On the Spectral Evolution of Large Networks

Jérôme Kunegis

Committee: Prof. Dr. Staab
Prof. Dr. Bauckhage
Prof. Dr. Obermayer

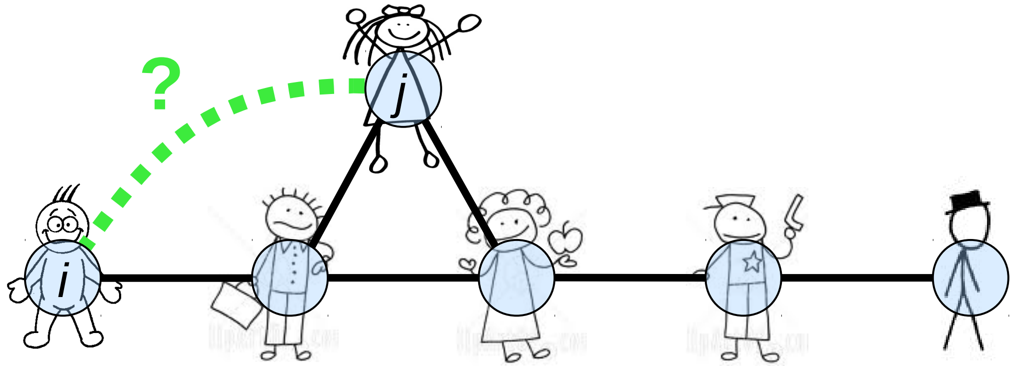
WeST
People and Knowledge Networks

UNIVERSITÄT KOBLENZ · LANDAU

Outline

1. Algebraic Link Prediction
2. Spectral Transformations
3. Learning Link Prediction

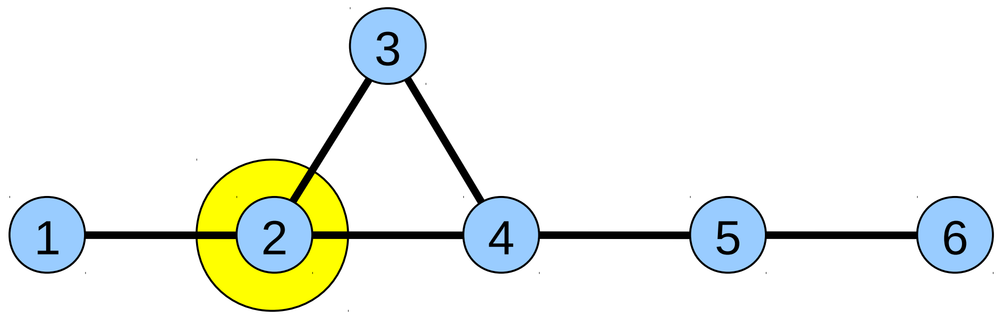
1. Example: Recommend Friends on Facebook



A network of friends connected by friendship links

- Recommendation: Given a person *i*, find new friends *j* for that person
- Link prediction: Find edges (*i*, *j*) that will appear in the future

Algebraic Graph Theory



Represent a network
by an adjacency matrix **A**:

A$_{ij}$ = 1 when *i* and *j* are connected
A$_{ij}$ = 0 when *i* and *j* are not connected

A is square and symmetric.

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

Eigenvalue Decomposition

Write the matrix **A** as a product:

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

where

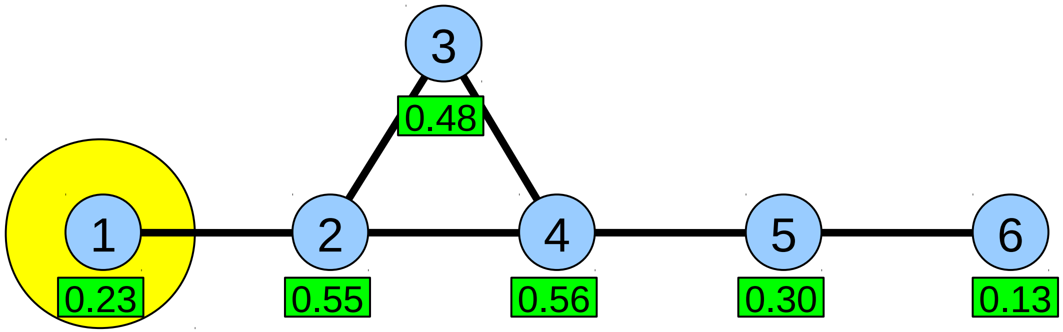
U is an orthogonal matrix, i.e. $\mathbf{U}^\mathrm{T}\mathbf{U} = \mathbf{I}$

Λ is a diagonal matrix, i.e. $\mathbf{\Lambda}_{ij} = 0$ when $i \neq j$

The eigenvalue decomposition always exists for symmetric matrices.

