] = 473$; the observed asymmetry $W_+ : W_- \approx 4 : 1$ is itself a direct visualization of the directional effect.
- One-sided $p = 2.3 \times 10^{-4}$ (direction predicted by Theorem 2);
- Sign test on 43 non-tie pairs: $p = 9.6 \times 10^{-4}$ (32 of 43 favor KNN-Optimal);
- Paired Cohen's $d_z = 0.397$ (small).

All three significance tests agree at $p < 10^{-3}$. We conclude that the empirical advantage of the dimension-aware rule predicted by Theorem 2 is statistically robust, with a small but reliable effect size at the benchmark sample size of 48 datasets.

Other Nemenyi findings.

Beyond the headline pair, KNN-CV is statistically indistinguishable from SVM-RBF, Random Forest, XGBoost, Gradient Boosting, and MLP under the all-pairs Nemenyi procedure; KNN-Optimal is significantly worse than SVM-RBF ($p = 0.001$); and Decision Tree and KNN-Classic together sit at the bottom of the ranking, both significantly worse than the top group. The classical $\sqrt{n}$ rule, in other words, is empirically not just suboptimal in expectation—it is significantly worse than every other strategy tested except Decision Tree.

6. Practitioner Toolkit

To support the adoption of the methods described above, we provide a complete Python toolkit, `knn_unified.py`, whose core interface and a script-to-result reproducibility map are given in Appendix B. The toolkit automatically selects an optimal k for any dataset and is compatible with the scikit-learn API. The simplest usage is:

```
from knn_unified import recommend_k
k = recommend_k(X, y) # Returns optimal k with diagnostics
```

The toolkit includes:

1. Three k -selection methods: Classic ($k = \lfloor \sqrt{n} \rfloor$), Optimal ($k = \lfloor n^{4/(d+4)} \rfloor$), and cross-validated.
2. A `HassanatKNN` class compatible with scikit-learn (`cross_val_score`, `GridSearchCV`, etc.).
3. A `ClassifierBenchmark` utility for comparison with modern methods (Random Forest, SVM, XGBoost, MLP, and others).
4. Publication-quality figure generation reproducing Figures 1–4 of this paper.

Practical Guidelines

Practical suggestions for choosing k depending on data dimensionality are presented in Table 10. These recommendations are based on both the actual findings in Section 5 and the theoretical analysis in Section 2.

Table 10. Recommended k -selection strategy by dimensionality.

Dimensionality	Recommended k	Rationale
$d = 1\text{--}2$	$k \approx n^{0.7}$	Much larger than $\sqrt{n}$ needed
$d = 3\text{--}5$	$k \approx \sqrt{n}$	Classic rule (Theorem 1) applies
$d = 6\text{--}15$	$k \approx n^{0.35}$ *	Smaller than $\sqrt{n}$ needed
$d > 15$	Use cross-validation	Fixed rules less reliable
$d > 100$	Consider SVM/RF	KNN suffers curse of dimensionality

* The exact minimax-optimal exponent $4/(d+4)$ ranges from 0.40 at $d = 6$ to 0.21 at $d = 15$ across this regime. We round to $k \approx n^{0.35}$ as a convenient practical heuristic; practitioners who want tighter values can apply the dimension-aware formula $k^* = \lfloor n^{4/(d+4)} \rfloor$ directly.

In low-dimensional environments ($d = 1\text{--}2$), bigger neighbourhood sizes are needed to reduce variance and stabilise predictions. For middle dimensions ($d = 3\text{--}5$), the traditional rule $k = \sqrt{n}$ remains successful, consistent with the theoretical result that it is optimal at $d = 4$. As dimensionality grows ($d = 6\text{--}15$), smaller k values are preferred to avoid including irrelevant neighbours. Fixed rules are less accurate in higher-dimensional regimes ($d > 15$), where cross-validated k -selection is recommended. Finally, for very high-dimensional data ($d > 100$), KNN's efficacy suffers due to the curse of dimensionality, and alternative classifiers such as SVM or Random Forest may be more appropriate.

7. Discussion

7.1. Theoretical Contribution

Our central theoretical result—Theorem 2 together with Corollary 1—establishes that, under the second-order Hölder smoothness assumption, the classical $k = \sqrt{n}$ rule attains the asymptotic minimax-optimal exponent if and only if $d = 4$ (a statement about the rate exponent, not about the finite-sample optimal k), and quantifies the sub-optimality of the rule outside this special case as a sharp rate.

