Few-Sample Feature Selection via Feature Manifold Learning

David Cohen [1] **Tal Shnitzer** [2] **Yuval Kluger** [3 4] **Ronen Talmon** [1]

Abstract

In this paper, we present a new method for few-sample supervised feature selection (FS). Our method first learns the manifold of the feature space of each class using kernels capturing multi-feature associations. Then, based on Riemannian geometry, a composite kernel is computed, extracting the differences between the learned feature associations. Finally, a FS score based on spectral analysis is proposed. Considering multi-feature associations makes our method multivariate by design. This in turn allows for the extraction of the hidden manifold underlying the features and avoids overfitting, facilitating few-sample FS. We showcase the efficacy of our method on illustrative examples and several benchmarks, where our method demonstrates higher accuracy in selecting the informative features compared to competing methods. In addition, we show that our FS leads to improved classification and better generalization when applied to test data.

1. Introduction

Feature selection (FS) plays a vital role in facilitating effective and efficient learning in problems involving high-dimensional data (Bolón-Canedo et al., 2022; Hastie et al., 2009; Duda et al., 2006). By selecting the relevant features, FS methods, in effect, reduce the dimension of the data, which has been shown useful in improving learning, especially in terms of generalization and noise reduction (Remeseiro & Bolon-Canedo, 2019). In contrast to feature extraction (FE), in which the whole feature space is projected into a lower dimension space, FS eliminates irrelevant and redundant features and preserves interpretability (Hira & Gillies, 2015; Alelyani et al., 2018). This characteristic of FS has a significant societal impact since it enhances the explainability of AI models.

In the literature, there are three main approaches for FS: (i) wrapper, (ii) embedded, and (iii) filter (Tang et al., 2014; Alelyani et al., 2018). In the *wrapper approach*, the classifier performance plays an integral role in the feature selection. Concretely, the performance of a specific classifier is maximized by searching for the best subset of features. Since examining all possible subsets is an NP-hard problem, a suboptimal search is often applied. Still, this approach is considered computationally heavy for problems with large feature spaces (Hira & Gillies, 2015; Alelyani et al., 2018).

The *embedded approach* mitigates the wrapper limitations by incorporating the feature selection in the model training, and thus, avoids multiple optimization processes. Consequently, embedded methods are considered computationally feasible and often preserve the advantages of the wrapper approach. However, both wrapper and embedded methods are prone to overfitting because the selection is part of the training (Brown et al., 2012; Bolón-Canedo et al., 2013; Venkatesh & Anuradha, 2019).

In the *filter approach*, each feature is ranked according to particular criteria independent of the model learning. The various ranking techniques aim to identify the features that best discriminate between the different classes. Then, highly-ranked features are selected and utilized in downstream learning tasks. This approach is computationally efficient and scalable for high-dimensional data. In addition, the classifier performance is not controlled during the FS, mitigating overfitting and enhancing generalization capabilities (Jain & Singh, 2018). Existing filter FS methods consider each feature independently and ignore the underlying feature structure (Bolón-Canedo et al., 2015; Li et al., 2017). However, multivariate information could be particularly important for feature selection due to various feature interactions in many real-world applications (Li et al., 2017).

In this work, we propose a filter FS method that identifies the meaningful features by comparing the underlying geometry of the feature spaces of different classes in a supervised setting. First, the *feature manifold* of each class

[1]Viterbi Faculty of Electrical and Computer Engineering, Technion – Israel Institute of Technology, Haifa, Israel [2]CSAIL, Massachusetts Institute of Technology, Cambridge, USA [3]Department of Pathology, Yale School of Medicine, New Haven, CT 06511, USA [4]Applied Mathematics Program, Yale University, New Haven, CT 06511, USA. Correspondence to: David Cohen <davidco@campus.technion.ac.il>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

is learned using a symmetric kernel. Then, based on the Riemannian geometry of the obtained kernels (Pennec et al., 2006; Bhatia, 2009), we build a composite symmetric kernel that captures the differences between the geometries underlying the feature spaces (Shnitzer et al., 2022; 2019). Specifically, we apply spectral analysis to the composite kernel and propose a score that reveals discriminative features. We note that to the best of our knowledge, our work is the first filter FS method to combine feature selection with manifold learning applied to the feature space rather than the typical sample space. This allows us to naturally take into account multivariate information, which is lacking in existing univariate filter FS methods.

The proposed method, which we term *ManiFeSt* (Manifold-based Feature Selection), is theoretically grounded and tested on several benchmark datasets. We show empirically that ManiFeSt improves the identification of informative features compared to other filter FS methods. Using kernels to capture multi-feature associations, and particularly, their spectral analysis, makes ManiFeSt multivariate by design. We posit that such multivariate information enhances the ability to identify the optimal subset of features, leading to improved performance and generalization capabilities, as demonstrated in various experiments. In addition, these properties facilitate few-sample FS, i.e., identifying informative features with only a few labeled samples. In univariate methods, identifying the discriminative features is based only on differences in feature values between classes. When only a small number of samples is available, this procedure becomes very sensitive to noise. In contrast, the feature associations are less sensitive to small sample size, as they are governed by the (large) feature space rather than by the (small) sample size. Indeed, we empirically show that ManiFeSt is superior compared to competing univariate methods when the sample size is small.

Our main contributions are as follows. (i) We present a new approach for FS from a multivariate standpoint, exploiting the geometry underlying the features. We show that considering multi-feature associations, rather than a univariate perspective based on single features, is useful for identifying the features with high discriminative capabilities. (ii) We employ a new methodology for feature manifold learning that combines classical manifold learning with the Riemannian geometry of matrix spaces. (iii) We propose a new algorithm for supervised FS. Our algorithm demonstrates high performance, specifically, improved generalization capabilities, promoting accurate few-sample FS.

2. Related Work

Classical filter methods use statistical tests to rank the features. One of the most straightforward methods is based on computing the Pearson's correlation of each feature with the class label (Battiti, 1994). The ANOVA F-value (Kao & Green, 2008), the t-test (Davis & Sampson, 1986), and the Fisher score (Duda et al., 2006) are similarly used for selecting discriminative features. Other scoring techniques, such as information gain (IG) (Vergara & Estévez, 2014; Ross, 2014) and Gini-index (Shang et al., 2007), select features that maximize the purity of each class.

In addition to statistical methods, a fast-growing class of filter methods rely on geometric considerations. One popular method is the Laplacian score (He et al., 2005; Zhao & Liu, 2007; Lindenbaum et al., 2021), which attempts to evaluate the importance of each feature using a graph-Laplacian that is constructed from the samples. Similarly to the Laplacian score, most of the geometric FS methods consider the geometry underlying the samples. One exception is Relief (Kira & Rendell, 1992) (including its popular extensions (Kononenko, 1994; Robnik-Šikonja & Kononenko, 2003)), in which the score increases or decreases according to the differences between the values of the feature and its nearest neighbors. In contrast to Relief-based methods, our method captures multi-feature associations using kernels, and therefore, it is not limited to nearest neighbors local geometry.

Most existing filter FS methods are univariate, i.e., they consider each feature separately and do not account for multi-feature associations (Bolón-Canedo et al., 2015; Shah & Patel, 2016; Jain & Singh, 2018). Thus, the ability to identify the optimal feature subset may be limited (Li et al., 2017), leading to degraded performance. To mitigate this limitation, mRMR (Minimum Redundancy and Maximum Relevance) (Ding & Peng, 2005; Zhao et al., 2019) and CFS (Correlation-based Feature Selection) (Hall, 1999) algorithms assume that highly correlated features do not contribute to the model and attempt to control feature redundancy. The key idea is to balance between two measures: a relevance measure and a redundancy measure. While mRMR and CFS algorithms may consider feature associations to avoid selecting highly correlated features, thereby controlling the redundancy in a multivariate manner, the relevance measure is univariate. Two notable exceptions are methods that use the trace ratio (Nie et al., 2008) and the generalized fisher score (Gu et al., 2011) as relevance metrics, which are computed based on a subset of features. However, both methods involve optimization that requires more resources than standard FS filter methods.

Traditional geometric FS methods such as the Laplacian score (He et al., 2005), SPEC (Zhao & Liu, 2007), and Relief (Robnik-Šikonja & Kononenko, 2003) evaluate the importance of the features based on the sample space. In the Laplacian score and SPEC, the constructed kernel reflects the sample associations, and in Relief, the nearest neighbors are determined based on the geometry of the samples. In contrast, our method is applied to the feature space rather

than the sample space. Consequently, our method is multivariate, designed to capture complex structures underlying the feature space, leading to improved generalization and consistency.

One concurrent work, DiSC (Sristi et al., 2022), takes a similar approach of examining differences between feature associations through kernels in the feature space. However, DiSC is a feature extraction technique, which constructs meta-features and identifies groups of features, whereas our method is a feature selection technique, allowing the choice of the feature subset size. Moreover, DiSC identifies groups of discriminative features with no score measure within these groups, which may result in very large feature spaces since the number of highlighted features cannot be controlled. We revisit DiSC in Appendix C.

The FS problem shares similarities with the two-sample test (Lehmann et al., 2005), which focuses on determining whether two distributions (classes) are identical or not. FS goes beyond this binary assessment and provides a more detailed understanding of the differences. While interpretable methods like (Jitkrittum et al., 2016) and (Lopez-Paz & Oquab, 2017) offer insights into the regions of difference, they do not provide explicit scores for individual samples. Recent approaches (Kim et al., 2019; Cazáis & Lhéritier, 2015; Landa et al., 2020) employ a refined task termed local two-sample testing to identify the regions of difference. However, it is crucial to differentiate between the two-sample test that identifies specific samples and the FS problem that seeks specific features. This distinction is important because, in the two-sample test, samples are assumed to be i.i.d., while in the feature selection problem, the features may exhibit multivariate relations. Indeed, our approach inherently accounts for these intricate relationships.

3. ManiFeSt - Manifold-based Feature Selection

The proposed algorithm for feature selection consists of three stages. First, a feature space representation is constructed for each class using a kernel. Then, we build a composite kernel that is specifically-designed to capture the difference between the classes. Finally, to reveal the significant features, we apply spectral analysis to the composite kernel and propose a FS score.

3.1. Feature Manifold Learning

Consider a dataset $\boldsymbol{X} \in \mathbb{R}^{N\times d}$ with N samples and d features consisting of two classes. In order to capture differences in the feature associations between the two classes, we propose to learn the underlying geometry of the feature space of each class using a kernel. For this purpose, according to the class labels, the dataset is divided into two subsets $\boldsymbol{X}^{(1)} = [\boldsymbol{x}_1^{(1)}, \cdots, \boldsymbol{x}_d^{(1)}] \in \mathbb{R}^{N_1\times d}$ and $\boldsymbol{X}^{(2)} = [\boldsymbol{x}_1^{(2)}, \cdots, \boldsymbol{x}_d^{(2)}] \in \mathbb{R}^{N_2\times d}$, where $\boldsymbol{x}_i^{(\ell)} \in \mathbb{R}^{N_\ell}$ denotes the ith *feature* in the ℓth class, N_1 and N_2 denote the number of samples in the first and second class, respectively, and $N = N_1 + N_2$.

For each class $\ell = 1, 2$, a radial basis function (RBF) kernel $\boldsymbol{K}_\ell \in \mathbb{R}^{d\times d}$ is constructed as follows:

$$\boldsymbol{K}_\ell[i,j] = e^{-\left\|\boldsymbol{x}_i^{(\ell)} - \boldsymbol{x}_j^{(\ell)}\right\|^2/2\sigma_\ell^2},\ i,j = 1,\ldots,d \quad (1)$$

where σ_ℓ is a scale factor, typically set to the median of the Euclidean distances up to some scalar.

Using kernels is common practice in nonlinear dimension reduction and manifold learning methods (Schölkopf et al., 1997; Tenenbaum et al., 2000; Roweis & Saul, 2000; Belkin & Niyogi, 2003; Coifman & Lafon, 2006). From the standpoint of this approach, the features are viewed as nodes of an undirected weighted graph and the kernel prescribes the weights of the edges connecting the nodes (features). This graph is considered a discrete approximation of the continuous manifold, on which the features reside. Importantly, in contrast to classical manifold learning methods (Tenenbaum et al., 2000; Roweis & Saul, 2000; Belkin & Niyogi, 2003; Coifman & Lafon, 2006), which typically attempt to learn the manifold underlying the samples, our method learns the manifold underlying the *features*, capturing information on the feature associations, and thus, making our approach multivariate. This viewpoint is tightly related to graph signal processing (Shuman et al., 2013; Sandryhaila & Moura, 2013), where graphs, whose nodes are the features of the signals, are similarly computed.

One important property of RBF kernels is that they are symmetric positive semi-definite (SPSD) matrices, a fact that we will exploit next. To simplify the exposition, we will assume here that they are strictly positive (SPD), and address the general case of SPSD matrices in Appendix B. We note that our method is not limited to RBF kernels, and other SPSD kernels could be used instead.

3.2. Operator Composition on the SPD Manifold

One way to extract the differences between the feature spaces through their kernel representation is to simply subtract the kernels $\boldsymbol{K}_1 - \boldsymbol{K}_2$. Although natural, applying such a linear operation violates the SPD geometry of the kernels and in fact assumes that the kernels live in a linear (vector) space. In order to "respect" and exploit the underlying Riemannian SPD geometry, we propose to implement the following two-step procedure in a Riemannian manner. We first find the midpoint $\boldsymbol{M} = (\boldsymbol{K}_1 + \boldsymbol{K}_2)/2$, and then, we compute the differences $\boldsymbol{M} - \boldsymbol{K}_1$ or $\boldsymbol{M} - \boldsymbol{K}_2$. The Riemannian counterparts of the above Euclidean additions and subtractions are described next. See Appendix A for background

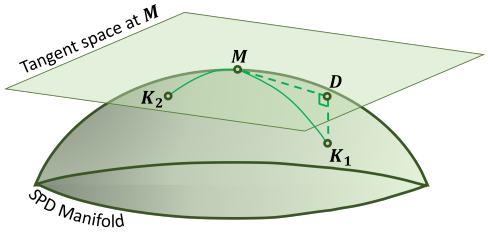

Figure 1. Illustration of the definitions of the operators $\boldsymbol{M}$ and $\boldsymbol{D}$.

on the Riemannian geometry of SPD and SPSD matrices.

Firstly, we compute the mid-point $\boldsymbol{M}$ on the geodesic path connecting $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ (Katz et al., 2020) (which coincides with the Riemannian mean (Pennec et al., 2006)):

$$\boldsymbol{M} = \gamma_{\boldsymbol{K}_1 \to \boldsymbol{K}_2}(\tfrac{1}{2}) = \boldsymbol{K}_1^{1/2}\left(\boldsymbol{K}_1^{-1/2}\boldsymbol{K}_2\boldsymbol{K}_1^{-1/2}\right)^{1/2}\boldsymbol{K}_1^{1/2}. \tag{2}$$

Second, to compute the differences, the two kernels are projected onto the tangent space to the SPD manifold at the mid-point $\boldsymbol{M}$. By definition, this is given by the logarithmic map of $\boldsymbol{K}_1$ at $\boldsymbol{M}$:

$$\boldsymbol{D} = \mathrm{Log}_{\boldsymbol{M}}(\boldsymbol{K}_1) = \boldsymbol{M}^{1/2}\log\left(\boldsymbol{M}^{-1/2}\boldsymbol{K}_1\boldsymbol{M}^{-1/2}\right)\boldsymbol{M}^{1/2}. \tag{3}$$

Note that since $\mathrm{Log}_{\boldsymbol{M}}(\boldsymbol{K}_2) = -\mathrm{Log}_{\boldsymbol{M}}(\boldsymbol{K}_1)$, one projection is sufficient. In Fig. 1, we present an illustration of the construction of the two kernels $\boldsymbol{M}$ and $\boldsymbol{D}$.

