Various Distance Metrics Evaluation on Neural Spike Classification

Shengpeng Guo, Liyuan Guo, Seyed Mohammad Ali Zeinolabedin, and Christian Mayr
Chair of Highly-Parallel VLSI-Systems and Neuro-Microelectronics Circuits
Technische Universität Dresden, Dresden, Germany
Email:shengpeng.guo@mailbox.tu-dresden.de
{liyuan.guo, ali.zeinolabedin, christian.mayr}@tu-dresden.de

***Abstract*—State-of-the-art neuroscience applications require the classification of action potential activities (or spikes) recorded by multi-channel electrodes, named spike sorting. A typical spike sorting algorithm involves three parts: detection, feature extraction, and training/classification. Training/classification stages are mainly based on distance-based algorithms like K-means. For the classification in particular, the accuracy depends highly on the distance metric performance. In this paper, we analyze several distance metrics to classify various datasets with different noise levels using the F1-score. This includes Manhattan, Euclidean and, Mahalanobis distance metrics. As a result, the Mahalanobis distance metric outperforms other metrics in noisy conditions by an average of 10% in the presence of the noise.**

***Index Terms*—Biomedical signal processing, spike sorting, distance measurement, Mahalanobis distance, Euclidean distance, classification.**

I. Introduction

The extracellular signals generated by the neural activity are the basis of neural prostethics applications and neuroscience research [1]–[3]. The raw data is recorded by the analog-front-end (AFE) and it contains actions potential activity, known as spikes, and the background noise. Spikes are basically generated by the active neurons near-by the electrodes. In many neural studies and applications, individual units of activity, i.e., spikes are required as data inputs [1]. Therefore, precise classification is essential to neuronal signal processing. The process of mapping the recorded action potential (or spikes) to the neurons from which they are generated is referred to as ”spike sorting”. This process usually has three main steps: detection and alignment, feature extraction, and training/classification [2]–[6] as shown in Fig. 1. First spikes are separated from the background noise using the detection methods. Then they are aligned to a certain point and after that, some informative features are computed. Having the features calculated, training is done to obtain the required information for the classification like the number of clusters and the clusters mean. Training is the most computationally complex part and is only running intermittently [7] to (re)train the system. In various implementation, the training stage are often performed off-chip [2], [4], [5], [7], [8]. The trained parameters provided by the training are stored separately and used for real-time classification.

During a real-time classification, each spike is assigned to a cluster identified in the training phase according to the distance calculation. The minimum distance calculation has an impact on the overall accuracy of classification. In most previous implementations, the l_1-distance (i.e., Manhattan distance) is the most commonly used metrics, and it has acceptable, sometimes even better performance compared to the l_2-distance (i.e., Euclidean distance) metric [2], [4]. However, the Euclidean distance metric does not take into consideration the distribution of the clusters and the correlation among different clusters [9]. The Euclidean distance between a point and the cluster means can give little or misleading information about how close a point really is to the cluster. The Mahalanobis distance metrics is a multivariate equivalent of the Euclidean distance which can solve the problems mentioned above [10]. In this paper, we evaluate the performance of different distance metrics using datasets with various levels of noise. Based on this analysis, different distance metrics can be wisely chosen to fit the different scenarios. Furthermore, the performance of the distance metrics with respect to the noise introduced by the predecessor stages is also evaluated.

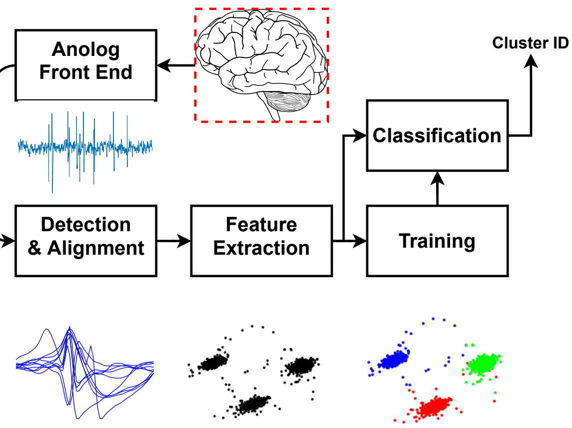
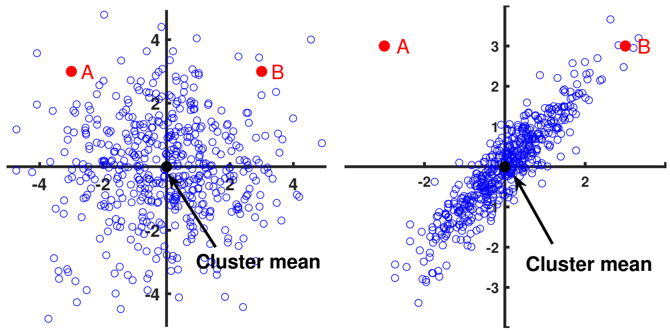


Fig. 1. Spike sorting flow [2].

