Exploiting Sparsity to Improve the Accuracy of Nyström-based Large-scale Spectral Clustering

Mahesh Mohan
George Washington University
Email: mahesh_mohan@gwu.edu

Claire Monteleoni
George Washington University
Email: cmontel@gwu.edu

***Abstract*—The Nyström method is a matrix approximation technique that has shown great promise in speeding up spectral clustering. However, when the input matrix is sparse, we show that the traditional Nyström method requires a prohibitively large number of samples to obtain a good approximation. We propose a novel sampling approach to select the landmark points used to compute the Nyström approximation. We show that the proposed sampling approach obeys the same error bound as in Bouneffouf and Birol (2015). To control sample complexity, we propose a selective densification step based on breadth-first traversal. We show that the proposed densification does not change the optimal clustering. Results on real world datasets show that by combining the proposed sampling and densification schemes, we can obtain better accuracy compared to other techniques used for the Nyström method while using significantly fewer samples.**

I. INTRODUCTION

Spectral clustering is a popular technique for clustering [3] based on the eigenvectors of the normalized graph Laplacian. Low rank approximations are used to scale up these methods by simplifying the computation of eigenvectors and eigenvalues. Several iterative methods to compute the low rank approximation like [4], [5] involve the use of the entire matrix making them infeasible for large matrices. An alternate approach, the Nyström approximation, has been a standard tool for low rank approximation of symmetric positive semi-definite (SPSD) matrices since its introduction in [6]. In cases where the input matrix has low rank, the Nyström approximation is known to return an exact approximation as shown in [7].

Given an input matrix A, the Nyström method chooses a subset of l columns $C \in R^{n \times l}$, and reconstructs the complete kernel matrix by $\hat{A} \approx CW_k^+C^T$, where W is the principal sub-matrix of A induced by the selected columns and W_k^+ is the pseudo-inverse of its rank-k approximation. Various methods have been proposed in the literature to construct the matrices C and W. These can be broadly divided into three categories: projection based, sampling based and clustering based.

Projection based methods use a data-independent projection matrix to represent the entries of the matrix as points in lower dimensional space. In other words, the matrices C and W are given by $C = AS$ and $W = S'AS$ respectively, where S is the projection matrix. Examples of projection matrices used include Gaussian projections, Subsampled Random Fourier Transforms as described in [8].

There has been recent work in improving the efficiency of approximation in the case of spectral clustering by applying clustering-based techniques to columns of normalized graph Laplacians. [9] and [10] used the k-means algorithm (KS), to select k centroids as landmark points. These landmark points are used to compute the Nyström approximation. However, the results in [11] show that both methods perform poorly for non-convex clusters.

The incremental sampling (IS) algorithm proposed in [12], first randomly samples two points from a dataset, to compute a similarity matrix between the sampled points and the remaining points. The algorithm picks the point with the smallest variance, and then iteratively repeats the process until a desired number of landmarks is reached. However, as shown in [13], in higher dimensions the variance of the Euclidean distance tends to zero. In such cases IS may pick inappropriate landmark points and perform similarly to uniform sampling. In order to address this behavior of points in high dimensional spaces, [13] proposed minimum similarity sampling (SS). However, it is outperformed by IS on low dimensional data. The approach proposed in this paper relies on the norms of the columns as opposed to the distance between them. Hence it performs equally well for both low and high dimensional data.

[11] introduced Minimum Sum of Squared Similarities (MSSS), which approximately maximizes the determinant of the reduced similarity matrix that represents the mutual similarities between sampled data points. However all these methods can become computationally expensive as the size of the matrix grows. In contrast to these methods, the proposed algorithm relies on the sparsity of the input matrix to return a good approximation efficiently even in high dimensional spaces.

Both projection based approaches and clustering based approaches require the computation of the entire matrix. For sampling based methods, the matrices C and W are given by $C = AS$ and $W = S'AS$ respectively, where S is the sampling matrix, i.e. C is a subset of the columns of A and W is its induced principal sub-matrix. Hence, the entire input matrix does not need to be computed; a sub-matrix C suffices. Examples include uniform sampling [14], column norm sampling [15], leverage score sampling [8]. Since W is constructed without using a data-independent projection matrix, sampling based approaches offer a better generalization bound compared to projection based approaches

when the input matrix has a large eigen gap [16].

The contributions of this paper are,

1. When the input matrix is sparse, we show that techniques that rely on the distances between columns fail to perform well. Additionally, the Nyström method based on uniform sampling requires a prohibitively large number of samples to obtain a good approximation.
2. We propose a novel sampling technique to select the initial landmark points and show that for sparse matrices, it has an error bound that is equal to MSSS [11] and requires far fewer computations.
3. To control sample complexity, we propose a technique for selective densification based on breadth first traversal.
4. We show that the proposed densification does not change the optimal clustering when the input matrix is block diagonal.

II. Observations For Sparse Matrices

In this section we outline two major issues that are faced by sampling based approaches when the matrix is extremely sparse. First we show that distances between columns are not useful for selecting landmark points when the matrix is sparse. Second, we show that the expected number of samples required for uniform sampling can be $O(n)$ when the matrix is sparse. Let the average degree of a node be d_{avg} and the maximum degree of any node be d_{max}. For sparse matrices, we assume that d_{max}^2 is extremely small compared to the number of vertices n.