Jérôme Kunegis
kunegis@uni-koblenz.de

Eigenvalue Decomposition: Example



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

$$
\underbrace{\begin{bmatrix} 0.25 & -0.43 & -0.17 & 0.76 & 0.30 & 0.23 \\ -0.44 & 0.59 & 0.10 & 0.21 & 0.33 & 0.55 \\ -0.10 & -0.56 & 0.51 & -0.39 & 0.18 & 0.48 \\ 0.61 & 0.18 & -0.41 & -0.32 & -0.12 & 0.56 \\ -0.52 & -0.28 & -0.37 & 0.09 & -0.64 & 0.30 \\ 0.30 & 0.20 & 0.63 & 0.34 & -0.59 & 0.13 \end{bmatrix}}_{\mathbf{U}}
\underbrace{\begin{bmatrix} -1.74 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1.37 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.59 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.27 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2.33 \end{bmatrix}}_{\mathbf{\Lambda}}
\underbrace{\begin{bmatrix} 0.25 & -0.43 & -0.17 & 0.76 & 0.30 & 0.23 \\ -0.44 & 0.59 & 0.10 & 0.21 & 0.33 & 0.55 \\ -0.10 & -0.56 & 0.51 & -0.39 & 0.18 & 0.48 \\ 0.61 & 0.18 & -0.41 & -0.32 & -0.12 & 0.56 \\ -0.52 & -0.28 & -0.37 & 0.09 & -0.64 & 0.30 \\ 0.30 & 0.20 & 0.63 & 0.34 & -0.59 & 0.13 \end{bmatrix}^T}_{\mathbf{U}}
$$

Eigenvector Eigenvalue

U contains eigenvectors, **Λ** contains eigenvalues

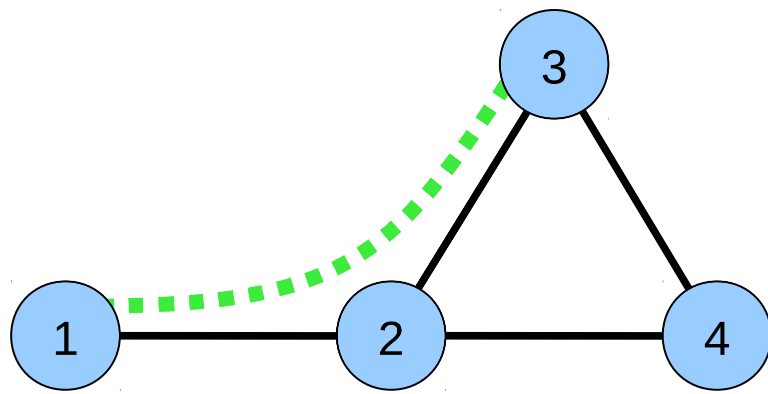
WeST People and Knowledge Networks
Jérôme Kunegis
kunegis@uni-koblenz.de

UNIVERSITÄT KOBLENZ · LANDAU

The eigenvalue decomposition can be used to implement the *Friend of a Friend* count for link prediction:

Consider the matrix product **A A** = $\mathbf{A}^2$

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^2 = \begin{bmatrix} 1 & 0 & \boxed{1} & 1 & 0 & 0 \\ 0 & 3 & 1 & 1 & 1 & 0 \\ 1 & 1 & 2 & 1 & 1 & 0 \\ 1 & 1 & 1 & 3 & 0 & 1 \\ 0 & 1 & 1 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$



$(\mathbf{A}^2)_{ij}$ contains the number of paths of length two between i and j, i.e. the friend-of-a-friend score

Jérôme Kunegis
kunegis@uni-koblenz.de

Computing A²

Use the eigenvalue decomposition $\mathbf{A} = \mathbf{U}\,\mathbf{\Lambda}\,\mathbf{U}^T$

$$\mathbf{A}^2 = \mathbf{U}\,\mathbf{\Lambda}\,\boxed{\mathbf{U}^T\,\mathbf{U}}\,\mathbf{\Lambda}\,\mathbf{U}^T = \mathbf{U}\,\mathbf{\Lambda}\,\mathbf{\Lambda}\,\mathbf{U}^T = \mathbf{U}\,\boxed{\mathbf{\Lambda}^2}\,\mathbf{U}^T$$

Exploit **U** and **Λ**:

- $\boxed{\mathbf{U}^T\,\mathbf{U}} = \mathbf{I}$ because **U** is orthogonal
- $(\boxed{\mathbf{\Lambda}^2})_{ii} = \mathbf{\Lambda}_{ii}^2$ because **Λ** is diagonal

Result: Just square all eigenvalues!

We call this a **spectral transformation**.

Jérôme Kunegis
kunegis@uni-koblenz.de

Spectral Transformation

A spectral transformation is a function of a matrix **A** that can be expressed as a transformation of **A**'s eigenvalues.

Given the eigenvalue decomposition

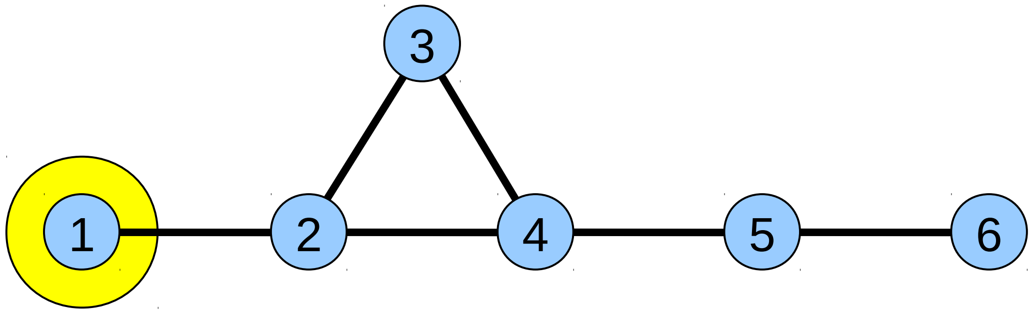
$$\mathbf{A} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^T,$$

then F is a spectral transformation when

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

and F(**Λ**) is diagonal.

