Noname manuscript No.
(will be inserted by the editor)

Music genre profiling based on Fisher manifolds and Probabilistic Quantum Clustering

Raúl V. Casaña-Eslava · Ian H. Jarman · Sandra Ortega-Martorell · Paulo J. G. Lisboa · José D. Martín-Guerrero

Received: date / Accepted: date

Abstract Probabilistic classifiers induce a similarity metric at each location in the space of the data. This is measured by the Fisher Information Matrix. Pairwise distances in this Riemannian space, calculated along geodesic paths, can be used to generate a similarity map of the data. The novelty in the paper is twofold; to improve the methodology for visualisation of data structures in low-dimensional manifolds, and to illustrate the value of inferring the structure from a probabilistic classifier by metric learning, through application to music data. This leads to the discovery of new structures and song similarities beyond the original genre classification labels. These similarities are not directly observable by measuring Euclidean distances between features of the original space, but require the correct metric to reflect similarity based on genre. The results quantify the extent to which music from bands typically associated with one particular genre can, in fact, crossover strongly to another genre.

Raúl V. Casaña-Eslava
Liverpool John Moores University, 3 Byrom Street, Liverpool, Merseyside L3 3AF, UK
Tel.: +0151 231 2777
E-mail: raulcasana@gmail.com

Ian H. Jarman
Liverpool John Moores University
E-mail: i.h.jarman@ljmu.ac.uk

Sandra Ortega-Martorell
Liverpool John Moores University
E-mail: s.ortegamartorell@ljmu.ac.uk

Paulo J. G. Lisboa
Liverpool John Moores University
E-mail: p.j.lisboa@ljmu.ac.uk

José D. Martín-Guerrero
Departament d'Enginyeria Electrònica - ETSE, Universitat de València (UV), Av. Universitat, SN, 46100 Burjassot, València, Spain
E-mail: jose.d.martin@uv.es

Keywords Fisher metric · Manifold structure · Multidimensional scaling · Probabilistic quantum clustering · Music information retrieval

1 Introduction

A unique property of probabilistic classifiers stems from the fact that the metric is induced in either the space of model parameters, or the space of data. This does not apply to classifiers arising from computational learning theory, for instance Support Vector Machines, which work on the basis of discriminant vector spaces. The metric is linked to the Fisher information matrix which is calculated directly from the conditional probabilities inferred from the model. This property is most often ignored but it holds the key to derive important properties about the data structure, which provide insights on the question addressed by the classifier. In binary or multinomial classification, the question addressed by the classifier is the probability of class membership of any given test point. Therefore the metric will make explicit the similarity structure of the data, by weighting each input variable precisely according to the information it contains about class membership. New questions may be asked by changing the class labels and so the data structure will change accordingly. In other words, the Fisher information opens the door to the discovery of knowledge that is otherwise implicit in the scalar output of the model.

This paper illustrates this process by using class membership of three musical genres as the driver to map the structure of recordings from the Million Song Dataset (MSD). This will show how some songs appear at the extremes of the distribution of the data, reflecting a genre that might be considered close to pure, while others lie between genres, so indicating a fusion of two or more types. This is helpful in the context of the MSD as the genre can be associated with the band that plays the song, when reality can be quite different. So music from say the Rolling Stones is generally classed under Pop Rock, but at least one instance can be found halfway to Rap. Listening to the song Cherry Oh Baby confirms that this is the correct assignment of genre.

The main novelty in the paper is to improve the methodology of the original derivation of Fisher information from estimates of the posterior probability modelled by the Multi-Layer Perceptron (MLP) [42] by the use of classical Multidimensional Scaling (cMDS) [11,51] to derive a faster and more stable Euclidean embedding of the data structure, compared with the use of Sammon mapping in the original paper. The Euclidean embedding is obtained from the pairwise distances along geodesics of the Riemannian space induced by the Fisher metric. This is crucial to enable the application of projective methods, such as clustering approaches, to profit from the re-scaling of the dimensions for each variable according to their importance from classification. This provides a practical solution to a general question: how can we mitigate the bias suffered by clustering methods as a result of the potentially arbitrary distance measure, typically calculated as the Euclidean distance between data points, when the scale of the axis for each variable is set by the user without

reference to any objective principle? The answer we provide is that in the context of allocating data to two or more classes with a probabilistic model, an appropriate metric can be calculated which will enable clustering (or perhaps semi-supervised clustering) to be carried out in a principled manner.

The second improvement to the methodology is the application of Quantum Clustering [18] to map out the data density in the embedded Euclidean manifold. This is made possible by an extension of the method that sets it within a probabilistic framework [7]. The aim in doing so was to provide objective measures to quantify how well each clustering solution fits the data. In this paper, we show that Probabilistic Quantum Clustering (PQC) is well suited to fine tune the granularity of the clustering to the structure of the data.

The third novelty is the application of the Fisher information to derive the structure of songs originally labelled with one of three common genres, Pop Rock, Rap and Jazz. The paper will show that songs lie on a continuum that links all three genres, with clusters of higher population density appearing at the corners of this triangle. To our knowledge, this is the first time that the manifold of data induced by genre-specific content has been mapped explicitly.

Finally, an important element of the paper is to use this case study to validate the plausibility of the results and show the value that Fisher information has for knowledge discovery from data sets, by exposing the richness of structure that is concealed within the numerical predictions from the model.

The MSD set was used as the source for our study cohort because individual songs often combine multiple musical genres. We provide a principled method to derive and visualise the manifold of musical data represented in the data set across the three specific genres. Neighbourhood structures within the map of songs can potentially be used by subject experts to validate the prediction made by the MLP. This is the final, and arguably most important, message of the paper, namely that it is possible to find graphical ways of interpreting the operation of neural networks using data structures that end users can understand and reason with; and this enables end users to validate the operation of the model, by testing whether the plausibility of the similarities that the model induces in the data.

Under the scope of Music Information Retrieval (MIR), the dataset comprises spectral features of multiple songs which are labelled by music genre. We will use this label for classification in order to generate the Fisher manifold onto which the songs are projected to obtain a genre-informed low-dimensional visualisation and clustering of the data.

The MSD has been one of the first benchmark datasets for large-scale applications in the MIR research, where historically there have been difficulties in sharing information among the research community due to copyright issues. The original MSD comes with a set of features extracted by the API of The Echonest [1] and meta-data songs; nevertheless, the audio files are not easily accessible and the features provided by The Echonest services are lim-

[1] `http://the.echonest.com/`

ited. For this reason, some works [47,48] focused on adding more features, including features of temporal domain, ground truth assignments and labels for supervised machine learning tasks. The features and meta-data are publicly available without copyright restrictions, helping to extend the use of the dataset as a benchmark. One of the most popular tasks in Music Information retrieval research is musical genre classification, which is where the present work fits in.

Clustering and segmentation will apply PQC [7]. This is a Bayesian extension of the Quantum Clustering (QC) method [18] that provides an objective function for setting the main adjustable parameter in QC, namely the length scale inherent in the Schrödinger equation, which drives the granularity of the resulting clusters.

MIR has evolved considerably in the last two decades [45], not only extracting information from audio signals, but also from contextual data sources, user experience, folksonomy or collaborative tags [8]. More recently and powered by the music streaming services, models have been shifted from system-centric towards user-centric designs, focusing on aspects like novelty, popularity, serendipity or location/time-awareness, all concepts that are very useful for recommender systems [59].

More related to the proposals made in this work, the comparison between a human classification and an automatic one has also been a topic of study since the first works on MIR [52]. However, within user-centred frameworks, tasks like music similarity or genre classification still remain without a clear consensus [29,49]. Different studies [53,44,54,21] state that human agreement on the similarity of two music pieces is roughly bounded at 80%.

Different music similarity measures have been evaluated in the context of recommender systems [3], where results using audio content-based distances can be comparable to results based on high-level semantic measures formed by Support Vector Machine (SVM) classifiers. Genre classification can also be tackled using text-based features [46,25] or a combination of content-based features with temporal/co-occurrence context-based ones in order to improve auto-tagging tasks with conditional Restricted Boltzmann Machines [32,35], a weighted vote k-Nearest Neighbour classifier [50,24] or random forest classifiers [49].

Some works have made use of a user-interface map representation as music browser to enhance the user experience; for instance, a Self-Organizing Map (SOM) on content-based features to reduce the dimensionality is presented in [26]; [31] uses a exponential similarity model to heuristically combine three distances of different nature: content-based, metadata-based and collaborative tag distances; the combined pairwise distances are embedded into a Multidimensional Scaling (MDS) with high dimensionality to later apply a SOM for visualization. Other music browsers with a similarity map visualization can be found in [14,16].

Our approach differs from all these previous works that are mainly user-centred models within the framework of recommender systems. As far as the

authors' knowledge, this paper is the first attempt to map the manifold of data induced by genre-specific content explicitly.

The rest of the paper is now outlined. Section 2 introduces the three methods applied in this paper, namely Fisher Information (FI), classical multidimensional scaling (cMDS) for embedding the Fisher manifold, and PQC to segment the data. The data projections form a continuum, hence the purpose of clustering is to identify regions with high data density, for which PQC is efficient. Section 3 describes the data set and the procedure to select benchmark datasets based on its MIR features. Section 4 summarizes the proposed methodology in a pipeline. The experimental results are presented and discussed in section 5, the limitations and challenges of the implementation in section 6, ending up the paper with the conclusion of the work in section 7.

2 Methodology

2.1 Fisher manifold

2.1.1 Fisher metric introduction

Generally, most of the works in the literature that involve the FI metric [2, 20, 41, 1, 27, 4] use the metric defined in parameter manifolds based on generative models, $p(\mathbf{x}|\theta)$. However based on [22, 23], it is possible change the approach and apply the Fisher metric on discriminative models $p(y|\mathbf{x})$ that classify an external auxiliary information y. In this case, the metric measures parameter distortions with reference to the input space $\mathbf{x}$ instead of the parameters θ:

$$\begin{aligned} d\left(\mathbf{x},\, \mathbf{x}+d\mathbf{x}\right)^2 &= d\mathbf{x}^T \mathbf{FI}\left(\mathbf{x}\right) d\mathbf{x} \\ \mathbf{FI}\left(\mathbf{x}\right) &= E_y\left[\nabla_{\mathbf{x}} \log p(y|\mathbf{x}) \cdot \nabla_{\mathbf{x}} \log p(y|\mathbf{x})^T\right] \end{aligned} \tag{1}$$

Generally, the auxiliary information y represents a class label C composed by J discrete values, $y \in [c_1, \cdots, c_J]$. The probability function $p(y|\mathbf{x})$ represents the discrete probability distribution conditioned on $\mathbf{x}$, where the expected value E_y over $p(y|\mathbf{x})$ is the summation of FI matrix over each class in $p(y|\mathbf{x})$:

$$\mathbf{FI}\left(\mathbf{x}\right) = \sum_{j=1}^{J} \nabla_{\mathbf{x}} \log p(c_j|\mathbf{x}) \cdot \nabla_{\mathbf{x}} \log p(c_j|\mathbf{x})^T \, p(c_j|\mathbf{x}) \tag{2}$$

This metric measures local distances in the input space $d\mathbf{x}$ as a function of the variations on the class probabilities, assigning longer distances in the direction of the posterior probabilities that have more variation, and shorter distances where there are no class probability changes. In other words, the FI metric contains local relevant information about the probability rate of change of class y membership.

