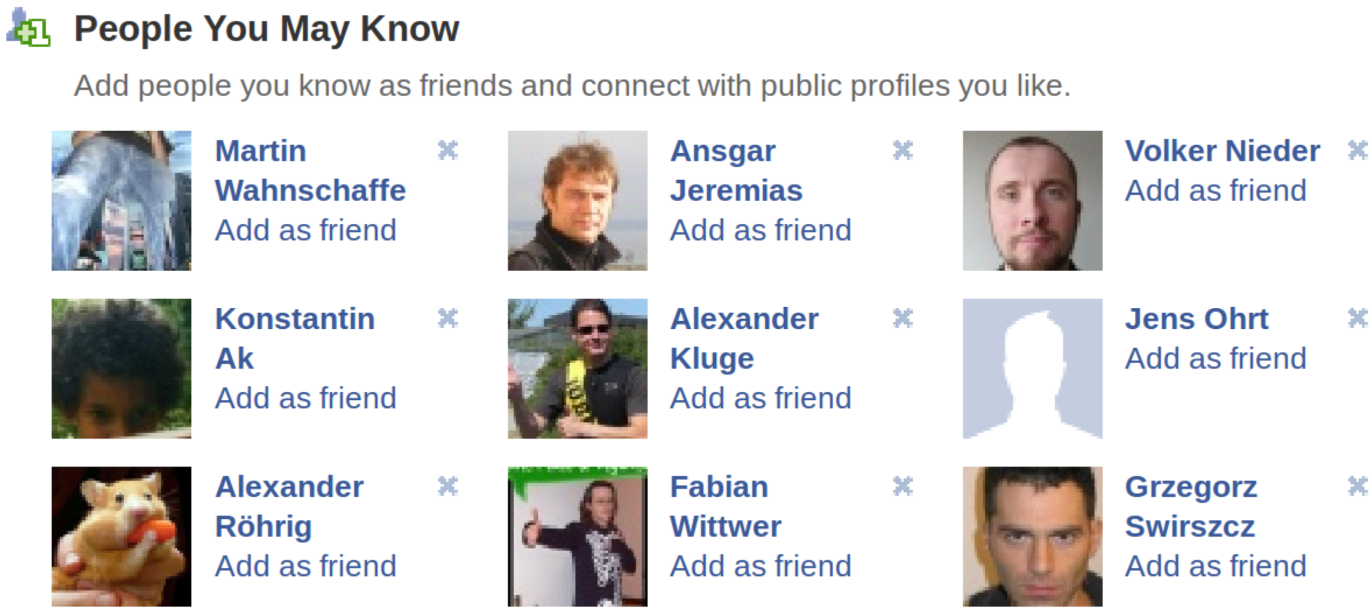
Network Growth and the Spectral Evolution Model

Jérôme Kunegis[1], Damien Fay[2], Christian Bauckhage[3]
[1] University of Koblenz-Landau
[2] University of Cambridge
[3] Fraunhofer IAIS