Friend of a Friend of a Friend



Compute the number of friends-of-friends-of-friends:

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^3 = \begin{array}{cc} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} & \\ \begin{bmatrix} \boxed{0 \quad 3 \quad 1 \quad 1 \quad 1 \quad 0} \\ 3 \quad 2 \quad 4 \quad 5 \quad 1 \quad 1 \\ 1 \quad 4 \quad 2 \quad 4 \quad 1 \quad 1 \\ 1 \quad 5 \quad 4 \quad 2 \quad 4 \quad 0 \\ 1 \quad 1 \quad 1 \quad 4 \quad 0 \quad 2 \\ 0 \quad 1 \quad 1 \quad 0 \quad 2 \quad 0 \end{bmatrix} & \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array} \end{array}$$

$$\mathbf{A^3 = U\, \Lambda\, \boxed{U^T\, U}\, \Lambda\, \boxed{U^T\, U}\, \Lambda\, U^T = U\, \Lambda^3\, U^T}$$

Jérôme Kunegis
kunegis@uni-koblenz.de

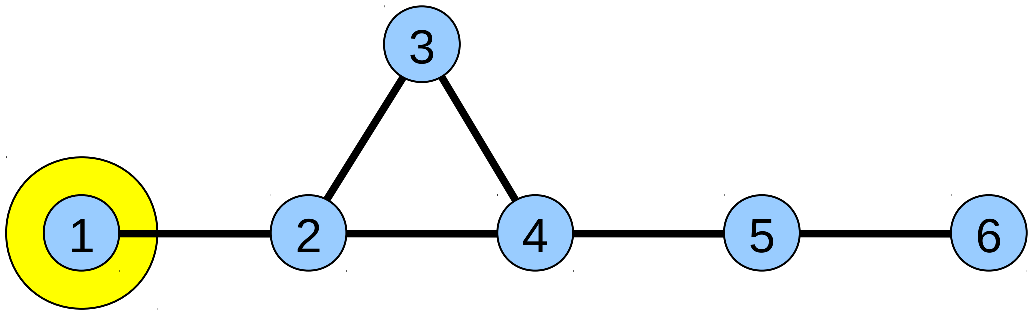
Link Prediction with Power Sums

Use power sums as link prediction functions:

- Every power $\mathbf{A}^n$ represents the number of paths of length n between all node pairs
  $\rightarrow$ New edges more likely to appear when there are **many paths** already

- A power sum $a\,\mathbf{A}^2 + b\,\mathbf{A}^3 + c\,\mathbf{A}^4 + \cdots$ represents a sum over all paths between all node pairs
  $\rightarrow$ When $a > b > c > \cdots > 0$, **short paths** are weighted more

Matrix Exponential



The matrix exponential can be written as a power sum with decreasing coefficients:

$$\exp(\mathbf{A}) = \mathbf{I} + \mathbf{A} + \tfrac{1}{2}\,\mathbf{A}^2 + \tfrac{1}{6}\,\mathbf{A}^3 + \cdots$$

$$\exp \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} = \begin{array}{c} \begin{array}{cccccc} \;\;1\;\; & \;\;2\;\; & \;\;3\;\; & \;\;4\;\; & \;\;5\;\; & \;\;6\;\; \end{array} \\ \left[\begin{array}{cccccc} \boxed{1.66} & \boxed{1.72} & \boxed{0.93} & \boxed{0.98} & \boxed{0.28} & \boxed{0.06} \\ 1.72 & 3.57 & 2.70 & 2.92 & 1.04 & 0.28 \\ 0.93 & 2.70 & 2.86 & 2.71 & 0.99 & 0.27 \\ 0.98 & 2.93 & 2.71 & 3.62 & 1.94 & 0.71 \\ 0.28 & 1.04 & 0.99 & 1.94 & 2.31 & 1.39 \\ 0.06 & 0.28 & 0.27 & 0.71 & 1.39 & 1.59 \end{array}\right] \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array}$$

Example: To ①, recommend ④, ③, ⑤, then ⑥

WeST People and Knowledge Networks
Jérôme Kunegis
kunegis@uni-koblenz.de

UNIVERSITÄT KOBLENZ · LANDAU

Let p(**A**) be a power sum:

$$
\begin{aligned}
\mathrm{p}(\mathbf{A}) &= a\,\mathbf{A}^2 + b\,\mathbf{A}^3 + c\,\mathbf{A}^4 + \cdots \\
&= a\,\mathbf{U}\,\mathbf{\Lambda}^2\,\mathbf{U}^\mathrm{T} + b\,\mathbf{U}\,\mathbf{\Lambda}^3\,\mathbf{U}^\mathrm{T} + c\,\mathbf{U}\,\mathbf{\Lambda}^4\,\mathbf{U}^\mathrm{T} + \cdots \\
&= \mathbf{U}\,(a\,\mathbf{\Lambda}^2 + b\,\mathbf{\Lambda}^3 + c\,\mathbf{\Lambda}^4 + \cdots)\,\mathbf{U}^\mathrm{T} \\
&= \mathbf{U}\,\mathrm{p}(\mathbf{\Lambda})\,\mathbf{U}^\mathrm{T}
\end{aligned}
$$

Because $\mathbf{\Lambda}$ is diagonal, we have $\mathrm{p}(\mathbf{\Lambda})_{ii} = \mathrm{p}(\mathbf{\Lambda}_{ii})$

Therefore:

Power sums are **spectral transformations!**

2. Looking at Real Facebook Data

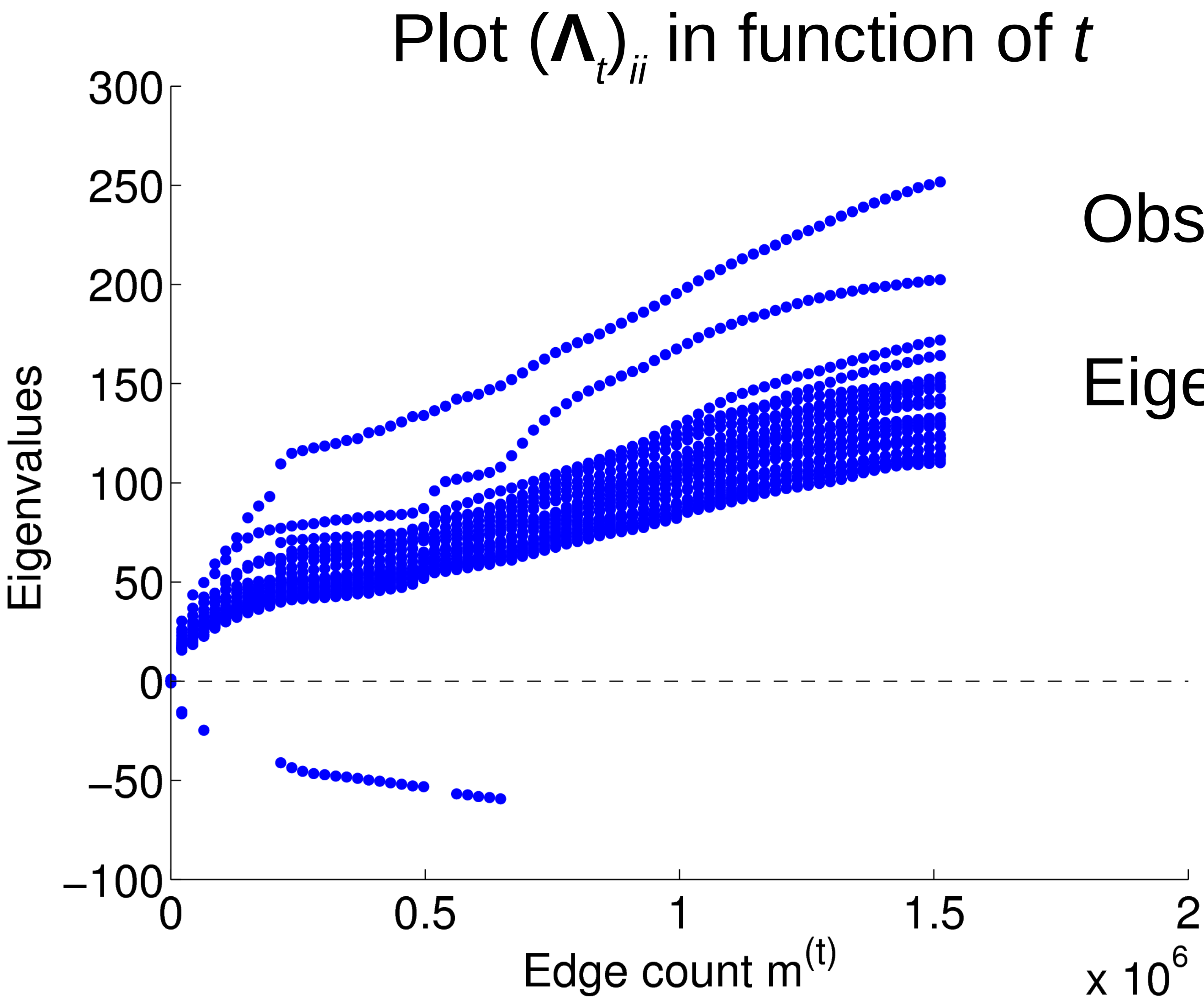
Dataset: Facebook New Orleans friendship links (Viswanath 2009)
http://socialnetworks.mpi-sws.org/data-wosn2009.html

63,731 persons
1,545,686 friendship links with **formation dates**

Let $\mathbf{A}_t$ be the adjacency matrix at time t ($1 \leq t \leq n = 71$)
$\mathbf{A}_t$ contains all edges formed before time t

Compute all eigenvalue decompositions $\mathbf{A}_t = \mathbf{U}_t \mathbf{\Lambda}_t \mathbf{U}_t^{\mathrm{T}}$