A. Distances Between Sparse Columns

Observation 1. *Suppose we are given a sparse, symmetric positive semi definite matrix A of size $n \times n$. For any pair of columns of A, say x, y, with probability $\geq 1 - d_{max}^2/n$,*

$$\|x - y\|^2 = \|x\|^2 + \|y\|^2$$

Proof. For a column x, we call column y *favorable* if there exists a row i such that $x_i \neq 0$ and $y_i \neq 0$. $\|x - y\|^2 \neq \|x\|^2 + \|y\|^2$ only if y is favorable with respect to x. For y to be favorable with respect to x, x and y must have at least one common neighbor. We make two observations: x has at most d_{max} neighbors and each neighbor can have at most d_{max} neighbors. Thus, there are at most d_{max}^2 ways to choose a favorable y. This implies,

$$\begin{aligned} Pr[\text{ choosing a favorable } y] &\leq d_{max}^2/n \\ Pr[\|x - y\|^2 \neq \|x\|^2 + \|y\|^2] &\leq d_{max}^2/n \end{aligned}$$

Since $Pr\left[\|x - y\|^2 = \|x\|^2 + \|y\|^2\right] = 1 - Pr[\|x - y\|^2 \neq \|x\|^2 + \|y\|^2]$, the statement in Observation 1 now follows. □

Consequence: In [9], the k-means algorithm is used to select the landmark points used to compute the Nyström approximation. The k-means algorithm has three major steps: initialization, cluster assignment and updating the center. In the initialization step, candidate centers are chosen by sampling k columns uniformly at random. Among these centers $\{c_1, c_2 ..., c_k\}$, let c_{min} be the center with the smallest column norm.

In the cluster assignment step, all the columns are assigned to their closest center. For a column x, let its closest center be represented as c_x, i.e.

$$c_x = \arg\min_c \|x - c\|^2$$

From Observation 1, this can be rewritten as,

$$c_x = \arg\min_c \|x\|^2 + \|c\|^2$$

This shows, points will be assigned to c_{min} with high probability. This leads to bad landmarks being chosen, resulting in a poor approximation. Note, other techniques are available to select the initial candidate centers. However, these techniques are computationally intensive. Thus the k-means based landmark selection is not useful when the matrix is sparse.

B. Sample Complexity

Let C be the sub-matrix created by sampling l columns of the input matrix A. A node in the graph is said to be $covered$ if its corresponding *row* in C has norm greater than zero. All nodes that are not covered are mapped to the origin. This results in a bad clustering and has to be avoided. Consider a matrix A with n rows and columns.

Observation 2. *When uniform sampling is used to select the landmark points, to ensure any node x is covered, the number of samples l, satisfies,*

$$E[l] \geq n/(d_{max} + 1)$$

Proof. x is covered only if x or one of its neighbors is sampled. Since there are at most d_{max} neighbors, the probability of choosing x or one of its neighbors is at most $(d_{max}+1)/n$. Thus the number of samples needed l is a geometric random variable with the probability of success being $(d_{max} + 1)/n$. Thus we have $E[l] \geq n/(d_{max} + 1)$ which is significantly large if $d_{max} \ll n$. □

In this paper, we address this issue by selectively densifying the sampled matrix C as described in the next section. Selective densification increases both d_{avg} and d_{max}. This enables sampling based approaches to cover the vertices of the graph with a smaller number of samples. Since the computational complexity of the Nyström approach scales cubically with the number of samples, using fewer samples results in a significant speedup for large datasets.

III. Proposed Approach

The proposed approach is outlined in Algorithm 1. It can be divided into three phases. Let $0 \leq \alpha \leq 1$ be user-specified constant. In the first phase, we sample a subset of αl samples as follows: choose a set of αl initial columns uniformly at random. Additional columns are sampled uniformly at random and only the αl columns with the highest column norm are retained. Let us call this subset S_t and the associated sub-matrix C (i.e. $C = A(:, S_t)$). In this paper, we use an

iterative approach to selecting the columns, but it can be easily parallelized to obtain further improvement in efficiency.

In the second phase, C is densified as follows: For each column $c \in S_t$, at step i, the function $can\_be\_reached()$ returns the set of nodes that can be reached from c in i steps. This is represented in the Algorithm as nn and the predecessor for each node in nn is returned in $prev$. For each node $p \in nn$ (let q be its corresponding node in $prev$) and column s, if $C(p,s) = 0$, the new densified value is given as, $C(p,s) = C(q,s) * A(q,p)$.

Finally, in the third phase, a pass is made over all the columns to cover the columns that are not covered even after the densification step.

Once the columns have been selected, the eigenvectors are approximated using the approach outlined in [2]. We use the procedure described in [2] due to its ability to handle matrices that are not positive semi-definite. In case the input matrix is positive semi-definite, we can use the simpler procedure described in [1].

