# Weighted Principal Component Analysis based on statistical properties of features for Spike Sorting

Roxana Ioana Aldea
*Computer Science*
*Technical University of Cluj-Napoca*
Cluj-Napoca, Romania
aldearoxanaioana@gmail.com

Mihaela Dinsoreanu
*Computer Science*
*Technical University of Cluj-Napoca*
Cluj-Napoca, Romania
Mihaela.Dinsoreanu@cs.utcluj.ro

Rodica Potolea
*Computer Science*
*Technical University of Cluj-Napoca*
Cluj-Napoca, Romania
Rodica.Potolea@cs.utcluj.ro

Camelia Lemnaru
*Computer Science*
*Technical University of Cluj-Napoca*
Cluj-Napoca, Romania
Camelia.Lemnaru@cs.utcluj.ro

Raul Cristian Muresan
*Experimental and Theoretical Neuroscience Lab*
*Transylvanian Institute of Neuroscience*
Cluj-Napoca, Romania
muresan@tins.ro

Vasile Vlad Moca
*Experimental and Theoretical Neuroscience Lab*
*Transylvanian Institute of Neuroscience*
Cluj-Napoca, Romania
moca@tins.ro

***Abstract*—Spike Sorting is a challenging problem in Computational Neuroscience because of the complexity of neural data. One of the greatest issues are overlapping clusters. This paper focuses on the feature extraction step in the Spike Sorting pipeline and proposes an adaptation of Principal Component Analysis (PCA) to increase the separability between clusters. This is achieved by weighting the features before applying PCA, taking into consideration the multimodality and the distance between probability distributions. The information extracted from the characteristics of a multimodal distribution is the number of modes (peaks). The distance between the probability distributions is quantified using Jensen-Shannon divergence. The computed information, number of modes and distance, is aggregated into a coefficient representing the weight of the features. The new approach has been validated on a synthetic dataset and shows improvements compared with the state-of-the-art PCA.**

***Index Terms*—feature extraction, overlapping clusters, multimodal distributions, Jensen-Shannon divergence, Machine Learning, Spike Sorting**

## I. Introduction

An action potential or **spike** is an electrical signal discharged by a neuron when an external impulse occurs. Spike Sorting refers to grouping action potentials into clusters based on the similarity of their characteristics, aiming to match each cluster to a firing neuron. Each neuron has a particular firing pattern and produces spikes similar in shape. The shape and width of a spike is affected by the configuration of its dendritic tree, by the activation-deactivation kinetics of the voltage-dependent channels of sodium, potassium and calcium and by the influx-outflux and movements of the ions through the neural cell [1].

Usually, data is collected extracellularly by implanting tiny electrodes into the brain tissue and recording the activity of a population of neurons from a certain part of the brain and at a specific time.

Therefore, an accurate labeling of spikes would be essential for neuroscientists to study the activity of individual populations of neurons and better understand specific brain processes.

One of the greatest challenges of neural data is the problem of overlapping clusters, which arises because of spikes similar in shape but produced by different neurons or because of recording issues such as electrode drift.

The performance of the clustering is highly dependent on the quality of the data. Due to overlapping clusters, the clustering algorithm has difficulties in assigning a spike from an overlapping region to a cluster. Therefore, our aim is to overcome this challenge by **increasing** the **separability between clusters** in the feature extraction step.

The rest of the paper is organised as follows: section II presents related work and state-of-the-art methods of feature extraction. In section III is described the proposed solution. Section IV presents the dataset used and covers the validation of the feature extraction approach. The conclusions, limitations of the presented method and future improvements that can be made are discussed is section V.

## II. Related Work

Feature extraction is defined as modifying the geometry of the feature space by creating new discriminant features based on a combination of the original ones.