CIKM 2010, Toronto, Canada

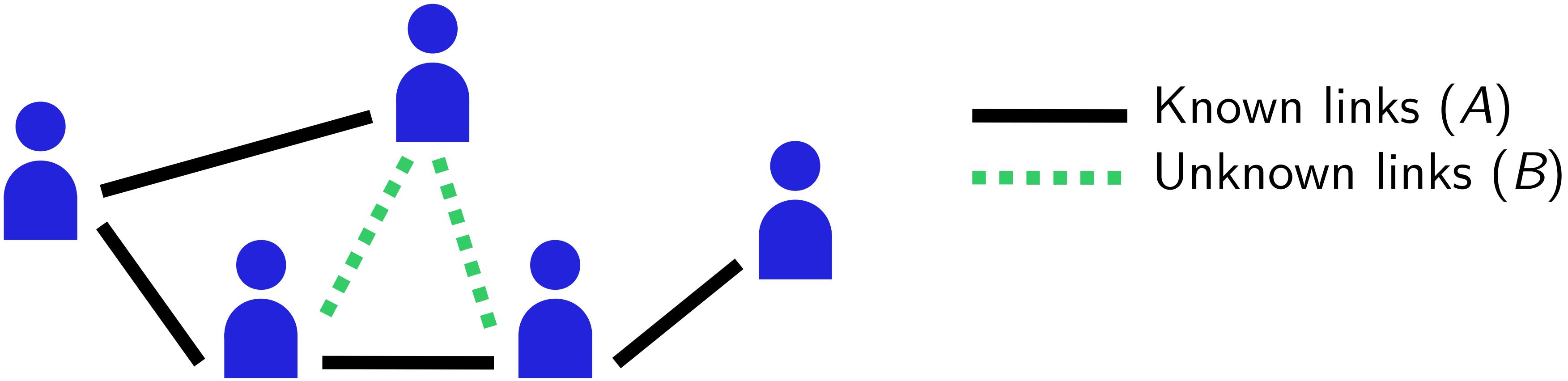
People and Knowledge Networks

Recommender Systems



Example: Find friends of Facebook

Jérôme Kunegis
kunegis@uni-koblenz.de

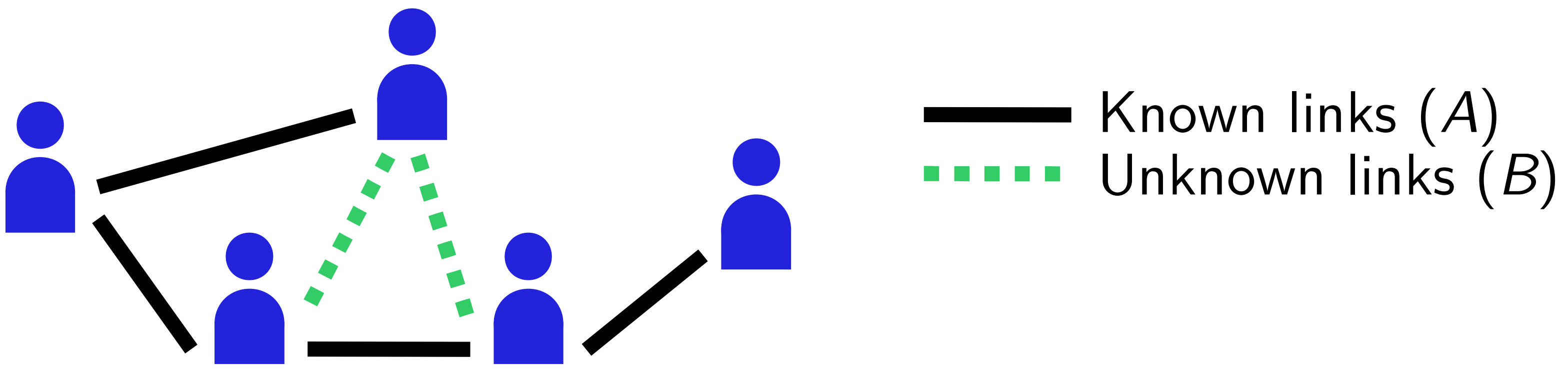


The users of Facebook are connected by friendship links, forming a graph. This graph is undirected.

Let A be the set of links in the network. Let B be the set of links that will appear in the future.

Task: Find a suitable function $f(A) = B$.

Algebraic Graph Theory



Use adjacency matrices $\mathbf{A}, \mathbf{B} \in \{0, 1\}^{n \times n}$:

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Use th eigenvalue decomposition :

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$$
$$\mathbf{B} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T$$

- **U** and **V** are orthogonal and contain eigenvectors
- **Λ** and **Σ** are diagonal and contain eigenvalues

Task : find an *f* of the following form :

$$f(\mathbf{U}\mathbf{\Lambda}\mathbf{U}^T) = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T$$

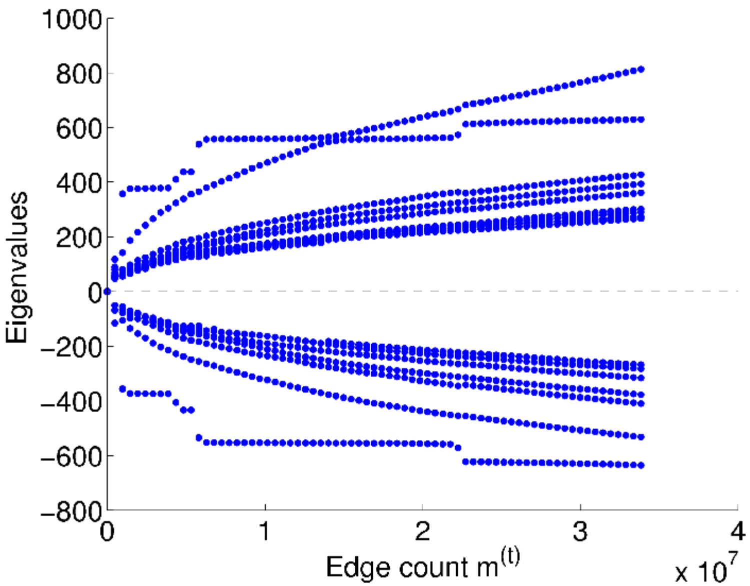
In this talk :

- The observation that $\mathbf{U} \approx \mathbf{V}$
- Extrapolation of **Λ** to **Σ**

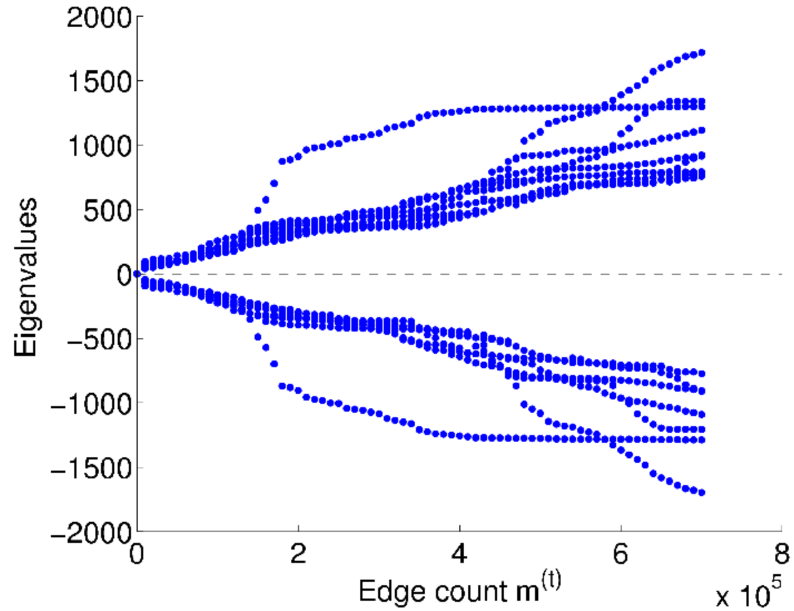
Outline

- Eigenvalue evolution
- Eigenvector evolution
- Diagonality test
- The spectral evolution model
- Explanations
- Control tests
- Spectral extrapolation