2.1.2 Choice of discriminative model for the FI matrix

The form of the FI matrix (eq. 2) strictly depends on the election of the discriminative model $p(y|\mathbf{x})$. There are some constraints that the discriminative model must follow:

- Be a multinomial classifier with a probabilistic output.
- Be able to deal with non-linear data.
- Have a mechanism to avoid over-fitting.
- Allow the model to be easily differentiable up to second order with respect to the input space. This requirement is due to the fact that the Fisher Information matrix [41] can be derived as the Hessian of the KL-divergence (also called relative entropy), implying derivatives of second order.

Because of these constraints, a Multi-Layer Perceptron (MLP) [17] classifier regularized with weight decay was chosen for the implementation. Although the MLP fits very well the above-mentioned requirements indeed, other classifiers could also be chosen; for instance, the vanilla SVM [10] do not have a probabilistic output, but they can be adapted [40] and can be useful for feature selection [39].

Using the MLP as a discriminative model, the posterior probability estimation is evaluated with the soft-max activation:

$$p(c_j|\mathbf{x}) = \frac{\exp\left(a_j(\mathbf{x})\right)}{\sum_{k=1}^{J}\exp\left(a_k(\mathbf{x})\right)} \tag{3}$$

where c_j are the J different class labels, and a_j are the MLP outputs described in the following expression:

$$a(\mathbf{x}) = \mathbf{W}^O \cdot \mathbf{\Theta}\left(\mathbf{W}^H \cdot \mathbf{x} + \mathbf{B}^H\right) + \mathbf{B}^O \tag{4}$$

where $\mathbf{W}$ and $\mathbf{B}$ are the MLP weights of the hidden layer (H) and output layer (O), and $\mathbf{\Theta}(z)$ is the sigmoid function.

Using the eq. 3 in the FI matrix (eq. 2) and applying derivatives, the FI matrix can be expressed as a function of the MLP output, a_j, in the following equation:

$$\mathbf{FI}\left(\mathbf{x}\right) = \sum_{j=1}^{J}\sum_{k=1}^{J}\sum_{l=1}^{J}\nabla\left(a_j - a_k\right)\nabla\left(a_j - a_l\right)^T p_j p_k p_l \tag{5}$$

A more detailed derivation can be found in appendix A.

The MLP architecture is set empirically with one hidden layer of 10 neurons, which provides good enough results to discriminate any non-linear shape without adding too much model complexity. The other hyper-parameters are listed below:

- MLP architecture: One hidden layer of 10 neurons
- Neuron activation: Sigmoid

– Learning rate: 0.001
– Momentum: 0.9
– Weight decay: 0.05
– Maximum epochs: 2000

The weights are updated by training a log-likelihood objective function with a regularized back-propagation:

$$\epsilon_{LL} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{J} c_j(\mathbf{x}_i)\log\left(p(c_j|\mathbf{x}_i)\right) + \left(1 - c_j(\mathbf{x}_i)\right)\log\left(1 - p(c_j|\mathbf{x}_i)\right) \quad (6)$$

With the eq. 5 the metric is estimated locally, computing differential distances as:

$$d(\mathbf{x},\, \mathbf{x} + d\mathbf{x})^2 = d\mathbf{x}^T \cdot \mathbf{FI}(\mathbf{x}) \cdot d\mathbf{x} \quad (7)$$

2.1.3 Fisher pairwise distances

The Fisher manifold is formed by the pairwise distances between observations under the Fisher metric. Because the metric is variable depending on the input space, two approximations are needed to calculate the geodesic distances, one to estimate local distances and another for global distances. It should be taken into account that there is an inherent problem of scalability when measuring pairwise distances due to the $\mathcal{O}(n^2)$ sample size dependence; this problem was already addressed in [6].

The first approximation estimates local distances by path integrals, where the path is a straight line between observations and the metric is sampled across this path.

In more detail, if two points are close enough, the distance between them can be approximated as:

$$d(\mathbf{x}_A, \mathbf{x}_B)^2 \approx (\mathbf{x}_B - \mathbf{x}_A)^T \cdot \mathbf{FI}\left(\frac{\mathbf{x}_B + \mathbf{x}_A}{2}\right) \cdot (\mathbf{x}_B - \mathbf{x}_A) \quad (8)$$

Since usually the points are not close enough, the theoretical solution for this case is to use the path integral:

$$d(\mathbf{x}_A, \mathbf{x}_B) = \left| \int_{t_A}^{t_B} \sqrt{\dot{\mathbf{x}}(t)^T \cdot \mathbf{FI}\left(\mathbf{x}(t)\right) \cdot \dot{\mathbf{x}}(t)}dt \right| \quad (9)$$

The MLP density estimators, $a(\mathbf{x})$, are non-linear functions of $\mathbf{x}$ making the path integral impossible to be solved analytically. Therefore, the distances must be estimated numerically, which approximates the path to a straight line that connects both points, being $\mathbf{FI}(\mathbf{x})$ evaluated by taking T samples across the path. The total distance is approximated as the sum of T small segments computed like in eq. 8.

$$d_T\left(\mathbf{x}_A, \mathbf{x}_B\right)^2 = \sum_{t=1}^{T} d\left(\mathbf{x}_A + \frac{t-1}{T}\left(\mathbf{x}_B - \mathbf{x}_A\right),\ \mathbf{x}_A + \frac{t}{T}\left(\mathbf{x}_B - \mathbf{x}_A\right)\right)^2 \quad (10)$$

The quantity of segments T can be set empirically; one option with a good trade-off in runtime versus accuracy is fixing the segments number in $T = 10$, where the segments will have a variable length depending on the Euclidean distances between $\mathbf{x}_A$ and $\mathbf{x}_B$.

The second approximation estimates global distances by shortest path algorithms applied to a fully connected network formed by nodes (observations) and edges (local distances).

The Floyd-Warshall algorithm [13,56] was used; it is an algorithm of the kind All-Pairs-Shortest-Path (APSP) based on weighted graphs. In our case, the nodes are the data samples and the edges are the local pairwise distances, creating a fully connected graph. The global distances are found by searching paths through previously calculated edges, thus shortening those global distances.

With these two approximations a manifold with geodesic distances can be estimated, forming an adjacency matrix that defines the structure of the Fisher manifold.

2.2 Multidimensional scaling

Fisher manifolds are usually embedded into a low-dimensional space (typically in two or three dimensions) for the purpose of visualization. Our preferred method is to transform the pairwise distances of the Riemannian manifold into coordinates, embedded in a Euclidean space.

A commonly used embedding is Sammon Mapping [43]. However, this method is computationally expensive for large data sets and can be unstable in the sense that removing only a few points can significantly change the overall map. We will apply classical Multidimensional Scaling (cMDS) [11,51, 57], also known as Principal Coordinate Analysis (PCoA) [15]. This method is computationally efficient and much less affected by small changes in the data set. It is a powerful method for preserving the global structure of the manifold, but it needs more dimensions to properly embed a Riemannian manifold with the advantage of gaining mapping accuracy. It gives information about the eigenvalues associated with each eigenvector of the Euclidean embedding, measuring the relative importance of each dimension, and therefore discarding the least relevant dimensions given a threshold of the cumulative sum of the eigenvalues, thus providing a clear indication of the number of dimensions that are needed to effectively map the Fisher manifold.

From a theoretical standpoint, it is important to mention the Nash embedding theorem [36,37], which states that every Riemannian manifold of dimension D can be isometrically embedded into a Euclidean space of dimension M, where $M \geq D + 1$. Isometric meaning the length of every path is preserved.

In the case of cMDS, it tries to find a solution $X = [\mathbf{x}_1, \cdots \mathbf{x}_N]$ hence $d_{i,j} = ||\mathbf{x}_i - \mathbf{x}_j||$, where $\mathbf{x}_i \in \mathbb{R}^M$ and $M \geq N - 1$. The solution can be expressed as a function of the $N \times N$ Gram matrix $B = X^T \cdot X$, where now the distances depend on B:

$$d_{ij}^2 = b_{ii} + b_{jj} - 2b_{ij} \tag{11}$$

where the expression has been obtained taking into account $||\mathbf{x}_i - \mathbf{x}_j||^2 = \mathbf{x}_i^2 + \mathbf{x}_j^2 - 2\mathbf{x}_i\mathbf{x}_j$, and considering the assumption of centred configuration:

$$\sum_{i=1}^{N} \mathbf{x}_{ik} = 0 \quad \forall \quad k \tag{12}$$

This assumption will serve as a constraint for obtaining a unique solution, and for the purpose of dimensionality reduction. Summing for all variables in eq. 11, using the constraint eq. 12 and rearranging the terms, the final solution can be obtained:

$$b_{ij} = \frac{-1}{2\left(d_{ij}^2 - \sum_i d_{ij}^2 - \sum_j d_{ij}^2 + \sum_i \sum_j d_{ij}^2\right)} \tag{13}$$

If B is decomposed by its eigenvectors, $B = V \cdot \Lambda \cdot V^T$, then $X = \Lambda^{1/2} \cdot V^T$. In this way, X is expressed as the eigenvectors of B, allowing a dimensionality reduction similar to PCA, just discarding the eigenvectors whose eigenvalues have less weight (variance). In fact, the coordinates are ordered from the largest to the smallest variances, allowing any dimension from 1 to M to be selected.

The distance d_{ij} is called a Euclidean distance if there exists a finite M:

$$d_{ij} \equiv ||\mathbf{x}_i - \mathbf{x}_j|| \quad \forall \quad i, j \tag{14}$$

Otherwise, d_{ij} is called a non-Euclidean distance, which is the case of the distances obtained in the Fisher manifold.

