A Flag Decomposition for Hierarchical Datasets

Supplementary Material

We provide alternative methods for flag recovery in Sec. 1, proofs of each proposition in Sec. 2, a discussion of block matrix decompositions in Sec. 3, a formal presentation of the Flag-BMGS algorithm in Sec. 4, and additional details for the results in Sec. 5.

1. SVD and QR for Flag Recovery

The SVD and QR decomposition recover individual subspaces of the flag $[\mathbf{Q}_i]$ and, in certain hierarchies, recover the entire flag $[\![\mathbf{Q}]\!]$. We first discuss SVD and QR for subspace recovery, then we provide examples of using each method for flag recovery.

1.1. Subspace recovery

SVD and QR decomposition can be used to solve the optimization problem

$$\mathbf{Q}_i = \underset{\mathbf{X}\in St(m_i,n)}{\arg\min} \sum_{j\in\mathcal{B}_j} \|\mathbf{\Pi}_{\mathbf{X}^\perp}\mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp}\cdots\mathbf{\Pi}_{\mathbf{Q}_1^\perp}\tilde{\mathbf{d}}_j\|_2^2$$

with $m_i = \operatorname{rank}(\mathbf{\Pi}_{\mathbf{X}^\perp}\mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp}\cdots\mathbf{\Pi}_{\mathbf{Q}_1^\perp}\mathbf{B}_i)$. The QR decomposition with pivoting outputs $\mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp}\cdots\mathbf{\Pi}_{\mathbf{Q}_1^\perp}\mathbf{B}_i = \mathbf{Q}_i'\mathbf{R}_i'\mathbf{P}_i'^\top$. We then assign the columns of $\mathbf{Q}_i$ to columns of $\mathbf{Q}_i'$ associated with non-zero rows of $\mathbf{R}_i'$. Using the SVD, we have $\mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp}\cdots\mathbf{\Pi}_{\mathbf{Q}_1^\perp}\mathbf{B}_i = \mathbf{U}_i\mathbf{\Sigma}_i\mathbf{V}_i^\top$. Then we assign the columns of $\mathbf{Q}_i$ to the columns of $\mathbf{U}_i$ associated with non-zero singular values.

1.2. Flag recovery

Both the SVD and QR decomposition can be used for flag recovery for certain column hierarchies and flag types. In both examples, we take $\mathbf{D}\in\mathbb{R}^{n\times p}$.

Example 1.1 (QR decomposition)**.** *For a tall and skinny ($p \le n$) $\mathbf{D}$ with the column hierarchy $\{1,\dots,p_1\}\subset\{1,\dots,p_2\}\subset\cdots\subset\{1,\dots,p\}$ and the flag type is $(p_1,p_2,\dots,p;n)$, the QR decomposition $\mathbf{D}=\mathbf{QR}$ outputs the hierarchy-preserving flag $[\![\mathbf{Q}]\!]\in\mathcal{FL}(p_1,p_2,\dots,p;n)$ because $[\mathbf{q}_1|\mathbf{q}_2|\cdots|\mathbf{q}_i] = [\mathbf{d}_1|\mathbf{d}_2|\cdots|\mathbf{d}_i] = [\mathbf{D}_{\mathcal{A}_i}]$ for $i=1,2\dots,k$.*

Example 1.2 (SVD)**.** *Suppose $\mathbf{D}$ has the column hierarchy $\{1,\dots,p\}$ and the rank n_k. The SVD of $\mathbf{D}$ is $\mathbf{D}=\mathbf{U\Sigma V}^\top$. Let $\mathbf{Q}$ be the n_k left singular vectors (columns of $\mathbf{U}$) associated with non-zero singular values. Then $[\![\mathbf{Q}]\!]\in\mathcal{FL}(n_k;n)$ is a hierarchy-preserving flag because $[\mathbf{Q}]=[\mathbf{D}]$.*

2. Theoretical Justifications

We prove each proposition from the Methods section. For the sake of flow, we re-state the propositions from the Methods section before providing the proofs. Throughout these justifications we use $\operatorname{rank}(\mathbf{D})$ as the dimension of the column space of $\mathbf{D}$. This is equivalent to the dimension of the subspace spanned by the columns of $\mathbf{D}$, denoted $\dim([\mathbf{D}])$

Proposition 1. *Suppose $\mathcal{A}_1\subset\mathcal{A}_2\subset\cdots\subset\mathcal{A}_k$ is a column hierarchy for $\mathbf{D}$. Then there exists $\mathbf{Q} = [\mathbf{Q}_1\,|\,\mathbf{Q}_2\,|\,\cdots\,|\,\mathbf{Q}_k]$ that are coordinates for the flag $[\![\mathbf{Q}]\!]\in\mathcal{FL}(n_1,n_2,\dots,n_k;n)$ where $n_i=\operatorname{rank}(\mathbf{D}_{\mathcal{A}_i})$ that satisfies $[\mathbf{Q}_i] = [\mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp}\cdots\mathbf{\Pi}_{\mathbf{Q}_1^\perp}\mathbf{B}_i]$ and the **projection property** (for $i=1,2\dots,k$):*

$$\mathbf{\Pi}_{\mathbf{Q}_i^\perp}\mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp}\cdots\mathbf{\Pi}_{\mathbf{Q}_1^\perp}\mathbf{B}_i = \mathbf{0}. \tag{1}$$

Proof. For $i=1$ define $m_1=n_1=\operatorname{rank}(\mathbf{B}_1)=\operatorname{rank}(\mathbf{D}_{\mathcal{A}_1})$. Now define $\mathbf{C}_1=\mathbf{B}_1$ and $\mathbf{Q}_1\in St(m_1,n)$ whose columns are an orthonormal basis for the column space of $\mathbf{C}_1$, specifically $[\mathbf{Q}_1]=[\mathbf{C}_1]$.

For ease of notation, denote $\mathbf{Q}_{:i}=[\mathbf{Q}_1|\mathbf{Q}_2|\cdots|\mathbf{Q}_i]$. Define (for $i=2,3,\dots,k$) the projector onto the null space of $[\mathbf{Q}_1,\mathbf{Q}_2,\dots,\mathbf{Q}_i]$, as

$$\mathbf{\Pi}_{\mathbf{Q}_{:i}^\perp} = \mathbf{I}-\mathbf{Q}_{:i}\mathbf{Q}_{:i}^\top. \tag{2}$$

We use this to define $\mathbf{C}_i$ through

$$\mathbf{C}_i = \mathbf{\Pi}_{\mathbf{Q}_{:i-1}^\perp}\mathbf{B}_i \tag{3}$$

and $\mathbf{Q}_i\in St(m_i,n)$ so that $[\mathbf{Q}_i]=[\mathbf{C}_i]$.

