Relative Relation in KNN Classification for Gene Expression Data. A Preliminary Study

Izabela Justyna Kartowicz-Stolarska
Faculty of Computer Science Bialystok University of Technology
Bialystok, Poland *i.stolarska@pb.edu.pl*

Marcin Czajkowski
Faculty of Computer Science Bialystok University of Technology
Bialystok, Poland *m.czajkowski@pb.edu.pl*

Abstract

This paper introduces an innovative approach to the classification of gene expression data using the k-nearest neighbors (KNN) algorithm. High dimensionality and limited sample sizes continue to present significant challenges for conventional classification techniques, including KNN. In response, we propose the Relative Relation Metric (RRM), a novel metric that diverges from traditional distances which typically rely on direct numerical or spatial comparisons. RRM instead focuses on the count of relational changes between pairs of data points, drawing conceptual inspiration from Relative Expression Analysis, which identifies the most discriminating gene pairs between classes, and Kendall's Tau. Applied to real gene expression datasets for disease classification and compared with established metrics, our preliminary study suggests that RRM has potential as an effective alternative for high-dimensional data classification, especially in contexts requiring resistance to methodological variations and the transformational aspects of biological data.

Keywords: knn, relative expression analysis, classification, gene expression data

1. Introduction

The swift expansion in gene expression data, driven by advancements in genomic technologies, heralds a significant transformation in biomedical research. This surge of data brings the promise of deeper insights into the genetics of diseases but introduces substantial analytical challenges, primarily due to its complex and high-dimensional nature alongside limited sample sizes [22]. Traditional computational methods often find themselves at a disadvantage, hampered by these factors and the intricate character of biological data [20]. Addressing these challenges necessitates agile and innovative approaches in information systems development (ISD), especially in an era reshaped by post-COVID-19 adjustments and the advent of generative AI technologies.

In this context, we introduce the Relative Relation Metric (RRM) within the KNN algorithm [34] to specifically address these challenges in gene expression data analysis. The development of RRM, inspired by both Relative Expression Analysis (RXA) [14] and Kendall's Tau distance [37], marks a strategic shift from traditional numerical comparisons to relational assessments. This investigation evaluates RRM's ability to manage the complexities of gene expression data, supporting KNN to be robust against standard preprocessing methods like normalization and standardization, and potentially to improve its post-hoc knowledge extraction. By comparing RRM with traditional metrics in disease classification, we examine its potential as a novel method for high-dimensional data analysis.

The paper is structured as follows: it begins with a background section that situates our study within the broader fields of Artificial Intelligence (AI) and Data Mining, highlighting

the specific challenges posed by omics data analysis. This is followed by a detailed exposition of the methodology underpinning the application of RRM within the KNN framework, our experimental validation using real-world gene expression datasets.

Moreover, we have made the implementation of the algorithm and the description of the experiments available on the public gitlab repository. Through this discussion, we aim to highlight our contributions and contemplate their broader implications for ISD in a world increasingly influenced by data and AI-enhanced technologies.

2. Background

This section provides a comprehensive overview of the computational methods used in gene expression data analysis, focusing on the k-nearest neighbors (KNN) algorithm and the associated challenges in handling high-dimensional biological data. Additionally, we introduce the principles of Relative Expression Analysis (RXA), which emphasize the relational aspects of gene expressions crucial for understanding complex genetic networks.

2.1. KNN Algorithm: Fundamentals

The k-nearest neighbors (KNN) algorithm [34] is a cornerstone of machine learning, widely utilized for its simplicity and efficacy in classification tasks. It predicts the classification of a new sample based on the majority vote from its closest neighbors in the feature space.

- **Euclidean Distance:** The standard metric for KNN, calculated as $d(\text{vector1}, \text{vector2}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$.
- **Manhattan Distance (L1 norm):** Computes the sum of the absolute differences between coordinates, $\sum_{i=1}^{n} |x_i - y_i|$.
- **Minkowski Distance:** A generalization of Euclidean and Manhattan distances, $(\sum_{i=1}^{n} |x_i - y_i|^p)^{1/p}$, where $pge1$ and p is a real number.
- **Chebyshev Distance:** Calculates the maximum difference along any coordinate dimension, $d(\text{vector1}, \text{vector2}) = \max_i |x_i - y_i|$, which makes it particularly useful in scenarios where a single large difference is more significant than smaller differences in multiple dimensions.
- **Kendall's Tau [37]:** This metric measures the concordance of rankings between two datasets, defined as $\tau = \frac{2}{n(n-1)} \sum_{i<j} \text{sgn}(x_i - x_j)\text{sgn}(y_i - y_j)$, where sgn is the sign function, indicating the similarity in the order of data points and n is a number of data points.