Evolution of Eigenvalues



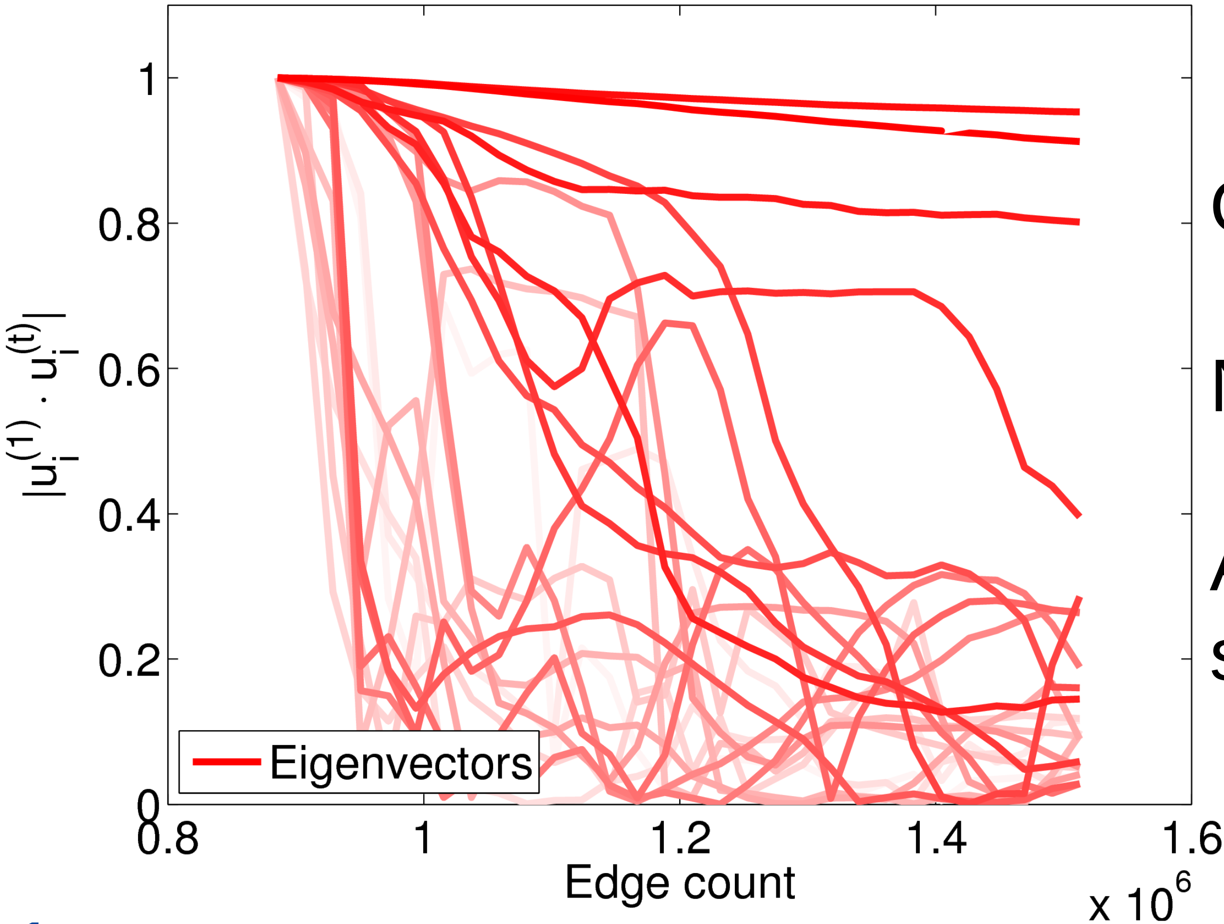
Observation:

Eigenvalues grow

Jérôme Kunegis
kunegis@uni-koblenz.de

Eigenvector Evolution

For each eigenvector nr. *i*, cosine similarity of each eigenvector with initial value (1 = same eigenvector)



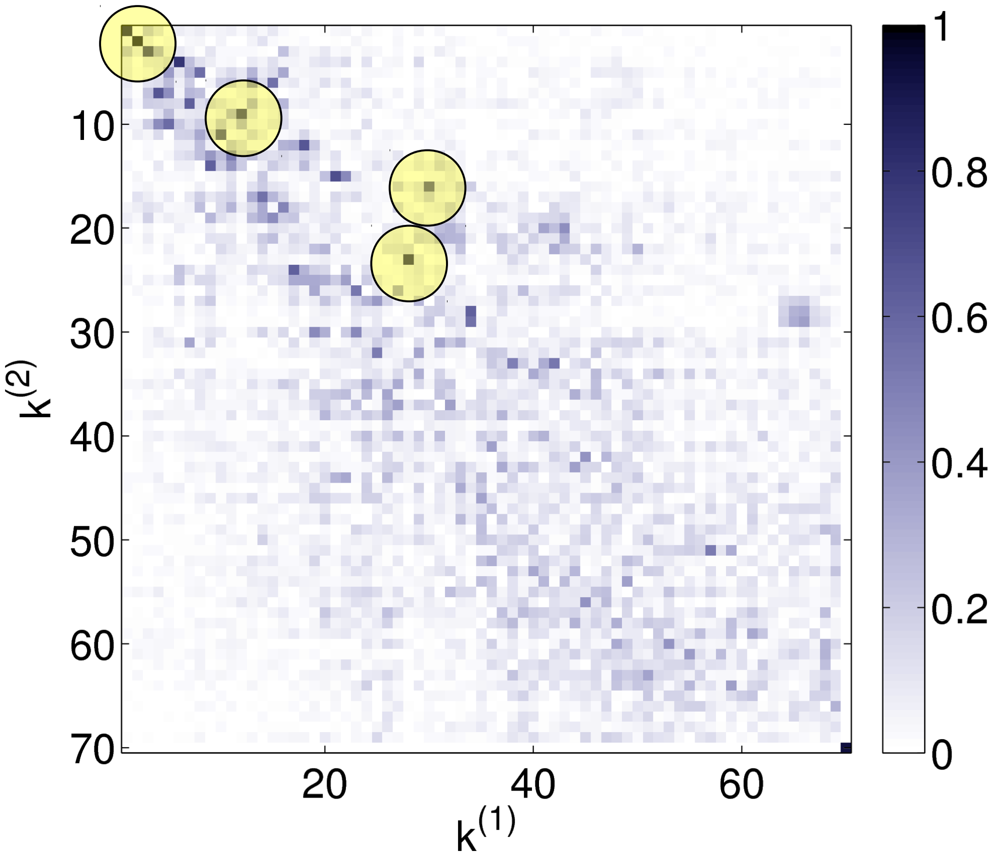
Observations:

No clear pattern

A few eigenvectors stay constant

Eigenvector Permutation

For all pairs (i,j), compare the i^{th} eigenvector at $m = (3/4)n$ with the j^{th} eigenvector at time n, using the cosine similarity (1 = same eigenvector)



Observations:

Some eigenvectors permute

Jérôme Kunegis
kunegis@uni-koblenz.de

Diagonality Test (Kunegis 2010)

Let $1 \leq m \approx ¾\, n \leq n$.