We use mathematical induction to prove the following:

1. *Non-zero* $\mathbf{C}_i\neq\mathbf{0}$,
2. *Coordinates* $\mathbf{Q}_{:i}=[\mathbf{Q}_1|\mathbf{Q}_2|\cdots|\mathbf{Q}_i]$ is in Stiefel coordinates (*e.g.*, $\mathbf{Q}_{:i}\mathbf{Q}_{:i}^\top=\mathbf{I}$),
3. *Hierarchy* $[\mathbf{B}_1,\mathbf{B}_2,\dots,\mathbf{B}_i]=[\mathbf{Q}_1,\mathbf{Q}_2,\dots,\mathbf{Q}_i]$,
4. *Projection property* $\mathbf{\Pi}_{\mathbf{Q}_i^\perp}\cdots\mathbf{\Pi}_{\mathbf{Q}_1^\perp}\mathbf{B}_i=\mathbf{0}$ and $\mathbf{\Pi}_{\mathbf{Q}_i^\perp}\cdots\mathbf{\Pi}_{\mathbf{Q}_1^\perp}=\mathbf{\Pi}_{\mathbf{Q}_{:i}^\perp}$,
5. *Dimensions* $\mathbf{Q}_i\in St(m_i,n)$ with $m_i=n_i-n_{i-1}$ where $n_i=\operatorname{rank}(\mathbf{D}_{\mathcal{A}_i})$.

We proceed with the base case $i=1$. (1) $\mathbf{C}_1=\mathbf{B}_1=\mathbf{D}_{\mathcal{A}_1}\neq\mathbf{0}$. (2) $\mathbf{Q}_1\in St(n_1,n)$ because its columns form an orthonormal basis for $[\mathbf{C}_1]$. (3) $[\mathbf{B}_1]=[\mathbf{C}_1]=[\mathbf{Q}_1]$. (4) Since $\mathbf{\Pi}_{\mathbf{Q}_1^\perp}$ projects into the nullspace of $\mathbf{Q}_1$ and $[\mathbf{Q}_1]=[\mathbf{B}_1]$, we have $\mathbf{\Pi}_{\mathbf{Q}_1^\perp}\mathbf{B}_1=\mathbf{0}$. (5) Since $m_1=n_1$, $n_1=\dim([\mathbf{Q}_1])=\dim([\mathbf{B}_1])=\dim([\mathbf{D}_{\mathcal{A}_1}])$, and the columns of $\mathbf{Q}_1$ form an orthonormal basis, we have $\mathbf{Q}_1\in St(m_1,n)$.

Fix some $j \in \{2, 3, \ldots, k\}$. Suppose statements (1-5) hold true for all $i < j$.

1. Non-zero. By way of contradiction, assume $\mathbf{C}_j = \mathbf{0}$. Then $\mathbf{\Pi}_{\mathbf{Q}_{:j-1}^\perp} \mathbf{B}_j = \mathbf{0}$. This means each column of $\mathbf{B}_j$ is in the column space of $\mathbf{Q}_{:j-1}$. In terms of subspaces, this implies

$$[\mathbf{B}_j] \subseteq [\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_{j-1}] = [\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_{j-1}] \quad (4)$$

where the second equality follows from the induction hypothesis part 3. Eq. (4) implies

$$\dim([\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_j]) = \dim([\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_{j-1}]). \quad (5)$$

By construction (see first paragraph of Methods section),

$$\dim([\mathbf{D}_{\mathcal{A}_j}]) = \dim([\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_j]). \quad (6)$$

So Eqs. (5) and (6) imply $\operatorname{rank}(\mathbf{D}_{\mathcal{A}_j}) = \operatorname{rank}(\mathbf{D}_{\mathcal{A}_{j-1}})$. This contradicts the assumption of a column hierarchy for $\mathbf{D}$, namely $\operatorname{rank}(\mathbf{D}_{\mathcal{A}_j}) > \operatorname{rank}(\mathbf{D}_{\mathcal{A}_{j-1}})$.

2. Coordinates. It suffices to show

$$\mathbf{Q}_j^\top \mathbf{Q}_{:j-1} = [\mathbf{Q}_j^\top \mathbf{Q}_1 | \mathbf{Q}_j^\top \mathbf{Q}_2 | \cdots | \mathbf{Q}_j^\top \mathbf{Q}_{j-1}] = \mathbf{0} \quad (7)$$

which is equivalent to showing $[\mathbf{Q}_j]$ is orthogonal to $[\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_{j-1}]$. By construction,

$$[\mathbf{Q}_j] = [\mathbf{C}_j] = [\mathbf{\Pi}_{\mathbf{Q}_{:j-1}^\perp} \mathbf{B}_j] \quad (8)$$

which is orthogonal to $[\mathbf{Q}_1, \ldots, \mathbf{Q}_{j-1}]$.

3. Hierarchy. Using $\mathbf{Q}_j^\top \mathbf{Q}_{:j-1} = \mathbf{0}$, we have

$$\begin{aligned}
\mathbf{\Pi}_{\mathbf{Q}_{:j}^\perp} &= \mathbf{I} - \mathbf{Q}_{:j} \mathbf{Q}_{:j}^\top \\
&= \mathbf{I} - \sum_{\ell=1}^{j} \mathbf{Q}_\ell \mathbf{Q}_\ell^\top \\
&= \mathbf{I} - \mathbf{Q}_j \mathbf{Q}_j^\top - \mathbf{Q}_{:j-1} \mathbf{Q}_{:j-1}^\top \\
&= \mathbf{I} - \mathbf{Q}_j \mathbf{Q}_j^\top - \mathbf{Q}_{:j-1} \mathbf{Q}_{:j-1}^\top \\
&\quad + \mathbf{Q}_j \underbrace{\mathbf{Q}_j^\top \mathbf{Q}_{:j-1}}_{\mathbf{0}} \mathbf{Q}_{:j-1}^\top, \\
&= (\mathbf{I} - \mathbf{Q}_j \mathbf{Q}_j^\top)(\mathbf{I} - \mathbf{Q}_{:j-1} \mathbf{Q}_{:j-1}^\top), \\
&= \mathbf{\Pi}_{\mathbf{Q}_j^\perp} \mathbf{\Pi}_{\mathbf{Q}_{:j-1}^\perp}.
\end{aligned} \quad (9)$$

