[WORK IN PROGRESS]
Report on Paper "Compressive Spectral Clustering" by Nicolas Tremblay, Gilles Puy, Rémi Gribnoval, Pierre Vandergheynst

Sourabh Antani

Chapter 1

Introduction

This is a paper report on the paper 'Compressive Spectral Clustering' by by Nicolas Tremblay, Gilles Puy, Rémi Gribnoval and Pierre Vandergheynst [1]. All the sections in this report are taken or paraphrased from [1] and not original work. The structure of this report also follows the structure of the paper so that it is easy to follow.

According to the authors, Spectral Clustering has three computational bottlenecks when N and/or k is large: Creation of the similarity matrix W, partial eigendecomposition of L and k-means.

In order to circumvent these, sevaral authors have published various ideas. Following are few examples:

Power method [2], [3]

A careful optimisation of diagonalisation algorithms in the context of SC [4]

Matrix column-subsampling such as in the Nyström method [5]

nSPEC and cSPEC methods [6], or in [7] or [8]

Reduce k-means complexity [9]

Line of work on coresets [10]

Reduction of Graph by successive aggregation of nodes [11] [12]

Compressive clustering circumvents two last bottlenecks using $\mathcal{O}(\log(k))$ randomly fitered signals on the graph to serve as feature vectors instead of eigenvectors and by clustering random subset of $\mathcal{O}(k\log(k))$ nodes using random feature vectors and infering the cluster label of all N nodes. Thus, the complexity

of k-means is reduced from $\mathcal{O}(Nk^2)$ to $\mathcal{O}(k^2 \log^2(k))$

Chapter 2

Background

2.1 Graph Signal Processing

Graph Fourier Matrix

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, W)$ be an undirected graph with $\mathcal{V}$ the set of N nodes, $\mathcal{E}$ the set of edges and W set weighted adjacency matrix of non-negative weights. The normalized Laplacian of $\mathcal{G}$ is given by $L = I - D^{1/2} W D^{-1/2}$. Here I is $N \times N$ identity matrix and D is a diagnoal matrix with $D_{ii} = \sum_{j \neq i} W_{ij}$. L is real, symmetric and positive semi-definite, hence diagnizable as $L = U \Lambda U^T$ with orthonormal set of eigenvectors $u_1, \ldots, u_N$ and eigenvalues $0 = \lambda_1 \leq \cdots \leq \lambda_N \leq 2$. By analogy to continuous Laplacian operator with classical fourier modes as eigenfunctions and squared frequencies as eigenvalues, the eigenvectors are considered Fourier modes and square-roots of eigenvalues are considered frequencies of the Graph.

Graph filtering

The graph Fourier transform $\hat{x}$ of signal x is $\hat{x} = U^T x$. Given a continuous filter function h defined on $[0, 2]$, its associated graph filter operator $H \in \mathbb{R}^{N \times N}$ is defined as $H := h(L) = U h(\Lambda) U^T$ where $h(\Lambda) := diag(h(\lambda_1), \ldots, h(\lambda_N))$. The filtered signal is then Hx. Now, considering an ideal low-pass filter h_{λ_c} defined as below, H_{λ_c} is the graph filter operator associated with h_{λ_c}.

$$h_{\lambda_c} = \begin{cases} 1, & \text{if } \lambda \leq \lambda_c \\ 0, & \text{otherwise} \end{cases}$$

Fast graph filtering

To filter the signal by h without doagonalizing L, we approximate h by a polynomial of degree p $\tilde{h}(\lambda) = \sum_{l=0}^{p} \alpha_l L^l \simeq h(\lambda), \forall \lambda \in [0, 2], \alpha_1, \ldots, \alpha_p \in \mathbb{R}$. Thus $\tilde{H} := \tilde{h}(L) = \sum_{l=0}^{p} \alpha_l L^l \simeq H$. Thus instead of computing dense $\tilde{H}$ we approximate $\tilde{H}x = \sum_{l=0}^{p} \alpha_l L^l x$ using successive matrix-vector products with L.