Eigenvalue Evolution



Wikipedia

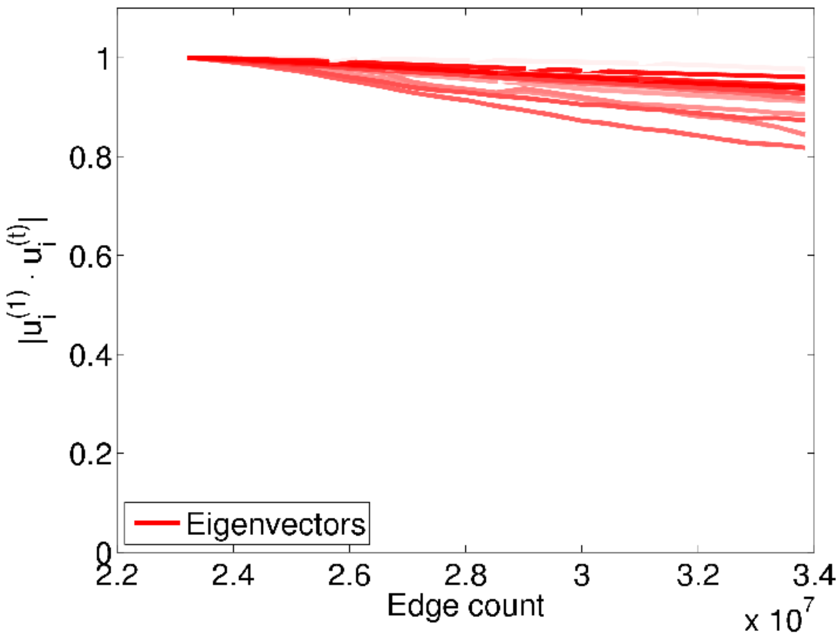


Facebook

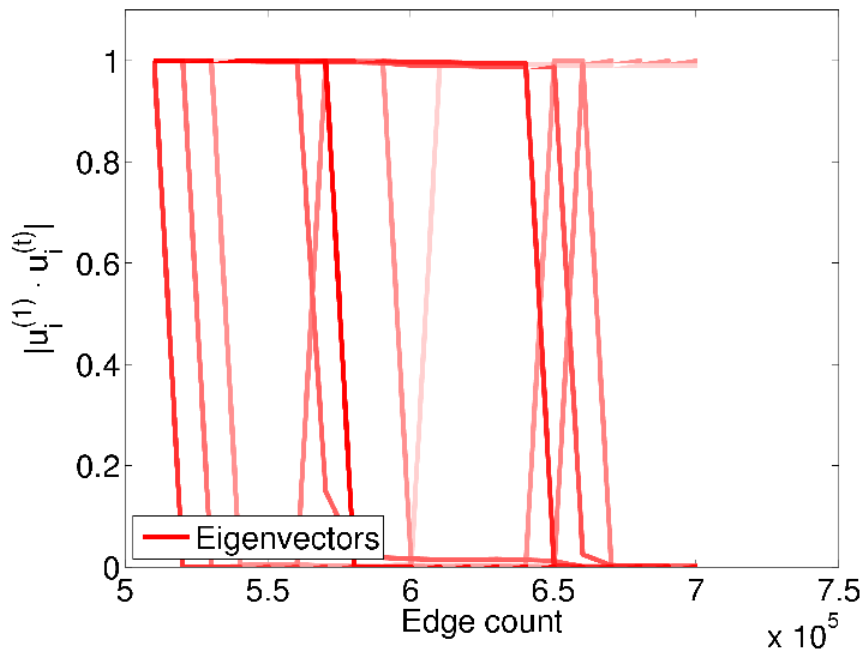
- Eigenvectors grow
- Not all eigenvectors grow at the same speed, even in a single network

Eigenvector Evolution

Compute the cosine over time between eigenvectors and their initial value.



