Subspace Quantization on the Grassmannian

Shannon Stiverson, Michael Kirby, and Chris Peterson

Colorado State University, Fort Collins CO 80523, USA
(stiverso,kirby,peterson)@math.colostate.edu

Abstract. We extend the K-means and LBG algorithms to the framework of the Grassmann manifold to perform subspace quantization. For K-means it is possible to move a subspace in the direction of another using Grassmannian geodesics. For LBG the centroid computation is now done using a flag mean algorithm for averaging points on the Grassmannian. The resulting unsupervised algorithms are applied to the MNIST digit data set and the AVIRIS Indian Pines hyperspectral data set.

Keywords: Grassmannian, LBG, K-means, flag mean.

1 Introduction

The Grassmann manifold provides a robust geometric framework for analyzing complex, high-dimensional data sets where observations are characterized by a variation in state. For example, variations in illumination confound pattern recognition systems given the sensitivity of the representation to the angle of illumination. The use of the Grassmannian greatly mitigates this problem [6, 4]. Similarly, satellite imaging systems collect hyperspectral data with high-resolution both spectrally and spatially. A given substance, e.g., a field of corn, will show significant variability in the spectral signature over even small image patches. In one study, it was shown that the classes *soybean with tilling* versus *soybean with no tilling* could be separated with perfect accuracy using the Grassmannian, whereas the best vector space methods could not [7]. Also, since the Grassmannian is itself a manifold, this framework lends itself naturally to analysis using topological and geometric methods, see, e.g., [9, 8, 2, 13]. Given the robust performance of the Grassmannian in these examples, it is desirable to explore the extension of core tools in data analysis in the geometric setting of the Grassmann manifold. Two building blocks for algorithms on the Grassmannian are a means to compare distances between points and a means to compute averages of points on the Grassmann manifold. These techniques are described in Sections 2 and 3. We note that the self-organizing mapping algorithm of Kohonen has been adapted to this geometric framework with success [16]. The goal of this work is to extend the K-means and LBG vector quantization algorithms to Grassmann manifolds in order to provide a robust unsupervised method for quantizing data subspaces. In this work, K-means refers to the online method by MacQueen [18] while LBG refers to the the batch version of the algorithm developed by Linde, Buzo, and Gray [17].

The outline of this paper is as follows: In Section 2 we provide background on the Grassmann manifold. In Section 3 we describe how to average points on the Grassmannian via the *flag mean*. Section 4 outlines the Grassmann K-means algorithm, and Section 5 presents Grassmannian LBG. In Sections 6.1 and 6.2 we present applications.

2 The Grassmannian

The real Grassmann manifold $Gr(p, n)$ is a manifold whose points parameterize the linear subspaces of dimension p in $\mathbb{R}^n$ [1]. One can construct $Gr(p, n)$ as a quotient manifold of the Stiefel manifold $St(p, n)$ [12]. This relationship allows for an intuitive representation of points on the Grassmannian that lends itself well to computations. Any point on $Gr(p, n)$ can be identified with a matrix $X \in St(p, n)$ whose column space spans the desired subspace $[X]$. Orthogonally invariant norms on the Grassmannian may be expressed in terms of principal angles θ_i between subspaces [12]. This is computationally appealing since principal angles can be determined from the singular values of the SVD of $X^T Y$ [5]. For example, the *chordal norm* is given by

$$d_c([X], [Y]) = \| \sin \theta \|_2. \tag{1}$$

Moreover, any set of points on the Grassmannnian, with distances measured in this way, can be isometrically embedded into Euclidean space using multidimensional scaling (MDS). In practice, the smallest angle pseudometric generally gives the best data separation, although the embedding into Euclidean space is no longer isometric [7].

3 Averaging Subspaces

The flag mean is an algorithm for computing averages of points on Grassmannians [19, 20, 11]. One can use such an algorithm to determine common attributes, within a set of points on the Grassmannian, expressed as a set of nested subspaces [19]. The flag mean algorithm, which we summarize below, is at the heart of the Grassmannian LBG procedure.