For Riemannian distances (non-Euclidean) some of the eigenvalues of B are negative, hence these eigenvectors are discarded. In all the cases where the cMDS with the Fisher manifold has been tested, M is quite high but the eigenvalues present an exponential decay with a long tail, with the smallest eigenvalues being negative. Therefore, they carry little variance; in practice, only the first two or three eigenvalues are kept, particularly when the accumulated sum of variance is greater than 80%. On the other hand, when the cMDS is applied to Euclidean pairwise distances the same results as the PCA are recovered, with no eigenvalue being negative.

2.3 Probabilistic Quantum Clustering

A probabilistic approach to QC by means of wave functions comprising normalised joint probability distributions is proposed. This enables the parameters for local covariance estimation to be optimised by maximising a Bayesian probability of cluster allocation.

2.3.1 Introduction to original Quantum Clustering

Quantum Clustering (QC), originally proposed in [19], is a paradigm to find clusters or data profiles based on the Schrödinger equation, Eq. (15), one of the cornerstones of Quantum Mechanics. In particular, Eq. (15) generates a potential function $V(\mathbf{x})$ from a wave function $\Psi(\mathbf{x})$ as a constant energy solution of the Schrödinger equation:

$$H\Psi \equiv \left(-\frac{\sigma^2}{2}\nabla^2 + V(\mathbf{x})\right)\Psi(\mathbf{x}) = E\Psi(\mathbf{x}) \tag{15}$$

where H is the Hamiltonian, E the energy, and σ is a length scale parameter assciated with the wave function. Therefore, the potential can be expressed as:

$$V(\mathbf{x}) = E + \frac{\sigma^2}{2}\frac{\nabla^2\Psi(\mathbf{x})}{\Psi(\mathbf{x})} \tag{16}$$

QC has the potential to match complex data structure by connecting neighbouring points by defining a potential function derived from a Parzen density estimator. The key idea is to associate clusters with potential wells, determined by identifying the connected regions around that potential. The main advantage of such an approach is that clusters with different shapes can be found by connecting nearby points together using the potential as a smoothing function.

The data points are allocated into clusters performing a stochastic gradient descend (SGD) over the potential to find the potential wells (some of them can be local minima), which are identified as clusters.

Nevertheless some aspects of QC remain open questions. In particular, the accuracy in matching the correct data density structure depends strongly on the assumed length scale [5]. However, when heteroscedasticity is present this length scale will vary across feature space. Therefore, it is necessary to estimate the length scale locally for which several methods of local covariance estimation have been proposed [28,55,58].

2.3.2 The use of local length scales

One of the main novelties of the PQC is the introduction of a local estimation of the length scale and a probabilistic approach of the cluster allocation. These improvements will allow to define a likelihood function of cluster membership introduced in next subsection.

In order to deal with heteroscedastic data, information about local density changes is required. This can be done by means of the length scale by setting σ not to a a constant value, but as a function of the KNNs:

$$\sigma_i \equiv \frac{1}{K} \sum_{j \in knn(\mathbf{x}_i)}^{K} dist(\mathbf{x}_i, \mathbf{x}_j) \tag{17}$$

Assuming Gaussian kernels for simplicity, each observation is associated with a different Gaussian contributing to the overall wave function:

$$\Psi(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} \psi_i(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} \frac{e^{-\frac{(\mathbf{x}-\mathbf{x}_i)^2}{2\sigma_i^2}}}{\left(\sqrt{2\pi}\sigma_i\right)^d} \tag{18}$$

where d is the dimensionality of the sample. The potential will be given by

$$V(\mathbf{x}) = E + \frac{\sum_i \frac{\sigma_i^2}{2} \nabla^2 \psi_i}{\sum_i \psi_i} = E - \frac{d}{2} + \left\langle \frac{(\mathbf{x} - \mathbf{x}_i)^2}{2\sigma_i^2} \right\rangle_{\Psi} \tag{19}$$

As a result of having a length scale dependent on nearest neighbours, the shape of the wave function therefore relies on the local density, thus showing narrow and high peaks in areas of high density, whilst smooth and flat shapes are associated with low density regions. An additional advantage of this approach based on a variable σ stems from the fact that outliers can be easily detected since an outlier will have the average distance of its nearest neighbours considerably larger than the rest of the observations, and hence, the corresponding wave function will be flat.

Given the length scale of the Gaussian can be considered as the area of influence of each observation, the same interpretation can be extended to the potential, i.e., regions with high density will create deep potential wells with a steep decay ("volcano" shape), and this property can be used, in turn, to discriminate clusters depending on its density.

This model based on local length scales can be generalised to kernels that are not hyper-spherical by analysing how the nearest neighbours are distributed. Therefore, the resulting wave functions can be a more accurate representation of the probability density function, thus being able to model complicated shapes in the data distribution. To this end, the length scale can be estimated by means of a covariance matrix based on the local manifold information [55], so that a local covariance matrix, Σ_i is computed using the KNNs of each observation:

$$\Sigma_i = \frac{1}{N_k - 1} \sum_{j \in knn}^{N_k} (\mathbf{x_j} - \mathbf{x}_i)^T (\mathbf{x_j} - \mathbf{x}_i) \tag{20}$$

As each observation has a kernel with the form of a multivariate normal distribution, the following wave function is obtained:

$$\Psi(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} \psi_i(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{\sqrt{|2\pi\Sigma_i|}} e^{-\frac{1}{2}(\mathbf{x}-\mathbf{x}_i)^T \Sigma_i (\mathbf{x}-\mathbf{x}_i)} \tag{21}$$

2.3.3 The probabilistic approach

In particular, clusters are no longer defined by the groups of points found after a SGD; those groups are now used to define component elements (subfunctions) that add to make the overall wave function. Assuming that the joint probability of observing the cluster k in the position $\mathbf{x}$ corresponds with the sum of Gaussian functions associated with the observations grouped in the cluster (subfunction) k:

$$\Psi(\mathbf{x}) = \sum_{k=1}^{K} \frac{\sum_{i\in k}^{\#k} \psi_i(\mathbf{x})}{n} = \sum_{k=1}^{K} P(k,\mathbf{x}) = P(\mathbf{x}) \tag{22}$$

where n is the sample size, K the total number of clusters, and $\#k$ the number of observations in cluster k.

The probability of k can be obtained by marginalizing the joint probability over $\mathbb{R}$:

$$\begin{aligned} P(k) &= \int_{\mathbb{R}} P(k,\mathbf{x}) d\mathbf{x} = \int_{\mathbb{R}} \frac{\sum_{i\in k}^{\#k} \psi_i(\mathbf{x})}{n} d\mathbf{x} \\ &= \sum_{i\in k}^{\#k} \frac{\int_{\mathbb{R}} \psi_i(\mathbf{x})\, d\mathbf{x}}{n} = \sum_{i\in k}^{\#k} \frac{1}{n} = \frac{\#k}{n} \end{aligned}$$

Once the joint probability is defined, the required conditional probabilities follow by application of Bayes'rule:

$$P(k|\mathbf{x}) = \frac{P(k,\mathbf{x})}{P(\mathbf{x})} = \frac{\sum_{i\in k}^{\#k} \psi_i(\mathbf{x})}{\sum_{k=1}^{K} \sum_{i\in k}^{\#k} \psi_i(\mathbf{x})} \tag{23}$$

$$P(\mathbf{x}|k) = \frac{P(k,\mathbf{x})}{P(k)} = \frac{\sum_{i\in k}^{\#k} \psi_i(\mathbf{x})}{\frac{\#k}{n}} \tag{24}$$

Cluster allocation follows from the most likely value of k:

$$k_w = \arg\max_k P(k|\mathbf{x}) \quad / \quad \mathbf{x} \in \text{ cluster } k_w \tag{25}$$

This is a significant improvement over the original method for cluster allocation because any region of the input space can be allocated to a cluster without the need to apply SGD over the potential [7]. The probabilistic cluster allocation draws a probability map to define the boundaries between clusters.

2.3.4 Unsupervised performance assessment

It now remains to find an objective function to determine, or at least provide an indication for an appropriate value for the length scale. A maximum likelihood approach applied to the probability of cluster allocation is proposed:

$$\mathrm{LL}(K|\mathbf{X}) = log\left(\prod_{i}^{n} P(k_w|\mathbf{x}_i)\right) = \sum_{i}^{n} log\left(P(k_w|\mathbf{x}_i)\right) \tag{26}$$

This is normalised to the range $[0, 1]$ as follows:

$$\mathrm{ALL}(K|\mathbf{X}) = \frac{-\sum_{i}^{n} log\left(P(k_w|\mathbf{x}_i)\right)}{N} \tag{27}$$

The intuition for this approach is that the lower the ALL, the better the model in terms of the probability assigned to each observation. The crucial aspect is to determine whether ALL is correlated with an accepted score for supervised classification, such as the Jaccard score. In [7], it is shown that they are correlated, hence ALL can be an unsupervised metric for an indirect measure of the clustering performance without the need of external (supervised) labels.

In addition to the size of the neighbourhood given by %KNN, an improvement in the use of ALL involves including a hyperparameter, E_{th} to control how to merge close clusters with small potential differences between their local minima. Due to the fact that ALL is sensitive to the different hierarchical solutions provided by each %KNN, E_{th} sets a threshold to merge two clusters if the maximum potential difference between their centroids is lower than E_{th}. This avoids producing sub-clusters around the same minimum when the potential shape is very flat.

Therefore, as the value of E_{th} increases, clusters are merged starting from those clusters with the lowest potential differences, and ending when all clusters are merged, i.e., E_{th} allows the generation of a quantum hierarchical clustering without modifying the length scale %KNN. The ALL score decreases when the clusters are merged, reaching zero for the trivial case of a unique cluster. Nevertheless, ALL may also have small values, even zero, when the density functions associated to each cluster are perfectly separated.

In summary PQC addresses open questions for QC, which is an objective framework to optimise the local estimation of the length scale around each data point. This provides a more reliable cluster allocation which better reflects the data structure as measured by the Jaccard score. In particular, this framework detects overlapping clusters thus handling heteroscedacity, a common situation in real-world data. The proposed framework is robust to outliers and allows the use of ALL supported by E_{th} and %KNN as an indirect measure of clustering performance, crucial in assessing the correct number of clusters in a given data set.

2.4 K-means comparison with Se-Co framework

For comparison purposes, K-means has also been used in the cMDS embedding. The SeCo framework [30, 9, 5] has been applied to find out the most stable number of K-means clusters; this framework basically repeats K-means multiple times with different centroid initializations for a range of K clusters, and then analyses the separation (intra-cluster centroid distance) and the concordance (using Cramer's V statistic) of the cluster solutions for the same K; these pairs of separation-concordance measures are represented in a graph, where one can observe which K provides the most concordant solutions (as a measure of stability) and offers the highest possible separation. Those solutions with highest concordance indicate the K values where K-means finds similar clusters independently of the centroid initialization.

