DNC and FNC Methods for Spectral Clustering

Yana Khassan, Vladimir Omelyusik, Yuliya Tukmacheva
Spectral Economists

Skoltech, Numerical Linear Algebra Course

December 16, 2020

Problem Statement

The Problem

Classic spectral clustering algorithms have high computational complexity of $\mathcal{O}(n^3)$ due to performing eigendecomposition.

- The idea is to pose an equivalent optimization problem which can be solved iteratively.
- **Direct Normalized Cut**: $\mathcal{O}(n^2 c)$ and **Fast Normalized Cut**: $\mathcal{O}(dnm + nmc)$ based on **Balanced k-Means**: $\mathcal{O}(dnc)$.

We compare their performance with classic spectral clustering, multiclass spectral clustering (2003) and deep clustering on synthetic and real data sets.

Brief Discussion of Algorithms

- $X_{n \times d}$ is the data set, based on which we construct an affinity matrix A. The goal is to find the cluster indicator matrix $Y_{n \times c}$.

- Classic algorithms (e.g. Normalized Cut):

$$\min_{Y^T D_A Y = I} \mathrm{Tr}\Big(Y^T L_A Y\Big),$$

where $L_A = D_A - A$.

- Direct Normalized Cut (DNC):
  1. Rewrite the problem in a clever way to obtain

$$\max_{\mathbf{Y} \in \Psi^{n \times c},\, \mathbf{F} = \mathbf{D}_A^{\frac{1}{2}} \mathbf{Y} \left(\mathbf{Y}^T \mathbf{D}_A \mathbf{Y}\right)^{-\frac{1}{2}}} Tr\left(F^T M F\right)$$

  2. Iteratively calculate MF and solve the resulting optimization problem on $\mathrm{Tr}\Big(F^T G\Big)$. The update formulas can be calculated explicitly.

Brief Discussion of Algorithms

- Balanced K-Means (BKM):
  1. The problem is

$$\min_{F,H} \|X - HF^T\|_F^2 + \gamma \|F\|_e,$$

     where $H_{d \times c}$ is the cluster center matrix, $F_{n \times c}$ is the cluster indicator matrix.
  2. Derive exact update formulas and update H, G and F iteratively.
- Fast Normalized Cut (FNC):
  1. Use BKM on X to obtain H. Use H to construct similarity matrix B.
  2. Apply DNC using B.

Data Sets

Synthetic data set: `sklearn blobs` (100×2).

Real data sets:

- `isolet5`: imbalanced cluster sizes (1559×256), 26 classes.
- `segment`: singular affinity matrix (2310×256), 7 classes.
- `glass`: small data set (214×9), 6 classes.
- `MnistData-10`: large data set (6000×784), 10 classes.

We measure quality of clusterization via self-defined accuracy:

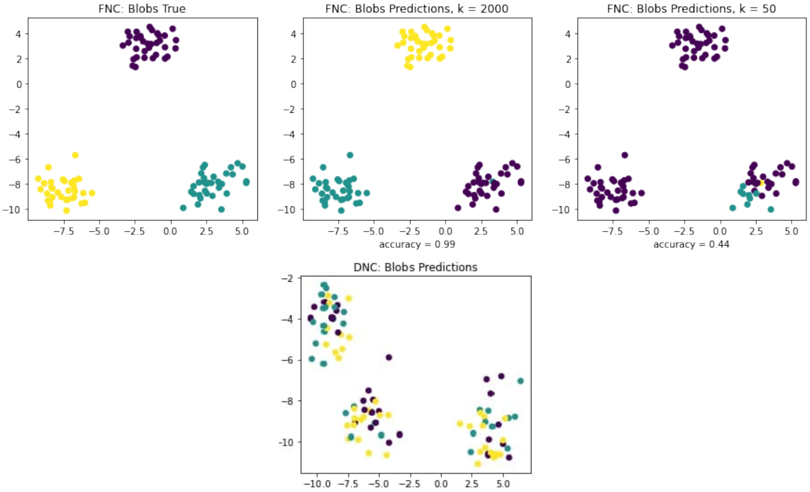
$$\text{accuracy} = \frac{\sum_{x \in X} \mathbb{I}[y_{pred}(x) == y_{true}(x)]}{\text{len}(X)}$$

where $y_{pred}(x) = \sum_{c \in C} \mathbb{I}[x \in c] \times \text{center}(c)$, where $\text{center}(c)$ is the most frequent value of true target in the cluster.