A flag is a nested sequence of subspaces. Given a finite collection of subspaces, the flag mean algorithm computes the best flag representation of the collection. Denote the flag by $\{[u_1], [u_2], ..., [u_r]\}$, where the u_i are orthogonal unit vectors with $r \leq n$. Let $\{[X_i]\}$ be a set of points in $Gr(p, n)$ and $\{X_i\}$ be their corresponding matrix representations. To construct the flag mean, iteratively solve the optimization problem

$$[u_j] = \arg \min_{[u] \in Gr(1,n)} \sum_{i=1}^{p} d_c([u], [X_i])^2 \tag{2}$$

$$\text{subject to } [u] \perp [u_l] \text{ for all } l < j$$

for $[u_1], ..., [u_r]$ [11]. Optimality is achieved when

$$\left(\sum_{i=1}^{m} X_i X_i^T \right) u = \lambda u$$

and the problem is reduced to an eigenvector computation [11].

4 Grassmann K-means Algorithm

The K-means algorithm operates on a stream of data, assigning each data point to its nearest center [18]. The chosen center is then updated in the direction of the new data point. Let x be the n^{th} data point assigned to a center c. In Euclidean space, the center c is then updated by

$$c_{new} = c + \frac{1}{n}(x - c) \tag{3}$$

To adapt this algorithm to the Grassmann manifold, we require a way to move one subspace a specified distance towards another. This is accomplished by parameterizing the geodesic between two subspaces $[X]$ and $[Y]$ [1, 15]. Given orthonormal matrix representations X and Y, respectively, the velocity matrix H that induces a geodesic between $[X]$ and $[Y]$ is given by

$$H = (I - XX^T)Y(X^TY)^{-1}. \tag{4}$$

The singular value decomposition of the velocity matrix $H = U\Sigma V^T$ is then used to parameterize a geodesic curve between X and Y by

$$\Phi(t) = XV\cos(\Theta t) + U\sin(\Theta t) \tag{5}$$

where $\Theta = \arctan(\Sigma)$. Note $\Phi(0) = X$ and $\Phi(1) = \tilde{Y}$, with $[\tilde{Y}] = [Y]$. Using this, we can update K-means by letting $t = 1/n$. The K-means algorithm on the Grassmannian is then:

1. Construct points on $Gr(k, n)$ using raw data.
2. Select k random initial centers from the data on $Gr(k, n)$.
3. For each data point $[X]$:
   (a) Find the center $[C_i]$ nearest to $[X]$.
   (b) Update $[C_i]$ according to Equation (5).
4. Calculate the average distortion error and check if it is smaller than the specified threshold. If not, repeat step 2.

This process can be applied several times to improve the clusters.

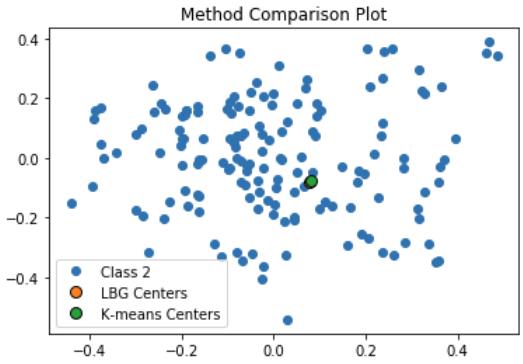

Fig. 1: MDS embedding of handwritten digit 2 and centers selected by each algorithm; the LBG center is beneath the K-means center.

5 The LBG Algorithm on the Grassmannian

The Linde-Buzo-Gray algorithm (LBG) performs vector quantization in Euclidean space by associating all points to their nearest center [17]. These centers serve as prototypes for the data in the sense that they minimize the distortion error locally. On the Grassmannian, the centroid of all points closest to a given center C_i is obtained using the flag mean $[u_1, \ldots, u_r]$. The averaging is done over elements of Voronoi sets, i.e., the collection of data points closest to a given center. The definition of a Voronoi set S_i is given by

$$S_i = \{x : d(x, c_i) \leq d(x, c_j), \quad i \neq j\}$$

The centroid is found using $c_i = M(S_i)$ where the function M represents a mapping from the members of the Voronoi set S_i to its "mean". In Euclidean space the mean is the usual centroid

$$c_i = \frac{1}{|S_i|} \sum_{x \in S_i} x \tag{6}$$

for $x \in \mathbb{R}^n$. For Grassmannians this mean is the output of Equation (2), i.e., the flag mean. On $Gr(p, n)$, the distortion of the clusters with centroids $[C_i]$ is given by

$$D(\{S_i\}_{i=1}^k) = \sum_{i=1}^{k} \sum_{[X] \in S_i} d([X], [C_i])$$

The Grassmannian LBG algorithm is as follows:

1. Initialize k random centers on $Gr(p, n)$.
2. Assign each subspace $[X_i]$ to its nearest center $[C_{i^*}]$.
3. Update the centers using Equation (2).

