# Select to *Better* Learn: Fast and Accurate Deep Learning using Data Selection from Nonlinear Manifolds: Supplementary Material

Mohsen Joneidi*, Saeed Vahidian†, Ashkan Esmaeili*, Weijia Wang†, Nazanin Rahnavard*, Bill Lin†, and Mubarak Shah#

* University of Central Florida, Department of Electrical Engineering and Computer Science

† University of California, San Diego, Department of Electrical and Computer Engineering

# University of Central Florida, Center for Research in Computer Vision

The supplementary material provided in this document is organized as follows. In Section 1, we present a theoretical result on the equivalence of the locally linear selection with the linear selection after applying kernel. Then, in Section 2, further experiments are provided to investigate the performance of the proposed approaches on several different real datasets.

## 1. Theoretical Results

The following lemma shows how the introduced locally linear selection problem in the original paper can turn into the plain linear selection on the kernelized version of data.

**Lemma 1** *Consider $M$ data points and the neighborhood for each one are denoted by $\boldsymbol{a}_m$ and $\Omega_m$, respectively. The following problems have the same selection results using the SP algorithm.*

$$P1: \underset{|\mathbb{S}|\leq K}{argmin} \sum_{m=1}^{M} \|\boldsymbol{a}_m - \pi_{\mathbb{S}_m}(\boldsymbol{a}_m)\|_F^2 \;\; s.t.\; \mathbb{S}_m \subseteq \mathbb{S} \cap \Omega_m,$$

*and,*

$$P2: \underset{|\mathbb{S}|\leq K}{argmin} \|\boldsymbol{H} - \pi_{\mathbb{S}}(\boldsymbol{H})\|_2^2,$$

*where $h_{ij} = [\ |\Omega_i \cap \Omega_j| \boldsymbol{a}_i^T \boldsymbol{a}_j]$ and $|.|$ denotes the cardinality of a set.*

**Proof of Lemma 1**: Matrix $\boldsymbol{X}_m \in \mathbb{R}^{M\times N}$ is defined as an all-zero matrix except in rows indexed by $\Omega_m$. The non-zero rows are equal to $\boldsymbol{a}_m^T$ (repeated for all those rows). Matrix $\boldsymbol{X} \in \mathbb{R}^{MN\times M}$ is defined as follows,

$$\boldsymbol{X} = [vec(\boldsymbol{X}_1), \cdots, vec(\boldsymbol{X}_M)].$$

Operator $vec(.)$ reshapes a matrix into a vector. Using the definition of $\boldsymbol{X}$, Problem P1 can be cast in terms of $\boldsymbol{X}$ as follows,

$$\underset{|\mathbb{S}|\leq K}{\text{argmin}} \sum_{m=1}^{M} \|\boldsymbol{x}_m - \pi_{\mathbb{S}}(\boldsymbol{x}_m)\|_2^2$$

Please note that neighborhood information has been infused in matrix $\boldsymbol{X}$ and neighborhood constraints are removed in comparison with P1. In other words, each data is allowed to be approximated using its neighbors as aimed by P1. Non-neighbor samples have no impact on the least square cost function since $\boldsymbol{x}_i^T \boldsymbol{x}_j = 0$ for all pairs of $(i, j)$ non-neighbor samples. Thus, P1 can be re-written as,

$$\underset{|\mathbb{S}|\leq K}{\text{argmin}} \|\boldsymbol{X} - \pi_{\mathbb{S}}(\boldsymbol{X})\|_F^2.$$

