3 LAPLACIAN of a NETWORK

Laplacian matrix of a network describes/contains all relevant information of a network.

Is defined as: $\boxed{L = D - A}$

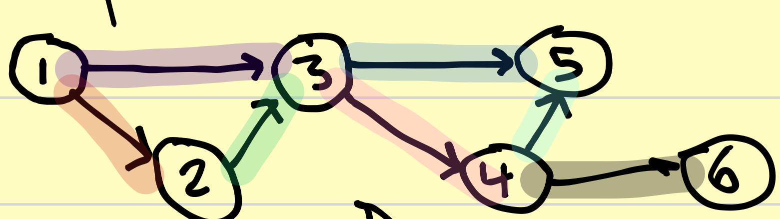
L = Laplacian Matrix
D = Degree Matrix
A = Adjacency Matrix

$$D \doteq \text{degree Matrix} \equiv D = \begin{cases} d_{ij} & i = j \\ 0 & i \neq j \end{cases}$$

d_{ij} = degree of the node ≡
≡ #neighbours of the node

$$A \equiv \text{adjacency Matrix} \equiv A = \begin{cases} 1 & \text{if } i \text{ and } j \text{ have a connection} \\ 0 & \text{if } i \text{ and } j \text{ don't have a connection} \end{cases}$$

Example: Please calculate the laplacian (matrix) of the network.



NICHT EUKLYDISCHEN DATASET

$$L = D - A = \begin{array}{c} D \\ \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} 2 & & & & & \\ & 2 & & & 0 & \\ & & 4 & & & \\ & 0 & & 3 & & \\ & & & & 2 & \\ & & & & & 1 \end{bmatrix} \end{array} \end{array} - \begin{array}{c} A \\ \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$$L = \begin{array}{c|cccccc} & 1 & 2 & 3 & 4 & 5 & 6 \\ \hline 1 & 2 & -1 & -1 & 0 & 0 & 0 \\ 2 & 0 & 2 & -1 & 0 & 0 & 0 \\ 3 & 0 & 0 & 4 & -1 & -1 & 0 \\ 4 & 0 & 0 & 0 & 3 & -1 & -1 \\ 5 & 0 & 0 & 0 & 0 & 2 & 0 \\ 6 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}$$

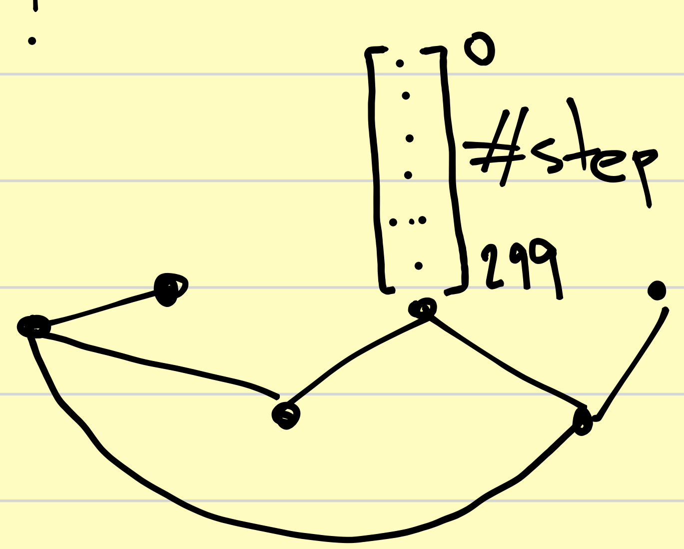
EUKLYDISCHEN DATASET

- DIE EIGENVEKTOREN DER LAPLACIAN MATRIX ENTHALTEN DIE SPEKTRALE INFORMATION DES GRAPHS / NETZWERKS.