Details on the implementation using SPSD geometry appear in Appendix B.

3.3. Proposed Feature Score

The proposed FS score relies on the spectral analysis of the composite kernel $\boldsymbol{D}$. Let $\lambda_i^{(\boldsymbol{D})}$ and $\boldsymbol{\phi}_i^{(\boldsymbol{D})} \in \mathbb{R}^d$ be the eigenvalues and eigenvectors of $\boldsymbol{D}$, respectively. Note that $\boldsymbol{D}$ is symmetric, and therefore, its eigenvalues are real and its eigenvectors form an orthonormal basis.

The proposed FS score $\boldsymbol{r} \in \mathbb{R}^d$ is given by

$$\boldsymbol{r} = \sum_{i=1}^{d} |\lambda_i^{(\boldsymbol{D})}| \cdot (\boldsymbol{\phi}_i^{(\boldsymbol{D})} \odot \boldsymbol{\phi}_i^{(\boldsymbol{D})}) \tag{4}$$

where $\odot$ is the Hadamard (element-wise) product and $\boldsymbol{r}(j)$ is the score of feature j. In words, the magnitude of the eigenvectors is weighted by the eigenvalues and summed over to form the ManiFeSt score. Note that this score is multivariate from two perspectives. First, the kernels capture the pairwise associations of each feature with all other features. Second, due to the spectral decomposition of the difference operator, higher-order associations are captured as well by the eigenvectors $\boldsymbol{\phi}_i^{(\boldsymbol{D})} \in \mathbb{R}^d$, thus providing richer multivariate information.

Our score is related to several previous frameworks that extract new representations (signatures) of data using SPD and SPSD kernels. Two notable signatures, defined for shape analysis tasks, are the heat kernel signature (Sun et al., 2009) and the wave kernel signature (Aubry et al., 2011), both are of the form $\sum_i f(\mu_i)\phi_i^2(x)$, where μ_i and ϕ_i are the eigenvalues and eigenvectors of the Laplace-Beltrami operator and x is a point on the shape. In another recent work (Cheng & Mishne, 2020), such a score was shown to facilitate separation of clustered samples from background. Inspired by these signatures, our score relies on the eigenpairs of the operator $\boldsymbol{D}$, defined in Eq. (3) as the Riemannian difference between the kernels representing the feature spaces of the two classes, $\boldsymbol{K_1}$ and $\boldsymbol{K_2}$. While our score resembles classical kernel signatures, it has two important distinctions. (i) To the best of our knowledge, such kernel signatures have not been used in the context of feature selection in the past. (ii) Perhaps more importantly, the kernel we use (the difference operator $\boldsymbol{D}$) is significantly different than the kernels typically used in the kernel signatures.

ManiFeSt is summarized in Algorithm 1. For simplicity, here we described an algorithm for binary classification problems. A natural geometric extension to multi-class problems is detailed in Appendix C, along with an example depicting the properties of the multi-class ManiFeSt.

Algorithm 1 ManiFeSt Score

Input: Dataset with two classes $\boldsymbol{X}^{(1)}$ and $\boldsymbol{X}^{(2)}$
Output: FS score $\boldsymbol{r}$

1: Construct kernels $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ ▷ According to (1)
2: Build the mean operator $\boldsymbol{M}$ ▷ According to (2)
3: Build the difference operator $\boldsymbol{D}$ ▷ According to (3)
4: Apply eigendecomposition to $\boldsymbol{D}$
5: Compute the FS score $\boldsymbol{r}$ ▷ According to (4)

3.4. Illustrative Example

We illustrate our approach using MNIST (Deng, 2012). We generate two sets consisting of 1500 images of 4 and 1500 images of 9. In this example, the pixels are viewed as features, and we aim to identify pixels that bear discriminative information on 4 and 9. We apply Algorithm 1 and present the results in Fig. 2. Note that the kernel construction approach in this example is not invariant to image transformations such as rotation, due to the Euclidean distance in (1). To accommodate invariance to various transformations in more complicated image datasets, the Euclidean metric in the kernel construction can be replaced by other metrics, e.g., by first embedding the data in some invariant space.

We see in Fig. 2(left) that the two leading eigenvectors of the mid-point kernel, $\boldsymbol{M}$, correspond to the common background and to the common structure of both digits, 4 and

9. In Fig. 2(middle), we see that the leading eigenvectors of the composite difference kernel, $\boldsymbol{D}$, indeed capture the main conceptual differences between the two digits. These differences include the gap at the top of the digit 4, the tilt differences in the digits' legs, and the differences between the round upper part of 9 and the square upper part of 4. As shown in Fig. 2(right), the ManiFeSt score, which weighs the eigenvectors by their respective eigenvalues, provides a consolidated measure of the discriminative pixels. In Appendix D.2.1, we present an additional illustrative example.

4. Theoretical Foundation

We begin with a characterization of the spectrum of $\boldsymbol{D}$. Each kernel, $\boldsymbol{K}_\ell$, captures intrinsic feature associations that characterize the samples in each class. The eigenvectors of these kernels can be used as new representations for the feature spaces, extracting intra-class similarities between features. For each kernel, $\boldsymbol{K}_\ell$, the most dominant components of these feature associations are captured by eigenvectors that correspond to the largest eigenvalues. The motivation for our score then comes from the spectral properties of the difference operator, $\boldsymbol{D}$, which were recently analyzed in (Shnitzer et al., 2022). This work proved that the leading eigenvectors of $\boldsymbol{D}$ (corresponding to the largest eigenvalues in absolute value) are related to similar eigenvectors of $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ that correspond to significantly different eigenvalues. Specifically, it was shown that the eigenvalues of $\boldsymbol{D}$ that correspond to eigenvectors that are (approximately) shared by $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$, are equal to $\lambda^{(\boldsymbol{D})} = \frac{1}{2}\sqrt{\lambda^{(\boldsymbol{K}_1)}\lambda^{(\boldsymbol{K}_2)}}\left(\log(\lambda^{(\boldsymbol{K}_1)}) - \log(\lambda^{(\boldsymbol{K}_2)})\right)$. The term $\sqrt{\lambda^{(\boldsymbol{K}_1)}\lambda^{(\boldsymbol{K}_2)}}$ implies that $\lambda^{(\boldsymbol{D})}$ is dominant only if both $\lambda^{(\boldsymbol{K}_1)}$ and $\lambda^{(\boldsymbol{K}_2)}$ are dominant. In addition, the term $\left(\log(\lambda^{(\boldsymbol{K}_1)}) - \log(\lambda^{(\boldsymbol{K}_2)})\right)$ indicates that $\lambda^{(\boldsymbol{D})}$ is dominant only if $\lambda^{(\boldsymbol{K}_1)} \gg \lambda^{(\boldsymbol{K}_2)}$ or $\lambda^{(\boldsymbol{K}_2)} \gg \lambda^{(\boldsymbol{K}_1)}$. Therefore, in the context of our work, the operator $\boldsymbol{D}$ emphasizes components representing feature associations that are (i) dominant, and (ii) significantly different in the two classes.

Here, we show that high absolute values in eigenvectors of $\boldsymbol{D}$ with large eigenvalues (in absolute value), represent features with significantly different associations between the two classes. Therefore, the eigenvalue weighting in the ManiFeSt score (4) ensures that discriminative features will get high scores.

Proposition 1. *Assume that ϕ is a shared eigenvector of $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ with respective eigenvalues $\lambda^{(\boldsymbol{K}_1)}$ and $\lambda^{(\boldsymbol{K}_2)}$. Then ϕ is an eigenvector of $\boldsymbol{D}$ with an eigenvalue $\lambda^{(\boldsymbol{D})} = \sqrt{\lambda^{(\boldsymbol{K}_1)}\lambda^{(\boldsymbol{K}_2)}}\left(\log\lambda^{(\boldsymbol{K}_1)} - \log\lambda^{(\boldsymbol{K}_2)}\right)$ that satisfies:*

$$\left|\lambda^{(\boldsymbol{D})}\right| \leq 2\sum_{i,j=1}^{d} |\boldsymbol{K}_1[i,j] - \boldsymbol{K}_2[i,j]||\phi(i)||\phi(j)| \quad (5)$$

where $\boldsymbol{K}_\ell[i,j] = e^{-\|x_i^{(\ell)} - x_j^{(\ell)}\|^2/2\sigma^2}$, and $x_i^{(\ell)}$ and $x_j^{(\ell)}$ are vectors containing the values of features i and j, respectively, from all the samples in class $\ell = 1, 2$.

This bound implies that if $\lambda^{(\boldsymbol{D})}$ is large, there must exist pairs of features i_0 and j_0 that significantly contribute to the sum in the right-hand side by satisfying: (i) $|\phi(i_0)|$ and $|\phi(j_0)|$ are large, and (ii) $e^{-\|x_i^{(1)} - x_j^{(1)}\|^2/2\sigma^2} - e^{-\|x_i^{(2)} - x_j^{(2)}\|^2/2\sigma^2}$ is large, implying on a significant difference of the feature associations between the two classes. In other words, this derivation indicates that a feature i that is discriminative in a multivariate sense, i.e., a feature whose associations with other features are significantly different between the two classes, is represented by a high value $|\phi(i)|$ in eigenvectors corresponding to large eigenvalues, $|\lambda^{(\boldsymbol{D})}|$. Observing the expression in (4), it is evident that features i with high values of $|\phi(i)|$ in eigenvectors corresponding to large $|\lambda^{(\boldsymbol{D})}|$ are assigned high ManiFeSt scores.

The proof of Proposition 1 appears in Appendix E, along with more general result for approximately shared eigenvectors in Proposition 2.

Additional motivation for using $\boldsymbol{D}$ to recover differences between the feature spaces can be demonstrated through small perturbations of the kernels, as shown in (Shnitzer et al., 2022, Proposition 3) and repeated here for completeness.

Proposition 3. *Assume $\boldsymbol{K}_2 = \boldsymbol{K}_1 + \boldsymbol{E}$ such that $\|\boldsymbol{E}\boldsymbol{K}_1^{-1}\| < 1$, then $\boldsymbol{D} \approx -\frac{1}{2}\left(\boldsymbol{K}_2 - \boldsymbol{K}_1\right)\left(\boldsymbol{K}_1^{-1}\boldsymbol{K}_2\right)^{1/2}$.*

This result shows that $\boldsymbol{D}$ is related to the differences between the graph kernels factored by a term related to the Riemannian metric of the space of SPD matrices. More intuitively, the definition of $\boldsymbol{D}$ as the logarithmic map, which maps a point from the manifold to the tangent space, is the Riemannian counterpart of subtraction in a linear space. The proof of Proposition 3 appears in Appendix E.

To conclude this section, we summarize the main results concerning the properties of $\boldsymbol{D}$ and the ManiFeSt score:

- The eigenvalue structure of $\boldsymbol{D}$ (Propositions 1 and 2) and the characterization of $\boldsymbol{D}$ (Proposition 3) indicate that feature relations that are dominant only in one feature graph, represented by $\boldsymbol{K}_1$ or $\boldsymbol{K}_2$, will be captured by an eigenvector with a large eigenvalue in $\boldsymbol{D}$. Therefore, the leading eigenvectors of $\boldsymbol{D}$ highlight the differences between the feature graphs.
- Proposition 1 demonstrates that the eigenvalues of $\boldsymbol{D}$ are bounded from above by the differences between the feature graph kernels and the eigenvector values. This strengthens the claim that the largest eigenvalues of $\boldsymbol{D}$ (in absolute values) correspond to eigenvectors that highlight features which are differently connected in the feature graphs of the classes. This provides additional motivation for the feature score that is weighted by these eigenvalues.

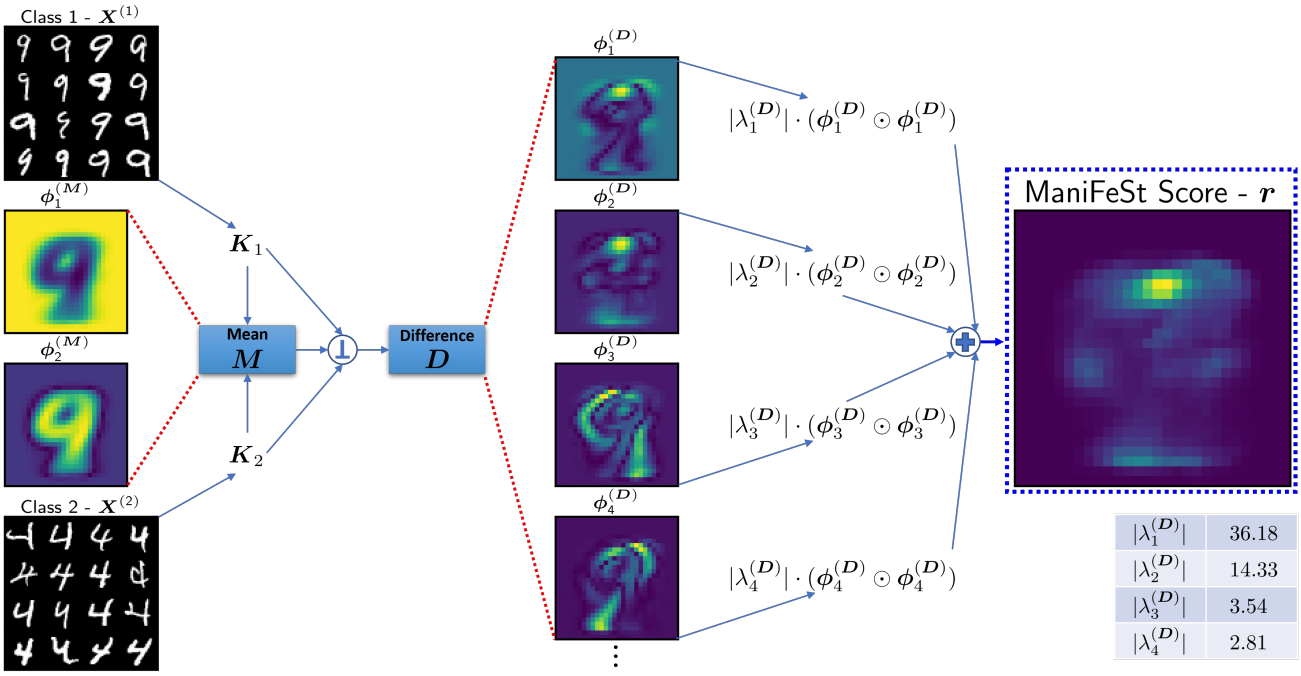

Figure 2. Illustration of the proposed approach and the resulting ManiFeSt score for digit recognition.