3 Data description and feature selection

Data was acquired from the Information Management and Preservation Lab, at the Department of Software Technology and Interactive Systems, Vienna University of Technology. This laboratory has extensively used the MSD in MIR[2]. There are many benchmark datasets based on different MIR features. The list of features eventually tested in this work are listed in table 1; they were chosen according to the following criteria:

1. The higher accuracy of the discriminative model the better. The Fisher manifold requires the discriminative model to be able to classify with a minimum predictive power; experimentally it is shown that a $\approx 65\%$ accuracy is good enough. The accuracy metric is used because the Fisher metric only needs the classifier to match as many cases as possible.
2. The lower number of features the better; it affects the computation load of the Fisher manifold, but it is not critical.
3. The higher number of labels (genres) the better; the dimensionality of the Fisher manifold will be increased with the number of different labels. However, the model performance will decrease as the number of labels to predict increase.
4. The lower linearity of the feature set the higher dimensionality of the Fisher manifold.
5. The noise in the data tends to smooth the decision borders of the classifier, drops its accuracy, and implicitly reduces the dimensionality of the Fisher manifold.

A feature selection process was carried out making use of a MLP that was trained on the data set; the goal was to assess the performance of the MLP in classifying the different musical genres depending on the features used for training, thus selecting a data set that yields an appropriate trade-off between

[2] `http://www.ifs.tuwien.ac.at/mir/msd/`

dimensionality and performance. Logistic Regression (LR) was also applied for comparison purposes.

Table 1 shows a summary of the MLP and LR accuracy on the test set (20% sample size) for different feature sets related to MIR domain. All datasets were standardized (z-score) as pre-processing. Originally, the data contained 13 different genres, but they were reduced to only three (keeping the same samples per genre) because the MLP performance worsened considerably with more labels. MLP and LR showed similar performances thus suggesting the linearity of the data set, probably due to the high noise levels. According to the MLP accuracy for three genres, where there are almost 5,000 observations with roughly 1,600 songs per genre, it can be concluded that the best trade-off between MLP accuracy and dimensionality was obtained with ***Low-level features***, made up of 16 features and achieving an accuracy of 72.5%. This performance is crucial for next steps since the Fisher metric is based on the information contained in the classifier model.

Table 1 – Multilayer Perceptron and Logistic Regression: Performance on different feature sets.

Features set	**Dim**	**MLP acc. (%)**		**LR acc. (%)**
		13 genres	**3 genres**	**3 genres**
Rhythm histogram	60	28.4	60.0	60.8
Statistical Spectrum Descriptors	168	41.1	73.7	77.3
Area moments	20	20.5	54.4	55.0
MFCC	26	34.6	67.3	69.6
Low-level features	***16***	***31.8***	***72.5***	***70.3***
Low-level features Derivatives	96	36.7	71.3	76.5
LPC	20	29.9	66.2	64.6
Moment Methods	20	27.0	64.3	64.6

A more detailed MLP performance of the selected benchmark, *Low-level features*, is depicted in figure 1, where the results (accuracy, sensitivity, specificity, PPV and NPV) are presented following the format of the table 2. In the figure, the genres correspond with the following numbers: Rap is 1, Pop-rock is 2 and Jazz is 3.

Table 2 – Template for showing the MLP results.

	Target 1	*Target 2*	
Predicted 1	**True Positive**	**False Positive**	PPV
Predicted 2	**False Negative**	**True Negative**	NPV
	Sensitivity	Specificity	Accuracy

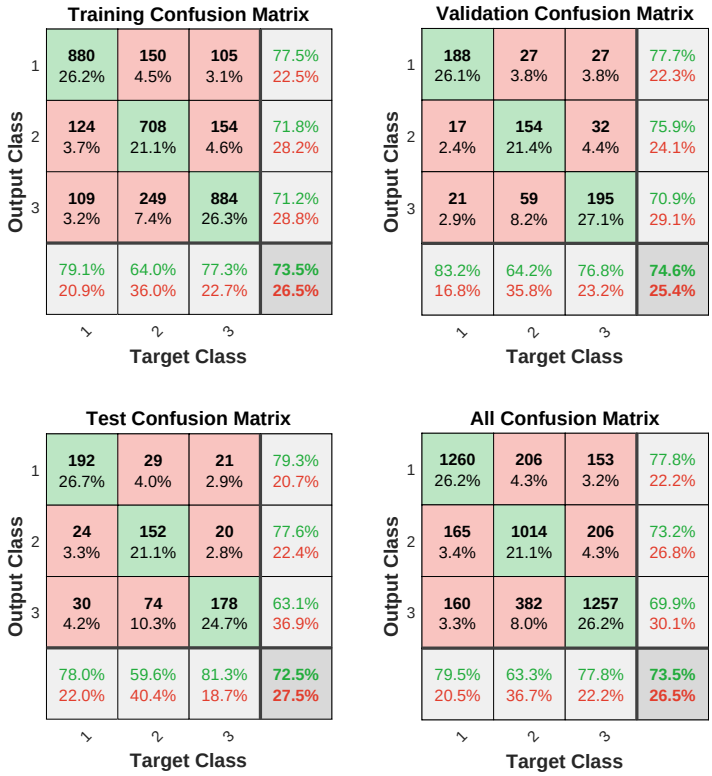

Fig. 1 – Music MLP performance for genre labels, where Rap is 1, Pop-rock is 2 and Jazz is 3.

The description of the spectral low-level features is shown in table 3, where there are two main different types: features based on the standard deviation ($[X1, X8]$) and those based on average features ($[X9, X16]$). Bear in mind that these means and standard deviations are the features themselves, as they were defined in this benchmark. Additional information of this benchmark can be found in [34, 33].

Table 3 – Music spectral low-level features

Feature	**Names of Low-Level features**
X1	Spectral Centroid Std
X2	Spectral Rolloff Point Std
X3	Spectral Flux Std
X4	Compactness Std
X5	Spectral Variability Std
X6	Root Mean Square Std
X7	Fraction of Low Energy Windows Std
X8	Zero Crossings Std
X9	Spectral Centroid Mean
X10	Spectral Rolloff Point Mean
X11	Spectral Flux Mean
X12	Compactness Mean
X13	Spectral Variability Mean
X14	Root Mean Square Mean
X15	Fraction of Low Energy Windows Mean
X16	Zero Crossings Mean

4 Pipeline

This section summarizes the pipeline of the methods described in section 2 in the following steps:

0. Select one of the benchmark features sets based on the MSD, handling the best trade-off between classifier-accuracy, features-number and the amount of genre labels. This dataset will be used for the rest of the pipeline.
1. Choice a random sample of the benchmark dataset, in our experiments a sample size of 5000 observations was used, with well-balanced genre labels.
2. The data is standardized with z-score as pre-processing for the MLP.
3. Compute the Fisher metric with the discriminative model (MLP) and obtain the local pairwise distances.
4. Compute the global pairwise distance with Floyd-Warshall algorithm (APSP), the outcome is the Fisher manifold defined by the adjacency matrix of pairwise distances.
5. Apply the spectral clustering to obtain communities from the similarity matrix, the outcome is a set of community labels.
6. Apply a Euclidean embedding of the Fisher manifold with cMDS and analyse the manifold density distribution.
7. Choose a representative number of main eigenvectors based on its eigenvalues, as estimation of the variance explained in principal component analysis (PCA).
8. Once in the cMDS space, it is important to mention that any kind of preprocessing that modify the relation between eigenvectors should be avoided,

like z-scoring, as the Euclidean distance between points should be preserved.
9. Apply the PQC in the embedded Fisher manifold and obtain the cluster labels.
10. Apply K-means and check cluster concordance in the embedded Fisher manifold just to compare with PQC labels.
11. Compare label results between both clustering methods.
12. Analyse the clusters patterns and how they are distributed in the embedded Fisher manifold.
13. Show the location of some famous songs in the manifold.

5 Results

This section analyses the Fisher manifold structure under the Euclidean embedding once the MLP is trained and the Fisher pairwise distances computed.

5.1 Manifold structure with cMDS

Figure 2 shows the embedding obtained with Sammon mapping; colours are associated with the original genre labelling of the songs. The Sammon mapping tends not to preserve the global distances, and as a result of this, some outliers with distorted distances are observed. When compared to the embedding obtained by cMDS, Sammon mapping appears to be less reliable to represent density distributions, and hence, we focus on cMDS embedding as the main purpose of the work is to find clusters directly on the manifold embedding.

Figure 3 shows the eigenvalues associated with the eigenvectors of the cMDS embedding, their relative accumulated sum can be interpreted as the relative variance retained by the embedding as a function of the number of dimensions used. In this case, figure 3 suggests that the Fisher manifold is bi-dimensional, i.e., the MLP only needs two dimensions to discriminate the genres. This low-dimensionality is partly due to the noisy data, that makes linear boundaries equally efficient as a non-linear MLP as previously shown in table 1.

The two main eigenvectors of the FIN cMDS embedding are represented in figure 4 (left), where songs are coloured by genres. The high noise of the data can be appreciated, with all three genres being mixed. In contrast, MLP predictions, shown in figure 4 (right), depict simple boundaries separating the genres in approximately equal areas of the Fisher manifold; this is why LR achieves a similar performance. It should be emphasized that the cMDS embedding for a manifold based on Euclidean pairwise distances would need at least four or five eigenvalues to retain more than 80% of the variance.