Wikipedia



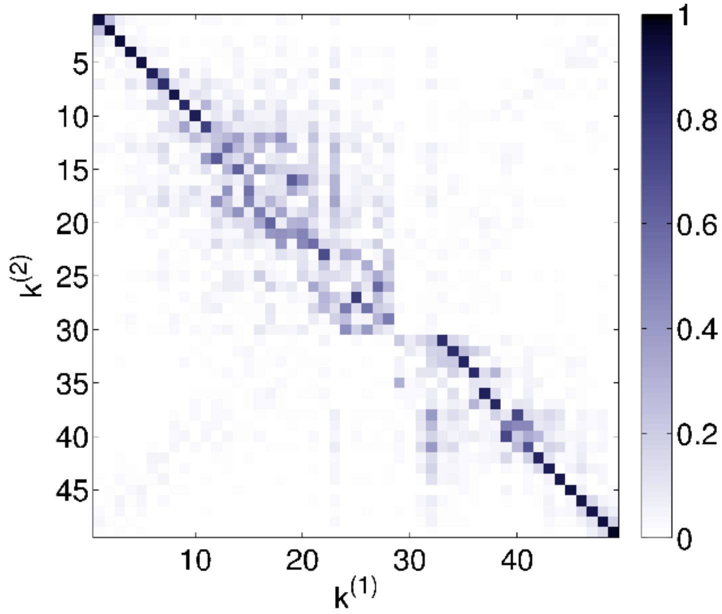
Facebook

- Some eigenvectors stay constant
- Some eigenvectors change suddenly

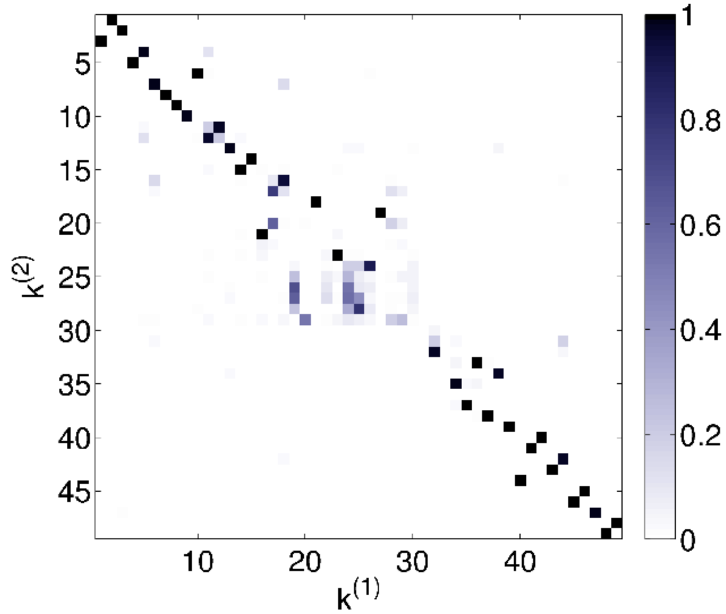
Jérôme Kunegis
kunegis@uni-koblenz.de

Eigenvector Permutation

Compute the cosine between all eigenvector pairs at two times.



Wikipedia



Facebook

Eigenvalues get permuted :

$$\mathbf{A} = \mathbf{U}\Lambda\mathbf{U}^T$$
$$\mathbf{B} = \mathbf{V}\Sigma\mathbf{V}^T$$
$$\mathbf{U}_i = \mathbf{V}_j$$

Jérôme Kunegis
kunegis@uni-koblenz.de

Diagonality Test

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$$

$$\mathbf{B} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T$$

Diagonalize **B** using **U**.

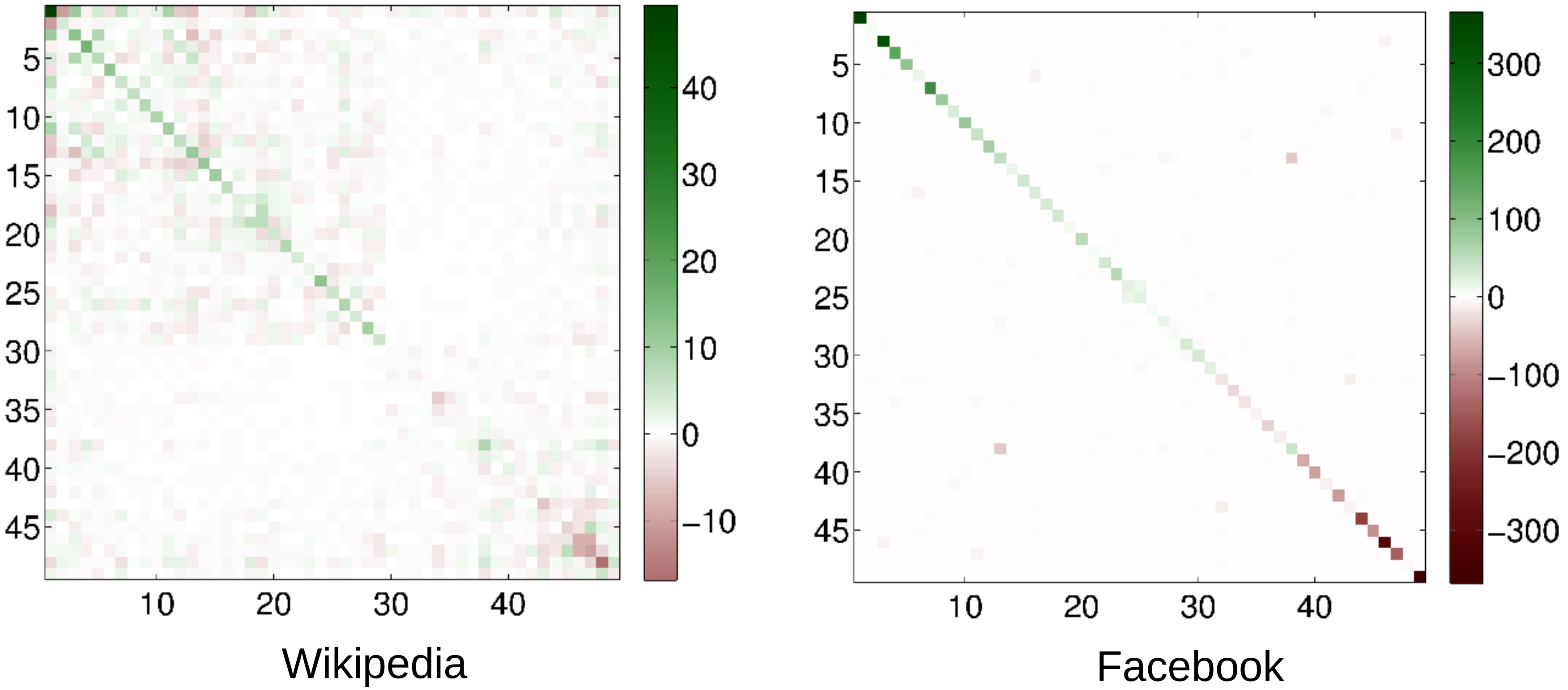
$$\mathbf{B} = \mathbf{U}\mathbf{D}\mathbf{U}^T$$

$$\mathbf{D} = \mathbf{U}^{-1}\mathbf{B}(\mathbf{U}^T)^{-1}$$

$$\mathbf{D} = \mathbf{U}^T\mathbf{B}\mathbf{U}$$

$\mathbf{U}^T\mathbf{B}\mathbf{U}$ should be diagonal!