Thus the computational complexity is $\mathcal{O}(pe)$ where e is number of edges.

Chapter 3

Principles of Compressive Spectral Clustering

3.1 Ideal filtering of random signals

Definition 3.1.1 (Local cumulative coherence)**.** : Given a graph $\mathcal{G}$, the local cumulative coherence of order k at node i is $v_k(i) = ||U_k^T \delta_i|| = \sqrt{\sum_{=1}^{k} U_{ij}^2}$, i.e. ℓ_2-norm of ith row of U

Next, the authors define the diagonal matrix V_k such that $V_k(i,i) = 1/v_k(i)$, assuming $v_k > 0$. Consider matrix $R = (r_1|r_2|\cdots|r_d) \in \mathbb{R}^{N\times d}$ consisting of d random signals r_i whose components are independent, Bernouilli, Gaussian or sparse random variables with mean zero and variance $1/d$. Considering the coherence=normalized filtered version of R, $V_k H_{\lambda_k}, R \in \mathbb{R}^{N\times d}$, and define node i's new feature vector $f_i \in \mathbb{R}^d$ as transposed i-th row of the filtered matrix, i.e. $\tilde{f}_i := (V_k H_{\lambda_k}, R)^T \delta_i$. The following theorem shows that, for a large enough d,

$$\tilde{D_{ij}} := ||\tilde{f}_i - \tilde{f}_j|| = ||(V_k H_{\lambda_k} R)^T (\delta_i - \delta_j)||$$

is a good estimation of D_{ij} with high probability.

Theorem 1. Let $\epsilon \in (0,1]$ and $\beta > 0$ be given. If d larger than

$$\frac{4+2\beta}{\epsilon^2/2 - \epsilon^3/3} \log N,$$

then with probability at least $1 - N^{-\beta}$, we have

$$(1-\epsilon)D_{ij} \leq \tilde{D}_i j \leq (1+\epsilon)D_{ij}, \forall (i,j) \in \{1,\ldots,N\}^2$$

Proof.

$$\tilde{D}_{ij} = ||\tilde{f}_i - \tilde{f}_j|| = ||(V_k H_{\lambda_k} R)^T (\delta_i - \delta_j)|| = ||R^T H_{\lambda_k}^T V_k^T (\delta_i - \delta_j)|| = ||R^T U_k U_k^T V_k^T (\delta_i - \delta_j)||$$

$$= ||R^T U_k Y_k^T (\delta_i - \delta_j)|| = ||R^T U_k (f_i - f_j)|| \qquad [\because H_{\lambda_k} = U_k U_k^T, Y_k = V_k U_k, Y_k^T \delta_i = f_i]$$

Johnson-Linderstrauss Lemma essentially states that if we wish to construct a d-dimentional vector created from from k-dimensional vector by applying a linear transformation by a $d \times k$ matrix whose columns are randomly selected vectors from gaussian distribution, the probability of existance of such a matrix R^T is greater than 0 , i.e. it is always possible to find such a matrix. whose columns are elements from Gaussian distributions. [Achiloptas, 2003] gives a lower bound of $1 - N^{-\beta}$ on probability of such existance if d is largern than the quantity menioned above.

Hence

$$(1-\epsilon)||U_k(f_i - f_j)|| \leq \tilde{D}_i j \leq (1+\epsilon)||U_k(f_i - f_j)||$$

Since U_k has orthonormal columns, left multiplication by U_k does not change the norm, $||U_k(f_i - f_j)|| = ||(f_i - f_j)|| = D_{ij}$. This proves the required result. □

3.2 Downsampling and interpolation

Let $c_j \in \mathbb{R}^N, j = 1, \ldots, k$ be the indicator vectors of clusters C_j. The authors propose to estimate c_j by running k-means on small subset of n feature vectors only using 1) low-dimentional model that captures the regularity of c_j, 2) make sure enough information is preserved after sampling, 3) algorithem that rapidly and accuraely estimates the vectors c_j.

