A DISTANCE-BASED FORMULATION FOR SAMPLING SIGNALS ON GRAPHS

Ajinkya Jayawant, Antonio Ortega

Department of Electrical Engineering, University of Southern California

ABSTRACT

We consider the problem of sampling signals defined on the nodes of a graph. This problem arises in many contexts where the data is not structured and needs to be reconstructed from a few samples. While other graph signal sampling techniques have been recently developed in the literature, these are based on graph spectral concepts. In contrast, here we develop a method that incorporates distances between graph vertices, and thus can provide additional insights about desirable properties of sampling sets relative to state-of-the-art techniques. We compare the accuracy of our method with two other fast methods in the literature and show that it achieves similar performance.

Index Terms— Graphs, sampling, bandwidth, coherence, distance

1. INTRODUCTION

Graphs with data associated to their nodes (graph signals) and edge weights given by similarities between two data elements are a convenient way to represent and analyze data with irregular relationships between data points [1]. Graphs are useful in a variety of different scenarios such as characterizing the Web [2], semi-supervised learning [3], community detection [4], or traffic analysis [5].

Similar to traditional signal processing, we may want to estimate a graph signal, based on observations on a few nodes of the graph, leading to the problem of sampling graph signals [6, 7]. For this we need to choose a set of vertices, $\mathcal{S}$, called the sampling set, on which we observe the signal values in order to predict the signal values over the other vertices (set $\mathcal{S}^c$). In the presence of noise some sampling sets give better reconstruction than others and the goal of *sampling set selection* is to find the best such sampling set.

Most techniques developed to date for sampling set selection [6, 7] are based on spectral criteria, which require computing multiple eigenvectors of a Laplacian or adjacency matrix. These methods have been shown to be superior to random sampling, in the presence of noise or model mismatches, but they have the disadvantage of not providing insights about how local vertex connectivity affects node selection. Moreover, deploying these algorithms for large scale graphs or in a distributed manner may be challenging due to the eigendecompositions required.

As an alternative, in this paper we propose a vertex-domain sampling approach inspired by an interpretation of spectral proxy method of [7] in terms of vertex distances. This leads to a sampling set selection algorithm where nodes are selected based on their relative distances. We also provide an interpretation of the link between a sampling set that minimizes the reconstruction error and the corresponding distance properties between samples $\mathcal{S}$ and $\mathcal{S}^c$.

2. NOTATION AND SETUP

A graph is defined as the pair $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes or vertices and $\mathcal{E}$ is the set of edges [8]. The set of edges is a subset of the set of unordered pairs of elements of $\mathcal{V}$. The graph signal is a real-valued function defined on the vertices of the graph $f : \mathcal{V} \to \mathbb{R}$.

We index the vertices $v \in \mathcal{V}$ with the set $\{1, \cdots, N\}$ and define w_{ij} as the weight of the edge between i and j. The adjacency matrix of the graph $\mathbf{A}$ has elements $\mathbf{A}_{ij} = w_{ij}$ and $\mathbf{A}_{ii} = 0$. The degree matrix $\mathbf{D}$ of a graph is a diagonal matrix having the elements $\mathbf{D}_{ii} = \sum_j w_{ij}$ along the diagonal. The combinatorial Laplacian for the graph is given by $\mathbf{L} = \mathbf{D} - \mathbf{A}$. When the nonzero weights w_{ij} are all 1 we call the graph unweighted and the graph is weighted otherwise. We assume that there are no self loops and the edge weights are positive. The eigendecomposition of the Laplacian matrix is $\mathbf{L} = \mathbf{U}\Sigma\mathbf{U}^T$ since the Laplacian matrix is symmetric and positive semidefinite. $\mathbf{\Sigma} = \mathrm{diag}(\lambda_1, \cdots, \lambda_N)$, with $\lambda_1 \leq \cdots \leq \lambda_N$ representing the frequencies, while the column vectors of $\mathbf{U}$ provide a frequency representation for graph signals, so that $\mathbf{U}$ is usually called the graph Fourier transform (GFT). The larger the eigenvalue λ_i corresponding to the eigenvector u_i of $\mathbf{L}$ the higher frequency it represents [1]. The sample set $\mathcal{S}$ is defined as a subset of $\mathcal{V}$ where the values of f are known. The problem we consider here is that of finding the set $\mathcal{S}$ such that when we interpolate the values of f on $\mathcal{S}^c$ from measurements on $\mathcal{S}$ the interpolation error is minimum. We will often work with sub-matrices of $\mathbf{U}$ with different frequency or vertex localization. Letting $\mathcal{R}$ be the set $\{1, \cdots, r\}$, where $r = |\mathcal{R}|$, the matrix constructed by selecting first r columns of $\mathbf{U}$ will be denoted by $\mathbf{U}_{V\mathcal{R}}$ or simply $\mathbf{U}_{\mathcal{R}}$ and the matrix constructed by further selecting rows of $\mathbf{U}_{\mathcal{R}}$ indexed by $\mathcal{S}$ will be written as $\mathbf{U}_{\mathcal{S}\mathcal{R}}$.