Diagonality Test



- **D** is nearly diagonal
- The diagonal of **D** is irregular

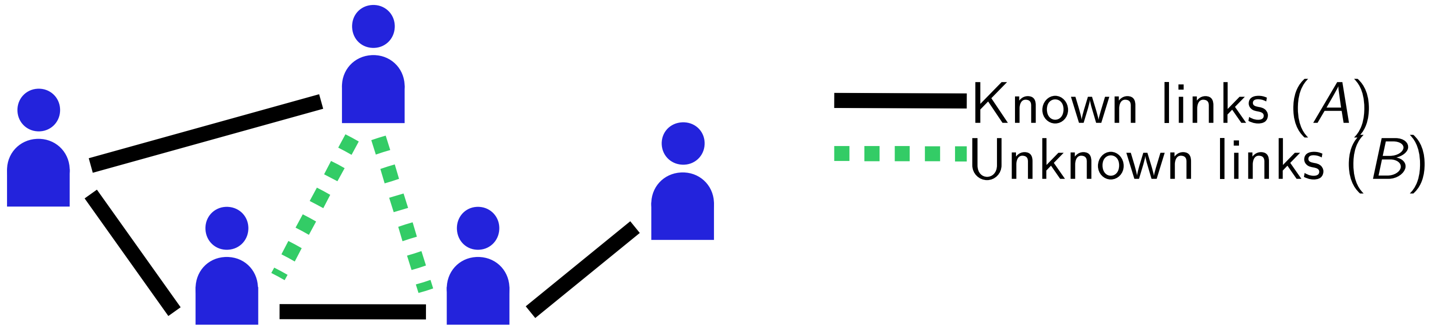
The Spectral Evolution Model

Networks grow spectrally

- Eigenvectors stay constant
- Eigenvalues change

Why?

Explanation : Matrix Powers



The square of **A** constains the number of paths of length two between any node pair:

$$\mathbf{A}^2 = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 \\ 1 & 0 & 2 & 0 & 1 \\ 0 & 1 & 0 & 2 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Generally, $\mathbf{A}^k$ contains the number of k-paths between any node pair.

Explanation: Polynomials

$$p(\mathbf{A}) = \alpha\mathbf{A}^2 + \beta\mathbf{A}^3 + \gamma\mathbf{A}^4 + \ldots$$

Polynomials are good link prediction functions :

- Count parallel paths
- Weight paths by length ($\alpha > \beta > \gamma > \ldots$)

The matrix power is a spectral transformation, e.g.:

$$\mathbf{A}^2 = (\mathbf{U\Lambda U}^T)(\mathbf{U\Lambda U}^T) = \mathbf{U\Lambda}^2\mathbf{U}^T$$

Polynomials are spectral transformations:

$$p(\mathbf{A}) = p(\mathbf{U\Lambda U}^T) = \mathbf{U}p(\mathbf{\Lambda})\mathbf{U}^T$$

Explanation: Graph Kernels

Matrix exponential

$$\exp(\mathbf{A}) = \mathbf{I} + \mathbf{A} + \tfrac{1}{2}\,\mathbf{A}^2 + \ldots = \mathbf{U}\exp(\boldsymbol{\Lambda})\mathbf{U}^T$$

Von Neumann kernel

$$(\mathbf{I} - \alpha\mathbf{A})^{-1} = \mathbf{I} + \alpha\mathbf{A} + \alpha^2\mathbf{A}^2 + \ldots = \mathbf{U}(\mathbf{I} - \alpha\boldsymbol{\Lambda})^{-1}\mathbf{U}^T$$

Explanation: Preferential Attachment

Write **A** as a sum of rank-1 matrices:

$$\mathbf{A} = \mathbf{A}_1 + \mathbf{A}_2 + \ldots$$

$$\mathbf{A}_i = \lambda_i u_i u_i^{\mathrm{T}}$$

- Interpret each $\mathbf{A}_i$ as the adjacency matrix of one weighted graph
- In $\mathbf{A}_i$, vertex j has degree $\sum_k \lambda_i u_{ij} u_{ik} \sim u_{ij}$

Consider the process of preferential attachment in each latent dimension separately:

$$\Sigma_i u_i u_i^{\mathrm{T}} = \lambda_i u_i u_i^{\mathrm{T}} + \varepsilon_i u_i u_i^{\mathrm{T}}$$

$$\mathrm{pa}(\mathbf{A}) = \mathbf{U}(\mathbf{\Lambda} + \mathbf{E})\mathbf{U}^{\mathrm{T}}$$

Control: Matrix Perturbation

Add edges at random to $\mathbf{A} = \mathbf{U\Lambda U}^T$.

The evolution of **A** should then be :

$$\mathbf{A} + \mathbf{E} = \tilde{\mathbf{U}}\,\tilde{\mathbf{\Lambda}}\,\tilde{\mathbf{U}}^T$$

$$\|\,\mathbf{\Lambda} - \tilde{\mathbf{\Lambda}}\,\|_F = O(\varepsilon^2)$$

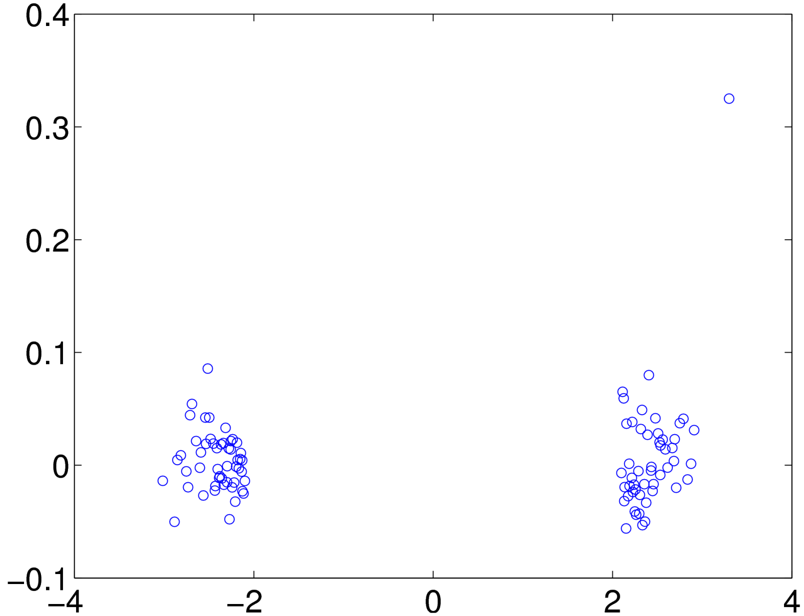
$$|\,\mathbf{U}_{.k}{}^T\,\tilde{\mathbf{U}}_{.k}\,| = O(\varepsilon)$$