Algorithm 1 Proposed Modified Spectral Clustering

Input : Matrix A, the required number of clusters k, number of samples l, fraction of points α, maximum number of densification steps max_hops
Output : Cluster assignment $\mathcal{C}$
Procedure:
##Select initial landmark points
$S_0 \leftarrow$ Sample αl columns uniformly at random
$q \leftarrow argmin_{p \in S_0} deg(p)$
$t \leftarrow 0$
while $t < (1-\alpha)l$ **do**
  Sample a row c_t uniformly at random
  if $\|c_t\| > \|q\|$ **then**
    $S_t \leftarrow S_{t-1} - \{q\} \cup \{c_t\}$
    $q \leftarrow argmin_{p \in S_t} \|p\|$
  end if
  $t \leftarrow t + 1$
end while
##Densify selected columns
$C \leftarrow A(S_t, :)$
for each $s \in S_t$ **do**
  for $i < max\_hops$ **do**
    $[nn, prev] \leftarrow can\_be\_reached(s, i)$
    $C(nn, s) \leftarrow C(prev, s) * A(prev, nn)$
  end for
end for
##Compute Approximate eigenvectors
$W \leftarrow C(S_t, :)$
$D_W \leftarrow degree(W)$
$D \leftarrow degree(C)$
$[U, \Lambda] \leftarrow eig(W)$
$Q \leftarrow D^{-1/2} C D_W^{-1/2} U \Lambda^{+}$
$\mathcal{C} \leftarrow discretize(Q)$

IV. ANALYSIS

A. Justification for the initial sampling scheme

We briefly outline the MSSS algorithm to facilitate comparison with the proposed approach. The MSSS algorithm proceeds iteratively by selecting the column that has minimum similarity with the centers chosen so far. In other words, at each iteration, the method chooses the column that is farthest from the landmark points chosen so far.

At the end of m iterations, the proposed approach retains $t = \alpha l$ columns which have the highest column norms.

Lemma 1. *Suppose, at iteration m, columns $x_1, x_2 ... x_t$ have been chosen so far. Suppose, the similarity between columns is given by $sim(x,y) = e^{-0.5 * \|x-y\|^2}$. In the sparse case, with probability $\geq 1 - d_{max}^2/n$*

$$\arg\max_y \|y\|^2 = \arg\min_y \sum sim^2(y, x_i)$$

In other words, selecting the column with the highest column norm is equivalent to selecting the column with the lowest similarity with all the columns chosen so far.

Proof. Since $sim(x,y) = e^{-0.5\|x-y\|^2}$, we can apply Observation 1 to the exponent of the expression and state that, with probability $\geq 1 - d_{max}^2/n$,

$$sim(x,y) = e^{-0.5(\|x\|^2 + \|y\|^2)} = e^{-0.5\|x\|^2} e^{-0.5\|y\|^2}$$

Thus,

$$\begin{aligned} \sum sim^2(y, x_i) &= \sum (e^{-0.5\|y-x_i\|^2})^2 \\ &= \sum (e^{-0.5\|y\|^2} e^{-0.5\|x_i\|^2})^2 \\ &= (e^{-0.5\|y\|^2})^2 \sum (e^{-0.5\|x_i\|^2})^2 \end{aligned}$$

Hence,

$$\begin{aligned} \arg\min_y \sum sim^2(y, x_i) &= \arg\min_y e^{-0.5\|y\|^2} \\ &= \arg\max_y \|y\|^2 \end{aligned}$$

□

At iteration m, the proposed method retains t columns that have the maximum column norms over the set of columns sampled till iteration m. Thus we see that at each iteration, with high probability, the proposed sampling step selects the column that has the least similarity to the columns chosen so far. Using the following theorem from [11] we can also say this increases the determinant at each step.

Theorem 1. *[11] Suppose columns $X = \{x_1, x_2 ... x_m\}$ have been chosen so far. Let A_X be the Nyström approximation obtained by selecting the columns of X. Then for any pair of columns p, q, if,*

$$\sum sim^2(p, x_i) \leq \sum sim^2(q, x_i)$$

then with high probability,

$$det(A_{X \cup \{p\}}) \geq det(A_{X \cup \{q\}})$$

This result can be used to prove that the Frobenius norm error of proposed sampling scheme has the same upper bound as MSSS. The upper bound on the Frobenius norm error is given in the following theorem.

Theorem 2. *[11] Suppose columns $X = \{x_1, x_2...x_l\}$ have been chosen so far. Let A_k be the optimal rank-k approximation and $\hat{A}_k$ be the rank-k Nyström approximation obtained by using the proposed method.*

$$\|A-\hat{A}_k\| \leq \|A-A_k\|+(l+1)\sum_{i=l+1}^{n}\lambda_i+\gamma\left(1+\sqrt{\frac{\theta d_S}{A_{max}}}\right)^{1/2}$$

Here $d_S = \max_{ij}(A_{ii}+2A_{jj}A_{ij})$, . For more details on the error bound, please refer to Theorem 2 in [11].

If the error bounds are similar, why should the proposed sampling procedure be used? The proposed sampling scheme eliminates the need to compare a candidate column with all columns chosen so far. This improves efficiency especially when a large number of columns need to be sampled. It also allows the selection of columns in parallel making it easier to scale to larger datasets while providing the same level of accuracy.