Given the singular value decomposition of $\boldsymbol{X}$ as $\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$, where $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal matrices and $\boldsymbol{\Sigma}$ is the diagonal matrix of singular values, we have $\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T = \boldsymbol{V}\boldsymbol{\Sigma}^2\boldsymbol{V}^T$. Thus, the $k$-th left eigenvector of $\boldsymbol{X}^T\boldsymbol{X}$ is a scaled version of $\boldsymbol{v}_k$, the $k$-th column of $\boldsymbol{V}$. Moreover,

$$\boldsymbol{X}^T\boldsymbol{u}_k = \sum_{i=1}^{rank(\boldsymbol{X})} \sigma_i \boldsymbol{v}_i \boldsymbol{u}_i^T \boldsymbol{u}_k = \sigma_k \boldsymbol{v}_k, \qquad (1)$$

where the last equality follows from orthogonality of $\boldsymbol{U}$. Therefore, $\boldsymbol{X}^T\boldsymbol{u}_k$ is a scaled version of $\boldsymbol{v}_k$, the $k$-th left eigenvector of $\boldsymbol{X}^T\boldsymbol{X}$.

As the following step of the proof, we proceed to state that the same data index, $m_1$, which maximizes $|\boldsymbol{x}_m^T\boldsymbol{u}_k|$ also maximizes $|\boldsymbol{h}_m^T\boldsymbol{X}^T\boldsymbol{u}_k|$, where $\boldsymbol{h}_m$ is the $m^{\text{th}}$ column of $\boldsymbol{H} = \boldsymbol{X}^T\boldsymbol{X}$. This can be proved as follow. Let

$$m_1 = \underset{m}{\text{argmax}}\ |\boldsymbol{x}_m^T\boldsymbol{u}_k\ |, \qquad (2)$$

i.e., the index which picks the largest magnitude in vector $\boldsymbol{X}^T\boldsymbol{u}_k = \sigma_k\boldsymbol{v}_k$. Similarly, one can write $\boldsymbol{h}_m^T\boldsymbol{X}^T\boldsymbol{u}_k =$

$[(\boldsymbol{X}^T\boldsymbol{X})_m]^T\boldsymbol{X}^T\boldsymbol{u}_k$. Let $m_2 = \underset{m}{\text{argmax}}\ w_m \quad \text{s.t. } \boldsymbol{w} = |\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{u}_k|$. We have

$$\underbrace{\boldsymbol{X}^T\boldsymbol{X}}_{\boldsymbol{H}}\overbrace{\boldsymbol{X}^T\boldsymbol{u}_k}^{k^{th}\text{ left singular vector of }\boldsymbol{H}} = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^T\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{U}^T\boldsymbol{u}_k = \boldsymbol{V}\boldsymbol{\Sigma}^3\boldsymbol{U}^T\boldsymbol{u}_k$$
$$= \sigma_k^3\boldsymbol{v}_k. \tag{3}$$

This means both optimizations (2) and (3) result in finding the index of the element in $\boldsymbol{v}_k$ with the largest absolute value. This means $m_1 = m_2$. Therefore, selection with SP results in the same selection by solving the following problem as solution of *P1*.

$$\underset{|\mathbb{S}|\leq K}{\text{argmin}}\ \|\boldsymbol{H} - \pi_{\mathbb{S}}(\boldsymbol{H})\|_F^2 \ \blacksquare$$

The SP algorithm performs an iterative selection. In each iteration selection is performed on the residual of data after projection on the null space of previously selected samples. Thus, in each iteration *P1* and *P2* are performed on the residual corresponding to the current iteration and they result in the same index.

Matrix $\boldsymbol{H}$ is equal to the weighted replica of auto-correlation matrix of data, $\boldsymbol{A}^T\boldsymbol{A}$. The weights come from the neighborhood information. For example, if data $i$ and data $j$ are not neighbors, then $h_{ij} = 0$. And if they share $P$ neighbors then $h_{ij} = P\boldsymbol{a}_i^T\boldsymbol{a}_j$. Matrix $\boldsymbol{H}$ is a similarity matrix and any other graph-based similarity matrix is reasonable to substitute $\boldsymbol{H}$. In the main paper, we employ normalized similarity matrix, the definition of which is inspired by Laplacian graph of neighborhood. This choice is a conventional similarity matrix in the context of manifold-based dimension reduction. Moreover, it can be employed easily for graph summarization which is investigated in the main manuscript. The neighborhood and weighting in definition of matrix $\boldsymbol{H}$ is hard, while the normalized similarity matrix based on Gaussian kernel provides a soft neighborhood definition via smooth weighting. Employing the normalized similarity matrix results in Problem (6) in the main paper.