The penalty is asymmetric: low-dimensional regimes ($d < 4$) suffer a variance-dominated penalty that scales as $n^{1/2-4/(d+4)}$ in the sub-optimality ratio, while high-dimensional regimes ($d > 4$) suffer a bias-dominated penalty bounded above by approximately $n^{0.086}$ across the relevant range. The intersection $d = 4$ is therefore the single point at which the rule is optimal, a fact directly mirrored in the iris experiment of Section 5.5.

7.2. Why $\sqrt{n}$ Persists Despite Suboptimality

Our derivation also explains why the $\sqrt{n}$ rule has remained practically useful in spite of its formal sub-optimality. Many real-world tabular datasets occupy a moderate-dimensional regime ($d \in [4, 20]$) where the asymmetric penalty is bounded by a slow power of n . At $d = 10$, for example, the asymptotic sub-optimality ratio is approximately $n^{0.086}$, which at $n = 10^4$ yields a multiplicative regret penalty of only about 2.3. For many practical purposes this is small enough that the simplicity of $\sqrt{n}$ outweighs the small accuracy loss, especially in light of the variance introduced by other modeling choices. Outside the moderate-dimensional regime, however, the penalty grows quickly: at $d = 1$, the sub-optimality ratio is $n^{0.3}$, which at $n = 10^4$ yields a regret penalty of approximately 16.

7.3. KNN Versus Modern Classifiers

The benchmark of Section 5.6 places KNN-CV in the company of SVM-RBF, Random Forest, XGBoost, and Gradient Boosting under the Nemenyi all-pairs procedure: at $N = 48$, no significant difference exists among this top group. The price of avoiding cross-validation is, however, real and measurable. KNN-Optimal (the dimension-aware fixed rule) is significantly worse than the top group under Nemenyi ($p = 0.001$ vs. SVM-RBF, rank 6.71 vs. 4.13), and the absolute mean accuracy gap is 2.2 percentage points.

The practical interpretation is: cross-validate k when affordable; when cross-validation is computationally infeasible (e.g., very large n or per-query computation budgets), the dimension-aware fixed rule is a principled non-cross-validated alternative that significantly outperforms the classical $\sqrt{n}$ heuristic.

7.4. Why the Global Noise-Aware Adjustment Did Not Generalize

Earlier versions of this work proposed a noise-aware adjustment that interpolated k between k_{classic} and k_{optimal} via a global noise ratio computed from mutual information and Fisher discriminant statistics. We retain a brief discussion of this failed direction here because the failure is informative about the structure of the underlying problem.

In our diagnostic analysis on the 48 OpenML-CC18 datasets, the oracle-optimal interpolation parameter α^* (the value that maps to the cross-validation-optimal k via the interpolation formula) was strongly negatively correlated with n (Spearman $r = -0.61$, $p < 10^{-4}$). On large datasets the ideal α^* approaches 0, meaning the dimension-aware rule should be trusted directly; on small datasets the ideal α^* is closer to 0.5, meaning some interpolation toward the classical rule is warranted. A fixed-weight blend of MI and Fisher noise indicators cannot capture this systematic n -dependence, and any post hoc re-weighting of the two components yields only marginal improvement (the best fixed blend reduces the MAE between predicted and oracle α by only 3.4%).

The structural lesson is that the right blend between k_{classic} and k_{optimal} is itself a function of the sample size n , not just of data-quality summaries that can be computed in a single pass.

This observation motivates the future-work direction in Section 8.

7.5. Limitations and Scope

Three limitations of this work deserve explicit acknowledgment.

First, the asymptotic ordering predicted by Corollary 1 is not observable in our finite- n verification experiment. The slope match in Table 2 is clean across $d \geq 2$, but verifying that $R_n(\sqrt{n}) > R_n(k^*)$ in absolute terms requires sample sizes at which the bias term dominates the variance term—a regime that, for our smooth (linear) η , requires n larger than we test here. The translation of Corollary 1 into a benchmark-level superiority of k^* over $\sqrt{n}$ does manifest at $n \leq 50,000$ on real-world data (Section 5.6), but the mechanism by which this happens involves dataset-specific bias-variance trade-offs not captured by the synthetic regression function.

Second, our theoretical analysis assumes binary classification with Hölder smoothness; the benchmark of Section 5.6 includes datasets with up to 26 classes. Extending the rate analysis to multi-class settings with margin assumptions [25] is a natural next step.

Third, the standard KNN classifier has $O(n)$ prediction cost, which limits the practical applicability of all three KNN variants compared here to small- and medium-scale datasets. Constant-time KNN ensemble architectures such as CTELC [32] bypass this ceiling by aggregating weak KNN classifiers on small random chunks. The dimension-aware k -selection of this paper applies directly to each weak learner.