### A. Principal Component Analysis

Principal Component Analysis or PCA [2] is an unsupervised, linear method of feature extraction which finds an appropriate change of vector basis by performing a linear combination, a weighted sum of the original features, with the aim to maximize the variance and minimize the correlation between the principal components. The principal components

978-1-6654-0976-6/21/$31.00 

represent the reprojected original data points an a new orthonormal vector basis. The vectors which form the new basis are the **eigenvectors** of the covariance matrix and they are ordered by their magnitudes, referred to as **eigenvalues**. As mentioned, PCA takes into consideration the **variance** of features and sometimes variance is not a strong enough criterion to impose a separation between clusters.

### *B. t-distributed Stochastic Neighbor Embedding*

t-distributed stochastic neighbor embedding or shortly, t-SNE [3] is a nonlinear method of dimensionality reduction. The algorithm performs a mapping from a high-dimensional space to a low-dimensional space that retains most of the relevant information. The idea of this algorithm is to convert the high-dimensional Euclidean distances between data points into conditional probabilities that correspond to their similarities. Therefore, t-SNE constructs a probability distribution for the high-dimensional samples and tries to approximate it with another similar probability distribution of the points in the low-dimensional space. This is achieved by minimizing the sum of Kullback-Leibler divergences over all data points using a gradient descent method.

Because it is used mostly as a visualization method and does not create new data that can be used by the clustering algorithm it could not be used as a feature extraction method in this work.

## III. A NEW APPROACH OF FEATURE EXTRACTION

The proposed solution offers a larger **bias**, a **weight** in the linear combination performed by PCA to those features that are prone to bring more valuable information in order to **increase the separability between clusters**.

Regarding the data format, each spike waveform is described by a number of values and can also be viewed as a point in a D dimensional space, where D represents the number of values. The feature space was reduced to 2 due to the fact that the first 2 principal components retain the most variance. As aforementioned, PCA does not overcome the overlapping clusters problem by considering the variance, therefore, other characteristics of features have to be considered.

### *A. Multimodal distributions*

Exploratory analysis of features' distributions is essential in order to understand the nature of data and draw insights regarding its structure. The probability distributions of features can be unimodal or multimodal, the latter one presenting more modes which appear as local maxima or peaks in the probability density function.

The clusters follow a Gaussian distribution, therefore, each mode of a multimodal distribution can represent the centroid of a cluster. The multimodal distributions can be seen as a mixture of Gaussian distributions where each Gaussian represents a cluster. A more accurate definition of the Gaussians in the mixture results in a better separation between clusters.

The main characteristic of a multimodal distribution is the number of peaks which was computed using the histogram. The most important parameter of the histogram is the number of bins. Finding the optimal number of bins is a crucial step because different numbers of bins can lead to very different results that have divergent interpretations about the nature of the data. A reduced number of bins, implies increased wideness (a bimodal distribution may appear as unimodal), while a higher number of bins may introduce a lot of noise and also, hide essential information about the underlying data. Thus, Freedman-Diaconis' method was chosen to compute the optimal bin size [4].

The number of peaks was computed by performing a comparison with the neighboring values. When identifying the peaks, some parameters have to be considered such as distance, width or prominence. By far the most important parameter is the **prominence** which represents how much a peak protrudes from the signal. The approach of computing the prominence of a point $p1$ consists in tracing first a horizontal line through $p1$ until the border of the window or another point located on the curve is intersected. Denote the intersection from the left with $p2$ and from the right with $p3$. Consider the point $m$ to be the minimum value on the $y$ axis of the interval defined by $p2$ and $p3$. The difference between the $y$ coordinate of $p1$ and $m$ represents the prominence. An illustration of prominence can be observed in Fig. 3 from Appendix. The value chosen for this parameter was selected empirically to be **50** by performing multiple experiments.

The importance of a feature is directly proportional with the number of peaks. Therefore, it is quantified with a weight based on this criteria. The weighting coefficient is computed relative to the whole set of features. The weighing formula for one feature was designed to be the division between the number of peaks of that feature and the maximum number of peaks presented by a feature from the dataset.

$$w_p(i) = \frac{peaks_i}{\max\limits_{1 \le j \le N}(peaks_j)} \tag{1}$$

### *B. Distance between probability distributions*

Regarding the relationship between two probability distributions, a well-known concept from information theory is Kullback-Leibler divergence [5]. It represents the statistical distance between two statistical objects quantifying how much they differ. When considering two probability distributions, it quantifies how much a distribution diverges from another reference distribution. KL divergence is also referred as *relative entropy*. KL divergence between two probability distributions $P$ and $Q$ is defined by (2).