By Eq. (9) and the construction $[\mathbf{Q}_j] = [\mathbf{\Pi}_{\mathbf{Q}_{:j-1}^\perp} \mathbf{B}_j]$, we have

$$\begin{aligned}
\mathbf{0} &= \mathbf{\Pi}_{\mathbf{Q}_j^\perp} \mathbf{\Pi}_{\mathbf{Q}_{:j-1}^\perp} \mathbf{B}_j \\
&= \mathbf{\Pi}_{\mathbf{Q}_{:j}^\perp} \mathbf{B}_j, \\
&= (\mathbf{I} - \mathbf{Q}_{:j} \mathbf{Q}_{:j}^\top) \mathbf{B}_j, \\
\mathbf{B}_j &= \mathbf{Q}_{:j} \mathbf{Q}_{:j}^\top \mathbf{B}_j.
\end{aligned}$$

Thus $[\mathbf{B}_j] \subseteq [\mathbf{Q}_1, \mathbf{Q}_2, \cdots, \mathbf{Q}_j]$. By the induction hypothesis (3), $[\mathbf{B}_1, \ldots, \mathbf{B}_{j-1}] = [\mathbf{Q}_1, \ldots, \mathbf{Q}_{j-1}]$. So $[\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_j] \subseteq [\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_j]$.

In contrast $[\mathbf{B}_j] \supseteq [\mathbf{\Pi}_{\mathbf{Q}_{:j-1}^\perp} \mathbf{B}_j] = [\mathbf{C}_j] = [\mathbf{Q}_j]$. So, also using $[\mathbf{B}_1, \ldots, \mathbf{B}_{j-1}] = [\mathbf{Q}_1, \ldots, \mathbf{Q}_{j-1}]$ (induction hypothesis 3), we have $[\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_j] \supseteq [\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_j]$.

4. Projection property. Using Eq. (9) and the induction hypothesis (4) that $\mathbf{\Pi}_{\mathbf{Q}_{:j-1}^\perp} = \mathbf{\Pi}_{\mathbf{Q}_{j-1}^\perp} \cdots \mathbf{\Pi}_{\mathbf{Q}_1^\perp}$, we have

$$\begin{aligned}
\mathbf{\Pi}_{\mathbf{Q}_{:j}^\perp} &= \mathbf{\Pi}_{\mathbf{Q}_j^\perp} \mathbf{\Pi}_{\mathbf{Q}_{:j-1}^\perp}, \\
&= \mathbf{\Pi}_{\mathbf{Q}_j^\perp} \mathbf{\Pi}_{\mathbf{Q}_{j-1}^\perp} \cdots \mathbf{\Pi}_{\mathbf{Q}_1^\perp}.
\end{aligned} \quad (10)$$

By construction $[\mathbf{Q}_j] = [\mathbf{\Pi}_{\mathbf{Q}_{:j-1}^\perp} \mathbf{B}_j]$. Thus

$$\mathbf{\Pi}_{\mathbf{Q}_j^\perp} \mathbf{\Pi}_{\mathbf{Q}_{:j-1}^\perp} \mathbf{B}_j = \mathbf{0}.$$

Using Eq. (10), we have

$$\mathbf{\Pi}_{\mathbf{Q}_j^\perp} \mathbf{\Pi}_{\mathbf{Q}_{j-1}^\perp} \cdots \mathbf{\Pi}_{\mathbf{Q}_1^\perp} \mathbf{B}_j = \mathbf{0}.$$

5. Dimensions. By the induction hypothesis (5), $\mathbf{Q}_i \in St(m_i, n)$ for $i = 1, 2, \ldots, j-1$. So $\mathbf{Q}_{:j-1} \in St(n_{j-1}, n)$ with $n_{j-1} = \sum_{i=1}^{j-1} m_i$. Let

$$\begin{aligned}
n_j &= \operatorname{rank}(\mathbf{D}_{\mathcal{A}_j}), \\
&= \dim([\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_j]), \\
&= \dim([\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_j]).
\end{aligned}$$

Thus $\mathbf{Q}_{:j} \in \mathbb{R}^{n \times n_j}$ and $\mathbf{Q}_j \in \mathbb{R}^{n \times m_j}$ with $m_j = n_j - n_{j-1}$. $\mathbf{Q}_j$ has orthonormal columns by construction, so $\mathbf{Q}_j \in St(m_j, n)$.

By way of mathematical induction, we have proven (1-5) for all $i = 1, 2, \ldots, k$. Specifically, given a column hierarchy on $\mathbf{D}$, we have found coordinates for a hierarchy-preserving flag $[\![\mathbf{Q}]\!] \in \mathcal{FL}(n_1, n_2, \ldots, n_k; n)$ that satisfies the projection property. □

Although we can write $\mathbf{R}_{i,j} = \mathbf{Q}_i^\top \mathbf{B}_j$ for $j \geq i$, an equivalent definition is provided in Eq. (11) because it is used in Alg. 2.

Proposition 2. *Suppose $\mathcal{A}_1 \subset \mathcal{A}_2 \subset \cdots \subset \mathcal{A}_k$ is a column hierarchy for $\mathbf{D}$. Then there exists some hierarchy-preserving $[\![\mathbf{Q}]\!] \in \mathcal{FL}(n_1, n_2, \ldots, n_k; n)$ (with $n_i = \operatorname{rank}(\mathbf{D}_{\mathcal{A}_i})$) that satisfies the projection property of $\mathbf{D}$ and can be used for a flag decomposition of $\mathbf{D}$ with*

$$\begin{aligned}
\mathbf{R}_{i,j} &= \begin{cases} \mathbf{Q}_i^\top \mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp} \cdots \mathbf{\Pi}_{\mathbf{Q}_1^\perp} \mathbf{B}_i, & i = j \\ \mathbf{Q}_i^\top \mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp} \cdots \mathbf{\Pi}_{\mathbf{Q}_1^\perp} \mathbf{B}_j, & i < j \end{cases}, \\
\mathbf{P}_i &= \left[\mathbf{e}_{b_{i,1}} \,|\, \mathbf{e}_{b_{i,2}} \,|\, \cdots \,|\, \mathbf{e}_{b_{i,|\mathcal{B}_i|}} \right]
\end{aligned} \quad (11)$$

where $\{b_{i,j}\}_{j=1}^{|\mathcal{B}_i|} = \mathcal{B}_i$ and $\mathbf{e}_b$ is the $b_{i,j}{}^{\text{th}}$ standard basis vector.

Proof. We define the permutation matrix $\mathbf{P} = [\mathbf{P}_1|\mathbf{P}_2|\cdots|\mathbf{P}_k]$ in Eq. (11). Specifically, we assign the non-zero values in each column of $\mathbf{P}_i$ to be the index of each element in $\mathcal{B}_i$. In summary, $\mathbf{P}_i$ is defined so that $\mathbf{DP} = [\mathbf{B}_1|\mathbf{B}_2|\cdots|\mathbf{B}_k]$ and $\mathbf{D} = \mathbf{B} = [\mathbf{B}_1|\mathbf{B}_2|\cdots|\mathbf{B}_k]\mathbf{P}^\top$.