The computational demand of KNN is significant, as it involves calculating distances to all training instances for each query, which can be particularly challenging in large datasets. Selecting the optimal k is essential for balancing sensitivity to noise against smoothing over data features. A common heuristic is the square root of the number of samples, with k set as an odd number to prevent ties. Nonetheless, its non-parametric nature makes it highly flexible, proving valuable in complex gene expression analyses where relationships between data points often hold more significance than the absolute values themselves.

2.2. Relative Expression Analysis (RXA)

Relative Expression Analysis (RXA) is a computational approach that emphasizes the relational order of gene expressions rather than their absolute magnitudes. This method is particularly adept at handling the biases and normalization issues common in gene expression analysis,

making it a robust tool for revealing regulatory patterns and understanding complex gene networks [16, 9].

The core of RXA is the Top Scoring Pair (TSP) technique, which focuses on the relationships between gene pairs within a sample. For instance, if one gene is expressed more than another in a disease state compared to a normal state, this relationship can indicate a potential biological switch [33, 14]. This method is expressed mathematically as:

$$\Delta_{ij} = \left| P_{ij}(\text{normal}) - P_{ij}(\text{disease}) \right|, \tag{1}$$

where Δ_{ij} quantifies the change in expression relationship between genes x_i and x_j across conditions. Such insights are invaluable for genomics and are increasingly being applied in proteomics and metabolomics [17].

2.3. Approaches and Challenges

Gene expression data analysis grapples with high-dimensional and intricate datasets, typically characterized by the "small n, large p" dilemma, which denotes a large number of variables (genes) in contrast to a relatively small number of samples. This disproportion often leads to complications in model training and a heightened risk of overfitting, posing significant challenges for traditional computational methods [20, 26]. Recent research on gene expression analysis has focused on solving the above problems using both conventional and deep machine learning-based approaches [4, 3, 28]. The k-nearest neighbors (KNN) algorithm, despite its simplicity, may be adapted to tackle these datasets by leveraging its ability to classify based on proximity in the feature space, which can provide meaningful biological insights [24, 1]. However, the conventional Euclidean metric often used in KNN loses its effectiveness in high-dimensional spaces, leading to the introduction of dimensionality reduction techniques like PCA and t-SNE before applying KNN [35]. Moreover, adaptations like Weighted KNN and the development of specialized distance metrics aim to better reflect biological or functional similarities, enhancing classification accuracy in these complex datasets [24]. Despite these advancements, fine-tuning KNN's parameters, such as the number of neighbors and the choice of distance metric, remains critical due to the inherent variability in gene expression data, necessitating robust cross-validation strategies to verify model performance [30].

The motivation for developing the Relative Relation Metric (RRM) arises from the need for robust computational strategies that maintain predictive accuracy and interpretability amidst the challenges posed by high-dimensional, noisy, and sparse gene expression data. Conventional metrics like Euclidean or Manhattan distances, which directly compare numerical values, are prone to being misled by outliers and scale variations. These issues are exacerbated by the heterogeneous nature of biological experiments, where differences in sample preparation and sequencing technologies can introduce significant variability. Proposed RRM approach aligns with the need for metrics that can robustly classify high-dimensional biological data without being overly sensitive to the problems typical in gene expression analysis.

3. Relative Relation Metric

The Relative Relation Metric (RRM) is inspired by the principles of Relative Expression Analysis (RXA) and the ordinal nature of Kendall's Tau, prioritizing the consistency of relational changes over absolute magnitudes. This approach diminishes the impact of noise and experimental variability inherent in gene expression data.