Newman's algorithm [38] was used for community detection. Figure 5 shows five regions that are apparently arbitrary and do not seem to be related to the density peak distributions. In fact, figure 6 shows a two-dimensional histogram

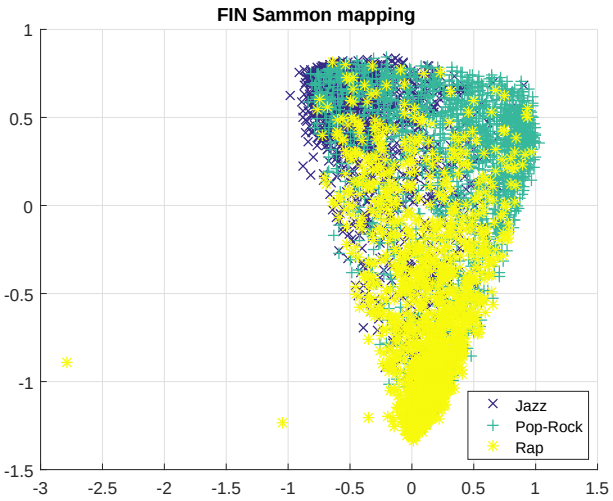

Fig. 2 – Sammon mapping. The original genre labelling is coded by colours

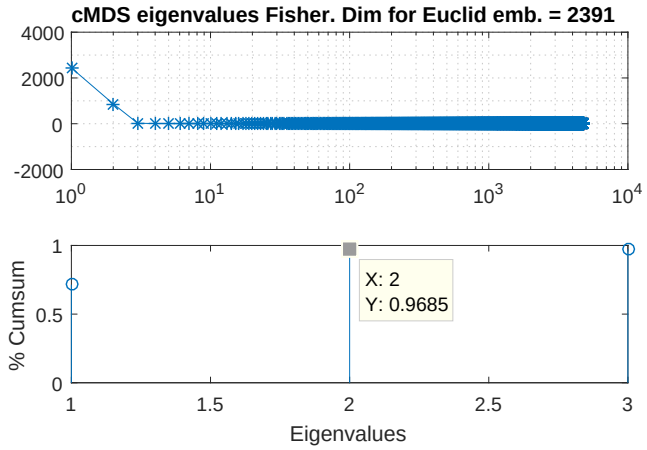

Fig. 3 – Music FIN cMDS eigenvalues (top). Their relative accumulated sum of the top three eigenvalues (bottom)

with density peaks centred at the edges of the Fisher manifold where the most pure genre concentrations appear. This effect can be observed in the plot legend of figure 5, that shows the ratio of maximum genre membership per community. Communities 1, 2 and 3 have ratios of 0.80, 0.79, 0.84 respectively, that contrast with communities 4 and 5 (in the middle of the manifold), that have ratios of 0.47 and 0.42, respectively.

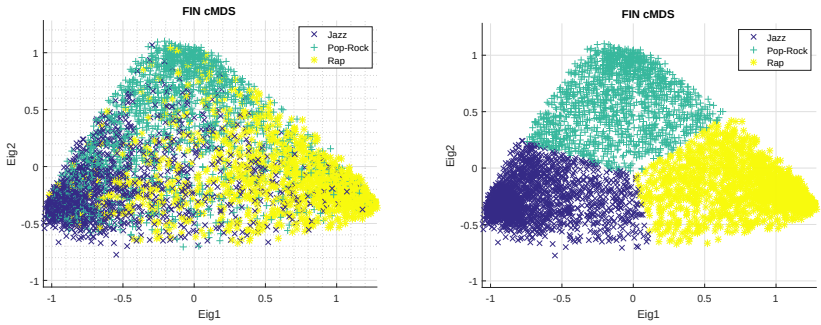

Fig. 4 – (Left) Two main eigenvectors of cMDS embedding. Songs are coloured by genres. (Right) Two main eigenvectors of cMDS embedding. Songs are coloured according to the predictions provided by MLP

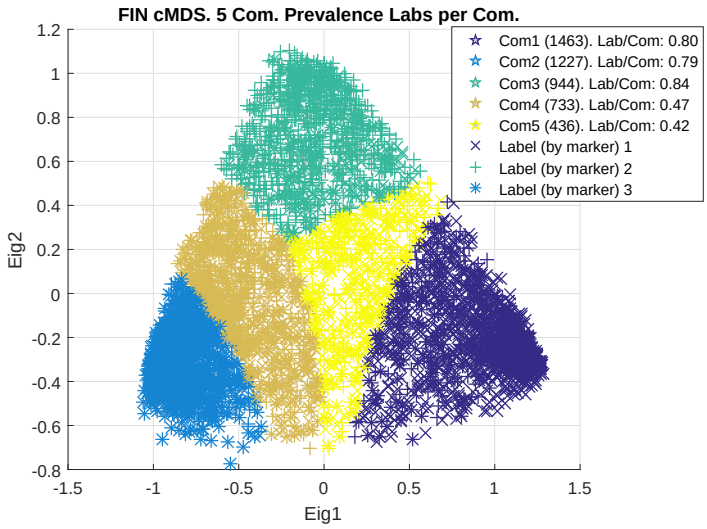

Fig. 5 – Newman's community detection for FIN cMDS

5.2 Cluster finding with PQC in cMDS embedding

PQC can be a suitable choice to find clusters in the cMDS embedding. As shown in [7], PQC hyperparameters can be optimized minimizing the Average-negative-Log-Likelihood (ALL) of cluster membership. In particular, the hyperparameters to be optimized are the length scale σ and the threshold E_{th} to merge two clusters. Figure 7 (left) shows that $\sigma = 15\% knn$ and $log(E_{th}) \in [-3, -1.5]$ are adequate choices for this data set. The $\max_K P(X|K)$, depicted in figure 7 (right), represents the maximum probability of belonging to a cluster; there are some regions where the cluster membership is clear, but there are

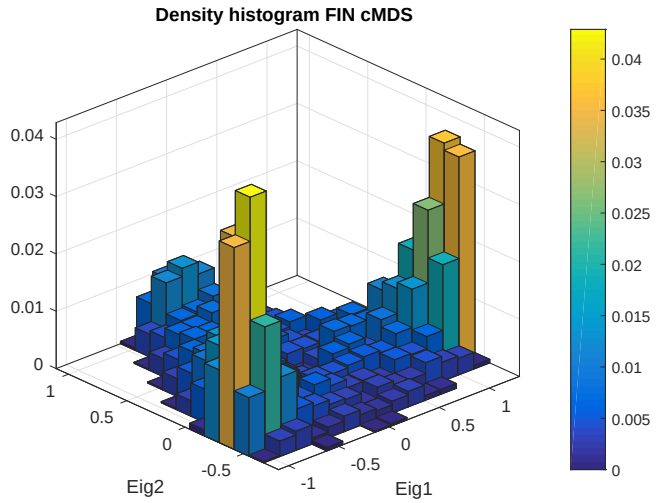

Fig. 6 – Bidimensional histogram of FIN cMDS

also some areas without a dominant cluster, that correspond with scenarios of genre mixing. With the selection of parameters provided by figure 7 (left), figure 8 is obtained, that shows the genre prevalence per cluster in the case of PQC clusters. There are six clusters, three of them belonging to high-density regions, namely, 0.82, 0.85 and 0.91 for Jazz, Pop-Rock and Rap, respectively. The other three clusters lie in intermediate regions, with lower prevalences $\in [0.47, 0.60]$.

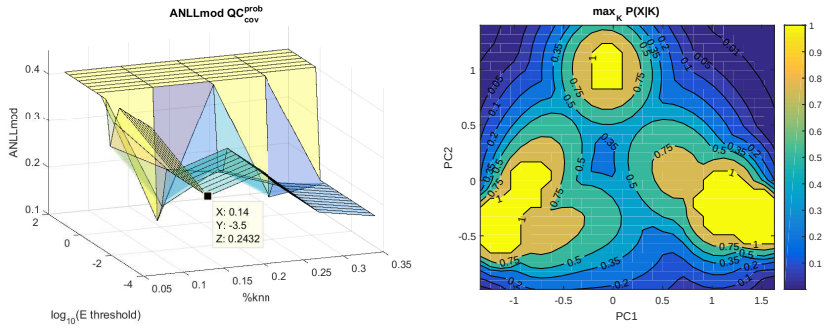

Fig. 7 – (Left) Analysis of the effect of PQC hyperparameters on ALL, an objective index to evaluate the performance of the cluster. (Right) Maximum probability of belonging to a cluster, represented in a bidimensional map defined by the two main principal components

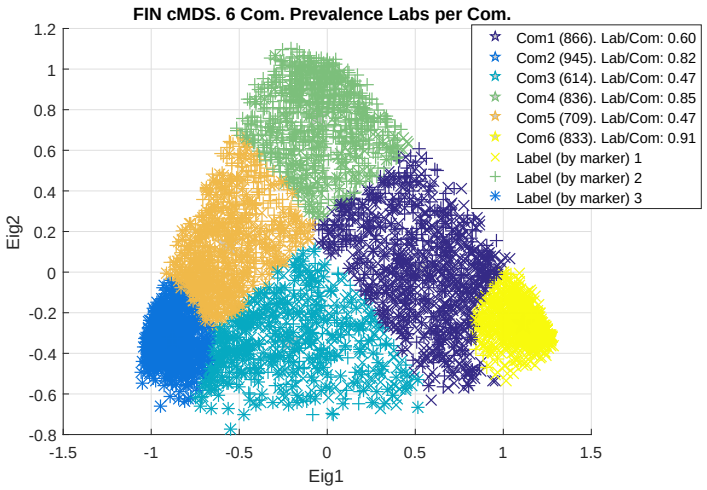

Fig. 8 – Clusters obtained by the Probabilistic Quantum Clustering

5.3 K-Means in cMDS embedding

K-means clustering also has been used to benchmark PQC solutions. Since K-means depends on the centroid initialization and the number of clusters (K) should be provided beforehand, we have used the Se-Co framework to detect the most stable value of K that at the same time maximizes the separation.

The SeCo framework is depited in figure 9; each group of points indicated in the plot legend corresponds with the best K-means solutions for the same K; those groups with high variability in concordance mean that this K is unstable, and each solution is affected by the random centroid initialization. Those solutions located in the top-right corner tend to be the most appropriate ones, giving priority to the concordance with respect to the separation. In particular, the best group of solutions corresponds with $K = 9$; the solution with the highest separation within this group is shown in figure 10. As expected, the communities are more segmented than those obtained by the PQC solutions (fig. 8) because of the higher cluster number; this is not a problem itself, because there is not a correct number of clusters and the purpose of this analysis is to find similarities in the Fisher manifold beyond the rough genre label classification.

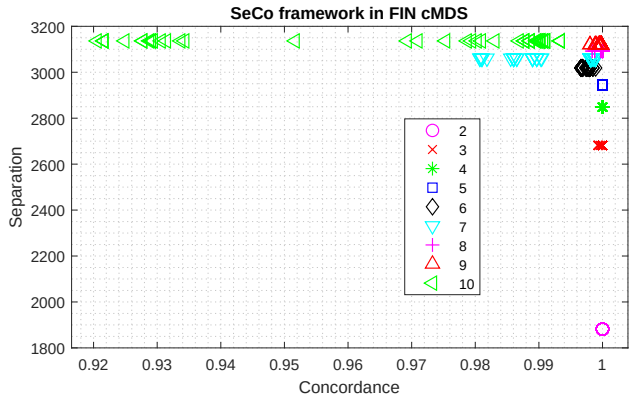

Fig. 9 – K-Means solutions of cMDS embedding, represented with the Se-Co framework for selecting the cluster number of K-means with highest concordance and separation. The group of solutions with $K = 9$ have high concordance and one of the highest separations.