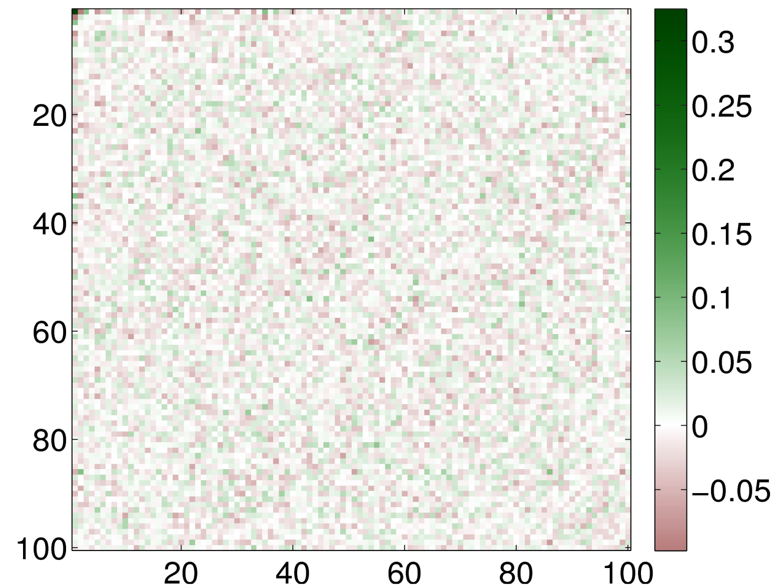
Using $\|\mathbf{E}\|_2 = \varepsilon$.

- Random growth is *not* spectral.

Control: Random Sampling

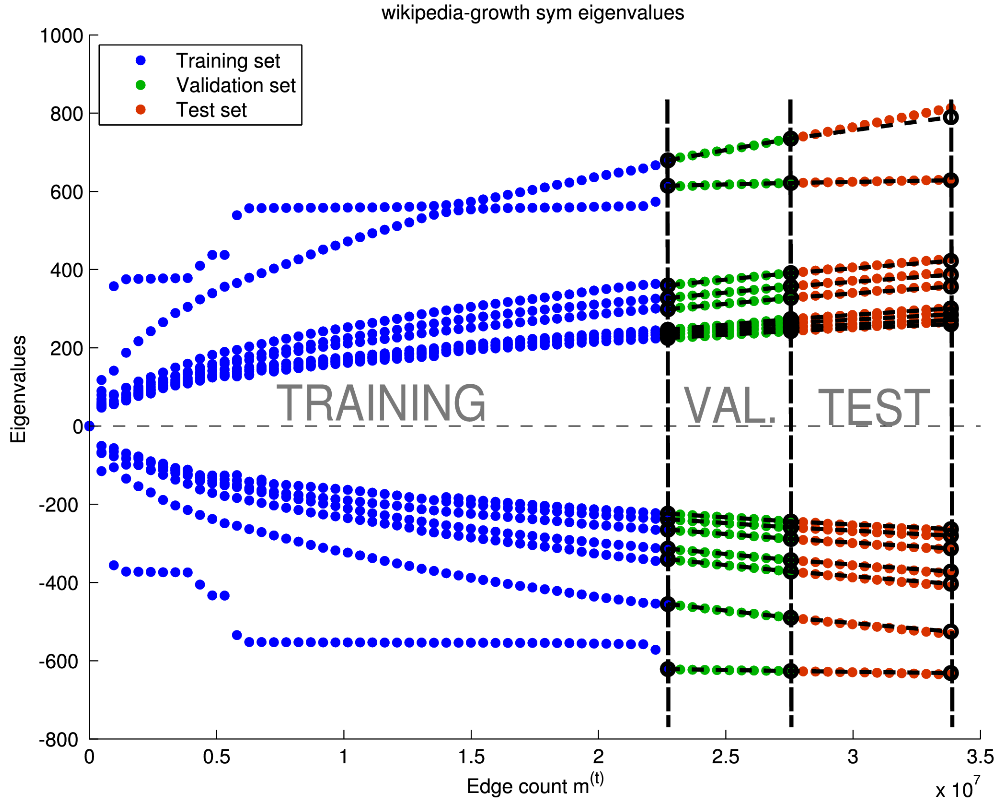
Split an Erdős–Rényi random graph into **A** + **B**. Apply the diagonality test for transforming **A** into **B**.





- Only one latent dimension is preserved.

To predict links, extrapolate the evolution of eigenvalues.