5. Experiments

We demonstrate the performance of ManiFeSt on synthetic and real datasets and compare it to commonly-used FS methods. In all experiments, the data is split to train and test sets with nested cross-validation. All competing FS methods are tuned to achieve best results on the train set. Additional implementation details and results are in Appendix D.

5.1. XOR-100 Problem

Following (Kim et al., 2010; Bolón-Canedo et al., 2011; Yamada et al., 2020), we generate a synthetic XOR dataset consisting of $d = 100$ binary features and $N = 50$ instances. Each feature is sampled from a Bernoulli distribution, and each instance is associated with a label given by $y = f_1 \oplus f_5$, where $\oplus$ is the XOR operation and f_i is the ith feature. Thus, only two features, f_1 and f_5, are relevant for the class label.

This seemingly simple problem is in fact challenging, especially for existing univariate filter FS methods that consider each feature independently and ignore the inherent feature structure (Bolón-Canedo et al., 2015; Li et al., 2017).

In Fig. 3, we present the normalized feature score obtained by the tested FS methods averaged over 200 Monte-Carlo iterations of data generation. The green circles denote the average score, while the red dashes indicate the standard deviation. In each iteration, the two features with maximal scores are selected, and the average number of correct selections for each method is denoted in parentheses. The results indicate that ManiFeSt perfectly identifies the multivariate behavior of f_1 and f_5, whereas all other compared methods, except for ReliefF, fail.

We note that this XOR-100 problem is a multivariate problem, because the XOR result depends on f_1 and f_5. Nevertheless, we see that ReliefF, which is arguably a univariate method (Jović et al., 2015), identifies the relevant features. Although ReliefF examines each feature separately, it considers neighboring samples, which evidently provide sufficient multivariate information to correctly detect the features in this example. Still, ManiFeSt, which is multivariate by design, outperforms ReliefF obtaining perfect identification of the two relevant features relative to 0.8 of ReliefF.

5.2. Madelon

We test ManiFeSt on the Madelon synthetic dataset (Guyon et al., 2008) from the NIPS 2003 feature selection challenge. Based on a 5-dimensional hypercube embedded in $\mathbb{R}^5$, the Madelon dataset consists of 2600 points grouped into 32 clusters. Each cluster is normally distributed and centered at one of the hypercube vertices. The clusters are randomly assigned to one of two classes. Each point is a vector of 500 features, where only 20 are relevant: 5 correspond to the coordinates of the hypercube, and 15 are random linear combinations of them. The remaining features are noise.

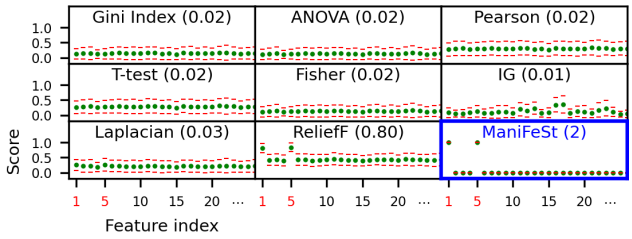

Figure 3. Feature ranking for the XOR-100 problem. Green dots denote the average score, and red lines indicate the standard deviation. The average number of correct FS is denoted in parentheses.

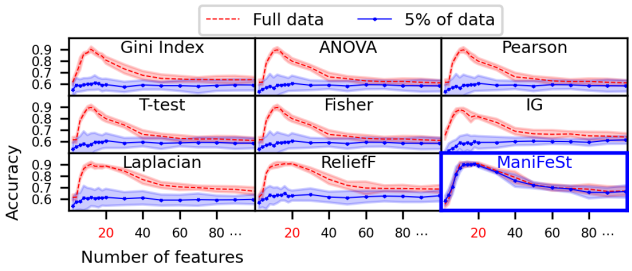

Figure 4. Classification accuracy as a function of the feature number on the Madelon dataset. Lines denote the average test accuracy, and the shaded area denote the standard deviation.

The data is divided into train and test sets with a 10-fold cross-validation. We consider two cases. In the first case, the FS is based on the whole train set (2340 points). In the second case, the FS is based on only 5 percent of the train set (117 points). Since no ground-truth is available for the relevant features, for evaluation, an SVM classifier is optimized using the entire train set in both cases.

Fig. 4 shows the classification accuracy obtained based on different subsets of features by the tested methods. The curves indicate the average test accuracy, and the shaded area represents the standard deviation. We see based on the red curves that all the methods identify relevant features when the assessment of the FS score is based on the entire train set. Note that selecting too few or too many features may lead to poor classification. The best average test accuracy is 90.73% and is achieved by both ManiFeSt and ReliefF (which obtained the best result in the NIPS 2003 challenge (Guyon et al., 2007)).

Furthermore, when the FS is based on a reduced number of samples, all the methods but ManiFeSt fail to capture the relevant features, as demonstrated by the blue curves. In contrast, the classification obtained by ManiFeSt is unaffected, showing a remarkable robustness to reduction in sample size, thus suggesting good generalization capabilities.

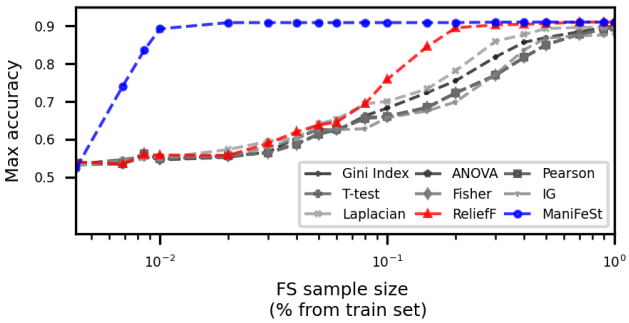

Figure 5. Max test classification accuracy as a function of the FS sample size on the Madelon dataset.

We further explore the effect of the sample size on the performance of ManiFeSt. Fig. 5 shows the obtained maximum classification accuracy versus the sample size (in log scale). We see that all the methods but ManiFeSt suffer from severe degradation when the sample size is reduced, and fail completely for sample size that is less than 10 percent (234 samples). Conversely, ManiFeSt demonstrates robustness to the sample size and even performs well when the sample size consists of only 1 percent of the samples (23 samples).

5.3. Clusters on a Hypercube

We simulate a variant of the Madelon dataset (Guyon, 2003) using the scikit-learn function make_classification(). We consider this variant because here the ground truth is available, whereas the Madelon dataset lacks information on the relevant features. See more details on the dataset generation in Appendix D.1.

We generate 2000 samples consisting of 200 features with 10 relevant features and split the dataset into train and test sets with 1500 and 500 samples, respectively. For FS, we use only 50 samples from the train set to emphasize the effectiveness of ManiFeSt with only a few labeled samples. We select the top 10 features according to each FS method, and an SVM is optimized using the entire train set with the selected features. We repeat this procedure using 50 cross-validation iterations.

Fig. 6(a) presents the number of correct selections obtained by the tested FS methods. The median and average values are denoted by red lines and circles. The boundaries of the box indicate the 25^{th} and 75^{th} percentiles. We see that ManiFeSt outperforms the competing methods by a large margin using a relatively small number of samples.

Fig. 6(b) shows the t-SNE visualization (Van der Maaten & Hinton, 2008) for a single realization of the test samples using all the features (left), top 10 features selected by ReliefF (middle), and top 10 features selected by ManiFeSt (right).

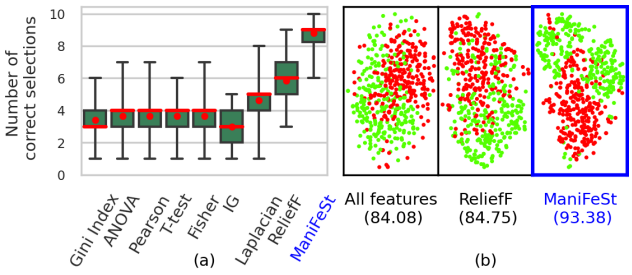

Figure 6. Results on the simulated variant of the Madelon dataset. (a) Boxplot of the number of correct selections indicating the 25$^{\text{th}}$ and the 75$^{\text{th}}$ percentiles. The median and average are denoted by red lines and circles, respectively. (b) t-SNE visualization of the test samples using all the features (left), top 10 features selected by ReliefF (middle), and top 10 features selected by ManiFeSt (right).

The color (red and green) denotes the (hidden) class label. Both FS methods lead to a better class separation, while the separation using ManiFeSt is more pronounced. We report that ManiFeSt yields 9 (out of 10) correct selections, whereas ReliefF only 6. In addition, the average accuracy on the selected features is depicted in parentheses, further demonstrating the advantage of ManiFeSt over ReliefF.

5.4. Colon Cancer Gene Expression

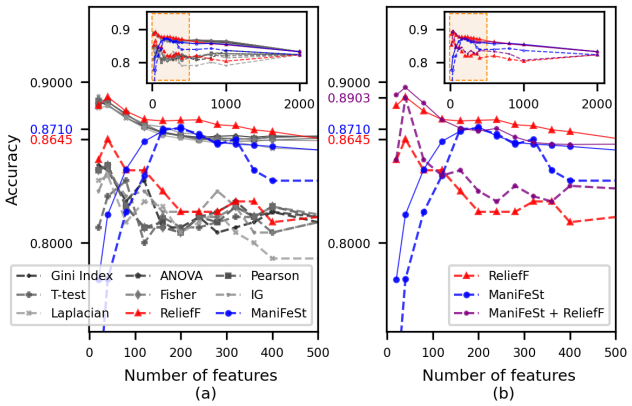

Figure 7. Accuracy as a function of the number of features on the colon cancer gene expression dataset. The dashed and solid lines represent the average test and validation accuracy, respectively.

We test a dataset of colon cancer gene expression samples (Alon et al., 1999), which is relatively small, typical to the biological domain. The dataset consists of the expression levels of 2000 genes (features) in 62 tissues (samples), of which 22 are normal, and 40 are of colon cancer. The samples are split to 90% train and 10% test sets. Results are averaged over 50 cross-validation iterations.

Fig. 7(a) shows the average classification accuracy for different subsets of features. The solid and dashed lines represent the average validation and test accuracy, respectively. ManiFeSt achieves the best test accuracy of 87.10%, whereas the test accuracy of competing methods is 86.45% or below.

Even though generalization can be improved by applying FS, the FS itself is still prone to overfitting. Indeed, from the gap between the validation and test accuracy, we see that ManiFeSt generalizes well compared to all other competing methods. To further test the generalization capabilities of ManiFeSt, in Appendix D.2.2, we present the generalization error obtained by ManiFeSt for three different kernel scales, i.e., σ_ℓ in Eq. (1). The results imply that the larger the scale is, i.e., the more feature associations are captured by the kernel, the smaller the generalization error becomes. This may suggest that considering the associations between the features enhances generalization, in contrast to the competing methods that only consider univariate feature properties.

While ManiFeSt exhibits enhanced generalization capabilities, its maximal performance is achieved by using more features (200 compared to 40). Our empirical examination revealed that ManiFeSt selects some irrelevant features because it analyzes feature associations rather than each feature separately. Therefore, ManiFeSt might identify features without any discriminative capabilities, through their connections to other relevant and discriminative features (see Appendix D.2.3). Yet, despite the selection of irrelevant features, ManiFeSt facilitates the best test accuracy.

To alleviate the selection of irrelevant features, we propose combining classical univariate criteria with our multivariate score of ManiFeSt (4). In Fig. 7(b), we present the results obtained when combining ManiFeSt with ReliefF by summing their normalized feature scores. We see that this simple combination results in improved performance. Now, the maximal accuracy is 89.03%, and it is obtained by selecting only 40 features. This result calls for further research, exploring systematic ways to combine the multivariate standpoint of ManiFeSt with univariate considerations.

The enhanced generalization capabilities are demonstrated here only with respect to *filter* methods since embedded and wrapper methods typically suffer from large generalization errors when applied to small datasets (Brown et al., 2012; Bolón-Canedo et al., 2013; Venkatesh & Anuradha, 2019). To support this claim, we report that a recent embedded method applied to the colon dataset obtained test accuracy of 83.85% (Yang et al., 2022), outperforming various other embedded methods. By using the same train-test split scheme (49/13) as in (Yang et al., 2022), ManiFeSt achieves test accuracy of 85.38% with 400 features. The combination with ReliefF obtains 84% accuracy with 80 features. See more comparisons in Appendix D.2.2. We note that filter methods are usually used as preprocessing

Table 1. Comparison of error, standard deviation, and number of features used for various filter FS methods on benchmark datasets.

	Datasets		
Methods (# Features \# Samples)	Gisette (5000 \7000)	Colon (2000 \62)	Prostate (5966 \102)
All features baseline	1.96 ± 0.57	17.74 ± 13.69	8.82 ± 9.23
Gini Index	1.41 ± 0.45 (700)	15.16 ± 12.23 (20)	5.88 ± 7.32 (119)
ANOVA	1.34 ± 0.47 (800)	14.52 ± 10.43 (16)	5.88 ± 6.88 (357)
Pearson	1.34 ± 0.47 (800)	14.52 ± 10.43 (16)	5.88 ± 6.88 (357)
T-test	1.34 ± 0.47 (800)	16.13 ± 12.56 (80)	5.56 ± 7.31 (715)
Fisher	1.34 ± 0.47 (800)	14.52 ± 10.43 (16)	5.88 ± 6.88 (357)
IG	1.40 ± 0.50 (800)	13.87 ± 11.58 (40)	6.21 ± 7.30 (20)
Laplacian	1.39 ± 0.40 (1500)	14.19 ± 10.51 (18)	6.21 ± 6.83 (8)
ReliefF	1.39 ± 0.44 (1500)	13.23 ± 11.70 (20)	5.56 ± 8.52 (1073)
ManiFeSt (ours)	1.29 ± 0.50 (700)	12.90 ± 12.71 (200)	$\mathbf{5.23} \pm 7.82$ (119)
ManiFeSt + ReliefF (ours)	$\mathbf{1.16} \pm 0.27$ (600)	$\mathbf{10.97} \pm 10.96$ (40)	$\mathbf{5.23} \pm 6.94$ (238)

for wrapper and embedded methods (Alshamlan et al., 2015; Shaban et al., 2020; Peng et al., 2010). In such an approach, ManiFeSt may provide explainable prepossessing without eliminating multivariate structures unlike existing filters.

5.5. Additional Results

We compare our approach with different FS methods on two additional datasets, Gisette and Prostate (see dataset details in Appendix D.1). We summarize the results, along with the colon cancer dataset, in Table 1. This table depicts that ManiFeSt obtains the lowest classification errors for all datasets. Furthermore, the combination of our multivariate approach with a classical univariate criteria (ManiFeSt + ReliefF) achieves best results with a few features compared to all other competing methods.

6. Limitations and Future Directions

Our method has several limitations; we outline them and propose possible remedies. First, as demonstrated empirically, analyzing multivariate associations rather than univariate, may lead to selection of irrelevant features. In future work, we will investigate combinations of (classical) univariate criteria and ManiFeSt, making systematic and precise the presented ad hoc combination of ReliefF and ManiFeSt. A similar approach can also aid in reducing feature redundancies, which our method currently does not account for.