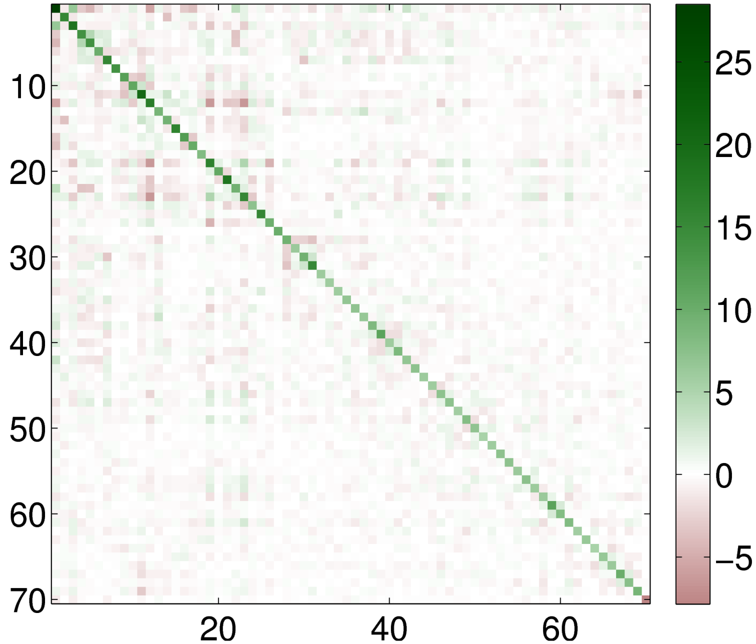
Diagonalize $\mathbf{A}_n - \mathbf{A}_m$ using the eigenvectors of $\mathbf{A}_m$

$$\mathbf{A}_m = \mathbf{U}\, \mathbf{\Lambda}\, \mathbf{U}^\mathrm{T}$$

$$\mathbf{A}_n - \mathbf{A}_m = \mathbf{U}\, \mathbf{X}\, \mathbf{U}^\mathrm{T}$$

| **U** is orthogonal

$$\Rightarrow \mathbf{X} = \mathbf{U}^\mathrm{T}\, (\mathbf{A}_n - \mathbf{A}_m)\, \mathbf{U}$$

$\mathbf{X} =$ 

Observation:

X is nearly diagonal

Growth is spectral!

Jérôme Kunegis
kunegis@uni-koblenz.de

To predict links, find a spectral transformation mapping past data to present data, and apply it to present data.

Idea: Use the adjacency matrices $\mathbf{A}_m$ and $\mathbf{A}_n$ for $1 \leq m \leq n$ with $m \approx ¾\, n$.

$$\mathbf{A}_m \xrightarrow{f} \mathbf{A}_n - \mathbf{A}_m$$

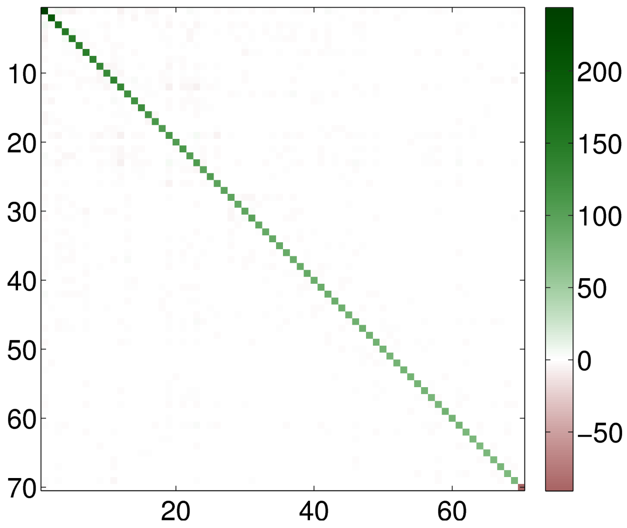
$$\mathbf{A}_n \xrightarrow{f} \mathbf{?}$$

Learning a Spectral Transformation

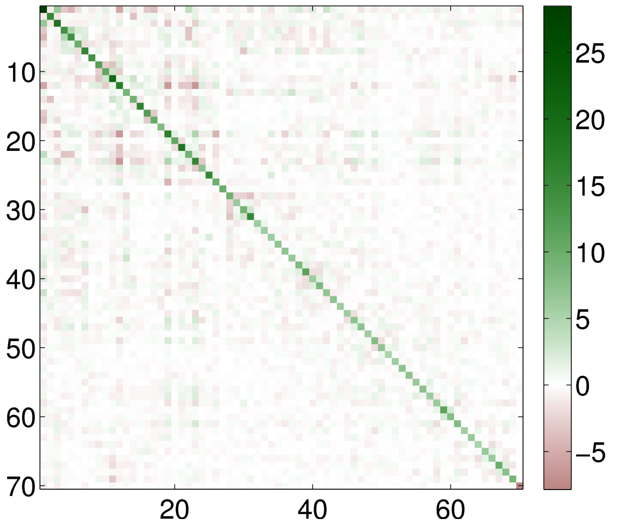
$$\mathbf{A}_m \longrightarrow \mathbf{A}_n - \mathbf{A}_m$$

$$\mathbf{U}\,\boldsymbol{\Lambda}\,\mathbf{U}^T \longrightarrow \boxed{\mathbf{U}\,\mathbf{U}^T}\,(\mathbf{A}_n - \mathbf{A}_m)\,\boxed{\mathbf{U}\,\mathbf{U}^T}$$

$$\boldsymbol{\Lambda} \longrightarrow \mathbf{U}^T\,(\mathbf{A}_n - \mathbf{A}_m)\,\mathbf{U}$$