8. Conclusions and Future Work

This work revisits the classical $k = \sqrt{n}$ rule of thumb in k -nearest neighbor classification. We prove (Theorem 1) that the rule is minimax-optimal under Hölder smoothness if and only if the feature dimension is $d = 4$, and we quantify the rate of sub-optimality of any fixed polynomial exponent $k = n^\beta$ via Theorem 2 and Corollary 1, with an explicit asymmetric penalty for the $\sqrt{n}$ rule when $d \neq 4$. The predicted convergence rate is empirically verified on synthetic data across $d \in \{2, \dots, 20\}$, and on 48 OpenML-CC18 datasets the dimension-aware rule $k = \lfloor n^{4/(d+4)} \rfloor$ outperforms the classical rule with small but statistically reliable effect size (paired Wilcoxon $p = 4.6 \times 10^{-4}$, mean accuracy gap +2.5 percentage points).

The Hassanat distance metric provides a complementary improvement, outperforming Euclidean across all k -rules and especially on unstandardized data.

Three future directions emerge from this analysis.

Sample-size-aware adaptive weighting.

Our diagnostic analysis identified a strong negative correlation between the oracle-optimal interpolation between k_{classic} and k_{optimal} and the sample size n (Spearman $r = -0.61$, $p < 10^{-4}$). A principled n -aware interpolation of the form

$$\alpha(n, d) = \sigma\left(\frac{\log n - \log n_0(d)}{\tau}\right),$$

where σ is a sigmoid, $n_0(d)$ is the crossover point of the variance- and bias-dominated regret terms predicted by Theorem 2, and τ is a transition-sharpness parameter derived from the rate analysis, is the most direct theoretical extension of this work. We identify this as the central question for follow-up work.

Locally-adaptive k -selection.

Theorem 2 and the dimension-aware rule treat k as a global quantity determined by the dataset-level parameters (n, d) . In settings where the marginal density $p(x)$ varies sharply across the support—manifold-supported data, mixture distributions, or class-imbalanced regions—the locally optimal k at a query point x can differ substantially from the global recommendation. Extending the rate analysis to local $k(x)$ that incorporates local density and local margin [14] is a natural next direction.

Constant-time KNN ensembles.

For very large datasets where $O(n)$ prediction cost is prohibitive, ensemble architectures such as CTELC [32] train each weak KNN classifier on a small random chunk of size $m \ll n$, placing the weak learners precisely in the small-sample regime (with m playing the role of the training size) where the dimension-aware rule of Theorem 2 applies most directly. Integrating the dimension-aware k -selection into CTELC promises both billion-sample scalability and a clean theoretical match between the rule and the data subset on which each weak learner operates.

Author Contributions: Conceptualization, A.S.T. and O.L.; methodology, A.B.H.; software, A.S.T.; validation, A.S.T., E.A., O.L. and A.S.T.; formal analysis, A.B.H., A.S.T. and K.A.; investigation, A.B.H., A.S.T., E.A., M.A. (Malek Alrashidi), O.L., A.A.A., K.A., A.A. and M.A. (Maha Alamri); resources, O.L. and A.A.; data curation, A.S.T. and A.A.A.; writing—original draft preparation, A.B.H., A.S.T., E.A., M.A. (Malek Alrashidi), O.L., A.A.A., K.A., A.A. and M.A. (Maha Alamri); writing—review and editing, A.B.H., A.S.T., E.A., M.A. (Malek Alrashidi), O.L., A.A.A., K.A., A.A. and M.A. (Maha Alamri); visualization, A.S.T.; supervision, O.L. and A.B.H.; project administration, O.L. and A.B.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original data presented in this study are openly available in the OpenML repository as the OpenML-CC18 benchmarking suite [28] at <https://www.openml.org/s/99> (suite ID 99) (accessed on 20 July 2026). The synthetic data were generated by the Python code described in the paper; this code, together with the implementation of the dimension-aware k -selection rule and the Hassanat distance, is openly available in the accompanying toolkit, permanently archived at Zenodo, <https://doi.org/10.5281/zenodo.20173214>.

Acknowledgments: The authors thank the Faculty of Information Technology at Mutah University for providing computational resources. Additional acknowledgments go to the Deanship of Scientific Research at Mutah University for supporting the fourth author (Omar Lasassmeh) under grant No. 2025/1051.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Detailed Per-Dataset Results

This appendix lists the 48 OpenML-CC18 datasets used in Section 5.6, the per-dataset cross-validation accuracies of the three KNN variants and the three top-performing baseline

classifiers, and the per-dataset head-to-head comparison underlying the paired Wilcoxon test for KNN-Optimal versus KNN-Classic reported in Section 5.8.