Second, as a kernel method, ManiFeSt cannot be applied to very large feature spaces (of an order of magnitude > 10K), though our algorithm can handle data with thousands of features. Combined with the ability to perform well with only few samples, our method applies to a broad range of real-world problems as demonstrated on multiple data sets in the paper, including the prostate data set with 5966 features. In addition, most of these results were obtained on a standard personal computer without GPUs. According to a recent work (Fawzi & Goulbourne, 2021), using GPUs could allow for a faster computation of the eigenvalue decomposition required by ManiFeSt. Therefore, using high performance computing resources will allow us to handle even larger feature spaces. While in the past saving computing resources was a primary goal in applying FS, today high-end computational resources are available, and the main goal of FS is to prevent overfitting in high-dimensional data.

7. Conclusions

In this work, we propose a theoretically grounded supervised FS method. The proposed method, which is termed ManiFeSt, identifies discriminative features by comparing the multi-feature associations of each class. To this end, ManiFeSt employs a geometric approach that combines manifold learning and Riemannian geometry. In contrast to common FS filter methods, our method learns the geometry in the feature space underlying the multi-feature associations rather than applying a univariate analysis. We demonstrate that our multivariate approach reveals various data structures and facilitates improved generalization and consistency for FS in small datasets, outperforming competing FS methods.

8. Acknowledgements

The work of D.C. and R.T. was supported by the European Union's Horizon2020 research and innovation programme under grant agreement No. 802735-ERC-DIFFOP. The work of Y.K. was supported by grants numbers: R01GM131642, UM1PA051410, R33DA047037, U54AG076043, U54AG079759, U01DA053628, P50CA121974, R01GM135928. The work of T.S. was supported by the MIT-IBM Watson AI Laboratory, through their generous support of the MIT Geometric Data Processing group. R.T. also acknowledges the support of the Schmidt Career Advancement Chair.

References

Alelyani, S., Tang, J., and Liu, H. Feature selection for clustering: A review. *Data Clustering*, pp. 29–60, 2018.

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., and Levine, A. J. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences*, 96(12): 6745–6750, 1999.

Alshamlan, H. M., Badr, G. H., and Alohali, Y. A. Genetic bee colony (GBC) algorithm: A new gene selection method for microarray cancer classification. *Computational biology and chemistry*, 56:49–60, 2015.

Arsigny, V., Fillard, P., Pennec, X., and Ayache, N. Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM journal on matrix analysis and applications*, 29(1):328–347, 2007.

Atashgahi, Z., Pieterse, J., Liu, S., Mocanu, D. C., Veldhuis, R., and Pechenizkiy, M. A brain-inspired algorithm for training highly sparse neural networks. *Machine Learning*, pp. 1–42, 2022.

Aubry, M., Schlickewei, U., and Cremers, D. The wave kernel signature: A quantum mechanical approach to shape analysis. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)*, pp. 1626–1633, 2011.

Barachant, A., Bonnet, S., Congedo, M., and Jutten, C. Classification of covariance matrices using a riemannian-based kernel for bci applications. *Neurocomputing*, 112: 172–178, 2013.

Battiti, R. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on neural networks*, 5(4):537–550, 1994.

Belkin, M. and Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Bhatia, R. Positive definite matrices. In *Positive Definite Matrices*. Princeton university press, 2009.

Bhatia, R., Jain, T., and Lim, Y. On the bures–wasserstein distance between positive definite matrices. *Expositiones Mathematicae*, 37(2):165–191, 2019.

Bolón-Canedo, V., Sánchez-Marono, N., and Alonso-Betanzos, A. On the behavior of feature selection methods dealing with noise and relevance over synthetic scenarios. In *The 2011 International Joint Conference on Neural Networks*, pp. 1530–1537. IEEE, 2011.

Bolón-Canedo, V., Sánchez-Maroño, N., and Alonso-Betanzos, A. A review of feature selection methods on synthetic data. *Knowledge and information systems*, 34 (3):483–519, 2013.

Bolón-Canedo, V., Sánchez-Maroño, N., and Alonso-Betanzos, A. *Feature selection for high-dimensional data*. Springer, 2015.

Bolón-Canedo, V., Alonso-Betanzos, A., Morán-Fernández, L., and Cancela, B. Feature selection: From the past to the future. In *Advances in Selected Artificial Intelligence Areas*, pp. 11–34. Springer, 2022.

Bonnabel, S. and Sepulchre, R. Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank. *SIAM Journal on Matrix Analysis and Applications*, 31(3):1055–1070, 2010.

Brown, G., Pocock, A., Zhao, M.-J., and Luján, M. Conditional likelihood maximisation: a unifying framework for information theoretic feature selection. *The journal of machine learning research*, 13(1):27–66, 2012.

Cazáis, F. and Lhéritier, A. Beyond two-sample-tests: Localizing data discrepancies in high-dimensional spaces. In *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–10. IEEE, 2015.

Cheng, X. and Mishne, G. Spectral embedding norm: Looking deep into the spectrum of the graph Laplacian. *SIAM journal on imaging sciences*, 13(2):1015–1048, 2020.

Coifman, R. R. and Lafon, S. Diffusion maps. *Applied and computational harmonic analysis*, 21(1):5–30, 2006.

Davis, J. C. and Sampson, R. J. *Statistics and data analysis in geology*, volume 646. Wiley New York, 1986.

Deng, L. The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

Ding, C. and Peng, H. Minimum redundancy feature selection from microarray gene expression data. *Journal of bioinformatics and computational biology*, 3(02):185–205, 2005.

Duda, R. O., Hart, P. E., et al. *Pattern classification*. John Wiley & Sons, 2006.

Fawzi, H. and Goulbourne, H. Faster proximal algorithms for matrix optimization using jacobi-based eigenvalue methods. *Advances in Neural Information Processing Systems*, 34, 2021.

Gu, Q., Li, Z., and Han, J. Generalized fisher score for feature selection. In *27th Conference on Uncertainty in Artificial Intelligence, UAI 2011*, pp. 266–273, 2011.

Guyon, I. Design of experiments of the NIPS 2003 variable selection benchmark. In *NIPS 2003 workshop on feature extraction and feature selection*, volume 253, pp. 40, 2003.

Guyon, I., Li, J., Mader, T., Pletscher, P. A., Schneider, G., and Uhr, M. Competitive baseline methods set new standards for the NIPS 2003 feature selection benchmark. *Pattern recognition letters*, 28(12):1438–1444, 2007.

Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L. A. *Feature extraction: foundations and applications*, volume 207. Springer, 2008.

Hall, M. A. *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato, 1999.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

He, X., Cai, D., and Niyogi, P. Laplacian score for feature selection. *Advances in neural information processing systems*, 18, 2005.

Hira, Z. M. and Gillies, D. F. A review of feature selection and feature extraction methods applied on microarray data. *Advances in bioinformatics*, 2015, 2015.

Hsu, C.-W., Chang, C.-C., Lin, C.-J., et al. A practical guide to support vector classification, 2003.

Izetta, J., Verdes, P. F., and Granitto, P. M. Improved multiclass feature selection via list combination. *Expert Systems with Applications*, 88:205–216, 2017.

Jain, D. and Singh, V. Feature selection and classification systems for chronic disease prediction: A review. *Egyptian Informatics Journal*, 19(3):179–189, 2018.

Jitkrittum, W., Szabó, Z., Chwialkowski, K. P., and Gretton, A. Interpretable distribution features with maximum testing power. *Advances in Neural Information Processing Systems*, 29, 2016.

Jović, A., Brkić, K., and Bogunović, N. A review of feature selection methods with applications. In *2015 38th international convention on information and communication technology, electronics and microelectronics (MIPRO)*, pp. 1200–1205. IEEE, 2015.

Kao, L. S. and Green, C. E. Analysis of variance: is there a difference in means and what does it mean? *Journal of Surgical Research*, 144(1):158–170, 2008.

Katz, O., Lederman, R. R., and Talmon, R. Spectral flow on the manifold of spd matrices for multimodal data processing. *arXiv preprint arXiv:2009.08062*, 2020.

Kim, G., Kim, Y., Lim, H., and Kim, H. An MLP-based feature subset selection for HIV-1 protease cleavage site analysis. *Artificial intelligence in medicine*, 48(2-3):83–89, 2010.

Kim, I., Lee, A. B., and Lei, J. Global and local two-sample tests via regression. *Electronic Journal of Statistics*, 13 (2):5253–5305, 2019.

Kira, K. and Rendell, L. A. A practical approach to feature selection. In *Machine learning proceedings 1992*, pp. 249–256. Elsevier, 1992.

Kononenko, I. Estimating attributes: Analysis and extensions of RELIEF. In *European conference on machine learning*, pp. 171–182. Springer, 1994.

Landa, B., Qu, R., Chang, J., and Kluger, Y. Local two-sample testing over graphs and point-clouds by random-walk distributions. *arXiv preprint arXiv:2011.03418*, 2020.

Lazar, C., Meganck, S., Taminau, J., Steenhoff, D., Coletta, A., Molter, C., Weiss-Solís, D. Y., Duque, R., Bersini, H., and Nowé, A. Batch effect removal methods for microarray gene expression data integration: a survey. *Briefings in bioinformatics*, 14(4):469–490, 2013.

Lehmann, E. L., Romano, J. P., and Casella, G. *Testing statistical hypotheses*, volume 3. Springer, 2005.

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., and Liu, H. Feature selection: A data perspective. *ACM computing surveys (CSUR)*, 50(6):1–45, 2017.

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., and Liu, H. Feature selection: A data perspective. *ACM Computing Surveys (CSUR)*, 50(6):94, 2018.

Lin, Z. Riemannian geometry of symmetric positive definite matrices via cholesky decomposition. *SIAM Journal on Matrix Analysis and Applications*, 40(4):1353–1370, 2019.

Lindenbaum, O., Salhov, M., Yeredor, A., and Averbuch, A. Gaussian bandwidth selection for manifold learning and classification. *Data mining and knowledge discovery*, 34: 1676–1712, 2020.

Lindenbaum, O., Shaham, U., Peterfreund, E., Svirsky, J., Casey, N., and Kluger, Y. Differentiable unsupervised feature selection based on a gated laplacian. *Advances in Neural Information Processing Systems*, 34:1530–1542, 2021.

Lopez-Paz, D. and Oquab, M. Revisiting classifier two-sample tests. In *International Conference on Learning Representations*, 2017.

Malago, L., Montrucchio, L., and Pistone, G. Wasserstein riemannian geometry of gaussian densities. *Information Geometry*, 1(2):137–179, 2018.

Massart, E. and Absil, P.-A. Quotient geometry with simple geodesics for the manifold of fixed-rank positive-semidefinite matrices. *SIAM Journal on Matrix Analysis and Applications*, 41(1):171–198, 2020.

Moakher, M. A differential geometric approach to the geometric mean of symmetric positive-definite matrices. *SIAM Journal on Matrix Analysis and Applications*, 26 (3):735–747, 2005.

Nie, F., Xiang, S., Jia, Y., Zhang, C., and Yan, S. Trace ratio criterion for feature selection. In *AAAI*, volume 2, pp. 671–676, 2008.

Peng, Y., Wu, Z., and Jiang, J. A novel feature selection approach for biomedical data classification. *Journal of Biomedical Informatics*, 43(1):15–23, 2010.

Pennec, X., Fillard, P., and Ayache, N. A Riemannian framework for tensor computing. *International Journal of computer vision*, 66(1):41–66, 2006.

Remeseiro, B. and Bolon-Canedo, V. A review of feature selection methods in medical applications. *Computers in biology and medicine*, 112:103375, 2019.

Robnik-Šikonja, M. and Kononenko, I. Theoretical and empirical analysis of ReliefF and RReliefF. *Machine learning*, 53(1):23–69, 2003.

Ross, B. C. Mutual information between discrete and continuous data sets. *PloS one*, 9(2):e87357, 2014.

Roweis, S. T. and Saul, L. K. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500): 2323–2326, 2000.

Sandryhaila, A. and Moura, J. M. Discrete signal processing on graphs. *IEEE transactions on signal processing*, 61 (7):1644–1656, 2013.

Schölkopf, B., Smola, A., and Müller, K.-R. Kernel principal component analysis. In *International conference on artificial neural networks*, pp. 583–588. Springer, 1997.

Shaban, W. M., Rabie, A. H., Saleh, A. I., and Abo-Elsoud, M. A new COVID-19 patients detection strategy (CPDS) based on hybrid feature selection and enhanced knn classifier. *Knowledge-Based Systems*, 205:106270, 2020.

Shah, F. P. and Patel, V. A review on feature selection and feature extraction for text classification. In *2016 international conference on wireless communications, signal processing and networking (WiSPNET)*, pp. 2264–2268. IEEE, 2016.

Shang, W., Huang, H., Zhu, H., Lin, Y., Qu, Y., and Wang, Z. A novel feature selection algorithm for text categorization. *Expert Systems with Applications*, 33(1):1–5, 2007.

Shnitzer, T., Ben-Chen, M., Guibas, L., Talmon, R., and Wu, H.-T. Recovering hidden components in multimodal data with composite diffusion operators. *SIAM Journal on Mathematics of Data Science*, 1(3):588–616, 2019.

Shnitzer, T., Wu, H.-T., and Talmon, R. Spatiotemporal analysis using Riemannian composition of diffusion operators. *arXiv preprint arXiv:2201.08530*, 2022.

Shuman, D. I., Narang, S. K., Frossard, P., Ortega, A., and Vandergheynst, P. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3):83–98, 2013.

Singh, D., Febbo, P. G., Ross, K., Jackson, D. G., Manola, J., Ladd, C., Tamayo, P., Renshaw, A. A., D'Amico, A. V., Richie, J. P., et al. Gene expression correlates of clinical prostate cancer behavior. *Cancer cell*, 1(2):203–209, 2002.

Sristi, R. D., Mishne, G., and Jaffe, A. Disc: Differential spectral clustering of features. In *Advances in Neural Information Processing Systems*, 2022.

Sun, J., Ovsjanikov, M., and Guibas, L. A concise and provably informative multi-scale signature based on heat diffusion. In *Computer graphics forum*, volume 28, pp. 1383–1392. Wiley Online Library, 2009.

Tang, J., Alelyani, S., and Liu, H. Feature selection for classification: A review. *Data classification: Algorithms and applications*, pp. 37, 2014.

Tenenbaum, J. B., Silva, V. d., and Langford, J. C. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000.

Van der Maaten, L. and Hinton, G. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11), 2008.

Vandereycken, B., Absil, P.-A., and Vandewalle, S. Embedded geometry of the set of symmetric positive semidefinite matrices of fixed rank. In *2009 IEEE/SP 15th Workshop on Statistical Signal Processing*, pp. 389–392. IEEE, 2009.

Vandereycken, B., Absil, P.-A., and Vandewalle, S. A riemannian geometry with complete geodesics for the set of positive semidefinite matrices of fixed rank. *IMA Journal of Numerical Analysis*, 33(2):481–514, 2013.

Venkatesh, B. and Anuradha, J. A review of feature selection and its methods. *Cybernetics and information technologies*, 19(1):3–26, 2019.