We find the coordinates $[\mathbf{Q}_1|\mathbf{Q}_2|\cdots|\mathbf{Q}_k] \in St(n_k, n)$ for the hierarchy-preserving flag $[\![\mathbf{Q}]\!] \in \mathcal{FL}(n_1, n_2, \ldots, n_k; n)$ with $n_k = \operatorname{rank}(\mathbf{D}_{\mathcal{A}_i})$ that satisfies the projection property using Prop. 1.

Now, we aim to find $\mathbf{R}$ so that $\mathbf{B} = \mathbf{QR}$. Using the projection property $\mathbf{\Pi}_{\mathbf{Q}_j^\perp} \cdots \mathbf{\Pi}_{\mathbf{Q}_1^\perp} \mathbf{B}_j = \mathbf{0}$ and the identity $\mathbf{\Pi}_{\mathbf{Q}_j^\perp} \cdots \mathbf{\Pi}_{\mathbf{Q}_1^\perp} = \mathbf{\Pi}_{[\mathbf{Q}_1|\cdots|\mathbf{Q}_j]^\perp}$ from Eq. (10), we can write

$$\mathbf{B}_j = \sum_{i=1}^{j} \mathbf{Q}_i \mathbf{Q}_i^\top \mathbf{B}_j = \sum_{i=1}^{j} \mathbf{Q}_i \mathbf{R}_{i,j}. \tag{12}$$

This is equivalent to the projection formulation of $\mathbf{R}_{i,j}$ in Eq. (11) because (for $i = 1, 2, \ldots, k$),

$$\begin{aligned}
&\mathbf{Q}_i^\top \mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp} \cdots \mathbf{\Pi}_{\mathbf{Q}_1^\perp} \\
&= \mathbf{Q}_i^\top \mathbf{\Pi}_{[\mathbf{Q}_{i-1}|\cdots|\mathbf{Q}_1]^\perp}, \\
&= \mathbf{Q}_i^\top (\mathbf{I} - [\mathbf{Q}_{i-1}|\cdots|\mathbf{Q}_1][\mathbf{Q}_{i-1}|\cdots|\mathbf{Q}_1]^\top), \\
&= \mathbf{Q}_i^\top - \underbrace{\mathbf{Q}_i^\top [\mathbf{Q}_{i-1}|\cdots|\mathbf{Q}_1]}_{\mathbf{0}} [\mathbf{Q}_{i-1}|\cdots|\mathbf{Q}_1]^\top, \\
&= \mathbf{Q}_i^\top.
\end{aligned} \tag{13}$$

Stacking the results from Eqs. (12) and (13) into block matrices gives $\mathbf{B} = \mathbf{QR}$ with $\mathbf{R}$ defined in Eq. (11). □

Proposition 3. *A data matrix $\mathbf{D}$ admits a flag decomposition of type $(n_1, n_2, \cdots, n_k; n)$ if and only if $\mathcal{A}_1 \subset \mathcal{A}_2 \subset \cdots \subset \mathcal{A}_k$ is a column hierarchy for $\mathbf{D}$.*

Proof. We first tackle the forward direction. Suppose $\mathbf{D}$ admits a flag decomposition with the hierarchy $\mathcal{A}_1 \subset \mathcal{A}_2 \subset \cdots \subset \mathcal{A}_k$. Then $\mathbf{D} = \mathbf{QRP}^\top$ and $\mathbf{DP} = \mathbf{QR}$ because $\mathbf{P}$ is a permutation matrix. Define

$$\mathbf{B} = [\mathbf{B}_1|\mathbf{B}_2|\cdots|\mathbf{B}_k] = \mathbf{DP} = \mathbf{QR}. \tag{14}$$

Since we have a flag decomposition, $[\![\mathbf{Q}]\!] \in \mathcal{FL}(n_1, n_2, \ldots, n_k; n)$ with $\mathbf{Q} = [\mathbf{Q}_1|\mathbf{Q}_2|\cdots|\mathbf{Q}_k] \in St(n_k, n)$. Since $\mathbf{Q}$ is in Stiefel coordinates we have $\mathbf{Q}_j^T \mathbf{Q}_i = \mathbf{0}$ for all $j < i$, so

$$\dim([\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_{i-1}]) < \dim([\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_i]). \tag{15}$$

Since $[\![\mathbf{Q}]\!]$ is hierarchy preserving, for $i = 1, 2, \ldots, k$ we have $[\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_i] = [\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_i]$. Using this and Eq. (15), we have

$$\dim([\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_{i-1}]) < \dim([\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_i]). \tag{16}$$

By construction $\dim([\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_i]) = \dim([\mathbf{D}_{\mathcal{A}_i}])$. So, using Eq. (16), we have shown $\dim([\mathbf{D}_{\mathcal{A}_{i-1}}]) < \dim([\mathbf{D}_{\mathcal{A}_i}])$ proving $\mathcal{A}_1 \subset \mathcal{A}_2 \subset \cdots \subset \mathcal{A}_k$ is a column hierarchy for $\mathbf{D}$.

The backward direction is proved in Prop. 1 and 2. Specifically, given a data matrix with an associated column hierarchy, Prop. 1 describes how to find a hierarchy-preserving flag. Then Prop. 2 shows how to find the permutation matrix $\mathbf{P}$ from the column hierarchy and the weight matrix $\mathbf{R}$ from $\mathbf{Q}$ so that $\mathbf{D} = \mathbf{QRP}^\top$. □

Recall the two optimization problems proposed in the Methods section:

$$[\![\mathbf{Q}]\!] = \underset{[\![\mathbf{X}]\!] \in \mathcal{FL}(n_1, n_2, \ldots, n_k; n)}{\arg\min} \sum_{i=1}^{k} \sum_{j \in \mathcal{B}_i} \|\mathbf{\Pi}_{\mathbf{X}_1^\perp} \cdots \mathbf{\Pi}_{\mathbf{X}_1^\perp} \tilde{\mathbf{d}}_j\|_r^q, \tag{17}$$

$$\mathbf{Q}_i = \underset{\mathbf{X} \in St(m_i, n)}{\arg\min} \sum_{j \in \mathcal{B}_j} \|\mathbf{\Pi}_{\mathbf{X}^\perp} \mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp} \cdots \mathbf{\Pi}_{\mathbf{Q}_1^\perp} \tilde{\mathbf{d}}_j\|_r^q. \tag{18}$$