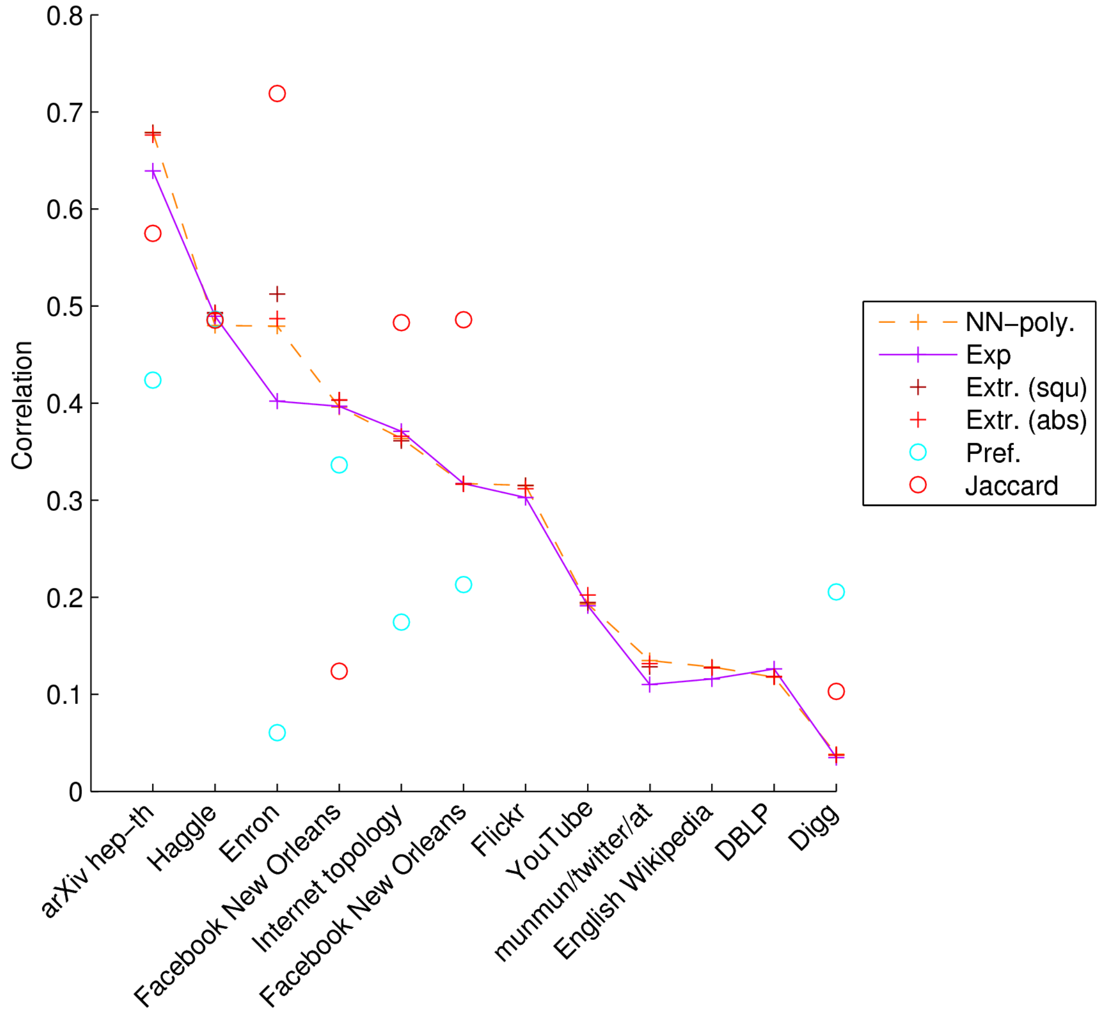


Experiments

Methodology :

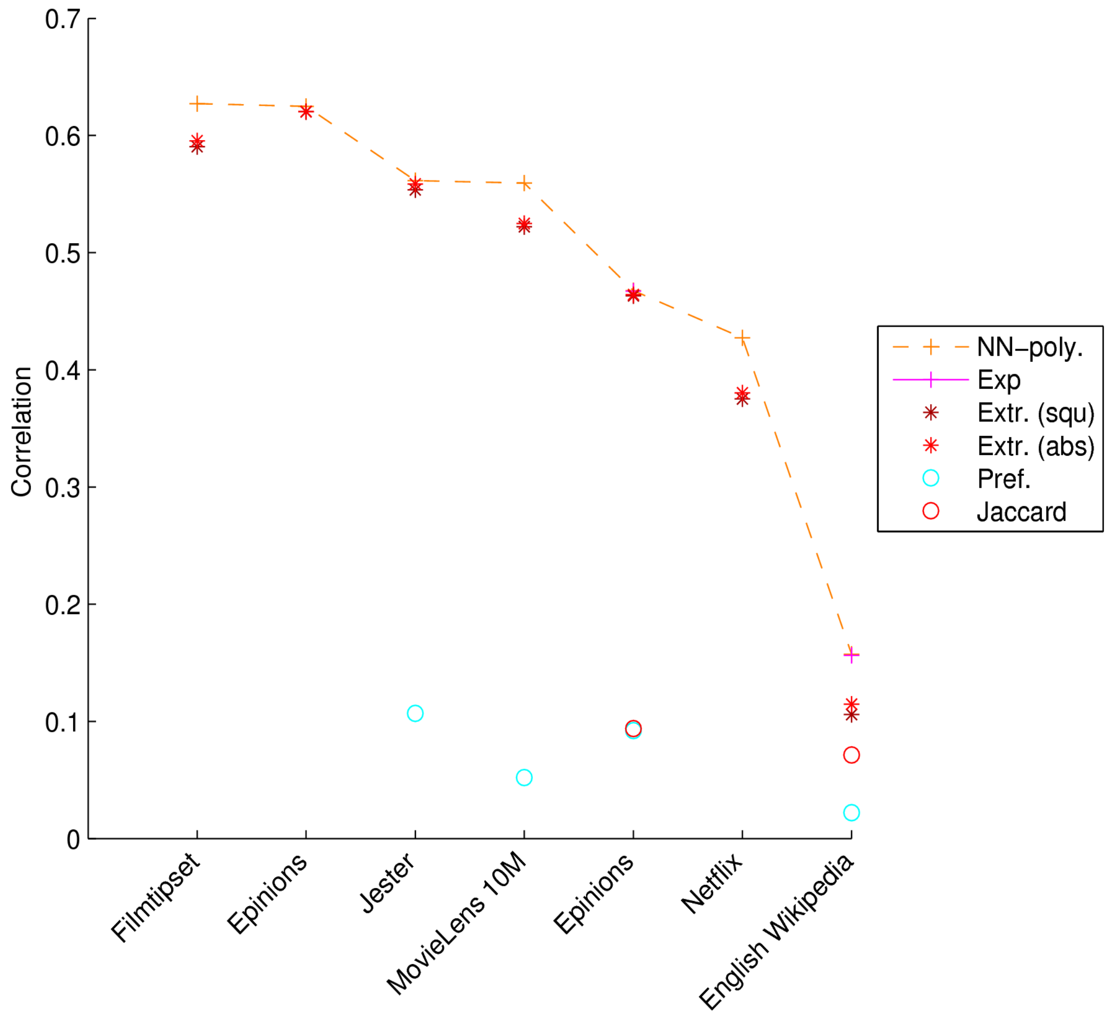
- Retain newest edges as training set
- Compute link prediction scores
- Evaluate using the mean average precision (MAP)
- User over a hundred datasets