4. Calculate the average distortion associated with the new partition using chordal distances on the Grassmannian.
5. If stopping citerion not met, then go to step 2.

The paper by Gruber and Theis contains a similar algorithm for clustering on the Grassmannian [14] based on the projection Frobenius norm, though they do not approach centroid calculations from the framework of flag subspaces.

6 Numerical Experiments

One goal of clustering methods is for each center to contain, as nearly as possible, points from only one class. We can express the *purity* of a cluster by the fraction of points belonging to the majority class for that cluster. We use this measure to establish quantitative comparisons below.

6.1 MNIST Results

We use the MNIST handwritten data set to illustrate these algorithms [10]. This data set contains 28 x 28 images of handwritten digits vectorized into a data point in $\mathbb{R}^{784}$. To construct points on $Gr(p, 784)$ we select p data points from the same class and form a $p \times 784$ orthonormnal matrix using the QR-decomposition. Subspaces are assigned the same label as the points used to construct them. Figure 1 illustrates the performance of both algorithms on a set of 156 data points in Gr(5,784) generated using the handwritten digit 2. One centroid was selected at random to initialize each algortihm. One LBG iteration and one K-means epoch generated essentially identical means. The visualization of the results was achieved by using MDS with the chordal matric to isometrically embed the subspaces and centroids into $\mathbb{R}^2$.

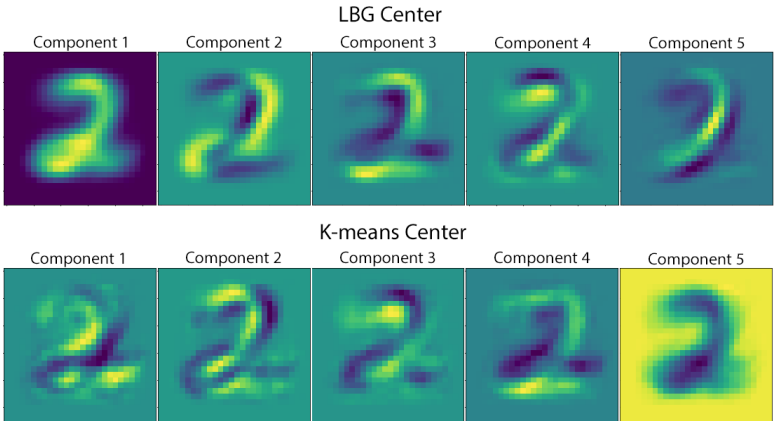

Fig. 2: Visualization of orthonormal components for each center.

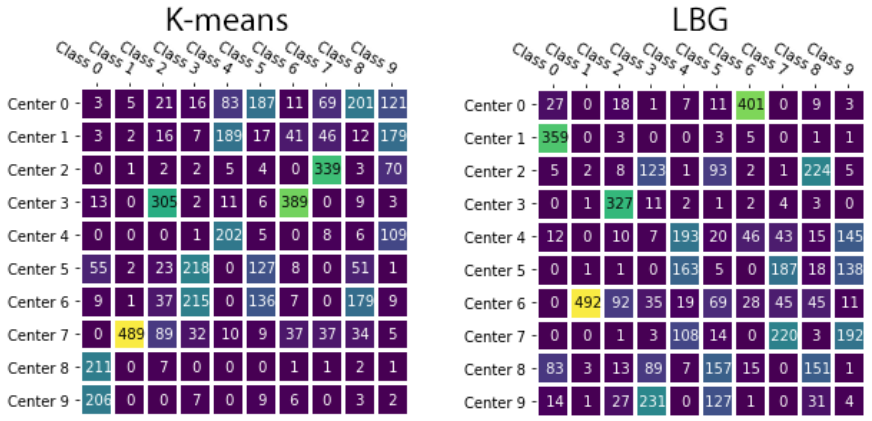

Fig. 3: Results from the Euclidean space algorithms applied to all ten MNIST digits.