Appendix A.1. List of 48 Datasets (Table A1)

Table A1. The 48 OpenML-CC18 datasets used in Section 5.6, sorted by sample size n . Columns: n , ambient dimensionality d , number of classes, and the three k -values used by the three KNN rules tested (KNN-Classic, KNN-Optimal, KNN-CV).

Dataset	n	d	Cl.	k_{classic}	k_{optimal}	k_{CV}
kc2	522	21	2	22	2	49
climate-model-simulation-crashes	540	18	2	23	3	7
wdbc	569	30	2	23	2	3
ilpd	583	10	2	24	6	47
balance-scale	625	4	3	25	25	21
blood-transfusion-service-center	748	4	2	27	27	13
diabetes	768	8	2	27	9	49
analcata_data_dmft	797	4	6	28	28	47
analcata_data_authorship	841	70	4	29	1	9
vehicle	846	18	4	29	3	11
tic-tac-toe	958	9	2	30	8	5
vowel	990	12	11	31	5	1
credit-g	1000	20	2	31	3	11
qsar-biodeg	1055	41	2	32	1	5
pc1	1109	21	2	33	3	11
banknote-authentication	1372	4	2	37	37	3
pc4	1458	37	2	38	2	9
cmc	1473	9	3	38	9	25
pc3	1563	37	2	39	2	35
car	1728	6	4	41	19	5
steel-plates-fault	1941	27	7	44	2	5
mfeat-zernike	2000	47	10	44	1	13
mfeat-morphological	2000	6	10	44	20	13
mfeat-fourier	2000	76	10	44	1	9
mfeat-karhunen	2000	64	10	44	1	1
kc1	2109	21	2	45	3	33
segment	2310	16	7	48	4	3
ozone-level-8hr	2534	72	2	50	1	15
dna	3186	180	3	56	1	49
splice	3190	60	3	56	1	39
kr-vs-kp	3196	36	2	56	2	3
spambase	4601	57	2	67	1	1
wilt	4839	5	2	69	43	3
churn	5000	20	2	70	4	5
phoneme	5404	5	2	73	45	1
wall-robot-navigation	5456	24	4	73	3	1
texture	5500	40	11	74	2	1
optdigits	5620	64	10	74	1	3
first-order-theorem-proving	6118	51	6	78	1	1
satimage	6430	36	6	80	2	3
GesturePhaseSegmentationProcessed	9873	32	5	99	2	1
pendigits	10,992	16	10	104	6	3
PhishingWebsites	11,055	30	2	105	2	1
letter	20,000	16	26	141	7	1
nomao	34,465	118	2	185	1	3
jungle_chess_2pcs_raw_endgame_complete	44,819	6	3	211	72	25
bank-marketing	45,211	16	2	212	8	19
electricity	45,312	8	2	212	35	3

Appendix A.2. Per-Dataset Accuracies (Table A2)

Table A2. Per-dataset accuracies (stratified 10-fold cross-validation) for the three KNN variants and the three top-performing baseline classifiers. Datasets are sorted by n (same order as Table A1).