3.2.1 The Low-Dimentional Model

For a simple regular graph (graph with nodes of same degree) with k disconnected clusters, one can see that the indicator vectors c_i form a set of k orthogonal eigenvectors of L with eigenvalue 0. Thus, al the indicator vectors live in $span(U_k)$. For general graph the authors assume that the indicator vectors live close to $span(U_k)$, i.e. there is a slight perturbation. The perturbation therory argument made in [13], can be applied here and we can say that, previous result applies for general graph. In graph signal processing words, one can say that c_j is approximately k-bandlimited. i.e. the first k graph Fourier coefficients bear most of its energy.

3.2.2 Sampling and Interpolation

Let the subset of feature vectors selected by drawing n indices be $\Omega = \{\omega_1, \ldots, \omega_n\}$. Running k-means on the subset of features $\{\tilde{f}_{\omega_1}, \ldots, \tilde{f}_{omega_n}\}$, thus yields a clustering of n sampled nodes into k clusters. Denote, by $c_j^r \in \mathbb{R}^n$, the low-dimentional indicator vectors. Our goal is to recover c_j from c_j^r.

Assuming that spectral clustering algorithm is able to correctly identify clusters $c_1, \ldots, c_k \in \mathbb{R}^N$, results in [ramasamyandmadhow; tremblay et al, 2016] shows that k-means is also able to identify the clusters using feature vectors $\tilde{f}_1, \ldots, \tilde{f}_N$, since Theorem 1 showed that the distance between all pairs of feature vectors. Then k-means should be able to correctly cluster the n sampled nodes, provided that each cluster has been sufficiently sampled.

Ideally, since we simply selected n nodes, $c_j^r = Mc_j$ where M is matrix containing sampling selection. Then according to [Puy 2015]

$$\min_{x \in \mathbb{R}^N} ||Mx - c_j^r||_2^2 + \gamma x^T g(L) x$$

is a faithful estimation of c_j as long as c_j is close to $span(U_k)$ and M satisfies restricted isometry (discussed later). Here $\gamma > 0$ is a regularization parameter and g is polynomial function.

3.2.3 How many features to sample?

Definition 3.2.1 (Global cumulative coherence)**.** of order k of graph $\mathcal{G}$ is $\nu_k = \sqrt{N} \cdot \max_{1 \leq i \leq N}\{v_k(i)\}$. It is shown by [puy 2015] that $\nu_k \in [\sqrt{k}, \sqrt{N}]$

Theorem 2. Let M be a random sampliing matrix. For any $\delta, \epsilon \in (0, 1)$,

$$(1 - \delta)||x||_2^2 \leq \frac{N}{n}||Mx||_2^2 \leq (1 + \delta)||x||_2^2 \forall x \in span(U_k)$$

(this is called restricted isometry property) with probability at least $1 - \epsilon$ provided

$$n \geq \frac{6}{\delta^2}\nu_k^2 log\left(\frac{k}{\epsilon}\right)$$

Proof. The proof for this theorem is simply application of Johnson-Linterstrauss lemma and noting that columns of M are orthogonal and scaling by N/n provides the normalization needed with the fact that norm is invariant under product with unitary matrix. □

The above theorem suggests that sampling $\mathcal{O}(\nu_k^2 \log(k))$ is sufficient.

Chapter 4

CSC in practice

4.1 The CSC Algorithm

Input: Laplacian Matrix L, number of clusters k, parameters (typically set to) $n = 2k \log k, d = 4 \log n, p = 50, \gamma = 10^{-3}$.