Experiment #1: Blobs

Algorithms	Time (sec)	Accuracy
DNC	0.2	0.44
BKM	0.39	1.0
FNC	2.8	0.99
MSC	0.61	0.3
sklearn Spectral	0.1	1.0

FNC: Number of Neighbours



Experiment #2: `isolet5`

Models	$c = 13$		$c = 26$		$c = 30$	
	Time	Acc	Time	Acc	Time	Acc
DNC	430.73	0.09	673.96	0.13	652.19	0.14
BKM	70.17	0.08	82.86	0.09	84.75	0.1
FNC	152.08	0.04	244.42	0.04	277.42	0.04
MSC	188.80	0.06	166.58	0.11	190.41	0.09
sklearn Spectral	269.78	0.05	520.21	0.048	598.18	0.046
Deep clustering	87.64	0.39	129.41	0.36	140.62	0.38

Experiments #3: `segments`

Models	$c = 3$		$c = 7$		$c = 10$	
	Time	Acc	Time	Acc	Time	Acc
DNC	413.22	0.14	841.27	0.14	1202.03	0.14
BKM	89.84	0.36	84.27	0.52	119.00	0.6
FNC	173.17	0.14	153.71	0.14	241.95	0.14
MSC	351.71	0.15	349.56	0.16	381.71	0.16
sklearn Spectral	1049.44	0.14	848.95	0.14	701.71	0.14
Deep clustering	78.53	0.15	75.82	0.38	114.39	0.42

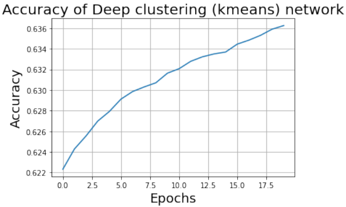
Experiments #4: `glass`

Models	$c=3$		$c=6$		$c=9$	
	Time	Acc	Time	Acc	Time	Acc
DNC	0.81	0.36	1.32	0.39	1.55	0.43
BKM	0.92	0.36	0.84	0.41	0.98	0.41
FNC	2.89	0.36	4.29	0.35	6.7	0.36
MSC	3.02	0.37	2.82	0.35	3.04	0.43
sklearn Spectral	0.13	0.58	0.11	0.485	0.12	0.61
Deep clustering	11.73	0.485	13	0.54	15	0.509

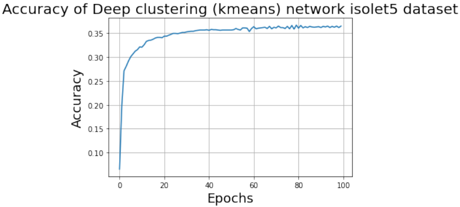
Experiments #5: MNIST

Algorithms	Time (sec)	Accuracy
DNC	4151.21	0.128
BKM	302.43	0.591
FNC	1805.63	0.11
MSC	3668.40	0.12
sklearn Spectral	8.52	0.12
Deep clustering	340	0.636

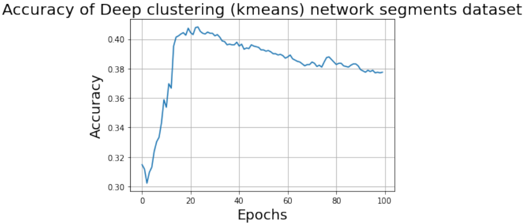
Deep clustering network: accuracy based on number of epochs



(a)



(b)



(c)

Pretraining of Deep Clustering Network

One important way to increase the accuracy of deep clustering network is performing pretraining before making clusterization.