Fig. 1 illustrates the impact of nonlinear modeling on a toy example containing a set of $100\times100$ images where each image is a rotated and resized version of other images (Fig. 1(a)). Since none of the images lie on the linear subspace spanned by the rest of images, the ensemble of these data do not form a linear subspace. Therefore, this dataset is of high rank and the union of linear subspaces is not a proper underlying model for it. The KSP algorithm is implemented using a Gaussian kernel with parameter $\alpha$, i.e., $s_{ij} \triangleq e^{-\alpha\|\boldsymbol{a}_i-\boldsymbol{a}_j\|^2}$. As shown in Fig. 1 (c), the nonlinear selection algorithm has been able to discover the intrinsic structure of data and select data from more distinguished angles than that of Fig. 1 (b) in which the plain SP is applied.

## 2. Supplementary Experiments

Further experiments in this section support experiments of the main paper.

### 2.1. Convergence of SP

Provably convergent version of SP algorithm needs a slight modification in the algorithm which is explained later in this section. However, lots of experiments show that the proposed SP algorithm in the main paper converges in less than $5K$ iterations for selecting $K$ samples. Fig. 2 and Fig. 3 show convergence behavior of SP and KSP for selecting from multi-pie face data set and Cora citation dataset within less than $5K$ iterations. We demonstrated our empirical results on the convergence of SP in this section. However, they do not guarantee that SP is provably convergent. A slight modification of SP can guarantee convergence. At each iteration of SP, a new sample is selected only if the resulted residual error decreases (Alg. 1 (SP), line 7). This way the error is non-increasing. The error is also lower bounded by $\|\boldsymbol{A} - \boldsymbol{A}_K\|_F^2$. These two conditions guarantee that the algorithm converges and quality of the selected subset always improves or remains the same. Alg. 1 describes the provably convergent version of the SP algorithm. In line 7, 8 and 9 we check if the new updated sample provides a better minimizer than the previous sample. The initial selection of SP algorithm can affect the final selected set. However, regardless of initialization, SP converges to approximately the same cost as shown here in Fig. 4. Further, initialization of SP using a deterministic algorithm such as IPM [2] and SMRS makes SP independent of initialization.

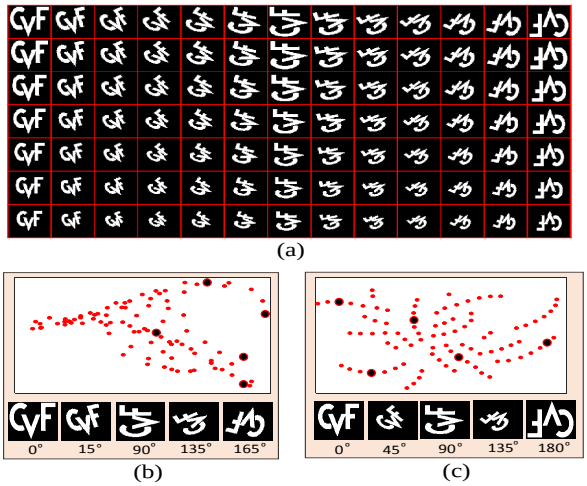


Figure 1: (a) A dataset lies on a two dimensional manifold identified by two parameters, rotation and size. However, the rank of corresponding matrix to this dataset is a large number. (b) Linear embedding using linear PCA and selection using linear SP. (c) nonlinear embedding using tSNE[1] and selection using kernel-SP. Un-selected and selected samples are shown as red and black dots in the embedded space, respectively. The non-linear embedding using a kernel is able to keep the intrinsic structure and non-linear selection provides more diverse samples.