The entropy [6] represents the uncertainty regarding a variable's possible outcomes. The entropy of a random variable $X$ is defined by (3). The intuition behind the logarithm function is that the surprise increases as a less probable event happens. Therefore, the logarithm of a probability equal to 1 is 0, because there is no uncertainty.

*Cross Entropy = Entropy + Kullback-Leibler divergence*

$$D_{KL}(P||Q) = \sum_{i=1}^{n} P(x_i) \log_2 \frac{P(x_i)}{Q(x_i)} \quad (2)$$

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log_2 P(x_i) \quad (3)$$

A higher score implies a larger distance between the probability distributions.

Jensen-Shannon divergence [7] is the normalized and symmetrical version of KL divergence. Therefore, JS divergence was chosen over KL divergence.

The formula below (4) defines the JS divergence between two probability distributions $P$ and $Q$:

$$D_{JS}(P||Q) = \frac{1}{2} D_{KL}(P||M) + \frac{1}{2} D_{KL}(Q||M) \quad (4)$$

$$M = \frac{1}{2}(P + Q) \quad (5)$$

To approximate the probability density functions (pdf) of features and compute the probability distributions, a kernel density estimation approach was used. Kernel density estimation or KDE is a more robust reflection of the actual data properties than the histogram. The parameters that describe a kernel density estimation are the kernel type and the kernel bandwidth. The kernel specifies the shape of the distribution placed at each point. Intuitively, it smoothes each data point into a small density bump and by computing the sum of all these bumps, the final density estimation is obtained [8]. The kernel type used is the Gaussian kernel.

The kernel bandwidth controls the size of the kernel at each point. Similar to the number of bins of a histogram, it has to be very carefully chosen in order not to misinterpret the underlying structure of data. A suitable method for finding the optimal bandwidth is to use a grid search cross-validation approach.

Regarding the implementation details, after obtaining the probability distributions, the Jensen-Shannon divergence matrix of size $(N \times N)$ was computed. $N$ represents the number of features.

The Jensen-Shannon divergence score, the weight assigned to one feature was calculated as the average of all divergence scores between it and all the other features:

$$w_{JS}(i) = \frac{\sum_{j=1}^{N} D_{JS}(P_i||P_j)}{N} \quad (6)$$

where $i$ represents the index of the feature of interest and $P_i$ and $P_j$ the probability distributions of features with indexes $i$ and $j$.

### C. Solution overview

As stated in the introduction, the new approach of feature extraction is based on PCA. A good practice before applying PCA is to center the data to zero. This operation is done by subtracting the mean of every feature from the data points.

Another preprocessing step which is generally indicated is feature scaling or standardization. After some experiments, we have concluded that in the context of Spike Sorting, standardizing the data is not useful because the variability of features matters and by standardizing the data each feature receives equal importance while our objective is to grant a higher weight to the important features.

The goal of increasing the **separability between clusters** is achieved by maximizing the contribution of **multimodal** and **distant** features' distributions. The total weighting formula for a feature (7) is designed as the product between the weight computed using the peak criteria (1) and the one using the divergence information (6).

$$w(i) = w_p(i) * w_{JS}(i) \quad (7)$$

Increasing the variance of the multimodal and distant features means modifying their contribution in the computation of new axes calculated by PCA. The correlation between the original features and the principal components can be assessed by computing the correlation coefficient [9] defined below:

$$r_{ik} = \frac{\sqrt{e_{ik}^2 \lambda_k}}{\sigma_i} \quad (8)$$

Regarding the parameters, $i$ represents the feature index and $k$ the principal component index. The eigenvector value $(e_{ik})$ multiplied by its eigenvalue $(\lambda_k)$ is also known as loading and in case the data is not normalized, it has to be divided by the standard deviation of the original feature $(\sigma_i)$.