3. MOTIVATION

For the signal to be determined by a finite number of samples we require the signal to be bandlimited or approximately bandlimited, i.e., all or most of the energy of the signal is contained in the first r columns of $\mathbf{U}$ and thus $\mathbf{f} \approx \mathbf{U}_{\mathcal{R}}\mathbf{a}$ for some $\mathcal{R}$ and $\mathbf{a}$. In this case, selection of the best sampling set is related to choosing the sampling set $\mathcal{S}$ that maximizes the minimum singular value of the matrix $\mathbf{U}_{\mathcal{S}\mathcal{R}}$ [6, 7]. Maximizing the minimum nonzero singular value of the submatrix of a matrix is a subset selection problem known to be NP-hard [9]. Thus, most graph signal sampling literature uses greedy techniques [6, 10, 11] after computing $\mathbf{U}$. Other recent research [12, 13] considers the problem of minimizing a scalar function of the error covariance matrix and solve a relaxed version of the optimization problem, but even these need the information of the eigenvector matrix U. However computing $\mathbf{U}$ itself involves computing the eigenvalue decomposition of $\mathbf{L}$ and so has $O(N^3)$ complexity. As alternatives, a number of sampling approaches [7, 14, 15] do not require explicit computation of $\mathbf{U}$ as a starting point, using instead local estimates of frequency [7] or random sampling [16, 15].

As an intuition for our work we note that because signals being sampled are smooth (low frequency), selecting a sampling set with nodes that are close to each other in the graph is unlikely to be an efficient strategy. Instead a set selection that chooses nodes that are further apart is likely to lead to more robust sampling. This intuition can be justified more formally by considering the lowest singular value of $\mathbf{U}_{\mathcal{SR}}$ and its relation with distances between nodes in $\mathcal{S}$. We are interested in maximizing the lowest non-zero singular value $\sigma_{\min}$ of the matrix $\mathbf{U}_{\mathcal{SR}}$. Note first that for any nonzero singular value of $\mathbf{U}_{\mathcal{SR}}$ we have:

$$\begin{aligned}\sigma_i(\mathbf{U}_{\mathcal{SR}}) &= \sqrt{\lambda_i(\mathbf{U}_{\mathcal{SR}}\mathbf{U}_{\mathcal{SR}}^T)} \\ &= \sqrt{\lambda_i(\mathbf{U}_{\mathcal{SR}}\mathbf{U}_{\mathcal{R}}^T\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{SR}}^T)} \\ &= \sigma_i(\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{SR}}^T), \end{aligned} \tag{1}$$

where the first and the last equality follow for any matrix $\mathbf{A}$ because the non-zero singular values of $\mathbf{A}$ are square roots of the non-zero eigenvalues of $\mathbf{A}\mathbf{A}^T$ [17]. The matrix $\mathbf{U}_{\mathcal{R}}^T\mathbf{U}_{\mathcal{R}}$ is the identity matrix $\mathbf{I}_{\mathcal{R}}$, and $\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{SR}}^T$ can be written as:

$$\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{SR}}^T = \mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{R}}^T\mathbf{I}_{\mathcal{S}} \tag{2}$$

whose columns are the low pass filtered versions of signals $\boldsymbol{\delta_v}$ (on r lowest eigenvectors) where $v \in \mathcal{S}$ and $\mathbf{I}_{\mathcal{S}}$ is the selection of columns of the identity matrix corresponding to the sampling set $\mathcal{S}$. A large $\sigma_{\min}(\mathbf{U}_{\mathcal{SR}}\mathbf{U}_{\mathcal{R}}^T)$ occurs when then the columns of the matrix are close to orthogonal and their magnitudes are high. A typical column $\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{R}}^T\boldsymbol{\delta}_v$ is the translation [18] of the low pass kernel at the vertex v:

$$\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{R}}^T\boldsymbol{\delta}_v = \sum_{l=1}^{r}\mathbf{u}_l(v)\mathbf{u}_l = \sum_{l=1}^{N}\tilde{\boldsymbol{\delta}}_v(l)\tilde{g}(\lambda_l)\mathbf{u}_l \tag{3}$$

where $\hat{g}(\lambda_l) = 1, \lambda_l \leq \lambda_r$ and 0 otherwise. Since the columns are the low pass kernel translated to the vertices in $\mathcal{S}$, each column decays as a function of the hop distance (Theorem 1, [18]). The relation between translated kernels and decay as a function of distance has also been discussed in [19]. Note that because translations of the low pass kernel affect the reconstruction accuracy directly by appearing as columns in (2) we should be able to improve accuracy beyond what would be achievable with random sampling by considering distances between the sampled vertices.