Dataset	KNN-Cla.	KNN-Opt.	KNN-CV	RF	XGB	SVM-RBF
kc2	0.8238	0.8333	0.8335	0.8163	0.8238	0.8354
climate-model-simulation-crashes	0.9148	0.9167	0.9315	0.9222	0.9500	0.9296
wdbc	0.9543	0.9561	0.9666	0.9561	0.9632	0.9754
ilpd	0.6964	0.7032	0.7101	0.7237	0.7082	0.7136
balance-scale	0.8960	0.8960	0.8912	0.8290	0.8529	0.9024
blood-transfusion-service-center	0.7954	0.7954	0.7861	0.7366	0.7687	0.7807
diabetes	0.7382	0.7486	0.7552	0.7734	0.7603	0.7617
analcadata_dmft	0.2383	0.2383	0.2094	0.1781	0.1944	0.2121
analcadata_authorship	0.9941	0.9941	0.9964	0.9917	0.9833	0.9953
vehicle	0.6880	0.7080	0.7011	0.7459	0.7506	0.7660
tic-tac-toe	0.7526	0.8486	0.8560	0.9540	0.9969	0.8998
vowel	0.5939	0.9414	0.9899	0.9727	0.9081	0.9596
credit-g	0.7210	0.7060	0.7140	0.7510	0.7710	0.7360
qsar-biodeg	0.8502	0.8418	0.8578	0.8796	0.8673	0.8749
pc1	0.9297	0.9197	0.9351	0.9387	0.9351	0.9324
banknote-authentication	0.9898	0.9898	0.9985	0.9934	0.9934	1.0000
pc4	0.8903	0.8937	0.9026	0.9102	0.9095	0.9060
cmc	0.5173	0.5166	0.5187	0.5200	0.5370	0.5478
pc3	0.8976	0.8906	0.8970	0.9034	0.8925	0.8983
car	0.8258	0.8871	0.9479	0.9826	0.9890	0.9184
steel-plates-fault	0.6888	0.7218	0.7444	0.7857	0.8052	0.7609
mfeat-karhunen	0.9195	0.9580	0.9580	0.9585	0.9470	0.9780
mfeat-morphological	0.6965	0.7155	0.7245	0.7090	0.7005	0.7190
mfeat-zernike	0.7930	0.7895	0.8065	0.7855	0.7725	0.8350
mfeat-fourier	0.7695	0.7905	0.8115	0.8320	0.8370	0.8245
kc1	0.8535	0.7966	0.8559	0.8620	0.8592	0.8540
segment	0.8424	0.8987	0.9013	0.9411	0.9342	0.8814
ozone-level-8hr	0.9369	0.9258	0.9408	0.9432	0.9455	0.9380
dna	0.8443	0.7288	0.8418	0.9548	0.9601	0.9614
splice	0.6831	0.6498	0.6837	0.9517	0.9605	0.8702
kr-vs-kp	0.9202	0.9190	0.9521	0.9922	0.9941	0.9865
spambase	0.8829	0.9150	0.9150	0.9550	0.9554	0.9339
wilt	0.9475	0.9494	0.9711	0.9833	0.9851	0.9808
churn	0.8652	0.8840	0.8940	0.9572	0.9594	0.9210
phoneme	0.8281	0.8362	0.9040	0.9128	0.8932	0.8446
wall-robot-navigation	0.7527	0.8752	0.8856	0.9952	0.9969	0.8864
texture	0.9440	0.9836	0.9895	0.9795	0.9824	0.9913
optdigits	0.9504	0.9786	0.9797	0.9826	0.9762	0.9875
first-order-theorem-proving	0.5109	0.5870	0.5870	0.6335	0.6180	0.5350
satimage	0.8622	0.9008	0.9082	0.9182	0.9182	0.8986
GesturePhaseSegmentationProcessed	0.5071	0.5807	0.6382	0.6756	0.6460	0.5371
pendigits	0.9568	0.9926	0.9941	0.9914	0.9897	0.9949
PhishingWebsites	0.9124	0.9492	0.9645	0.9721	0.9628	0.9524
letter	0.8081	0.9449	0.9556	0.9651	0.9519	0.9489
nomao	0.9262	0.9553	0.9556	0.9696	0.9684	0.9614
jungle_chess_2pcs_raw_endgame_complete	0.7980	0.8228	0.8697	0.7817	0.8581	0.8269
bank-marketing	0.8906	0.8935	0.8965	0.9054	0.9084	0.8985
electricity	0.7910	0.8147	0.8405	0.9109	0.8734	0.8004

Appendix A.3. Per-Dataset KNN-Optimal vs. KNN-Classic (Table A3)

Table A3. Per-dataset head-to-head accuracy of KNN-Optimal versus KNN-Classic on each of the 48 OpenML-CC18 datasets. Gap is KNN-Optimal minus KNN-Classic accuracy; the arrow column gives a visual win/lose indicator (↑: KNN-Optimal wins; ↓: KNN-Classic wins; =: tie). Across all 48 datasets, 32 of 48 favor KNN-Optimal, 11 favor KNN-Classic, and 5 are ties. The same data underlies the paired Wilcoxon test of Section 5.8.