(!) RECALL EIGENVALUES & EIGENVECTORS FROM ALGEBRA.

case study: we have two networks, each with 100 nodes. Each node represents an individual person and for each person we have a KPI (# steps every day) for 300 days:

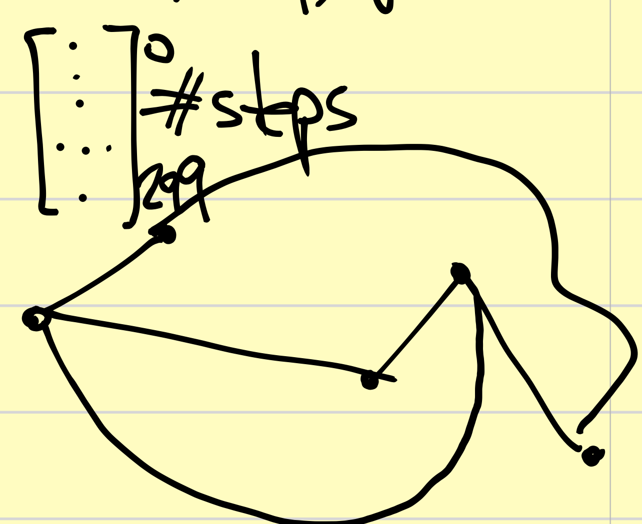
G_1



$N = 100$

$G1 = \{100, E_1\}$

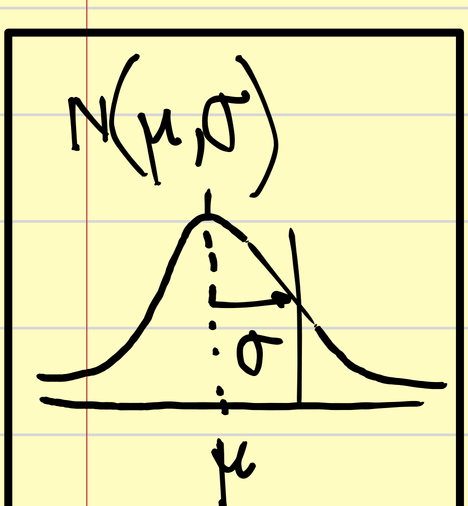
$G2$



$N = 100$

$G_2 = \{100, E_2\}$

EXERCISE. please write a code in python 3.x that generates two graphs with each 100 nodes, and each node has attached a vector of 300 elements that is normally distributed $N_1(1000, 30)$; $N_2(2500, 60)$.
NETWORK 1 NETWORK 2



The topology of Network 1 should be a ERDÖS-RENYI Netzwerk.

The topology of Network 2 should be a BARABASI Netzwerk with $\gamma = 2'4$.

python: networkx

NETWORK TOPOLOGY.