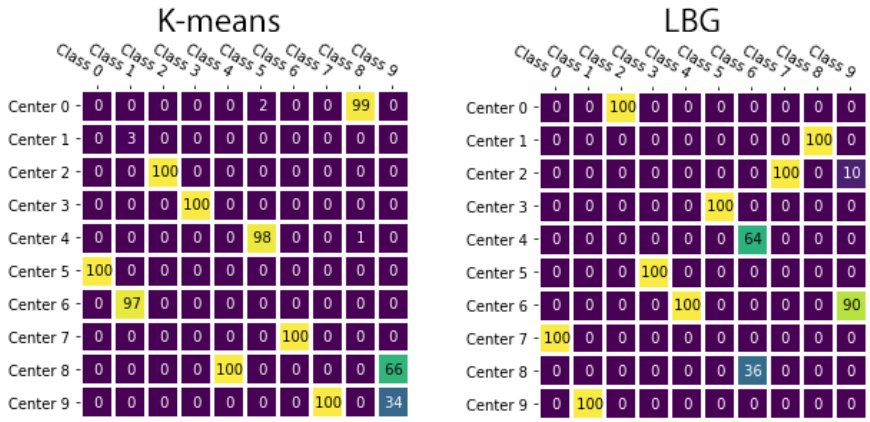

Fig. 4: Results the Grassmannian algorithms applied to all ten MNIST digits.

A closer look at the orthonormal components of each center reveals details about variations in the cluster. Figure 2 shows the five orthonormal components of each center reshaped to the original image size. In particular, because the flag mean yields an orthonormal basis ordered by energy [11], the first component of the LBG center contains the elements most commonly found among all points in the cluster, and represents first dimension of the "true" mean. Each consecutive component captures information about the most common variations from the mean, ordered from most common to least. K-means captures similar information about within-cluster variation but does not have any special ordering.

To further explore these methods, both algorithms were applied to all the MNIST digits. For each digit, 500 data points were randomly selected from the MNIST training set and used to construct subspaces. As a baseline comparison

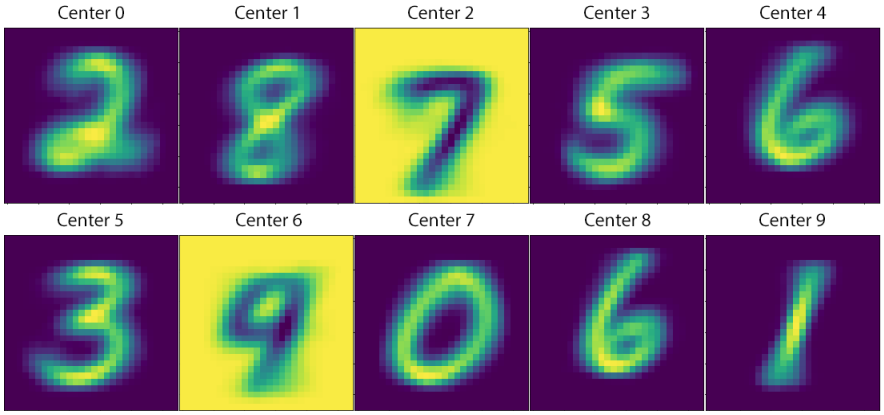

Fig. 5: Visualization of the first flag vector for each of the 10 LBG centers.

for the algorithms on the Grassmannian, the Euclidean versions of both K-means and LBG were performed on the randomly selected data in Euclidean space. The Grassmannian versions of both algorithms were then tested on a data set consisting of 1000 points in Gr(5,784), with 100 points per class. All tests were performed multiple times on the data set to account for variations due to randomized starting conditions. The best result for each algorithm was chosen based on lowest cluster distortion. The average purity across the ten clusters for Euclidean K-means is $58.84\% \pm .22$, and the average cluster purity for Euclidean LBG is $58.05\% \pm .23$; see Figure 3.

In contrast to the Euclidean algorithms, the Grassmannian algorithms performed the unsupervised clustering task very well; see Figure 4. For K-means, centers 0 through 7 have purity $> 98\%$, whereas center 8 has a purity of 60.24% and center 9 has purity of 74.63%. The average purity for the K-means algorithm is $93.19\% \pm .13$. For the LBG trial, center 2 has a purity of 90.91%, center 6 has purity of 52.63% while all other centers are 100% pure, resulting in an average purity of $94.35\% \pm .14$. Figure 5 shows the first component of each center chosen by LBG. Clearly, centers 4 and 8 can both be classified as the number 6, whereas center 6 appears to be a combination of digits 4 and 9. This highlights an interesting facet of subspace analysis. Because 4 and 9 are similar in overall shape, there is some amount of overlap in the subspaces spanned by these digits, making it difficult to distinguish the two. A similar effect is often seen with digits 7 and 9.