Vergara, J. R. and Estévez, P. A. A review of feature selection methods based on mutual information. *Neural computing and applications*, 24(1):175–186, 2014.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Yair, O., Lahav, A., and Talmon, R. Symmetric positive semi-definite riemannian geometry with application to domain adaptation. *arXiv preprint arXiv:2007.14272*, 2020.

Yamada, Y., Lindenbaum, O., Negahban, S., and Kluger, Y. Feature selection using stochastic gates. In *International Conference on Machine Learning*, pp. 10648–10659. PMLR, 2020.

Yang, J., Lindenbaum, O., and Kluger, Y. Locally sparse neural networks for tabular biomedical data. In *International Conference on Machine Learning*, pp. 25123–25153. PMLR, 2022.

Zhao, Z. and Liu, H. Spectral feature selection for supervised and unsupervised learning. In *Proceedings of the 24th international conference on Machine learning*, pp. 1151–1157, 2007.

Zhao, Z., Anand, R., and Wang, M. Maximum relevance and minimum redundancy feature selection methods for a marketing machine learning platform. In *2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 442–452, 2019.

A. Background on Riemannian Geometry

A.1. Riemannian Manifold of SPD Matrices

Let $\mathcal{S}_d$ denote the set of the symmetric matrices in $\mathbb{R}^{d\times d}$. $\boldsymbol{K} \in \mathcal{S}_d$ is an SPD matrix if all its eigenvalues are strictly positive. We denote the set of $d\times d$ SPD matrices as $\mathcal{P}_d$. The tangent space at $\boldsymbol{K} \in \mathcal{P}_d$ is the space of symmetric matrices and is denoted by $\mathcal{T}_{\boldsymbol{K}}\mathcal{P}_d$. When the tangent space is endowed with a proper metric the space of SPD matrices forms a differential Riemannian manifold (Moakher, 2005). Various metrics have been proposed in the literature (Arsigny et al., 2007; Bhatia et al., 2019; Lin, 2019; Malago et al., 2018; Pennec et al., 2006), of which the affine invariant metric (Pennec et al., 2006) and the log-Euclidean metric (Arsigny et al., 2007) are arguably the most widely used, both allow formal definitions of geometric notions such as the geodesic path on the manifold. We focus here on the affine invariant metric, defined for $\boldsymbol{S}_1, \boldsymbol{S}_2 \in \mathcal{T}_{\boldsymbol{K}}\mathcal{P}_d$ as follows:

$$\langle \boldsymbol{S}_1, \boldsymbol{S}_2 \rangle_{\boldsymbol{K}} = \left\langle \boldsymbol{K}^{-1/2}\boldsymbol{S}_1\boldsymbol{K}^{-1/2}, \boldsymbol{K}^{-1/2}\boldsymbol{S}_2\boldsymbol{K}^{-1/2} \right\rangle \tag{6}$$

where $\langle \boldsymbol{A}_1, \boldsymbol{A}_2 \rangle = \mathrm{Tr}\left(\boldsymbol{A}_1^T\boldsymbol{A}_2\right)$ is the standard Euclidean inner product. Based on this metric, the unique geodesic path on the SPD manifold connecting two matrices $\boldsymbol{K}_1, \boldsymbol{K}_2 \in \mathcal{P}_d$ is given by:

$$\gamma^{\mathcal{P}}_{\boldsymbol{K}_1\to\boldsymbol{K}_2}(t) = \boldsymbol{K}_1^{1/2}\left(\boldsymbol{K}_1^{-1/2}\boldsymbol{K}_2\boldsymbol{K}_1^{-1/2}\right)^t\boldsymbol{K}_1^{1/2} \tag{7}$$

where $0 \le t \le 1$. It holds that $\gamma^{\mathcal{P}}_{\boldsymbol{K}_1\to\boldsymbol{K}_2}(0) = \boldsymbol{K}_1$ and $\gamma^{\mathcal{P}}_{\boldsymbol{K}_1\to\boldsymbol{K}_2}(1) = \boldsymbol{K}_2$.

The projection of a point (symmetric matrix) in the tangent space $\boldsymbol{S} \in \mathcal{T}_{\boldsymbol{K}}\mathcal{P}_d$ to the SPD manifold is given by the following exponential map:

$$\mathrm{Exp}_{\boldsymbol{K}}(\boldsymbol{S}) = \boldsymbol{K}^{1/2}\exp\left(\boldsymbol{K}^{-1/2}\boldsymbol{S}\boldsymbol{K}^{-1/2}\right)\boldsymbol{K}^{1/2} \tag{8}$$

where the result $\widetilde{\boldsymbol{K}} = \mathrm{Exp}_{\boldsymbol{K}}(\boldsymbol{S}) \in \mathcal{P}_d$ is an SPD matrix.

The inverse projection of $\widetilde{\boldsymbol{K}} \in \mathcal{P}_d$ to the tangent space is given by the following logarithmic map:

$$\mathrm{Log}_{\boldsymbol{K}}(\widetilde{\boldsymbol{K}}) = \boldsymbol{K}^{1/2}\log\left(\boldsymbol{K}^{-1/2}\widetilde{\boldsymbol{K}}\boldsymbol{K}^{-1/2}\right)\boldsymbol{K}^{1/2} \tag{9}$$

where the result $\mathrm{Log}_{\boldsymbol{K}}(\widetilde{\boldsymbol{K}}) \in \mathcal{T}_{\boldsymbol{K}}\mathcal{P}_d$ is a symmetric matrix in the tangent space.

Further details on the SPD manifold are provided in (Bhatia, 2009; Pennec et al., 2006).

A.2. Riemannian Manifold of SPSD Matrices

To mitigate the requirement for full rank SPD matrices, several Riemannian geometries have been proposed for symmetric positive semi-definite matrices (SPSD) (Bonnabel & Sepulchre, 2010; Massart & Absil, 2020; Vandereycken et al., 2009; 2013). We focus on the one proposed in (Bonnabel & Sepulchre, 2010), which generalizes the affine-invariant geometry (presented in Appendix A.1), forming the basis of our method. This SPSD geometry coincides with the SPD affine-invariant metric, when restricted to SPD matrices.

Let $\mathcal{S}^+_{d,k}$ denote the set of SPSD matrices of size $d \times d$ and fixed rank $k < d$. Any $\boldsymbol{K} \in \mathcal{S}^+_{d,k}$ can be represented by $\boldsymbol{K} = \boldsymbol{G}\boldsymbol{P}\boldsymbol{G}^T$, where $\boldsymbol{P} \in \mathcal{P}_k$ is a $k\times k$ SPD matrix, $\boldsymbol{G} \in \mathcal{V}_{d,k}$, and $\mathcal{V}_{d,k}$ denotes the set of $d\times k$ matrices with orthonormal columns. This representation of $\boldsymbol{K}$ can be obtained by its eigenvalue decomposition for example. This representation implies that SPSD matrices can be represented by the pair $(\boldsymbol{G}, \boldsymbol{P})$, which is termed the structure space representation. Note that the structure space representation is unique up to orthogonal transformations, $\boldsymbol{O} \in \mathcal{O}_k$, i.e., $\boldsymbol{K} \cong \left(\boldsymbol{G}\boldsymbol{O}, \boldsymbol{O}^T\boldsymbol{P}\boldsymbol{O}\right)$. It follows that the space $\mathcal{S}^+_{d,k}$ has a quotient manifold representation, $\mathcal{S}^+_{d,k} \cong \left(\mathcal{V}_{d,k}\times\mathcal{P}_k\right)/\mathcal{O}_k$. The structure space representation pair is thus composed of SPD matrices $\boldsymbol{P} \in \mathcal{P}_k$, whose space forms a Riemannian manifold with the affine-invariant metric (6), and matrices $\boldsymbol{G} \in \mathcal{G}_{d,k}$, where $\mathcal{G}_{d,k}$ denotes the set of k-dimensional subspaces of $\mathbb{R}^d$. The set $\mathcal{G}_{d,k}$ forms the Grassmann manifold with an appropriate inner product on its tangent space $\mathcal{T}_{\boldsymbol{G}}\mathcal{G}_{d,k} = \{\boldsymbol{\Delta} = \boldsymbol{G}_\perp\boldsymbol{B} \mid \boldsymbol{B} \in \mathbb{R}^{(d-k)\times k}\}$, given by $\langle \boldsymbol{\Delta}_1, \boldsymbol{\Delta}_2 \rangle_{\boldsymbol{G}} = \langle \boldsymbol{B}_1, \boldsymbol{B}_2 \rangle$, where $\boldsymbol{G}_\perp \in \mathcal{V}_{d,d-k}$ is the orthogonal complement of $\boldsymbol{G}$. To define the geodesic path between two points, $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$, on the Grassmann manifold, let $\boldsymbol{G}_2^T\boldsymbol{G}_1 = \boldsymbol{O}_2\boldsymbol{\Sigma}\boldsymbol{O}_1^T$ denote the singular value decomposition (SVD), where $\boldsymbol{O}_1, \boldsymbol{O}_2 \in \mathbb{R}^{k\times k}$, $\boldsymbol{\Sigma}$ is a diagonal matrix with $\sigma_i = \cos\theta_i$ on its diagonal, and θ_i denote the principal angles

between the two subspaces represented by $\boldsymbol{G}_1$ and $\boldsymbol{G}_2$. Assuming $\max_i \theta_i \leq \pi/2$, the closed-form for the geodesic path is then given by:

$$\gamma^{\mathcal{G}}_{\boldsymbol{G}_1 \to \boldsymbol{G}_2}(t) = \boldsymbol{G}_1 \boldsymbol{O}_1 \cos(\boldsymbol{\Theta} t) + \boldsymbol{X} \sin(\boldsymbol{\Theta} t) \tag{10}$$

where $\boldsymbol{\Theta} = \operatorname{diag}(\theta_1, \ldots, \theta_k)$, $\theta_i = \arccos \sigma_i$, and $\boldsymbol{X} = \left(\boldsymbol{I} - \boldsymbol{G}_1 \boldsymbol{G}_1^T\right) \boldsymbol{G}_2 \boldsymbol{O}_2 (\sin \boldsymbol{\Theta})^{\dagger}$, where $()^{\dagger}$ denotes the pseudo-inverse.

Following the structure space representation of $\mathcal{S}^+_{d,k}$, its tangent space is defined in (Bonnabel & Sepulchre, 2010) by $\mathcal{T}_{(\boldsymbol{G},\boldsymbol{P})}\mathcal{S}^+_{d,k} = \{(\boldsymbol{\Delta}, \boldsymbol{S}) : \boldsymbol{\Delta} \in \mathcal{T}_{\boldsymbol{G}}\mathcal{G}_{d,k}, \boldsymbol{S} \in \mathcal{T}_{\boldsymbol{P}}\mathcal{P}_k\}$, and the inner product on the tangent space is given by the sum of the inner products on the two components:

$$\langle (\boldsymbol{\Delta}_1, \boldsymbol{S}_1), (\boldsymbol{\Delta}_2, \boldsymbol{S}_2) \rangle_{(\boldsymbol{G},\boldsymbol{P})} = \langle \boldsymbol{\Delta}_1, \boldsymbol{\Delta}_2 \rangle_{\boldsymbol{G}} + m \langle \boldsymbol{S}_1, \boldsymbol{S}_2 \rangle_{\boldsymbol{P}} \tag{11}$$

for $m > 0$, where $(\boldsymbol{\Delta}_\ell, \boldsymbol{S}_\ell) \in \mathcal{T}_{(\boldsymbol{G},\boldsymbol{P})}\mathcal{S}^+_{d,k}$, and $\langle \boldsymbol{S}_1, \boldsymbol{S}_2 \rangle_{\boldsymbol{P}}$ is defined as in (6). There is no closed-form expression for the geodesic path connecting two points on $\mathcal{S}^+_{d,k}$, however, the following approximation is proposed in (Bonnabel & Sepulchre, 2010):

$$\tilde{\gamma}_{\boldsymbol{K}_1 \to \boldsymbol{K}_2}(t) = \gamma^{\mathcal{G}}_{\boldsymbol{G}_1 \to \boldsymbol{G}_2}(t) \gamma^{\mathcal{P}}_{\boldsymbol{P}_1 \to \boldsymbol{P}_2}(t) (\gamma^{\mathcal{G}}_{\boldsymbol{G}_1 \to \boldsymbol{G}_2}(t))^T \tag{12}$$

where $\boldsymbol{K}_\ell \cong (\boldsymbol{G}_\ell, \boldsymbol{P}_\ell)$, $\boldsymbol{K}_\ell \in \mathcal{S}^+_{d,k}$, $\boldsymbol{P}_\ell = \boldsymbol{O}_\ell^T \boldsymbol{G}_\ell^T \boldsymbol{K}_\ell \boldsymbol{G}_\ell \boldsymbol{O}_\ell$ due to the non-uniqueness of the decomposition (up to orthogonal transformations), $\gamma^{\mathcal{P}}_{\boldsymbol{P}_1 \to \boldsymbol{P}_2}(t)$ is defined by (7) and $\gamma^{\mathcal{G}}_{\boldsymbol{G}_1 \to \boldsymbol{G}_2}(t)$ is defined by (10).

B. Difference Operator for SPSD Matrices

Following (Shnitzer et al., 2022), and based on the approximation of the geodesic path in (12), we define the mean and difference operators for two SPSD matrices, $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$, whose structure space representation is given by $\boldsymbol{K}_\ell \cong (\boldsymbol{G}_\ell, \boldsymbol{P}_\ell)$ where $\boldsymbol{G}_\ell \in \mathcal{V}_{d,k}$ and $\boldsymbol{P}_\ell \in \mathcal{P}_k$. Define $\boldsymbol{G}_2^T \boldsymbol{G}_1 = \boldsymbol{O}_2 \boldsymbol{\Sigma} \boldsymbol{O}_1^T$ as the SVD of $\boldsymbol{G}_2^T \boldsymbol{G}_1$ and set $\boldsymbol{P}_\ell = \boldsymbol{O}_\ell^T \boldsymbol{G}_\ell^T \boldsymbol{K}_\ell \boldsymbol{G}_\ell \boldsymbol{O}_\ell$, $\ell = 1, 2$. The mean operator is then defined analogously to (2) as the mid-point of $\tilde{\gamma}_{\boldsymbol{K}_1 \to \boldsymbol{K}_2}(t)$:

$$\widetilde{\boldsymbol{M}} = \tilde{\gamma}_{\boldsymbol{K}_1 \to \boldsymbol{K}_2}(0.5) = \gamma^{\mathcal{G}}_{\boldsymbol{G}_1 \to \boldsymbol{G}_2}(0.5) \gamma^{\mathcal{P}}_{\boldsymbol{P}_1 \to \boldsymbol{P}_2}(0.5) (\gamma^{\mathcal{G}}_{\boldsymbol{G}_1 \to \boldsymbol{G}_2}(0.5))^T \tag{13}$$

Denote the structure space representation of the mean operator by $\widetilde{\boldsymbol{M}} \cong (\boldsymbol{G}_{\boldsymbol{M}}, \boldsymbol{P}_{\boldsymbol{M}})$ and define $\boldsymbol{G}_1^T \boldsymbol{G}_{\boldsymbol{M}} = \widetilde{\boldsymbol{O}}_1 \widetilde{\boldsymbol{\Sigma}} \boldsymbol{O}_{\boldsymbol{M}}^T$ as the SVD of $\boldsymbol{G}_1^T \boldsymbol{G}_{\boldsymbol{M}}$. Set $\boldsymbol{P}_{\boldsymbol{M}} = \boldsymbol{O}_{\boldsymbol{M}}^T \boldsymbol{G}_{\boldsymbol{M}}^T \widetilde{\boldsymbol{M}} \boldsymbol{G}_{\boldsymbol{M}} \boldsymbol{O}_{\boldsymbol{M}}$, and $\widetilde{\boldsymbol{P}}_1 = \widetilde{\boldsymbol{O}}_1^T \boldsymbol{G}_1^T \boldsymbol{K}_1 \boldsymbol{G}_1 \widetilde{\boldsymbol{O}}_1$. The SPSD difference operator is defined by:

$$\widetilde{\boldsymbol{D}} = \gamma^{\mathcal{G}}_{\boldsymbol{G}_{\boldsymbol{M}} \to \boldsymbol{G}_1}(1) \operatorname{Log}_{\boldsymbol{P}_{\boldsymbol{M}}}(\widetilde{\boldsymbol{P}}_1) (\gamma^{\mathcal{G}}_{\boldsymbol{G}_{\boldsymbol{M}} \to \boldsymbol{G}_1}(1))^T \tag{14}$$

where the logarithmic map on the SPD manifold is defined in (9) and the geodesic path on the Grassmann manifold is defined in (10).