To start with, having close to orthogonal columns of $\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{SR}}^T$ leads to a larger lowest singular value of $\mathbf{U}_{\mathcal{SR}}$, which based on (3) can be achieved by selecting vertices v (to include in $\mathcal{S}$) that are as far away from each other as possible, so that their corresponding columns in $\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{SR}}^T$ have small magnitudes of mutual inner products. This suggests that distance between nodes can be used as part of a graph signal sampling strategy. We next analyze the algorithm in [7] to interpret it from the perspective of distances between nodes. This will allow us to define a greedy set selection algorithm based on distances that does not require explicit eigendecomposition. This algorithm will combine elements of deterministic approach compared to random sampling and which does not require the eigenpair computations as in [7].

4. CUTOFF FREQUENCY MINIMIZATION AND DISTANCES

The paper [7] starts off by defining a uniqueness set $\mathcal{S}$ for a frequency ω if all signals with frequency less than ω can be uniquely reconstructed by knowing their values on $\mathcal{S}$. The cutoff frequency $\mathcal{S}$ is defined as the minimum bandwidth of a signal which is zero over the set $\mathcal{S}$. Any set $\mathcal{S}$ is considered better with respect to sampling if it's cutoff frequency is more. The premise of the algorithm is maximizing the cutoff frequency of sets by greedily selecting next vertices. However since the cutoff frequency estimation requires the knowledge of eigendecomposition of L, the paper instead uses a parameter k which can be increased to get a better approximation of the cutoff frequency. Since we are interested in studying the behavior of the algorithm we take the perfect case of k going to infinity($k \to \infty$). The smoothest signal on $\mathcal{S}^c$ is then simply computed by using U as follows.

Consider a set $\mathcal{S}$ with s samples and assume it is a uniqueness set, so that columns of $\mathbf{U}_{\mathcal{SR}}$ are independent for $r \leq s$. Assuming that the $s+1$-th column vector of $\mathbf{U}$, $\mathbf{u}_{s+1}$ column vector is not all zero over $\mathcal{S}$, then $\mathbf{U}_{\mathcal{SR}}$, with $r = s+1$ columns will be a dependent set of columns. Thus a minimum bandwidth signal on $L_2(\mathcal{S}^c)$ can be given as some linear combination of the first $s+1$ columns of $\mathbf{U}$. The lowest bandwidth signal $\mathbf{h}$ with all zeros on $\mathcal{S}$ is given as follows.

$$\mathbf{h} = \mathbf{U}_{\mathcal{R}}\boldsymbol{\alpha}, \quad \mathbf{U}_{\mathcal{SR}}\boldsymbol{\alpha} = \mathbf{0} \tag{4}$$

The second equation above determines the coefficients $\boldsymbol{\alpha}$ using the condition that the signal is zero over $\mathcal{S}$.

We are interested in finding the vertex v that maximizes $|\mathbf{h}(v)|$. Let $\mathbf{d}_v = \mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{R}}^T\boldsymbol{\delta}_v$ denote the low pass filtered delta signal at vertex v. Then,

$$\begin{aligned}\mathbf{h}(v) = \boldsymbol{\delta}_v^T\mathbf{h} &= \boldsymbol{\delta}_v^T\mathbf{U}_{\mathcal{R}}\boldsymbol{\alpha} = \boldsymbol{\delta}_v^T\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{R}}^T\mathbf{U}_{\mathcal{R}}\boldsymbol{\alpha} \\ &= (\boldsymbol{\delta}_v^T\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{R}}^T)(\mathbf{U}_{\mathcal{R}}\boldsymbol{\alpha}) = \langle\mathbf{d}_v, \mathbf{h}\rangle \end{aligned} \tag{5}$$

So $|\mathbf{h}(v)|$ is maximized when $|\langle\mathbf{d}_v, \mathbf{h}\rangle|$ is maximized. Now since $\mathbf{h}(v) = 0$ for $v \in \mathcal{S}$, given (5) $\mathbf{d}_v, v \in \mathcal{S}$ are perpendicular to $\mathbf{h}$.

