

# Nested Subspace Learning with Flags

**Tom Szwagier**

*Université Côte d’Azur and Inria  
Sophia Antipolis, France*

TOM.SZWAGIER@INRIA.FR

**Xavier Pennec**

*Université Côte d’Azur and Inria  
Sophia Antipolis, France*

XAVIER.PENNEC@INRIA.FR

**Editor:** Aapo Hyvärinen

## Abstract

Many machine learning methods look for low-dimensional representations of the data. The underlying subspace can be estimated by first choosing a dimension  $q$  and then optimizing a certain objective function over the space of  $q$ -dimensional subspaces—the Grassmannian. Trying different  $q$  generally yields non-nested subspaces, which raises an important issue of consistency between the data representations. In this paper, we propose a simple and easily implementable principle to enforce nestedness in subspace learning methods. It consists in lifting Grassmannian optimization criteria to flag manifolds—the space of nested subspaces of increasing dimension—via nested projectors. We apply the flag trick to several classical machine learning methods and show that it successfully addresses the nestedness issue.

**Keywords:** subspace learning, Grassmann manifolds, flag manifolds, nested subspaces, dimensionality reduction

## 1. Introduction

Finding low-dimensional representations of data sets is a common objective in machine learning, notably due to the *curse of dimensionality* (Bellman, 1957). A classical way of building such a representation is by searching for a low-dimensional subspace that well represents the data, where “well” is defined by an application-dependent criterion (Cunningham and Ghahramani, 2015). Here is a non-exhaustive list of examples of subspace learning problems. *Principal component analysis* (PCA) (Pearson, 1901; Hotelling, 1933; Jolliffe, 2002) searches for a low-dimensional subspace that minimizes the average squared Euclidean distance to the data. *Robust subspace recovery* (RSR) (Lerman and Maunu, 2018b) minimizes the average (absolute) Euclidean distance to the data, which is less sensitive to outliers. Many matrix decomposition methods, like *robust PCA* (Candès et al., 2011) and *matrix completion* (Keshavan et al., 2010; Candès and Recht, 2012) look for low-rank approximations of the data matrix, which can be decomposed into a product of subspace and coordinate matrices. *Trace ratio* (TR) (Ngo et al., 2012) refers to a wide class of problems that look for subspaces making a tradeoff between desired yet antagonist properties of low-dimensional embeddings; an example is Fisher’s *linear discriminant analysis* (LDA) (Fisher, 1936) which seeks to maximize the between-class variance while minimizing the within-class variance. *Domain adaptation* methods learn some domain-

invariant subspaces (Baktashmotlagh et al., 2013) by minimizing the projected maximum mean discrepancy (MMD) (Gretton et al., 2012) between the source and target distributions. *Subspace tracking* methods incrementally minimize a distance between the current subspace and the available data (Balzano et al., 2010). Subspaces can also be estimated from adjacency matrices or graph Laplacians—rather than directly from the data—as done in the celebrated *Laplacian eigenmaps* (Belkin and Niyogi, 2003) and *spectral clustering* (Ng et al., 2001). Subspaces can also be estimated beyond Euclidean data, for instance on symmetric positive-definite matrix data sets (Harandi et al., 2018). In all the previously cited examples, the search space—the space of all linear subspaces of dimension  $q$  embedded in an ambient space of larger dimension  $p$ —is called the *Grassmannian*, or the *Grassmann manifold* (Bendokat et al., 2024) and is denoted  $\text{Gr}(p, q)$ . Consequently, many machine learning methods can be recast as an optimization problem on Grassmann manifolds.

**Remark 1 (Sequential Methods)** *Alternative methods to subspace learning via Grassmannian optimization exist. Notably, methods such as canonical correlation analysis (CCA) (Hotelling, 1936; Haroon et al., 2004), independent component analysis (ICA) (Hyvärinen and Oja, 2000), partial least squares (PLS) (Geladi and Kowalski, 1986) or projection pursuit (Huber, 1985; Tsakiris and Vidal, 2018) build a low-dimensional subspace out of the data sequentially. They first look for the best 1D approximation and then recursively add dimensions one-by-one via deflation schemes or orthogonality constraints (Lerman and Maunu, 2018b, Section II.C). These sequential methods however suffer from many issues—from a subspace estimation point of view—due to their greedy nature. These limitations are well described in Cunningham and Ghahramani (2015) and Lerman and Maunu (2018b). In light of the fundamental differences between those methods and the previously described subspace methods, we decide not to address sequential methods in this paper.*

Learning a low-dimensional subspace from the data through Grassmannian optimization requires to choose the dimensionality  $q$  *a priori*. This prerequisite has many important limitations, not only since a wrong choice of dimension might remove all the theoretical guarantees of the method (Lerman and Maunu, 2018b), but also because the observed data set might simply not have a favored dimension, above which the added information is worthless.<sup>1</sup> For these reasons, one might be tempted to “try” several dimensions—i.e., run the subspace optimization algorithm with different dimensions and choose the best one *a posteriori*, with cross-validation or statistical model selection techniques for instance. In addition to being potentially costly, such a heuristic raises an important issue that we later refer to as the *nestedness issue*: the optimal subspaces at different dimensions are *not nested* in general. This notably means that the data embeddings at two different dimensions might drastically differ, which is a pitfall for data analysis.

**Remark 2 (Importance of Nestedness)** *The importance of nestedness has been widely documented in the statistical literature: it yields a hierarchy of representations, improving interpretability, dimension selection, computational efficiency and stability (Huber, 1985).*

---

1. Many generative models assume that the data lie on a lower-dimensional subspace, up to isotropic Gaussian noise (Tipping and Bishop, 1999; Lerman et al., 2015). However, anisotropy is much more realistic (Maunu et al., 2019), which nuances the theoretical guarantees and removes hopes for a clear frontier between signal and noise at the intrinsic dimension.

*In geometric statistics, nestedness is considered as a key feature for multilevel analysis (Damon and Marron, 2014). Many works consequently directly impose the nestedness of the approximation subspaces by construction (e.g., via forward or backward sequential methods), in contrast to subspace-based methods which cannot guarantee nestedness. One can notably cite the geodesic versions of PCA (Fletcher et al., 2004; Sommer et al., 2014; Huckemann et al., 2010), the principal nested spheres of Jung et al. (2012), the barycentric subspace analysis of Pennec (2018) and subsequent contributions (Eltzner et al., 2018; Huckemann and Eltzner, 2018; Dryden et al., 2019; Jaquier and Rozo, 2020; Yang and Vemuri, 2021; Fan et al., 2022; Mankovich and Birdal, 2023; Rabenoro and Pennec, 2024; Mankovich et al., 2025; Su and Yao, 2025; Rauba and Schaar, 2026). All these works show the interest of nestedness in statistical inference, with theoretical advantages (e.g., sharper confidence intervals) and meaningful applications. It is also worth noticing that the review paper from Lerman and Maunu (2018b) emphasizes that the lack of nestedness is a fundamental issue in robust subspace recovery, which is left open for future research.*

We illustrate in Figure 1 the nestedness issue on toy data sets related to three important machine learning problems: robust subspace recovery, linear discriminant analysis and sparse spectral clustering (Lu et al., 2016; Wang et al., 2017). We can see that the scatter plots of the projected data for  $q = 1$  and  $q = 2$  are *inconsistent*—in that the 1D representation is not the projection of the 2D representation onto the horizontal axis.

In this paper, we propose a natural solution to the nestedness issue with a geometrical object that has not been much considered yet in machine learning: *flags*—i.e. sequences of nested subspaces of increasing dimension. The idea is to first select a sequence of candidate dimensions—the *signature* of the flag—then formulate a multilevel subspace learning criterion which extends the original one by integrating multiple dimensions simultaneously—with the *flag trick*—and finally optimize over the space of flags of the chosen signature, which has a manifold structure similar to the one of Grassmannians. The output subspaces are naturally nested, which solves the nestedness issue of subspace learning methods and provides the subspaces with a hierarchy that is good for interpretability. Moreover, the nested representations of the data can be fit to general machine learning algorithms and combined via *ensembling* methods. The ensembling weights can then be interpreted as a measure of importance for the different dimensions, echoing the perspectives of Minka (2000). Beyond the change of paradigm (from subspaces to flags of subspaces), the main contribution of the paper is the *flag trick*: a generic method to convert a *fixed-dimension* subspace criterion into a *multilevel* flag criterion, without nestedness issue.

The paper is organized as follows. In Section 2, we provide some prerequisites on flag manifolds and describe a steepest descent optimization algorithm. In Section 3, we introduce the flag trick and propose an ensemble learning algorithm to leverage the hierarchical information of the flag. In Section 4, we show the interest of the flag trick on several subspace learning methods such as robust subspace recovery, trace ratio and spectral clustering. In Section 5, we conclude and discuss the limits and perspectives of such a framework. In Appendix A, B, C, we show how the flag trick can be used to develop two advanced optimization algorithms on flag manifolds—an iteratively reweighted least squares (IRLS) method for robust subspace recovery and a Newton-Lanczos method for trace ratio problems—that go beyond the limitations of simple algorithms like the steepest descent, and we provide

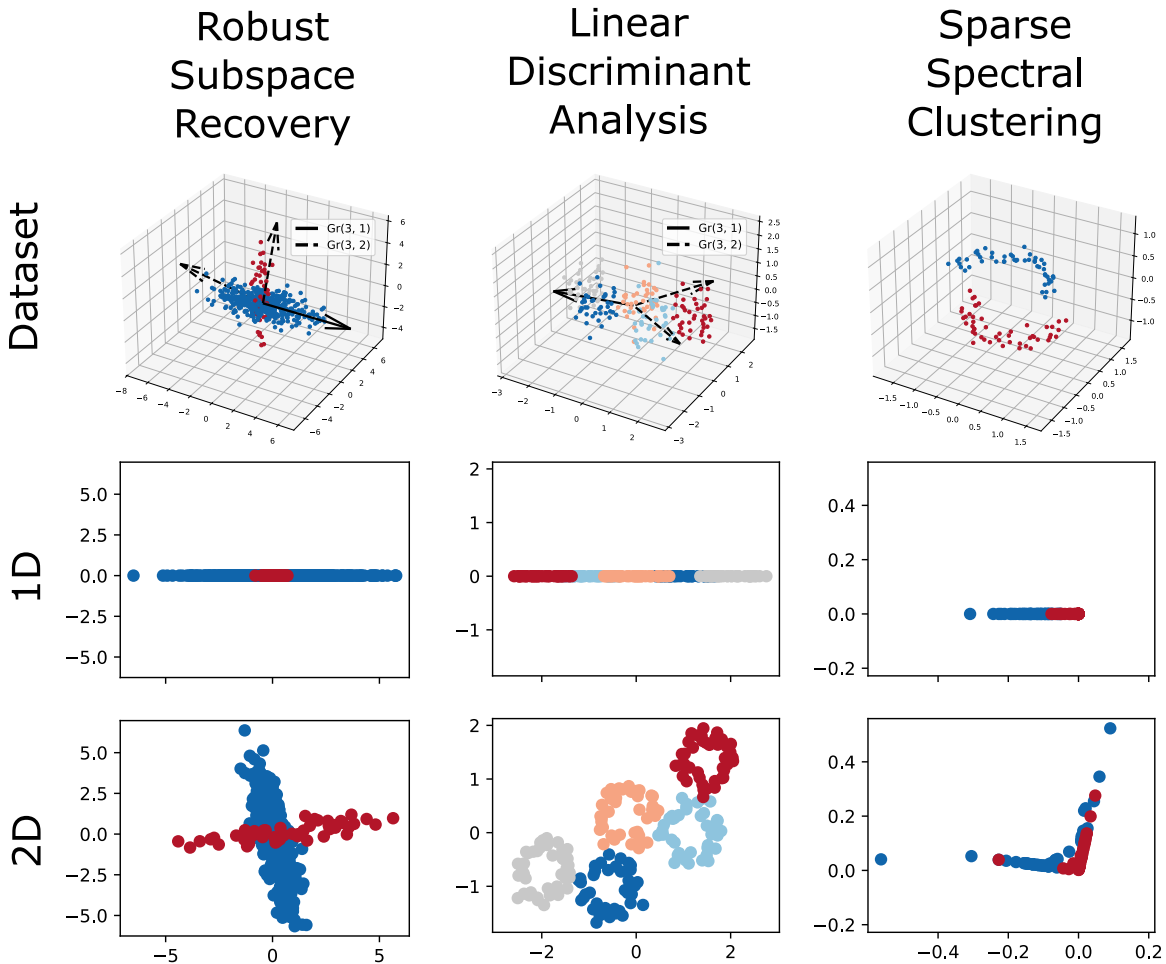


Figure 1: Illustration of the subspace nestedness issue in three important machine learning problems: robust subspace recovery (left), linear discriminant analysis (middle) and sparse spectral clustering (right). For each data set (top), we plot its projection onto the optimal 1D subspace (middle) and 2D subspace (bottom) obtained by solving the associated Grassmannian optimization problem. The 1D and 2D representations are *inconsistent*—in the sense that the 1D plot is not the projection of the 2D plot onto the horizontal axis—which is a pitfall for data analysis.

the proofs that are not in the main body. In Appendix D, we report the empirical running times for the flag trick in robust subspace recovery, trace ratio and spectral clustering.

## 2. Basics of Flag Manifolds

A flag is a sequence of nested linear subspaces of increasing dimension. This section introduces flag manifolds and provides the minimal tools to perform optimization on those spaces. More details and properties can be obtained in dedicated works (Szwagier, 2025).

### 2.1 Flags in the Scientific Literature

Flags, by providing a natural parametrization for the eigenspaces of symmetric matrices, have long been geometrical objects of interest in the scientific community, with traditional applications in physics (Arnold, 1972) and numerical analysis (Duistermaat et al., 1983; Ammar and Martin, 1986; Watkins and Elsner, 1991; Helmke, 1991; Helmke and Moore, 1994). The seminal work of Edelman et al. (1998) on the geometry of algorithms with orthogonality constraints, paved the way for Riemannian optimization algorithms on Stiefel, Grassmann and related matrix manifolds. The first work that explicitly proposed an optimization algorithm on flag manifolds dates back to 2006. Nishimori et al. (2006) reformulated *independent subspace analysis*—an important variant of independent component analysis which maximizes the independence between the norms of projected samples into mutually-orthogonal subspaces (Cardoso, 1998; Hyvärinen and Hoyer, 2000)—as an optimization problem on flag manifolds and solved it via a Riemannian gradient descent. This led to many papers developing or performing optimization on flag manifolds (Nishimori et al., 2007, 2008; Fiori, 2011, 2012; Ye et al., 2022; Zhu and Shen, 2024) and to implementations in open-source packages for differential geometry such as *Manifolds.jl* (Axen et al., 2023).

Recently, flag manifolds played a central role in the investigation of a *curse of isotropy* in principal component analysis. By means of a probabilistic covariance model parameterized with flags, Szwagier and Pennec (2026) show that *close eigenvalues*—whose relative distance is lower than a certain threshold—should be assumed to be *equal*. The resulting *principal subspace analysis* has led to several follow-up works, notably an efficient algorithm for covariance estimation in high dimensions (Szwagier et al., 2025) and an extension to parsimonious Gaussian mixture models (Szwagier et al., 2026).

It is important to notice that flags—as mutually-orthogonal subspaces—also implicitly appear in many works, with different keywords. For instance, the mutually-orthogonal class-subspaces of Watanabe and Pakvasa (1973), the adaptive-subspace self-organizing maps of Kohonen (1996) and follow-up works (Kim et al., 2010; Giguere et al., 2017) implicitly involve flags under the name of “mutually-orthogonal subspaces”. Flags also appear in many spectral methods under keywords such as “isospectral manifolds” (Deift et al., 1983; Watkins, 1984; Brockett, 1991; Helmke, 1991; Lim and Ye, 2024); indeed, at fixed eigenvalues, the remaining degrees of freedom in a symmetric matrix are the associated eigenspaces, which exactly correspond to flags.

Flags as sequences of nested subspaces were first introduced in the machine learning literature as robust prototypes to represent collections of subspaces of different dimensions (Draper et al., 2014; Santamaría et al., 2016; Mankovich et al., 2022). They were obtained via a sequential construction (where one dimension is added at a time), which can

be problematic for greediness reasons (Huber, 1985; Lerman and Maunu, 2018b). Pennecc (2018) was the first to show that PCA can be reformulated as an optimization problem on flag manifolds by summing the unexplained variance at different dimensions, raising interesting perspectives for multilevel data analysis on manifolds. This principle was recently applied to several variants of PCA under the name of *flagification* in Mankovich et al. (2024), showing an important robustness to outliers. Some works compute distance-related quantities on flag-valued data sets, with notable applications in computer vision and hyperspectral imaging (Ma et al., 2021, 2022; Nguyen, 2022; Mankovich and Birdal, 2023; Szwagier and Pennecc, 2023). Flags were recently used to learn nested representations from data with a prespecified hierarchy (Mankovich et al., 2025). Other applications include shape spaces (Ciuclea et al., 2023), quantum physics and chemistry (Vidal et al., 2024), system identification (Jin and Coulson, 2025) and deep learning (Rauba and Schaar, 2026).

## 2.2 Definition and Representation of Flag Manifolds

Let  $p \geq 2$  and  $q_{1:d} := (q_1, q_2, \dots, q_d)$  be a sequence of increasing integers such that  $0 < q_1 < q_2 < \dots < q_d < p$ . A *flag* of *signature*  $(p, q_{1:d})$  is a sequence of nested linear subspaces  $\{0\} \subset \mathcal{S}_1 \subset \mathcal{S}_2 \subset \dots \subset \mathcal{S}_d \subset \mathbb{R}^p$  of respective dimensions  $q_1, q_2, \dots, q_d$ , noted here  $\mathcal{S}_{1:d} := (\mathcal{S}_1, \dots, \mathcal{S}_d)$ .<sup>2</sup> A flag  $\mathcal{S}_{1:d}$  can be canonically represented as a sequence of symmetric matrices that are the *orthogonal projection matrices* onto the nested subspaces, i.e.,  $\mathcal{S}_{1:d} \mapsto (\Pi_{\mathcal{S}_1}, \dots, \Pi_{\mathcal{S}_d}) \in \text{Sym}(p)^d$ . We call this the *projection representation* of flags.

The set of flags of signature  $(p, q_{1:d})$  is a *smooth manifold* (Ye et al., 2022) denoted here  $\text{Fl}(p, q_{1:d})$ . Flag manifolds generalize Grassmannians when  $d = 1$ —as  $\text{Fl}(p, q) = \text{Gr}(p, q)$ —and therefore share many practical properties that are useful for optimization (Edelman et al., 1998). In the following, we use the notations  $q_0 := 0$ ,  $q := q_d$  and  $q_{d+1} := p$ .

For computational reasons, flags are often represented as orthonormal  $q$ -frames. Those correspond to points on the *Stiefel manifold*  $\text{St}(p, q) = \{U \in \mathbb{R}^{p \times q} : U^\top U = I_q\}$ . Let us define sequentially, for  $k \in \{1, \dots, d\}$ ,  $U_k \in \text{St}(p, q_k - q_{k-1})$  such that  $[U_1 \ \dots \ U_k]$  is an orthonormal basis of  $\mathcal{S}_k$  (this is possible thanks to the nestedness of the subspaces). Then,  $U_{1:d} := [U_1 \ \dots \ U_d] \in \text{St}(p, q)$  parametrizes the flag  $\mathcal{S}_{1:d}$ . We call this a *Stiefel representative* of  $\mathcal{S}_{1:d}$ . Such a representation is not unique—contrary to the projection representation defined previously—due to the *rotational-invariance* of orthonormal bases of subspaces. More precisely, if  $U_{1:d}$  is a Stiefel representative of the flag  $\mathcal{S}_{1:d}$ , then for any set of orthogonal matrices  $R_k \in \mathcal{O}(q_k - q_{k-1})$ , the matrix  $U'_{1:d} := [U_1 R_1 \ \dots \ U_d R_d]$  spans the same flag of subspaces  $\mathcal{S}_{1:d}$ . This provides flag manifolds with a quotient manifold structure (Edelman et al., 1998; Absil et al., 2008; Ye et al., 2022):

$$\text{Fl}(p, q_{1:d}) \cong \text{St}(p, q) / (\mathcal{O}(q_1) \times \mathcal{O}(q_2 - q_1) \times \dots \times \mathcal{O}(q_d - q_{d-1})). \quad (1)$$

**Remark 3 (Orthogonal Representation)** *For computations, one might have to perform the orthogonal completion of some Stiefel representatives. Let  $U_{1:d} := [U_1 \ \dots \ U_d] \in \text{St}(p, q)$ , then one denotes  $U_{d+1} \in \text{St}(p, p - q_d)$  to be any orthonormal basis such that  $U_{1:d+1} := [U_1 \ \dots \ U_d \ U_{d+1}] \in \mathcal{O}(p)$ . Such an orthogonal matrix  $U_{1:d+1}$  will be called*

2. Flags can equivalently be defined as sequences of *mutually orthogonal* subspaces  $\mathcal{V}_1 \perp \mathcal{V}_2 \perp \dots \perp \mathcal{V}_d$ , of respective dimensions  $q_1, q_2 - q_1, \dots, q_d - q_{d-1}$ , by taking  $\mathcal{V}_k$  to be the orthonormal complement of  $\mathcal{S}_{k-1}$  onto  $\mathcal{S}_k$ . This definition is more convenient for computations, but we will not need it in this paper.

an orthogonal representative of the flag  $\mathcal{S}_{1:d}$ . In the following, we may abusively switch from one representation to the other since they represent the same flag.