Proposition 4 (Block rotational ambiguity)**.** *Given the FD $\mathbf{D} = \mathbf{QRP}^\top$, any other Stiefel coordinates for the flag $[\![\mathbf{Q}]\!]$ produce an FD of $\mathbf{D}$ (via Prop. 2). Furthermore, different Stiefel coordinates for $[\![\mathbf{Q}]\!]$ produce the same objective function values in Eqs. (17) and (18) (for $i = 1, \cdots, k$).*

Proof. The flag manifold $\mathcal{FL}(n_1, n_2, \ldots, n_k; n)$ is diffeomorphic to $St(n_k, n)/(O(m_1) \times \cdots \times O(m_k))$ where $m_i = n_i - n_{i-1}$. Suppose $\mathbf{D} = \mathbf{QRP}^\top$ is a flag decomposition. Consider $\mathbf{QM} \in St(n_k, n)$ with $\mathbf{M} = \operatorname{diag}([\mathbf{M}_1|\mathbf{M}_2|\cdots|\mathbf{M}_k]) \in O(m_1) \times \cdots \times O(m_k)$, meaning $\mathbf{M}_i \in O(m_i)$ for $i = 1, 2, \ldots, k$.

Notice $\mathbf{Q}$ and $\mathbf{QM}$ are coordinates for the same flag, $[\![\mathbf{Q}]\!] = [\![\mathbf{QM}]\!]$.

The key property for this proof is that right multiplication by $\mathbf{M}_i$ does not change projection matrices $\mathbf{Q}_i \mathbf{Q}_i^\top$. Specifically $\mathbf{Q}_i \mathbf{Q}_i^\top = \mathbf{Q}_i \mathbf{M}_i (\mathbf{Q}_i \mathbf{M}_i)^\top$ for $i = 1, 2, \ldots, k$.

Both $\mathbf{Q}$ and $\mathbf{MQ}$ satisfy the projection property relative to $\mathbf{D}$ because (for $i = 1, 2, \ldots, k$)

$$\mathbf{\Pi}_{\mathbf{Q}_i^\perp} = \mathbf{I} - \mathbf{Q}_i \mathbf{Q}_i^\top = \mathbf{I} - \mathbf{Q}_i \mathbf{M}_i (\mathbf{M}_i \mathbf{Q}_i)^\top = \mathbf{\Pi}_{(\mathbf{Q}_i \mathbf{M}_i)^\perp} \tag{19}$$

which implies that the objective function values in Eqs. (17) and (18) (for $i = 1, 2, \ldots, k$) are the same for $\mathbf{Q}$ and $\mathbf{QM}$. Additionally, Eq. (19) implies

$$\mathbf{0} = \mathbf{\Pi}_{\mathbf{Q}_i^\perp} \mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp} \cdots \mathbf{\Pi}_{\mathbf{Q}_1^\perp} \mathbf{B}_i \tag{20}$$

$$= \mathbf{\Pi}_{(\mathbf{Q}_i \mathbf{M}_i)^\perp} \mathbf{\Pi}_{(\mathbf{Q}_{i-1} \mathbf{M}_{i-1})^\perp} \cdots \mathbf{\Pi}_{(\mathbf{Q}_1 \mathbf{M}_1)^\perp} \mathbf{B}_i. \tag{21}$$

Since $[\![\mathbf{Q}]\!]$ is hierarchy-preserving and rotations do not change subspaces, we have $[\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_i] = [\mathbf{Q}_1 \mathbf{M}_1, \mathbf{Q}_2 \mathbf{M}_2, \ldots, \mathbf{Q}_i \mathbf{M}_i]$. Thus $[\![\mathbf{QM}]\!]$ is hierarchy-preserving.

Define $\mathbf{R}^{(\mathbf{M})}$ with blocks $\mathbf{R}^{(\mathbf{M})}_{i,j} = (\mathbf{Q}_i\mathbf{M}_i)^\top\mathbf{B}_j$. Notice

$$\mathbf{B}_j = \sum_{i=1}^{j} \mathbf{Q}_i\mathbf{R}_{i,j}, \tag{22}$$

$$= \sum_{i=1}^{j} \mathbf{Q}_i\mathbf{Q}_i^\top\mathbf{B}_j, \tag{23}$$

$$= \sum_{i=1}^{j} (\mathbf{Q}_i\mathbf{M}_i)(\mathbf{Q}_i\mathbf{M}_i)^\top\mathbf{B}_j, \tag{24}$$

$$= \sum_{i=1}^{j} (\mathbf{Q}_i\mathbf{M}_i)\mathbf{R}^{(\mathbf{M})}_{i,j}. \tag{25}$$

Thus $\mathbf{D} = (\mathbf{QM})\mathbf{R}^{(\mathbf{M})}\mathbf{P}^\top$ is a hierarchy-preserving flag decomposition. □

3. Relationship to MLMD [2]

The Multiscale Low Rank Matrix Decomposition (MLMD) [2] models $\mathbf{D} = \sum_i \mathbf{X}_i$ where each block low-rank matrix $\mathbf{X}_i$ models finer-grained features than $\mathbf{X}_{i+1}$. Suppose $\mathbf{D} = [\mathbf{B}_1|\mathbf{B}_2] \in \mathbb{R}^{n\times p}$ is of rank n_k with columns sorted according to the hierarchy $\mathcal{A}_1 \subset \mathcal{A}_2$. The FD with flag type $(n_1, n_2; n)$ is $\mathbf{D} = \mathbf{QR}$ where $\mathbf{Q} = [\mathbf{Q}_1|\mathbf{Q}_2] \in St(n_k, n)$, $\mathbf{Q}_1 \in \mathbb{R}^{n\times n_1}$, and $\mathbf{R}$ is block upper triangular. FD does not seek block low-rank representations for different scales, rather it extracts a hierarchy-preserving flag $[\![\mathbf{Q}]\!] \in \mathcal{FL}(n_1, n_2; n)$. Moreover, MLMD partitions $\mathbf{D}$ into column blocks requiring the block partition P_2 to be an 'order of magnitude' larger than P_1 (1st par. Sec. II). FD is more general and free of this restriction. MLMD models $\mathbf{D} = \mathbf{X}_1 + \mathbf{X}_2$ with $\mathbf{X}_i = \sum_{b\in P_i} R_b(\mathbf{U}_b\mathbf{\Sigma}_b\mathbf{V}_b^\top)$ where R_b is a block reshaper. The output would be 3 bases (in each $\mathbf{U}_b$), two for the columns of $\mathbf{B}_1$ and $\mathbf{B}_2$, and one for all of $\mathbf{D}$. These are neither mutually orthogonal nor guaranteed to be hierarchy-preserving. FD outputs one basis in the columns of $\mathbf{Q} = [\mathbf{Q}_1|\mathbf{Q}_2]$ are hierarchy-preserving: $[\mathbf{Q}_1] = [\mathbf{B}_1]$, and $[\mathbf{Q}] = [\mathbf{D}]$.