$$\langle\mathbf{d}_v, \mathbf{h}\rangle = 0, v \in \mathcal{S} \tag{6}$$

We call the space spanned by those columns $\mathcal{D}_{\mathcal{S}}$ defined as $\mathcal{D}_{\mathcal{S}} = \{\text{span}(\mathbf{d}_v)|v \in \mathcal{S}\}$ From the equation (6), we know that $\mathbf{h}$ is orthogonal to the subspace $\mathcal{D}_{\mathcal{S}}$. Now we show that the subspace $\text{span}(\mathcal{D}_{\mathcal{S}} \cup \mathbf{h})$ is the space of bandlimited functions with bandwidth r. From (1) we know that $\text{rank}(\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{SR}}^T) = \text{rank}(\mathbf{U}_{\mathcal{SR}}^T)$. Since $\mathcal{S}$ is a uniqueness set by construction, it has rank s. Thus the columns $\mathbf{d}_v, v \in \mathcal{S}$ of $\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{SR}}^T$ are linearly independent. From the orthogonality relation of (6) we have that the subspace $\text{span}(\mathcal{D}_{\mathcal{S}} \cup \mathbf{h})$ has dimension $s+1 = r$ and so any vector with bandwidth of λ_r or less can be represented as their linear combination of $\mathbf{d}_u, u \in \mathcal{S}$ and $\mathbf{h}$. So we can write $\mathbf{d}_v$ as the following linear combination: $\mathbf{d}_v = \sum_{u\in\mathcal{S}} c_u\mathbf{d}_u + c_h\mathbf{h}$. and we have the following decomposition of $\mathbf{d}_v$: $\mathbf{d}_v = \mathbf{P}_{\mathcal{D}}\mathbf{d}_v + \langle\mathbf{d}_v, \mathbf{h}\rangle\mathbf{h}$, where $\mathbf{h}$ has unit magnitude and $\mathbf{P}_{\mathcal{D}}$ is the projection matrix onto the subspace $\mathcal{D}_{\mathcal{S}}$ (note that $\mathbf{P}_{\mathcal{D}}$ is a function of $\mathcal{S}$ but we do not make this explicit in the notation for simplicity.) The energy of $\mathbf{d}_v$ can be given in terms of its projection on the two orthogonal subspaces: $\|\mathbf{d}_v\|^2 = \|\mathbf{P}_{\mathcal{D}}\mathbf{d}_v\|^2 + \langle\mathbf{d}_v, \mathbf{h}\rangle^2$. The signal magnitude $|\mathbf{h}(v)|$ can be maximized using (5).

$$\arg\max_v \langle\mathbf{d}_v, \mathbf{h}\rangle^2 = \arg\max_v(\|\mathbf{d}_v\|^2 - \|\mathbf{P}_{\mathcal{D}}\mathbf{d}_v\|^2) \tag{7}$$

Note that $\|\mathbf{d}_v\|^2$ is the squared local graph coherence [14] $\left\|\mathbf{U}_{\mathcal{R}}^T\boldsymbol{\delta}_v\right\|^2$ of that vertex.

$$\|\mathbf{d}_v\|^2 = \left\|\mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{R}}^T\boldsymbol{\delta}_v\right\|^2 = \left\|\mathbf{U}_{\mathcal{R}}^T\boldsymbol{\delta}_v\right\|^2 \tag{8}$$

To maximize the expression in (7) we would like to select nodes that have i) large squared local graph coherence with respect to r

frequencies (the first term in (7), which is a property of each node and does not depend on $\mathcal{S}$) and ii) small squared magnitude of projection onto the subspace $\mathcal{D}_{\mathcal{S}}$ (which does depend on $\mathcal{S}$). In order to understand this second goal, note that a smooth kernel translated to a vertex v is expected to decay as a function of distance from v. The subspace $\mathcal{D}_{\mathcal{S}}$ is a linear combination of smooth kernels translated to all vertices in $\mathcal{S}$ and therefore any signal in $\mathcal{D}_{\mathcal{S}}$ will have a small magnitude at vertices "far away" from $\mathcal{S}$. The signal $\mathbf{d}_v$ is also concentrated around v. Therefore a vertex $v \in \mathcal{S}^c$ whose distance to the vertices $\mathcal{S}$ is large is likely to lead to a small projection $\|\mathbf{P}_{\mathcal{D}}\mathbf{d}_v\|^2$ and thus would tend to maximize (7). Our proposed algorithm combines both conditions, by first identifying vertices that are at a sufficiently large distance from already chosen vertices in $\mathcal{S}$ and then selecting among those the one having largest value of $\|\mathbf{d}_v\|^2$.