### 2.3 Optimization on Flag Manifolds

There is a rich literature on optimization on smooth manifolds (Edelman et al., 1998; Absil et al., 2008; Boumal, 2023), and the particular case of flag manifolds has been notably addressed in Ye et al. (2022) and Zhu and Shen (2024). Since flag manifolds can be represented as quotient spaces of Stiefel manifolds, which themselves are embedded in a Euclidean matrix space, one can develop some optimization algorithms without much difficulty. In this paper, we will use a *steepest descent* algorithm, which is drawn from several works (Chikuse, 2003; Nishimori et al., 2006; Ye et al., 2022; Zhu and Shen, 2024). Let  $f: \text{Fl}(p, q_{1:d}) \rightarrow \mathbb{R}$  be a smooth function on a flag manifold, expressed in the Stiefel representation (e.g.,  $f(U_{1:d}) = \sum_{k=1}^d \|U_k U_k^\top x\|$  for some  $x \in \mathbb{R}^p$ ). Given  $U_{1:d} \in \text{St}(p, q)$ , let  $\text{Grad } f(U_{1:d}) = (\partial f / \partial U_{ij})_{i,j=1}^{p,q}$  denote the (Euclidean) gradient of  $f$ . To “stay” on the manifold, one first computes the *Riemannian gradient* of  $f$  at  $U_{1:d}$ , noted  $\nabla f(U_{1:d})$ . It can be thought of as a projection of the Euclidean gradient onto the tangent space and computed explicitly (Nishimori et al., 2006; Ye et al., 2022). Then, one moves in the opposite direction of  $\nabla f(U_{1:d})$  with a so-called *retraction*, which is chosen to be the polar retraction of Zhu and Shen (2024), combined with a line-search. We iterate until convergence. The final steepest descent algorithm is described in Algorithm 1.

---

#### Algorithm 1 Steepest Descent on Flag Manifolds

---

**Require:**  $f: \text{Fl}(p, q_{1:d}) \rightarrow \mathbb{R}$  a function;  $U_{1:d} \in \text{Fl}(p, q_{1:d})$  a flag (Stiefel representation)

**for**  $t \in \{1, 2, \dots\}$  **do**

$[G_1 \ \dots \ G_d] \leftarrow \text{Grad } f(U_{1:d})$  ▷ Euclidean gradient

$\nabla \leftarrow [G_k - (U_k U_k^\top G_k + \sum_{l \neq k} U_l G_l^\top U_k)]_{k=1 \dots d}$  ▷ Riemannian gradient

$U_{1:d} \leftarrow \text{polar}(U_{1:d} - \alpha \nabla)$  ▷ polar retraction + line search

**end for**

**Ensure:**  $U_{1:d}^* \in \text{Fl}(p, q_{1:d})$  an optimal flag

---

**Remark 4 (Initialization)** *We can initialize Algorithm 1 “randomly” (under the invariant measure of James, 1954): we generate a random  $p \times q$  matrix with normal entries and perform its polar decomposition; the orthogonal factor then follows a uniform distribution (Chikuse, 2003, Theorem 1.5.5). We can also initialize Algorithm 1 with a specific flag depending on the objective function, as we will see in Section 4. Such an initial flag can, for instance, be obtained via the eigendecomposition of a symmetric matrix related to the problem, or by refining the solution of the original subspace problem.*

**Remark 5 (Optimization Variants)** *Many extensions of Algorithm 1 can be considered: conjugate gradient (Ye et al., 2022), Riemannian trust region (Absil et al., 2008), etc. We can also replace the polar retraction with a geodesic step (Ye et al., 2022) or other retractions (Zhu and Shen, 2024).*

**Remark 6 (Complexity Analysis)** *The number of operations for the polar retraction is  $O(pq^2) + O(q^3)$  (see Zhu and Shen, 2024 and comparisons with other retractions therein).*

The number of operations for the Riemannian gradient (from the Euclidean gradient) is  $O(pq^2)$ . The number of operations for the Euclidean gradient depends on the objective function  $f$  (which itself depends on the dimension  $p$ , the signature  $q_{1:d}$  and the number of samples  $n$ ), and the overall complexity of Algorithm 1 also depends on the number of iterations (for both the main loop and the line search). Therefore, for completeness, we report the empirical running times of our steepest descent algorithm in Appendix D.

### 3. The Flag Trick in Theory

In this section, we motivate and introduce the flag trick to make subspace learning methods nested. The key result is Theorem 8, which states that the classical PCA at a fixed dimension can be converted into a nested multilevel method using nested projectors.

**Remark 7 (Centering)** *In the remaining of the paper, we assume that the data have been already centered around a point of interest (e.g., its mean or geometric median), so that we are only interested in fitting linear subspaces and not affine ones. One could directly include the center in the optimization variables—in which case the geometry would be the one of affine Grassmannians (Lim et al., 2019) or affine flags (Pennec, 2018)—but we will not do it in this work for conciseness.*

#### 3.1 From Subspaces to Flags of Subspaces: the Seminal Example of PCA

PCA is known as the eigendecomposition of the sample covariance matrix. Originally, it can be formulated as the search for a low-dimensional subspace that minimizes the unexplained variance (or maximizes the explained variance). Let  $X := [x_1 \ \cdots \ x_n] \in \mathbb{R}^{p \times n}$  be a data matrix with  $n$  samples, let  $\mathcal{S} \in \text{Gr}(p, q)$  be a  $q$ -dimensional subspace, and let  $\Pi_{\mathcal{S}} \in \mathbb{R}^{p \times p}$  be the orthogonal projection matrix onto  $\mathcal{S}$ . Then PCA consists in the following optimization problem on Grassmannians:

$$\mathcal{S}_q^* = \arg \min_{\mathcal{S} \in \text{Gr}(p, q)} \|X - \Pi_{\mathcal{S}} X\|_F^2, \quad (2)$$

where  $\|M\|_F^2 := \text{tr}(M^\top M)$  denotes the *Frobenius norm*. The solution to the optimization problem is the  $q$ -dimensional subspace spanned by the leading eigenvectors of the sample covariance matrix  $S := \frac{1}{n} X X^\top$ , that we note  $\mathcal{S}_q^* = \text{Span}(v_1, \dots, v_q)$ . It is unique when the sample eigenvalues  $q$  and  $q + 1$  are distinct, which is almost sure when  $q \leq \text{rank}(S)$ . We will assume to be in such a setting in the following for simplicity but it can be easily handled otherwise by “grouping” the repeated eigenvalues (Szwagier and Pennec, 2026). In this case, the principal subspaces are *nested* for increasing  $q$ , i.e., if  $\mathcal{S}_q^*$  is the  $q$ -dimensional principal subspace, then for any  $r > q$ , one has  $\mathcal{S}_q^* = \text{Span}(v_1, \dots, v_q) \subset \text{Span}(v_1, \dots, v_r) = \mathcal{S}_r^*$ .

Another way of performing PCA is in a sequential manner (cf. Remark 1). We first estimate the 1D subspace  $\mathcal{V}_1^*$  that minimizes the unexplained variance, then estimate the 1D subspace  $\mathcal{V}_2^*$  that minimizes the unexplained variance while being orthogonal to the previous one, and so on and so forth. This gives the following *constrained* optimization problem on 1D Grassmannians:

$$\mathcal{V}_q^* = \arg \min_{\substack{\mathcal{V} \in \text{Gr}(p, 1) \\ \mathcal{V} \perp \mathcal{V}_{q-1} \perp \cdots \perp \mathcal{V}_1}} \|X - \Pi_{\mathcal{V}} X\|_F^2. \quad (3)$$

This construction naturally yields a sequence of nested subspaces of increasing dimension—i.e., a flag of subspaces—better and better approximating the data set:

$$\{0\} \subset \mathcal{S}_1^* \subset \mathcal{S}_2^* \subset \cdots \subset \mathcal{S}_{p-1}^* \subset \mathbb{R}^p, \text{ with } \mathcal{S}_k^* = \bigoplus_{l=1}^k \mathcal{V}_l^*. \quad (4)$$

Those subspaces happen to be exactly the same as the ones obtained by solving the subspace learning optimization problem (2), although the way they are obtained (in a greedy manner) is different. This is generally not the case for other dimension reduction problems (for instance in robust subspace recovery, as it is raised in the final open questions of Lerman and Maunu, 2018b).

Hence, the *subspace* learning formulation of PCA (2) is equivalent to the *sequential* formulation of PCA (3)—in terms of cumulative spans (4)—and both yield a flag of subspaces best and best approximating the data. One can wonder if this result could be directly obtained by formulating an optimization problem on flag manifolds. The answer is *yes*, as first proven in Pennec (2018, Theorem 9) with an *accumulated unexplained variance* (AUV) technique, but there is not a unique way to do it. Motivated by the recent principal subspace analysis (Szwagier and Pennec, 2026), we propose in the following theorem a generic principle to formulate PCA as an optimization on flag manifolds.

**Theorem 8 (Nested PCA with Flag Manifolds)** *Let  $X := [x_1 \ \cdots \ x_n] \in \mathbb{R}^{p \times n}$  be a centered  $p$ -dimensional ( $p \geq 2$ ) data set with  $n$  samples. Let  $q_{1:d} := (q_1, q_2, \dots, q_d)$  be a sequence of increasing dimensions such that  $0 < q_1 < q_2 < \cdots < q_d < p$ . Let  $S := \frac{1}{n} X X^\top$  be the sample covariance matrix. Assume that it eigendecomposes as  $S := \sum_{j=1}^p \ell_j v_j v_j^\top$  where  $\ell_1 \geq \cdots \geq \ell_p$  are the eigenvalues and  $v_1 \perp \cdots \perp v_p$  are the associated eigenvectors. Then PCA can be reformulated as the following optimization problem on flag manifolds:*

$$\mathcal{S}_{1:d}^* = \arg \min_{\mathcal{S}_{1:d} \in \text{Fl}(p, q_{1:d})} \left\| X - \frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k} X \right\|_F^2. \quad (5)$$

*More precisely, one has  $\mathcal{S}_{1:d}^* = (\text{Span}(v_1, \dots, v_{q_1}), \text{Span}(v_1, \dots, v_{q_2}), \dots, \text{Span}(v_1, \dots, v_{q_d}))$ . The solution is unique if and only if  $\ell_{q_k} \neq \ell_{q_{k+1}}, \forall k \in \{1, \dots, d\}$ .*

**Proof** One has

$$\left\| X - \frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k} X \right\|_F^2 = \text{tr} \left( X^\top \left( I_p - \frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k} \right)^2 X \right), \quad (6)$$

$$= \frac{1}{d^2} \text{tr} \left( X^\top \left( \sum_{k=1}^d (I_p - \Pi_{\mathcal{S}_k}) \right)^2 X \right), \quad (7)$$

$$= \frac{n}{d^2} \text{tr} (W^2 S), \quad (8)$$

with  $W = \sum_{k=1}^d (I_p - \Pi_{\mathcal{S}_k})$  and  $S = \frac{1}{n} X X^\top$ . Let  $U_{1:d+1} := [U_1 \ \cdots \ U_d \ U_{d+1}] \in \mathcal{O}(p)$  be an orthogonal representative (cf. Section 2) of the optimization variable  $\mathcal{S}_{1:d} \in$

$\text{Fl}(p, q_{1:d})$ . Then one has  $\Pi_{\mathcal{S}_k} = U_{1:d+1} \text{diag}(I_{q_k}, 0_{p-q_k}) U_{1:d+1}^\top$ . Therefore, one has  $W = U_{1:d+1} \Lambda U_{1:d+1}^\top$ , with  $\Lambda = \text{diag}(0 I_{q_1}, 1 I_{q_2-q_1}, \dots, d I_{q_{d+1}-q_d})$ . Hence, one has

$$\arg \min_{\mathcal{S}_{1:d} \in \text{Fl}(p, q_{1:d})} \left\| X - \frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k} X \right\|_F^2 \iff \arg \min_{U \in \mathcal{O}(p)} \frac{n}{d^2} \text{tr} \left( U \Lambda^2 U^\top S \right). \quad (9)$$

The latter problem is exactly the same as the one of Szwagier and Pennecc (2026) for principal subspace analysis parameter inference. Hence, one can conclude the proof on existence and uniqueness via their maximum likelihood estimation theorem.  $\blacksquare$

The key element of the proof of Theorem 8 is that averaging the nested projectors yields a *hierarchical reweighting* of the (mutually-orthogonal) principal subspaces. More precisely, the  $k$ -th principal subspace has weight  $(k-1)^2$ , and this monotonic weighting enables to get a hierarchy of eigenspaces (Absil et al., 2008; Cunningham and Ghahramani, 2015; Pennecc, 2018; Oftadeh et al., 2020). In the following, we note  $\Pi_{\mathcal{S}_{1:d}} := \frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k}$  and call this symmetric matrix the *average multilevel projector*, which will be central in the extension of subspace methods into multilevel subspace methods.

**Remark 9 (Weighted Projectors)** *One can show that any convex combination of the nested projectors yields the same result for PCA. Indeed, let  $\alpha_1, \dots, \alpha_d$  be positive scalars. Let  $\mathcal{S}_{1:d} \in \text{Fl}(p, q_{1:d})$  and  $U_{1:d} := [U_1 \ \dots \ U_d] \in \text{St}(p, q)$  be a Stiefel representative of  $\mathcal{S}_{1:d}$  (cf. Section 2). One has  $\sum_{k=1}^d \alpha_k \Pi_{\mathcal{S}_k} = \sum_{k=1}^d \alpha_k \sum_{l=1}^k U_l U_l^\top = \sum_{l=1}^d (\sum_{k=l}^d \alpha_k) U_l U_l^\top$ . Therefore, the weights in front of each (mutually-orthogonal) subspace  $\text{Span}(U_l)$  are strictly decreasing, and the solutions of Theorem 8 are exactly the same. For criteria beyond PCA, the choice of weights might have an influence on the solution. In the following, we will by default consider uniform weights  $(\frac{1}{d}, \dots, \frac{1}{d})$ .*

### 3.2 The Flag Trick

As we will see in the following (Section 4), many important machine learning problems can be formulated as the optimization of a certain function  $f$  on Grassmannians. Theorem 8 shows that replacing the subspace projection matrix  $\Pi_{\mathcal{S}}$  appearing in the objective function by the average multilevel projector  $\Pi_{\mathcal{S}_{1:d}}$  yields a sequence of subspaces that meet the original objective of principal component analysis, while being nested. This leads us to introduce the *flag trick* for general subspace learning problems.

**Definition 10 (Flag Trick)** *Let  $p \geq 2$ ,  $0 < q < p$  and  $q_{1:d} := (q_1, q_2, \dots, q_d)$  be a sequence of increasing dimensions such that  $0 < q_1 < q_2 < \dots < q_d < p$ . The flag trick consists in replacing a subspace learning problem of the form*

$$\arg \min_{\mathcal{S} \in \text{Gr}(p, q)} f(\Pi_{\mathcal{S}}) \quad (10)$$

with the following optimization problem:

$$\arg \min_{\mathcal{S}_{1:d} \in \text{Fl}(p, q_{1:d})} f(\Pi_{\mathcal{S}_{1:d}}), \quad (11)$$

where  $\Pi_{\mathcal{S}_{1:d}} := \frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k}$  is the average multilevel projector.

Let us note that for  $d = 1$ —i.e., with signatures of the form  $q_{1:d} = (q)$ —the flag-tricked problem (11) is equivalent to the original subspace problem (10). For  $d > 1$ , the flag trick yields new objective criteria that we cannot a priori relate to the original subspace learning problem. Except for the very particular case of PCA where  $f_X(\Pi) = \|X - \Pi X\|_F^2$  (Theorem 8), we cannot expect to have an analytic solution to the flag problem (11); indeed, in general, subspace problems do not even have a closed-form solution as we shall see in Section 4. This justifies the introduction of optimization algorithms on flag manifolds like Algorithm 1.

**Remark 11 (Flag Trick vs. AUV)** *The original idea of accumulated unexplained variance (Pennec, 2018) (and its subsequent application to several variants of PCA under the name of “flagification” in Mankovich et al., 2024) consists in summing the subspace criteria at different dimensions, while the flag trick directly averages the orthogonal projection matrices that appear inside the objective function. While both ideas are worth experimenting with, we believe that the flag trick has a wider practical reach. Indeed, from a technical viewpoint, the flag trick appears at the covariance level and directly yields a hierarchical reweighting of the principal subspaces (see end of Section 3.1). This reweighting is only indirect with the AUV—due to the linearity of the trace operator—and is not expected to occur beyond PCA. Notably, as we shall see in Section 4, the flag trick enables to easily develop extensions of well-known methods involving PCA, like IRLS (Lerman and Maunu, 2018a) or Newton-Lanczos methods for trace ratio optimization (Ngo et al., 2012), it less likely reaches singularities of the objective function (see, notably, end of Section 4.2.1) and it is closer in spirit to the statistical formulations of PCA (Szwagier and Pennec, 2026).*

### 3.3 Multilevel Machine Learning

Subspace learning is often used as a preprocessing task before running a machine learning algorithm, notably to overcome the curse of dimensionality. One usually projects the data onto the optimal subspace  $\mathcal{S}^* \in \text{Gr}(p, q)$  and use the resulting lower-dimensional data set as an input to a machine learning task like clustering, classification or regression (Bouveyron et al., 2019). Since the flag trick problem (11) does not output one subspace but a hierarchical sequence of nested subspaces, it is legitimate to wonder what to do with such a multilevel representation. In this subsection, we propose a general *ensemble learning* method to aggregate the hierarchical information coming from the flag of subspaces.

Let us consider a data set  $X := [x_1 \ \cdots \ x_n] \in \mathbb{R}^{p \times n}$  (possibly with some associated labels  $Y := [y_1 \ \cdots \ y_n] \in \mathbb{R}^{m \times n}$ ). In machine learning, one often fits a model to the data set by optimizing an objective function of the form  $R_{X,Y}(g) = \frac{1}{n} \sum_{i=1}^n L(g(x_i), y_i)$ . With the flag trick, we get a filtration of projected data points  $Z_k = \Pi_{\mathcal{S}_k^*} X, k \in \{1, \dots, d\}$  that can be given as an input to different machine learning algorithms. This yields optimal predictors  $g_k^* = \text{argmin}_g R_{Z_k, Y}(g)$  which can be aggregated via ensembling methods. For instance, *voting* methods choose the model with the highest performance on holdout data; this corresponds to selecting the optimal dimension  $q^* \in q_{1:d}$  *a posteriori*, based on the machine learning objective. A more nuanced idea is the one of *soft voting*, which makes a weighted averaging of the predictions. The weights can be uniform, proportional to the performances of the individual models, or learned to maximize the performance of the weighted prediction (Perrone and Cooper, 1995). Soft voting gives different weights to the

nested subspaces depending on their contribution to the ensembled prediction and therefore provides a soft measure of the relative importance of the different dimensions. In that sense, it goes beyond the classical manifold assumption stating that data have one intrinsic dimension, and instead proposes a soft blend between dimensions that is adapted to the learning objective. This sheds light on the celebrated paper of Minka for the automatic choice of dimensionality in PCA (Minka, 2000, Section 5). Many other ensembling methods are possible like gradient boosting, Bayesian model averaging and stacking. The whole methodology is summarized in Algorithm 2 and illustrated in Figure 2.