II. Various Distance Metrics for Spike Classification

After detection, all detected spikes are transferred into a feature space that facilitates representing them by a smaller multidimensional vector. The more informative the features are, the better the training/classification will be. In terms of



(a) (b)

Fig. 2. Two distribution with (a) uncorrelated and (b) correlated dimensions. Point A and B are at the same distance from the origin. The Euclidean distance works in (a) but could not assign A as an outlier in (b).

training, K-means and its variants are the most popularly used algorithms in both hardware implementation and software simulation [2], [5], [7]. For classification, various distance metrics are introduced and the overall accuracy depends on their performance. In the following, three main distance metrics are introduced.

A. Euclidean and Manhattan distance metrics

Euclidean distance metric, as well as Manhattan distance metric, are the widely used distance metrics because of their good clustering accuracy and simple circuit implementation. The Euclidean distance (d_{EUC}) and Manhattan distance (d_{MAN}) are calculated as:

$$d_{EUC} = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad (1)$$

$$d_{MAN} = \sum_{i=1}^{n} |x_i - y_i| \,. \quad (2)$$

The Euclidean distance works appropriately as long as the dimensions are equally weighted and independent of each other [9]. However, in the neural datasets, the dimensions are typically correlated to one another. For instance, as shown in Fig. 2, points A and B are at the same distance from the origin. When the two dimensions are uncorrelated, the Euclidean distance from the cluster mean can be useful to determine if a point is a member of this cluster. On the other hand, if the cluster is essentially obeying a linear distribution with $f(x) = x$, point B is more likely to be a point in this cluster, while A is more likely to be an outlier. Fig. 2 illustrates this situation with Fig. 2(a) uncorrelated distribution and Fig. 2(b) correlated distribution. The Euclidean distance is simply the distance between two points. It does not take into account how the other points in the data set are changing. Therefore, it cannot be used to really determine how likely a new point belongs to a cluster distribution.

B. Mahalanobis distance metric

Mahalanobis distance rotates the variables by principal components so that the dimensions are independent of each other and then normalize to make the dimensions equally distributed. It is a multivariate equivalent of the Euclidean distance since it is calculated using the inverse of the covariance matrix of the dataset [10]. The calculation of Mahalanobis distance (d_{MAH}) is as follows:

$$d_{MAH} = \sqrt{(x_i - \bar{x})\boldsymbol{C}_x^{-1}(x_i - \bar{x})^T}, \quad (3)$$

where x_i is a sample of the observation, and $\bar{x}$ is the cluster mean provided by the training. The $\boldsymbol{C}_x^{-1}$ term is the inverse covariance matrix of the distributions provided by the training.

In Mahalanobis, the distance between the new sample and the distribution is divided by the covariance matrix. If the dimensions in the dataset are strongly correlated, then the covariance will be high, and dividing by a large covariance will reduce the distance value. On the contrary, if the dimensions are not correlated, the distance is not reduced accordingly. Thus, the Mahalanobis distance accomplishes the rotation and scaling of the distribution. So it effectively addresses both the scaling problems as well as the correlation of the clusters distribution.

III. Evaluation Method

In this section, all introduced distance metrics are compared in MATLAB using the datasets described in Section III-B. Fig. 3 gives the process of classification evaluation using F1-score described in Section III-C.

A. Simulation Setup

- Spike detection: spikes need to be extracted from raw data using the true spike times provided by the dataset. This ensures that the rest process is not affected by the possible inaccuracy of the spike detection method.
- Feature extraction: the detected spikes are first divided into two parts: train spikes and validation spikes. For both parts, a feature extraction method is required to reduce the dimensionality of spikes. Principal Component Analysis (PCA) has become a benchmark method in neural signal processing because of its accuracy [11]. The dimensions of extracted features can vary from 2 to 10.
- Training stage: the training is performed in ideal condition using the K-means algorithm and the train features. The K-mean algorithm uses the Euclidean distance to find the cluster mean. Unlike the conventional K-means, besides the cluster mean, the covariance matrix (cov matrix) and a threshold value of each predicted cluster (Thr matrix) are also computed and sent to the classification stage. The covariance matrix is required by Mahalanobis distance, and the threshold value of each cluster is utilized to separate the outliers.
- Classification stage and evaluation: for each spike in validation spikes, the distance to each cluster is calculated with Euclidean, Manhattan, and Mahalanobis distance metrics. The minimum distance value indicates that the