4. Algorithms

Our get_basis algorithm extracts $\mathbf{Q}_i \in St(m_i, n)$ from $\mathbf{C}_i \in \mathbb{R}^{n\times|\mathcal{B}_i|}$ so that $[\mathbf{Q}_i] = [\mathbf{C}_i]$ by solving the optimization inspired by Eq. (18):

$$\mathbf{Q}_i = \underset{\mathbf{X}\in St(m_i,n)}{\arg\min} \sum_{j=1}^{|\mathcal{B}_i|} \|\mathbf{\Pi}_{\mathbf{X}^\perp}\mathbf{c}_j^{(i)}\|_2^q \tag{26}$$

for $q = 1, 2$. We use $\mathbf{c}_j^{(i)}$ to denote the jth column of $\mathbf{C}_i$ and $\mathcal{B}_i = \mathcal{A}_i \backslash \mathcal{A}_{i-1}$. A naive implementation of IRLS-SVD addresses $q = 1$ and SVD addresses $q = 2$.

Algorithm 1: get_basis

Input: $\mathbf{C}_i \in \mathbb{R}^{n\times|\mathcal{B}_i|}$, $m_i \in \mathbb{R}$ (optional)
Output: $\mathbf{X}_i \in \mathbb{R}^{m_i}$

```
if SVD then
    U Σ V^T ← SVD(C_i);
    if m_i is none then
        m_i ← rank(C_i);
    Q_i ← U(1 : end, 1 : m_i);
if IRLS-SVD then
    while not converged do
        for j ← 1 to |B_i| do
            c_j^(i) ← C_i(1 : end, j);
            w_j ← max( ||c_j^(i) − Q_i Q_i^T c_j^(i)||_2, 10^−8 )^(−1/2);
        W_i ← diag(w_1, w_2, ..., w_|B_i|);
        U Σ V^T ← SVD(C_i W_i);
        if m_i is none then
            m_i ← rank(C_i W_i);
        Q_i ← U(1 : end, 1 : m_i);
```

Flag-BMGS is essentially BMGS [1] with a different get_basis function. The get_basis in Alg. 1 is used at each iteration of Flag-BMGS to extract a $\mathbf{Q}_i \in St(m_i, n)$ so that $[\mathbf{Q}_i] = [\mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp}\cdots\mathbf{\Pi}_{\mathbf{Q}_1^\perp}\mathbf{B}_i]$. The second nested for loop in Flag-BMGS defines $\mathbf{R}_{i,j}$ using Eq. (11) and updates $\mathbf{B}_j$ so that, at iteration i, we take run get_basis on $\mathbf{C}_i = \mathbf{\Pi}_{\mathbf{Q}_{i-1}^\perp}\cdots\mathbf{\Pi}_{\mathbf{Q}_1^\perp}\mathbf{B}_i$.

Remark 1 (Flag-BMGS operations count)**.** *In this remark, we use O to denote big-O notation and not the orthogonal group. We also denote $b_i = |\mathcal{B}_i|$ so $\mathbf{B}_i \in \mathbb{R}^{n\times b_i}$.*

The operations count for the SVD of a matrix $\mathbf{B}_i$ is $O(nb_i\min(n, b_i))$. FD runs k SVDs for each piece of the column hierarchy. Thus its operations count is $O\left(n\sum_{i=1}^k b_i\min(n, b_i)\right)$.

The IRLS-SVD operations count is $O(c_i nb_i\min(n, b_i))$ where c_i is the number of iterations until convergence for IRLS-SVD on $\mathbf{B}_i$. Since IRLS-SVD is run k times in Robust FD, the operations count is $O\left(n\sum_{i=1}^k c_i b_i\min(n, b_i)\right)$.

We summarize the properties for flag recovery methods in Tab. 1.

5. Results

We first describe data generation for each simulation. Then we provide details on the hyperspectral image clustering experiment and confidence intervals for few-shot learning.

Algorithm 2: Flag-BMGS

Input: A data matrix $\mathbf{D} \in \mathbb{R}^{n\times p}$,
c. hierarchy $\mathcal{A}_1 \subset \mathcal{A}_2 \subset \cdots \subset \mathcal{A}_k = \{1, 2, \ldots, p\}$,
flag type $(n_1, n_2, \cdots, n_k; n)$ with $n_k \leq p$
Output: Hierarchy-preserving flag
$[\![\mathbf{Q}]\!] \in \mathcal{FL}(n_1, n_2, \ldots, n_k; n)$,
weights $\mathbf{R} \in \mathbb{R}^{n_k\times p}$, perm. mat. $\mathbf{P} \in \mathbb{R}^{p\times p}$
with $\mathbf{D} = \mathbf{QRP}^\top$

for $i \leftarrow 1$ **to** k **do**
  $\mathcal{B}_i \leftarrow \mathcal{A}_i \setminus \mathcal{A}_{i-1}$;
  $\mathbf{B}_i \leftarrow \mathbf{D}(1:\text{end}, \mathcal{B}_i) \in \mathbb{R}^{n\times|\mathcal{B}_i|}$;
  $\mathbf{P}_i \leftarrow \left[\mathbf{e}_{b_{i,1}}|\mathbf{e}_{b_{i,2}}|\cdots|\mathbf{e}_{b_{i,|\mathcal{B}_i|}}\right]$

for $i \leftarrow 1$ **to** k **do**
  $m_i \leftarrow n_i - n_{i-1}$;
  $\mathbf{Q}_i \leftarrow \text{get\_basis}(\mathbf{B}_i, m_i)$;
  $\mathbf{R}_{i,i} \leftarrow \mathbf{Q}_i^\top \mathbf{B}_i$;
  for $j \leftarrow i+1$ **to** k **do**
    $\mathbf{R}_{i,j} \leftarrow \mathbf{Q}_i^\top \mathbf{B}_j$; %assign $\mathbf{R}_{i,j}$
    $\mathbf{B}_j \leftarrow \mathbf{B}_j - \mathbf{Q}_i\mathbf{R}_{i,j}$; %project: $\mathbf{B}_j$ into nullspace of $\mathbf{Q}_i$

$\mathbf{Q} \leftarrow \left[\mathbf{Q}_1|\mathbf{Q}_2|\cdots|\mathbf{Q}_k\right]$;

$$\mathbf{R} \leftarrow \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} & \cdots & \mathbf{R}_{1k} \\ \mathbf{0} & \mathbf{R}_{22} & \cdots & \mathbf{R}_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{R}_{kk} \end{bmatrix};$$

$\mathbf{P} \leftarrow \left[\mathbf{P}_1|\mathbf{P}_2|\cdots|\mathbf{P}_k\right]$;

Table 1. A summary of flag recovery methods and their properties.