---

**Algorithm 2** Flag Trick with Ensemble Learning

---

**Require:**  $X := [x_1 \ \cdots \ x_n] \in \mathbb{R}^{p \times n}$  a data matrix;  $q_{1:d} := (q_1, \dots, q_d)$  a flag signature;  
 $f$  a subspace learning objective; (opt.)  $Y := [y_1 \ \cdots \ y_n] \in \mathbb{R}^{m \times n}$  a label matrix  
 $\mathcal{S}_{1:d}^* \leftarrow \operatorname{argmin}_{\mathcal{S}_{1:d} \in \operatorname{Fl}(p, q_{1:d})} f(\Pi_{\mathcal{S}_{1:d}})$  ▷ flag trick (11) + optimization (Alg. 1)  
**for**  $k \in \{1, \dots, d\}$  **do**  
     $g_k^* \leftarrow \operatorname{fit}(\Pi_{\mathcal{S}_k^*} X, Y)$  ▷ learning on  $q_k$ -dimensional projected data  
     $Y_k^* \leftarrow g_k^*(\Pi_{\mathcal{S}_k^*} X)$  ▷ prediction on  $q_k$ -dimensional projected data  
**end for**  
 $Y^* \leftarrow \operatorname{ensembling}(Y_1^*, \dots, Y_d^*)$  ▷ weighted predictions  
**Ensure:**  $Y^*$  the ensembled predictions

---

Algorithm 2 is a general proposition of multilevel machine learning with flags, but many other uses of the optimal flag  $\mathcal{S}_{1:d}^*$  are possible, depending on the application. For instance, one may directly use the reweighted data matrix  $\Pi_{\mathcal{S}_{1:d}^*} X$  as an input to the machine learning algorithm. This enables to fit only one model instead of  $d$ . One can also simply analyze the projected data *qualitatively* via scatter plots or reconstruction plots as evoked in Section 1. The nestedness will automatically bring consistency contrarily to non-nested subspace methods, and therefore improve interpretability. Finally, many other ideas can be borrowed from the literature on subspace clustering and flag manifolds (Draper et al., 2014; Launay et al., 2021; Ma et al., 2021; Mankovich et al., 2022; Mankovich and Birdal, 2023; Mankovich et al., 2024, 2025), for instance the computation of distances between flags coming from different data sets as a multilevel measure of similarity between data sets.

**Remark 12 (Choice of Signature)** *In the following experiments, flag signatures are chosen heuristically. Although important, we believe that such a choice must be thoroughly addressed for each subspace learning problem, which leads us to leave that for future research. Alternative methodologies for the automatic choice of dimensions are discussed in Section 5. One could notably investigate penalties on the spectral gaps in the same spirit as Szwagier et al. (2025)—where the penalty factor would be tuned according to the downstream machine learning task. Beyond spectral gaps, the signature could be selected by starting from a flag of signature  $(p, (1, 2, \dots, p-1))$  and removing the nested subspaces that do not significantly increase the accuracy on the downstream machine learning task. For the specific example of subspace averaging, a hyperparameter-free signature selection method extending the order fitting rule of Santamaría et al. (2016) could be investigated via the flag trick.*

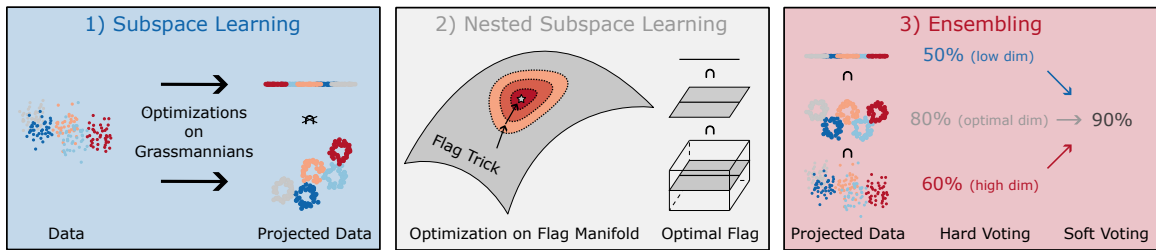


Figure 2: Illustration of the flag trick methodology. 1) We start with a subspace learning problem:  $\arg \min_{\mathcal{S} \in \text{Gr}(p,q)} f(\Pi_{\mathcal{S}})$ . Trying different  $q$  yields in general non-nested subspaces, which raises an issue of consistency between data representations. 2) We convert the subspace learning problem into a nested subspace learning problem via the flag trick:  $\arg \min_{\mathcal{S}_{1:d} \in \text{Fl}(p,q_{1:d})} f(\frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k})$ . We run a steepest descent on flag manifolds (Algorithm 1) and get a flag of nested subspaces. 3) We fit a machine learning algorithm (regression, classification, etc.) to the projected data at each dimension  $q_k \in q_{1:d}$ . We aggregate the estimators via ensembling methods (hard voting, soft voting, etc.) and get improved predictions.

## 4. The Flag Trick in Action

In this section, we provide some applications of the flag trick to several learning problems. We choose to focus on subspace recovery, trace ratio and spectral clustering problems. Other ones, like domain adaptation, matrix completion and subspace tracking are developed or mentioned in the last subsection but not experimented for conciseness.

### 4.1 Outline and Experimental Setting

For each application, we first present the learning problem as an optimization on Grassmannians. Second, we formulate the associated flag learning problem by applying the flag trick (Definition 10). Third, we optimize the problem on flag manifolds with the steepest descent method (Algorithm 1)—more advanced algorithms are also derived in the appendix. Finally, we perform various nestedness and ensemble learning experiments, via Algorithm 2, on both synthetic and real data sets.

The general methodology to compare Grassmann-based methods to flag-based methods is the following one. For each experiment, we first choose a flag signature  $q_{1:d} := (q_1, \dots, q_d)$ , then we run independent optimization algorithms on  $\text{Gr}(p, q_1), \dots, \text{Gr}(p, q_d)$  (10) and finally we compare the optimal subspaces  $\mathcal{S}_k^* \in \text{Gr}(p, q_k)$  to the optimal flag of subspaces  $\mathcal{S}_{1:d}^* \in \text{Fl}(p, q_{1:d})$  obtained via the flag trick (11). To show the nestedness issue in Grassmann-based methods, we compute the subspace distances  $\Theta(\mathcal{S}_k^*, \mathcal{S}_{k+1}^*)_{k \in \{1, \dots, d-1\}}$ , where  $\Theta$  is the generalized Grassmann distance of Ye and Lim (2016, Eq. 14). It consists in the  $\ell^2$ -norm of the principal angles, which can be obtained from the singular value decomposition (SVD) of the inner-product matrices  $U_k^\top U_{k+1}$ — $U_k \in \text{St}(p, q_k)$  being an orthonormal basis of  $\mathcal{S}_k^*$ .

Regarding the implementation of the steepest descent algorithm on flag manifolds (Algorithm 1), we develop a new class of manifolds in PyManOpt (Boumal et al., 2014; Townsend

et al., 2016), and run their SteepestDescent algorithm. Our implementation of the `Flag` class is based on the Stiefel representation of flag manifolds, detailed in Section 2, with the retraction being the polar retraction. For the computation of the gradient, we use automatic differentiation with the `autograd` package. We could derive the gradients by hand from the expressions we get, but we use automatic differentiation as strongly suggested in PyManOpt’s documentation. Finally, the real data sets and the machine learning methods used in the experiments can be found in scikit-learn (Pedregosa et al., 2011).

## 4.2 The Flag Trick for Robust Subspace Recovery Problems

As shown in introduction and particularly in Figure 1, robust subspace recovery (RSR) methods based on Grassmannian optimization might suffer from the nestedness issue, which notably yields inconsistent data representations across different dimensions (as seen on the left part of the figure, where outliers somewhat swap position with inliers from 1D to 2D). The RSR overview by Lerman and Maunu (2018b) particularly emphasizes the need for nested subspace methods, although it leaves this as an open perspective. In this subsection, we address the nestedness issue in robust subspace recovery by applying the flag trick (Definition 10) to the problem of absolute deviation minimization (Maunu et al., 2019).

Robust subspace recovery is an outlier-robust extension of classical dimension reduction methods such as PCA. Let us consider a data set that is a union of *inliers* and *outliers*—the inliers are assumed to lie near a low-dimensional subspace  $\mathcal{S}$  while the outliers live in the ambient space. The aim of RSR is to recover  $\mathcal{S}$ . Without further specifications, the RSR problem might not be well-posed. Therefore, the works in this domain often have to make some assumptions on the inlier and outlier distributions in order to obtain convergence and recovery guarantees. For instance, in Lerman and Maunu (2018b), it is assumed that the inliers “fill” the lower-dimensional subspace and that the outliers are not much “aligned”; this is rigorously defined in Lerman et al. (2015) and Maunu et al. (2019) through *permeance* and *alignment* statistics.

A generative model—the *haystack model*—following these assumptions is introduced in Lerman et al. (2015). It assumes an isotropic Gaussian distribution on the subspace for the inliers and an isotropic Gaussian distribution on the (full) ambient space for the outliers. A more realistic model—the *generalized haystack model*—is introduced in Maunu et al. (2019) to circumvent the simplistic nature of the haystack model. This one assumes general (anisotropic) Gaussian distributions for the inliers and outliers.<sup>3</sup> This makes the learning harder, since the anisotropy may keep the inliers from properly permeating the low-dimensional subspace—as discussed in Section 1. Therefore, one has to make some stronger assumptions on the inlier-outlier ratio and the covariance eigenvalues distributions to derive some convergence and recovery guarantees.

**Remark 13 (Parametrization of RSR Models)** *The inlier distribution in the haystack model follows the isotropic PPCA model (Bouveyron et al., 2007, 2011), while it follows the*

---

3. The generalized haystack model is more-precisely defined as follows: the  $n_{\text{in}}$  inliers are i.i.d sampled from a sub-Gaussian distribution  $\mathcal{N}(0, \Sigma_{\text{in}}/q)$ , with  $\Sigma_{\text{in}} = U\Lambda U^\top$ ,  $U \in \text{St}(p, q)$  and  $\Lambda \in \text{diag}(\mathbb{R}^q)$ , while the  $n_{\text{out}}$  outliers are i.i.d sampled from a sub-Gaussian distribution  $\mathcal{N}(0, \Sigma_{\text{out}}/\text{rk}(\Sigma_{\text{out}}))$ , with  $\Sigma_{\text{out}} \in \text{Sym}(p)$ . The (original) haystack model is a particular case of the generalized haystack model with isotropic outliers and (subspace-)isotropic inliers, i.e.,  $\Sigma_{\text{out}} = \sigma_{\text{out}}^2 I_p/p$  and  $\Sigma_{\text{in}} = \sigma_{\text{in}}^2 U U^\top/q$ .

*PPCA model (Tipping and Bishop, 1999) in the case of the generalized haystack model. Both models are a special case of the principal subspace analysis models (Szwagier and Pennec, 2026). However, as argued in Szwagier and Pennec (2026), while the haystack model is parameterized with Grassmannians, the generalized haystack model—which has more degrees of freedom accounting for the anisotropy—is parameterized with Stiefel manifolds. Therefore, from a statistical modeling perspective, it only makes sense to conduct subspace learning experiments on the haystack model and not the generalized one.*

#### 4.2.1 APPLICATION OF THE FLAG TRICK TO RSR

Among the large family of methods for robust subspace recovery presented in Lerman and Maunu (2018b), we consider the one of *least absolute deviation* (LAD):

$$\arg \min_{\mathcal{S} \in \text{Gr}(p,q)} \sum_{i=1}^n \|x_i - \Pi_{\mathcal{S}} x_i\|_2. \quad (12)$$

It is motivated by the sensitivity of squared norms to outliers, which makes PCA (2) unsuitable for outlier-contaminated data. Problem (12) has the advantage of being rotationally invariant (Ding et al., 2006) but the drawback of being NP-hard (McCoy and Tropp, 2011; Lerman and Maunu, 2018b) and obviously non-convex—since Grassmannians are not. A first body of works relaxes the problem, for instance by optimizing on the convex hull of Grassmannians (McCoy and Tropp, 2011; Xu et al., 2012; Zhang and Lerman, 2014; Lerman et al., 2015). A second body of works directly optimizes the LAD criterion on Grassmannians, either with an IRLS algorithm (Lerman and Maunu, 2018a) or with a geodesic gradient descent (Maunu et al., 2019), both achieving very good results in terms of recovery and speed. The following proposition applies the flag trick to the LAD problem (12).

**Proposition 14 (Flag Trick for RSR)** *The flag trick transforms the least absolute deviation problem (12) into*

$$\arg \min_{\mathcal{S}_{1:d} \in \text{Fl}(p,q_{1:d})} \sum_{i=1}^n \|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2, \quad (13)$$

*which is equivalent to the following optimization problem:*

$$\arg \min_{U_{1:d} \in \text{St}(p,q)} \sum_{i=1}^n \sqrt{\|x_i\|_2^2 - \|U_{1:d}^\top x_i\|_2^2 + \sum_{k=1}^d \left(\frac{k-1}{d}\right)^2 \|U_k^\top x_i\|_2^2}. \quad (14)$$

**Proof** The proof is given in Appendix A. ■

**Remark 15 (Generalization of RSR)** *We can check that Equation (14) generalizes the original RSR problem (12) from Maunu et al. (2019). Indeed, if we consider the particular*

case  $d = 1$ ,  $U := U_{1:d}$ ,  $\mathcal{S} := \text{Span}(U_1)$  (Grassmannian), then we get

$$\sum_{i=1}^n \sqrt{\|x_i\|_2^2 - \|U_{1:d}^\top x_i\|_2^2 + \sum_{k=1}^d \left(\frac{k-1}{d}\right)^2 \|U_k^\top x_i\|_2^2} = \sum_{i=1}^n \sqrt{\|x_i\|_2^2 - \|U_{1:d}^\top x_i\|_2^2 + 0} = \sum_{i=1}^n \|(I_p - UU^\top)x_i\|_2 = \sum_{i=1}^n \|x_i - \Pi_{\mathcal{S}}x_i\|_2. \quad (15)$$

Hence, (14) is a multilevel generalization of the RSR problem from Maunu et al. (2019). The new term under the square root,  $\sum_{k=1}^d \left(\frac{k-1}{d}\right)^2 \|U_k^\top x_i\|_2^2$ , is nonzero when  $d > 1$  (i.e., for flag manifolds refining Grassmannians). This additional term corresponds to a weighted sum over the blocks of the orthonormal frame  $U_{1:d} \in \text{St}(p, q)$ , with increasing weights  $((k-1)/d)^2$ . Therefore, intuitively, the variance in  $\text{Im}(U_k^*)$  is expected to decrease with  $k$ . In other words, the points that are far from the center—i.e., with large norm—should be in the first principal subspaces. Hence, we can interpret our new criterion as a robust generalization of the nested PCA (Theorem 8), with a least influence of outliers due to the square root.

Moreover, we can see that numerical issues are less prone to occur with the flag-tricked problem (13) than with the original one (12). Indeed, the numerical issues of the square root arise when the radicand  $\|x_i - \frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k} x_i\|_2^2$  is close to zero. By nestedness, this is the case if and only if  $\mathcal{S}_1$  is close to  $x_i$ . Consequently, while numerical issues arise in classical RSR (12) whenever the subspace  $\mathcal{S}$  is close to any data point, they arise in flag-tricked RSR (13) whenever both the subspace  $\mathcal{S}$  and the smaller nested subspaces are close to that data point. Hence, whenever  $q_1 := \min q_{1:d}$  is smaller than the dimension  $q$  that one would have tried for classical RSR, the non-differentiability and exploding-gradient issues caused by the square root are less likely to occur.

Finally, since the flag-tricked problem (13) is nothing but a robust version of the nested PCA of Theorem 8 ( $\sum_{i=1}^n \|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2^2$ ), a natural idea can be to initialize the optimization algorithm with the nested PCA solution. This is what is done in Maunu et al. (2019) for the original problem (12), and it is coming with exact recovery guarantees.

#### 4.2.2 NESTEDNESS EXPERIMENTS FOR RSR

We first consider a data set consisting in a mixture of two multivariate Gaussians: the inliers, with zero mean, covariance matrix  $\text{diag}(5, 1, .1)$ ,  $n_{\text{in}} = 450$  and the outliers, with zero mean, covariance matrix  $\text{diag}(.1, .1, 5)$ ,  $n_{\text{out}} = 50$ . The data set is therefore following the generalized haystack model of Maunu et al. (2019). The ambient dimension is  $p = 3$  and the intrinsic dimensions that we try are  $q_{1:2} = (1, 2)$ . We run Algorithm 1 on Grassmann manifolds to solve the LAD problem (12), successively for  $q_1 = 1$  and  $q_2 = 2$ . Then we plot the projections of the data points onto the optimal subspaces. We compare them to the nested projections onto the optimal flag output by running Algorithm 1 on  $\text{Fl}(3, (1, 2))$  to solve (13). The results are shown in Figure 3. We can see that the Grassmann-based projections are non-nested while their flag counterparts are not only nested but also robust to outliers. This could be explained by the nestedness constraint of flag manifolds which imposes the 2D subspace to contain the 1D subspace.

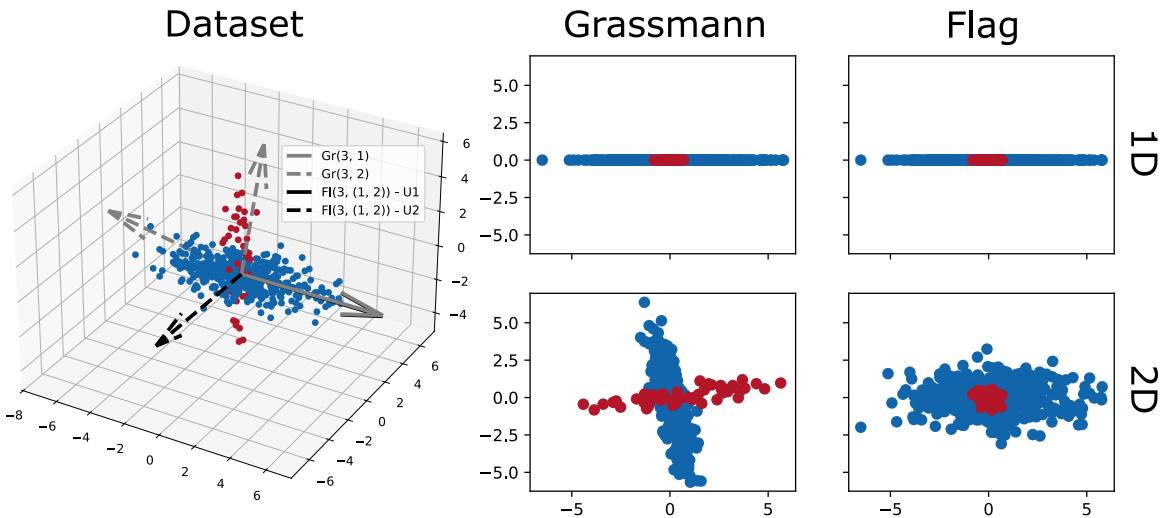


Figure 3: Illustration of the nestedness issue in robust subspace recovery. Given a data set consisting in a mixture of inliers (blue) and outliers (red) we plot its projection onto the optimal 1D subspace and 2D subspace obtained by solving the associated Grassmannian optimization problem (12) or flag optimization problem (13). We see that the Grassmann representations are not nested, while the flag representations are both nested and robust to outliers.

Second, we perform an outlier detection experiment. A common methodology to detect outliers in a corrupted data set is to first look for an outlier-robust subspace and then plot the distribution of distances between the data points and their projection onto the recovered subspace. This distribution is expected to show a clear gap between the inliers and outliers (Vidal et al., 2016, Fig. 3.7). However, in practice, one does not know which subspace dimension  $q$  to choose. If  $q$  is too large, then the recovered subspace may contain both inliers and outliers, and therefore the distribution of distances might be roughly 0. In contrast, if  $q$  is too small, then some inliers may lie too far from the recovered subspace and be detected as outliers. An idea in the spirit of the flag trick is to perform an average ensembling of the reconstruction errors. More specifically, if  $\|x_i - \Pi_{\mathcal{S}} x_i\|_2$  is the classical measure of robust reconstruction error, then we compute  $\|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2$ . Such a score extracts information from projections at different levels and might result in a richer multilevel analysis. We consider a data set where the inliers are images of  $8 \times 8$  handwritten 0's and outliers correspond to other digits from 1 to 9, all extracted from the classical handwritten digits data set (Alpaydin and Kaynak, 1998). The ambient dimension is  $p = 64$ , the number of inliers is  $n_{\text{in}} = 90$  and the number of outliers is  $n_{\text{out}} = 10$ . The intrinsic dimensions that we try are  $q_{1:3} = (1, 2, 5)$ . We plot the distribution of reconstruction errors for the points of the digits data set on the optimal flag  $\mathcal{S}_{1:3}^* \in \text{Fl}(p, (1, 2, 5))$  in Figure 4. We compare it to the distribution of reconstruction errors on  $\mathcal{S}^* \in \text{Gr}(p, 5)$ . We can see that the flag trick enables to clearly distinguish the inliers from the outliers compared to the Grassmann-based method, which is a consequence of the multilevel nature of flags.