## IV. Evaluation

### A. Dataset

The data used in this work was created by a research group from the Department of Engineering of the University of Leicester, United Kingdom [10]. The data consists of 95 synthetic datasets, also referred to as simulations, having between 2 and 20 clusters. To replicate a real scenario, a cluster representing the multi-unit cluster representing the noise, is present in each simulation. The other clusters represent the activity of a single-unit, consisting in spikes generated by a particular neuron. Both single-unit and multi-unit clusters were generated by using a database of 594 different spikes collected from recordings of a monkey neocortex and basal ganglia. Compared to a realistic scenario, the synthetic dataset contains ground truth, allowing a performance measurement using external clustering validation methods.

Originally, each waveform was sampled at 96 KHz, followed by a downsampling to 24 KHz. A sampling frequency of 96 KHz corresponds to 316 datapoints. Therefore, after downsampling, a spike waveform is described by 79 data

points, representing the magnitude, the voltage expressed in mV.

Preprocessing the data consists in aligning the spikes to their maximum value, to their peaks. This step is performed in order to avoid overclustering (separating a cluster into more clusters) as presented in [11].

### *B. Clustering validation metrics*

The results were measured using both external [12] and internal validation metrics [13].

Regarding the external validation the labels returned by the clustering algorithm were compared against the labels from ground truth. External methods can be categorised as pair-counting measures such as **Adjusted Rand-Index**, information theory based measures including **Adjusted Mutual Information** and Variation of Information and set-matching based measures, such as H criterion [12].

Rand-Index (RI) measures the agreements between pairs of points of two clusterings. It is defined at (9) where $N_{00}$ represents the number of agreements and $N_{11}$ the number of disagreements between the two partitionings. The denominator represents the total number of pairs.

$$RI = \frac{N_{00} + N_{11}}{\binom{n}{2}} \tag{9}$$

Mutual Information (MI) is a symmetric measure that quantifies the similarity, the information shared by two random variables. It is based on entropy and measures how much knowing one variable reduces the uncertainty about the other. In the clustering context, the random variable is represented by a cluster. Below (10) is presented the mutual information between two variables $X$ and $Y$ using Shannon entropy (3).

$$MI(Y, X) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \frac{p(y, x)}{p(x)p(y)} \tag{10}$$

The **corrected-for-chance** versions of Rand Index and Mutual Information are recommended because of the constant baseline property which improves the interpretability of results [14]. Both methods Adjusted Rand Index (ARI) and Adjusted Mutual Information (AMI) are normalized and upper-bounded by 1, meaning perfect matching and lower-bounded by -1. A score equal to 0 suggests a random clustering.

Internal validation is used when ground truth is not available and it is based on two criteria: **compactness** and **separation**. The goal of a clustering is to group similar objects in the same clusters and separate the unrelated ones. Therefore, compactness describes how related the points from a cluster are. It is usually measured using the variance (low variance suggests better compactness) or by computing the maximum pairwise distances. Separation measures how well-defined are the clustering partitions. It can be calculated either by using density or by computing the pairwise distances between the cluster's centers.

Silhouette (S) score measures the clustering performance using the pairwise difference of the within-cluster distances and the distances to the nearest cluster. The formula for computing it for a point is presented below (11): *a* parameter represents the average distance between a point and the other points from the same cluster and *b* denotes the average distance between that point and all the points from the nearest cluster.

$$S = \frac{b - a}{max(a, b)} \in [-1, 1] \tag{11}$$

The overall Silhouette coefficient is computed by averaging the scores of each point. It takes values in the range [-1, 1] and a result of 0 denotes overlapping clusters. An optimal clustering is obtained by maximizing the value of this score.

Calinski-Harabasz (CH) score measures the clustering performance by computing the ratio of the average of the inter and intra cluster dispersion defined as the sum of squares. The index measures compactness using the sum of the distances between the points and the cluster's center and separation using the maximum distance between the centers of all clusters [13]. It takes positive values and is unbounded. A larger score suggests a good clustering considering that the distances between clusters have to be larger and the within-distances have to be lower for an optimal result.