Although K-means solutions could be acceptable for segmentation in the Fisher manifold embedding, they have the inherent drawback to form clusters with spherical distributions and similar size, and this effect may split some non-spherical regions with a homogeneous density. Therefore, in terms of finding homogeneous regions with similar density, PQC is more adequate. As additional information, appendix B provides the Silhouette figure for all cluster solutions, included the genre labels and the MLP predictions.

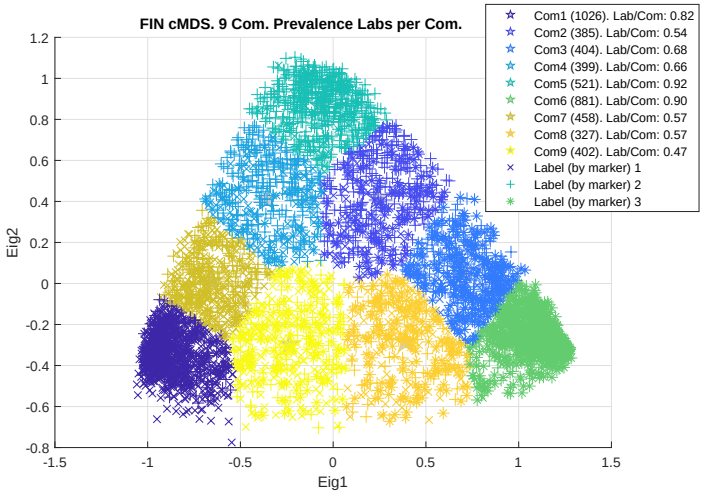

Fig. 10 – Clusters obtained by the best solution of K-means with $K = 9$.

In summary, the Fisher manifold tends to create similarity regions with high density and dissimilar regions with low density, but it is not always the case, especially if the discriminative model has a poor performance. For that reason any clustering method able to detect density variations would be suitable for this task, but it is a good practice to contrast the results with another clustering method able to segment regions with low density variations.

5.4 Cluster profiles

When dealing with a practical problem, like the one faced in this paper, it is especially relevant to analyse the Fisher manifold in terms of the cluster profiles:

$$\text{profile}_{com_k}(x_n) = \frac{\mu_{com_k}(x_n) - \mu(x_n)}{\sigma(x_n)} \tag{28}$$

where μ_{com_k} is the mean value of patterns assigned to the k-th cluster, $\mu(x_n)$ the mean value of the data set and $\sigma(x_n)$ the corresponding standard deviation. The most important attributes of each genre can be found in high-density clusters that can be compared to other clusters that are made up of fusion of different genres. It is also possible to illustrate the main attributes of the external labels without taking into account their distribution in the Fisher manifold. Tables 4 and 5 present the results for the features based on standard deviation and mean value, respectively. As the profiles are standardized, only those absolute values greater than one can be considered different enough from the mean data (significance in terms of one standard deviation).

The position notation is PR for Pop-Rock, J for Jazz and R for Rap. Those positions are referred to the cMDS embedding where Jazz is at lower left, Rap at lower right and Pop-Rock at the upper region of the plot. Analysing the features, Jazz and Rap hold more characteristic features, with greater absolute values. However, Pop-Rock has many feature attributes closer to zero; it makes sense because Pop-Rock includes many different music styles from heavy-metal to commercial-pop, producing average values that might be near zero. The intermediate clusters also have lower absolute values than the pure ones. With respect to the different kinds of features, the ones based on standard deviations present greater absolute values than the ones based on mean values.

Finally, it is important to remark that in this case, but also happens in general, the profiles based on external labels tend to be close to zero, due to the fact that external labels are not clustered by similarities or feature distances, producing averaged profiles close to the whole data average.

5.5 Analysis of popular songs

This section studies a sample of widely known songs from the database and maps them into the embedded Fisher manifold to find out where they are

Table 4 – Cluster profiling using features based on the standard deviation

Com.	**Position**	**X1**	**X2**	**X3**	**X4**	**X5**	**X6**	**X7**	**X8**
1	PR & R	0.40	0.45	0.42	-0.02	0.46	0.44	-0.15	0.45
2	J	-0.89	-1.02	-1.00	0.21	-1.00	-1.00	0.32	-0.98
3	J & R	-0.06	-0.07	-0.40	0.37	-0.25	-0.23	0.59	0.00
4	PR	-0.22	-0.09	0.22	-0.46	-0.05	-0.08	-0.50	-0.25
5	J & PR	-0.45	-0.41	-0.62	-0.22	-0.63	-0.61	-0.13	-0.44
6	R	1.26	1.19	1.29	0.15	1.44	1.44	-0.03	1.26
External labels		**X1**	**X2**	**X3**	**X4**	**X5**	**X6**	**X7**	**X8**
Rap		0.73	0.69	0.71	0.13	0.83	0.81	-0.03	0.74
Pop-Rock		-0.20	-0.11	-0.06	-0.24	-0.18	-0.18	-0.20	-0.20
Jazz		-0.52	-0.56	-0.64	0.11	-0.64	-0.61	0.23	-0.53

Table 5 – Cluster profiling using features based on the mean value

C	**Pos.**	**X9**	**X10**	**X11**	**X12**	**X13**	**X14**	**X15**	**X16**
1	PR & Rap	0.11	0.12	0.40	-0.35	0.39	0.37	-0.05	0.19
2	J	-0.74	-0.81	-1.00	0.68	-1.04	-1.05	-0.22	-0.95
3	J & R	-0.51	-0.54	-0.57	0.11	-0.53	-0.57	0.18	-0.55
4	PR	0.62	0.79	0.75	-0.34	0.80	0.90	-0.10	0.81
5	J & PR	0.01	-0.02	-0.56	0.22	-0.36	-0.33	-0.16	0.02
6	R	0.47	0.41	0.86	-0.34	0.67	0.61	0.42	0.45
Ext. labels		**X9**	**X10**	**X11**	**X12**	**X13**	**X14**	**X15**	**X16**
Rap		0.23	0.19	0.50	-0.30	0.42	0.37	0.21	0.22
Pop-Rock		0.21	0.31	0.20	-0.13	0.28	0.34	-0.11	0.31
Jazz		-0.43	-0.49	-0.69	0.43	-0.69	-0.69	-0.10	-0.53

located in the manifold, and thus, the corresponding genre. Table 6 lists the songs by ID in order to identify them in figure 11. Table 6 also provides information about the external genre and the one predicted by the MLP. On top of the most popular themes, two more songs were included due to their closeness and rare positioning in the upper area of the manifold. Their IDs are 19 (Pop-Rock) and 27 (Jazz); song #19 is authentic heavy metal although it was surprisingly labelled as Pop-Rock; with respect to song #27, its style is very experimental and it sounds like industrial noise, what explains why it is located far away from the main Jazz region (lower-left area).

Table 6 – Sample of widely known songs from the database mapped into the Fisher manifold.

Id	**Lab/MLP**	**Song Artist**	**Song title**
1	J - J	Count Basie	Segue In C
2	J - J	Duke Ellington	Black And Tan Fantasy
3	R - R	Eminem	We Made You
4	J - J	Frank Sinatra	I Should Care
5	J -R	George Benson	Stairway To Love
6	J - J	Glenn Miller	Happy In Love
7	R - R	Ice Cube	A Bird In The Hand
8	J - R	Jamie Cullum	Love Aint Gonna Let You Down
9	R - R	Jay-Z	Threat
10	J - J	Juliet Roberts	Carriacou Sunrise
11	R - R	Kanye West	Flashing Lights
12	PR - PR	Korn	Politics (Claude Le Gache Edit)
13	PR - J	Led Zeppelin	Since Ive Been Loving You
14	PR - J	Little Richard	Long Tall Sally (Take 1)
15	J - J	Louis Armstrong	I Can't Give You Anything But Love
16	J - J	Louis Armstrong	Alexanders Rag Time Band
17	PR - J	Martha Wainwright	These Flowers
18	J - J	Miles Davis	Dear Old Stockholm
19	PR - PR	Naer Mataron	The Life And Death Of Europa
20	J - J	Nat King Cole	I Get A Kick Out Of You
21	PR - J	Neil Diamond	Girl Youll Be A Woman Soon
22	PR - PR	Neil Young	Revolution Blues
23	PR - PR	Pet Shop Boys	Rent (2001 Digital Remaster)
24	PR - J	Robbie Williams	Morning Sun Reprise
25	J - J	Slavic Soul Party!	Juan Colorado
26	R - PR	Snoop Dogg	Gangsta Luv
27	J - PR	The Flying Luttenbachers	Clank
28	PR - PR	The Jimi Hendrix Exp.	Hey Joe
29	PR - R	The Rolling Stones	Cherry Oh Baby
30	PR - PR	The Velvet Underground	Rock And Roll (LP Version)
31	R - PR	Vanilla Ice	It's A Party
32	R - R	Will.I.Am	Tai Arrive

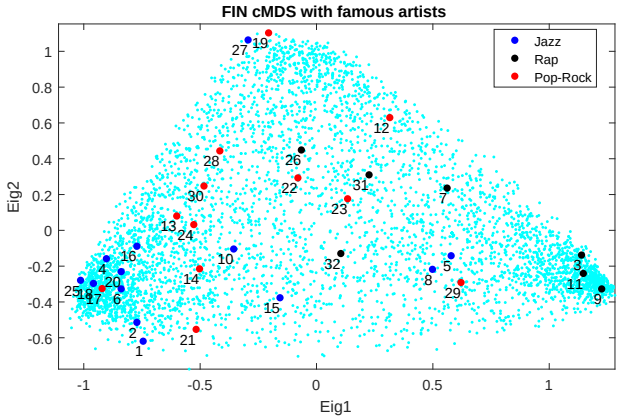

Fig. 11 – Location of famous songs in the cMDS embedding of the Fisher manifold

6 Challenges and limitations of the current implementation

6.1 Scalability

One of the main limitations of the Fisher manifold is the scalability with the sample size, mainly because the Riemannian manifold relies on estimating pairwise distances and this produces a bottleneck in the runtime. This work [6] has partly addressed the problem for 15,000 samples, still far away from big data environments. For instance, the current implementation could not have handled all the MSD at once. This point remains for future work. In any case, a good sampling is usually enough to build a Fisher manifold able to get data insights.

6.2 New data allocation

Another issue is how to allocate new samples in the space created by the embedded Fisher manifold. This space contains a projection of the network created by pairwise distances under the Fisher metric; we propose two options to allocate new data, the first one is more exact but requires more computational load, and the second option is faster but less exact.