Dataset	<i>n</i>	<i>d</i>	KNN-Cla.	KNN-Opt.	Gap (O – C)	
kc2	522	21	0.8238	0.8333	+0.0095	↑
climate-model-simulation-crashes	540	18	0.9148	0.9167	+0.0019	↑
wdbc	569	30	0.9543	0.9561	+0.0018	↑
ilpd	583	10	0.6964	0.7032	+0.0067	↑
balance-scale	625	4	0.8960	0.8960	+0.0000	=
blood-transfusion-service-center	748	4	0.7954	0.7954	+0.0000	=
diabetes	768	8	0.7382	0.7486	+0.0105	↑
analcatdata_dmft	797	4	0.2383	0.2383	+0.0000	=
analcatdata_authorship	841	70	0.9941	0.9941	+0.0000	=
vehicle	846	18	0.6880	0.7080	+0.0200	↑
tic-tac-toe	958	9	0.7526	0.8486	+0.0960	↑
vowel	990	12	0.5939	0.9414	+0.3475	↑
credit-g	1000	20	0.7210	0.7060	−0.0150	↓
qsar-biodeg	1055	41	0.8502	0.8418	−0.0084	↓
pc1	1109	21	0.9297	0.9197	−0.0099	↓
banknote-authentication	1372	4	0.9898	0.9898	+0.0000	=
pc4	1458	37	0.8903	0.8937	+0.0034	↑
cmc	1473	9	0.5173	0.5166	−0.0006	↓
pc3	1563	37	0.8976	0.8906	−0.0071	↓
car	1728	6	0.8258	0.8871	+0.0613	↑
steel-plates-fault	1941	27	0.6888	0.7218	+0.0330	↑
mfeat-karhunen	2000	64	0.9195	0.9580	+0.0385	↑
mfeat-morphological	2000	6	0.6965	0.7155	+0.0190	↑
mfeat-zernike	2000	47	0.7930	0.7895	−0.0035	↓
mfeat-fourier	2000	76	0.7695	0.7905	+0.0210	↑
kc1	2109	21	0.8535	0.7966	−0.0569	↓
segment	2310	16	0.8424	0.8987	+0.0563	↑
ozone-level-8hr	2534	72	0.9369	0.9258	−0.0110	↓
dna	3186	180	0.8443	0.7288	−0.1155	↓
splice	3190	60	0.6831	0.6498	−0.0332	↓
kr-vs-kp	3196	36	0.9202	0.9190	−0.0012	↓
spambase	4601	57	0.8829	0.9150	+0.0322	↑
wilt	4839	5	0.9475	0.9494	+0.0019	↑
churn	5000	20	0.8652	0.8840	+0.0188	↑
phoneme	5404	5	0.8281	0.8362	+0.0081	↑
wall-robot-navigation	5456	24	0.7527	0.8752	+0.1224	↑
texture	5500	40	0.9440	0.9836	+0.0396	↑
optdigits	5620	64	0.9504	0.9786	+0.0283	↑
first-order-theorem-proving	6118	51	0.5109	0.5870	+0.0760	↑
satimage	6430	36	0.8622	0.9008	+0.0386	↑
GesturePhaseSegmentationProcessed	9873	32	0.5071	0.5807	+0.0735	↑
pendigits	10,992	16	0.9568	0.9926	+0.0358	↑
PhishingWebsites	11,055	30	0.9124	0.9492	+0.0367	↑
letter	20,000	16	0.8081	0.9449	+0.1369	↑
nomao	34,465	118	0.9262	0.9553	+0.0291	↑
jungle_chess_2pcs_raw_endgame_complete	44,819	6	0.7980	0.8228	+0.0248	↑
bank-marketing	45,211	16	0.8906	0.8935	+0.0028	↑
electricity	45,312	8	0.7910	0.8147	+0.0237	↑

Appendix B. Practitioner Toolkit

All methods used in this paper are implemented in a small, self-contained Python toolkit compatible with the scikit-learn API. Rather than reproduce the full source in-line, we give its core interface here; the complete toolkit is archived at Zenodo (DOI: 10.5281/zenodo.20173214; see Data Availability).

Core Interface

The two essential ingredients—the dimension-aware neighbour count and the bounded Hassanat distance—are reproduced below; the full toolkit (`knn_unified.py`) wraps these in a scikit-learn-compatible estimator.

```
import numpy as np
from sklearn.neighbors import KNeighborsClassifier

def recommend_k(n, d, rule='optimal'):
    """Dimension-aware k* or classical sqrt(n) neighbour count."""
    if rule == 'classic':
        return max(1, int(np.floor(np.sqrt(n))))
    return max(1, int(np.floor(n ** (4.0 / (d + 4))))) # k* = n^{4/(d+4)}

def hassanat_distance(x, y):
    """Bounded Hassanat distance in [0, d]; handles negative features."""
    x, y = np.asarray(x, float), np.asarray(y, float)
    mn, mx = np.minimum(x, y), np.maximum(x, y)
    pos = mn >= 0
    D = np.where(pos,
                  1.0 - (1.0 + mn) / (1.0 + mx),
                  1.0 - (1.0 + mn + np.abs(mn)) / (1.0 + mx + np.abs(mn)))
    return float(np.sum(D))

# n, d = X_train.shape
# clf = KNeighborsClassifier(n_neighbors=recommend_k(n, d)).fit(X_train,
# y_train)
```