More specifically, assume we have a set $\mathcal{S}$ of vertices selected already. Then, the next vertex v^* will be selected by first finding a set of nodes that are sufficiently far from $\mathcal{S}$, $\mathcal{V}_d(\mathcal{S})$ defined as follows

$$\mathcal{V}_d(\mathcal{S}) = \{v \in \mathcal{S}^c | d(\mathcal{S}, v) > \Delta . \max_u d(\mathcal{S}, u)\},$$

where Δ is between 0 and 1, $d(\mathcal{S}, v) = \min_{u \in \mathcal{S}} d(u, v)$ and d is the geodesic distance on the graph. The distance between two adjacent vertices i, j is given by $d(i, j) = 1/w(i, j)$. Then, the second step is to select the node with largest $\|\mathbf{d}_v\|^2$ in $\mathcal{V}_d(\mathcal{S})$:

$$v* = \arg \max_{v \in \mathcal{V}_d(\mathcal{S})} \left\| \mathbf{U}_{\mathcal{R}} \mathbf{U}_{\mathcal{R}}^T \boldsymbol{\delta}_{\boldsymbol{v}} \right\|^2 .$$

The parameter Δ is used to control how many nodes can be included in $\mathcal{V}_d(\mathcal{S})$. With a Δ small, more nodes will be considered, at the cost of increased complexity.

5. IMPLEMENTATION

5.1. Signal Models and sampling

The graph toolboxes [20, 21] are used to construct the following types of graphs having 1000 nodes.

KNN: Nodes are selected uniformly at random on a 2 dimensional square patch. Each node is then connected with it's 10-nearest neighbours via an undirected edge weighted as $e^{-d^2(i,j)/2}$ with $d(i, j)$ being the Euclidean distance between the pairs of points i and j. Since we do this operation for each node, we may have nodes with more than 10 neighbours.

Scale free graphs: A Barabási Albert model with 6 initial nodes is chosen.

Community model: An unweighted graph with 10 communities is constructed using the function `gsp_community(1000,10)` in the GSP toolbox [20].

Erdős Rényi model: In this model, the probability of selecting an edge between two nodes is taken to be 0.01.

USPS dataset [22]: Graphs with 1000 vertices are constructed by taking 100 samples from each class. 10 such graphs are constructed. Construction of the graphs follows the same procedure as in [7]. Samples are chosen and reconstruction is done followed by one vs all classification.

With a perfectly bandlimited signal most sampling schemes perform comparably in terms of reconstruction error. However, in practice signals are rarely perfectly bandlimited and noise-free. Therefore it is necessary to compare the performance of the sampling methods on non-ideal signals. We consider two models, the first signal model we consider is bandlimited with additive noise and the second one has energy outside expected bandwidth, similar to that in [7].

Model 1: The signal $\mathbf{x}$ is bandlimited with added noise $\mathbf{n}$. The resulting signal is $\mathbf{f} = \mathbf{U}_{\mathcal{R}}\tilde{\mathbf{x}} + \mathbf{n}$ with the frequency components of $\mathbf{x}$ and noise being random variables distributed as multivariate normal distributions: $\tilde{\mathbf{x}} \sim \mathcal{N}(\mathbf{0}, 0.02\mathbf{I}_R)$, $\mathbf{n} \sim \mathcal{N}(\mathbf{0}, 5 \times 10^{-5}\mathbf{I}_N)$.

Model 2: The signal $\mathbf{x}$ is not bandlimited with a decaying out of bandwidth spectrum. The resulting signal is $\mathbf{f} = \mathbf{U}\tilde{\mathbf{x}}$. $\tilde{\mathbf{x}}(i) = \tilde{\mathbf{x}}_n(i)e^{-\beta(\lambda_i - \lambda_{\mathcal{R}})^+/(\lambda_N - \lambda_1)}$ with $\tilde{\mathbf{x}}_n \sim \mathcal{N}(\mathbf{0}, 0.02\mathbf{I}_N)$ and $(\lambda_i - \lambda_{\mathcal{R}})^+ = \max(\lambda_i - \lambda_{\mathcal{R}}, 0)$. β is chosen as the numerical solution of the equation $\sum_{i>R} e^{-2\beta(\lambda_i - \lambda_R)/(\lambda_N - \lambda_1)} = 2.5$.