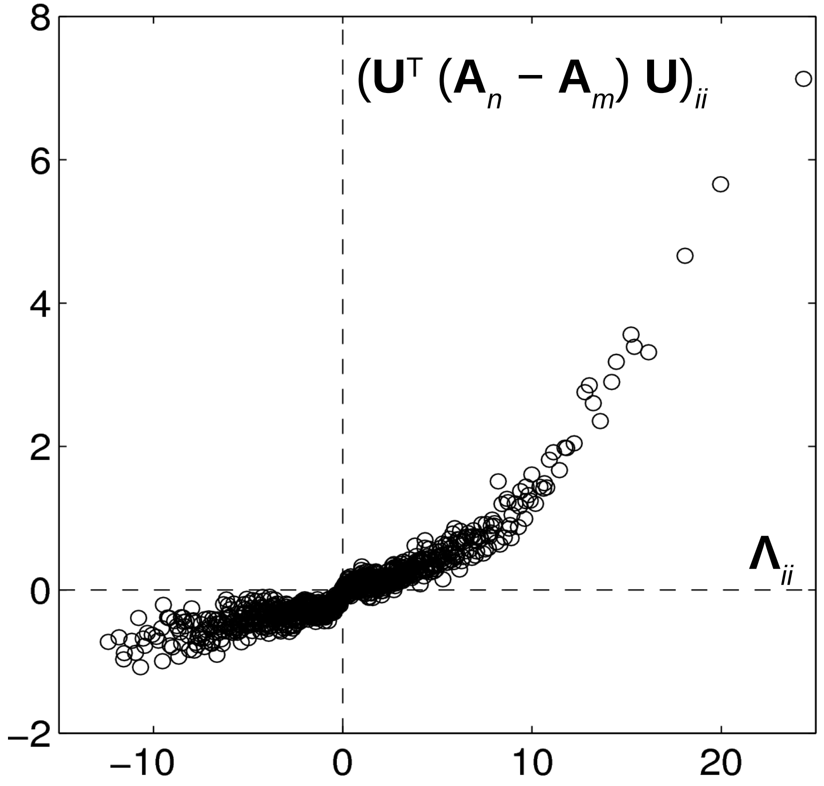


$\longrightarrow$



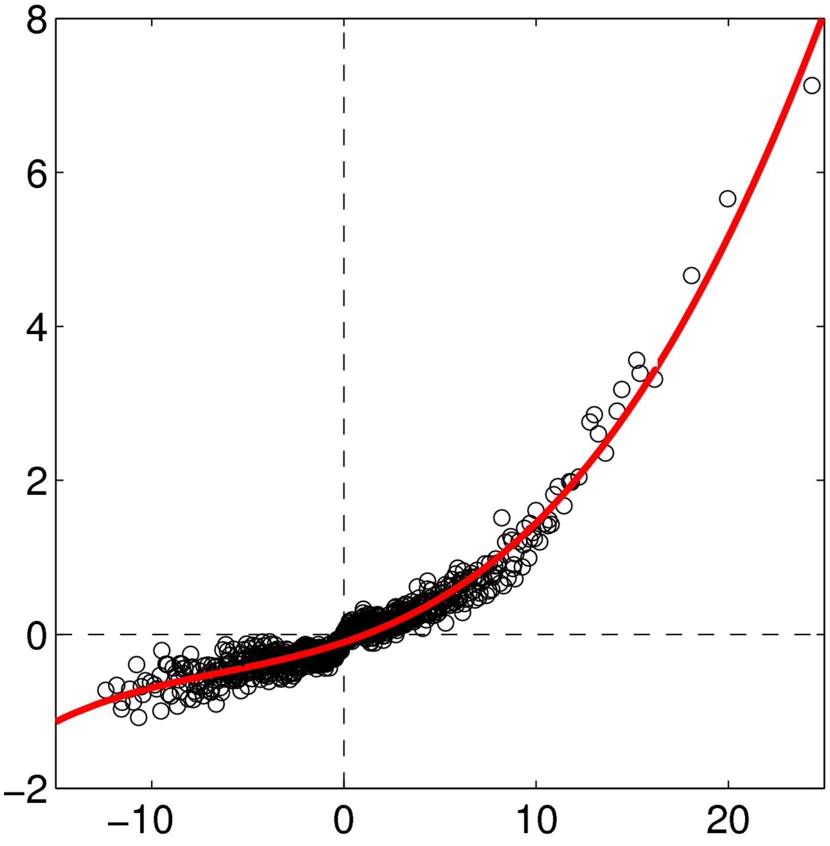
Jérôme Kunegis
kunegis@uni-koblenz.de

This is a one-dimensional curve-fitting problem.
Find f(x) such that $(\mathbf{U}^\mathrm{T}(\mathbf{A}_n - \mathbf{A}_m)\mathbf{U})_{ii} - \mathbf{\Lambda}_{ii}$ is small.