References

1. Fix, E.; Hodges, J.L. *Discriminatory Analysis: Nonparametric Discrimination, Consistency Properties*; Technical Report Project 21-49-004, Report No. 4; USAF School of Aviation Medicine: Randolph Field, TX, USA, 1951.
2. Cover, T.; Hart, P. Nearest neighbor pattern classification. *IEEE Trans. Inf. Theory* **1967**, *13*, 21–27. [\[CrossRef\]](#)
3. Halder, R.K.; Uddin, M.N.; Uddin, M.A.; Aryal, S.; Khraisat, A. Enhancing K-nearest neighbor algorithm: A comprehensive review and performance analysis of modifications. *J. Big Data* **2024**, *11*, 113. [\[CrossRef\]](#)
4. Mrudula, S.T.; Ritonga, M.; Sivakumar, S.; Jawarneh, M.; Keerthika, T.; Rane, K.P.; Roy, B. Internet of things and optimized knn based intelligent transportation system for traffic flow prediction in smart cities. *Meas. Sens.* **2024**, *35*, 101297. [\[CrossRef\]](#)
5. Altamimi, A.; Alarfaj, A.A.; Umer, M.; Alabdulqader, E.A.; Alsubai, S.; Kim, T.; Ashraf, I. An automated approach to predict diabetic patients using KNN imputation and effective data mining techniques. *BMC Med. Res. Methodol.* **2024**, *24*, 221. [\[CrossRef\]](#) [\[PubMed\]](#)
6. Nguyen, L.V.; Vo, Q.; Nguyen, T. Adaptive knn-based extended collaborative filtering recommendation services. *Big Data Cogn. Comput.* **2023**, *7*, 106. [\[CrossRef\]](#)
7. Aljarah, I.; Alzaqebah, A.; Al-Madi, N.; Al-Zoubi, A.; Saleh, A. mDA: Evolutionary Machine Learning Algorithm for Feature Selection in Medical Domain. *Computation* **2025**, *13*, 292. [\[CrossRef\]](#)
8. Hamadneh, T.; Montazeri, Z.; Dehghani, M.; Smerat, A.; Humod, A.T.; AlShboul, R.; Malik, O.P.; Eguchi, K. Repairman Algorithm (RA): A Novel Human-based Metaheuristic Algorithm for Solving Optimization Problems. *Int. J. Intell. Eng. Syst.* **2026**, *19*, 945. [\[CrossRef\]](#)