The computation of $\widetilde{\boldsymbol{M}}$ and $\widetilde{\boldsymbol{D}}$, as well as the resulting ManiFeSt score for the SPSD case, are summarized in Algorithm 2.

C. Multi-class ManiFeSt Extension

In the main body of the paper we have focused on binary classification problems, since it is the standard stepping-stone for multi-class feature selection (Izetta et al., 2017). We now detail a natural geometric extension of our approach to the multi-class setting, along with a demonstration on MNIST. This extension generally follows the same steps as in Algorithm 1, where we first compute the geometric mean of the kernels, representing the feature space of each class. Second, we compute the difference operators between each class-kernel and the mean, and third, the overall score for each feature is constructed by aggregating the feature scores of the difference operators from all classes.

Concretely, to compute the ManiFeSt score for multi-class datasets, we first construct a kernel $\boldsymbol{K}_\ell \in \mathbb{R}^{d \times d}$, $\ell = 1, \ldots, C$, for samples from each class according to (1), where C is the number of classes in the dataset and d denotes the number of features. The geometric mean of all kernels is then computed according to $\boldsymbol{M} = \arg\min_{\boldsymbol{K}} \sum_{\ell=1}^{C} \left\| \log\left(\boldsymbol{K}^{-1} \boldsymbol{K}_\ell\right) \right\|_F^2$ (Moakher, 2005) using an iterative procedure proposed in (Barachant et al., 2013, Algorithm 1). Note that such an iterative algorithm is required in the multi-class setting since only the geometric mean of *two* SPD matrices has a closed form expression, as in (2). We then compute the differences between the mean and the kernel of each class according to $\boldsymbol{D}_\ell = \operatorname{Log}_{\boldsymbol{M}}(\boldsymbol{K}_\ell)$ from

Algorithm 2 ManiFeSt Score for SPSD Matrices

Input: Dataset with two classes $\boldsymbol{X}^{(1)}$ and $\boldsymbol{X}^{(2)}$
Output: FS score $\boldsymbol{r}$ (SPSD case)

Construct kernels $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ for the two datasets ▷ According to (1)
Set $k = \min\{\text{rank}(\boldsymbol{K}_1), \text{rank}(\boldsymbol{K}_2)\}$
Define $\boldsymbol{K}_\ell = \boldsymbol{G}_\ell \boldsymbol{P}_\ell \boldsymbol{G}_\ell$ and $\boldsymbol{P}_\ell = \boldsymbol{O}_\ell^T \boldsymbol{G}_\ell^T \boldsymbol{K}_\ell \boldsymbol{G}_\ell \boldsymbol{O}_\ell$ where $\boldsymbol{G}_2^T \boldsymbol{G}_1 = \boldsymbol{O}_2 \boldsymbol{\Sigma} \boldsymbol{O}_1$
Compute $\gamma^{\mathcal{G}}_{\boldsymbol{G}_1 \to \boldsymbol{G}_2}(0.5)$ ▷ According to (10)
Compute $\gamma^{\mathcal{P}}_{\boldsymbol{P}_1 \to \boldsymbol{P}_2}(0.5)$ ▷ According to (7)
Build the mean operator $\widetilde{\boldsymbol{M}}$ ▷ According to (13)
Define $\widetilde{\boldsymbol{M}} = \boldsymbol{G}_{\boldsymbol{M}} \boldsymbol{P}_{\boldsymbol{M}} \boldsymbol{G}_{\boldsymbol{M}}$, $\boldsymbol{P}_{\boldsymbol{M}} = \boldsymbol{O}_{\boldsymbol{M}}^T \boldsymbol{G}_{\boldsymbol{M}}^T \boldsymbol{M} \boldsymbol{G}_{\boldsymbol{M}} \boldsymbol{O}_{\boldsymbol{M}}$ and $\widetilde{\boldsymbol{P}}_1 = \widetilde{\boldsymbol{O}}_1^T \boldsymbol{G}_1^T \boldsymbol{K}_1 \boldsymbol{G}_1 \widetilde{\boldsymbol{O}}_1$ where $\boldsymbol{G}_1^T \boldsymbol{G}_{\boldsymbol{M}} = \widetilde{\boldsymbol{O}}_1 \widetilde{\boldsymbol{\Sigma}} \boldsymbol{O}_{\boldsymbol{M}}$
Compute $\text{Log}_{\boldsymbol{P}_{\boldsymbol{M}}}(\widetilde{\boldsymbol{P}}_1)$ ▷ According to (9)
Compute $\gamma^{\mathcal{G}}_{\boldsymbol{G}_{\boldsymbol{M}} \to \boldsymbol{G}_1}(1)$ ▷ According to (10)
Build the difference operator $\widetilde{\boldsymbol{D}}$ ▷ According to (14)
Apply eigenvalue decomposition to $\widetilde{\boldsymbol{D}}$ and compute the FS score $\boldsymbol{r}$ ▷ According to (4)

(3). The jth feature score is defined by $\boldsymbol{r}(j) = \max\{\boldsymbol{r}_1(j), \ldots, \boldsymbol{r}_C(j)\}$, where $\boldsymbol{r}_\ell = \sum_{i=1}^{d} |\lambda_i^{(\boldsymbol{D}_\ell)}| \cdot (\boldsymbol{\phi}_i^{(\boldsymbol{D}_\ell)} \odot \boldsymbol{\phi}_i^{(\boldsymbol{D}_\ell)})$, $\ell = 1, \ldots, C$.

In the multi-class scenario, the scores could be merged using summation instead of the maximum. We note that using the maximum over the feature scores of the different classes highlights class-specific features, whereas summation emphasizes discriminative features that are shared across classes. Both approaches have their own strengths and weaknesses, and it is indeed important to evaluate the performance of each method in order to determine the most appropriate option for a particular task.

An equivalent framework for SPSD matrices can be defined based on the geometric mean proposed in (Yair et al., 2020) and the difference operator presented in Appendix B.

We demonstrate our multi-class FS algorithm on MNIST. To highlight the few-sample capabilities of our algorithm, we use only 300 samples from each class (digit) for computing the FS score, resulting in a total of $N = 3000$ samples. We apply multi-class ManiFeSt to the 3000 samples and choose a subset of the features based on its score, $\boldsymbol{r}$. Fig. 8 presents examples of (left) the data, (middle-left) the leading eigenvectors of the geometric mean of all the class (digit) kernels, $\boldsymbol{M}$, (middle-right) the leading eigenvectors of the difference operators between each class kernel and the mean, and (right) the final multi-class ManiFeSt score. The two right-most plots are overlaid with orange and red circles, which mark the highest ranked 20 and 50 features, respectively, according to the final ManiFeSt score. This figure depicts that the eigenvectors of the geometric mean capture the general region in which all digits typically appear, whereas the leading eigenvectors of the difference operators from the different classes capture digit-specific properties, e.g., the highlighted center of the digit '0', and the highlighted bottom-left edge of '2'. The final ManiFeSt score aggregates this information and highlights the most influential features (pixels) from each class.

For a quantitative comparison with other FS methods, we take a subset of $k = \{20, 50\}$ features with the highest scores from each FS method and optimize an SVM classifier on the same 3000 samples used for computing the FS scores. The test set comprised of 57000 additional images, and the train-test split was repeated 10 times with random shuffling. More details on the hyperparameter tuning and cross-validation appear in Appendix D.1. In Fig. 9, we present the comparison of several FS methods along with a baseline in which the feature subset was chosen randomly (denoted by 'Random'). Each image in this figure presents the feature (pixel) scores obtained by each of the compared FS methods, along with orange and red circles, denoting the highest ranked 20 and 50 features, respectively. Below each image we report the average test accuracy obtained when training the SVM classifier on the top 20 (orange) or 50 (red) highlighted features. Note that ManiFeSt results in the highest test accuracy in both cases, indicating that it indeed captures the most discriminative features in the multi-class setting as well.

As additional motivation for our geometric approach to extending ManiFeSt to a multi-class setting, we considered a naive one vs. one extension, where we aggregated feature scores (taking the maximum) from pairwise comparisons of the different

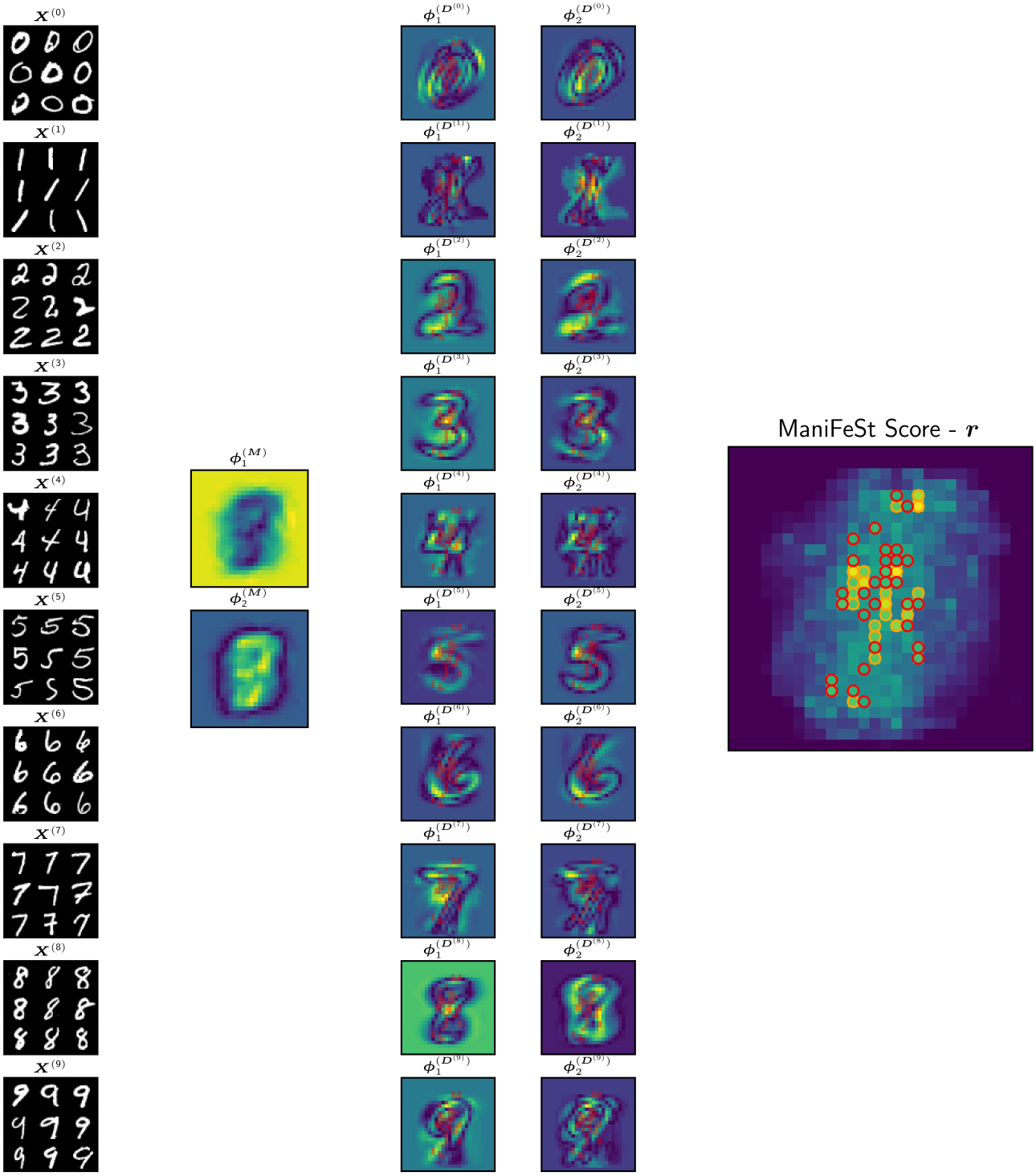

Figure 8. Mulit-class ManiFeSt scheme and the resulting score for images from the MNIST dataset.

digits using Algorithm 1. This resulted in lower average test accuracies of 70.41% (20 features) and 85.49% (50 features), indicating the advantage of our geometric extension.

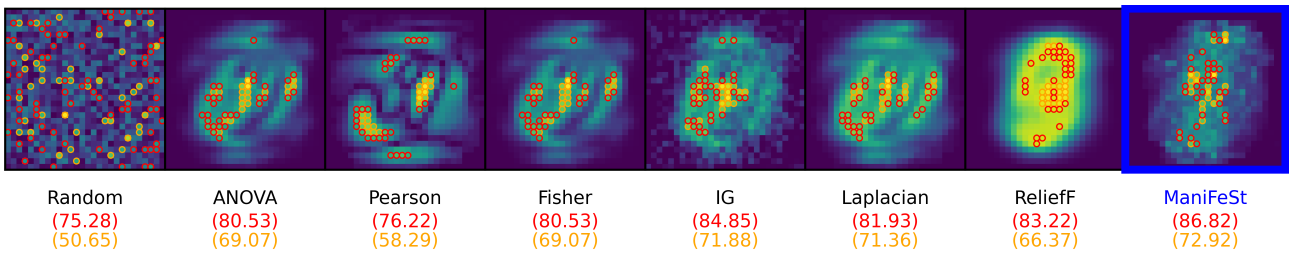

Figure 9. Mulit-class ManiFeSt scheme compared to competing methods on the MNIST dataset. The orange and red circles denote the best ranked 20 and 50 features, respectively. The numbers in parenthesis denote the classification accuracy obtained using the best ranked 20 (orange) and 50 (red) features.