B. Why Matrix norms are insufficient for spectral clustering

The Nyström approximation has been extensively studied with respect to various matrix error norms, such as the Frobenius norm, trace norm and the spectral norm. In this paper we show that approximation of these norms is not sufficient for producing a good clustering. For simplicity, we focus on the trace norm error. However the examples can be extended to any matrix norm.

1) Not Sufficient: Here we provide an approach to construct two matrices that are extremely similar in terms of their trace norms, but have a significantly different norm cut objective. We begin by describing the norm cuts objective. Suppose, we are given a graph $G = (V, A)$, which is made up of a set of n vertices V. The affinity matrix A is $n \times n$ whose entries represent the similarity between vertices. If V_1, V_2 are subsets of V, let $links(V_1, V_2) = \sum_{i\in V_1, j\in V_2} A_{ij}$.

Furthermore, let $degree(V_i) = links(V_i, V)$. The graph partitioning problem seeks to partition the graph into k disjoint clusters $V_1, ..., V_k$. For a possible clustering $\{V_1..V_k\}$, the norm cuts objective is described as,

$$norm\_cut(G, \{V_1..V_k\}) = \sum_{i=1}^{k}\frac{links(V_i, V\backslash V_i)}{degree(V_i)}$$

Also, let $norm\_cut(A, k) = \min_{V_1,V_2..V_k} norm\_cut(G, \{V_1, ...V_k\})$.

In order to show that a good approximation with respect to the trace norm error is not necessarily a good approximation with respect to the norm cuts ratio, we show that for any value of $\epsilon > 0$, there exists n, k, A and B such that, A and B both represent graphs with n vertices, such that $\|A - B\| \leq (1+\epsilon)\|A\|_1$ and $norm\_cut(A, k) = O(k) * norm\_cut(B, k)$. Note, that the maximum value of $norm\_cut(A, k)$ is k.

Let A and B be the affinity matrices for undirected, unweighted graphs with n vertices and k and $k-1$ equal sized components respectively. Further, we assume that each component is a clique. It can be shown that the normalized symmetric laplacian of A (denoted by $L(A)$) has eigenvalues $n/(n-k)$ with multiplicity $n-k$ and 0 with multiplicity k. Similarly, $L(B)$ has eigenvalues $n/(n-k+1)$ with multiplicity $n-k+1$ and 0 with multiplicity $k-1$. More generally, we can state that

Lemma 2. *[17] Let A, D be the affinity and degree matrix of a graph, with n vertices and k equal sized components, Then the eigenvalues of the normalized symmetric laplacian $L(A) = I - D^{-0.5}AD^{-0.5}$ are $n/(n-k)$ with multiplicity $n-k$ and 0 with multiplicity k.*

We can use Lemma 2 to compare the trace norm of the difference of the two Laplacians, $L(A)$ and $L(B)$, as follows,

$$\begin{aligned}
\|L(A)-L(B)\|_1^2 &= \sum_{i=1}^{n-k}|\lambda_i(A)-\lambda_i(B)| \\
&= \sum_{i=1}^{n-k}\left(\frac{n}{n-k}-\frac{n}{n-k'}\right)+\frac{n}{n-k'} \\
&= (n-k)\left[\frac{n}{n-k}-\frac{n}{n-k'}\right]+\frac{n}{n-k'} \\
&= (n-k)\left[\frac{n}{(n-k)(n-k')}\right]+\frac{n}{n-k'} \\
&= \frac{2n}{n-k+1}
\end{aligned}$$

where, $k' = k-1$. Given a value of ϵ, we can choose appropriate values of n, k such that $\|L(A) - L(B)\|_1^2 \leq (1+\epsilon)\|L(A)\|_1^2$. Now we examine the norm cuts ratio when we try to partition these graphs into exactly k clusters. It is easy to see that the $norm\_cuts(A) = 0$, since the graph has exactly k components. However for B, one of the $k-1$ components will have to be split into two equal parts to minimize the norm cuts ratio. This results in a cut that involves $\frac{n}{k-1}(\frac{n}{k-1}+1)/2$ edges. Thus the norm cuts ratio for B is given as,

$$\begin{aligned}
norm_cut(B) &= k - \frac{\left(n(n+2k-2)/(k+1)^2\right)}{n(n+k-1)/2(k+1)^2} \\
&= k - \frac{n+2k-2}{4(n+k-1)}
\end{aligned}$$

The second term reduces to a constant for a sufficiently large value of n, resulting in a norm cuts ratio of $O(k)$.

2) Not Necessary: To show that preserving matrix norms is not necessary for an approximation to be good with respect to the norm cuts objective, we consider the case of a block diagonal matrix A with k blocks. Let L be the corresponding normalized Laplacian. We are interested in the eigenvectors of L, which are the same as the eigenvectors of L^2, L^3, etc. However $\|L - L^i\|_1$ can be arbitrarily high.