Given M new samples in a manifold created by N samples, being $M < N$, the first option consists in computing the new $M \cdot N$ and the $M(M-1)/2$ local pairwise distances, and then apply a single source shortest path algorithm (SSSP); for instance, the Dijkstra's algorithm [12] computes faster than APSP algorithms the shortest path between a single source (one sample of M) with reference to the rest of the data ($N + (M-1)$). Once the global distance is computed, the Fisher manifold is embedded again with cMDS.

The problem is that recomputing the cMDS with new data may change the absolute positions of the previous data (the N samples) because the structure of this space is based on relative positions. Therefore, the clustering should be computed again.

The second option is focused on allocating the new (M) points just computing a similarity network based on Euclidean distances with the features of the input space. For instance a Gaussian kernel that transforms distances into similarities:

$$A_{ij} = \exp\left(-\frac{dist\left(\mathbf{x}_i, \mathbf{x}_j\right)^2}{\sigma_G^2}\right) \tag{29}$$

where A_{ij} is the element of the adjacency matrix, being $i \in M$ and $j \in N$, and $\mathbf{x}$ is a vector of the input space, and σ_G is a scale factor that can be heuristically estimated. Using this similarity matrix, the new data M can be allocated directly in the cMDS space using a weighting average of the coordinates of N:

$$\tilde{\mathbf{y}}_i = \frac{1}{\sum_j^N A_{ij}} \sum_j^N A_{ij}\mathbf{y}_j \tag{30}$$

where $\mathbf{y}$ are the coordinates in the cMDS space. This approximation allows a direct allocation of new samples without the need to compute pairwise distances on the Fisher manifold (nor re-compute the clustering).

6.3 Manifold interpretation

As the manifold is actually the structure created by pairwise distances, its interpretation is related to analyze similar data (grouped in some clusters with higher density), and dissimilar data (distributed in sparse regions).

The reason behind these distributions lies in how the Fisher metric modulates the Euclidean pairwise distance of the input space depending on the information contained in the discriminative model, in that region of the input space. At the same time, the discriminative model depends on the class labels used to train it.

The Fisher metric tends to shorten Euclidean distances in those regions with low rate-of-change of the discriminative model probabilities, $p(c|\mathbf{x})$; in contrast, it lengthens Euclidean distances in regions where the discriminative model probabilities have a high rate-of-change.

Therefore, the structure is partly defined by the Euclidean distances between the features of the input space, and partly by the information contained in the discriminative model (whose performance depends on the ability to predict the labels using the information present in the features). Changing the labels will modify the discriminative model, and in consequence the structure of Fisher manifold.

7 Conclusion

We have described a principled approach to explicitly map the data manifold induced by genre-specific content in three categories, Pop Rock, Rap and Jazz. This comprises a case study in a challenging application, namely the manifold structure of the genre content in popular music, which may be used for Musical Information Retrieval.

The approach is generic, since all probabilistic classifiers define such a metric. By making explicit the similarity measure that applies through the space of input data, the paper shows how important knowledge about the data, which is implicit in the classifier, can be pulled out and visualised by end users. This enables the validation of the similarity structure according to the plausibility of local neighbourhoods and global clusters induced in the data. The paper also demonstrates the practical feasibility of using classical multidimensional scaling to generate low-dimensional Euclidean embeddings from the Riemannian space induced in high-dimensional data by the Fisher Information matrix. Furthermore, we have shown that Probabilistic Quantum Clustering can map the data density even for complex data structures that form a continuum between high density peaks.

The similarities between songs are based on spectral wave sound features, which purely objective measures without any subjective human-perception whatsoever. The resulting Fisher manifold identifies and quantifies mixtures between genres in songs that are typically labelled by the prevalent genre of the band playing it. In this way, the results in this paper have potential for more accurate labelling of the three genres.

A Derivation of FI matrix as a function of MLP

For obtaining $\mathbf{FI}(\mathbf{x})$ as a function of the MLP output estimators, the soft-max logarithms and their derivatives are needed:

$$p_j = \frac{\exp(a_j)}{\sum_{k=1}^{J}\exp(a_k)} \tag{31}$$

$$\log(p_j) = a_j - \log\left(\sum_{k=1}^{J}\exp(a_k)\right) \tag{32}$$

$$\nabla\log(p_j) = \nabla a_j - \sum_{k=1}^{J} p_k \nabla a_k \tag{33}$$

where $p_j = p(c_j|\mathbf{x})$, $\nabla = \nabla_{\mathbf{x}} = \frac{d}{d\mathbf{x}}$ and $a_j = a_j(\mathbf{x})$ for notation abbreviation. Now, combining eq. 2 and eq. 31 and expanding the product:

$$\begin{aligned}\mathbf{FI}(\mathbf{x}) &= \sum_{j=1}^{J}\left(\nabla a_j - \sum_{k=1}^{J} p_k \nabla a_k\right)\left(\nabla a_j - \sum_{l=1}^{J} p_l \nabla a_l\right)^T p_j \\ &= \sum_{j=1}^{J}\Bigg((\nabla a_j)(\nabla a_j)^T - \sum_{l=1}^{J}(\nabla a_j)(\nabla a_l)^T p_l \\ &\quad - \sum_{k=1}^{J}(\nabla a_k)(\nabla a_j)^T p_k + \sum_{k=1}^{J}\sum_{l=1}^{J}(\nabla a_k)(\nabla a_l)^T p_k p_l\Bigg) p_j\end{aligned} \tag{34}$$

Rearranging terms and considering that any variable t can be expressed as $\sum_i^J t p_i = t\sum_i^J p_i = t$, we get:

$$\begin{aligned}\mathbf{FI}(\mathbf{x}) &= \sum_{j=1}^{J}\Bigg(\sum_{k=1}^{J}\sum_{l=1}^{J}(\nabla a_j)(\nabla a_j)^T p_k p_l \\ &\quad - \sum_{k=1}^{J}\sum_{l=1}^{J}(\nabla a_j)(\nabla a_l)^T p_k p_l - \sum_{k=1}^{J}\sum_{l=1}^{J}(\nabla a_k)(\nabla a_j)^T p_k p_l \\ &\quad + \sum_{k=1}^{J}\sum_{l=1}^{J}(\nabla a_k)(\nabla a_l)^T p_k p_l\Bigg) p_j \\ &= \sum_{j=1}^{J}\Bigg(\sum_{k=1}^{J}\sum_{l=1}^{J}\Big((\nabla a_j)(\nabla a_j)^T - (\nabla a_j)(\nabla a_l)^T \\ &\quad - (\nabla a_k)(\nabla a_j)^T + (\nabla a_k)(\nabla a_l)^T\Big) p_k p_l\Bigg) p_j\end{aligned} \tag{35}$$

After merging the summations, the final expression of the $\mathbf{FI}(\mathbf{x})$ for the MLP is obtained:

$$\mathbf{FI}(\mathbf{x}) = \sum_{j=1}^{J}\sum_{k=1}^{J}\sum_{l=1}^{J}\nabla(a_j - a_k)\,\nabla(a_j - a_l)^T p_j p_k p_l \tag{36}$$

With this eq. 36 (equivalent to eq. 5) the metric is estimated locally, computing differential distances as:

$$d(\mathbf{x}, \mathbf{x} + d\mathbf{x})^2 = d\mathbf{x}^T \cdot \mathbf{FI}(\mathbf{x}) \cdot d\mathbf{x} \tag{37}$$

B Silhouette figures

The Silhouette metric is especially adequate when the clustering method is based on minimizing pairwise distances within the cluster members. However, when the clustering also takes account of density similarity, non-spherical shapes are expected and hence, the Silhouette metric of a PQC probably might be worse than K-means Silhouette even if the PQC cluster better reflects the data profiles. In any case, the Silhouette metric is computed for all cluster solutions based in the cMDs data.

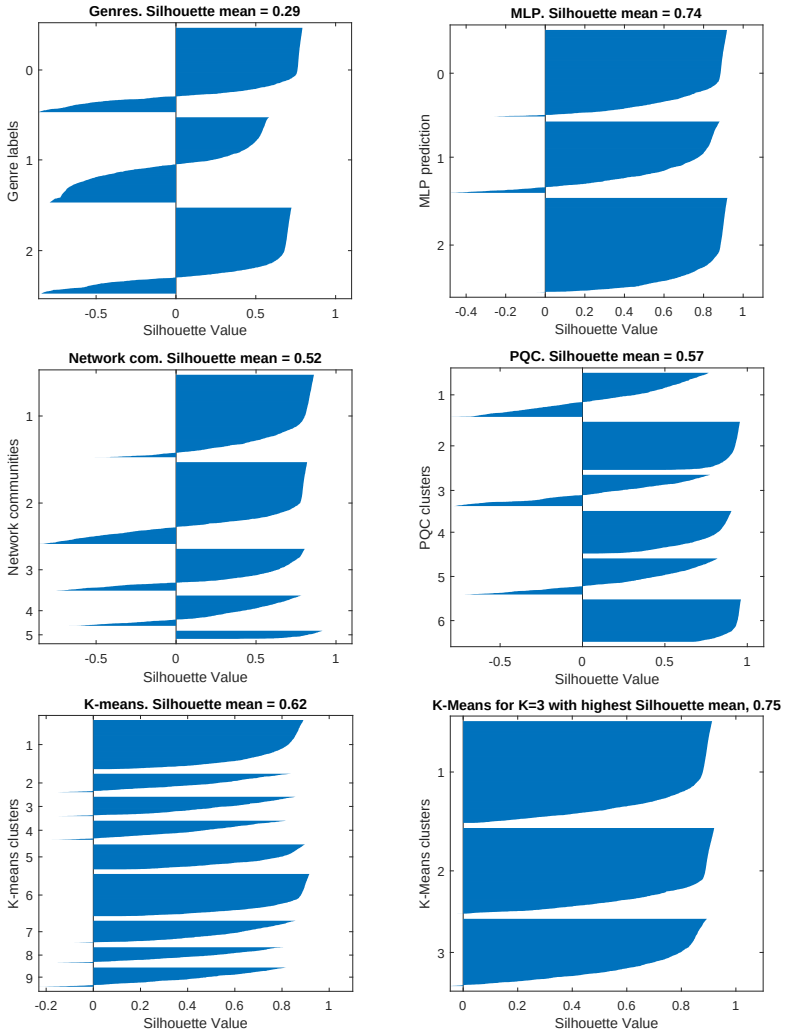

Fig. 12 – Silhouette figures for the different cluster solutions. The genre labels correspond with: Rap (0), Pop-Rock (1), Jazz (2)