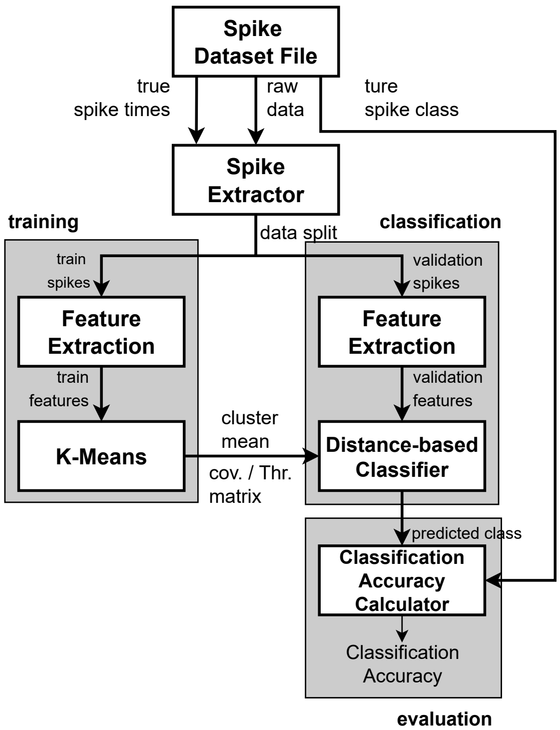


Fig. 3. Flowchart of various distance metrics evaluation.

current sample is a member of the cluster, and the predicted labels will be compared with the ground truth to calculate the F1-score.

B. Datasets

The distance metrics are evaluated with datasets introduced in [12], in which four different dataset groups are included, with varying degrees of difficulty in spike detection and identification. Each dataset contains simulated spikes generated by three neurons with an approximate firing rate of 20 Hz. In addition, different levels of background noise are superimposed and placed randomly to the waveform. The shape of noise is chosen randomly from the spike library to emulate the background noise generated by distant neuronal activity in the actual recordings. Noise levels are determined based on their standard deviation relative to the amplitude of the integrated spike waveform, which is equal to 0.05, 0.1, 0.15, and 0.2, respectively. For datasets group EASY 1, four additional noise levels are applied with values of 0.25, 0.3, 0.35, and 0.4. Thus, a total of 22 datasets is used in the simulation, and in each dataset, approximately 3000 spikes are available [12]. In each dataset file, along with the raw data, the ground truth information such as true spike time and the class are also given, which can be used to calculate accuracy (Fig. 3).

C. Accuracy Calculation

Performance of distance metrics is evaluated after classification using F1-score. F1-score is the harmonic mean between precision and recall, which gives not only how precise the classifier is, but also how robust it is. F1-score can be calculated using the confusion matrix described in [13]. The true positive rate (Recall) is given by

$$Recall = \frac{TruePositives}{TruePositives + FalseNegative} \tag{4}$$

and the positive predictive value (Precision) is defined as

$$Precision = \frac{TruePositives}{TruePositives + FalsePositive} \tag{5}$$

F1-score tries to find the balance between precision and recall. Mathematically, it can be expressed as:

$$F_1 = \frac{2}{Recall^{-1} + Precision^{-1}}. \tag{6}$$

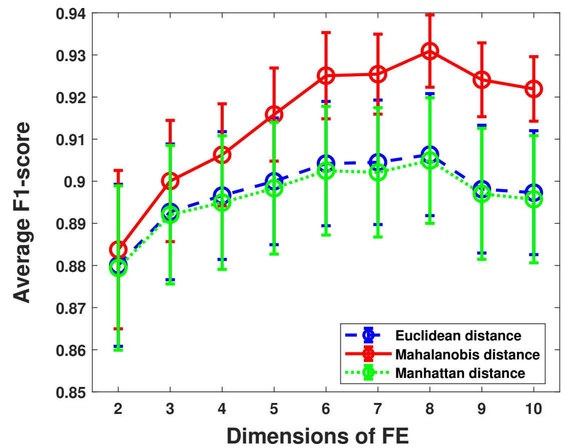


Fig. 4. Comparison of classification accuracy for all three distance metrics. The data is based on simulation with 22 datasets and calculate the average F1-score. The dimension of extracted features varies from 2 to 10. The standard deviation is also presented in this Figure.