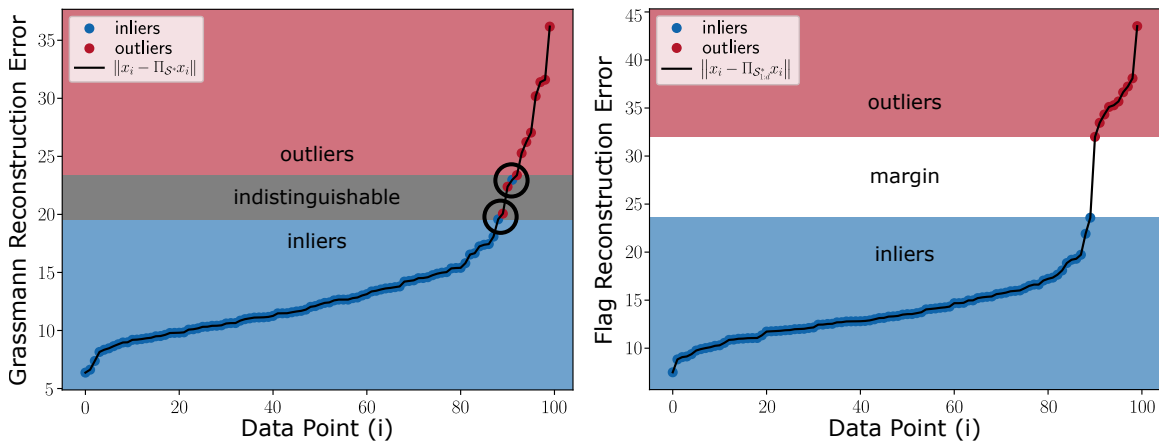


Figure 4: Euclidean reconstruction errors (sorted in ascending order) on the corrupted digits data set for robust subspace recovery ( $\|x_i - \Pi_{\mathcal{S}^*} x_i\|$ , left) and its flag-tricked version ( $\|x_i - \Pi_{\mathcal{S}_{1:d}^*} x_i\|$ , right). With the Grassmann-based method, the distributions of reconstruction errors for the inliers and outliers intersect, meaning that we cannot fully distinguish the inliers from the outliers (see the gray zone on the left plot). In contrast, with the flag-based method, the distributions of reconstruction errors for the inliers and outliers are clearly separated, meaning that we can easily distinguish the inliers from the outliers (see the white zone on the right plot). This phenomenon can be explained by the multilevel nature of the flag trick (see the main text for more details).

#### 4.2.3 DISCUSSION ON RSR OPTIMIZATION AND OBJECTIVE FUNCTIONS

To substantiate the generality of the flag trick, we propose two extensions of the previously proposed multilevel RSR method: an IRLS optimization algorithm and a collection of new multilevel criteria. The experimentation of these extensions is left to future work since the goal is rather to show how the flag trick principle can be easily applied in different contexts.

**An IRLS Algorithm** In all the experiments of this paper, we use a steepest descent method on flag manifolds (Algorithm 1) to solve the flag problems. However, for the specific problem of RSR (13), we believe that more adapted algorithms should be derived, notably due to the non-differentiability and exploding-gradient issues caused by the square root. To that extent, we derive in Appendix A an IRLS scheme (Algorithm 3) for RSR. In short, the RSR problem (13) can be reformulated as a weighted least squares problem  $\sum_{i=1}^n w_i \|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2^2$  with  $w_i = 1/\|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2$  and optimized iteratively, with explicit expressions obtained via our central Theorem 8. We insist on the fact that such an IRLS algorithm could not be developed with the flagfication of Pennecc (2018) and Mankovich et al. (2024), since a sum of square roots does not correspond to a least squares problem.

**More RSR Problems** In this work, we explore one specific problem of RSR for conciseness (12), but we could investigate many other related problems, including robust PCA.

Notably, drawing from the *Grassmann averages* (GA) method (Hauberg et al., 2016), one could develop many new multilevel RSR and Robust PCA objective functions. The idea behind GA is to replace data points with 1D subspaces ( $\mathcal{S}_i = \text{Span}(x_i)$ ) and then perform subspace averaging methods to find a robust prototype for the data set. GA ends up solving problems of the form  $\text{argmin}_{\mathcal{S} \in \text{Gr}(p,1)} \sum_{i=1}^n w_i \text{dist}_{\text{Gr}(p,1)}^2(\text{Span}(x_i), \mathcal{S})$ , where  $w_i$  are some weights and  $\text{dist}_{\text{Gr}(p,1)}$  is a particular subspace distance detailed in Hauberg et al. (2016). Using instead some multidimensional subspace distances, like the principal angles and its variants (Hamm and Lee, 2008; Ye and Lim, 2016), we can develop many variants of the Grassmann averages, of the form  $\text{argmin}_{\mathcal{S}_{1:d} \in \text{Gr}(p,q)} \sum_{i=1}^n w_i \rho(\sqrt{x_i^\top \Pi_{\mathcal{S}} x_i})$ , where  $\rho: \mathbb{R} \rightarrow \mathbb{R}$  is a real function, like  $\rho(x) = \arccos(x)$  if we want subspace-angle-like distances,  $\rho(x) = -x^2$  if we want PCA-like solutions, and many other possible robust variants. Applying the flag trick to those problems yields the following robust multilevel problem:  $\text{argmin}_{\mathcal{S}_{1:d} \in \text{Fl}(p,q_{1:d})} \sum_{i=1}^n w_i \rho(\sqrt{x_i^\top \Pi_{\mathcal{S}_{1:d}} x_i})$ . Similarly, the *dual principal component pursuit* of Tsakiris and Vidal (2018) could benefit from a multilevel version via the flag trick.

### 4.3 The Flag Trick for Trace Ratio Problems

As shown in introduction and particularly in Figure 1, trace ratio methods based on Grassmannian optimization such as linear discriminant analysis might suffer from the nestedness issue. This notably yields inconsistent data representations across different dimensions, as seen on the middle part of the figure, where the clusters get rotated and reflected, leading to cluster identification issues. In this subsection, we address the nestedness issue by applying the flag trick (Definition 10) to trace ratio problems (Ngo et al., 2012).

Trace ratio (TR) problems are ubiquitous in machine learning (Ngo et al., 2012):

$$\text{arg max}_{U \in \text{St}(p,q)} \frac{\text{tr}(U^\top AU)}{\text{tr}(U^\top BU)}, \quad (16)$$

where  $A, B \in \mathbb{R}^{p \times p}$  are positive semi-definite matrices, with  $\text{rank}(B) > p - q$ . A famous example of trace ratio problem is Fisher’s linear discriminant analysis (LDA) (Fisher, 1936; Bellhumeur et al., 1997). It is common in machine learning to project the data onto a low-dimensional subspace before fitting a classifier, in order to circumvent the curse of dimensionality. It is well known that performing an unsupervised dimension reduction method like PCA comes with the risks of mixing up the classes, since the directions of maximal variance are not necessarily the most discriminating ones (Chang, 1983). The goal of LDA is to use the knowledge of the data labels to learn a linear subspace that does not mix the classes. Let  $X := [x_1 \ \cdots \ x_n] \in \mathbb{R}^{p \times n}$  be a data set with labels  $Y := [y_1 \ \cdots \ y_n] \in \{1, \dots, C\}^n$ . Let  $\mu = \frac{1}{n} \sum_{i=1}^n x_i$  be the data set mean and  $\mu_c = \frac{1}{\#\{i: y_i=c\}} \sum_{i: y_i=c} x_i$  be the class-wise means. The idea of LDA is to search for a subspace  $\mathcal{S} \in \text{Gr}(p, q)$  that simultaneously maximizes the projected *between-class variance*  $\sum_{c=1}^C \|\Pi_{\mathcal{S}} \mu_c - \Pi_{\mathcal{S}} \mu\|_2^2$  and minimizes the projected *within-class variance*  $\sum_{c=1}^C \sum_{i: y_i=c} \|\Pi_{\mathcal{S}} x_i - \Pi_{\mathcal{S}} \mu_c\|_2^2$ . This can be reformulated as a trace ratio problem (16), with  $A = \sum_{c=1}^C (\mu_c - \mu)(\mu_c - \mu)^\top$  and  $B = \sum_{c=1}^C \sum_{i: y_i=c} (x_i - \mu_c)(x_i - \mu_c)^\top$ .

More generally, a large family of dimension reduction methods can be reformulated as a TR problem. The seminal work of Yan et al. (2007) shows that many dimension reduction and manifold learning objective functions can be written as a trace ratio involving Laplacian matrices of attraction and repulsion graphs. Intuitively, those graphs determine

which points should be close in the latent space and which ones should be far apart. Other methods involving a ratio of traces are *multi-view learning* (Wang et al., 2023), *partial least squares* (PLS) (Geladi and Kowalski, 1986; Barker and Rayens, 2003) and *canonical correlation analysis* (CCA) (Hardoon et al., 2004), although these methods are originally *sequential* problems (cf. Remark 1) and not *subspace* problems.

Classical Newton-like algorithms for solving the TR problem (16) come from the seminal works of Guo et al. (2003), Wang et al. (2007) and Jia et al. (2009). The interest of optimizing a trace-ratio instead of a ratio-trace (of the form  $\text{tr}((U^\top BU)^{-1}(U^\top AU))$ )—which enjoys an explicit solution given by a generalized eigenvalue decomposition—is also tackled in those papers. The *repulsion Laplacians* (Kokiopoulou and Saad, 2009) instead propose to solve a regularized version  $\text{tr}(U^\top BU) - \rho \text{tr}(U^\top AU)$ , which enjoys an explicit solution but now has a hyperparameter  $\rho$ —whereas the latter is directly optimized in the previous approaches.

#### 4.3.1 APPLICATION OF THE FLAG TRICK TO TRACE RATIO PROBLEMS

The trace ratio problem (16) can be straightforwardly reformulated as an optimization problem on Grassmannians, due to the orthogonal invariance of the objective function:

$$\arg \max_{\mathcal{S} \in \text{Gr}(p,q)} \frac{\text{tr}(\Pi_{\mathcal{S}} A)}{\text{tr}(\Pi_{\mathcal{S}} B)}. \quad (17)$$

The following proposition applies the flag trick to the TR problem (17).

**Proposition 16 (Flag Trick for TR)** *The flag trick transforms the trace ratio problem (17) into*

$$\arg \max_{\mathcal{S}_{1:d} \in \text{Fl}(p,q_{1:d})} \frac{\text{tr}(\Pi_{\mathcal{S}_{1:d}} A)}{\text{tr}(\Pi_{\mathcal{S}_{1:d}} B)}, \quad (18)$$

*which is equivalent to the following optimization problem:*

$$\arg \max_{U_{1:d} \in \text{St}(p,q)} \frac{\sum_{k=1}^d (d - (k - 1)) \text{tr}(U_k^\top A U_k)}{\sum_{l=1}^d (d - (l - 1)) \text{tr}(U_l^\top B U_l)}. \quad (19)$$

**Proof** The proof is given in Appendix B. ■

Equation (19) tells us several things. First, the subspaces  $\text{Span}(U_1) \perp \cdots \perp \text{Span}(U_d)$  are weighted decreasingly, which means that they have less and less importance with respect to the TR objective. Second, we can see that the nested trace ratio problem (18) somewhat maximizes the numerator  $\text{tr}(\Pi_{\mathcal{S}_{1:d}} A)$  while minimizing the denominator  $\text{tr}(\Pi_{\mathcal{S}_{1:d}} B)$ . Both subproblems have an explicit solution corresponding to our nested PCA Theorem 8. Hence, one can naturally initialize the steepest descent algorithm with the  $q$  highest eigenvalues of  $A$  or the  $q$  lowest eigenvalues of  $B$  depending on the application. For instance, for LDA, initializing Algorithm 1 with the highest eigenvalues of  $A$  would spread the classes far apart, while initializing it with the lowest eigenvalues of  $B$  would concentrate the classes, which seems less desirable since we do not want the classes to concentrate at the same point.

## 4.3.2 NESTEDNESS EXPERIMENTS FOR TRACE RATIO PROBLEMS

We run some nestedness and classification experiments for the specific Trace Ratio problem of LDA. Many other applications—like *marginal Fisher analysis* (Yan et al., 2007) and *local discriminant embedding* (Chen et al., 2005)—could be similarly investigated.

First, we consider a synthetic data set with five clusters. The ambient dimension is  $p = 3$  and the intrinsic dimensions that we try are  $q_{1:2} = (1, 2)$ . We adopt a preprocessing strategy similar to Ngo et al. (2012): we first center the data, then run a PCA to reduce the dimension to  $n - C$  (if  $n - C < p$ ), then construct the LDA scatter matrices  $A$  and  $B$ , then add a diagonal covariance regularization of  $10^{-5}$  times their trace and finally normalize them to have unit trace. We run Algorithm 1 on Grassmann manifolds to solve the TR maximization problem (17), successively for  $q_1 = 1$  and  $q_2 = 2$ . Then we plot the projections of the data points onto the optimal subspaces. We compare them to the nested projections onto the optimal flag output by running Algorithm 1 on  $\text{Fl}(3, (1, 2))$  to solve (18). The results are shown in Figure 5. We can see that the Grassmann representations are non-nested while their flag counterparts perfectly capture the filtration of subspaces that best and best approximates the distribution while discriminating the classes. Even if the colors make us realize that the issue in this experiment for LDA is not much about the non-nestedness but rather about the rotation of the principal axes within the 2D subspace, we still have an important issue of consistency.

Second, we consider the (full) handwritten digits data set (Alpaydin and Kaynak, 1998). It contains  $8 \times 8$  pixels images of handwritten digits, from 0 to 9, almost uniformly class-balanced. One has  $n = 1797$ ,  $p = 64$  and  $C = 10$ . We run a steepest descent algorithm to solve the trace ratio problem (18). We choose the *full signature*  $q_{1:d} = (1, 2, \dots, 63)$  with  $d = 63$  and compare the output flag to the individual subspaces output by running optimization on  $\text{Gr}(p, q_k)$  for  $q_k \in q_{1:d}$ . We plot the subspace angles  $\Theta(\mathcal{S}_k^*, \mathcal{S}_{k+1}^*)$  and the explained variance  $\text{tr}(\Pi_{\mathcal{S}_k^*} X X^\top) / \text{tr}(X X^\top)$  as a function of the  $k$ . The results are illustrated in Figure 6. We see that the subspace angles are always positive and even very large sometimes with the LDA. Worst, the explained variance is not monotonic. This implies that we sometimes *lose* some information when *increasing* the dimension, which is extremely paradoxical.

Third, we perform some classification experiments on the optimal subspaces. For each data set, we run the optimization problems on  $\text{Fl}(p, q_{1:d})$ , then project the data onto the different subspaces in  $\mathcal{S}_{1:d}^*$  and run a nearest neighbors classifier with 5 neighbors. The predictions are then ensembled (cf. Algorithm 2) by weighted averaging, either with uniform weights or with weights minimizing the average cross-entropy:

$$w_1^*, \dots, w_d^* = \arg \min_{\substack{w_k \geq 0 \\ \sum_{k=1}^d w_k = 1}} -\frac{1}{nC} \sum_{i=1}^n \sum_{c=1}^C y_{ic} \ln \left( \sum_{k=1}^d w_k y_{kic}^* \right), \quad (20)$$

where  $y_{kic}^* \in [0, 1]$  is the predicted probability that  $x_i \in \mathbb{R}^p$  belongs to class  $c \in \{1, \dots, C\}$ , by the classifier  $g_k^*$  that is trained on  $Z_k := U_k^{*\top} X \in \mathbb{R}^{q_k \times n}$ . One can show that the latter is a convex problem, which we optimize using the cvxpy Python package (Diamond and Boyd, 2016). We repeat the experiment 10 times in a stratified train-test fashion and report the average test cross-entropy in Table 4.3.2. The wine example is particularly interesting. It

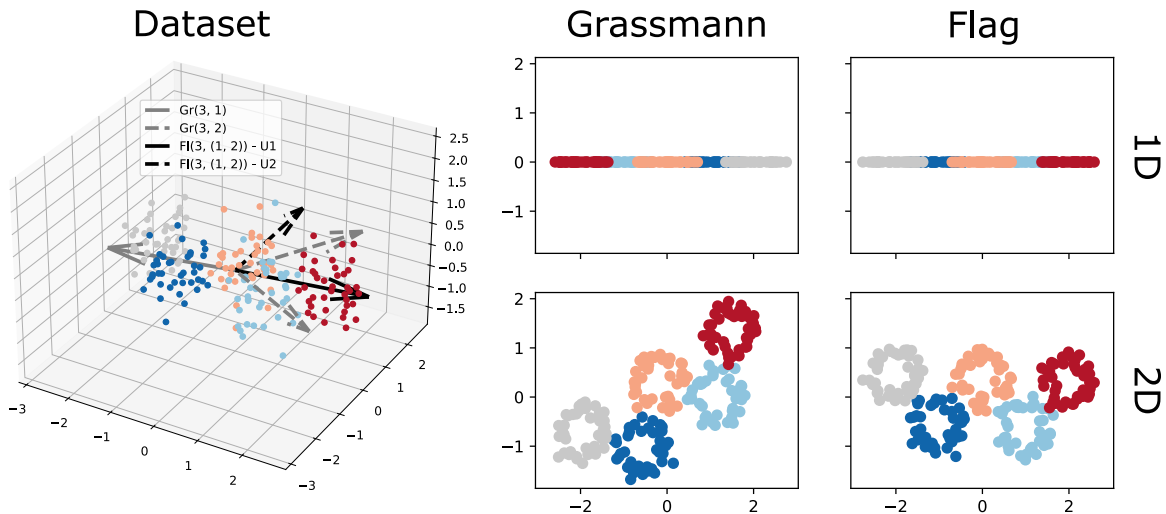


Figure 5: Illustration of the nestedness issue in linear discriminant analysis (trace ratio problem). Given a data set with five clusters, we plot its projection onto the optimal 1D subspace and 2D subspace obtained by solving the associated Grassmannian optimization problem (17) or flag optimization problem (18). We see that the Grassmann representations are not nested, while the flag representations are nested and well capture the distribution of clusters. In this example, it is less the nestedness than the *rotation* of the optimal axes inside the 2D subspace that is critical to the analysis of the Grassmann-based method.

data set	$n$	$p$	$q_{1:d}$	Gr	Fl	Fl-U	Fl-W	weights
digits	1797	64	(1, 2, 5, 10)	5.1	4.6	3.1	2.9	(0.03, 0.05, 0.30, 0.62)
wine	178	13	(1, 2, 5)	0.71	0.69	0.43	0.29	(0, 0.68, 0.32)
breast	569	30	(1, 2, 5)	0.534	0.537	0.485	0.475	(0.26, 0.24, 0.50)
iris	150	4	(1, 2, 3)	0.275	0.271	0.281	0.265	(0.27, 0.12, 0.62)

Table 1: Results of the TR-based classification experiment. For each method—Gr: Grassmann optimization (17), Fl: flag optimization (18), Fl-U: flag optimization + uniform soft voting, Fl-W: flag optimization + optimal soft voting (20)—we give the average test cross-entropy between the predictions and the true labels.

first tells us that the optimal 5D subspace obtained by Grassmann optimization discriminates less between the classes than the 5D subspace from the optimal flag. This may show that the flag takes into account some lower-dimensional variability that enables to better discriminate the classes. Second, the uniform averaging of the predictors at different dimensions improves the classification. Third, the optimal weights improve even more the

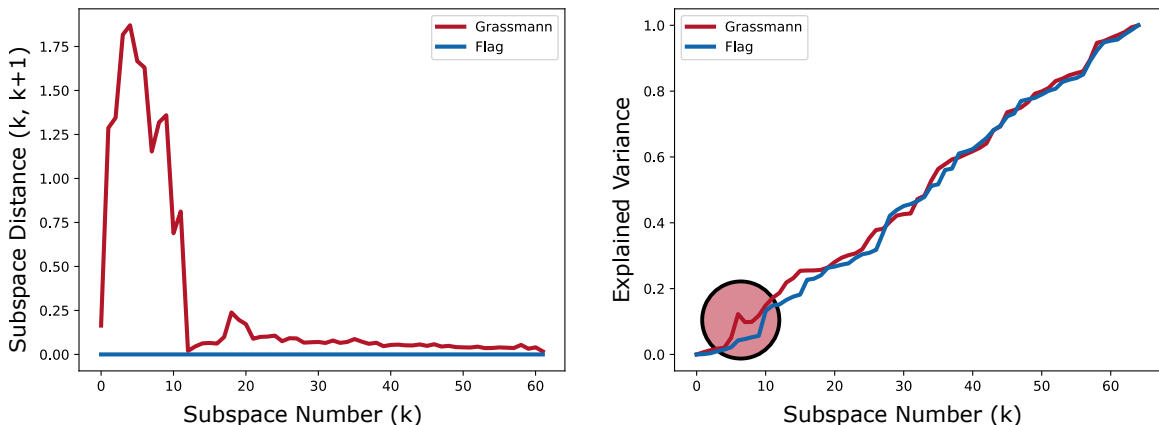


Figure 6: Illustration of the nestedness issue in linear discriminant analysis (trace ratio problem) on the digits data set. For  $q_k \in (1, 2, \dots, 63)$ , we solve the Grassmannian optimization problem (17) on  $\text{Gr}(64, q_k)$  and plot the subspace angles  $\Theta(\mathcal{S}_k^*, \mathcal{S}_{k+1}^*)$  (left) and explained variances  $\text{tr}(\Pi_{\mathcal{S}_k^*} X X^\top) / \text{tr}(X X^\top)$  (right) as a function of  $k$ . We compare those quantities to the ones obtained by solving the flag optimization problem (18). We see that the Grassmann-based method is highly non-nested and even yields an extremely paradoxical non-increasing explained variance (cf. red circle on the right).

classification and tell that the best discrimination occurs by taking a soft blend of classifiers at dimensions 2 and 5. Similar kinds of analyses can be made for the other examples.

