On the Spectral Evolution of Large Networks

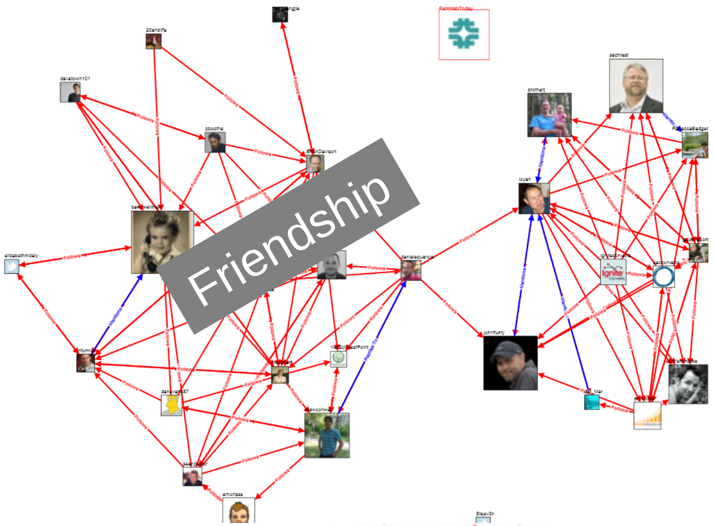
Jérôme Kunegis

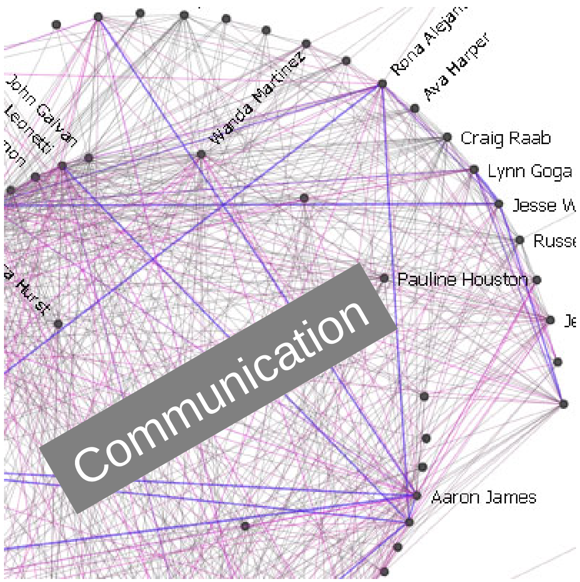
WeST
People and Knowledge Networks

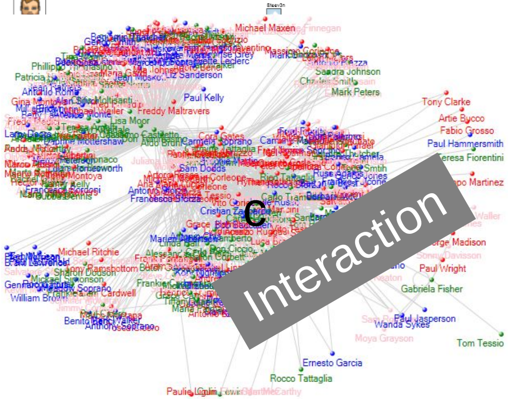
UNIVERSITÄT KOBLENZ · LANDAU