IV. Evaluation Results

In this section, we present the classification result of various distance metrics for neural signal classification.

A. Overall Comparison

Fig. 4 evaluates all mentioned distance metrics applied to all 22 datasets in [12]. In this simulation, the spike detection is not first utilized to avoid including the false positive samples and to analyze distance metrics in ideal case. The average F1-score of Mahalanobis is at 91.48%, with an average standard deviation of 0.77% for all datasets. Euclidean distance achieves 89.78% with an average standard deviation of 1.45%, and Manhattan distance achieves 89.63% with 1.49%.

In general, the Manhattan and the Euclidean distance metrics have the same performance with increasing the number of extracted features. Despite the fact that there is not much difference between the Mahalanobis and the Euclidean at the beginning, as the number of features increases, the Mahalanobis is gradually outperforming the Euclidean in both average F1-score and standard deviation by 2.6% and 0.69%

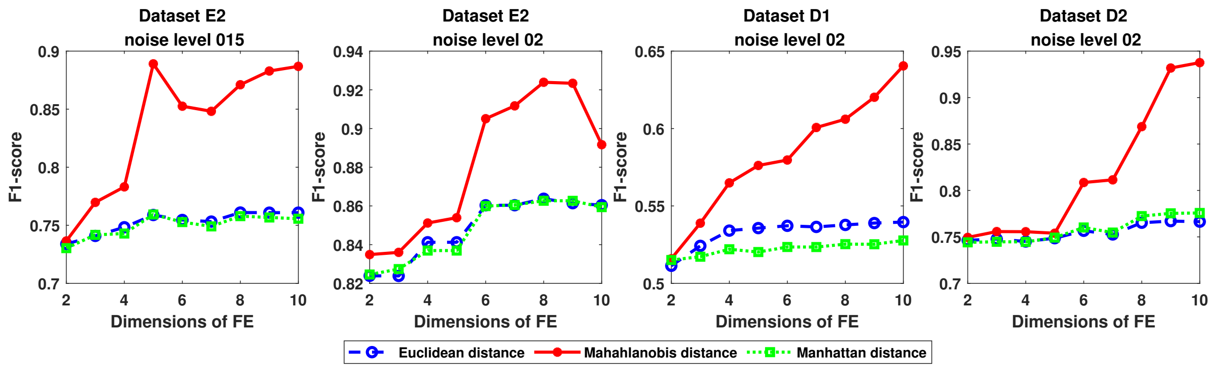


Fig. 5. Evaluation results of various distance metrics for the noisy datasets. The Mahalanobis distance metric improves the F1-score by 10% compared to the Euclidean distance.

with 9 extracted features. Because for most datasets, both distance metrics achieve high accuracy, which is above 90%, thus, the difference may seem not considerable. The difference emerges in datasets with higher noise level, which is explained in Section IV-B.

B. Evaluation for the noisy datasets

Fig. 5 illustrates the performance of the distance metrics on noisy and difficult conditions. In Fig. 5, we choose the most noisy datasets. It shows the performance of three distance metrics in terms of F1-score in regard to the ground truth information. In most noisy cases, except for datasets EASY1, in which both distance metrics achieve a high F1-score, the Mahalanobis distance metric outperforms the Euclidean and the Manhattan distance metric by more than 10%. The performance of the Manhattan and Euclidean distance remains similar.

C. Spike detection impact on the classification performance

The spike detection has an impact on the overall classification performance. In the spike detection stage, any signals with an amplitude above a certain threshold value will be labeled as a spike signal. Therefore, there are two sources of error [14]. The first is called the false positive, which is caused by high background noise exceeding the threshold value. The second is the false negative, which is caused by the improper selection of the threshold value. The simulation results, as illustrated in Fig. 6 show that the Mahalanobis distance metric still outperforms other distance metrics for the noisy datasets. Furthermore, the F1-score slightly drops for all distance metrics because the threshold value is constant over time.

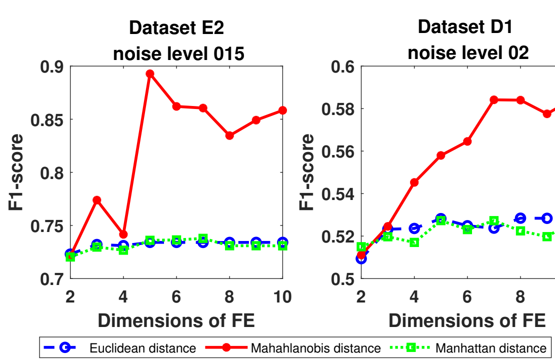


Fig. 6. Performance evaluation in the presence of the spike detection.