Note that in both the signal models the desirable signal has expected power 1 whereas the undesirable signal has power 0.05. The reconstruction errors $\left\|\hat{\mathbf{f}} - \mathbf{x}\right\|$ and $\left\|\hat{\mathbf{f}} - \mathbf{f}\right\|$ representing the errors in reconstruction with respect to the observed signal and the underlying bandlimited signal, respectively, where $\hat{\mathbf{f}}$ is the reconstructed signal.

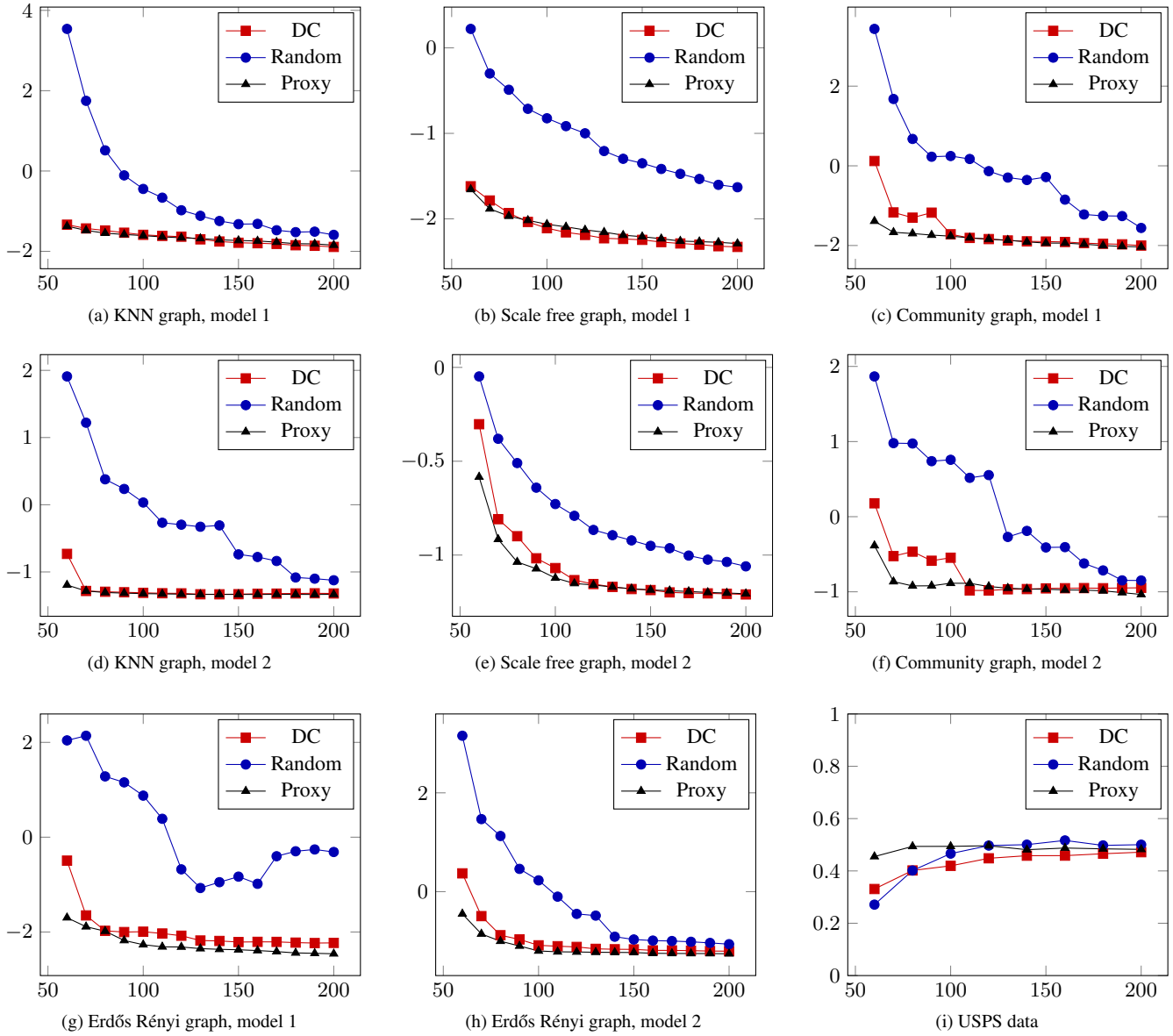
Acquiring samples of a signal usually has a cost associated to it so the algorithms which provide better reconstruction with smaller number of samples are preferable. We compare our method with the methods in [14] and [7]. Let us call these methods Distance-Coherence (*DC*), *Random* and *Proxy* respectively. *DC* and *Proxy* both return unique samples but *Random* does weighted random sampling with replacement. As a result the samples it returns are not always unique. But in order to compare the sampling methods on a fair footing we only care about the number of samples the method returns, irrespective of whether they are unique or not.