- Estimate L's k-th eigenvalue λ_k as in Section 4.3
- Compute the polynomial approximation $\tilde{h_{\lambda_k}}$ of order p of the ideal low-pass filter j_{λ_k}
- Generate d random Gaussian signals of mean 0 and variance $1/d$ $R = (r_1|r_2|\cdots|r_d) \in \mathbb{R}^{N \times d}$
- Filter R with $\tilde{H}_{\lambda_k} = h_{\lambda_k}(L)$ and define, for each node i, its feature vector $\tilde{f}_i \in \mathbb{R}^d$

$$\tilde{f}_i = \left[\left(\tilde{H}_{\lambda_k} R\right)^T \delta_i\right] \Big/ \left|\left|\left(\tilde{H}_{\lambda_k} R\right)^T \delta_i\right|\right|$$

- Generate a random sampling matrix $M \in \mathbb{R}^{n \times N}$ and keep only n feature vectors: $(\tilde{f}_{\omega_1}|\cdots|\tilde{f}_{\omega_n})^T = M(\tilde{f}_1|\cdots|\tilde{f}_n)^T$
- Run k-means on the reduced dataset with the euclidian distance $\tilde{D}_{ij}^r = ||\tilde{f}_{\omega_i} - \tilde{f}_{\omega_j}||$ to obtain k reduced indicator vectors $c_j^r \in \mathbb{R}^n$, one for each cluster
- Interpolate each reduced indicator vector c_j^r with the optimisation problem stated in section above, to obtain the vectors $\tilde{c}_j^* \in \mathbb{R}^N$

NOTE 1: $\tilde{c}_j^*$ is not binary and quatifies how much the node i belongs to cluster j. This can be treated as fuzzy partitioning or as a probability score. the node i is assigned to cluster j for which $\tilde{c}_j^*(i)/||\tilde{c}_j^*||$ is maximal.

NOTE 2: In step 4, the normalization as applied may not be intuitive. The feature matrix required is $V_k \tilde{H}_{lambda_k} R$. This requirs knowledge of λ_k and $v_k(i)$, which are not known in practice. Authors estimate k^{th} eigenvalue λ_k in section 4.3. To estimate $v_k(i)$ one can use the results of section 4 of [Puy 2015] showing $v_k(i) \approx ||U_k^T \delta_i|| \approx ||(H_{\lambda_k} R)^T \delta_i||$. Thus, practical way is to compute $\tilde{H}_{\lambda_k} R$ and normalize its rows to unit length as in step 4.

4.2 Non-ideal filtering of random signals

In this section, the authors study the effect of the error of polynomial approximation ($\tilde{h_{\lambda_k}}$) of the filter h_{λ_k} on the spectral distance estimation and by applying k-means on a reduced set of features. Let $MY_k \in \mathbb{R}^{n \times k}$ be the ideal reduced feature matrix. The distances we want to measure are, $D_{ij}^r := ||f_{\omega_i} - f_{\omega_j}|| = ||Y_k^T M^T (\delta_i^r - \delta_j^r)||$ where $\{\delta_i^r\}$ are the Direacs in n dimensions.

Let $R \in \mathbb{R}^{N \times d}$ be constructed as in Section 3.1. Its filtered, normalized, reduced version is $MV_k \tilde{H}_{\lambda_k} \in \mathbb{R}^{n \times d}$. Thus, the new filtered, normalized and reduced feature vector associated to node ω_i is $\tilde{f}_{omega_i} =$

$(MV_k\tilde{H}_{\lambda_k}R)^T\delta_i^r$, and the distance between two such features is

$$\tilde{D}_{ij}^r := ||R^T\tilde{H}_{\lambda_k}^T V_k^T M^T(\delta_i^r - \delta_j^r)||$$

Approximation Error: Denote by $e(\lambda)$, the approximation error of ideal low-pass filter: $\forall\lambda \in [0,2], e(\lambda) := \tilde{h}_{\lambda_k}(\lambda) - h_{\lambda_k}(\lambda)$. Thus $\tilde{h}_{\lambda_k}(L) = \tilde{H}_{\lambda_k}(L) = h_{\lambda_k}(\lambda) + e(L)$. Let the error be modelled using two parameters $e_1 := \sup_{\lambda\in\lambda_1,\ldots,\lambda_k}|e(\lambda)|$ and $e_2 := \sup_{\lambda\in\lambda_{k+1},\ldots,\lambda_N}|e(\lambda)|$.