Decomp.	QR	SVD	IRLS-SVD	FD	RFD
Robust	✗	✗	✓	✗	✓
Order-pres.	✓	✗	✗	✓	✓
Flag-type	✗	✗	✗	✓	✓
Hier.-pres.	✗	✗	✗	✓	✓

5.1. Reconstruction Simulations

We consider either additive noise to the data or data contamination with outliers. For both experiments, we generate a Stiefel matrix $[\mathbf{X}_1|\mathbf{X}_2] = \mathbf{X} \in St(10, 4)$ that represents $[\![\mathbf{X}]\!] \in \mathcal{FL}(2, 4; 10)$. Then we generate the data matrix $\mathbf{D}$ with the feature hierarchy $\mathcal{A}_1 \subset \mathcal{A}_2 = \{1, 2, \cdots, 20\} \subset \{1, 2, \cdots, 40\}$. We attempt to recover $[\![\mathbf{X}]\!]$ and $\mathbf{D} = [\mathbf{B}_1|\mathbf{B}_2] \in \mathbb{R}^{10\times 40}$ using FD and Robust FD with a flag type of $(2, 4; 10)$, and the first 4 left singular vectors from SVD. We evaluate the estimated $[\![\hat{\mathbf{X}}]\!]$ and $\hat{\mathbf{D}}$ using chordal distance and LRSE.

Additive noise. We consider the following model for $\mathbf{D}$:

$$\mathbf{d}_i = \begin{cases} \mathbf{X}_1\mathbf{s}_{1i}, & i \in \mathcal{B}_1 = \{1, \cdots, 20\} \\ \mathbf{X}\mathbf{s}_{2i}, & i \in \mathcal{B}_2 = \{21, \cdots, 40\} \end{cases}$$

where each entry of $\mathbf{s}_{ji}$ from a normal distribution with mean 0 and variance 1. We contaminate $\mathbf{D}$ with noise by $\tilde{\mathbf{D}} = \mathbf{D} + \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon}$ is sampled from either a normal, exponential, or uniform distribution of increasing variance. The goal is to recover $\mathbf{D}$ and $\mathbf{X}$ from $\tilde{\mathbf{D}}$. FD and Robust FD improve flag recovery over SVD and produce similar reconstruction errors.

Outliers columns. We randomly sample a subset of columns of $\tilde{\mathbf{D}}$ to be in the set of outliers $\mathcal{O}$. Each outlier column is in the nullspace of $[\mathbf{X}]$ and each entry of $\mathbf{o}_i$ is sampled from a normal distribution with mean 0 and variance 1. We use the same scheme as the additive noise case to sample $\mathbf{s}_{ji}$. Using these quantities, we sample the i^{th} column of $\tilde{\mathbf{D}}$ as

$$\tilde{\mathbf{d}}_i = \begin{cases} \mathbf{X}_1\mathbf{s}_{1i}, & i \in \mathcal{B}_1 \setminus \mathcal{O} \\ \mathbf{X}\mathbf{s}_{2i}, & i \in \mathcal{B}_2 \setminus \mathcal{O} \\ (\mathbf{I} - \mathbf{X}\mathbf{X}^T)\mathbf{o}_i, & i \in \mathcal{O}. \end{cases}$$

We define $\mathbf{D}$ as the matrix containing only inlier columns of $\tilde{\mathbf{D}}$. We attempt to recover $\mathbf{D}$ and $[\![\hat{\mathbf{X}}]\!]$ from $\tilde{\mathbf{D}}$. We measure the chordal distance between our estimated $[\![\hat{\mathbf{X}}]\!]$ and $[\![\mathbf{X}]\!]$ and the LRSE between our inlier estimates $\hat{\mathbf{D}}$ and $\mathbf{D}$.

5.2. Clustering Simulation

We generate three Stiefel matrices to serve as centers of our clusters $\left[\mathbf{X}_1^{(c)}|\mathbf{X}_2^{(c)}\right] = \mathbf{X}^{(c)} \in St(4, 10)$ that represent $[\![\mathbf{X}^{(c)}]\!] \in \mathcal{FL}(2, 4; 10)$ for $c = 1, 2, 3$. We use each of these centers to generate 20 $\mathbf{D}$-matrices with the feature hierarchy $\mathcal{A}_1 = \{1, 2, \cdots, 20\}$, $\mathcal{A}_2 = \{1, 2, \cdots, 40\}$ in each cluster. The ith column in cluster c of the data matrix $\mathbf{D}_i^{(c)}$ is generated as

$$\mathbf{d}_i^{(c)} = \begin{cases} \mathbf{X}_1^{(c)}\mathbf{s}_{1i}, & i \in \mathcal{B}_1 \\ \mathbf{X}^{(c)}\mathbf{s}_{2i}, & i \in \mathcal{B}_2. \end{cases} \tag{27}$$

Then we generate the detected data matrices as $\tilde{\mathbf{D}}_i^{(c)} = \mathbf{D}_i^{(c)} + \boldsymbol{\epsilon}_i^{(c)}$. We sample $\boldsymbol{\epsilon}_{1i}^{(c)}$ and $\boldsymbol{\epsilon}_{2i}^{(c)}$ from a normal distribution with mean 0 and standard deviation $.95$ and $\mathbf{s}_{1i}$ and $\mathbf{s}_{2i}$ from a normal distribution with mean 0 and standard deviation 1.

5.3. Hyperspectral image clustering

A total of 326 patches were extracted, each with a shape of (3×3), with the following distribution: 51 patches of class Scrub, 7 of Willow swamp, 12 of Cabbage palm hammock, 10 of Cabbage palm/oak hammock, 11 of Slash pine,

13 of Oak/broad leaf hammock, 7 of Hardwood swamp, 20 of Graminoid marsh, 39 of Spartina marsh, 25 of Cattail marsh, 29 of Salt marsh, 24 of Mudflats, and 78 of Water.

In this experiment, we measure the distance between two flags $[\![\mathbf{X}]\!], [\![\mathbf{Y}]\!]$ as

$$\frac{1}{\sqrt{2}}\|\mathbf{X}_1\mathbf{X}_1^T - \mathbf{Y}_1\mathbf{Y}_1^T\|_F + \frac{1}{\sqrt{2}}\|\mathbf{X}_2\mathbf{X}_2^T - \mathbf{Y}_2\mathbf{Y}_2^T\|_F. \quad (28)$$

5.4. Few-shot learning

We now expand on the methodological details of the baseline methods for few-shot learning and report further results including standard deviations.