C. Justification for the densification step

Now we proceed to provide approximation guarantees for the proposed densification scheme with respect to the norm cuts objective. Specifically, we state that

Lemma 3. *If the affinity matrix is block diagonal, the clustering induced by the affinity matrix does not change after densification.*

Proof. We use the connection to the weighted kernel k-means objective shown in [18] which showed that spectral clustering using the norm cuts objective is equivalent to weighted kernel k-means. Specifically, given an affinity matrix A and its associated degree matrix D, number of clusters k, minimizing the norm cuts objective was shown to be equivalent to weighted kernel k-means problem with kernel matrix $K = D^{-1}AD^{-1}$ and weight matrix $\mathcal{W} = D$. Thus,

$$w_j = d_j$$

is the weighted degree of vertex j and

$$K_{ij} = A_{ij}/(d_i * d_j)$$

Suppose m_c is the center of cluster π_c. They show that the distance of any point to the center of cluster π_c is given as

$$\|\phi(a_i) - m_c\|^2 = K_{ii} - 2\frac{\sum\limits_{a_j \in \pi_c} w_j K_{ij}}{\sum\limits_{a_j \in \pi_c} w_j} + \frac{\sum\limits_{a_j, a_l \in \pi_c} w_j w_l K_{jl}}{(\sum\limits_{a_j \in \pi_c} w_j)^2}$$

Plugging these values in the expression for $\|\phi(a_i) - m_c\|^2$, we get,

$$\|\phi(a_i) - m_c\|^2 = K_{ii} - 2\frac{\sum\limits_{a_j \in \pi_c} A_{ij}}{d_i \sum\limits_{a_j \in \pi_c} d_j} + \frac{\sum\limits_{a_j, a_l \in \pi_c} A_{jl}}{(\sum\limits_{a_j \in \pi_c} d_j)^2}$$

Since A is block diagonal, no entries of column i are modified except those corresponding to vertices in π_1. Thus we have,

$$\sum_{a_j \in \pi_c} A_{ij} = d_i$$

and

$$\sum_{a_j, a_l \in \pi_c} A_{jl} = \sum_{a_j \in \pi_c} d_j$$

Thus we get,

$$\|\phi(a_i) - m_c\|^2 = K_{ii} - \frac{1}{\sum\limits_{a_j \in \pi_c} d_j}$$

Suppose column i belonging to cluster π_c is densified. Since a_i belongs to π_c, $\|\phi(a_i) - m_c\|^2$ is smaller than the distance to any other center. Let the new affinity matrix after densification be A'. The distance of the point a_i from its center is given by,

$$\|\phi'(a_i) - m_c\|^2 = K'_{ii} - \frac{1}{\sum\limits_{a_j \in \pi_c} d'_j}$$

$$\begin{aligned}
&\|\phi'(a_i) - m_c\|^2 - \|\phi(a_i) - m_c\|^2 \\
&= K'_{ii} - \frac{1}{\sum\limits_{a_j \in \pi_c} d'_j} - K_{ii} + \frac{1}{\sum\limits_{a_j \in \pi_c} d_j} \\
&\leq -\frac{1}{\sum\limits_{a_j \in \pi_c} d'_j} + \frac{1}{\sum\limits_{a_j \in \pi_c} d_j} \quad (1) \\
&\leq -\frac{1}{\sum\limits_{a_j \in \pi_c} d_j} + \frac{1}{\sum\limits_{a_j \in \pi_c} d_j} \quad (2) \\
&\leq 0
\end{aligned}$$

Equation 1 follows from the fact that $K_{ii} \geq K'_{ii}$. Equation 2 follows from the observation that $\sum\limits_{a_j \in \pi_c} d'_j \geq \sum\limits_{a_j \in \pi_c} d_j$. This holds because, the densification only adds positive entries to A.

Thus we see that after densification, the point moves closer to its own center, while its distance to other clusters remains unchanged. This ensures that the clustering remains unchanged.

□

V. Experimental Results

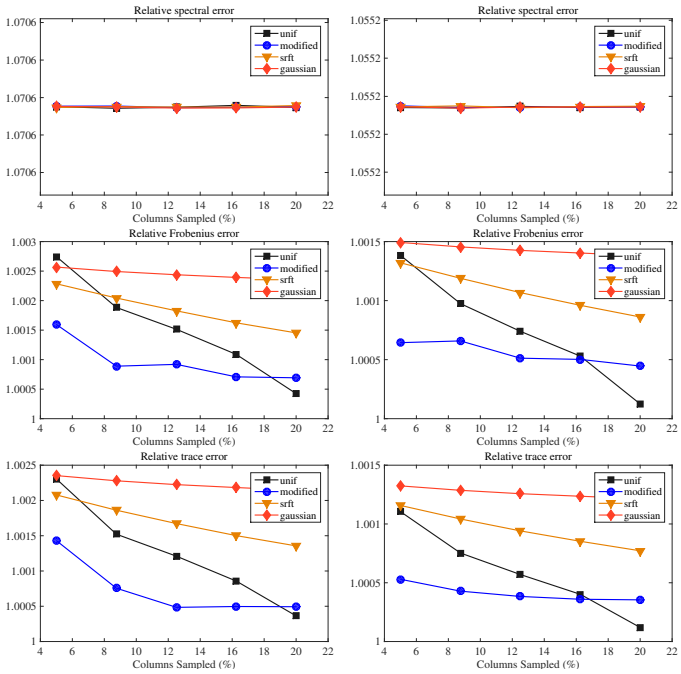
A. Matrix Norm Errors

For the sake of completeness, we present the errors with respect to various matrix norms. For all our experiments, we used the experimental framework in [8]. The following errors were used: $\|A\|_2 = \|Diag(\Sigma)\|_\infty$ denotes the spectral norm of A; $\|A\|_F = \|Diag(\Sigma)\|_2$ denotes the Frobenius norm of A; $\|A\|_* = \|Diag(\Sigma)\|_1$ is the trace norm of A.

