Reading Summary for GraSPy: Graph Statistics in Python

**1.Graspy: a strong graph analysis tool**

* Uses NetworkX for graph operations
* Mostly focus on Graph Analysis

**2.Basisc Modules in Graspy:**

**Utils :**

* IOs,based on NetworkX &NumPy basic operations/preprocessings

Embed:

* Graph->Matrix(low dimension Euclid space representation )
* Single Graph embedding:

Adjacency spectral embedding (ASE)

Laplacian spectral embedding (LSE)

* Multiple Graphs Embedding:

Omnibus embedding

multiple adjacency spectral embedding (MASE)

**Models :**

* fitting random graph models to an input graph
* ER, SBM, DCER, DCSBM, and RDPG

**Inference:**

* hypothesistests:
* latent position test & latent distribution test(under RDPG models)

**Cluster:**

* Scikit-learn based and compatible
* GMM(Bayesian estimation for no. of clusters)
* K-Means

**Plot:**

* Visualizations

Reading Summary for: ”Co-clustering directed graphs to discover asymmetries and directional communities”

Key proposal:

A DI-SIM algorithm for dual-direction clustering in directed graph

Method DI-SIM:

**Routine:**

1.Compute the regularized graph Laplacian

2. Compute the top K left and right singular vectors: XL∈Rn×K,XR∈Rn×K whereK=min{ky,kz}.

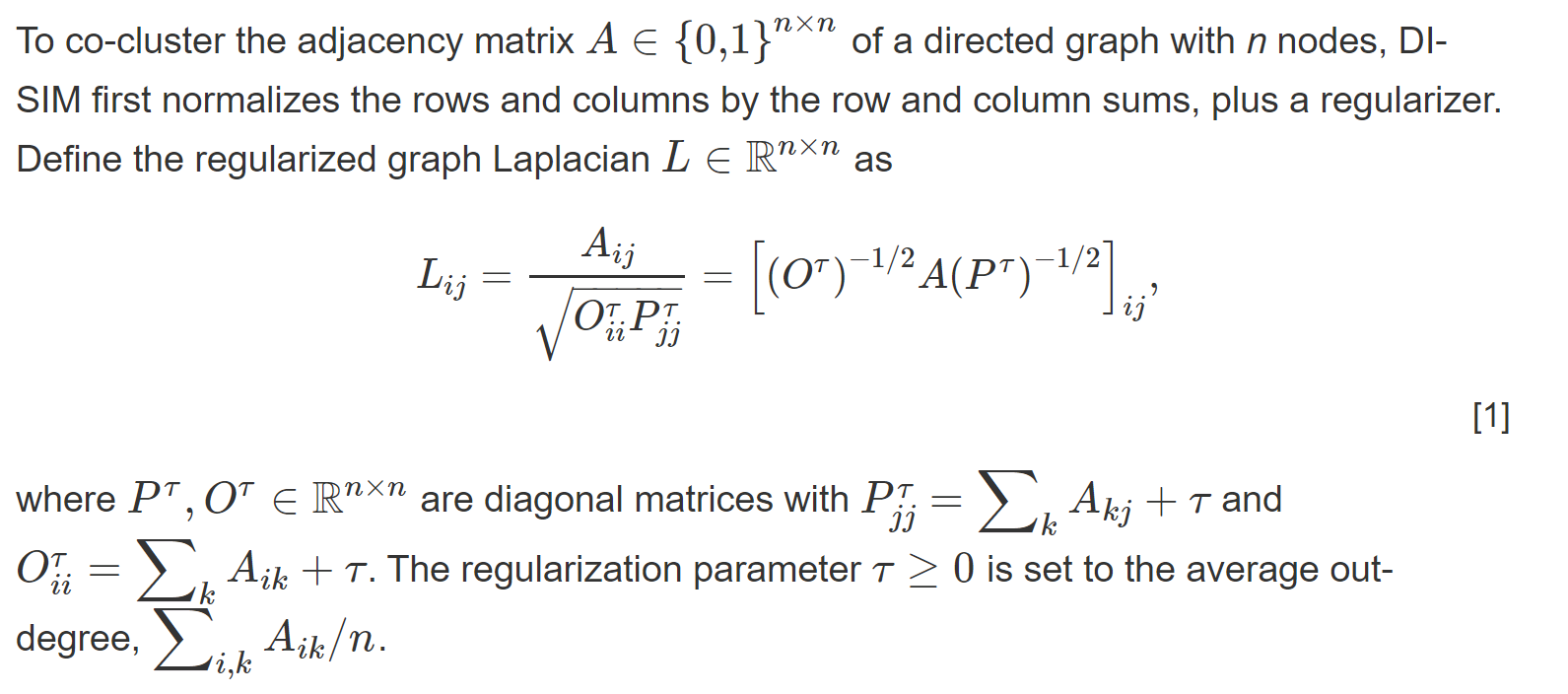
3. Normalize each row of XL and XR to have unit length.