5.2. Reconstruction

The sampled signal is given by $f_{\mathcal{S}}$ and the nonzero frequencies of the signal are given by $\tilde{\mathbf{f}}_{\mathcal{R}} = \mathbf{U}_{\mathcal{R}}^T\mathbf{f}$. For the purpose of reconstruction we assume that we know the Graph Fourier basis and we can do the best possible reconstruction given the samples. The solution is given by the least squares solution to $\left\|\mathbf{U}_{\mathcal{S}\mathcal{R}}\tilde{\mathbf{f}}_{\mathcal{R}} - \mathbf{f}_{\mathcal{S}}\right\|_2$. $\mathbf{p}_{\mathcal{S}}$ denotes the sampling probabilities corresponding as in *Random*. We define the matrix $\mathbf{P} = \mathrm{diag}(\mathbf{p}(\mathcal{S}))$. For *Random* the reconstruction is given by its standard decoder which minimizes $\left\|\mathbf{P}^{-1/2}(\mathbf{U}_{\mathcal{S}\mathcal{R}}\tilde{\mathbf{f}}_{\mathcal{R}} - \mathbf{f}_{\mathcal{S}})\right\|_2$. For both the minimization problems the ideal solutions are given by the least squares solutions. However the matrices $\mathbf{U}_{\mathcal{S}\mathcal{R}}$ and $\mathbf{P}^{-1/2}\mathbf{U}_{\mathcal{S}\mathcal{R}}$ might not have full column rank so we use Pseudo inverse to get the solution. The solutions of Moore Penrose pseudo inverse and minimum least squares coincide when the matrices in question have full column rank. Additionally, pseudo inverse provides the minimum norm solution in case of multiple solutions to the minimization problem. So the reconstruction for the random samples is done by $\hat{\mathbf{f}} = \mathbf{U}_{\mathcal{R}}(\mathbf{P}^{-1/2}\mathbf{U}_{\mathcal{S}\mathcal{R}})^{\dagger}\mathbf{P}^{-1/2}\mathbf{f}_{\mathcal{S}}$ whereas the reconstruction for the other two methods is done as follows $\hat{\mathbf{f}} = \mathbf{U}_{\mathcal{R}}\mathbf{U}_{\mathcal{S}\mathcal{R}}^{\dagger}\mathbf{f}_{\mathcal{S}}$.

5.3. Details and Observations

Note the effect of Δ on the next sample in *DC*. For a high value of Δ we only consider vertices that are at the maximum distance from $\mathcal{S}$ and choose the one with maximum local graph coherence. For a low value of Δ we consider most vertices in the graph so the next vertex is selected mainly on the basis of local graph coherence, in this case *DC* is similar to *Random*. We choose $\Delta = 0.9$. For evaluating the local graph coherences, we use the method in [14]. The coherence in *Random* needs $O(\log N)$ random vectors to be filtered. We use $20\log(N)$ random vectors, which is the same amount used



(a) KNN graph, model 1 (b) Scale free graph, model 1 (c) Community graph, model 1

(d) KNN graph, model 2 (e) Scale free graph, model 2 (f) Community graph, model 2

(g) Erdős Rényi graph, model 1 (h) Erdős Rényi graph, model 2 (i) USPS data

Fig. 1: Figures 1a-1h plot $\log_{10}(\text{MSE})$ while figure 1i plots the average classification accuracy. All the plots are plotted against number of samples taken.

[15]. The low pass filter for filtering the random vectors is approximated using a polynomial of order 30. 10 different instances of each type of graph are used for computing the average squared error with different signals for each instance.

In the experiments in Fig. 1 we can see that the random sampling method has error power equal to the signal power at about two times the number of samples as bandwidth and power equal to $(1/10)^{th}$ the signal power at about four times the number of samples as the bandwidth. The *DC* method performs significantly better compared to *Random* and comparably to the *Proxy* method in all the graphs we experimented with. Since the plot for the USPS data denotes classification accuracy and not mean squared errors the difference between the curves is expected to be less. The *DC* method was faster than the *Proxy* method in all the experimented graphs. A more formal and complete complexity analysis is part of our future work.

6. CONCLUSION

Most sampling schemes perform reasonably well when dealing with perfectly bandlimited signal. However in the presence of noise or the signal not being perfectly bandlimited, some schemes perform much better. In the scenario that only a limited number of samples can be chosen we want an algorithm to perform well without requiring expensive operations. The method presented in this paper relies on the intuition of smooth translated kernels and their decay to come up with a vertex based sampling method which is an improvement over the Weighted Random sampling method and providing performance similar to Graph spectral proxies method.