The HEP, GR datasets were obtained from [19]. The datasets are extremely sparse in terms of their non-zero entries. In addition, it has been noted that their spectra decays slowly. We restrict the rank of the Laplacian for each dataset to 20. In other words, the low-rank approximation is 'filtered' through a space of rank 20.

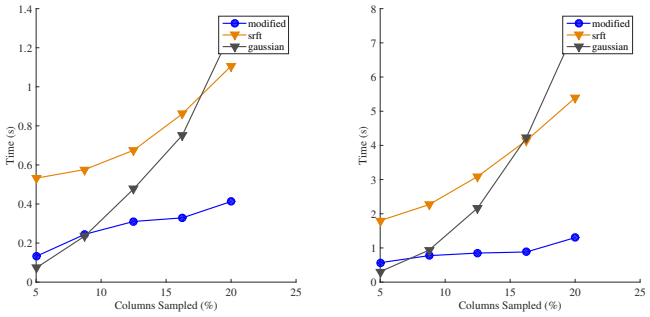
Figure 1 shows a comparison between the various matrix norm errors, for the proposed algorithm (called "modified") and Nyström using uniform sampling (unif), Subsampled Random Fourier Transforms (srft) and Gaussian projections (gauss). The result shows that the proposed method yields a better approximation than srft and gauss for all the errors considered. Even though uniform sampling has lower matrix norm error compared to the other methods, it is important to note that a significantly large number of rows and columns in the resulting approximation had norm zero. This is shown in Figure 3. In contrast, due to the use of the densification and clean up steps, the proposed approach significantly reduces the number of uncovered columns.

Figure 2 shows a comparison of the computation time for the various methods. This shows that as the number of samples being considered increases, the proposed method requires significantly lower time compared to Nyström with srft and gauss making it better suited for large datasets.



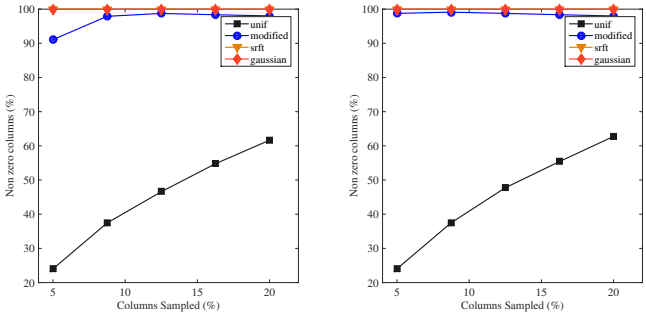
(a) GR (b) HEP

Fig. 1: Comparison of errors for proposed algorithm (modified), Nyström with uniform sampling (unif), Subsampled Random Fourier Transform (srft), Gaussian Projections (gaussians) for different datasets. We restrict the rank of these datasets to 20.



(a) GR (b) HEP

Fig. 2: Comparison of running time: Modified Nyström (modified) v/s Nyström with Subsampled Random Fourier Transform (srft), Gaussian Projections (gauss) for different datasets. We restrict the rank of these datasets to 20. The plots show that the proposed method is computationally better than both gauss and srft.



(a) GR (b) HEP

Fig. 3: Comparison of columns with norm zero: Modified Nyström (modified) v/s Nyström with Subsampled Random Fourier Transform (srft), Gaussian Projections (gauss) for different datasets. The plots show that the proposed method covers all the columns with significantly fewer samples. In contrast, uniform sampling fails to cover all the columns even after sampling 20% of the columns.

Name	Instances	Attributes	Classes
Aggregation (Agg)	788	2	7
D31	3100	2	31
Flame (Fla)	240	2	2
A.K Jain's toy problem (AK)	373	2	2

TABLE I: The synthetic datasets used in our experiments.