TABLE I
HARDWARE EFFORTS OF DISTANCE MATRICES PER CLASSIFICATION[*]

	clock cycles	memory (kB)	add/sub	mul/div
Mahalanobis	1976	∼7.2	800	880
Euclidean	256	∼0.8	152	80
Manhattan	168	∼0.8	152	-

[*] The analysis is based on the hardware efforts per classification in the worst case.

D. Hardware efforts

The experiments on various datasets showcase that in the worst case, 10 features (refer to Figs. 4 and 5) per cluster are enough to perform the classification achieving good accuracy. Table I shows the hardware implementation requirements of three methods and Mahalanobis is the most computationally intensive one.

As a result, we propose an independently configurable multi-channel neural classifier working often based on the Euclidean or Manhattan distance matrices. It can be switched to the Mahalanobis depending on the noising level of datasets.

V. Conclusion

In this paper, we present a complete analysis of various distance metrics, i.e., Manhattan, Euclidean, and Mahalanobis methods for the classification stage of the spike sorting algorithm. The Mahalanobis distance outperforms the other ones by an average of 10% in the noisy datasets. For future work, we will analyze the hardware complexity of mentioned distance metrics in terms of the computational complexity, area, and power consumption for the real-time applications.

References

[1] S. Gibson, J. W. Judy, and D. Marković, "Spike sorting: The first step in decoding the brain: The first step in decoding the brain," *IEEE Signal Processing Magazine*, vol. 29, no. 1, pp. 124–143, 2012.

[2] A. T. Do, S. M. A. Zeinolabedin, D. Jeon, D. Sylvester, and T. T.-H. Kim, "An area-efficient 128-channel spike sorting processor for real-time neural recording with 0.175 μ w/channel in 65-nm cmos," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 1, pp. 126–137, 2019.

[3] S. M. A. Zeinolabedin, A. T. Do, K. S. Yeo, and T. T.-H. Kim, "Design of a hybrid neural spike detection algorithm for implantable integrated brain circuits," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2015, pp. 794–797.

[4] V. Karkare, S. Gibson, and D. Marković, "A 75-µw, 16-channel neural spike-sorting processor with unsupervised clustering," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 9, pp. 2230–2238, 2013.

[5] P. C. Petrantonakis and P. Poirazi, "A novel and simple spike sorting implementation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 4, pp. 323–333, 2017.

[6] S. M. A. Zeinolabedin, F. M. Schüffny, R. George, F. Kelber, H. Bauer, S. Scholze, S. Hänzsche, M. Stolba, A. Dixius, G. Ellguth, D. Walter, S. Höppner, and C. Mayr, "A 16-channel fully configurable neural SoC with 1.52 w/ch signal acquisition, 2.79 w/ch real-time spike classifier, and 1.79 TOPS/w deep neural network accelerator in 22 nm FDSOI," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 16, no. 1, pp. 94–107, 2022.

[7] M. Saeed, A. A. Khan, and A. M. Kamboh, "Comparison of classifier architectures for online neural spike sorting," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 4, pp. 334–344, 2017.

[8] M. Zamani, J. Sokolić, D. Jiang, F. Renna, M. R. D. Rodrigues, and A. Demosthenous, "Accurate, very low computational complexity spike sorting using unsupervised matched subspace learning," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 14, no. 2, pp. 221–231, 2020.

[9] M. Kim, "Multivariate outliers and decompositions of mahalanobis distance," *Communications in Statistics-theory and Methods - COMMUN STATIST-THEOR METHOD*, vol. 29, pp. 1511–1526, 07 2000.

[10] R. De Maesschalck, D. Jouan-Rimbaud, and D. Massart, "The mahalanobis distance," *Chemometrics and Intelligent Laboratory Systems*, vol. 50, no. 1, pp. 1–18, 2000.

[11] A. M. Kamboh and A. J. Mason, "Computationally efficient neural feature extraction for spike sorting in implantable high-density recording systems," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 21, no. 1, pp. 1–9, 2013.

[12] R. Quian, Z. Nadasdy, and Y. Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering," *Neural computation*, vol. 16, pp. 1661–87, 2004.

[13] D. Powers, "Evaluation: From precision, recall and f-factor to roc, informedness, markedness correlation," *International Journal of Machine Learning Technology*, vol. 2, 2008.

[14] S. Gibson, J. W. Judy, and D. Markovic, "Comparison of spike-sorting algorithms for future hardware implementation," pp. 5015–5020, 2008.