#### 4.3.3 DISCUSSION ON TR OPTIMIZATION AND KERNELIZATION

To substantiate the generality of the flag trick, we propose two extensions of the previously proposed multilevel TR method: a Newton optimization algorithm and a kernelization. The experimentation of these extensions is left to future work since the goal is rather to show how the flag trick principle can be easily applied in different contexts.

**A Newton Algorithm** In all the experiments of this paper, we use a steepest descent method on flag manifolds (Algorithm 1) to solve the flag problems. However, for the specific problem of TR (18), we believe that more adapted algorithms should be derived to take into account the specific form of the objective function, which is classically solved via a Newton-Lanczos method (Ngo et al., 2012). To that extent, we develop in the appendix (B) an extension of the baseline Newton-Lanczos algorithm for the flag-tricked problem (18). In short, the latter can be reformulated as a penalized optimization problem of the form  $\text{argmax}_{\mathcal{S}_{1:d} \in \text{Fl}(p, q_{1:d})} \sum_{k=1}^d \text{tr}(\Pi_{\mathcal{S}_k} (A - \rho B))$ , where  $\rho$  is updated iteratively according to a Newton scheme. Once again, our central Theorem 8 enables to get an explicit solution to the penalized optimization problem.

**A Non-Linearization via the Kernel Trick** The classical trace ratio problems look for *linear* embeddings of the data. However, in most cases, the data follow a *nonlinear* distribution, which may cause linear dimension reduction methods to fail. The *kernel trick* (Hofmann et al., 2008) is a well-known method to embed nonlinear data into a linear space and fit linear machine learning methods. As a consequence, we propose in appendix (B) a kernelization of the trace ratio problem (18) in the same fashion as the one of the seminal graph embedding work (Yan et al., 2007). This is expected to yield much better embedding and classification results.

#### 4.4 The Flag Trick for Spectral Clustering Problems

As shown in introduction and particularly in Figure 1, spectral clustering methods based on Grassmannian optimization might suffer from the nestedness issue, which notably yields inconsistent data representations across different dimensions. This can be seen on the right part of the figure, where the clusters are almost separable in 1D but surprisingly intertwine in 2D. In this subsection, we address the nestedness issue by applying the flag trick (Definition 10) to Grassmann-based sparse spectral clustering (Wang et al., 2017).

Spectral clustering (Ng et al., 2001) is a baseline clustering technique. It can be applied to general cluster distributions compared to the classical k-means (round clusters) and Gaussian mixture models (ellipsoidal clusters). Given a data set  $X := [x_1 \ \cdots \ x_n] \in \mathbb{R}^{p \times n}$ , spectral clustering roughly consists in the eigenvalue decomposition of a Laplacian matrix  $L \in \text{Sym}(n)$  associated to a pairwise similarity matrix, for instance  $M_{ij} = \exp(\|x_i - x_j\|^2 / 2\sigma^2)$ . The eigenvectors are then used as new embeddings for the data points, on which standard clustering algorithms like k-means can be performed. This method is closely related to the celebrated Laplacian eigenmaps (Belkin and Niyogi, 2003) which are used for nonlinear dimension reduction. The good performances of such a simple method as spectral clustering are theoretically justified by the particular structure of the Laplacian matrix  $L$  in an ideal case—block-diagonal with a multiple eigenvalue related to the number of clusters (Ng et al., 2001). The recent *sparse spectral clustering* (SSC) method (Lu et al., 2016) builds on such an ideal case and encourages the block-diagonality by looking for a sparse and low-rank representation:

$$\arg \min_{U \in \text{St}(n,q)} \langle UU^\top, L \rangle_F + \beta \left\| UU^\top \right\|_1, \quad (21)$$

that they optimize over the convex hull of Grassmann manifolds with an ADMM algorithm.

##### 4.4.1 APPLICATION OF THE FLAG TRICK TO SPECTRAL CLUSTERING

The Grassmannian spectral clustering method (Wang et al., 2017) directly optimizes (21) on Grassmann manifolds:

$$\arg \min_{S \in \text{Gr}(n,q)} \langle \Pi_S, L \rangle_F + \beta \|\Pi_S\|_1. \quad (22)$$

The authors use a Riemannian trust region method (Absil et al., 2008) for optimization and show the interest of directly optimizing over the Grassmann manifold instead of convexifying the optimization space (Lu et al., 2016). The following proposition applies the flag trick to sparse spectral clustering.

**Proposition 17 (Flag Trick for SSC)** *The flag trick transforms the sparse spectral clustering problem (22) into*

$$\arg \min_{\mathcal{S}_{1:d} \in \text{Fl}(n, q_{1:d})} \langle \Pi_{\mathcal{S}_{1:d}}, L \rangle_F + \beta \|\Pi_{\mathcal{S}_{1:d}}\|_1, \quad (23)$$

which is equivalent to the following optimization problem:

$$\arg \min_{U_{1:d} \in \text{St}(n, q)} \sum_{k=1}^d (d - (k - 1)) \text{tr} \left( U_k^\top L U_k \right) + \beta \left\| \sum_{k=1}^d (d - (k - 1)) U_k U_k^\top \right\|_1. \quad (24)$$

**Proof** The proof is given in Appendix C. ■

We can see that the case  $\beta = 0$  corresponds to classical spectral clustering. Indeed, with a similar reasoning as in the proof of Theorem 8, we can easily show that the solution to problem (23) is explicit and corresponds to the flag of nested eigenspaces of  $L$  (for increasing eigenvalues). Therefore, initializing the algorithm with the smallest  $q$  eigenvectors of  $L$  seems like a natural idea. Moreover, one may intuitively analyze the relative weighting of the mutually-orthogonal subspaces  $\text{Span}(U_1) \perp \cdots \perp \text{Span}(U_d)$  in (24) as favoring a model with  $q_1$  clusters, and then adding successively  $q_k - q_{k-1}$  clusters to improve the modeling of the Laplacian matrix, with a tradeoff between too much and too few clusters.

#### 4.4.2 NESTEDNESS EXPERIMENTS FOR SPECTRAL CLUSTERING

We run some nestedness experiments for the sparse spectral clustering problem. First, we consider a 3D extension of the classical two-moons data set for clustering with  $n = 100$ . The ambient dimension is  $p = 3$  and the intrinsic dimensions that we try are  $q_{1:2} = (1, 2)$ . We adopt a pre-processing strategy similar to Lu et al. (2016) and Wang et al. (2017): we compute the affinity matrix  $W \in \mathbb{R}^{n \times n}$  using an exponential kernel with standard deviation being the median of the pairwise Euclidean distances between samples. Then we compute the normalized Laplacian matrix  $L = I_n - D^{-\frac{1}{2}} L D^{-\frac{1}{2}}$  where  $D \in \mathbb{R}^{n \times n}$  is a diagonal matrix with diagonal elements  $D_{ii} = \sum_{j=1}^n w_{ij}$ . We run Algorithm 1 on Grassmann manifolds to solve the sparse optimization problem (22), successively for  $q_1 = 1$  and  $q_2 = 2$ . Then we plot the projections of the data points onto the optimal subspaces. We compare them to the nested projections onto the optimal flag output by running Algorithm 1 on  $\text{Fl}(3, (1, 2))$  to solve (23). The results are shown in Figure 7. We can see that the Grassmann representations are not only non-nested but also mix the two clusters in 2D, while the flag representations are nested and much better discriminate the clusters. This could be explained by the nestedness constraint of flag manifolds which imposes the 2D subspace to contain the 1D subspace.

Second, we consider the breast cancer data set (Wolberg et al., 1993). It contains 569 samples—from which we extract a subset of 100 samples for computational time—with 30 numerical attributes and two classes. One has  $n = 100$ ,  $p = 30$  and  $C = 2$ . Then we run the steepest descent algorithm 1 to solve the SSC problem (23). We choose the *full signature*  $q_{1:99} = (1, 2, \dots, 99)$  and compare the output flag to the individual subspaces output by running optimization on  $\text{Gr}(n, q_k)$  for  $q_k \in q_{1:d}$ . We perform scatter plots in 1D and 2D and subspace error computations as a function of the  $q_k$ . The results are illustrated in

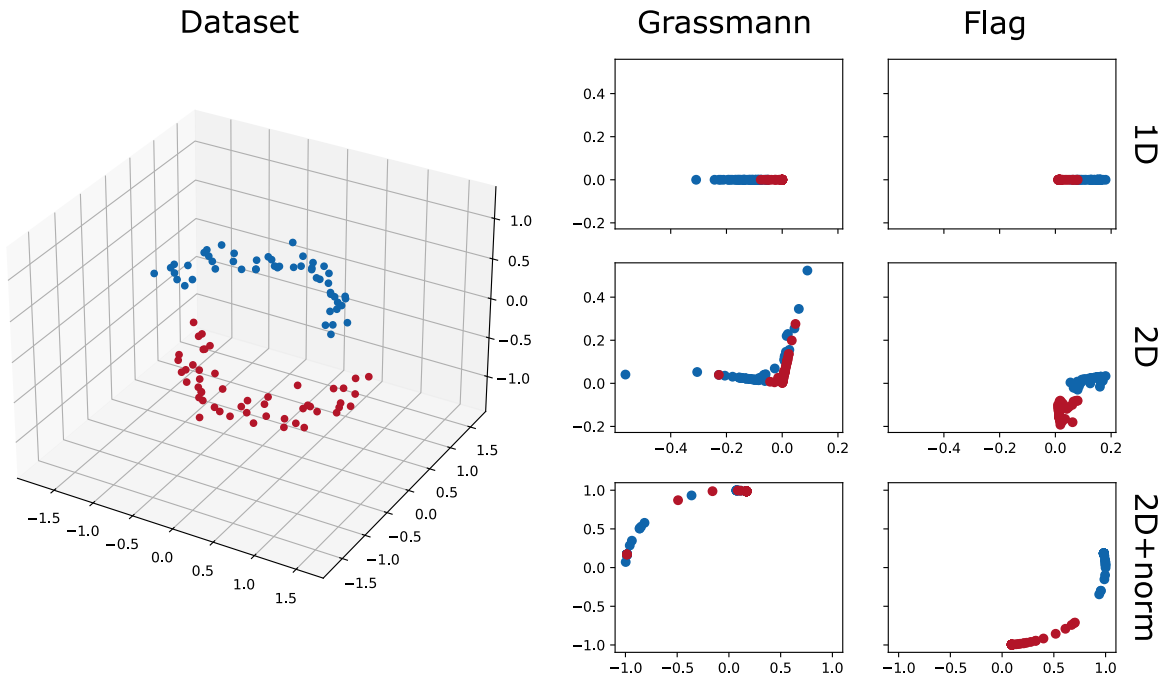


Figure 7: Illustration of the nestedness issue in sparse spectral clustering. Given a data set with two half-moon-shaped clusters, we plot its projection onto the optimal 1D subspace and 2D subspace obtained by solving the associated Grassmannian optimization problem (22) or flag optimization problem (23). We see that the Grassmann representations are not nested, while the flag representations are nested and better clustered. The last row corresponds to dividing the 2D embeddings by their norms, as commonly done in spectral clustering (Ng et al., 2001).

Figure 8. The subspace angle plot tells us that the Grassmann spectral clustering yields highly non-nested subspaces, while the flags are by nature nested. The scatter plots show that the Grassmann representations are totally inconsistent, while the flag representations are consistent with the cluster structures.

#### 4.4.3 DISCUSSION ON SSC OPTIMIZATION

Similarly to the previous examples, we can see that the steepest descent algorithm 1 might not be the most suited to solve the optimization problem (23), notably due to the  $\ell^1$ -penalization. Contrary to robust subspace recovery, where we want to avoid the non-differentiable points (cf. Section 4.2.3), here we would ideally like to exactly attain those points, since they encode the desired sparsity of the Laplacian matrix. Therefore, we believe that sparsity-inducing optimization algorithms (Bach et al., 2012) on flag manifolds would be more adapted than the steepest descent we use.

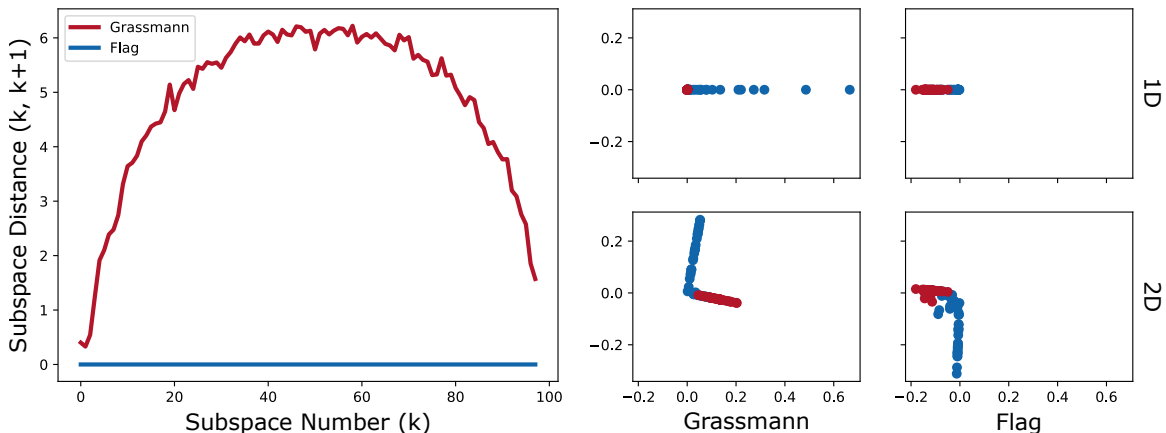


Figure 8: Illustration of the nestedness issue in sparse spectral clustering on the breast cancer data set. On the left, for  $q_k \in (1, 2, \dots, 99)$ , we solve the Grassmannian optimization problem (22) on  $\text{Gr}(100, q_k)$  and plot the subspace angles  $\Theta(\mathcal{S}_k^*, \mathcal{S}_{k+1}^*)$  as a function of  $k$ . On the right, we plot the projection of the data set onto the output 1D and 2D subspaces. We compare those quantities to the ones obtained by solving the flag optimization problem (23). We see that the Grassmann-based method is highly non-nested while the flag-tricked one is not only nested but also seems to be coherent with the cluster structure: the first axis is for the red cluster while the second axis is for the blue cluster.

#### 4.5 The Flag Trick for Other Machine Learning Problems

Subspace learning finds many applications beyond robust subspace recovery, trace ratio and spectral clustering problems, as evoked in Section 1. The goal of this subsection is to provide a few more examples in brief, without experiments.

##### 4.5.1 DOMAIN ADAPTATION

In machine learning, it is often assumed that the training and test data sets follow the same distribution. However, some *domain shift* issues—where training and test distributions are different—might arise, notably if the test data have been acquired from a different source (for instance a professional camera and a phone camera) or if the training data have been acquired a long time ago. *Domain adaptation* is an area of machine learning that deals with domain shifts, usually by matching the training and test distributions—often referred to as *source* and *target* distributions—before fitting a classical model (Farahani et al., 2021). A large body of works (called “subspace-based”) learn some intermediary subspaces between the source and target data, and perform the inference for the projected target data on these subspaces. The *sampling geodesic flow* (Gopalan et al., 2011) first performs a geodesic interpolation on Grassmannians between the source and target subspaces, then projects both data sets on (a discrete subset of) the interpolated subspaces, which results in a new representation of the data distributions, that can then be given as an input to a machine learning

model. The higher the number of intermediary subspaces, the better the approximation, but the larger the dimension of the representation. The celebrated *geodesic flow kernel* (Gong et al., 2012) circumvents this issue by integrating the projected data onto the continuum of interpolated subspaces. This yields an inner product between infinite-dimensional embeddings that can be computed explicitly and incorporated in a kernel method for learning. The *domain invariant projection* (Baktashmotlagh et al., 2013) learns a *domain-invariant* subspace that minimizes the maximum mean discrepancy (MMD) (Gretton et al., 2012) between the projected source  $X_s := [x_{s1} \ \cdots \ x_{sn_s}] \in \mathbb{R}^{p \times n_s}$  and target distributions  $X_t := [x_{t1} \ \cdots \ x_{tn_t}] \in \mathbb{R}^{p \times n_t}$ :  $\operatorname{argmin}_{U \in \operatorname{St}(p,q)} \operatorname{MMD}^2(U^\top X_s, U^\top X_t)$ , where  $\operatorname{MMD}(X, Y) = \left\| \frac{1}{n} \sum_{i=1}^n \phi(x_i) - \frac{1}{m} \sum_{i=1}^m \phi(y_i) \right\|_{\mathcal{H}}$ . Following Baktashmotlagh et al. (2013, Eq. 4), we can rewrite the unsupervised domain adaptation problem as

$$\operatorname{arg min}_{\mathcal{S} \in \operatorname{Gr}(p,q)} \left( \frac{1}{n_s^2} \sum_{i,j=1}^{n_s} \exp \left( -\frac{z_{ij}^s \top \Pi_{\mathcal{S}} z_{ij}^s}{2\sigma^2} \right) + \frac{1}{n_t^2} \sum_{i,j=1}^{n_t} \exp \left( -\frac{z_{ij}^t \top \Pi_{\mathcal{S}} z_{ij}^t}{2\sigma^2} \right) - \frac{2}{n_s n_t} \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} \exp \left( -\frac{z_{ij}^{st} \top \Pi_{\mathcal{S}} z_{ij}^{st}}{2\sigma^2} \right) \right), \quad (25)$$

where  $z_{ij}^s := x_{si} - x_{sj}$ ,  $z_{ij}^t := x_{ti} - x_{tj}$  and  $z_{ij}^{st} := x_{si} - x_{tj}$ . One can easily apply the flag trick (11) to this problem to make domain adaptation multilevel:

$$\operatorname{arg min}_{\mathcal{S}_{1:d} \in \operatorname{Fl}(p,q_{1:d})} \left( \frac{1}{n_s^2} \sum_{i,j=1}^{n_s} \exp \left( -\frac{z_{ij}^s \top \Pi_{\mathcal{S}_{1:d}} z_{ij}^s}{2\sigma^2} \right) + \frac{1}{n_t^2} \sum_{i,j=1}^{n_t} \exp \left( -\frac{z_{ij}^t \top \Pi_{\mathcal{S}_{1:d}} z_{ij}^t}{2\sigma^2} \right) - \frac{2}{n_s n_t} \sum_{i=1}^{n_s} \sum_{j=1}^{n_t} \exp \left( -\frac{z_{ij}^{st} \top \Pi_{\mathcal{S}_{1:d}} z_{ij}^{st}}{2\sigma^2} \right) \right). \quad (26)$$

Some experiments similar to the ones of Baktashmotlagh et al. (2013) can be performed. For instance, one can consider the benchmark visual object recognition data set of Saenko et al. (2010), learn nested domain invariant projections, fit some support vector machines to the projected source samples at increasing dimensions, and then perform soft-voting ensembling by learning the optimal weights on the target data according to Equation (20).