Figure 1 illustrates how RRM selects neighbors based on the relational ordering of feature expressions rather than their numerical values. The test sample Y and four training samples A, B, C, and D, each with six features (X_1 to X_6), are presented. The relational orderings within each instance are depicted, with the test sample's features ranked according to their ex-

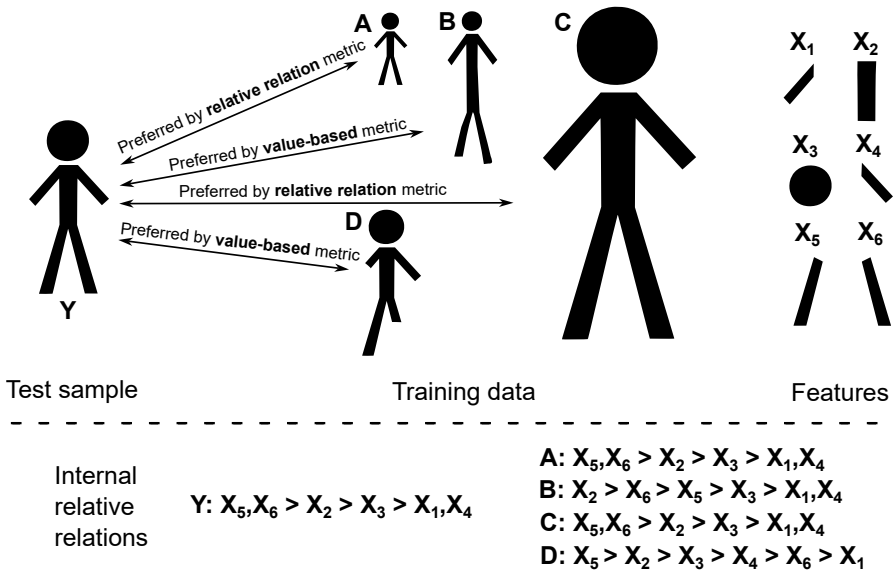

Fig. 1. Conceptual representation of the Relative Relation Metric (RRM) within the k-nearest neighbors (KNN) algorithm, highlighting the selection of neighbors based on feature relation consistency.

pressions. For instance, Y shows X_5 and X_6 as the most expressed features, followed by X_2, X_3, X_1, and X_4. RRM identifies training data A and C as having relational orderings akin to the test sample, favoring them over B and D, which a value-based metric might prefer due to closer numerical proximity. This selection method echoes the ranking approach of Kendall's Tau while also aligning with the sequence pair relations found in RXA algorithm families, suggesting a relational rather than absolute interpretative framework.

Moreover, the test sample is not directly compared with any training data. RRM examines whether, and to what extent, the relational orderings present in one sample occur in others. This perspective allows for the inclusion of test samples that have not undergone similar preprocessing steps, such as normalization or standardization, making it particularly appealing for biomedical data with varying protocols and standards.

3.1. Algorithm

The RRM algorithm is based on comparing two n-dimensional vectors of real numbers. The mathematical formula can be represented as:

Let vector1 $= [x_1, x_2, \ldots, x_n]$ and vector2 $= [y_1, y_2, \ldots, y_n]$ be two vectors of real numbers of length n. The Relative Relation Metric can then be defined mathematically as:

$$\rho = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} H(-(x_i - x_j)(y_i - y_j))$$

H is the Heaviside step function, which takes the value 1 when the condition inside the parentheses is satisfied, and 0 otherwise [2]:

$$H(z) = \begin{cases} 0 & z < 0 \\ 1 & z \geqslant 0 \end{cases}$$

ρ is the number of all pairs (i, j), for which the difference between x_i and x_j has the opposite sign to the difference between y_i and y_j.

The above mathematical formula is represented by the Algorithm 1.

Algorithm 1 Relative Relation Metric Algorithm

Parameters:

- x $= [x_1, x_2, \ldots, x_n]$: A vector of length n
- y $= [y_1, y_2, \ldots, y_n]$: A vector of length n

procedure RELATIVE RELATION METRIC(x, y)
    $n \leftarrow$ length(x)
    $\rho \leftarrow 0$
    for $i \leftarrow 1$ **to** $n - 1$ **do**
        for $j \leftarrow i + 1$ **to** n **do**
            $a \leftarrow$ x$[i]$ − x$[j]$
            $b \leftarrow$ y$[i]$ − y$[j]$
            if sign$(a) \neq$ sign(b) **then**
                $\rho \leftarrow \rho + 1$
            end if
        end for
    end for
    return ρ
end procedure

The RRM algorithm has a quadratic complexity related to the number of genes analysed in the dataset and can be described by:

$$O(n^2)$$

where n is number of genes in the dataset. This problem is well known from RXA algorithms, where the complexity can even be multidimensional. It is worth mentioning that this problem has been addressed by parallelizing computations on GPGPU [5] and/or using evolutionary algorithms [32].