**Algorithm 1** Provably Convergent SP

**Require:** $\boldsymbol{A}$, $P$ and $K$
**Output**: $\boldsymbol{A}_{\mathbb{S}}$
**Initialization:**
$\mathbb{S} \leftarrow$A random subset of $\{1, \ldots, M\}$ with $|\mathbb{S}| = K$
$\{\mathbb{S}_k\}_{k=1}^{K} \leftarrow$ Partition $\mathbb{S}$ into $K$ sets that each one has 1 element.
iter$= 0$
while the stopping criterion is not met
  $k =$mod(iter,K)+1
  $\boldsymbol{U}_{\overline{k}} =$ normalize column$(\boldsymbol{A}_{\mathbb{S}\setminus\mathbb{S}_k})$
  $\boldsymbol{V}_{\overline{k}} = \boldsymbol{A}^T \boldsymbol{U}_{\overline{k}} (\boldsymbol{U}_{\overline{k}}^T \boldsymbol{U}_{\overline{k}})^{-1}$
  $\boldsymbol{E}_{\overline{k}} = \boldsymbol{A} - \boldsymbol{U}_{\overline{k}} \boldsymbol{V}_{\overline{k}}^T$
  $\boldsymbol{u}_k =$ first left singular-vector of $\boldsymbol{E}_{\overline{k}}$
  $\Omega_k \leftarrow$ indices of the most correlated columns of $\boldsymbol{E}_{\overline{k}}$ with $\boldsymbol{u}$
  $\Omega = \Omega_k \bigcup \mathbb{S}_k$
  $\mathbb{S}_k \leftarrow \underset{c \in \Omega}{\text{argmin}} \| \boldsymbol{E}_{\overline{k}} - \boldsymbol{u}\boldsymbol{v}^T \|$ s.t. $\boldsymbol{u} = \tilde{\boldsymbol{a}}_c$
  $\mathbb{S} \leftarrow \bigcup_{k'=1}^{K} \mathbb{S}_{k'}$
  iter=iter+1
end while

## 2.2. GAN on Multi-pie Face Dataset

As it is discussed in the main paper, we select only 9 images from each subject (1800 total subjects), and train the network with the reduced dataset for 300 epochs using the batch size of 36. Fig. 5 shows the generated images of a subject in the testing set, using the trained network on the reduced dataset, as well as using the complete dataset. The network trained on samples selected by KSP (fifth row) is able to generate more realistic images, with fewer artifacts,

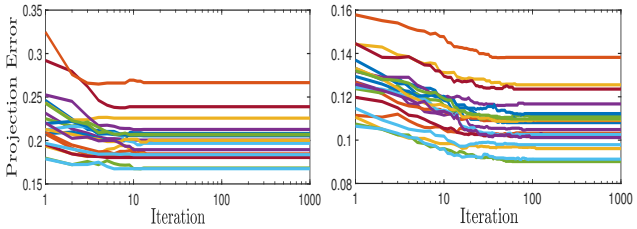


Figure 2: Selecting 5 and 20 representatives from the first 20 classes of Multi-pie dataset. Each class has 520 samples and the error trajectory of each single implementation is depicted in order to show that SP algorithm converges for each independent selection. (Left) Projection error for selecting 5 samples versus iterations. (Right) Projection error for selecting 20 samples versus iterations. Typically, SP selects $K$ representatives in $5K$ iterations.

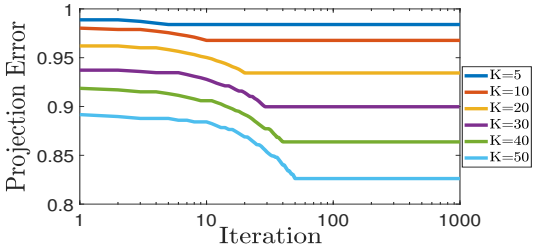


Figure 3: Selecting different number of nodes from Cora dataset which is a graph-based dataset. SP on the similarity matrix of this graph converges in only $K$ iterations which is the minimum number of iterations for updating $K$ selected nodes.