Davies-Bouldin (DB) index is computed as the average of the maximum similarity between clusters (13). A lower value indicates better separation between clusters. The similarity is defined as $R_{ij}$ by (12) where $s_i$ denotes the average distance between each point of cluster $i$ and the centroid of that cluster (same meaning for $s_j$) and $d_{ij}$ represents the distance between the centroids $i$ and $j$.

$$R_{ij} = \frac{s_i + s_j}{d_{ij}} \tag{12}$$

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{i \neq j} R_{ij} \tag{13}$$

### *C. Results*

The evaluation was performed on 37 simulations, 28 simulations having up to 8 clusters and the rest having 9, 10, 11, 12, 15, 16, 17 and 19 clusters.

The performance of the feature extraction method was first evaluated using the internal validation metrics (Silhouette, Davies-Bouldin and Calinski-Harabasz) directly on the ground truth in order to avoid the bias introduced by the clustering algorithm.

The table below presents the **average difference** (Avg Delta) between the results of the **weighting method** and **classical PCA 2D** applied on the ground truth labeled data in order to observe the performance of the feature extraction method avoiding the bias introduced by a clustering algorithm. The first row presents the results using the simulations that

have less than 8 clusters and the second row presents the results obtained on the subset of 37 simulations.

Increased attention was paid to Silhouette score because it is more interpretable being normalized.

TABLE I
INTERNAL VALIDATION SCORES ON GROUND TRUTH

| | *Avg Delta S* | *Avg Delta DB* | *Avg Delta CH* |
|---|---|---|---|
| *$\leq$ 8 clusters simulations* | **0.098** | 0.048 | 5598.454 |
| *all tested simulations* | **0.07** | 0.131 | 4382.249 |

The performance was assessed also from the graphical representations of data. In Fig. 1, are presented the data points of simulation 8 after using classical PCA 2D as feature extraction, in the left and after applying **weighting on features then PCA 2D**, in the right. From these figures, it can be observed that the **separability** between the overlapping white and red clusters is **increased**.

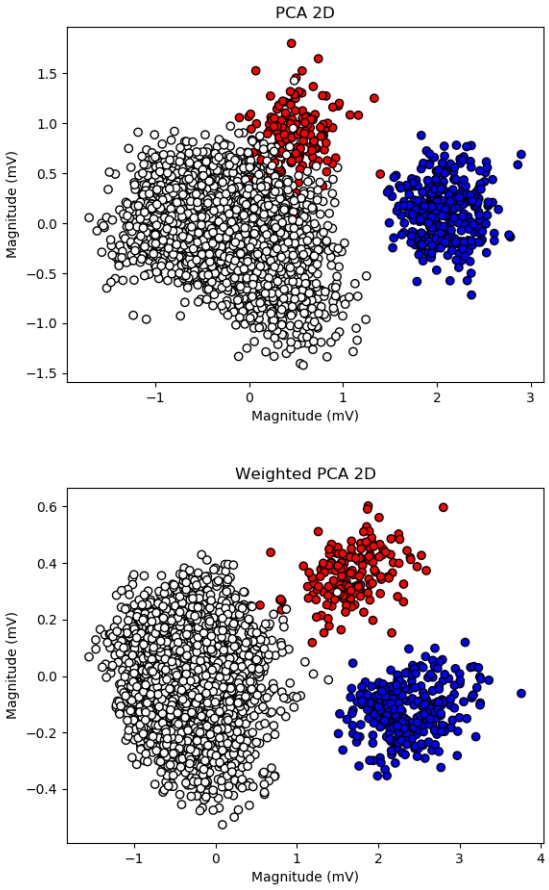


Fig. 1. Simulation 8 after applying **PCA 2D** (up) and after applying **Weighted PCA 2D** (down)

Another simulation that shows a better performance than PCA 2D is simulation 46, illustrated in Fig. 2.