3.2. Implementation

The solution was implemented in Python 3.12 and made available as a open-source library ($pyrrm$) in the Gitlab public repository [25]. The library provides a `relative_relation_metric` function that takes two vectors of type $narrray$ as parameters. In order to parallelize and optimise the calculations the $numba$ library [23] was used. The $numba$ translates Python functions to optimised machine code at runtime using the industry-standard LLVM compiler library [21].

The $pyrrm$ library can be installed in a virtual Python environment and used as a metric in $KNeighborsClassifier$ from the $sklearn$ module [29]. An example of the use is shown in the library repository [25].

4. Results

In this section, we present a comprehensive experimental evaluation of the Relative Relation Metric (RRM) applied within the KNN algorithm denoted as KNN_{RRM}. We describe the datasets and algorithms used, compare various metrics, and provide a comparative study with popular solutions.

4.1. Experimental Setup

Experiments were conducted using gene expression-based datasets related to cancer, obtained from NCBI's Gene Expression Omnibus [8]. A 10-fold stratified cross-validation approach was employed, providing average accuracy and standard deviation from 10 iterations. Relief-F feature selection [27] was applied, and the number of genes was capped at 1000 for computational efficiency.

Table 1. Summary of gene expression datasets: abbreviation with name, number of genes, number of instances, class ratio and description.

	Datasets	Genes	Instances	Ratio (yes:no)	Description
(a)	GDS2771	22215	192	102:90	Lung cancer
(b)	GSE10072	22284	107	58:49	Adenocarcinoma
(c)	GSE17920	54676	130	92:38	Classic Hodgkin's Lymphoma
(d)	GSE19804	54613	120	60:60	NSCLC
(e)	GSE27272	24526	183	128:55	Tobacco effects on pregnancy
(f)	GSE3365	22284	127	85:42	PBMCs and Crohn's disease
(g)	GSE6613	22284	105	55:50	Parkinson disease

We compared KNN_{RRM} with widely-used KNN metrics such as Manhattan, Euclidean, Minkowski, and Chebyshev. The optimal number of neighbors for KNN_{RRM} was determined using additional datasets ($GSE25837$, $GSE4290$, and $GSE5772$), testing $N = 3, 5, 7, 9$. No significant differences were observed, hence $N = 3$ was selected for all KNN models in subsequent experiments.

Further, we assessed the overall performance of KNN_{RRM} against established machine learning classifiers:

- **k-Top Scoring Pairs (k-TSP)[33]:** Utilizes Relative Expression Analysis, with a default $k = 5$.
- **C4.5[36]:** A renowned decision tree classifier with univariate splits.
- **Random Forest (RF)[6]:** An ensemble-based method utilizing multiple decision trees.
- **Naive Bayes (NB)[12]:** A probabilistic classifier known for its simplicity.
- **Support Vector Machine (SVM)[7]:** Effective for linear and nonlinear datasets.

Algorithms were evaluated using the WEKA software [15] suite, with $C4.5$ refers to the decision tree implementation as $J48$, SVM refers to the implementation using the Sequential Minimal Optimization (SMO) algorithm, and kNN refers to the implementation as IBk. The KNN_{RRM} and $k - TSP$ were processed using the AUERA software [13].

4.2. Results for KNN

We evaluated the performance of the RRM against other popular KNN metrics: Chebyshev, Euclidean, Manhattan, and Minkowski. The evaluation criteria focused on the accuracy of the generated models. Figure 2 presents individual violin plots for each of the seven datasets, supported by box plots that highlight the median accuracy value and beeswarm plots to display the distribution of accuracy scores across runs (avoiding overlapping data points and enhancing visual clarity).