To complete the comparison to related work, we compare our results with DiSC (Sristi et al., 2022). Since DiSC does not provide a score measure for individual features and only highlights discriminative *groups* of features, a feature subset of fixed size cannot be obtained from it. Therefore, to compare our results with DiSC, we report the number of features required to obtain a similar accuracy. In this dataset, DiSC obtains an accuracy of 89.75%, however, it uses $400-500$ features from each digit class to compute the meta-features for the classification, resulting in a total of approximately 700 different features. In contrast, multi-class ManiFeSt achieves an accuracy of 90.71% when using only 100 features.

D. Experiments – More Details and Additional Results

D.1. Implementation Details.

In all the experiments, the data is split to train and test sets with nested cross-validation. The nested train set is divided using 10-fold cross-validation for all datasets. This whole procedure is repeated with shuffled samples for small datasets for better tuning. The data normalization, FS, and SVM hyperparameters tuning are applied to the train set to prevent test-train leakage.

The optimization of SVM hyperparameters is performed over validation sets to improve the generalization of the model, while the hyperparameters of the FS methods are selected to achieve the highest accuracy on the training set. This approach is adopted to avoid the optimization of both SVM and FS hyperparameters simultaneously.

The process of FS hyperparameter tuning depends on the selected number of features. In the results in Table 1, we tune the hyperparameters using the training set for each number of selected features. The highest resulting test accuracy is then presented for the number of features that yielded the best results. In the rest of the paper, we first tune the hyperparameters for different number of features (from a regular grid). Then, we fix the hyperparameters corresponding to the number of features that yielded the highest train accuracy. Last, we use this fixed set of hyperparameters to demonstrate the behavior of the score across various feature subsets.

Data normalization. Following (Atashgahi et al., 2022), the features in the Madelon dataset are normalized by removing the mean and rescaling to unit variance. This is implemented using the standard sklearn function. The other datasets do not require normalization.

Kernel type, scale, and distance metric. The choice of the kernel type and scale as well as the distance metric can significantly impact the performance of our method and is typically task-specific. We postulate that these choices are common to almost all kernel and manifold learning methods. They do not have a definitive solution and still attract research, e.g., (Lindenbaum et al., 2020).

Our method is not limited to RBF kernels, and other symmetric positive semi-definite (SPSD) kernels could be used instead. However, in our experiments, we choose to use the Gaussian kernel with the Euclidean metric due to its simplicity and popularity in manifold learning and classification tasks. Our aim was to emphasize the contribution of our approach rather than the kernel type selection, which is important but common to all kernel and manifold learning methods. For the choice of kernel scale, we used cross-validation to estimate the optimal bandwidth, as outlined in Appendix D.1.

We remark that there may be multiple scales for each task, and in our future work, we plan to investigate the use of multi-scale ManiFeSt to capture additional information in different scales.

FS hyperparameter tuning. Both IG and ReliefF FS methods have a number of nearest neighbors parameter, since the extension of the classic IG score from discrete to continuous features makes use of the k nearest neighbors (Ross, 2014). We tune the number of neighbors for IG and ReliefF over the grid $k = \{1, 3, 5, 10, 15, 20, 30, 50, 100\}$. For the Laplacian score, the samples' kernel scale is tuned to the i_{th} percentile of Euclidean distances over the grid $i = \{1, 5, 10, 30, 50, 70, 90, 95, 99\}$. ManiFeSt only requires tuning of the features' kernel scale. For the illustrative example, the scale σ_ℓ is set to the median of Euclidean distances, for best visualization. For the XOR and Madelon problems, the scale is set to the median of Euclidean distances multiplied by a factor 0.1. Since the multi-feature associations of the relevant features are distinct in these two datasets, no scale tuning was required. Conversely, for the remaining datasets, the scale factor is tuned to the i_{th} percentile of the Euclidean distances over the grid $i = \{5, 10, 30, 50, 70, 90, 95\}$. The optimization of the hyperparameters of the combination of ReliefF and ManiFeSt is achieved through a single-variable optimization approach. Specifically, the optimal parameter for ReliefF is selected based on its individual performance and subsequently, the ManiFeSt algorithm is tuned to attain the highest combination score on the training set. The scale factor of ManiFeSt is fine-tuned by including two additional grid points of the i^{th} percentile of Euclidean distances, specifically $i = \{1, 99\}$, to effectively capture multivariate relations that are more prominent in such regions and are not captured by traditional filter methods.

SVM hyperparameter tuning. When the ground-truth of which features are relevant is not available, we apply an SVM classifier to the selected subset of features in order to evaluate the FS. For the SVM hyperparameter tuning, we follow (Hsu et al., 2003). We use an RBF kernel and perform a grid search on the penalty parameter C and the kernel scale γ. C and γ are tuned over exponentially growing sequences, $C = \{2^{-5}, 2^{-2}, 2^{1}, 2^{4}, 2^{7}, 2^{10}, 2^{13}\}$ and $\gamma = \{2^{-15}, 2^{-12}, 2^{-9}, 2^{-6}, 2^{-3}, 2^{0}, 2^{3}\}$.

Computing resources. All the experiments were performed using Python on a standard PC with an Intel i7-12700 CPU and 64GB of RAM without GPUs. We note that according to a recent work (Fawzi & Goulbourne, 2021), using GPUs could allow a faster computation of the eigenvalue decomposition required by ManiFeSt.

FS source code. The competing methods were implemented as follows. The IG (Vergara & Estévez, 2014) and ANOVA (Kao & Green, 2008) methods were computed using the scikit-learn package. For Gini-index (Shang et al., 2007), t-test (Davis & Sampson, 1986), Fisher (Duda et al., 2006), Laplacian (He et al., 2005), and ReliefF (Robnik-Šikonja & Kononenko, 2003), we use the skfeature repository developed by the Arizona State University (Li et al., 2018). The Pearson correlation (Battiti, 1994) is implemented by the built-in Panda package correlation.

The code implementing ManiFeSt, along with the script for reproducing the illustrative example, is available on GitHub[1].

Details on the hypercube dataset. For the experiment in Subsection 5.3, we create a 10-dimensional hypercube embedded in $\mathbb{R}^{10}$. Then, 2000 points are generated and grouped into 4 clusters. The data in each cluster are normally distributed and centered at one of vertices of the hypercube. We define two classes, where each class consists of two clusters. The partition of the 4 clusters to the two classes is performed in an arbitrarily manner.

To create the dataset, these 10-dimensional points are mapped to $\mathbb{R}^{200}$ by appending coordinates with random noise, so that each point in the dataset consists of 200 features out of which only 10 are relevant.

Details on Gisette and Prostate datasets. The Gisette dataset (Guyon et al., 2008) is a synthetic dataset from the NIPS 2003 feature selection challenge. The dataset contains 7000 samples consisting of 5000 features, of which only 2500 are relevant features. The classification problem aims to discriminate between the digits 4 and 9, which were mapped into a high dimensional feature space. For more details, see (Guyon, 2003).

The Prostate cancer dataset (Singh et al., 2002) consists of the expression levels of 5966 genes (features) and 102 samples, of which 50 are normal and 52 are tumor samples.

[1] https://github.com/DavidCohen2/ManiFeSt

In both datasets, the samples are split into 90% train and 10% test sets. Results are averaged over 10 and 30 cross-validation iterations for Gisette and Prostate, respectively.

D.2. Additional Results

D.2.1. Fashion-MNIST: another illustrative example

We use the Fashion-MNIST dataset (Xiao et al., 2017) for illustration. We generate two sets: one consists of 1500 images of pants and the other consists of 1500 images of shirts. Fig. 10 is the same as Fig. 2, presenting the results obtained by ManiFeSt for this example.

In Fig. 10(middle), we see that the leading eigenvectors of the composite difference kernel, $\boldsymbol{D}$, indeed capture the main conceptual differences between the two clothes. These differences include the gap between the pants' legs, the gap between the shirts' sleeves, and the shirt collar. As shown in Fig. 10(right), the ManiFeSt score, which weighs the eigenvectors by their respective eigenvalues, provides a consolidated measure of the discriminative pixels.

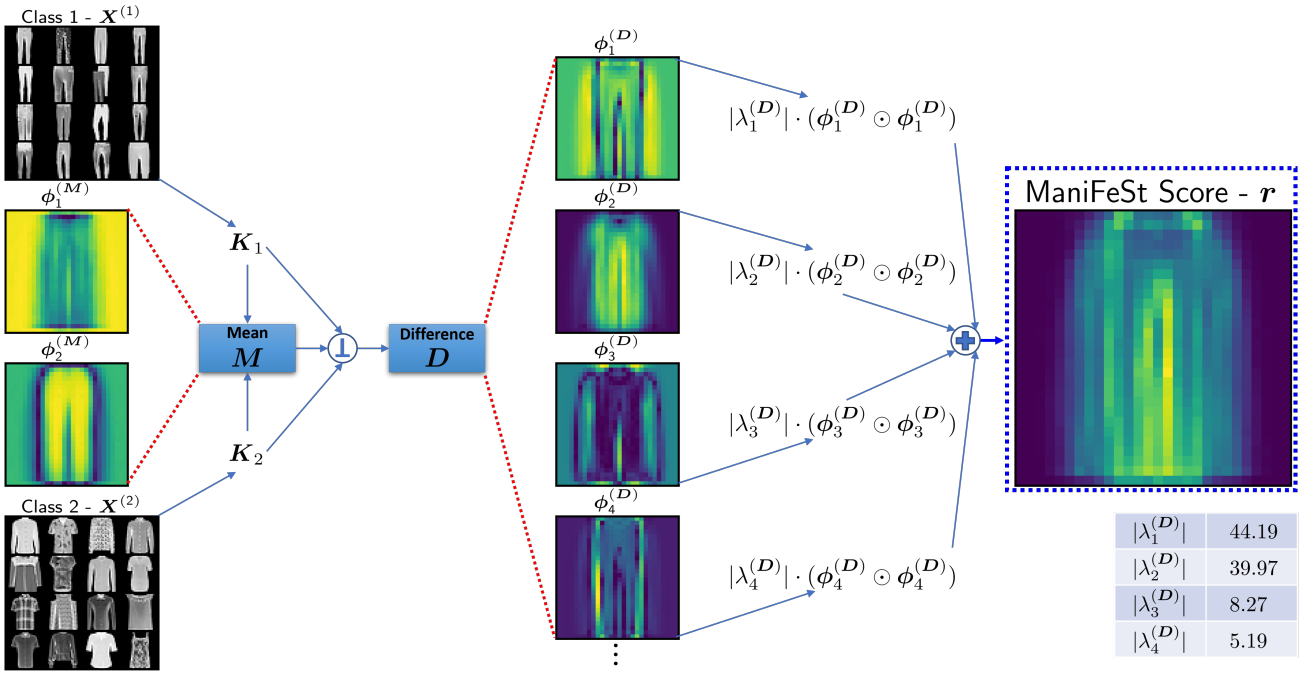

Figure 10. Ilustration of the proposed scheme and the resulting ManiFeSt score for images from the Fashion-MNIST dataset.

D.2.2. Colon cancer gene expression: additional results

More generalization tests. To further demonstrate the generalization capabilities of ManiFeSt, we examine the effect of the kernel scale, i.e., σ_ℓ in Eq. (1), on the results of the colon dataset. In Fig. 11, we present the generalization error obtained by ManiFeSt for three different kernel scales. We see that the larger the scale is, i.e., the more feature associations are captured by the kernel, the smaller the generalization error becomes. This result indicates that the multi-feature associations taken into account by ManiFeSt play a central role in its favorable generalization capabilities.

We conclude this section on generalization with a possible direction for future investigation. We speculate that the large generalization error demonstrated in Fig. 7 by the competing methods may suggest the presence of significant batch effects in the colon dataset, which is prototypical to such biological data (Lazar et al., 2013). In contrast, the smaller generalization error achieved by ManiFeSt could indicate its robustness to batch effects. Therefore, the robustness of ManiFeSt to batch effects will be studied in future work.

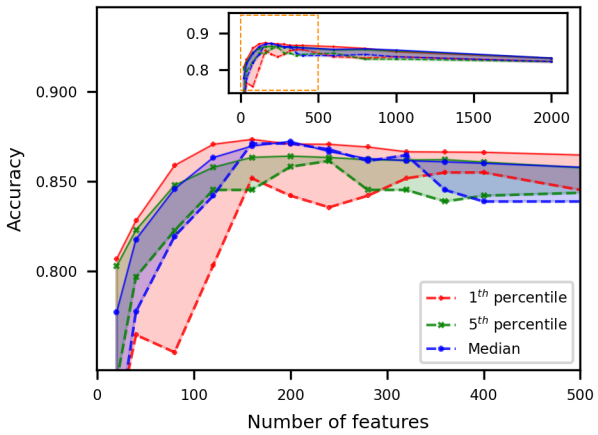

Figure 11. ManiFeSt with three different kernel scales. The dashed and solid lines represent the average test and validation accuracy, and the shaded area represents the validation-test generalization error.

Table 2. ManiFeSt Comparison with Embedded Methods

Method	Accuracy $\pm$ STD
LASSO	81.54 ± 9.85
SVC	76.15 ± 9.39
RF	79.23 ± 9.76
XGBoost	76.15 ± 12.14
MLP	81.54 ± 7.84
Linear STG	74.62 ± 11.44
Nonlinear STG	76.15 ± 13.95
INVASE	76.92 ± 12.40
L2X	78.46 ± 8.28
TabNet	64.62 ± 12.02
REAL-x	75.38 ± 12.78
LSPIN	71.54 ± 6.92
LLSPIN	83.85 ± 5.38
ManiFeSt	$\mathbf{85.38} \pm 8.60$

Comparison with embedded methods. To complement the experimental study, we report here recent results on the colon dataset (Yang et al., 2022) obtained by various embedded methods. These results are displayed in Table 2 along with our result obtained by ManiFeSt. For a fair comparison, in this experiment we use the same train-test split (49/13) and average the results over 50 cross-validation iterations. We see in the table that ManiFeSt achieves the best mean accuracy, but with a larger standard deviation, compared with the leading competing embedded method (LLSPIN proposed in (Yang et al., 2022)).

D.2.3. Toy example: limitations of ManiFeSt

ManiFeSt considers multivariate associations rather than univariate properties. In some scenarios, this might lead to the selection of irrelevant features or the misselection of relevant features.

We demonstrate this limitation using a toy example. We simulate data consisting of $d = 20$ binary features and $N = 500$ instances. Each feature is sampled from a Bernoulli distribution. We consider two cases. In the first case, each instance is associated with a label that is equal to the first feature f_1, where f_i denotes the ith feature. Accordingly, only f_1 is a relevant for the classification of the label. In the second case, we set $f_5 = 0$ to be a fixed constant, while the label is still determined by f_1.