Prototypical networks. Prototypical networks [4] are a classical few-shot architecture that uses averages for class representatives and Euclidean distance for distances between representatives and queries. Specifically, a prototype for class c is

$$\mathbf{q} = \frac{1}{s}\sum_{i=1}^{s} f_\Theta(\boldsymbol{x}_{c,i}) \quad (29)$$

and the distance between a query point, $f_\Theta(\boldsymbol{x})$, is

$$\|\mathbf{q} - f_\Theta(\boldsymbol{x})\|_2^2. \quad (30)$$

In experiments, we refer to this method as 'Euc.'

Subspace classifiers. Subspace classifiers from adaptive subspace networks [3] use subspace representatives and measure distances between subspace representatives and queries via projections of the queries onto the subspace representatives. Although the original work suggests mean subtraction before computing subspace representatives and for classification, we notice that there is no mean-subtraction in the code provided on GitHub. Therefore, we summarize the model used on GitHub as

$$\tilde{\boldsymbol{X}}_c = [f_\Theta(\boldsymbol{x}_{c,1})|f_\Theta(\boldsymbol{x}_{c,2})|\cdots|f_\Theta(\boldsymbol{x}_{c,s}),] \quad (31)$$

$$\mathbf{U}_c\boldsymbol{\Sigma}_c\mathbf{V}_c^\top = \tilde{\boldsymbol{X}}_c, \quad (32)$$

$$\mathbf{Q}_c = \mathbf{U}_c(1:\text{end}, 1:s-1). \quad (33)$$

We say that the span of the columns of $\mathbf{Q}_c$ serves as the subspace representative for class c. This can be seen as a mapping of a set feature space representation of the shots from one class to $Gr(s-1,n)$ via the SVD. The distance between a query $f_\Theta(\boldsymbol{x})$ and class c is

$$\|f_\Theta(\boldsymbol{x}) - \mathbf{Q}_c\mathbf{Q}_c^\top f_\Theta(\boldsymbol{x})\|_F^2. \quad (34)$$

This is the residual of the projection of a query point onto the subspace representative for class c.

Stacking features. Our application of flag classifiers uses an alexnet backbone $f_\Theta = f_\Theta^{(2)} \circ f_\Theta^{(1)}$. Given a sample $\boldsymbol{f}$, flag classifiers leverage both the information extracted by f_Θ *and* $f_\Theta^{(1)}$. This is already an advantage over the baseline methods because flag classifiers see more features. Therefore, we modify prototypical network and subspace classifiers for a fair baseline to flag nets. Specifically, we replace $f_\Theta(\boldsymbol{x})$ with

$$\begin{bmatrix} f_\Theta^{(1)}(\boldsymbol{x}) \\ f_\Theta(\boldsymbol{x}) \end{bmatrix}. \quad (35)$$

This doubles the dimension of the extracted feature space and thereby exposes these algorithms to problems like the curse of dimensionality. Additionally, it assumes *no order* on the features extracted by f_Θ and $f_\Theta^{(1)}$ therein not respecting the natural hierarchy of the alexnet feature extractor.

Further results. We provide the classification accuracies along with standard deviations over 20 random trials in Tabs. 2 and 3.

Table 2. *Classification accuracy (↑)* with s shots, 5 ways, and 100 evaluation tasks each containing 10 query images, averaged over 20 random trials. Flag types for 'Flag' are $(s-1, 2(s-1))$ and the subspace dimension is $s-1$. Baselines see stacked features from both $f_\Theta^{(1)}$ and f_Θ.

s	Dataset	Flag	Euc.	Subsp.
3	EuroSat	**77.7** ± 1.0	76.7 ± 1.0	77.6 ± 1.0
	CIFAR-10	**59.6** ± 1.0	58.6 ± 0.9	**59.6** ± 1.0
	Flowers102	**90.2** ± 0.7	88.2 ± 1.0	**90.2** ± 0.7
5	EuroSat	**81.8** ± 0.7	80.7 ± 0.8	**81.8** ± 0.7
	CIFAR-10	**65.2** ± 0.9	**65.2** ± 0.9	**65.2** ± 0.9
	Flowers102	**93.2** ± 0.5	91.4 ± 0.6	**93.2** ± 0.5
7	EuroSat	**83.9** ± 0.8	82.6 ± 0.8	83.8 ± 0.8
	CIFAR-10	68.0 ± 0.7	**68.6** ± 0.8	68.1 ± 0.7
	Flowers102	**94.5** ± 0.5	92.7 ± 0.5	**94.5** ± 0.5

Table 3. *Classification accuracy (↑)* with s shots, 5 ways, and 100 evaluation tasks each containing 10 query images, averaged over 20 random trials. Flag types for 'Flag' are $(s-1, 2(s-1))$ and the subspace dimension is $s-1$. Baselines see features only from f_Θ.

s	Dataset	Flag	Euc.	Subsp.
3	EuroSat	**77.7** ± 1.0	75.9 ± 0.9	76.8 ± 1.1
	CIFAR-10	**59.6** ± 1.0	58.4 ± 0.8	58.5 ± 0.9
	Flowers102	**90.2** ± 0.7	87.9 ± 0.9	88.8 ± 0.8
5	EuroSat	**81.8** ± 0.7	79.8 ± 0.8	80.8 ± 0.8
	CIFAR-10	**65.2** ± 0.9	64.5 ± 1.0	63.8 ± 0.9
	Flowers102	**93.2** ± 0.5	91.1 ± 0.6	92.0 ± 0.5
7	EuroSat	**83.9** ± 0.8	81.7 ± 0.8	82.9 ± 0.8
	CIFAR-10	**68.0** ± 0.7	67.9 ± 0.8	66.7 ± 0.7
	Flowers102	**94.5** ± 0.5	92.3 ± 0.5	93.4 ± 0.5

References

[1] William Jalby and Bernard Philippe. Stability analysis and improvement of the block Gram–Schmidt algorithm. *SIAM journal on scientific and statistical computing*, 12(5):1058–1073, 1991. 4

[2] Frank Ong and Michael Lustig. Beyond low rank+ sparse: Multiscale low rank matrix decomposition. *IEEE journal of selected topics in signal processing*, 10(4), 2016. 4

[3] Christian Simon, Piotr Koniusz, Richard Nock, and Mehrtash Harandi. Adaptive subspaces for few-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4136–4145, 2020. 6

[4] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017. 6