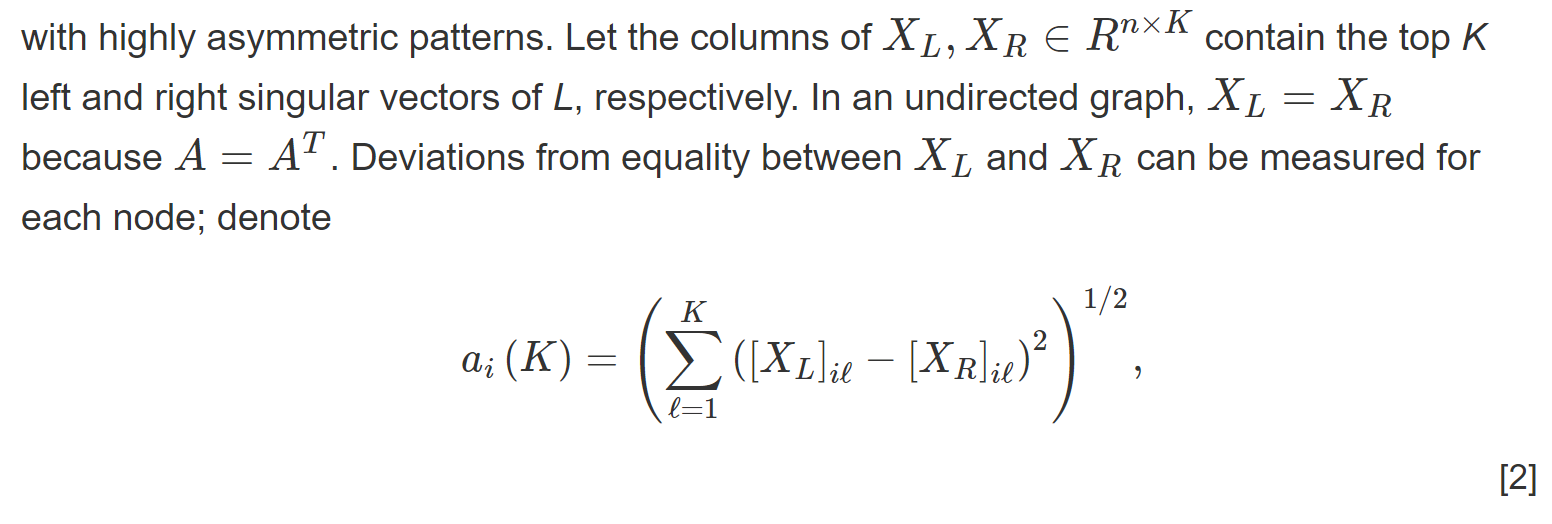
4A). If ky=kz=K, run *k-*means on the rows of X\*=(X\*L X\*R)T∈R2n×K

4B). run *k* means separately on rows of X\*L and X\*R, using ky and kz clusters, respectively.

**KeyPoints:**

* Laplacian Regularization:



* Using average degree to regularize Adjacency matrix
* Selection of the No.of clusters:
  + Asymmetric score
    - If symmetric ai=0, measurement of the asymmetric property for
  + K selected by inspecting the singular values. Or prior knowledge indicates a reasonable choice of K
* Using 2-norm to uniform the eig vectors on unit sphere