The new approach of feature extraction was validated also after applying the clustering algorithms: K-Means–partitioning

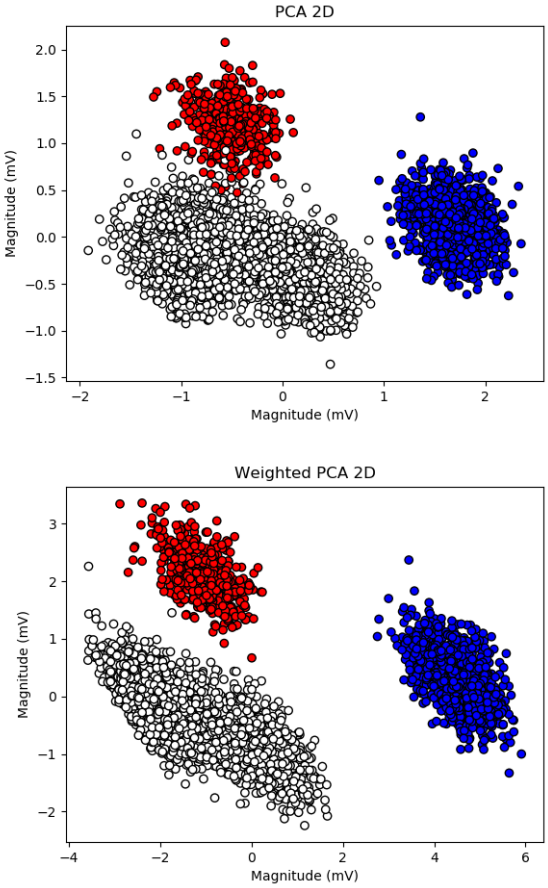


Fig. 2. Simulation 46 after applying **PCA 2D** (up) and after applying **Weighted PCA 2D** (down)

based and Space Breakdown Method (SBM)–density based algorithm [15]. The latter one was developed especially for neural data challenges. The evaluation was performed using both internal (Silhouette, Davies-Bouldin and Calinski-Harabasz) and external clustering metrics (Adjusted Rand Index and Adjusted Mutual Information). Regarding the external validation, the authors of SBM [15] propose an evaluation without considering the noise points (nnp setting) left by the algorithm as belonging to no cluster.

The tables below present the **average differences** between the **weighting method** and **PCA 2D** on which the two clustering algorithms were applied. It can be observed that the weighting method presents an increase in scores no matter the clustering method. Between the 2 clustering algorithms, SBM performs better [15].

TABLE II
EXTERNAL VALIDATION SCORES FOR CLUSTERING ALGORITHMS

| | *Avg Delta ARI* | *Avg Delta AMI* | *Avg Delta ARI-nnp* | *Avg Delta AMI-nnp* |
|---|---|---|---|---|
| ***K-Means*** | 0.044 | 0.025 | 0.044 | 0.025 |
| ***SBM*** | 0.087 | 0.061 | 0.089 | 0.069 |

TABLE III
INTERNAL VALIDATION SCORES FOR CLUSTERING ALGORITHMS

| | *Avg Delta S* | *Avg Delta DB* | *Avg Delta CH* |
|---|---|---|---|
| ***K-Means*** | 0.005 | -0.004 | 7284.433 |
| ***SBM*** | 0.054 | -0.17 | 2773.211 |

## V. DISCUSSION AND CONCLUSIONS

According to the results presented in the previous section, the new approach of feature extraction achieves better performance in average than classical PCA, the increase in separability between clusters being visible also in Fig. 1 and Fig. 2. Better results are obtained for simulations with less than 8 clusters, but besides the number of clusters, the exposed complexities matter too. The variance percentage of the first two principal components is increased by using the new approach.

The method presented in this work can be improved by taking into consideration other characteristics of a multimodal distribution. Another future improvement would be to apply the idea of multimodality using Wavelets [16] or combine the feature extraction method with other feature extraction methods from frequency domain: Fourier or Hilbert. Improvement is welcomed also in the clustering step, focusing on algorithms that can benefit from the new feature extraction approach. Last but not least, the complexity of the clustering problem highly depends on the particularities and challenges presented by the dataset. As aforementioned, the presented approach was based on a synthetic dataset, while a real dataset would increase the complexity, as more challenges are present and the ground truth is not available.