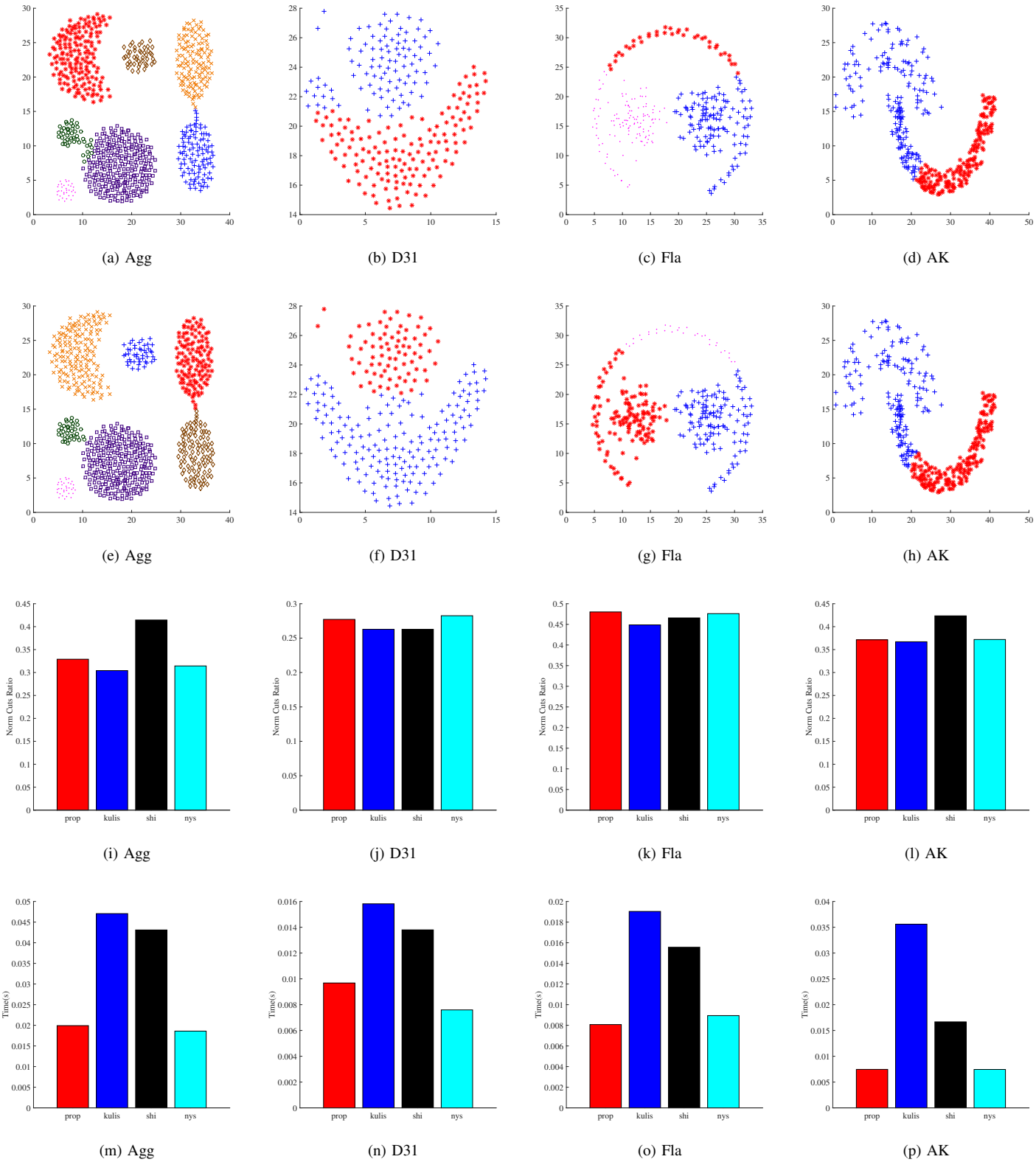
B. *Spectral Clustering*

1) Synthetic Data: We compared the performance of the proposed approach to the results of approaches described in [18] and [20]. Both of these methods use the entire affinity matrix and do not perform any sampling. We evaluated the norm cuts ratio and the computation time on six commonly used synthetic datasets [11], described in Table I, and repeated our evaluations 10 times. The proposed approach sampled 30% of the points. We measured the clustering quality of each algorithm using the average accuracy across different datasets. The results are shown in Figure 4. The proposed approach, the weighted kernel k-means approach in [18] and the spectral method in [20] are denoted in Figure 4 as "proposed", "kulis", "shi" respectively. For all the datasets, the proposed approach results in comparable accuracy while resulting in a significant computational speedup.

2) Image Segmentation: One of the most popular applications of spectral clustering is image segmentation. In this section, we describe results obtained on an image segmentation benchmark [20]. The affinity matrix and the final discretization were computed using the approach of [20]. Since we had access to the function that was used to generate the affinity matrix, we used a simpler densification step. Namely,

$$\begin{aligned} C_{ij} &= similarity(i,j) \text{ if } j = \arg\max_p similarity(i,p) \\ &= 0 \text{ otherwise} \end{aligned}$$

The final segmentation was refined using the connected



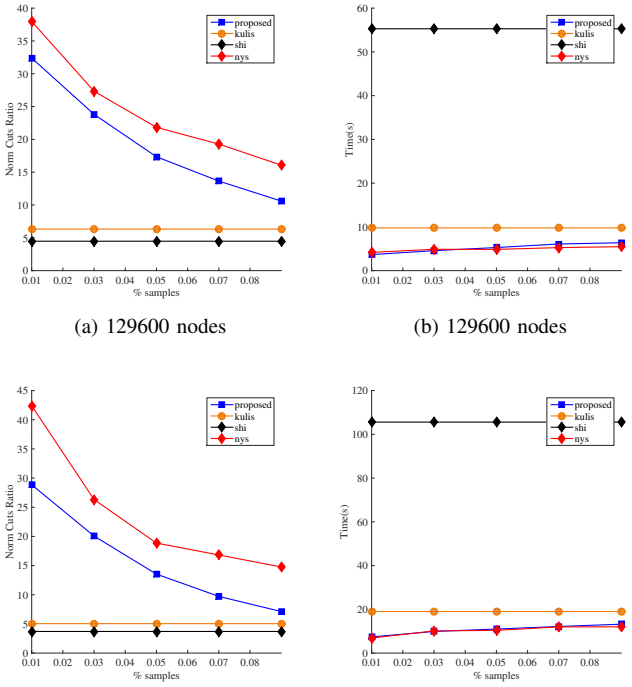
(a) Agg (b) D31 (c) Fla (d) AK

(e) Agg (f) D31 (g) Fla (h) AK

(i) Agg (j) D31 (k) Fla (l) AK

(m) Agg (n) D31 (o) Fla (p) AK

Fig. 4: The first and second rows show the clustering obtained by [18] and the proposed approach respectively on several toy problems. The third and fourth rows show the norm cuts ratio and the time taken for each dataset. It can be seen that the proposed approach yields very similar results while taking comparable amount of time.