Note that non-ideal filter is not possible to guarantee approximation of cases where ideal distance is $D_{ij}^r = 0$ since the sampling would break the equality of the features. Hence the authors introduce a tolerence (or resolution) parameter D_{min}^r as the maximum allowable ideal distance that does not need to be approximated exactly.

Theorem 3. Genreal norm conservation theroem. Let $D_{min}^r \in (0, \sqrt{2}]$ be a chosen resolution parameter. For any $\delta \in (0,1], \beta > 0$ if d is larger than $\frac{16(2+\beta)}{\delta^2-\delta^3/3}\log n$, then $\forall(i,j) \in \{1,\ldots,n\}^2$,

$$\begin{cases} (1-\delta) \le \tilde{D}_{ij}^r \le (1+\delta)D_{ij}^r, & D_{ij}^r \ge D_{min}^r \\ \tilde{D}_{ij}^r < (1+\delta)D_{ij}^r, & D_{ij}^r < D_{min}^r \end{cases}$$

with probability at least $1 - 2n^{-\beta}$ provided that

$$\sqrt{|e_1^2 - e_2^2|} + \frac{\sqrt{2}e_2}{D_{min}^r \min_r\{v_k(i)\}} \le \frac{\delta}{2+\delta}$$

Proof. TO BE FILLED IN. □

Consequence: All distance smaller (or larger) than chosen resolution parameter D_{min}^R are correctly estimated with relative error δ. Moreover, to keep error δ fixed, lower telerence D_{min}^r would mean lower e_1 & e_2, which means higher order polynomial is required for approximation of ideal filter h_{λ_k}. Thus increase in computation is the result.

4.3 Polynomial approximation of λ_k

To simply the analysis, the authors use $e_m = max(e_1, e_2)$ as maximal error such that theorem 4.1 still applies with last inquality replaced by

$$\frac{\sqrt{2}e_m}{D_{min}^r \min_r\{v_k(i)\}} \le \frac{\delta}{2+\delta}$$

. The authors further suggest use of Jackson-Chebychev polynomials, which adds a damping multipliers to Chebychev polynomials and alleviate Gibbs ossillations around cutoff frequency λ_k. Experimentally, $p = 50$ yields good results.

Estimating λ_k. For fast filtering step, the λ_k is the parameter required for low pass filter h_{λ_k}. The authors suggest use of eigencount techniques [Napoli 2013], based on low-pass filtering with cut-off frequency λ of random signals, one obtains estimation of number of enclosed eigenvalues in $[0, \lambda]$. Starting with $\lambda = 2$ and proceeding by dichotomy on λ, one stops the algorithm as soon as number of enclosed eigenvalues equals k. $2\log N$ random signals with Jackson-Chebychev polynomial approximation of ideal low-pass filters are used for each λ.

4.4 Complexity

The fast filtering of graph signal costs $\mathcal{O}(p\#\mathcal{E})$, where $\#\mathcal{E}$ is number of edges. Hence Step 1 costs $\mathcal{O}(p\#\mathcal{E}\log n)$. Step 4 costs $\mathcal{O}(p\#\mathcal{E}\log n)$. Step 7 costs, for each of the k iterations, one fast filtering operation, i.e. total of $\mathcal{O}(p\#\mathcal{E}k)$. k-means would cost $\mathcal{O}(kn\log n)$ since there are n clusters and vectors are in $\log n$ dimensions. Finally, setting $n = \mathcal{O}(k\log k)$, CSC's complexity simplifies to $\mathcal{O}(k^2\log^2 k + pN(\log N + k))$.