## VI. ACKNOWLEDGEMENTS

This work was partially funded by: NO Grants 2014-2021, under Project contract number 20/2020 (RO-NO-2019-0504), three grants from the Romanian National Authority for Scientific Research and Innovation, CNCS-UEFISCDI (codes PN-III-P2-2.1-PED-2019-0277, PN-III-P3-3.6-H2020-2020-0118, PN-III-P3-3.6-H2020-2020-0109), and a H2020 grant funded by the European Commission (grant agreement 952096, NEUROTWIN)

## REFERENCES

[1] B. P. Bean, "The action potential in mammalian central neurons", Nature Reviews Neuroscience, vol. 8, 2007
[2] H. Hotelling, "Analysis of a complex of statistical variables into principal components", Journal of Educational Psychology, vol. 24, pp. 417—441 & 498—520, 1933
[3] L. van der Maaten and G. Hinton, "Visualizing data using t-sne", Journal of Machine Learning Research, vol. 9, pp. 2579—2605, 2008
[4] D. Freedman and P. Diaconis. "On the histogram as a density estimator: L2 theory", Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete. vol. 57, pp. 453-–476, 1981
[5] S. Kullback and R. A. Leibler, "On information and sufficiency", Ann. Math. Statistics, vol. 22, nb. 1, pp. 79-–86, 1951
[6] T.M. Cover, J.A. Thomas, "Elements of Information Theory", John Wiley & Sons: Hoboken, NJ, USA, pp. 13–57, 2012
[7] J. Lin, "Divergence measures based on the Shannon entropy", IEEE Trans. Inf. Theory, vol. 37, pp. 145–151, 1991
[8] Y.-C. Chen. Lecture 6: Density estimation: "Histogram and kernel density estimator", 2018, unpublished
[9] I. T. Jolliffe and C. Jorge, "Principal component analysis: a review and recent developments", Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 2016, doi:10.1098/rsta.2015.0202
[10] C. Pedreira, J. Martinez, M. J. Ison, R. Q. Quiroga, "How many neurons can we see with current spike sorting algorithms? ", Journal of Neuroscience Methods, vol. 211, pp. 58–65, 2012, doi:10.1016/j.jneumeth.2012.07.010
[11] R. Q. Quiroga. "Spike sorting". Scholarpedia, vol. 3, nb. 12, p. 3583, 2007
[12] N. X. Vinh, J. Epps, and J. Bailey, "Information theoretic measures for clusterings comparison: Is a correction for chance necessary? " In Proceedings of the 26 th International Conference on Machine Learning, 2009.
[13] Y. Liu, Z. Li, H. Xiong, X. Gao, and J. Wu, "Understanding of internal clustering validation measures", IEEE International Conference on Data Mining, pp. 911–916, 2010.
[14] S. Romano, N. X. Vinh, J. Bailey, and K. Verspoor, "Adjusting for chance clustering comparison measures", Journal of Machine Learning Research, pp. 1–32, 2016
[15] E. R. Ardelean, A. Stanciu, M. Dinsoreanu, R. Potolea, and C. Lemnaru, "Space Breakdown Method. A new approach for density-based clustering", IEEE 15th International Conference on Intelligent Computer Communication and Processing, 2019, doi: 10.1109/ICCP48234.2019.8959795.
[16] H. G. Rey, C. Pedreira, R. Q. Quiroga, "Past, present and future of spike sorting techniques", Brain Research Bulletin, vol. 119, Part B, pp. 106–117, 2015, doi: 10.1016/j.brainresbull.2015.04.007

## VII. APPENDIX

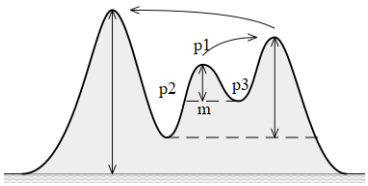


Fig. 3. Prominence highlighted.