(a) 129600 nodes (b) 129600 nodes

(c) 230400 nodes (d) 230400 nodes

Fig. 5: Figures (a) and (c) show the norm cuts ratio, whereas (b), (d) show the time taken by different methods. The proposed method outperforms Nyström with uniform sampling (nys) in terms of norm cuts ratio while taking a similar amount of time. The baseline methods shi [20] and kulis [18] use the entire affinity matrix. Thus they offer lower norm cuts ratio, and have a significantly higher computation time.

components algorithm in [21]. The proposed approach is compared against the approaches in [20] (shi) and [18] (kulis). Experimental results in Figure 5 show that the proposed approach obtains the same accuracy as the Nyström approximation with uniform sampling, while using significantly fewer samples.

VI. CONCLUSION

When the input matrix is sparse, we showed that the traditional Nyström method requires a prohibitively large number of samples to obtain a good approximation. To control sample complexity, we propose a selective densification step based on breadth first traversal to ensure all nodes are covered. We show that the proposed densification does not change the optimal clustering when the input matrix is block diagonal. Results on real world datasets show that the proposed method outperforms other techniques used for the Nyström approximation.

REFERENCES

[1] A. Choromanska, T. Jebara, H. Kim, M. Mohan, and C. Monteleoni, "Fast spectral clustering via the nyström method," in *Algorithmic Learning Theory*. Springer, 2013, pp. 367–381.

[2] M. Li, X.-C. Lian, J. T. Kwok, and B.-L. Lu, "Time and space efficient spectral clustering via column sampling," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 2011, pp. 2297–2304.

[3] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[4] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU Press, 2012, vol. 3.

[5] A. J. Smola and B. Schlkopf, "Sparse greedy matrix approximation for machine learning," in *International Conferene on Machine Learning*. Morgan Kaufmann, 2000, pp. 911–918.

[6] C. Williams and M. Seeger, "Using the nyström method to speed up kernel machines," in *Proceedings of the 14th Annual Conference on Neural Information Processing Systems*, no. EPFL-CONF-161322, 2001, pp. 682–688.

[7] A. Talwalkar and A. Rostamizadeh, "Matrix coherence and the nystrom method," *arXiv preprint arXiv:1004.2008*, 2010.

[8] A. Gittens and M. W. Mahoney, "Revisiting the nystrom method for improved large-scale machine learning," *arXiv preprint arXiv:1303.1849*, 2013.

[9] K. Zhang, I. W. Tsang, and J. T. Kwok, "Improved nyström low-rank approximation and error analysis," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 1232–1239.

[10] H. Shinnou and M. Sasaki, "Spectral clustering for a large data set by reducing the similarity matrix size." in *LREC*, 2008.

[11] D. Bouneffouf and I. Birol, "Sampling with minimum sum of squared similarities for nystrom-based large scale spectral clustering," in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, 2015.

[12] X. Zhang and Q. You, "Clusterability analysis and incremental sampling for nyström extension based spectral clustering," in *Data Mining (ICDM), 2011 IEEE 11th International Conference on*. IEEE, 2011, pp. 942–951.

[13] Z. Zeng, M. Zhu, H. Yu, and H. Ma, "Minimum similarity sampling scheme for nyström based spectral clustering on large scale high-dimensional data," in *Modern Advances in Applied Intelligence*. Springer, 2014, pp. 260–269.

[14] S. Kumar, M. Mohri, and A. Talwalkar, "Sampling methods for the nyström method," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 981–1006, 2012.

[15] A. Frieze, R. Kannan, and S. Vempala, "Fast monte-carlo algorithms for finding low-rank approximations," *Journal of the ACM (JACM)*, vol. 51, no. 6, pp. 1025–1041, 2004.

[16] T. Yang, Y.-F. Li, M. Mahdavi, R. Jin, and Z.-H. Zhou, "Nyström method vs random fourier features: A theoretical and empirical comparison," in *Advances in neural information processing systems*, 2012, pp. 476–484.

[17] S. Butler, "Interlacing for weighted graphs using the normalized laplacian," *Electronic Journal of Linear Algebra*, vol. 16, no. 1, p. 8, 2007.

[18] I. S. Dhillon, Y. Guan, and B. Kulis, *A unified view of kernel k-means, spectral clustering and graph cuts*. Citeseer, 2004.

[19] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, p. 2, 2007.

[20] S. X. Yu and J. Shi, "Multiclass spectral clustering," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 313–319.

[21] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.