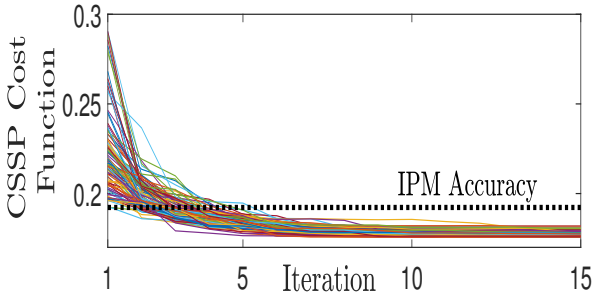


Figure 4: CSSP cost function of selecting $K = 5$ out of 520 samples using SP with 100 random Init. as the first iteration vs. the IPM algorithm, which is deterministic. Interestingly, the accuracy of IPM is comparable with SP using only $K$ iterations with a rough random initialization. However, SP continues iterations.

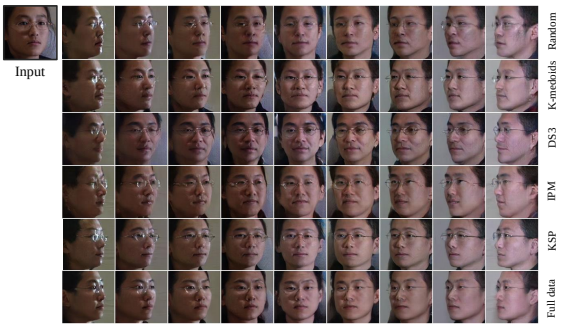


Figure 5: Multi-view face generation results for a sample subject in testing set using CR-GAN [3]. The network is trained on a selected subset of training set (9 images per subject) using random selection (first row), K-medoids (second row), DS3 [4] (third row), IPM (fourth row), and our proposed KSP algorithm. The sixth row shows the results generated by the network trained on all the data (360 images per subject). KSP generates closest results to the complete dataset. In the main paper, a quantitative measure is studied for comparing the generated images and the ground truth from different viwes.

compared to other selection methods (rows 1-4). The parameter of KSP is set as $1e-4$ for constructing the similarity matrix.

## 2.3. Graph Summarization

In Section 4.3 of the paper we presented one of the important applications of KSP algorithm i.e., graph summarization. Here in Fig 6 we compare the central vertex selection and community detection capability of KSP with other state-of-the-art algorithms provided in table 2 for the Powerlaw Cluster graph [5].

## 2.4. Open-set Identification

It is worth noting that in some contexts, open-set is defined as the set containing both known and unknown classes. In this paper, *we have assumed that open-set is only used for the unknown classes* and the known classes at the time of training are called the closed-set.

Here, we provide a discussion on how to select the

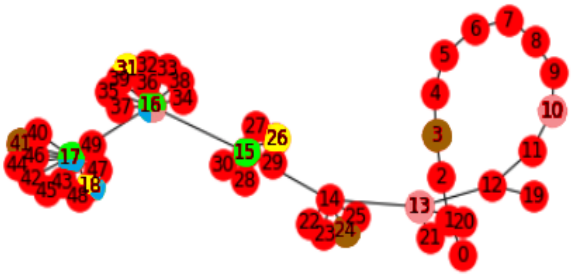

Figure 6: We apply KSP and other algorithms as in table 2, to choose three of the main vertices from another graph, i.e., Powerlaw Cluster graph for which the quantitative results were provided in table 2. The nodes selected by different methods are: GIGA, MP and FW select ●, IS selects ●, VS selects ●, DS3 selects ●, and KSP and FFS select ●. As is evident, KSP and FFS are the only ones that are able to detect the clusters and their corresponding vertices.