References

1. Amari, S.I.: Natural gradient works efficiently in learning. Neural computation **10**(2), 251–276 (1998)
2. Amari, S.i., Wu, S.: Improving support vector machine classifiers by modifying kernel functions. Neural Networks **12**(6), 783–789 (1999)
3. Bogdanov, D., Serra, J., Wack, N., Herrera, P.: From low-level to high-level: Comparative study of music similarity measures. In: 2009 11th IEEE International Symposium on Multimedia, pp. 453–458. IEEE (2009)
4. Carter, K.M., Raich, R., Finn, W.G., III, A.O.H.: Fine: Fisher information nonparametric embedding. IEEE Transactions on Pattern Analysis and Machine Intelligence **31**(11), 2093–2098 (2009). DOI 10.1109/TPAMI.2009.67
5. Casaña-Eslava, R.V., Jarman, I.H., Lisboa, P.J., Martín-Guerrero, J.D.: Quantum clustering in non-spherical data distributions: Finding a suitable number of clusters. Neurocomputing **268**, 127–141 (2017)
6. Casaña-Eslava, R.V., Martín-Guerrero, J.D., Ortega-Martorell, S., Lisboa, P.J., Jarman, I.H.: Scalable implementation of measuring distances in a riemannian manifold based on the fisher information metric. In: 2019 International Joint Conference on Neural Networks (IJCNN), pp. 1–7. IEEE (2019)
7. Casaña-Eslava, R.V., Lisboa, P.J., Ortega-Martorell, S., Jarman, I.H., Martín-Guerrero, J.D.: Probabilistic quantum clustering. Knowledge-Based Systems p. 105567 (2020). DOI https://doi.org/10.1016/j.knosys.2020.105567. URL `http://www.sciencedirect.com/science/article/pii/S0950705120300587`
8. Casey, M.A., Veltkamp, R., Goto, M., Leman, M., Rhodes, C., Slaney, M.: Content-based music information retrieval: Current directions and future challenges. Proceedings of the IEEE **96**(4), 668–696 (2008)
9. Chambers, S.J., Jarman, I.H., Etchells, T.A., Lisboa, P.J.G.: Inference of number of prototypes with a framework approach to k-means clustering. International Journal of Biomedical Engineering and Technology **13**(4), 323–340 (2013)
10. Cortes, C., Vapnik, V.: Support-vector networks. Machine learning **20**(3), 273–297 (1995)
11. Cox, M.A., Cox, T.F.: Multidimensional scaling. In: Handbook of data visualization, pp. 315–347. Springer (2008)
12. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numerische mathematik **1**(1), 269–271 (1959)
13. Floyd, R.W.: Algorithm 97: Shortest path. Commun. ACM **5**(6), 345– (1962). DOI 10.1145/367766.368168. URL `http://doi.acm.org/10.1145/367766.368168`
14. Goto, M., Goto, T.: Musicream: New music playback interface for streaming, sticking, sorting, and recalling musical pieces. In: ISMIR, pp. 404–411 (2005)
15. Gower, J.C.: Some distance properties of latent root and vector methods used in multivariate analysis. Biometrika **53**(3-4), 325–338 (1966)
16. Hamasaki, M., Goto, M.: Songrium: A music browsing assistance service based on visualization of massive open collaboration within music content creation community. In: Proceedings of the 9th International Symposium on open collaboration, pp. 1–10 (2013)
17. Haykin, S.S.: Neural networks and learning machines, third edn. Pearson Education, Upper Saddle River, NJ (2009)
18. Horn, D., Gottlieb, A.: Algorithm for data clustering in pattern recognition problems based on quantum mechanics. Physical review letters **88**(1), 018702 (2001)
19. Horn, D., Gottlieb, A.: The method of quantum clustering. In: Proceedings of Neural Information Processing Systems NIPS 2001, pp. 769–776 (2001)
20. Jaakkola, T.S., Haussler, D., et al.: Exploiting generative models in discriminative classifiers. Advances in neural information processing systems pp. 487–493 (1999)
21. Jones, M.C., Downie, J.S., Ehmann, A.F.: Human similarity judgments: Implications for the design of formal evaluations. In: ISMIR, pp. 539–542 (2007)
22. Kaski, S., Sinkkonen, J.: Metrics that learn relevance. In: Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium, vol. 5, pp. 547–552 vol.5 (2000). DOI 10.1109/IJCNN.2000.861526

23. Kaski, S., Sinkkonen, J., Peltonen, J.: Bankruptcy analysis with self-organizing maps in learning metrics. IEEE Transactions on Neural Networks **12**(4), 936–947 (2001). DOI 10.1109/72.935102
24. Kim, J.H., Tomasik, B., Turnbull, D.: Using artist similarity to propagate semantic information. In: ISMIR, vol. 9, pp. 375–380 (2009)
25. Knees, P., Pampalk, E., Widmer, G.: Artist classification with web-based data. In: ISMIR (2004)
26. Knees, P., Schedl, M., Pohle, T., Widmer, G.: An innovative three-dimensional user interface for exploring music collections enriched. In: Proceedings of the 14th ACM international conference on Multimedia, pp. 17–24 (2006)
27. Kullback, S.: Information theory and statistics. Courier Corporation (1997)
28. Li, Y., Wang, Y., Wang, Y., Jiao, L., Liu, Y.: Quantum clustering using kernel entropy component analysis. Neurocomputing **202**, 36–48 (2016)
29. Lippens, S., Martens, J.P., De Mulder, T.: A comparison of human and automatic musical genre classification. In: 2004 IEEE international conference on acoustics, speech, and signal processing, vol. 4, pp. iv–iv. IEEE (2004)
30. Lisboa, P.J.G., Etchells, T.A., Jarman, I.H., Chambers, S.J.: Finding reproducible cluster partitions for the k-means algorithm. BMC Bioinformatics **14**(Suppl. 1), S8 (2013)
31. Lübbers, D., Jarke, M.: Adaptive Multimodal Exploration of Music Collections. In: Proceedings of the 10th International Society for Music Information Retrieval Conference, pp. 195–200. ISMIR, Kobe, Japan (2009). DOI 10.5281/zenodo.1415518. URL `https://doi.org/10.5281/zenodo.1415518`
32. Mandel, M.I., Pascanu, R., Eck, D., Bengio, Y., Aiello, L.M., Schifanella, R., Menczer, F.: Contextual tag inference. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) **7**(1), 1–18 (2011)
33. McKay, C.: Automatic music classification with jMIR. Citeseer (2010)
34. McKay, C., Fujinaga, I., Depalle, P.: jaudio: A feature extraction library. In: Proceedings of the International Conference on Music Information Retrieval, pp. 600–3 (2005)
35. Miotto, R., Barrington, L., Lanckriet, G.R.: Improving auto-tagging by modeling semantic co-occurrences. In: ISMIR, pp. 297–302 (2010)
36. Nash, J.: C1 isometric imbeddings. Annals of mathematics pp. 383–396 (1954)
37. Nash, J.: The imbedding problem for riemannian manifolds. Annals of mathematics pp. 20–63 (1956)
38. Newman, M.E.: Detecting community structure in networks. The European Physical Journal B-Condensed Matter and Complex Systems **38**(2), 321–330 (2004)
39. Parisi, L., RaviChandran, N., Manaog, M.L.: A novel hybrid algorithm for aiding prediction of prognosis in patients with hepatitis. Neural Computing and Applications pp. 1–14 (2019)
40. Platt, J., et al.: Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Advances in large margin classifiers **10**(3), 61–74 (1999)
41. Rao, C.R.: Information and the accuracy attainable in the estimation of statistical parameters. In: Breakthroughs in statistics, pp. 235–247. Springer (1992)
42. Ruiz, H., Etchells, T.A., Jarman, I.H., Martín, J.D., Lisboa, P.J.: A principled approach to network-based classification and data representation. Neurocomputing **112**, 79–91 (2013)
43. Sammon, J.W.: A nonlinear mapping for data structure analysis. IEEE Transactions on computers **18**(5), 401–409 (1969)
44. Schedl, M., Flexer, A., Urbano, J.: The neglected user in music information retrieval research. Journal of Intelligent Information Systems **41**(3), 523–539 (2013)
45. Schedl, M., Gómez Gutiérrez, E., Urbano, J.: Music information retrieval: Recent developments and applications. Foundations and Trends in Information Retrieval. 2014 Sept 12; 8 (2-3): 127-261. (2014)
46. Schedl, M., Pohle, T., Knees, P., Widmer, G.: Exploring the music similarity space on the web. ACM Transactions on Information Systems (TOIS) **29**(3), 1–24 (2011)
47. Schindler, A., Mayer, R., Rauber, A.: Facilitating comprehensive benchmarking experiments on the million song dataset. In: ISMIR, pp. 469–474 (2012)
48. Schindler, A., Rauber, A.: Capturing the temporal domain in echonest features for improved classification effectiveness. In: International Workshop on Adaptive Multimedia Retrieval, pp. 214–227. Springer (2012)

49. Seyerlehner, K., Schedl, M., Pohle, T., Knees, P.: Using block-level features for genre classification, tag classification and music similarity estimation. Submission to Audio Music Similarity and Retrieval Task of MIREX **2010** (2010)
50. Sordo, M., et al.: Semantic annotation of music collections: A computational approach. Ph.D. thesis, Universitat Pompeu Fabra (2012)
51. Torgerson, W.S.: Multidimensional scaling: I. theory and method. Psychometrika **17**(4), 401–419 (1952)
52. Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. IEEE Transactions on speech and audio processing **10**(5), 293–302 (2002)
53. Urbano, J.: Evaluation in audio music similarity. Ph.D. thesis, Universidad Carlos III de Madrid (2013)
54. Urbano, J., Morato, J., Marrero, M., Martín, D.: Crowdsourcing preference judgments for evaluation of music similarity tasks. In: ACM SIGIR workshop on crowdsourcing for search evaluation, pp. 9–16. ACM New York (2010)
55. Vincent, P., Bengio, Y.: Manifold parzen windows. In: Advances in Neural Information Processing Systems, pp. 849–856 (2003)
56. Warshall, S.: A theorem on boolean matrices. J. ACM **9**(1), 11–12 (1962). DOI 10.1145/321105.321107. URL `http://doi.acm.org/10.1145/321105.321107`
57. Young, G., Householder, A.S.: Discussion of a set of points in terms of their mutual distances. Psychometrika **3**(1), 19–22 (1938)
58. Zelnik-Manor, L., Perona, P.: Self-tuning spectral clustering. In: Advances in neural information processing systems, pp. 1601–1608 (2005)
59. Zhang, Y.C., Séaghdha, D.Ó., Quercia, D., Jambor, T.: Auralist: introducing serendipity into music recommendation. In: Proceedings of the fifth ACM international conference on Web search and data mining, pp. 13–22 (2012)