6.2 Indian Pines Results

Select classes from the Indian Pines data set [3] were used to further evaluate the performance of K-means and LBG. The classes *alfalfa* and *corn* were compared to test the algorithms on separable but unbalanced clusters. The data set contains 237 data points for *corn*, but only 46 for *alfalfa*. Points were generated

	Alfalfa v. Corn				**Pasture v. Trees**				**Pasture v. Trees**			
Manifold	Gr(5, 200)				Gr(5, 200)				Gr(10, 200)			
Method	K-means		LBG		K-means		LBG		K-means		LBG	
Class	1	4	1	4	5	6	5	6	5	6	5	6
Center 0	0	47	9	0	94	0	2	145	0	23	4	8
Center 1	9	0	0	47	2	146	94	1	4	0	0	15

Table 1: Results on two-class experiments on several Indian Pines classes.

in Gr(5,200) in the same manner used for the MNIST trials. Due to the class size disparity and the reduction in the total number of points when generating subspaces, class 1 (*alfalfa*) contained 9 points and class 4 (*corn*) contained 47. As seen in Table 1, both algorithms clustered the data perfectly. A second trial was performed on the classes for *pasture* and *trees*, which are unbalanced and contain overlap. Both algorithms were tested using points generated first in Gr(5,200), then in Gr(10,200). There are 483 data points for *pasture* and 730 data points for *trees*. In Gr(5,200), class 5 (*pasture*) contained 96 points and class 6 (*trees*) contained 146 points, and both methods yielded cluster purity $> 98\%$. In Gr(10,200), class 5 contained only 4 points and class 6 contained 23 points. K-means succeeded in clustering the data perfectly, but LBG had a cluster with only 66% purity. In this case, the randomly selected initial conditions were poor, which caused the algorithm to terminate in a local minimum rather than obtaining the optimal clustering. This highlights one of the pitfalls of these clustering algorithms, especially in cases where the data set is small. Results for both experiments are included in Table 1.

The final test was performed on the soybean classes *soybean with tilling* (class 10), *soybean with no tilling* (class 11), *soybeans clean* (class 12). We explore two options for clustering this data. First, we embed the points on the Grassmannian into Euclidean space by applying MDS to a matrix of pairwise smallest principal angle distances and then cluster using the standard Euclidean space algorithms. Second, we cluster directly on the manifold using the pseudometric. Constructing subspaces in Gr(5,200) similar to previous experiments resulted in 194 points in class 10, 491 points in class 11, and 118 points in class 12. A total of 10 trials were performed, and some for these experiments are displayed in Figure 6. Once again, the Euclidean algorithms yielded mediocre results, with average cluster purity less than 60%. For the LBG algorithm, it appears to be beneficial to embed the points in Gr(5,200) back into Euclidean space using MDS before clustering. This resulted in an average cluster purity of $83.74\% \pm .14$, verses $72.44\% \pm .18$ when clustering was done on the manifold itself. For K-means, however, performing clustering after embedding yielded an average purity of $83.50\% \pm .19$, with all but one trial yielding at least one cluster containing only a single point. Clustering directly on the manifold raised the average purity to $85.92\% \pm .13$ and resulted in a much more even and consistent distribution of points among centers. The best method for clustering appears to vary based on the algorithm used, and likely also changes based on the data itself.

Fig. 6: A sample of the comparison trials of the K-means and LBG algorithms on classes 10-12 of the Indian Pines data set.

7 Conclusions

In this paper we extend the K-means and LBG algorithms to the framework of the real Grassmannian. We demonstrate that both approaches result in high classification purity, i.e., the cluster membership consists of either exclusively, or predominantly, data from a single label. The flag mean provides nested subspaces that capture the essence of the signature of the data in the centroid. We are able to capture the constituent patterns and their variations, which is vital for discovering new patterns of the same class but in a different variation of state. On the Indian Pines data set, we demonstrate clustering directly on the manifold and on embeddings. These algorithms for subspace quantization provide a robust means to characterize the variability in complex, high-dimensional data sets.