7. REFERENCES

[1] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[2] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, Dandapani Sivakumar, Andrew Tompkins, and Eli Upfal, "The web as a graph," in *Proceedings of the nineteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. ACM, 2000, pp. 1–10.

[3] Xiaojin Zhu, Zoubin Ghahramani, and John D Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proceedings of the 20th International conference on Machine learning (ICML-03)*, 2003, pp. 912–919.

[4] Santo Fortunato, "Community detection in graphs," *Physics reports*, vol. 486, no. 3, pp. 75–174, 2010.

[5] Mark Crovella and Eric Kolaczyk, "Graph wavelets for spatial traffic analysis," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*. IEEE, 2003, vol. 3, pp. 1848–1857.

[6] Siheng Chen, Rohan Varma, Aliaksei Sandryhaila, and Jelena Kovačević, "Discrete signal processing on graphs: Sampling theory," *IEEE Transactions on Signal Processing*, vol. 63, no. 24, pp. 6510–6523, 2015.

[7] Aamir Anis, Akshay Gadde, and Antonio Ortega, "Efficient sampling set selection for bandlimited graph signals using graph spectral proxies," *IEEE Transactions on Signal Processing*, vol. 64, no. 14, pp. 3775–3789, 2015.

[8] Béla Bollobás, *Modern graph theory*, vol. 184, Springer Science & Business Media, 2013.

[9] Ali Çivril and Malik Magdon-Ismail, "On selecting a maximum volume sub-matrix of a matrix and related problems," *Theoretical Computer Science*, vol. 410, no. 47-49, pp. 4801–4811, 2009.

[10] Mikhail Tsitsvero, Sergio Barbarossa, and Paolo Di Lorenzo, "Signals on graphs: Uncertainty principle and sampling," *IEEE Transactions on Signal Processing*, vol. 64, no. 18, pp. 4845–4860, 2016.

[11] Han Shomorony and A Salman Avestimehr, "Sampling large data on graphs," in *Signal and Information Processing (GlobalSIP), 2014 IEEE Global Conference on*. IEEE, 2014, pp. 933–936.

[12] Xuan Xie, Hui Feng, Junlian Jia, and Bo Hu, "Design of sampling set for bandlimited graph signal estimation," *arXiv preprint arXiv:1711.07345*, 2017.

[13] Abolfazl Hashemi, Rasoul Shafipour, Haris Vikalo, and Gonzalo Mateos, "Sampling and reconstruction of graph signals via weak submodularity and semidefinite relaxation," *arXiv preprint arXiv:1711.00142*, 2017.

[14] Gilles Puy, Nicolas Tremblay, Rémi Gribonval, and Pierre Vandergheynst, "Random sampling of bandlimited signals on graphs," *Applied and Computational Harmonic Analysis*, 2016.

[15] Nicolas Tremblay, Pierre-Olivier Amblard, and Simon Barthelme, "Graph sampling with determinantal processes," *arXiv preprint arXiv:1703.01594*, 2017.

[16] Gilles Puy and Patrick Pérez, "Structured sampling and fast reconstruction of smooth graph signals," *arXiv preprint arXiv:1705.02202*, 2017.

[17] Lloyd N Trefethen and David Bau III, *Numerical linear algebra*, vol. 50, Siam, 1997.

[18] David I Shuman, Benjamin Ricaud, and Pierre Vandergheynst, "Vertex-frequency analysis on graphs," *Applied and Computational Harmonic Analysis*, vol. 40, no. 2, pp. 260–291, 2016.

[19] Nathanaël Perraudin, Benjamin Ricaud, David Shuman, and Pierre Vandergheynst, "Global and local uncertainty principles for signals on graphs," *arXiv preprint arXiv:1603.03030*, 2016.

[20] Nathanaël Perraudin, Johan Paratte, David Shuman, Lionel Martin, Vassilis Kalofolias, Pierre Vandergheynst, and David K. Hammond, "GSPBOX: A toolbox for signal processing on graphs," *ArXiv e-prints*, Aug. 2014.

[21] Benjamin Girault, Shrikanth Narayanan, Paulo Gonçalves, Antonio Ortega, and Eric Fleury, "Grasp: A matlab toolbox for graph signal processing," in *42nd IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017)*, 2017.

[22] "Usps handwritten digits," `https://cs.nyu.edu/~roweis/data.html`.