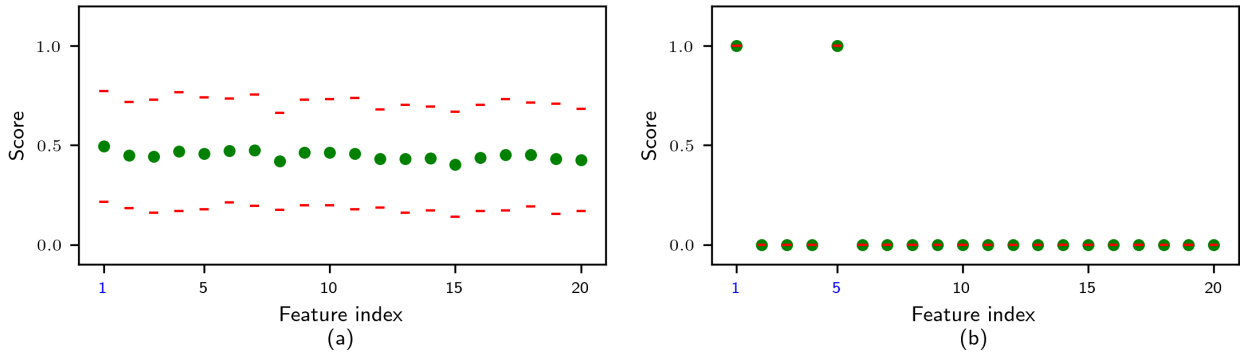

Figure 12. ManiFeSt score on the toy example, averaged over 50 Monte-Carlo iterations of data generation. The green circles denote the average score, and the red lines indicate the standard deviation. (a) The first case. (b) The second case ($f_5 \equiv 0$).

In Fig. 12, we present the normalized feature score obtained by ManiFeSt, averaged over 200 Monte-Carlo iterations of data generation. The green circles denote the average score, and the red dashes indicate the standard deviation.

In Fig. 12(a), we show the results for the first case. We see that ManiFeSt does not identify f_1, since the associations of this feature to the other features are not distinct. More specifically, the differences $f_1 - f_j$ for every j have the same statistics over samples from the two classes. In Fig. 12(b), we show the results for the second case. We see that the relevant feature f_1 is captured, yet the irrelevant feature f_5 is also detected. This is due to the fact that now the association of f_1 with f_5 becomes distinct between the two classes. This result implies that ManiFeSt may identify features without any discriminative capabilities through their associations to other relevant and discriminative features.

Fig. 7(b) suggests that the combination of a classical univariate criterion with the multivariate ManiFeSt score has potential to mitigate this limitation. In future work, we will further investigate the simultaneous utilization of univariate and multivariate properties.

E. Additional Theoretical Foundation

We begin by proving Proposition 1 from Section 4, which provides additional justification for the ManiFeSt score (4). We reiterate the proposition for completeness:

Proposition 1. *Assume that ϕ is a shared eigenvector of $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ with respective eigenvalues $\lambda^{(\boldsymbol{K}_1)}$ and $\lambda^{(\boldsymbol{K}_2)}$. Then ϕ is an eigenvector of $\boldsymbol{D}$ with a corresponding eigenvalue $\lambda^{(\boldsymbol{D})} = \sqrt{\lambda^{(\boldsymbol{K}_1)}\lambda^{(\boldsymbol{K}_2)}}\left(\log\lambda^{(\boldsymbol{K}_1)} - \log\lambda^{(\boldsymbol{K}_2)}\right)$ that satisfies:*

$$\left|\lambda^{(\boldsymbol{D})}\right| \leq 2\sum_{i,j=1}^{d}\left|e^{-\|x_i^{(1)}-x_j^{(1)}\|^2/2\sigma^2} - e^{-\|x_i^{(2)}-x_j^{(2)}\|^2/2\sigma^2}\right||\phi(i)||\phi(j)| \tag{15}$$

where $x_i^{(\ell)}$ and $x_j^{(\ell)}$ are vectors containing the values of features i and j, respectively, from all the samples in class ℓ.

Proof. First, the claim that ϕ is also an eigenvector of $\boldsymbol{D}$ with its corresponding eigenvalue, given by $\lambda^{(\boldsymbol{D})} = \sqrt{\lambda^{(\boldsymbol{K}_1)}\lambda^{(\boldsymbol{K}_2)}}\left(\log\lambda^{(\boldsymbol{K}_1)} - \log\lambda^{(\boldsymbol{K}_2)}\right)$, is proved in (Shnitzer et al., 2022, Theorem 2). Second, to prove the bound for $\lambda^{(\boldsymbol{D})}$, we start from the definition of the eigenvalue decomposition, for $\ell = \{1, 2\}$:

$$\lambda^{(\boldsymbol{K}_\ell)} = \phi^T\boldsymbol{K}_\ell\phi = \sum_{i,j=1}^{d} e^{-\|x_i^{(\ell)}-x_j^{(\ell)}\|^2/2\sigma^2}\phi(i)\phi(j) \tag{16}$$

The difference between the eigenvalues $\lambda^{(\boldsymbol{K}_1)}$ and $\lambda^{(\boldsymbol{K}_2)}$ is then given by:

$$\left|\lambda^{(\boldsymbol{K}_1)} - \lambda^{(\boldsymbol{K}_2)}\right| = \left|\sum_{i,j=1}^{d}\left(e^{-\|x_i^{(1)}-x_j^{(1)}\|^2/2\sigma^2} - e^{-\|x_i^{(2)}-x_j^{(2)}\|^2/2\sigma^2}\right)\phi(i)\phi(j)\right| \tag{17}$$

Assume w.l.o.g that $\lambda^{(\boldsymbol{K}_1)} > \lambda^{(\boldsymbol{K}_2)}$. Then, we have

$$|\log\lambda^{(\boldsymbol{K}_1)} - \log\lambda^{(\boldsymbol{K}_2)}| \leq 2\frac{\left|\sqrt{\lambda^{(\boldsymbol{K}_1)}} - \sqrt{\lambda^{(\boldsymbol{K}_2)}}\right|}{\sqrt{\lambda^{(\boldsymbol{K}_2)}}}$$

since $\lambda^{(\boldsymbol{K}_1)}, \lambda^{(\boldsymbol{K}_2)} > 0$, $\log(x) - \log(y) = \log(x/y)$ and $0 \leq \log(x) \leq 2(\sqrt{x} - 1)$ for $x \geq 1$. Multiplying both sides by $\sqrt{\lambda^{(\boldsymbol{K}_1)}\lambda^{(\boldsymbol{K}_2)}}$ gives the following inequality

$$\left|\sqrt{\lambda^{(\boldsymbol{K}_1)}\lambda^{(\boldsymbol{K}_2)}}\left(\log\lambda^{(\boldsymbol{K}_1)} - \log\lambda^{(\boldsymbol{K}_2)}\right)\right| \leq 2\left|\lambda^{(\boldsymbol{K}_1)} - \sqrt{\lambda^{(\boldsymbol{K}_1)}\lambda^{(\boldsymbol{K}_2)}}\right| \leq 2\left|\lambda^{(\boldsymbol{K}_1)} - \lambda^{(\boldsymbol{K}_2)}\right| \tag{18}$$

where the last transition is due to $\sqrt{\lambda^{(\boldsymbol{K}_1)}} > \sqrt{\lambda^{(\boldsymbol{K}_2)}}$, and the left hand side of this equation is equal to $\lambda^{(\boldsymbol{D})}$. Combining (17) and (18) concludes the proof, leading to the following upper bound of the absolute value of $\lambda^{(\boldsymbol{D})}$:

$$\left|\lambda^{(\boldsymbol{D})}\right| \leq 2\sum_{i,j=1}^{d}\left|e^{-\|x_i^{(1)} - x_j^{(1)}\|^2/2\sigma^2} - e^{-\|x_i^{(2)} - x_j^{(2)}\|^2/2\sigma^2}\right||\phi(i)||\phi(j)| \tag{19}$$

□

Note that a similar derivation can be done for eigenvectors of $\boldsymbol{K}_1$ and $\boldsymbol{K}_2$ that are only approximately similar (not identically shared), as stated by the following proposition.

Proposition 2. *Let ϕ denote an eigenvector of $\boldsymbol{K}_1$ with eigenvalue $\lambda^{(\boldsymbol{K}_1)}$ and $\phi^{(2)}$ denote an eigenvector of $\boldsymbol{K}_2$ with eigenvalue $\lambda^{(\boldsymbol{K}_2)}$. Assume that $\phi^{(2)} = \phi + \phi_\epsilon$, where $\|\phi_\epsilon\|_2 < \epsilon$ for some small $\epsilon > 0$. Then ϕ and $\lambda^{(\boldsymbol{D})} = \sqrt{\lambda^{(\boldsymbol{K}_1)}\lambda^{(\boldsymbol{K}_2)}}\left(\log\lambda^{(\boldsymbol{K}_1)} - \log\lambda^{(\boldsymbol{K}_2)}\right)$ are an approximate eigen-pair of $\boldsymbol{D}$, such that $\lambda^{(\boldsymbol{D})}$ satisfies:*

$$\left|\lambda^{(\boldsymbol{D})}\right| \leq 2\sum_{i,j=1}^{d}\left|e^{-\|x_i^{(1)} - x_j^{(1)}\|^2/2\sigma^2} - e^{-\|x_i^{(2)} - x_j^{(2)}\|^2/2\sigma^2}\right||\phi(i)||\phi(j)| + 2a\epsilon^2 \tag{20}$$

where $a = \max_i \sum_j K_2(i,j)$.

Proof. A proof showing that ϕ and $\lambda^{(\boldsymbol{D})}$ are an approximate eigen-pair of $\boldsymbol{D}$ can be found in (Shnitzer et al., 2022, Theorem 4). For the bound on $\lambda^{(\boldsymbol{D})}$, note that equation (16) holds for $\boldsymbol{K}_1$ with no change, whereas for $\boldsymbol{K}_2$ we have:

$$\begin{aligned}\lambda^{(\boldsymbol{K}_2)} &= \left(\phi^{(2)}\right)^T\boldsymbol{K}_2\phi^{(2)} = \phi^T\boldsymbol{K}_2\phi + \phi_\epsilon^T\boldsymbol{K}_2\phi_\epsilon \\ &= \sum_{i,j=1}^{d} e^{-\|x_i^{(2)} - x_j^{(2)}\|^2/2\sigma^2}\phi(i)\phi(j) + \phi_\epsilon^T\boldsymbol{K}_2\phi_\epsilon\end{aligned} \tag{21}$$

The difference between the eigenvalues can then be bounded by:

$$\begin{aligned}\left|\lambda^{(\boldsymbol{K}_1)} - \lambda^{(\boldsymbol{K}_2)}\right| &= \left|\sum_{i,j=1}^{d}\left(e^{-\|x_i^{(1)} - x_j^{(1)}\|^2/2\sigma^2} - e^{-\|x_i^{(2)} - x_j^{(2)}\|^2/2\sigma^2}\right)\phi(i)\phi(j) - \phi_\epsilon^T\boldsymbol{K}_2\phi_\epsilon\right| \\ &\leq \left|\sum_{i,j=1}^{d}\left(e^{-\|x_i^{(1)} - x_j^{(1)}\|^2/2\sigma^2} - e^{-\|x_i^{(2)} - x_j^{(2)}\|^2/2\sigma^2}\right)\phi(i)\phi(j)\right| + \lambda_{\max}^{(\boldsymbol{K}_2)}\epsilon^2 \\ &\leq \left|\sum_{i,j=1}^{d}\left(e^{-\|x_i^{(1)} - x_j^{(1)}\|^2/2\sigma^2} - e^{-\|x_i^{(2)} - x_j^{(2)}\|^2/2\sigma^2}\right)\phi(i)\phi(j)\right| + a\epsilon^2\end{aligned} \tag{22}$$

where $\lambda_{\max}^{(2)}$ is the maximal eigenvalue of $\boldsymbol{K}_2$, which is bounded by $a = \max_i \sum_j K_2(i,j)$ (maximal row sum of $\boldsymbol{K}_2$) from the Perron-Frobenius theorem. Note that since $\boldsymbol{K}_2$ is positive semi-definite with 1s on its diagonal, its maximal eigenvalue

can also be crudely bounded by d, the dimension of the feature space. The rest of the derivation for the approximate case follows from (18), resulting in:

$$\left|\lambda^{(\boldsymbol{D})}\right| \leq 2\sum_{i,j=1}^{d}\left|e^{-\|x_i^{(1)}-x_j^{(1)}\|^2/2\sigma^2}-e^{-\|x_i^{(2)}-x_j^{(2)}\|^2/2\sigma^2}\right||\phi(i)||\phi(j)|+2a\epsilon^2 \tag{23}$$

□

Lastly, we reiterate the proof for Proposition 3 from (Shnitzer et al., 2022, Proposition 3), which states:

Proposition 3. *Assume* $\boldsymbol{K}_2=\boldsymbol{K}_1+\boldsymbol{E}$ *such that* $\|\boldsymbol{E}\boldsymbol{K}_1^{-1}\|<1$*, then* $\boldsymbol{D}\approx-\frac{1}{2}\left(\boldsymbol{K}_2-\boldsymbol{K}_1\right)\left(\boldsymbol{K}_1^{-1}\boldsymbol{K}_2\right)^{1/2}$.

Proof.

$$\begin{aligned}
\boldsymbol{D} &= \log\left(\boldsymbol{K}_1\boldsymbol{M}^{-1}\right)\boldsymbol{M} && (24)\\
&= \log\left(\boldsymbol{K}_1\left((\boldsymbol{K}_2\boldsymbol{K}_1^{-1})^{1/2}\boldsymbol{K}_1\right)^{-1}\right)\left(\boldsymbol{K}_2\boldsymbol{K}_1^{-1}\right)^{1/2}\boldsymbol{K}_1 && (25)\\
&= -\tfrac{1}{2}\log\left(\boldsymbol{K}_2\boldsymbol{K}_1^{-1}\right)\left(\boldsymbol{K}_2\boldsymbol{K}_1^{-1}\right)^{1/2}\boldsymbol{K}_1 && (26)\\
&= -\tfrac{1}{2}\log\left(\boldsymbol{I}+\boldsymbol{E}\boldsymbol{K}_1^{-1}\right)\left(\boldsymbol{K}_2\boldsymbol{K}_1^{-1}\right)^{1/2}\boldsymbol{K}_1 && (27)\\
&\approx -\tfrac{1}{2}\boldsymbol{E}\boldsymbol{K}_1^{-1}\left(\boldsymbol{K}_2\boldsymbol{K}_1^{-1}\right)^{1/2}\boldsymbol{K}_1 && (28)\\
&= -\tfrac{1}{2}\boldsymbol{E}\left(\boldsymbol{K}_1^{-1}\boldsymbol{K}_2\right)^{1/2} && (29)\\
&= -\tfrac{1}{2}\left(\boldsymbol{K}_2-\boldsymbol{K}_1\right)\left(\boldsymbol{K}_1^{-1}\boldsymbol{K}_2\right)^{1/2} && (30)
\end{aligned}$$

where the approximation is given by a power series of the matrix logarithm, and the transition before last is due to $\boldsymbol{K}_2\boldsymbol{K}_1^{-1}$ being similar to a symmetric matrix, and therefore, diagonalizable, and due to $\boldsymbol{X}^{-1}\boldsymbol{A}^{1/2}\boldsymbol{X}=\left(\boldsymbol{X}^{-1}\boldsymbol{A}\boldsymbol{X}\right)^{1/2}$ for a diagonalizable matrix $\boldsymbol{A}$ and an invertible matrix $\boldsymbol{X}$. □

Note that the definitions of $\boldsymbol{M}$ and $\boldsymbol{D}$ in the proof are equivalent to definitions (2) and (3) in the paper, as shown in (Shnitzer et al., 2022, Proposition 1).