#### 4.5.2 LOW-RANK DECOMPOSITION

Many machine learning methods involve finding low-rank representations of a data matrix. This is the case of *matrix completion* (Candès and Recht, 2012) where one looks for a low-rank representation of an incomplete data matrix by minimizing the discrepancy with the observed entries, and which finds many applications including the well-known Netflix problem. Although its most-known formulation is as a convex relaxation, it can also be formulated as an optimization problem on Grassmann manifolds (Keshavan et al., 2010; Boumal and Absil, 2011) to avoid optimizing the nuclear norm in the full space which can be of high dimension. The intuition is that a low-dimensional point can be described by the subspace it belongs to and its coordinates within this subspace. More precisely, the

SVD-based low-rank factorization  $M = UW$ , with  $M \in \mathbb{R}^{p \times n}$ ,  $U \in \text{St}(p, q)$  and  $W \in \mathbb{R}^{q \times n}$  is orthogonally-invariant—in the sense that for any  $R \in \mathcal{O}(q)$ , one has  $(UR)(R^\top W) = UW$ . One could therefore apply the flag trick to such problems, with the intuition that we would try low-rank matrix decompositions at different dimensions. The application of the flag trick would however not be as straightforward as in the previous problems since the subspace-projection matrices  $\Pi_S := UU^\top$  do not appear explicitly, and since the coefficient matrix  $W$  also depends on the dimension  $q$ .

Many other low-rank problems can be formulated as a Grassmannian optimization. *Robust PCA* (Candès et al., 2011) looks for a low rank + sparse corruption factorization of a data matrix. *Subspace Tracking* (Balzano et al., 2010) incrementally updates a subspace from streaming and highly-incomplete observations via small steps on Grassmann manifolds.

### 4.5.3 OTHER SUBSPACE LEARNING PROBLEMS

Finally, many other general machine learning problems involve optimization on Grassmannians. For instance, *linear dimensionality reduction* (Cunningham and Ghahramani, 2015) encompasses the already-discussed PCA and LDA, but also many other problems like *multi-dimensional scaling* (Torgerson, 1952), *slow feature analysis* (Wiskott and Sejnowski, 2002) and *locality preserving projections* (He and Niyogi, 2003). All these machine learning problems involving optimization on Grassmannians can, likewise, be turned into multilevel problems via the flag trick.

## 5. Discussion

We introduced a general principle to make subspace learning problems multilevel and nested. The non-nestedness issue of Grassmannian-based methods was clearly demonstrated on a variety of machine learning problems, yielding inconsistent scatter plots and paradoxical learning curves. The flag trick was shown to both solve this nestedness issue and output filtrations of subspaces that meet the original goals of the problems. As a positive side-effect, the flag trick sometimes even output substantially-better subspaces than the original subspace problems, due to its multilevel nature which enables to aggregate information from several dimensions. When combined with ensemble learning, the flag trick showed an overall improvement over the subspace-based predictors that work at a fixed dimension and it raised some interesting dimension-blending perspectives that question the manifold hypothesis. The code for the experiments can be found at <https://github.com/tomswagier/flag-trick>.

One major limitation of this paper is the lack of in-depth study of the specific interest of the flag trick to each subspace learning method. We indeed focus on general practical results—the nestedness of flag-based methods with respect to classical subspace methods and the interest of ensembling dimensions instead of considering each dimension individually—and not on some application-specific questions—like the robustness of the flag to outliers for robust subspace recovery, the quality of the embedding for trace ratio problems, and the quality of the clustering for spectral clustering problems. This paper should thus be viewed as an invitation to the various communities to explore multilevel extensions of subspace learning, rather than as a definitive solution to a long-standing challenge in machine learning. To that extent, we end this paper with a list of perspectives.

First, one should perform in-depth experiments to show the interest and the limitations of the flag trick for each specific subspace learning problem among—but not limited to—the list of examples developed in Section 4. Throughout this paper, we focused on linear flags, but nonlinear extensions could be considered to improve expressivity. Beyond the kernel trick considered in Section B.2 for nonlinear graph embedding, a general methodology based on data mappings to linear spaces—such as the latent space of a deep generative model or the tangent space at the Fréchet mean of a manifold-valued data set—could be simply implemented. More advanced methodologies relying on nested submanifolds, borrowed from the geometric statistics community (see Section 1), could be investigated as optimization problems on flag manifolds; typical examples are the principal nested spheres from Jung et al. (2012), which can be seen as intersections between spheres and nested hyperplanes.

Second, one should experiment with more complex and more efficient optimization algorithms on flag manifolds such as the ones developed in the appendix (IRLS in Appendix A and Newton in Appendix B) or the ones proposed in Ye et al. (2022), Nguyen (2022) and Zhu and Shen (2024), and develop new ones that are specifically adapted to the properties of the objective function. We invite the interested researchers from the manifold optimization community to test their existing algorithms (and develop new ones) on our statistically-grounded optimization problems (13), (18), (23) and (26) on flag manifolds.

Third, we derived a very general principle for transitioning from fixed-dimension subspace methods to increasing-dimension flag methods, but this principle could be revisited specifically for each problem. This includes the problems that are not specifically formulated as a Grassmannian optimization, as long as they somewhat involve low-dimensional subspaces, like in domain adaptation (Gopalan et al., 2011; Gong et al., 2012) and sparse subspace clustering (Elhamifar and Vidal, 2013).

Fourth, we leave the question of the choice of flag signature (i.e., the sequence of dimensions to try) open in this paper (see, notably, Remark 12). A first step for PCA in the Gaussian case has been achieved with the principal subspace analysis of Szwagier and Pennec (2026), where the dimensions are chosen based on the relative eigengaps of the covariance matrix. Other generative models could be designed specifically for each subspace learning problem, and the recent penalty of Szwagier et al. (2025) on the spectral gaps could be added to automatically select the flag signature.

## Acknowledgments

The authors would like to thank the editorial board, the action editor Aapo Hyvärinen, the anonymous reviewers and Pierre-Antoine Absil for their constructive comments and ideas which have helped improve the quality of the paper. This work was supported by the ERC grant #786854 G-Statistics from the European Research Council under the European Union’s Horizon 2020 research and innovation program and by the French government through the 3IA Côte d’Azur Investments ANR-23-IACL-0001 managed by the National Research Agency.

## Appendix A. Robust Subspace Recovery: Extensions and Proofs

In this section, we develop a reweighted least squares algorithm for robust subspace recovery (13) and we provide the proof of Proposition 14.

### A.1 An IRLS Algorithm for Robust Subspace Recovery

Iteratively reweighted least squares (IRLS) is a ubiquitous method to solve optimization problems involving  $\ell^p$ -norms. Motivated by the computation of the geometric median (Weiszfeld, 1937), IRLS is highly used to find robust maximum likelihood estimates of non-Gaussian probabilistic models (typically those containing outliers) and finds application in robust regression (Huber, 1964), sparse recovery (Daubechies et al., 2010), etc.

The recent fast median subspace (FMS) algorithm of Lerman and Maunu (2018a), achieving state-of-the-art results in RSR, uses an IRLS scheme to optimize the absolute deviation (12). The idea is to first rewrite the absolute deviation as

$$\sum_{i=1}^n \|x_i - \Pi_{\mathcal{S}} x_i\|_2 = \sum_{i=1}^n w_i(\mathcal{S}) \|x_i - \Pi_{\mathcal{S}} x_i\|_2^2, \quad (27)$$

with  $w_i(\mathcal{S}) = \frac{1}{\|x_i - \Pi_{\mathcal{S}} x_i\|_2}$ , and then successively compute the weights  $w_i$  and update the subspace according to the weighted objective. More precisely, the FMS algorithm creates a sequence of subspaces  $\mathcal{S}^1, \dots, \mathcal{S}^m$  such that

$$\mathcal{S}^{t+1} = \arg \min_{\mathcal{S} \in \text{Gr}(p,q)} \sum_{i=1}^n w_i(\mathcal{S}^t) \|x_i - \Pi_{\mathcal{S}} x_i\|_2^2. \quad (28)$$

This weighted least-squares problem enjoys a closed-form solution which relates to the eigenvalue decomposition of the weighted covariance matrix  $\sum_{i=1}^n w_i(\mathcal{S}^t) x_i x_i^\top$  (Vidal et al., 2016, Chapter 3.3).

We wish to derive an IRLS algorithm for the flag-tricked version of the LAD problem (13). In order to stay close in mind to the recent work of Peng et al. (2023) who proved convergence of a general class of IRLS algorithms under some mild assumptions, we first rewrite (13) as

$$\arg \min_{\mathcal{S}_{1:d} \in \text{Fl}(p,q_{1:d})} \sum_{i=1}^n \rho(r(\mathcal{S}_{1:d}, x_i)), \quad (29)$$

where  $r(\mathcal{S}_{1:d}, x) = \|x - \Pi_{\mathcal{S}_{1:d}} x\|_2$  is the *residual* and  $\rho(r) = |r|$  is the *outlier-robust* loss function. Following Peng et al. (2023), the IRLS scheme associated with (29) is

$$\begin{cases} w_i^{t+1} = \rho'(r(\mathcal{S}_{1:d}^t, x_i)) / r(\mathcal{S}_{1:d}^t, x_i) = 1 / \|x_i - \Pi_{\mathcal{S}_{1:d}^t} x_i\|_2, \\ (\mathcal{S}_{1:d}^t)^{t+1} = \arg \min_{\mathcal{S}_{1:d} \in \text{Fl}(p,q_{1:d})} \sum_{i=1}^n w_i^{t+1} \|x_i - \Pi_{\mathcal{S}_{1:d}^t} x_i\|_2^2. \end{cases} \quad (30)$$

We now show that the second step enjoys a closed-form solution.

**Proposition 18 (Reweighted Nested PCA)** *The RLS problem*

$$\arg \min_{\mathcal{S}_{1:d} \in \text{Fl}(p,q_{1:d})} \sum_{i=1}^n w_i \|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2^2 \quad (31)$$

has a closed-form solution  $\mathcal{S}_{1:d}^* \in \text{Fl}(p, q_{1:d})$ , which is given by the eigenvalue decomposition of the weighted sample covariance matrix  $S_w = \sum_{i=1}^n w_i x_i x_i^\top = \sum_{j=1}^p \ell_j v_j v_j^\top$ , i.e.,

$$\mathcal{S}_k^* = \text{Span}(v_1, \dots, v_{q_k}) \quad (k \in \{1, \dots, d\}). \quad (32)$$

**Proof** One has

$$\sum_{i=1}^n w_i \|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2^2 = \text{tr} \left( (I - \Pi_{\mathcal{S}_{1:d}})^2 \left( \sum_{i=1}^n w_i x_i x_i^\top \right) \right). \quad (33)$$

Therefore, we are exactly in the same case as in Theorem 8, if we replace  $XX^\top$  with the reweighted covariance matrix  $\sum_{i=1}^n w_i x_i x_i^\top$ . This does not change the result, so we conclude with the end of the proof of Theorem 8 (itself relying on Szwagier and Penne, 2026). ■

Hence, one gets an IRLS scheme for the LAD problem. One can modify the robust loss function  $\rho(r) = |r|$  by a Huber-like loss function to avoid weight explosion. Indeed, one can show that the weight  $w_i := 1/\|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2$  goes to infinity when the first subspace  $\mathcal{S}_1$  of the flag gets close to  $x_i$ . Therefore in practice, we take

$$\rho(r) = \begin{cases} r^2/(2p\delta) & \text{if } |r| \leq p\delta, \\ r - p\delta/2 & \text{if } |r| > p\delta, \end{cases} \quad (34)$$

which yields

$$w_i = 1/\max(p\delta, \|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2). \quad (35)$$

The final proposed scheme is given in Algorithm 3, named *fast median flag* (FMF), in reference to the fast median subspace algorithm of Lerman and Maunu (2018a).

---

**Algorithm 3** Fast Median Flag

---

**Require:**  $X := [x_1 \ \dots \ x_n] \in \mathbb{R}^{p \times n}$  a data matrix;  $q_{1:d} := (q_1, \dots, q_d)$  a flag signature;  $\varepsilon > 0$  a saturation parameter

$U_{1:d} \leftarrow \text{SVD}(X, q_d)$  ▷ top  $q_d$  left singular vectors (least squares)

**for**  $t \in \{1, 2, \dots\}$  **do**

$\Pi_{\mathcal{S}_{1:d}} \leftarrow \frac{1}{d} \sum_{k=1}^d (d - k + 1) U_k U_k^\top$  ▷ multilevel projector (flag trick)

$w_i \leftarrow 1/\max(\varepsilon, \|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2)$  ▷ weights (saturated)

$S_w \leftarrow \sum_{i=1}^n w_i x_i x_i^\top$  ▷ reweighted covariance

$U_{1:d} \leftarrow \text{EVD}(S_w, q_d)$  ▷ top  $q_d$  eigenvectors

**end for**

**Ensure:**  $U_{1:d}^* \in \text{Fl}(p, q_{1:d})$  an optimal flag (Stiefel representation)

---

We can easily check that FMF is a direct generalization of FMS for Grassmannians (i.e., when  $d = 1$ ).

**Remark 19 (Convergence Analysis)** *Although beyond the scope of the paper, we believe that the convergence result of Peng et al. (2023, Theorem 1) could be generalized to the FMF algorithm, due to the compactness of flag manifolds and the expression of the residual function  $r$ .*

## A.2 Proof of Proposition 14

Let  $\mathcal{S}_{1:d} \in \text{Fl}(p, q_{1:d})$  and  $U_{1:d+1} := [U_1 \ U_2 \ \cdots \ U_d \ U_{d+1}] \in \mathcal{O}(p)$  be an orthogonal representative of  $\mathcal{S}_{1:d}$  (cf. Section 2). One has

$$\|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2 = \sqrt{(x_i - \Pi_{\mathcal{S}_{1:d}} x_i)^\top (x_i - \Pi_{\mathcal{S}_{1:d}} x_i)}, \quad (36)$$

$$= \sqrt{x_i^\top (I_p - \Pi_{\mathcal{S}_{1:d}})^2 x_i}, \quad (37)$$

$$= \sqrt{x_i^\top \left( I_p - \frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k} \right)^2 x_i}, \quad (38)$$

$$= \sqrt{\frac{1}{d^2} x_i^\top \left( \sum_{k=1}^d (I_p - \Pi_{\mathcal{S}_k}) \right)^2 x_i}. \quad (39)$$

By definition, one has  $\Pi_{\mathcal{S}_k} = \sum_{l=1}^k U_l U_l^\top$  and, in particular,  $I_p = \sum_{l=1}^{d+1} U_l U_l^\top$ . Therefore, by re-injection and double summation inversion, one gets:

$$\begin{aligned} \sum_{k=1}^d (I_p - \Pi_{\mathcal{S}_k}) &= \sum_{k=1}^d \left( \sum_{l=1}^{d+1} U_l U_l^\top - \sum_{l=1}^k U_l U_l^\top \right) = \sum_{k=1}^d \sum_{l=k+1}^{d+1} U_l U_l^\top = \sum_{l=2}^{d+1} \sum_{k=1}^{l-1} U_l U_l^\top = \\ &= \sum_{l=2}^{d+1} (l-1) U_l U_l^\top = \sum_{l=1}^{d+1} (l-1) U_l U_l^\top. \end{aligned} \quad (40)$$

Re-injecting the previous quantity in the computation of  $\|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2$ , one gets

$$\|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2 = \sqrt{\frac{1}{d^2} x_i^\top \left( \sum_{k=1}^{d+1} (k-1) U_k U_k^\top \right)^2 x_i}, \quad (41)$$

$$= \sqrt{\frac{1}{d^2} x_i^\top \left( \sum_{k=1}^{d+1} (k-1)^2 U_k U_k^\top \right) x_i}, \quad (42)$$

$$= \sqrt{\sum_{k=1}^{d+1} \left( \frac{k-1}{d} \right)^2 x_i^\top U_k U_k^\top x_i}, \quad (43)$$

$$\|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2 = \sqrt{\sum_{k=1}^{d+1} \left( \frac{k-1}{d} \right)^2 \|U_k^\top x_i\|_2^2}. \quad (44)$$

Hence, the flag-tricked robust subspace recovery problem (13) is equivalent to the following problem on the orthogonal group:

$$\arg \min_{U_{1:d+1} \in \mathcal{O}(p)} \sum_{i=1}^n \sqrt{\sum_{k=1}^{d+1} \left( \frac{k-1}{d} \right)^2 \|U_k^\top x_i\|_2^2}. \quad (45)$$

This problem can be further simplified into an optimization problem on Stiefel manifolds, by noting that

$$\|x_i - \Pi_{\mathcal{S}_{1:d}} x_i\|_2 = \sqrt{\sum_{k=1}^d \left( \left( \frac{k-1}{d} \right)^2 \|U_k^\top x_i\|_2^2 \right) + \left( \frac{d+1-1}{d} \right)^2 \|U_{d+1}^\top x_i\|_2^2}, \quad (46)$$

$$= \sqrt{\sum_{k=1}^d \left( \left( \frac{k-1}{d} \right)^2 \|U_k^\top x_i\|_2^2 \right) + x_i^\top U_{d+1} U_{d+1}^\top x_i}, \quad (47)$$

$$= \sqrt{\sum_{k=1}^d \left( \left( \frac{k-1}{d} \right)^2 \|U_k^\top x_i\|_2^2 \right) + x_i^\top (I_p - U_{1:d} U_{1:d}^\top) x_i}, \quad (48)$$

$$= \sqrt{\sum_{k=1}^d \left( \left( \frac{k-1}{d} \right)^2 \|U_k^\top x_i\|_2^2 \right) + \|x_i\|_2^2 - \|U_{1:d}^\top x_i\|_2^2}, \quad (49)$$

which concludes the proof.

## Appendix B. Trace Ratio Problems: Extensions and Proofs

In this section, we successively develop a Newton-Lanczos algorithm for the multilevel trace ratio problem (18), a kernelization of the same problem to handle nonlinear data, and we provide the proof of Proposition 16.

### B.1 A Newton Method for Multilevel Trace Ratio Problems

We wish to solve the following optimization problem (cf. (18)):

$$\arg \max_{\mathcal{S}_{1:d} \in \text{Fl}(p, q_{1:d})} \frac{\text{tr}(\Pi_{\mathcal{S}_{1:d}} A)}{\text{tr}(\Pi_{\mathcal{S}_{1:d}} B)}. \quad (50)$$

In this subsection, we propose a Newton method—drawn from Ngo et al. (2012), which is itself based on Wang et al. (2007) and Jia et al. (2009)—to solve this problem. The following proposition states that the multilevel trace ratio problem (50) can be reformulated as the search for the root (or zero) of a certain function, which can be then classically solved via Newton’s method.

**Proposition 20 (Multilevel Trace Ratio as a Newton Method)** *For  $M$  a symmetric matrix, let  $(\ell_j(M), v_j(M))$  denote its  $j$ -th largest (eigenvalue, eigenvector)-pair and  $\mathcal{S}_k(M) := \text{Span}(v_1(M), \dots, v_{q_k}(M))$ . Let  $f(\rho) := \sum_{k=1}^d \sum_{j=1}^{q_k} \ell_j(A - \rho B)$ . Then  $f$  admits a unique root  $\rho^*$  and one has*

$$\arg \max_{\mathcal{S}_{1:d} \in \text{Fl}(p, q_{1:d})} \frac{\text{tr}(\Pi_{\mathcal{S}_{1:d}} A)}{\text{tr}(\Pi_{\mathcal{S}_{1:d}} B)} = (\mathcal{S}_1(A - \rho^* B), \dots, \mathcal{S}_d(A - \rho^* B)) \quad (51)$$

**Proof** Let  $\phi: \text{Fl}(p, q_{1:d}) \ni \mathcal{S}_{1:d} \mapsto \text{tr} \left( \sum_{k=1}^d \Pi_{\mathcal{S}_k} A \right) / \text{tr} \left( \sum_{k=1}^d \Pi_{\mathcal{S}_k} B \right)$  be the objective function we want to maximize. We can see that  $\phi$  is well defined on  $\text{Fl}(p, q_{1:d})$  if and only if  $\text{rank}(B) > p - q_d$ , which we assume in the following.