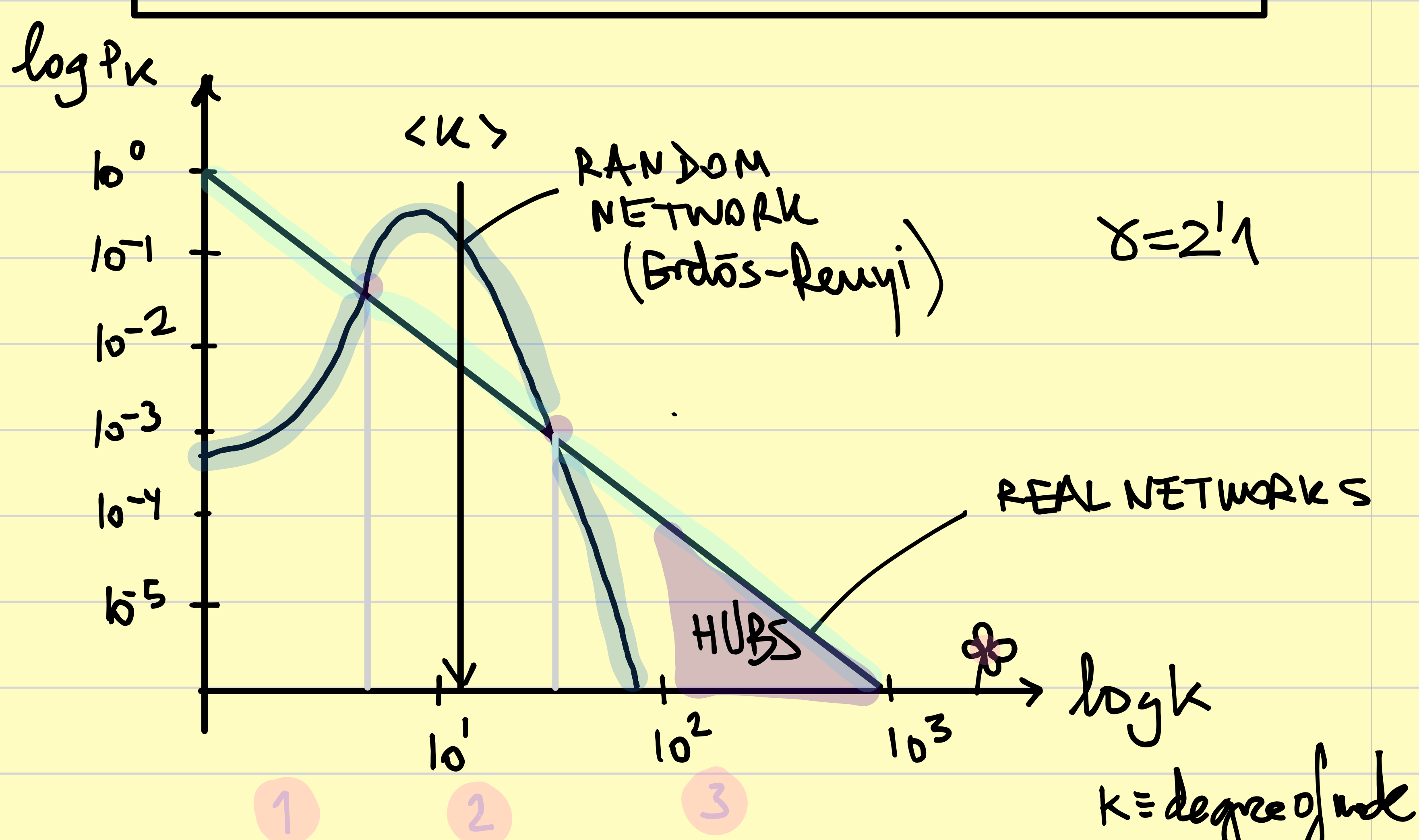
· The probability that a node with "k" neigbours in a random network attaches a new node is given by a POISSON distribution with parameter λ:

$$P_k \equiv \boxed{P_{Random}(X = k) = \frac{\lambda^k \cdot e^{-\lambda}}{k!}}$$

· The probability that a node is connected to another with "k" neighbours in real networks is given by:

POWER LAW

$$P_k \equiv \boxed{P_{Real}(X = k) = k^{-\gamma} \qquad \gamma = \text{Exponent degree.}}$$



1 RANDOM << REAL . For small "k" the powerlaw (real) is above the poisson (random), indicating that the real world has more nodes with few neigbours.

2 RANDOM > REAL . for nodes around the average degree $\langle k \rangle$, the random network has an excess of nodes.

3 RANDOM << REAL . For large „k“ the real network presents many nodes with high „k“.

The role of the exponent γ:

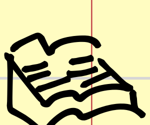
$\gamma \leq 2$. ANOMALOUS REGIME.

The number of links connected to the largest hub grows faster than the size of the network.
This means the largest hub has more nodes that the overall network. ↯.
This networks break down (!)

$2 < \gamma < 3$. SCALE FREE REGIME

In this case, networks can grow in a healthy manner.

$\gamma > 3$. RANDOM NETWORK.


NETWORK SCIENCE . (Baràbasi, 2016)