Experiments: Symmetric Networks



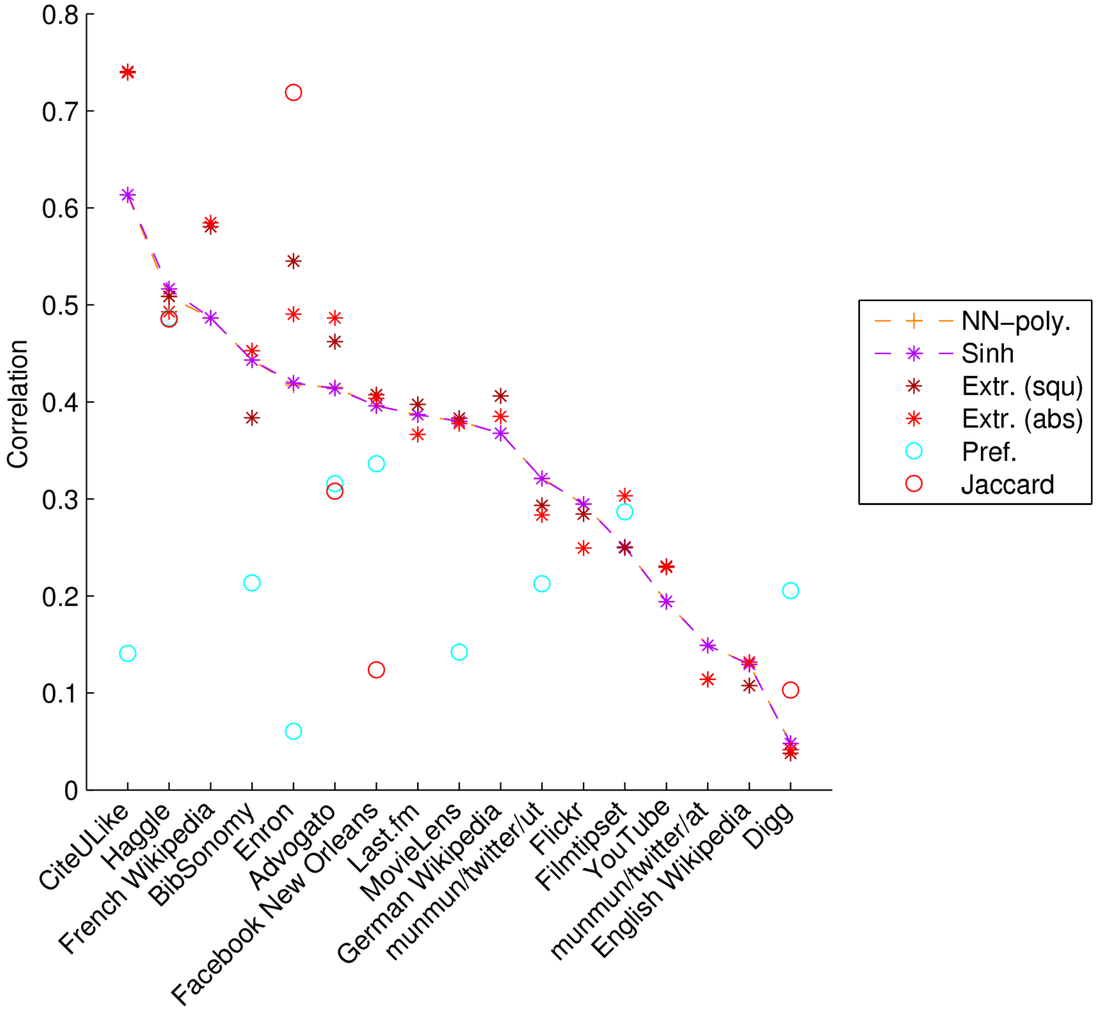
Experiments: Weighted Networks

Use the weighted adjacency matrix **A**.



Experiments: Bipartite and Directed Networks

Use the singular value decomposition $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^{\mathrm{T}}$.



Summary & Discussion

- Eigenvectors remain constant
- Eigenvalues grow irregularly
- Extrapolate the eigenvalues to predict links

Experimental results:

- Extrapolation works best for bipartite and directed networks