On the otherhand, Spectral Clustering has k-means complexity of $\mathcal{O}(Nk^2)$, and k eigenvector calculation complexity of $\mathcal{O}(k^3 + NK^2)$ (cost of ARPACK). This suggests that CSC is faster than SC for large N and/or k.

Chapter 5

Experiments & Conclusion

At the end the authors provide the experimental results on Stochastic Block Model and Amazon co-purchasing graph to show that the CSC algorithm is able to successfully extract the clusters in synthetically generated and real world data. They do mention that in stochastic block model, the ratio of probabilities of edge between two nodes based on them being in same/different clusters does affect the performance of algorithm. This is a known factor that determines if the clusters are detectable or not [14].

Bibliography

[1] Nicolas Tremblay, Gilles Puy, Remi Gribonval, and Pierre Vandergheynst. Compressive spectral clustering, arXiv:1602.02018, 2016.

[2] Christos Boutsidis, Prabhanjan Kambadur, and Alex Gittens. Spectral clustering via the power method - provably. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 40–48, Lille, France, 07–09 Jul 2015. PMLR.

[3] Frank Lin and William W. Cohen. Power iteration clustering. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10, pages 655–662, 2010.

[4] Tie-Yan Liu, Huai-Yuan Yang, Xin Zheng, Tao Qin, and Wei-Ying Ma. Fast large-scale spectral clustering by sequential shrinkage optimization. In *Proceedings of the 29th European Conference on IR Research*, ECIR'07, pages 319–330, Berlin, Heidelberg, 2007. Springer-Verlag.

[5] C. Fowlkes, S. Belongie, F. Chung, and J. Malik. Spectral grouping using the nystrom method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):214–225, 2004.

[6] Liang Wang, Christopher Leckie, Kotagiri Ramamohanarao, and James Bezdek. Approximate spectral clustering. In Thanaruk Theeramunkong, Boonserm Kijsirikul, Nick Cercone, and Tu-Bao Ho, editors, *Advances in Knowledge Discovery and Data Mining*, pages 134–146, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[7] Xinlei Chen and Deng Cai. Large scale spectral clustering with landmark-based representation. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, AAAI'11, pages 313–318. AAAI Press, 2011.

[8] Tomoya Sakai and Atsushi Imiya. Fast spectral clustering with random projection and sampling. In Petra Perner, editor, *Machine Learning and Data Mining in Pattern Recognition*, pages 372–384, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[9] Anil K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666, 2010. Award winning papers from the 19th International Conference on Pattern Recognition (ICPR).

[10] Sariel Har-Peled and S. Mazumdar. Coresets for k-means and k-median clustering and their applications. *ArXiv*, abs/1810.12826, 2004.

[11] Inderjit Dhillon, Yuqiang Guan, and Brian Kulis. Weighted graph cuts without eigenvectors a multilevel approach. *IEEE transactions on pattern analysis and machine intelligence*, 29:1944–57, 12 2007.

[12] Maurizio Filippone, Francesco Camastra, Francesco Masulli, and Stefano Rovetta. A survey of kernel and spectral methods for clustering. *Pattern Recogn.*, 41(1):176–190, January 2008.

[13] U. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[14] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka ZdeborovÃ¡. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84(6), Dec 2011.

[15] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, MD, 3rd edition, 1996.

[16] B. N. Parlett. *The Symmetric Eigenvalue Problem*. Number 20 in Classics in Applied Mathematics. SIAM, Philadelphia, PA, 1998.

[17] G. W. Stewart and J. G. Sun. *Matrix perturbation theory*. Academic Press, Boston, MA, 1990.

[18] G. Cheung, E. Magli, Y. Tanaka, and M. K. Ng. Graph spectral image processing. *Proceedings of the IEEE*, 106(5):907–930, 2018.