References

1. Absil, P.A., Mahony, R., Sepulchre, R.: Optimization Algorithms on Matrix Manifolds. Princeton University Press, Princeton, NJ, USA (2007)
2. Adams, H., Chepushtanova, S., Emerson, T., Hanson, E., Kirby, M., Motta, F., Neville, R., Peterson, C., Shipman, P., Ziegelmeier, L.: Persistent images: A stable vector representation of persistent homology. Journal of Machine Learning **18**(1) (2017) 218–252
3. Baumgardner, M.F., Biehl, L.L., Landgrebe, D.A.: 220 band AVIRIS hyperspectral image data set: June 12, 1992 Indian Pine Test Site 3 (Sep 2015)
4. Beveridge, J.R., Draper, B.A., Chang, J.M., Kirby, M., Kley, H., Peterson, C.: Principal angles separate subject illumination spaces in YDB and CMU-PIE. IEEE Transactions on Pattern Analysis and Machine Intelligence **31**(2) (2009) 351–363

5. Björck, A., Golub, G.H.: Numerical methods for computing angles between linear subspaces. Mathematics of computation **27**(123) (1973) 579–594
6. Chang, J.M., Kirby, M., Kley, H., Peterson, C., Beveridge, J., Draper, B.: Recognition of digital images of the human face at ultra low resolution via illumination spaces. In: Springer Lecture Notes in Computer Science. Volume 4844. (2007) 733–743
7. Chepushtanova, S., Kirby, M.: Sparse Grassmannian embeddings for hyperspectral data representation and classification. IEEE Geoscience and Remote Sensing Letters **14**(3) (March 2017) 434–438
8. Chepushtanova, S., Kirby, M., Peterson, C., Ziegelmeier, L.: An application of persistent homology on Grassmann manifolds for the detection of signals in hyperspectral imagery. In: 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), IEEE (2015) 449–452
9. Chepushtanova, S., Kirby, M., Peterson, C., Ziegelmeier, L.: Persistent homology on Grassmann manifolds for analysis of hyperspectral movies. In: International Workshop on Computational Topology in Image Context. Volume 9667 of Lecture Notes in Computer Science., Springer (2016) 228–239
10. Cireşan, D.C., Meier, U., Gambardella, L.M., Schmidhuber, J.: Deep, big, simple neural nets for handwritten digit recognition. Neural computation **22**(12) (2010) 3207–3220
11. Draper, B., Kirby, M., Marks, J., Marrinan, T., Peterson, C.: A flag representation for finite collections of subspaces of mixed dimensions. Linear Algebra and its Applications **451** (2014) 15–32
12. Edelman, A., Arias, T.A., Smith, S.T.: The geometry of algorithms with orthogonality constraints. SIAM journal on Matrix Analysis and Applications **20**(2) (1998) 303–353
13. Gallivan, K.A., Srivastava, A., Liu, X., Van Dooren, P.: Efficient algorithms for inferences on Grassmann manifolds. In: Statistical Signal Processing, 2003 IEEE Workshop on, IEEE (2003) 315–318
14. Gruber, P., Theis, F.J.: Grassmann clustering. In: 2006 14th European Signal Processing Conference, IEEE (2006) 1–5
15. Kirby, M., Peterson, C.: Visualizing data sets on the Grassmannian using self-organizing mappings. In: 2017 12th International Workshop on Self-Organizing Maps and Learning Vector Quantization, Clustering and Data Visualization (WSOM), IEEE (2017) 1–6
16. Kohonen, T.: Self-organization and Associative Memory. Springer-Verlag, Berlin (1984)
17. Linde, Y., Buzo, A., Gray, R.: An algorithm for vector quantizer design. IEEE Transactions on communications **28**(1) (1980) 84–95
18. MacQueen, J., et al.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. Volume 1., Oakland, CA, USA (1967) 281–297
19. Marrinan, T., Beveridge, J.R., Draper, B., Kirby, M., Peterson, C.: Flag manifolds for the characterization of geometric structure in large data sets. In: Numerical Mathematics and Advanced Applications-ENUMATH 2013. Springer (2015) 457–465
20. Marrinan, T., Draper, B., Beveridge, J.R., Kirby, M., Peterson, C.: Finding the subspace mean or median to fit your need. In: Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on, IEEE (2014) 1082–1089