9. Seyghaly, R.; Garcia, J.; Masip-Bruin, X.; Kuljanin, J. SBNRR: Small-size bat-optimized KNN regression. *Future Internet* **2024**, *16*, 422. [\[CrossRef\]](#)
10. Acito, F. *Predictive Analytics with KNIME: Analytics for Citizen Data Scientists*, 1st ed.; Springer: Cham, Switzerland, 2023. [\[CrossRef\]](#)
11. Farid, D.M.; Sworna, N.S.; Amin, R.; Sadia, N.; Rahman, M.; Liton, N.K.; Mukta, M.S.H.; Shatabda, S. Boosting k-nearest neighbour (knn) classification using clustering and adaboost methods. In *Proceedings of the 2022 IEEE Region 10 Symposium (TENSYP)*; IEEE: New York, NY, USA, 2022.
12. Alfeilat, H.A.A.; Hassanat, A.B.; Lasassmeh, O.; Tarawneh, A.S.; Alhasanat, M.B.; Salman, H.S.E.; Prasath, V.S. Effects of distance measure choice on k-nearest neighbor classifier performance: A review. *Big Data* **2019**, *7*, 221–248. [\[CrossRef\]](#) [\[PubMed\]](#)
13. Jirina, M.; Jirina, M., Jr. Classifiers based on inverted distances. In *New Fundamental Technologies in Data Mining*; Funatsu, K., Ed.; InTech: Rijeka, Croatia, 2011; Chapter 19, pp. 369–387.
14. Anava, O.; Levy, K.Y. k*-Nearest neighbors: From global to local. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, Barcelona, Spain, 5–10 December 2016; Volume 29, pp. 4916–4924.
15. Hassanat, A.B.; Abbadi, M.A.; Altarawneh, G.A.; Alhasanat, A.A. Solving the problem of the K parameter in the KNN classifier using an ensemble learning approach. *arXiv* **2014**, arXiv:1409.0919.
16. Ruziq, F.; Wayahdi, M.R.; Ginting, S.H.N. Diabetes Prediction Based on Medical Records (Pima Indians Diabetes Dataset) Using K-NN. *J. Sci. Soc. Res.* **2025**, *8*, 3776–3782.
17. Al-Khamees, H.A.A.; Sani, N.S.; Gifal, A.S.; Liu, L.X.W.; Esa, M.I. A dynamic model using k-NN algorithm for predicting diabetes and breast cancer. *Comput. Biol. Med.* **2025**, *192*, 110276. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Sudipa, I.G.I.; Azdy, R.A.; Arfiani, I.; Setiohardjo, N.M.; Sumiyatun. Leveraging K-Nearest Neighbors for Enhanced Fruit Classification and Quality Assessment. *Indones. J. Data Sci.* **2024**, *5*, 30–36. [\[CrossRef\]](#)
19. Agbedanu, P.R.; Yang, S.J.; Musabe, R.; Gatare, I.; Rwigema, J. A Scalable Approach to Internet of Things and Industrial Internet of Things Security: Evaluating Adaptive Self-Adjusting Memory K-Nearest Neighbor for Zero-Day Attack Detection. *Sensors* **2025**, *25*, 216. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Bao, H.; Gao, J. Network intrusion detection based on improved KNN algorithm. *Sci. Rep.* **2025**, *15*, 29842. [\[CrossRef\]](#) [\[PubMed\]](#)
21. Çetin, A.İ.; Büyüklü, A.H. A new approach to K-nearest neighbors distance metrics on sovereign country credit rating. *Kuwait J. Sci.* **2025**, *52*, 100324. [\[CrossRef\]](#)
22. Othman, M.A.; Raad, A.; Ghandour, A. Loan Repayment Default Prediction Using Supervised Machine Learning Techniques on Financial Data. *InfoTech Spectr. Iraqi J. Data Sci.* **2025**, *2*, 41–58. [\[CrossRef\]](#)
23. Devroye, L.; Györfi, L.; Lugosi, G. *A Probabilistic Theory of Pattern Recognition*; Stochastic Modelling and Applied Probability; Springer: Berlin/Heidelberg, Germany, 1996; Volume 31.
24. Samworth, R. Optimal weighted nearest neighbour classifiers. *Ann. Stat.* **2012**, *40*, 2733–2763. [\[CrossRef\]](#)
25. Audibert, J.Y.; Tsybakov, A.B. Fast learning rates for plug-in classifiers. *Ann. Stat.* **2007**, *35*, 608–633. [\[CrossRef\]](#)
26. Hassanat, A.; Alkafaween, E.A.; Tarawneh, A.S.; Elmougy, S. Applications review of hassanat distance metric. In *Proceedings of the International Conference on Emerging Trends in Computing and Engineering Applications (ETCEA)*, Karak, Jordan, 23–25 November 2022.
27. Tarawneh, A.S. HasLoss: A novel Hassanat distance-based loss functions for binary classification. *Front. Artif. Intell.* **2026**, *8*, 1690830. [\[PubMed\]](#)
28. Bischl, B.; Casalicchio, G.; Feurer, M.; Gijsbers, P.; Hutter, F.; Lang, M.; Mantovani, R.G.; van Rijn, J.N.; Vanschoren, J. OpenML Benchmarking Suites. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, Vancouver, BC, Canada 6–14 December 2021; Volume 1.
29. Hassanat, A.B.; Alqaralleh, M.K.; Tarawneh, A.S.; Almohammadi, K.; Alamri, M.; Alzahrani, A.; Altarawneh, G.A.; Alhalaseh, R. A Novel Outlier-Robust Accuracy Measure for Machine Learning Regression Using a Non-Convex Distance Metric. *Mathematics* **2024**, *12*, 3623. [\[CrossRef\]](#)
30. Kelly, M.; Longjohn, R.; Nottingham, K. The UCI Machine Learning Repository. Available online: <https://archive.ics.uci.edu> (accessed on 29 April 2026).
31. Demšar, J. Statistical Comparisons of Classifiers over Multiple Data Sets. *J. Mach. Learn. Res.* **2006**, *7*, 1–30.
32. Tarawneh, A.S.; Alamri, E.S.; Al-Saedi, N.N.; Alauthman, M.; Hassanat, A.B. CTELC: A Constant-Time Ensemble Learning Classifier Based on KNN for Big Data. *IEEE Access* **2023**, *11*, 89791–89802. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.