$\phi$  is continuous on  $\text{Fl}(p, q_{1:d})$  which is a smooth compact manifold, therefore it admits a maximum  $\rho^*$ . As a consequence, for any  $\mathcal{S}_{1:d} \in \text{Fl}(p, q_{1:d})$ , one has  $\phi(\mathcal{S}_{1:d}) \leq \rho^*$ , which is equivalent to  $\psi(\mathcal{S}_{1:d}) := \sum_{k=1}^d \text{tr}(\Pi_{\mathcal{S}_k}(A - \rho^* B)) \leq 0$ . Similarly to Theorem 8, one can show that the maximum of  $\psi$ , which is 0, is attained in  $\mathcal{S}_{1:d}^* := (\mathcal{S}_1(A - \rho^* B), \dots, \mathcal{S}_d(A - \rho^* B))$ . Therefore, one has  $\psi(\mathcal{S}_{1:d}^*) = \sum_{k=1}^d \text{tr}(\Pi_{\mathcal{S}_k(A - \rho^* B)}(A - \rho^* B)) = f(\rho^*) = 0$ .

Let us now show that  $\rho^*$  is the unique root of  $f$ . One can show that the derivative of  $\sum_{j=1}^{q_k} \ell_j(A - \rho B)$  with respect to  $\rho$  is  $-\text{tr}(\Pi_{\mathcal{S}_k(A - \rho B)} B)$  (Ngo et al., 2012). Therefore, the derivative of  $f$  is  $f'(\rho) = -\sum_{k=1}^d \text{tr}(\Pi_{\mathcal{S}_k(A - \rho B)} B)$ , which is everywhere (strictly) negative since  $B$  is positive semidefinite and  $\text{rank}(B) > p - q_d$ . Therefore,  $f$  is a (strictly) decreasing function. Hence,  $\rho^*$  is the unique root of  $f$ .  $\blacksquare$

Newton's method to find the root  $\rho^*$  of  $f$  is described by the following iteration:

$$\rho^{t+1} = \rho^t - \frac{f(\rho^t)}{f'(\rho^t)} = \frac{\sum_{k=1}^d \text{tr}(\Pi_{\mathcal{S}_k(A - \rho^t B)} A)}{\sum_{k=1}^d \text{tr}(\Pi_{\mathcal{S}_k(A - \rho^t B)} B)}. \quad (52)$$

The optimal flag  $\mathcal{S}_{1:d}^*$  is given by the eigenvalue decomposition of  $A - \rho^\infty B$ , according to Proposition 20. The only remaining question is the one of initialization. The following proposition provides a possible answer.

**Proposition 21 (Bounds on Newton's Root)** *Let  $\ell_j(A, B)$  be the  $j$ -th largest eigenvalue of the matrix pencil  $(A, B)$ . If  $B \succ 0$ , then  $\rho^* \in [\ell_{q_d}(A, B), \ell_1(A, B)]$ .*

**Proof** Let  $Z \in \text{GL}_p$  such that  $Z^\top B Z = I_p$  and  $Z^\top A Z = \text{diag}(\ell_1(A, B), \dots, \ell_p(A, B))$ . Then, using Sylvester's law of inertia, one gets that the number of negative eigenvalues of  $Z^\top(A - \rho B)Z$  and of  $A - \rho I_p$  are the same. Therefore, for  $\rho = \ell_1(A, B)$ , all the eigenvalues of  $A - \rho B$  are nonpositive, so  $f(\rho) = \sum_{k=1}^d \sum_{j=1}^{q_k} \ell_j(A - \rho B) \leq 0$ . Similarly, for  $\rho = \ell_{q_d}(A, B)$ , the  $q_d$  largest eigenvalues of  $A - \rho B$  are nonnegative, so in particular for all  $k \in \{1, \dots, d\}$ , the  $q_k$  largest eigenvalues of  $A - \rho B$  are nonnegative, so in the end  $f(\rho) \geq 0$ .  $\blacksquare$

The bounds given by Proposition 21 can be used to initialize Newton's method. Alternatively, one can initialize the algorithm with a random flag (see Remark 4), similarly to Ngo et al. (2012, Algorithm 4.1), in order to circumvent the potentially costly computation of the generalized eigenvalues. The final proposed scheme is given in Algorithm 4, called *flag iterative trace ratio* in reference to the iterative trace ratio of Wang et al. (2007).

---

#### Algorithm 4 Flag Iterative Trace Ratio

---

**Require:**  $A, B \in \text{Sym}_+(p)$ ;  $q_{1:d} := (q_1, \dots, q_d)$  a flag signature

$\rho \in [\ell_{q_d}(A, B), \ell_1(A, B)]$

$\triangleright$  generalized eigenvalues

**for**  $t \in \{1, 2, \dots\}$  **do**

$U_{1:d} \leftarrow \text{EVD}(A - \rho B, q_d)$

$\triangleright$  top  $q_d$  eigenvectors

$\Pi_{\mathcal{S}_{1:d}} \leftarrow \frac{1}{d} \sum_{k=1}^d (d - k + 1) U_k U_k^\top$

$\triangleright$  multilevel projector (flag trick)

$\rho \leftarrow \text{tr}(\Pi_{\mathcal{S}_{1:d}} A) / \text{tr}(\Pi_{\mathcal{S}_{1:d}} B)$

$\triangleright$  trace ratio

**end for**

**Ensure:**  $U_{1:d}^* \in \text{Fl}(p, q_{1:d})$  an optimal flag (Stiefel representation)

---

## B.2 Kernel Trick for Graph Embedding

The seminal paper of Yan et al. (2007) shows that many dimensionality reduction methods can be seen as a trace ratio problem known as *graph embedding*. Let  $X := [x_1 \ \dots \ x_n] \in \mathbb{R}^{p \times n}$  be a  $p$ -dimensional data set. We look for a  $q$ -dimensional linear embedding of  $X$  that satisfies pairwise attraction and repulsion relationships. For instance, if we know the classes like in *linear discriminant analysis* (Fisher, 1936), we might want same-class points to be close and different-class points to be far in the lower-dimensional subspace. Writing, respectively,  $S_{ii'} \in \mathbb{R}$  and  $S_{ii'}^p \in \mathbb{R}$  the desired similarity and penalty between points  $i$  and  $i'$ , this yields to the following optimization problem (Yan et al., 2007; Wang et al., 2007):

$$\arg \min_{U \in \text{St}(p,q)} \frac{\sum_{i \neq i'} \|U^\top x_i - U^\top x_{i'}\|_2^2 S_{ii'}}{\sum_{i \neq i'} \|U^\top x_i - U^\top x_{i'}\|_2^2 S_{ii'}^p}. \quad (53)$$

Writing  $L \in \text{Sym}(n)$  and  $L^p \in \text{Sym}(n)$  the graph Laplacians associated with the adjacency matrices  $S \in \text{Sym}(n)$  and  $S^p \in \text{Sym}(n)$ —i.e.,  $L = D - S$ , with  $D$  a diagonal matrix such that  $D_{ii} = \sum_{i' \neq i} S_{ii'} \forall i$ —this can be rewritten as the following trace ratio problem:

$$\arg \max_{U \in \text{St}(p,q)} \frac{\text{tr}(U^\top X L^p X^\top U)}{\text{tr}(U^\top X L X^\top U)}. \quad (54)$$

Finding a linear embedding satisfying such attraction and repulsion properties might be complicated in practice, as real data distributions are often nonlinear. A classical trick—known as the *kernel trick* (Hofmann et al., 2008)—consists in nonlinearly mapping the data to an infinite-dimensional Hilbert space  $(\mathcal{H}, \langle \cdot, \cdot \rangle_{\mathcal{H}})$ —known as the reproducing kernel Hilbert space (RKHS)—via a so-called *kernel*  $\mathcal{K}: \mathbb{R}^p \times \mathbb{R}^p \rightarrow \mathbb{R}$ , and then performing the linear graph embedding in the RKHS. This kernel must satisfy symmetry and positive-definiteness properties—a classical example is the radial basis function (RBF) kernel  $\mathcal{K}(x, y) = e^{-\|x-y\|_2^2/(2\sigma^2)}$ . The nonlinear map  $\phi: \mathbb{R}^p \rightarrow \mathcal{H}$  is then chosen to be  $\phi(x) = \mathcal{K}(x, \cdot)$ , and the scalar product is  $\langle \phi(x), \phi(y) \rangle_{\mathcal{H}} = \mathcal{K}(x, y)$ .

Let us now write  $K := \mathcal{K}(x_i, x_j)_{i,j \in \{1, \dots, n\}} \in \text{Sym}(n)$ . Let  $f = (f_1, \dots, f_q) \in \mathcal{H}^q$  be an orthonormal family, with  $f_j = \sum_{i=1}^n v_{ij} \mathcal{K}(x_i, \cdot)$ . The weights  $V \in \mathbb{R}^{n \times q}$  must then satisfy the constraint  $V^\top K V = I_q$ . Moreover, the coefficients of the orthogonal projection of  $\phi(x_i)$  onto  $\text{Span}(f)$  are  $\langle f_j, \phi(x_i) \rangle_{\mathcal{H}} = \sum_{i'=1}^n K_{ii'} v_{i'j}$ . Consequently, linear graph embedding in the RKHS boils down to

$$\arg \min_{V^\top K V = I_q} \frac{\sum_{i \neq i'} \|V^\top K_i - V^\top K_{i'}\|_2^2 S_{ii'}}{\sum_{i \neq i'} \|V^\top K_i - V^\top K_{i'}\|_2^2 S_{ii'}^p}, \quad (55)$$

which is equivalent to

$$\arg \max_{V^\top K V = I_q} \frac{\text{tr}(V^\top K L^p K V)}{\text{tr}(V^\top K L K V)}. \quad (56)$$

Let  $K := Q \Lambda Q^\top$  be an eigendecomposition of  $K$ , with  $Q \in \mathcal{O}(n)$  and  $\Lambda \in \text{diag}(\mathbb{R}^n)$ . Let us make the change of variable  $U := \Lambda^{\frac{1}{2}} Q^\top V$ , and write  $A = \Lambda^{1/2} Q^\top L^p Q \Lambda^{1/2}$  and

$B = \Lambda^{1/2} Q^\top L Q \Lambda^{1/2}$ . Then the trace ratio problem in the RKHS can be rewritten as

$$\arg \max_{U \in \text{St}(n, q)} \frac{\text{tr}(U^\top A U)}{\text{tr}(U^\top B U)}. \quad (57)$$

One can now simply apply the flag trick to it, and get the following multilevel kernel graph embedding problem:

$$\arg \max_{\mathcal{S}_{1:d} \in \text{Fl}(n, q_{1:d})} \frac{\sum_{k=1}^d \text{tr}(\Pi_{\mathcal{S}_k} A)}{\sum_{k=1}^d \text{tr}(\Pi_{\mathcal{S}_k} B)}. \quad (58)$$

Let  $U^* := [U_1^* \ \cdots \ U_d^*] \in \text{St}(n, q)$  be a Stiefel representative of the optimal flag  $\mathcal{S}_{1:d}^* \in \text{Fl}(n, q_{1:d})$ . Then, one has for any  $x \in \mathbb{R}^p$  the kernel graph embedding map  $\Pi^*(x)_k = \sum_{i=1}^n U_{ik}^* K(x_i, x)$ . In particular,  $\Pi^*(X) = U^{*\top} K \in \mathbb{R}^{q \times n}$ .

### B.3 Proof of Proposition 16

Let  $\mathcal{S}_{1:d} \in \text{Fl}(p, q_{1:d})$  and  $U_{1:d} := [U_1 \ \cdots \ U_d] \in \text{St}(p, q)$  be a Stiefel representative of  $\mathcal{S}_{1:d}$  (cf. Section 2). One has

$$\frac{\text{tr}(\Pi_{\mathcal{S}_{1:d}} A)}{\text{tr}(\Pi_{\mathcal{S}_{1:d}} B)} = \frac{\text{tr}\left(\frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k} A\right)}{\text{tr}\left(\frac{1}{d} \sum_{l=1}^d \Pi_{\mathcal{S}_l} B\right)}, \quad (59)$$

$$= \frac{\text{tr}\left(\sum_{k=1}^d (d - (k - 1)) U_k U_k^\top A\right)}{\text{tr}\left(\sum_{l=1}^d (d - (l - 1)) U_l U_l^\top B\right)}, \quad (60)$$

$$\frac{\text{tr}(\Pi_{\mathcal{S}_{1:d}} A)}{\text{tr}(\Pi_{\mathcal{S}_{1:d}} B)} = \frac{\sum_{k=1}^d (d - (k - 1)) \text{tr}(U_k U_k^\top A)}{\sum_{l=1}^d (d - (l - 1)) \text{tr}(U_l U_l^\top B)}, \quad (61)$$

which concludes the proof.

### Appendix C. Spectral Clustering: Proof of Proposition 17

Let  $\mathcal{S}_{1:d} \in \text{Fl}(n, q_{1:d})$  and  $U_{1:d} := [U_1 \ \cdots \ U_d] \in \text{St}(n, q)$  be a Stiefel representative of  $\mathcal{S}_{1:d}$  (cf. Section 2). One has

$$\langle \Pi_{\mathcal{S}_{1:d}}, L \rangle_F + \beta \|\Pi_{\mathcal{S}_{1:d}}\|_1 = \left\langle \frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k}, L \right\rangle_F + \beta \left\| \frac{1}{d} \sum_{k=1}^d \Pi_{\mathcal{S}_k} \right\|_1, \quad (62)$$

$$= \frac{1}{d} \left( \left\langle \sum_{k=1}^d (d - (k - 1)) U_k U_k^\top, L \right\rangle_F + \beta \left\| \sum_{k=1}^d (d - (k - 1)) U_k U_k^\top \right\|_1 \right), \quad (63)$$

$$= \frac{1}{d} \left( \sum_{k=1}^d (d - (k - 1)) \langle U_k U_k^\top, L \rangle_F + \beta \left\| \sum_{k=1}^d (d - (k - 1)) U_k U_k^\top \right\|_1 \right), \quad (64)$$

$$= \frac{1}{d} \left( \sum_{k=1}^d (d - (k - 1)) \text{tr}(U_k^\top L U_k) + \beta \left\| \sum_{k=1}^d (d - (k - 1)) U_k U_k^\top \right\|_1 \right). \quad (65)$$

## Appendix D. Empirical Running Times

In this section, we describe the empirical running times of the flag trick (and the subsequent optimization on flag manifolds via Algorithm 1) for robust subspace recovery, trace ratio and sparse spectral clustering problems. We consider the three synthetic data sets described in Section 4, with varying  $n$  and  $p$ , and report the average running times over 10 independent experiments in Figure 9. The code is run on a CPU with 16 GB of RAM.

We see that the running times are in the order of a second for all methods. They increase approximately linearly with the dimension  $p$  (as expected from the complexity analysis of the Riemannian gradient and the polar retraction in Remark 6). They do not increase much with the number of samples  $n$ , except for the sparse spectral clustering problem which relies on the  $n \times n$  adjacency matrix.

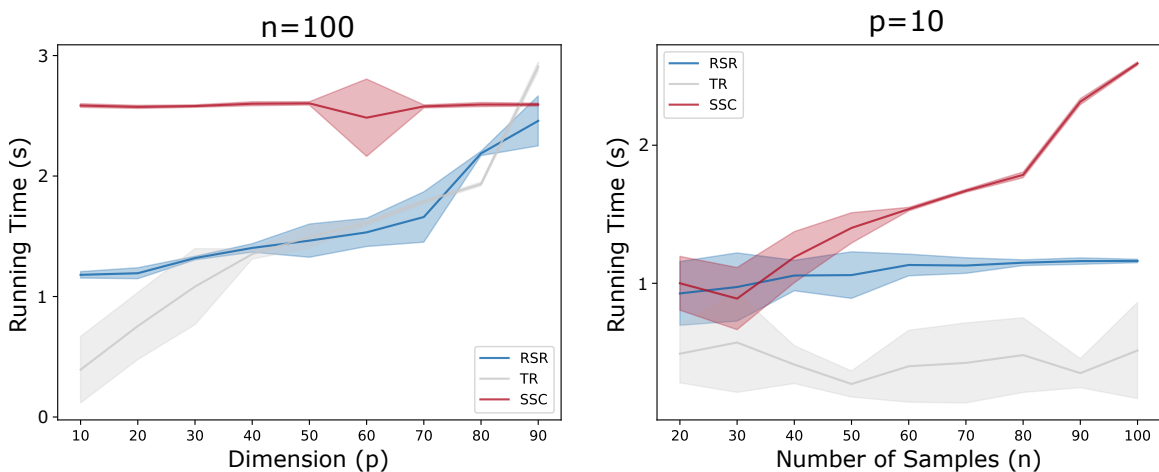


Figure 9: Empirical running times of the flag trick and the subsequent steepest descent on flag manifolds (Algorithm 1) for three machine learning problems: robust subspace recovery (RSR), trace ratio (TR) and sparse spectral clustering (SSC). On the left plot, the number of samples is fixed to  $n = 100$  and the dimension varies from  $p = 10$  to  $p = 90$ . On the right plot, the dimension is fixed to  $p = 10$  and the number of samples varies from  $n = 20$  to  $n = 100$ . We set  $q_{1:d} := (1, 2, 5)$ .

## References

- Pierre-Antoine Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2008.
- Ethem Alpaydin and Cenk Kaynak. Optical Recognition of Handwritten Digits [Dataset], 1998. URL <https://doi.org/10.24432/C50P49>. UCI Machine Learning Repository.
- Gregory Ammar and Clyde Martin. The geometry of matrix eigenvalue methods. *Acta Applicandae Mathematica*, 5(3):239–278, 1986.
- Vladimir I. Arnold. Modes and quasimodes. *Functional Analysis and Its Applications*, 6(2):94–101, 1972.
- Seth D. Axen, Mateusz Baran, Ronny Bergmann, and Krzysztof Rzecki. Manifolds.jl: An Extensible Julia Framework for Data Analysis on Manifolds. *ACM Transactions on Mathematical Software*, 49(4):33:1–33:23, 2023.
- Francis Bach, Rodolphe Jenatton, Julien Mairal, and Guillaume Obozinski. Optimization with Sparsity-Inducing Penalties. *Foundations and Trends® in Machine Learning*, 4(1):1–106, 2012.
- Mahsa Baktashmotlagh, Mehrtash T. Harandi, Brian C. Lovell, and Mathieu Salzmann. Unsupervised Domain Adaptation by Domain Invariant Projection. In *International Conference on Computer Vision (ICCV)*, pages 769–776, 2013.
- Laura Balzano, Robert Nowak, and Benjamin Recht. Online identification and tracking of subspaces from highly incomplete information. In *Allerton Conference on Communication, Control, and Computing*, pages 704–711, 2010.
- Matthew Barker and William Rayens. Partial least squares for discrimination. *Journal of Chemometrics*, 17(3):166–173, 2003.
- Peter N. Belhumeur, João P. Hespanha, and David J. Kriegman. Eigenfaces vs. Fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.
- Mikhail Belkin and Partha Niyogi. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation*, 15(6):1373–1396, 2003.
- Richard Bellman. *Dynamic programming*. Princeton University Press, 1957.
- Thomas Bendokat, Ralf Zimmermann, and Pierre-Antoine Absil. A Grassmann manifold handbook: basic geometry and computational aspects. *Advances in Computational Mathematics*, 50(1), 2024.
- Nicolas Boumal. *An Introduction to Optimization on Smooth Manifolds*. Cambridge University Press, 2023.