To provide a more detailed perspective, we further elaborate on the accuracy results for each dataset. The additional information covers statistical significance analysis using the Friedman

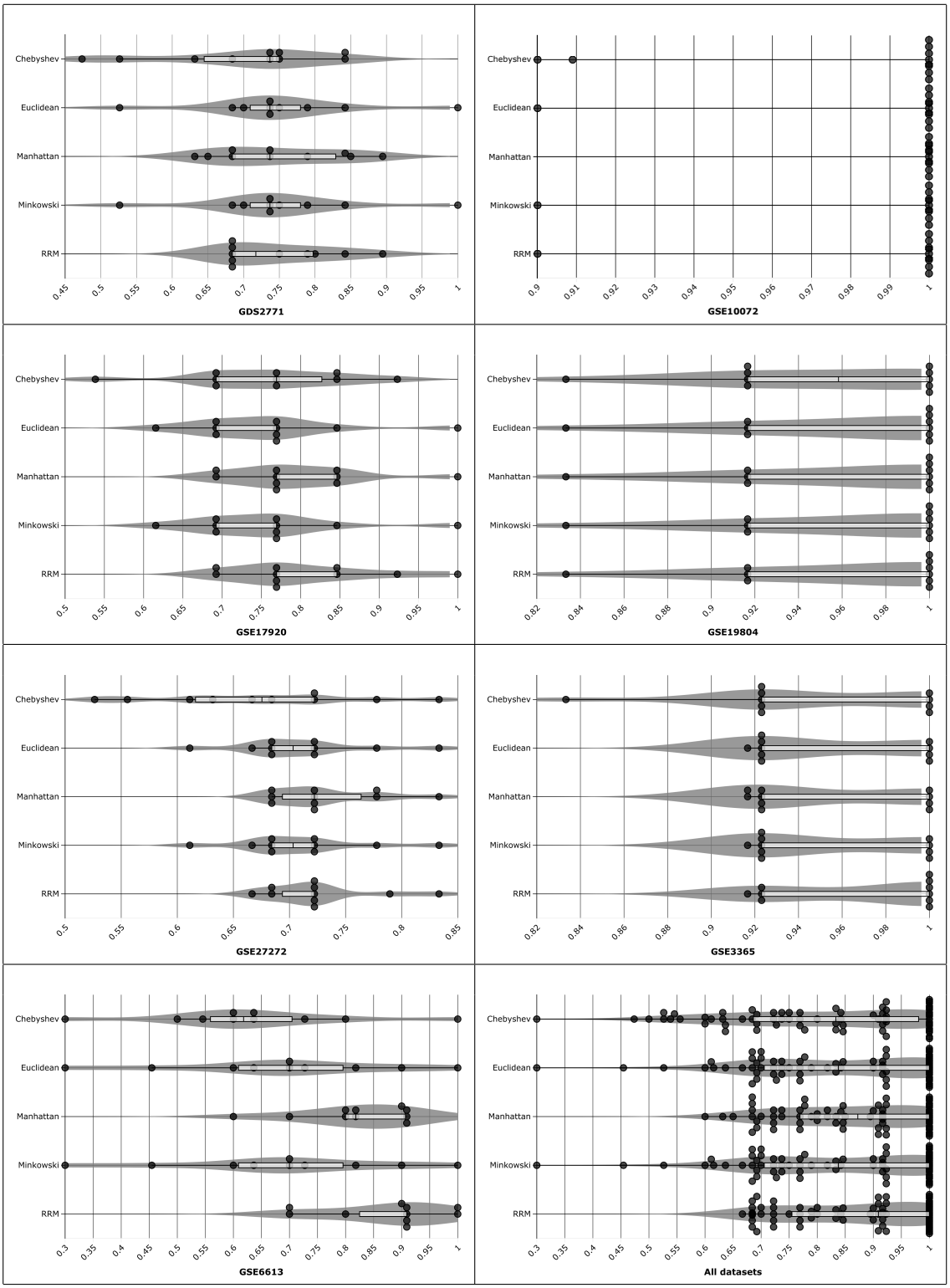

Fig. 2. Individual violin plots for each dataset with embedded box plots and beeswarm plots, representing the distribution of accuracy scores across seven datasets for each KNN metric: Chebyshev, Euclidean, Manhattan, Minkowski, and RRM. Each point corresponds to the accuracy of one run of the respective algorithm. The central line in each box plot denotes the median accuracy value, while the spread of points illustrates the variation in performance across runs.

test and the corresponding Dunn's multiple comparison test at a significance level of 0.05, as recommended by Demsar [11]. We report our comments on the accuracy scores achieved for each algorithm:

- **GDS2771:** No significant differences were noted as all metrics (except Chebyshev) achieved equal averaged accuracy; however, RRM was the only one without any occasional catastrophic failures.
- **GSE10072:** The dataset shows significantly different results compared to the other datasets due to its high separability. In most of the 10 runs, the ideal solution was found, with only one or two runs not achieving perfect classification. As a result, the violin plots could not show a meaningful distribution since the accuracy was consistently high across all runs. No significant differences were noted across all metrics; however, $Manhattan$ was the only metric that achieved a flawless score.
- **GSE17920:** Although the Friedman test found statistical differences between the metrics (P-value: 0.0407, F-statistics: 9.986), the corresponding Dunn's multiple comparison tests did not find any significant differences. RRM achieved the highest average accuracy and outperformed the rest of the metrics by 0.8 to 5.4 percentage points.
- **GSE19804, GSE27272, GSE3365:** No significant differences were noted across all metrics.
- **GSE6613:** The Friedman test found statistical differences between the metrics (P-value: 0.0007, F-statistics: 19.3). Corresponding Dunn's multiple comparison tests found that the proposed RRM metric significantly outperformed the $Chebyshev$ distance (P-value < 0.01). In addition, RRM was again the only metric without any occasional catastrophic failures.

Despite the absence of statistically significant differences among the distance metrics, it is imperative to acknowledge that the average performance of KNN_{RRM} across all datasets was 86.78 percentage points. This performance not only outstripped the Manhattan metric by a margin of 0.9 percentage points, but it also exceeded Euclidean and Minkowski by 3.8 percentage points, and Chebyshev by 6.3 percentage points. These observations are supported by the aggregated results depicted in the collective plot of Figure 2, which underscores the consistent achievement of KNN_{RRM} scores above 67 percentage points with no recorded instances of extreme failure. This contrast with other metrics, which did exhibit such failures, confirms the stability and resilience of the RRM against outlier values. The robust performance of RRM arises from its design principle, where even the occurrence of extreme values will not impact the ranking more than the induced changes in the feature ordering with respect to the test instance. Hence, the RRM offers a reliable and stable classification even in the face of high-dimensional data variability.

Moreover, a single run of the cross-validation algorithm using Euclidean, Minkowski, Chebyshev and Manhattan metrics is similar and takes approximately 0.019s. In the case of the RRM metric, the time is extended to 0.3s. Tests were made in Python3.12 using the $pandas$ and $sklearn$ modules and run on an Ubuntu 22.04 Linux environment with an Intel Core Processor 6 (30 cores) and 96GB of RAM.

4.3. KNN with RRM vs popular algorithms

The performance comparison of KNN with RRM against other well-known algorithms is presented in Table 2. This comparison sheds light on how KNN with RRM stands up to various challenges presented by gene expression data.

Table 2. Comparison of KNN_{RRM} classification performance to popular ML algorithms. Each algorithm's performance is detailed in terms of accuracy and standard deviation (acc.±SD).

Dataset	k-TSP	C4.5	RF	NB	SVM	KNN_{RRM}
	acc.±SD	acc.±SD	acc.±SD	acc.±SD	acc.±SD	acc.±SD
a)	62.90±3.3	66.30±10	76.05±10	71.01±11	79.93±9.2	74.97±7.4
b)	90.15±2.5	93.26±6.7	99.05±2.8	98.11±3.7	99.0±2.8	99.00±3.0
c)	67.26±3.2	74.61±11	80.30±8.2	81.23±12	94.07±6.3	80.77±9.3
d)	94.11±1.6	90.66±8.4	95.25±6.5	95.75±6.0	93.83±6.9	95.83±5.6
e)	58.40±4.0	59.80±10	70.20±3.5	70.53±10	83.71±7.3	72.69±4.7
f)	87.29±2.1	90.34±8.8	95.03±6.3	90.18±7.6	97.64±4.5	96.86±3.8
g)	55.81±5.3	57.40±15	76.66±12	79.17±11	88.15±8.3	87.36±5.4
AVG	79.70±3.1	76.05±9.9	84.64±7.0	83.71±8.7	90.90±6.4	86.78±6.3

It's clear from the results that while all algorithms have their strengths, KNN with RRM shows competitive performance, especially when compared to other traditional algorithms. The use of RRM within KNN provides an advantage in datasets that are prone to outliers and noise, as evidenced by the consistent accuracy across the different datasets. Notably, while SVM still leads in overall accuracy, KNN with RRM holds its ground as a close contender, offering a simpler yet effective model. While SVM can be something of a black box, making it difficult to extract decision rules that humans can easily understand, KNN with RRM allows for more transparency. This aspect can be crucial when the decisions made by the model need to be justified or explained in a clear manner, such as in clinical settings.