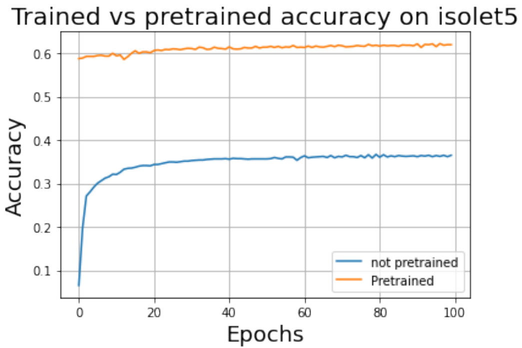
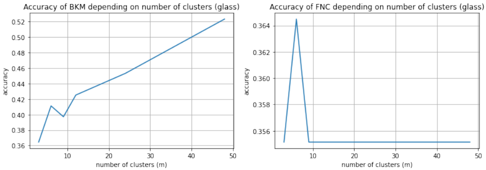
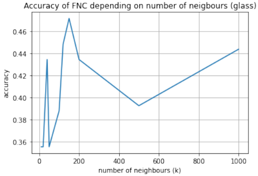


Figure: Trained vs pretrained accuracy on `isolet5`.

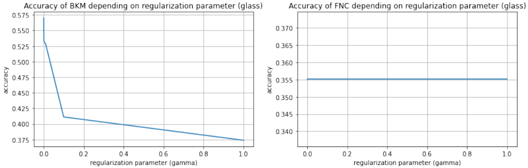
Dependence of FNC on Hyperparameters Choice



(a)



(b)



(c)

Interesting Observations

- Affinity matrix calculation method may have impact on clusterization result [6][7]. The BMK and deep clustering do not depend on similarity matrix and give better accuracy score.
- The pretraining in deep clustering increases the accuracy.
- For clustering the accuracy may not increase with increase in number of epochs.
- The performance of studied algorithms (FNC!) on blobs depends on the choice of hyperparameters. Hence, FNC can work better with optimal choice!
- On the unbalanced `isolet5` data set most of the algorithms work not worse with lower number of clusters, while working with more clusters – decreases accuracy.

Conclusions

- The differences between our results and results from [1] may be explained by variations in (a) method of construction of the similarity matrix; (b) approach to measure accuracy; (c) initialization of matrix Y, especially on unbalanced cluster sizes.

- The most powerful methods are deep clustering and BKM $\Rightarrow$ opens new paths for experiments and quality improvement.

- FNC outperformed most of the algorithms on the majority of data sets in [1] $\Rightarrow$ the optimal choice of hyperparameters matters!

- The proposed methods still beat the standard approaches in both time and accuracy.

Thank you for your attention!

References

1. Chen, Xi., Hong, Wei., Nie, F., He, D., Yang, M., Huang, J.Zh.. Spectral Clustering of Large-scale Data by Directly Solving Normalized Cut. 2018. KDD.
2. Yu, S. X., Shi., J.. Multiclass Spectral Clustering. 2003. 9th IEEE International Conference on Computer Vision.
3. Yang, Xu.,Deng, Ch., Zheng, F., Yan, J., Liu, W.. Deep Spectral Clustering using Dual Autoencoder Network. 2019. CVR.
4. Quinn, Sh.. Spectral Clustering. 2015. Carnegie Mellon Univesity.
5. Chen, Xi., Nie, F., Huang, J.Zh., Yang, M.. Scalable Normalized Cut with Improved Spectral Rotation. 2017. IJCAI-17.

References

7. Nie, F., Zhu, W., Li, Xu.. Unsupervised Large Graph Embedding Based on Balanced and Hierarchical k-Means. 2020. IEEE Transactions on Knowledge and Data Engineering.
8. Wang, F., Zhao, C., Liu, J., Huang, H.. A Variational Image Segmentation Model based on Normalized Cut with Adaptive Similarity and Spatial Regularization. 2020. Cornell University.
9. Park, S., Zhao, H.. Spectral Clustering Based on Learning Similarity Matrix. 2018. Bioinformatics.