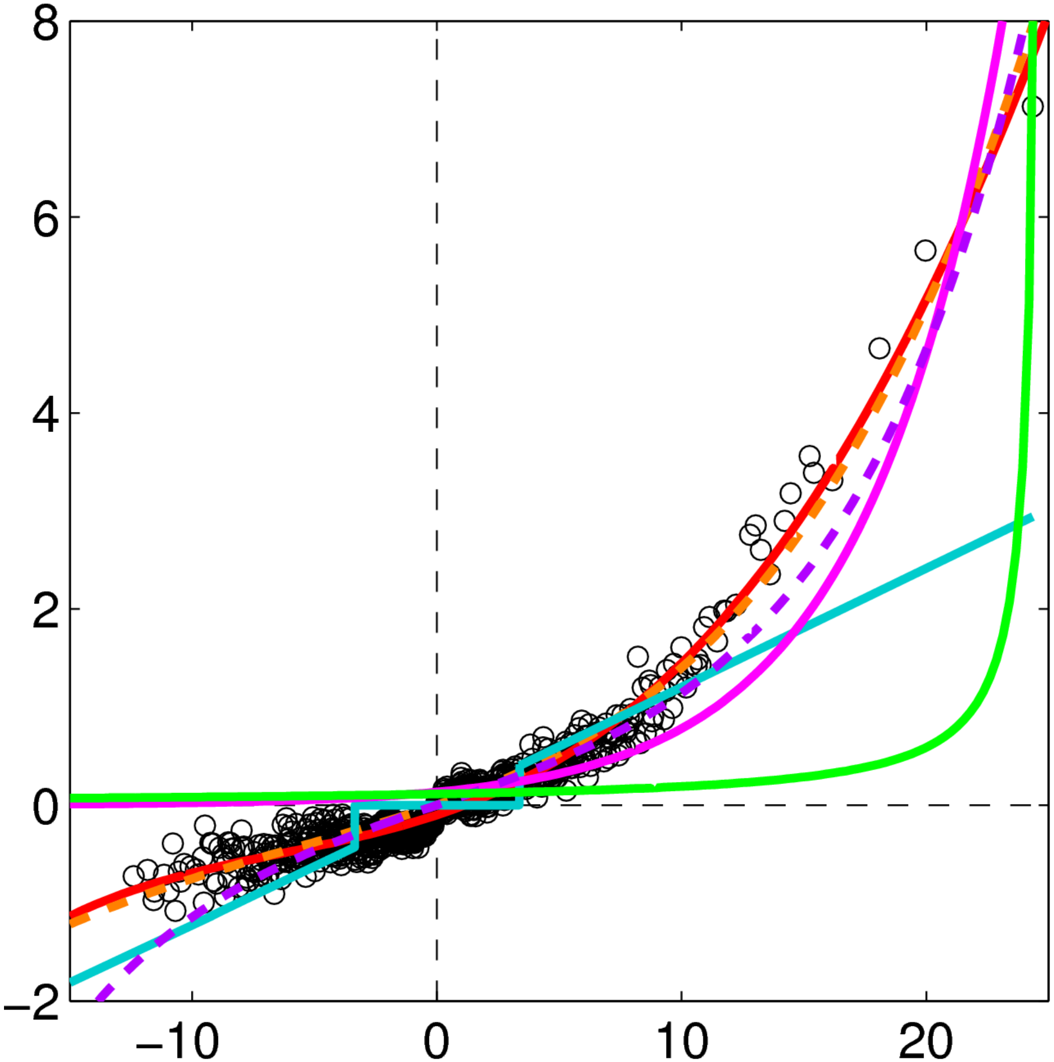


Polynomial Curve Fitting

Fit a polynomial $a + b\,x + c\,x^2 + d\,x^3 + e\,x^4$



Other Curves



Polynomial
$a + b\,x + c\,x^2 + d\,x^3 + e\,x^4$

Nonnegative Polynomial
$a + \ldots + h\,x^7 \qquad a, \ldots, h \geq 0$

Von Neumann kernel
$b\,/\,(1 - a\,x)$

Matrix exponential
$b\,\exp(a\,x)$

Hyperbolic sine
$b\,\sinh(a\,x)$

Rank reduction
$a\,x$ if $|x| \geq x_0$, 0 otherwise

WeST People and Knowledge Networks
Jérôme Kunegis
kunegis@uni-koblenz.de

UNIVERSITÄT KOBLENZ · LANDAU

Methodology:

- Split each dataset into three edge sets by formation date (training, validation, test)
- Learn the spectral transformation from training to validation set
- Predict edges in test set

Dataset	Best spectral transformation	Mean average precision (MAP)
Facebook friendship	Hyperbolic sine	0.667
Astro-ph coauthorship	Polynomial	0.690
DBLP coauthorship	Matrix exponential	0.759
Internet topology	Matrix exponential	0.726

Conclusion

Can the eigenvalue decomposition model the growth of networks?
Yes, because only eigenvalues change, not eigenvectors.

Is there an explanation for constant eigenvectors?
There are several: Graph kernels, triangle closing, the sum-over-paths model, rank reduction, etc.

Can we exploit this in recommender systems?
Yes, by learning the spectral transformation for a given dataset.

Is this scalable?
Yes, because we can reduce the learning problem to a one dimensional curve fitting problem.

References

J. Kunegis, A. Lommatzsch, Learning spectral graph transformations for link prediction. In Proc. Int. Conf. on Machine Learning, pp. 561–568, 2009.

J. Kunegis, D. Fay, C. Bauckhage, Network growth and the spectral evolution model. In Proc. Conf. on Information and Knowledge Management, pp. 739–748, 2010.

B. Viswanath, A. Mislove, M. Cha, K. P. Gummadi, On the evolution of user interaction in Facebook. In Proc. Workshop on Online Social Networks, pp. 37–42, 2009.

Backup: Other Findings

- **Evaluation on 114 network datasets**
- **Control tests**: Spectral evolution is not observed in random graph growth models
- **Link prediction by extrapolation of the spectrum**
- **Networks with positive and negative edges**: Powers of the adjacency matrix implement the *multiplication rule*
- **Directed networks**: Use the singular value decomposition
- **Bipartite networks**: Use the singular value decomposition
- **Spectral transformations of the Laplacian matrix**
- **The Laplacian matrix with negative edges**