5. Conclusion and Future Works

In summary, this paper has introduced the Relative Relation Metric (RRM) as an innovative concept in the domain of distance metrics for k-nearest neighbors (KNN) algorithm. We argue that while RRM may not fit all data types, it shows promise particularly with high-dimensional omics data—such as genomic, where variables often span similar ranges. One of RRM's strengths is its resistance to outliers and its insensitivity to methodological variations and transformations that are common in biological data. By focusing on internal feature relations of an instance rather than comparing between instances, RRM ensures that the relative order within an individual is paramount, preserving intrinsic biological relationships. While the results don't show a dramatic difference in accuracy between tested metrics, the reliability and transparency of KNN with RRM could make it a preferred choice.

For future works, we acknowledge that the results presented are preliminary. While we employed the simplest form of KNN to illustrate the use of RRM, we understand that more advanced algorithms may better suit the specificity of biomedical data. Here, we focus on introducing a new distance metric rather than a comprehensive solution. We also see significant potential in knowledge extraction from KNN coupled with RRM. The capacity for post-KNN interpretation by validating which features most effectively discriminate clusters within the context of ordering; presents an exciting avenue for identifying markers, either as single features or longer relational sequences. These types of relations, albeit in a simpler form, are already used by physicians and clinicians in algorithms generated by RXA. Beyond the biological interpretation of KNN with RRM, we are also exploring the application of this metric to multi-omic data integration, which could simplify the integration process through the use of straightforward ordinal relations.

Lastly, we are considering the introduction of more advanced relations including weight or hierachical [10], to recapture some of the information lost when abstracting to relational met-

rics, akin to more advanced RXA algorithms. Such enhancements would enrich the RRM framework and potentially increase its applicability and accuracy in omics data analysis. Furthermore, we are considering applying computational parallelization on the GPGPU and/or the use of evolutionary algorithms to improve the performance of our approach in our future work.

Acknowledgement

This project is supported by the grant WZ/WI-IIT/3/2023 from BUT founded by Polish Ministry of Science and Higher Education (first author) and by the Polish National Science Centre and allocated on the basis of decision 2019/33/B/ST6/02386 (second author).

References

1. Ayyad, S.M., Saleh, A.I., Labib, L.M.: Gene expression cancer classification using modified K-Nearest Neighbors technique, Biosystems 176: 41–51 (2019), ISSN 0303-2647, doi: 10.1016/j.biosystems.2018.12.009.
2. Abramowitz, M., Stegun, I.A.: Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, Dover Publications INC (1972).
3. Ahmed, O., Brifcani, A.: Gene Expression Classification Based on Deep Learning, 2019 4th Scientific International Conference Najaf (SICN), Al-Najef, Iraq, 2019, pp. 145-149, doi: 10.1109/SICN47020.2019.9019357.
4. Alharbi, F.; Vakanski, A.Machine Learning Methods for Cancer Classification Using Gene Expression Data: A Review. Bioengineering 2023, 10, 173, doi: 10.3390/bioengineering10020173
5. Bhat, N.G., Balaji, S.: Modelling and simulation of lac-operon gene expression using heterogeneous parallel platforms. Int. j. inf. tecnol. 15, 2293–2302 (2023). doi: 10.1007/s41870-023-01256-0
6. Breiman, L.: Random Forests. Machine Learning, 45:5–32 (2001).
7. Burges, C.J.C.: A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery, 2(2):121–167, (1998).
8. Clough, E., Barrett, T.: The Gene Expression Omnibus database. In: Methods in Molecular Biology (2016).
9. Czajkowski, M., Czajkowska, A., Kretowski, M.: TIGER: an evolutionary search for Top Inter-GEne Relations. Int. J. Data Min. Bioinformatics 16(2):170–182 (2016).
10. Czajkowski, M., Jurczuk, K., Kretowski, M.: Generic Relative Relations in Hierarchical Gene Expression Data Classification. In Parallel Problem Solving from Nature: PPSN XVI, Springer-Verlag 372–384 (2020).
11. Demšar, J.: Statistical Comparisons of Classifiers over Multiple Data Sets. J. Mach. Learn. Res., 7:1–30 (2006).
12. Domingos, P., Pazzani, M.: The Optimality of Naive Bayes. In: Proceedings of the Thirteenth International Conference on Machine Learning, 118–126, (1996).
13. Earls, J.C., Eddy, J.A., et al.: AUREA: An open-source software system for accurate and user-friendly identification of relative expression molecular signatures. BMC Bioinformatics, 14:78 (2013).
14. Eddy, J.A., Sung, J., Geman, D., Price, N.D.: Relative expression analysis for molecular cancer diagnosis and prognosis. Technol Cancer Res Treat, 9(2), 149-159 (2010).
15. Frank, E., Hall, M.A., Witten, I.H.: The WEKA Workbench. Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann (2016).
16. Geman, D., d'Avignon, C., Naiman, D.Q., Winslow, R.L.: Classifying gene expression profiles from pairwise mRNA comparisons. Statistical Applications in Genetics and Molecular Biology, 3(19) (2004).