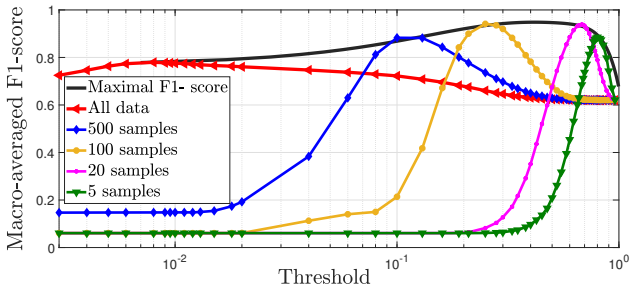


Figure 7: F1-score vs. threshold for different number of selected representatives (Accuracy-Sensitivity Trade-off)

threshold in the open-set identification experiment setup. First, a network is trained on the MNIST training data. Next, the validation data consisting of data from both the known and unknown classes is used to find the threshold as in algorithm 3 in the main text.

At the time of test, a pre-determined threshold is required for deciding on test samples. Our proposed method works based on accessing a set of error values by splitting them and deciding on the threshold. Using one test sample at a time does not lead to a set of error values for splitting at a time. Therefore, one can simply assign the threshold to be a value slightly larger than maximum of error values relating to projecting training samples on selected representatives from each class. Alternatively, if the learning framework is allowed to access validation data, the threshold can be achieved by clustering error values in the balanced validation data into two groups with two centroids, and then taking their average (1:1 sample ratio for Omniglot and MNIST in our case).

Fig. 7 contains the macro-averaged F1-score vs. threshold for different selected representatives using SP data selection. Fine-tuning the open-set identifier by selecting best representatives enhances the accuracy significantly as observed in Fig. 7. As the number of representatives decreases, the performance sensitivity to the threshold adjustment increases which means there is a trade-off between accuracy using selection-based scheme and the stability of

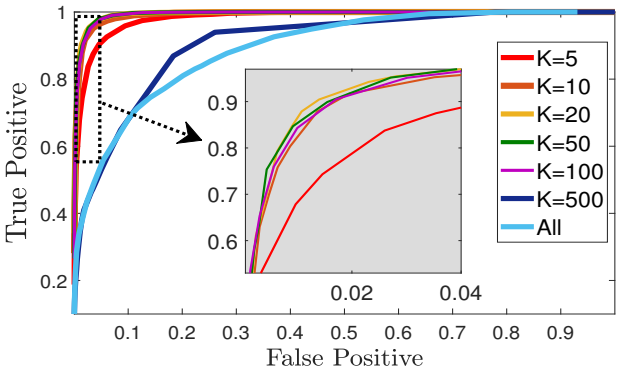


Figure 8: ROC of the proposed selection-based open-set identification employing KSP. The parameter of KSP for constructing the similarity matrix is set to 0.6.

performance w.r.t the designed threshold range. Fig. 7 also shows that between 50-100 samples from each training class (each containing about 6000) leads to optimal F1-score.

In Fig. 8, the receiver operating characteristic (ROC) of area under the curve (AUC) is plotted for the KSP method in the open-set identification. Different number of selected representatives in the proposed SOSIS algorithm (Alg. 3 in the main text) are considered. Sweeping through the threshold range, the ROC-AUC is achieved for SOSIS algorithm with each desired number of selected samples. As observed and magnified in Fig. 8, the best ROC-AUC performance (higher in plot) is achieved for about $20 - 50$ number of selected representatives.

## References


[1] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[2] Alireza Zaeemzadeh, Mohsen Joneidi, Nazanin Rahnavard, and Mubarak Shah. Iterative Projection and Matching: Finding Structure-Preserving Representatives and Its Application to Computer Vision. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5414–5423, 2019.

[3] Yu Tian, Xi Peng, Long Zhao, Shaoting Zhang, and Dimitris N. Metaxas. CR-GAN: Learning Complete Representations for Multi-view Generation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pages 942–948, California, 7 2018. International Joint Conferences on Artificial Intelligence Organization.

[4] Ehsan Elhamifar, Guillermo Sapiro, and S. Shankar Sastry. Dissimilarity based sparse subset selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2182–2197, 2016.

[5] William Aiello, Fan Chung Graham, and Linyuan Lu. A random graph model for power law graphs. *Experimental Mathematics*, 10(1):53–66, 2001.