- Nicolas Boumal and Pierre-Antoine Absil. RTRMC: A Riemannian trust-region method for low-rank matrix completion. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 24, 2011.
- Nicolas Boumal, Bamdev Mishra, Pierre-Antoine Absil, and Rodolphe Sepulchre. Manopt, a matlab toolbox for optimization on manifolds. *Journal of Machine Learning Research*, 15(1):1455–1459, 2014.
- Charles Bouveyron, Stéphane Girard, and Cordelia Schmid. High-dimensional data clustering. *Computational Statistics & Data Analysis*, 52(1):502–519, 2007.
- Charles Bouveyron, Gilles Celeux, and Stéphane Girard. Intrinsic dimension estimation by maximum likelihood in isotropic probabilistic PCA. *Pattern Recognition Letters*, 32(14):1706–1713, 2011.
- Charles Bouveyron, Gilles Celeux, Thomas B. Murphy, and Adrian E. Raftery. *Model-Based Clustering and Classification for Data Science: With Applications in R*. Cambridge University Press, 1 edition, 2019.
- Roger W. Brockett. Dynamical systems that sort lists, diagonalize matrices, and solve linear programming problems. *Linear Algebra and its Applications*, 146:79–91, 1991.
- Emmanuel Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Communications of the ACM*, 55(6):111–119, 2012.
- Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *Journal of the ACM*, 58(3):11:1–11:37, 2011.
- Jean-François Cardoso. Multidimensional independent component analysis. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 4, pages 1941–1944, 1998.
- Wei-Chien Chang. On Using Principal Components Before Separating a Mixture of Two Multivariate Normal Distributions. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 32(3):267–275, 1983.
- Hwann-Tzong Chen, Huang-Wei Chang, and Tyng-Luh Liu. Local discriminant embedding and its variants. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 846–853, 2005.
- Yasuko Chikuse. *Statistics on Special Manifolds*, volume 174 of *LNS*. Springer, 2003.
- Ioana Ciuclea, Alice B. Tumpach, and Cornelia Vizman. Shape Spaces of Nonlinear Flags. In *Geometric Science of Information (GSI)*, pages 41–50, 2023.
- John P. Cunningham and Zoubin Ghahramani. Linear Dimensionality Reduction: Survey, Insights, and Generalizations. *Journal of Machine Learning Research*, 16(89):2859–2900, 2015.
- James Damon and James S. Marron. Backwards Principal Component Analysis and Principal Nested Relations. *Journal of Mathematical Imaging and Vision*, 50, 2014.

- Ingrid Daubechies, Ronald DeVore, Massimo Fornasier, and Sinan Güntürk. Iteratively reweighted least squares minimization for sparse recovery. *Communications on Pure and Applied Mathematics*, 63(1):1–38, 2010.
- Percy A. Deift, Tara Nanda, and Carlos Tomei. Ordinary Differential Equations and the Symmetric Eigenvalue Problem. *SIAM Journal on Numerical Analysis*, 20(1):1–22, 1983.
- Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.
- Chris Ding, Ding Zhou, Xiaofeng He, and Hongyuan Zha. R1-PCA: rotational invariant L1-norm principal component analysis for robust subspace factorization. In *International Conference on Machine Learning (ICML)*, pages 281–288, 2006.
- Bruce Draper, Michael Kirby, Justin Marks, Tim Marrinan, and Chris Peterson. A flag representation for finite collections of subspaces of mixed dimensions. *Linear Algebra and its Applications*, 451:15–32, 2014.
- Ian L. Dryden, Kwang-Rae Kim, Charles A. Laughton, and Huiling Le. Principal nested shape space analysis of molecular dynamics data. *The Annals of Applied Statistics*, 13(4):2213–2234, 2019.
- Johannes J. Duistermaat, Johan A. C. Kolk, and Veeravalli S. Varadarajan. Functions, flows and oscillatory integrals on flag manifolds and conjugacy classes in real semisimple Lie groups. *Compositio Mathematica*, 49(3):309–398, 1983.
- Alan Edelman, Tomás A. Arias, and Steven T. Smith. The Geometry of Algorithms with Orthogonality Constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353, 1998.
- Ehsan Elhamifar and René Vidal. Sparse Subspace Clustering: Algorithm, Theory, and Applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2765–2781, 2013.
- Benjamin Eltzner, Stephan Huckemann, and Kanti V. Mardia. Torus principal component analysis with applications to RNA structure. *The Annals of Applied Statistics*, 12(2):1332–1359, 2018.
- Xiran Fan, Chun-Hao Yang, and Baba C. Vemuri. Nested Hyperbolic Spaces for Dimensionality Reduction and Hyperbolic NN Design. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 356–365, 2022.
- Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R. Arabnia. A Brief Review of Domain Adaptation. In *Advances in Data Science and Information Engineering*, pages 877–894. Springer International Publishing, 2021.
- Simone Fiori. Extended Hamiltonian Learning on Riemannian Manifolds: Theoretical Aspects. *IEEE Transactions on Neural Networks*, 22(5):687–700, 2011.

- Simone Fiori. Extended Hamiltonian Learning on Riemannian Manifolds: Numerical Aspects. *IEEE Transactions on Neural Networks and Learning Systems*, 23(1):7–21, 2012.
- Ronald A. Fisher. The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, 7(2):179–188, 1936.
- Thomas Fletcher, Conglin Lu, Stephen M. Pizer, and Sarang Joshi. Principal Geodesic Analysis for the Study of Nonlinear Statistics of Shape. *IEEE Transactions on Medical Imaging*, 23(8):995–1005, 2004.
- Paul Geladi and Bruce R. Kowalski. Partial least-squares regression: a tutorial. *Analytica Chimica Acta*, 185:1–17, 1986.
- Stephen Giguere, Francisco Garcia, and Sridhar Mahadevan. A Manifold Approach to Learning Mutually Orthogonal Subspaces, 2017. URL <http://arxiv.org/abs/1703.02992>.
- Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2066–2073, 2012.
- Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *International Conference on Computer Vision (ICCV)*, pages 999–1006. IEEE, 2011.
- Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A Kernel Two-Sample Test. *Journal of Machine Learning Research*, 13(25):723–773, 2012.
- Yue-Fei Guo, Shi-Jin Li, Jing-Yu Yang, Ting-Ting Shu, and Li-De Wu. A generalized Foley–Sammon transform based on generalized fisher discriminant criterion and its application to face recognition. *Pattern Recognition Letters*, 24(1):147–158, 2003.
- Jihun Hamm and Daniel D. Lee. Grassmann discriminant analysis: a unifying view on subspace-based learning. In *International Conference on Machine Learning (ICML)*, pages 376–383, 2008.
- Mehrtash Harandi, Mathieu Salzmann, and Richard Hartley. Dimensionality Reduction on SPD Manifolds: The Emergence of Geometry-Aware Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(1):48–62, 2018.
- David R. Hardoon, Sandor Szedmak, and John Shawe-Taylor. Canonical Correlation Analysis: An Overview with Application to Learning Methods. *Neural Computation*, 16(12):2639–2664, 2004.
- Søren Hauberg, Aasa Feragen, Raffi Enficiaud, and Michael J. Black. Scalable Robust Principal Component Analysis Using Grassmann Averages. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2298–2311, 2016.

- Xiaofei He and Partha Niyogi. Locality Preserving Projections. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 16, 2003.
- Uwe Helmke. Isospectral flows on symmetric matrices and the Riccati equation. *Systems & Control Letters*, 16(3):159–165, 1991.
- Uwe Helmke and John B. Moore. *Optimization and Dynamical Systems*. Communications and Control Engineering. Springer, 1994.
- Thomas Hofmann, Bernhard Schölkopf, and Alexander J. Smola. Kernel methods in machine learning. *The Annals of Statistics*, 36(3):1171–1220, 2008.
- Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441, 1933.
- Harold Hotelling. Relations Between Two Sets of Variates. *Biometrika*, 28(3/4):321–377, 1936.
- Peter J. Huber. Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, 1964.
- Peter J. Huber. Projection Pursuit. *The Annals of Statistics*, 13(2):435–475, 1985.
- Stephan Huckemann and Benjamin Eltzner. Backward nested descriptors asymptotics with inference on stem cell differentiation. *The Annals of Statistics*, 46(5), 2018.
- Stephan Huckemann, Thomas Hotz, and Axel Munk. Intrinsic Shape Analysis: Geodesic Pca for Riemannian Manifolds Modulo Isometric Lie Group Actions. *Statistica Sinica*, 20(1), 2010.
- Aapo Hyvärinen and Patrik Hoyer. Emergence of Phase- and Shift-Invariant Features by Decomposition of Natural Images into Independent Feature Subspaces. *Neural Computation*, 12(7):1705–1720, 2000.
- Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural Networks*, 13(4):411–430, 2000.
- Alan T. James. Normal Multivariate Analysis and the Orthogonal Group. *The Annals of Mathematical Statistics*, 25(1):40–75, 1954.
- Noémie Jaquier and Leonel Rozo. High-Dimensional Bayesian Optimization via Nested Riemannian Manifolds. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 20939–20951, 2020.
- Yangqing Jia, Feiping Nie, and Changshui Zhang. Trace Ratio Problem Revisited. *IEEE Transactions on Neural Networks*, 20(4):729–735, 2009.
- Dian Jin and Jeremy Coulson. Online Subspace Learning on Flag Manifolds for System Identification, 2025. URL <http://arxiv.org/abs/2511.06416>.

- Ian T. Jolliffe. *Principal Component Analysis*. Springer Series in Statistics. Springer-Verlag, 2002.
- Sungkyu Jung, Ian L. Dryden, and James S. Marron. Analysis of principal nested spheres. *Biometrika*, 99(3):551–568, 2012.
- Raghunandan H. Keshavan, Andrea Montanari, and Sewoong Oh. Matrix Completion From a Few Entries. *IEEE Transactions on Information Theory*, 56(6):2980–2998, 2010.
- Tae-Kyun Kim, Josef Kittler, and Roberto Cipolla. On-line Learning of Mutually Orthogonal Subspaces for Face Recognition by Image Sets. *IEEE Transactions on Image Processing*, 19(4):1067–1074, 2010.
- Teuvo Kohonen. Emergence of invariant-feature detectors in the adaptive-subspace self-organizing map. *Biological Cybernetics*, 75(4):281–291, 1996.
- Effrosyni Kokiopoulou and Yousef Saad. Enhanced graph-based dimensionality reduction with repulsion Laplaceans. *Pattern Recognition*, 42(11):2392–2402, 2009.
- Hugo Launay, François Willot, David Ryckelynck, and Jacques Besson. Mechanical assessment of defects in welded joints: morphological classification and data augmentation. *Journal of Mathematics in Industry*, 11(1), 2021.
- Gilad Lerman and Tyler Maunu. Fast, robust and non-convex subspace recovery. *Information and Inference: A Journal of the IMA*, 7(2):277–336, 2018a.
- Gilad Lerman and Tyler Maunu. An Overview of Robust Subspace Recovery. *Proceedings of the IEEE*, 106(8):1380–1410, 2018b.
- Gilad Lerman, Michael B. McCoy, Joel A. Tropp, and Teng Zhang. Robust Computation of Linear Models by Convex Relaxation. *Foundations of Computational Mathematics*, 15(2):363–410, 2015.
- Lek-Heng Lim and Ke Ye. Simple matrix models for the flag, Grassmann, and Stiefel manifolds, 2024. URL <http://arxiv.org/abs/2407.13482>.
- Lek-Heng Lim, Ken Sze-Wai Wong, and Ke Ye. Numerical Algorithms on the Affine Grassmannian. *SIAM Journal on Matrix Analysis and Applications*, 40(2):371–393, 2019.
- Canyi Lu, Shuicheng Yan, and Zhouchen Lin. Convex Sparse Spectral Clustering: Single-View to Multi-View. *IEEE Transactions on Image Processing*, 25(6):2833–2843, 2016.
- Xiaofeng Ma, Michael Kirby, and Chris Peterson. The Flag Manifold as a Tool for Analyzing and Comparing Sets of Data Sets. In *International Conference on Computer Vision (ICCV) Workshops*, pages 4168–4177, 2021.
- Xiaofeng Ma, Michael Kirby, and Chris Peterson. Self-organizing mappings on the flag manifold with applications to hyper-spectral image data analysis. *Neural Computing and Applications*, 34(1):39–49, 2022.

- Nathan Mankovich and Tolga Birdal. Chordal Averaging on Flag Manifolds and Its Applications. In *International Conference on Computer Vision (ICCV)*, pages 3881–3890, 2023.
- Nathan Mankovich, Emily J. King, Chris Peterson, and Michael Kirby. The Flag Median and FlagIRLS. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10329–10337, 2022.
- Nathan Mankovich, Gustau Camps-Valls, and Tolga Birdal. Fun with Flags: Robust Principal Directions via Flag Manifolds. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 330–340, 2024.
- Nathan Mankovich, Ignacio Santamaria, Gustau Camps-Valls, and Tolga Birdal. A Flag Decomposition for Hierarchical Datasets. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18738–18748, 2025.
- Tyler Maunu, Teng Zhang, and Gilad Lerman. A Well-Tempered Landscape for Non-convex Robust Subspace Recovery. *Journal of Machine Learning Research*, 20(37):1–59, 2019.
- Michael B. McCoy and Joel A. Tropp. Two proposals for robust PCA using semidefinite programming. *Electronic Journal of Statistics*, 5:1123–1160, 2011.
- Thomas Minka. Automatic Choice of Dimensionality for PCA. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 13, 2000.
- Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On Spectral Clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 14, 2001.
- Thanh T. Ngo, Mohammed Bellalij, and Yousef Saad. The Trace Ratio Optimization Problem. *SIAM Review*, 54(3):545–569, 2012.
- Du Nguyen. Closed-form Geodesics and Optimization for Riemannian Logarithms of Stiefel and Flag Manifolds. *Journal of Optimization Theory and Applications*, 194(1):142–166, 2022.
- Yasunori Nishimori, Shotaro Akaho, and Mark D. Plumbley. Riemannian Optimization Method on the Flag Manifold for Independent Subspace Analysis. In *Independent Component Analysis and Blind Signal Separation*, pages 295–302, 2006.
- Yasunori Nishimori, Shotaro Akaho, Samer Abdallah, and Mark D. Plumbley. Flag Manifolds for Subspace ICA Problems. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 4, pages IV–1417–IV–1420, 2007.
- Yasunori Nishimori, Shotaro Akaho, and Mark D. Plumbley. Natural Conjugate Gradient on Complex Flag Manifolds for Complex Independent Subspace Analysis. In *International Conference on Artificial Neural Networks (ICANN)*, pages 165–174. Springer, 2008.
- Reza Oftadeh, Jiayi Shen, Zhangyang Wang, and Dylan Shell. Eliminating the Invariance on the Loss Landscape of Linear Autoencoders. In *International Conference on Machine Learning (ICML)*, pages 7405–7413. PMLR, 2020.

- Karl Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011.
- Liangzu Peng, Christian Kümmeler, and René Vidal. On the Convergence of IRLS and Its Variants in Outlier-Robust Estimation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17808–17818, 2023.
- Xavier Pennec. Barycentric Subspace Analysis on Manifolds. *The Annals of Statistics*, 46(6A):2711–2746, 2018.
- Michael P. Perrone and Leon N. Cooper. When networks disagree: Ensemble methods for hybrid neural networks. In *How We Learn; How We Remember: Toward an Understanding of Brain and Neural Systems*, volume 10 of *World Scientific Series in 20th Century Physics*, pages 342–358. World Scientific, 1995.
- Dimbihery Rabenoro and Xavier Pennec. A geometric framework for asymptotic inference of principal subspaces in PCA, 2024. URL <http://arxiv.org/abs/2209.02025>.
- Paulius Rauba and Mihaela van der Schaar. Deep Hierarchical Learning with Nested Subspace Networks for Large Language Models. In *International Conference on Learning Representations (ICLR)*, 2026.
- Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting Visual Category Models to New Domains. In *European Conference on Computer Vision (ECCV)*, pages 213–226. Springer, 2010.
- Ignacio Santamaría, Louis L. Scharf, Christopher Peterson, Michael Kirby, and Joseph M. Francos. An order fitting rule for optimal subspace averaging. In *Statistical Signal Processing Workshop (SSP)*, pages 1–4, 2016.
- Stefan Sommer, Francois Lauze, and Mads Nielsen. Optimization over Geodesics for Exact Principal Geodesic Analysis. *Advances in Computational Mathematics*, 40, 2014.
- Jiaji Su and Zhigang Yao. Principal Decomposition with Nested Submanifolds, 2025. URL <http://arxiv.org/abs/2502.10010>.
- Tom Szwagier. *Rethinking statistical methods with flags*. Theses, Université Côte d’Azur, 2025. URL <https://inria.hal.science/tel-05423812>. 2025COAZ4027.
- Tom Szwagier and Xavier Pennec. Rethinking the Riemannian Logarithm on Flag Manifolds as an Orthogonal Alignment Problem. In *Geometric Science of Information (GSI)*, LNCS, pages 375–383. Springer, 2023.

- Tom Szwagier and Xavier Pennec. The curse of isotropy: from principal components to principal subspaces. *Statistical Science*, 2026.
- Tom Szwagier, Guillaume Olikier, and Xavier Pennec. Eigengap Sparsity for Covariance Parsimony. In *Geometric Science of Information (GSI)*, pages 50–59. Springer Nature Switzerland, 2025.
- Tom Szwagier, Pierre-Alexandre Mattei, Charles Bouveyron, and Xavier Pennec. Parsimonious Gaussian mixture models with piecewise-constant eigenvalue profiles. *Statistics and Computing*, 36(95), 2026.
- Michael E. Tipping and Christopher M. Bishop. Probabilistic Principal Component Analysis. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 61(3): 611–622, 1999.
- Warren S. Torgerson. Multidimensional scaling: I. Theory and method. *Psychometrika*, 17(4):401–419, 1952.
- James Townsend, Niklas Koep, and Sebastian Weichwald. Pymanopt: A Python Toolbox for Optimization on Manifolds using Automatic Differentiation. *Journal of Machine Learning Research*, 17(137):1–5, 2016.
- Manolis C. Tsakiris and René Vidal. Dual principal component pursuit. *The Journal of Machine Learning Research*, 19(1):684–732, 2018.
- Laurent Vidal, Tommaso Nottoli, Filippo Lipparini, and Eric Cancès. Geometric Optimization of Restricted-Open and Complete Active Space Self-Consistent Field Wave Functions. *The Journal of Physical Chemistry A*, 2024.
- René Vidal, Yi Ma, and Shankar Sastry. *Generalized Principal Component Analysis*, volume 40 of *Interdisciplinary Applied Mathematics*. Springer, 2016.
- Huan Wang, Shuicheng Yan, Dong Xu, Xiaoou Tang, and Thomas Huang. Trace Ratio vs. Ratio Trace for Dimensionality Reduction. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2007.
- Li Wang, Lei-Hong Zhang, and Ren-Cang Li. Trace ratio optimization with an application to multi-view learning. *Mathematical Programming*, 201(1):97–131, 2023.
- Qiong Wang, Junbin Gao, and Hong Li. Grassmannian Manifold Optimization Assisted Sparse Spectral Clustering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3145–3153, 2017.
- Satosi Watanabe and Nikhil Pakvasa. Subspace method of pattern recognition. In *International Joint Conference on Pattern Recognition*, pages 25–32, 1973.
- David S. Watkins. Isospectral Flows. *SIAM Review*, 26(3):379–391, 1984.
- David S. Watkins and Ludwig Elsner. Chasing Algorithms for the Eigenvalue Problem. *SIAM Journal on Matrix Analysis and Applications*, 12(2):374–384, 1991.

- Endre Weiszfeld. Sur le point pour lequel la somme des distances de  $n$  points donnés est minimum. *Tohoku Mathematical Journal, First Series*, 43:355–386, 1937.
- Laurenz Wiskott and Terrence J. Sejnowski. Slow Feature Analysis: Unsupervised Learning of Invariances. *Neural Computation*, 14(4):715–770, 2002.
- William H. Wolberg, Olvi L. Mangasarian, and Nick Street. Breast Cancer Wisconsin (Diagnostic) [Dataset], 1993. URL <https://doi.org/10.24432/C5DW2B>. UCI Machine Learning Repository.
- Huan Xu, Constantine Caramanis, and Sujay Sanghavi. Robust PCA via Outlier Pursuit. *IEEE Transactions on Information Theory*, 58(5):3047–3064, 2012.
- Shuicheng Yan, Dong Xu, Benyu Zhang, Hong-Jiang Zhang, Qiang Yang, and Stephen Lin. Graph Embedding and Extensions: A General Framework for Dimensionality Reduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1):40–51, 2007.
- Chun-Hao Yang and Baba C. Vemuri. Nested Grassmanns for Dimensionality Reduction with Applications to Shape Analysis. In *Information Processing in Medical Imaging (IPMI)*, pages 136–149. Springer International Publishing, 2021.
- Ke Ye and Lek-Heng Lim. Schubert Varieties and Distances between Subspaces of Different Dimensions. *SIAM Journal on Matrix Analysis and Applications*, 37(3):1176–1197, 2016.
- Ke Ye, Ken Sze-Wai Wong, and Lek-Heng Lim. Optimization on flag manifolds. *Mathematical Programming*, 194(1):621–660, 2022.
- Teng Zhang and Gilad Lerman. A novel M-estimator for robust PCA. *The Journal of Machine Learning Research*, 15(1):749–808, 2014.
- Xiaoqing Zhu and Chungeng Shen. Practical gradient and conjugate gradient methods on flag manifolds. *Computational Optimization and Applications*, 88(2):491–524, 2024.