17. Godlewski, A., Czajkowski, M., Mojsak, P. et al. A comparison of different machine-learning techniques for the selection of a panel of metabolites allowing early detection of brain tumors. Sci Rep 13:11044 (2023).
18. Hechenbichler, K., Schliep, K.: Weighted k-Nearest-Neighbor Techniques and Ordinal Classification. Discussion Papers in Statistics and Econometrics, 2:04 (2004).
19. Li, W., Cerise, J.E., Yang, Y., Han, H.: Application of t-SNE to human genetic data. J Bioinform Comput Biol, 15(4):1750017 (2017).
20. Lin, M.C., Iqbal, U., Li, Y.C.: AI in Medicine: Big Data Remains a Challenge. Computer Methods and Programs in Biomedicine 164 (2018).
21. LLVM official website. https://llvm.org.
22. Mirza, B., Wang, W., et al. Machine Learning and Integrative Analysis of Biomedical Big Data. Genes (Basel) 10(2):87 (2019).
23. Numba: High-Performance Python Compiler. https://numba.pydata.org/.
24. Parry, R., Jones, W., Stokes, T. et al.: k-Nearest neighbor models for microarray gene expression analysis and clinical outcome prediction. Pharmacogenomics J 10:292–309 (2010).
25. PyRRM project on GitLab. https://gitlab.com/izabeera/pyrrm.
26. Rauschert, S., Raubenheimer, K., et al. Machine learning and clinical epigenetics: a review of challenges for diagnosis and classification. Clin Epigenet 12:51 (2020).
27. Robnik-Sikonja, M., Kononenko, I.: Theoretical and empirical analysis of ReliefF and RReliefF. Machine Learning, 53(1–2), 23–69 (2003).
28. Rukhsar, L.; Bangyal, W.H.; Ali Khan, M.S.; Ag Ibrahim, A.A.; Nisar, K.; Rawat, D.B. Analyzing RNA-Seq Gene Expression Data Using Deep Learning Approaches for Cancer Classification. Appl. Sci. 2022, 12, 1850. https://doi.org/10.3390/app12041850.
29. Scikit-learn k-Nearest Neighbors classifier documentation. https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html.
30. Schrauf, M.F., de los Campos, G., Munilla, S.: Comparing Genomic Prediction Models by Means of Cross Validation. Front. Plant Sci., 12:734512 (2021).
31. Shuangge, M., Ying, D.: Principal component analysis based methods in bioinformatics studies. Briefings in Bioinformatics, 12(6), 714–722 (2011).
32. Slowik, A., Kwasnicka, H.: Evolutionary algorithms and their applications to engineering problems Neural Comput & Applic 32, 12363—12379 (2020), https://doi.org/10.1007/s00521-020-04832-8
33. Tan, A.C., et al.: Simple decision rules for classifying human cancers from gene expression profiles. Bioinformatics, 21, 3896–3904 (2005).
34. Taunk K., De S., Verma S. and Swetapadma A.: A Brief Review of Nearest Neighbor Algorithm for Learning and Classification. International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 1255–1260 (2019).
35. Tjärnberg, A., Mahmood, O., et al.: Optimal tuning of weighted kNN- and diffusion-based methods for denoising single cell genomics data. PLOS Computational Biology 17(1):e1008569 (2021).
36. Quinlan, R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo, CA, USA (1993).
37. Zhang, H., Liu, C.T., Wang, X.: An Association Test for Multiple Traits Based on the Generalized Kendall's Tau. J Am Stat Assoc. 105(490) 473–481 (2010).