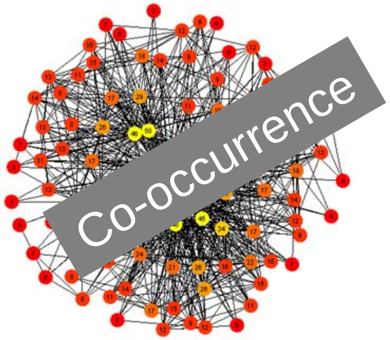
Networks

…are everywhere





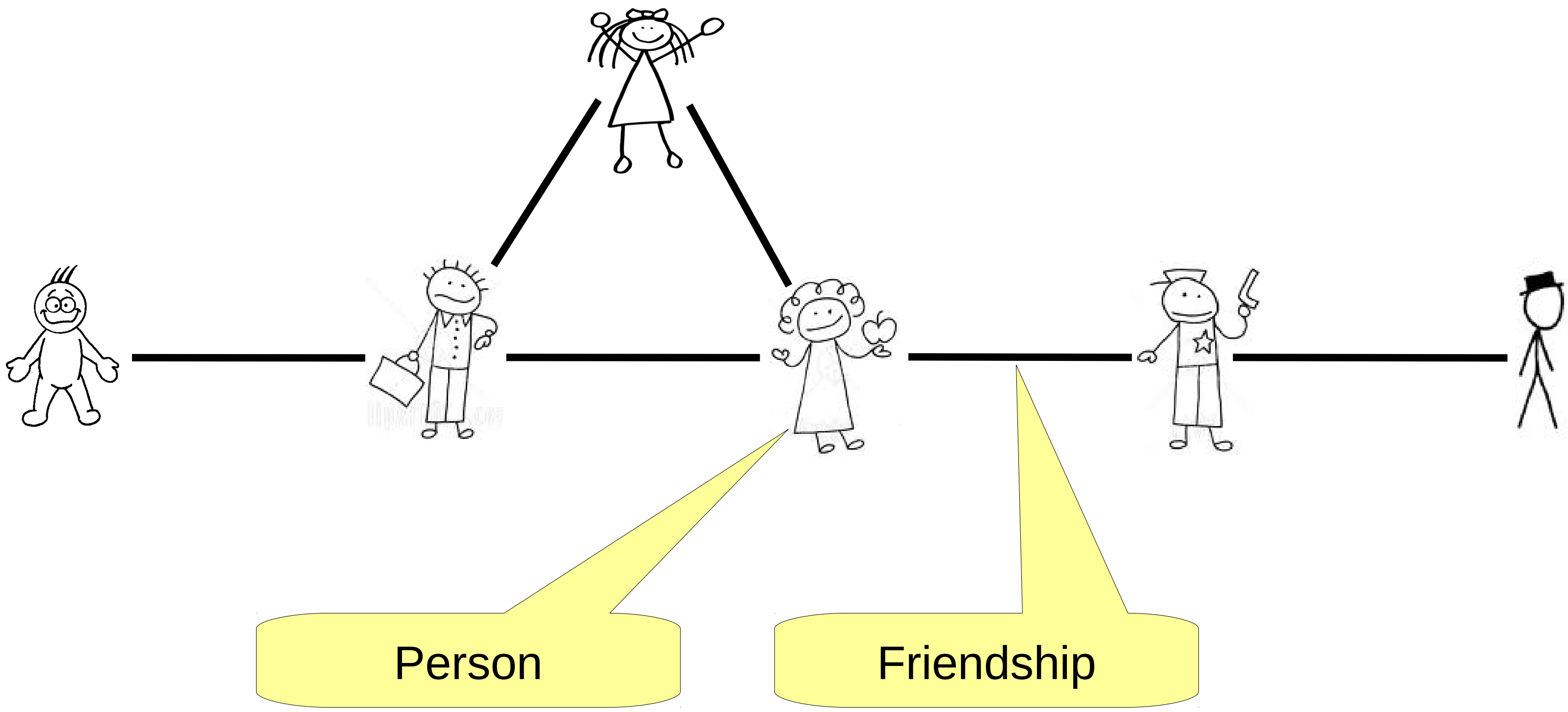




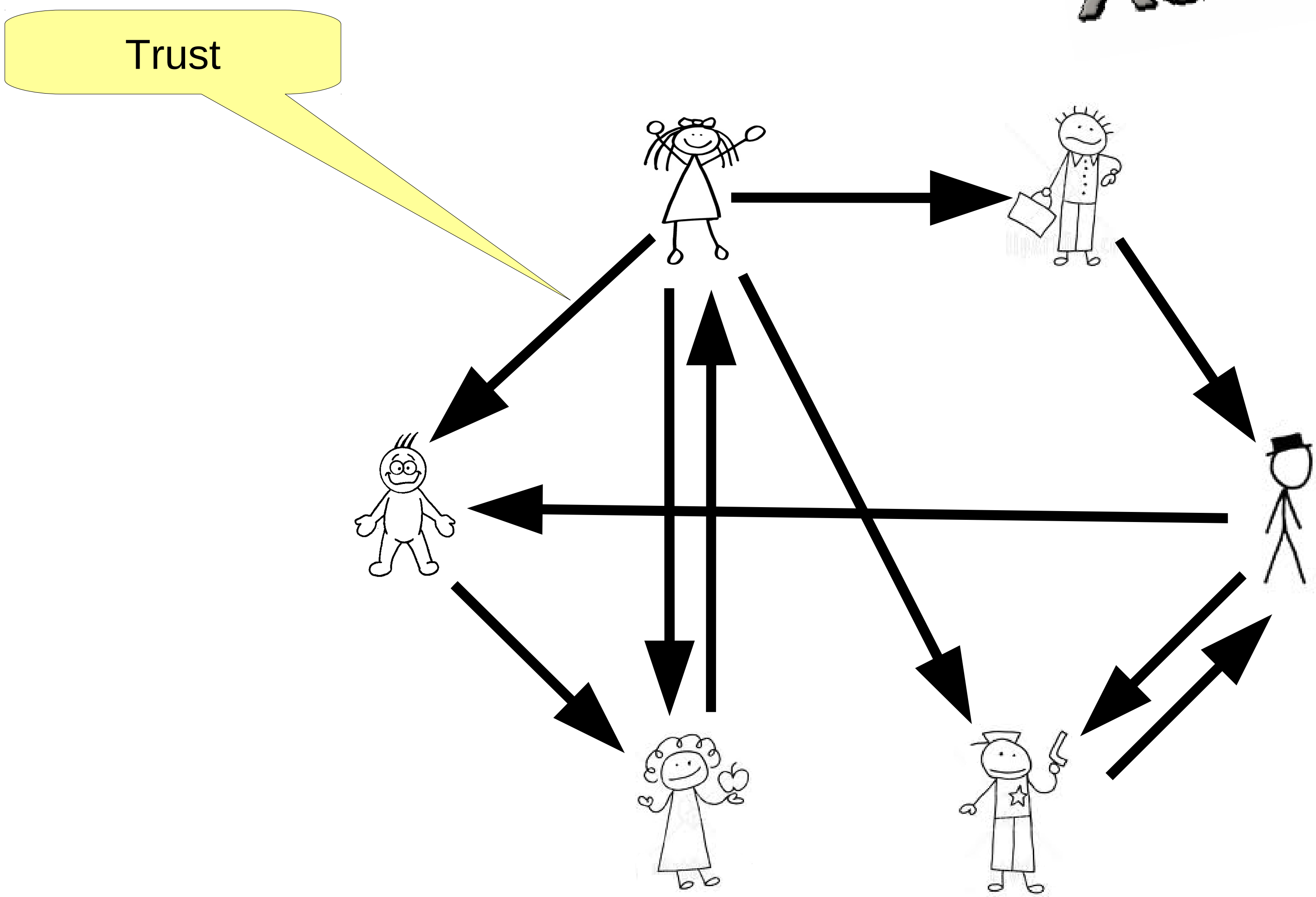
WeST People and Knowledge Networks
Jérôme Kunegis
kunegis@uni-koblenz.de

UNIVERSITÄT KOBLENZ · LANDAU

Social Network

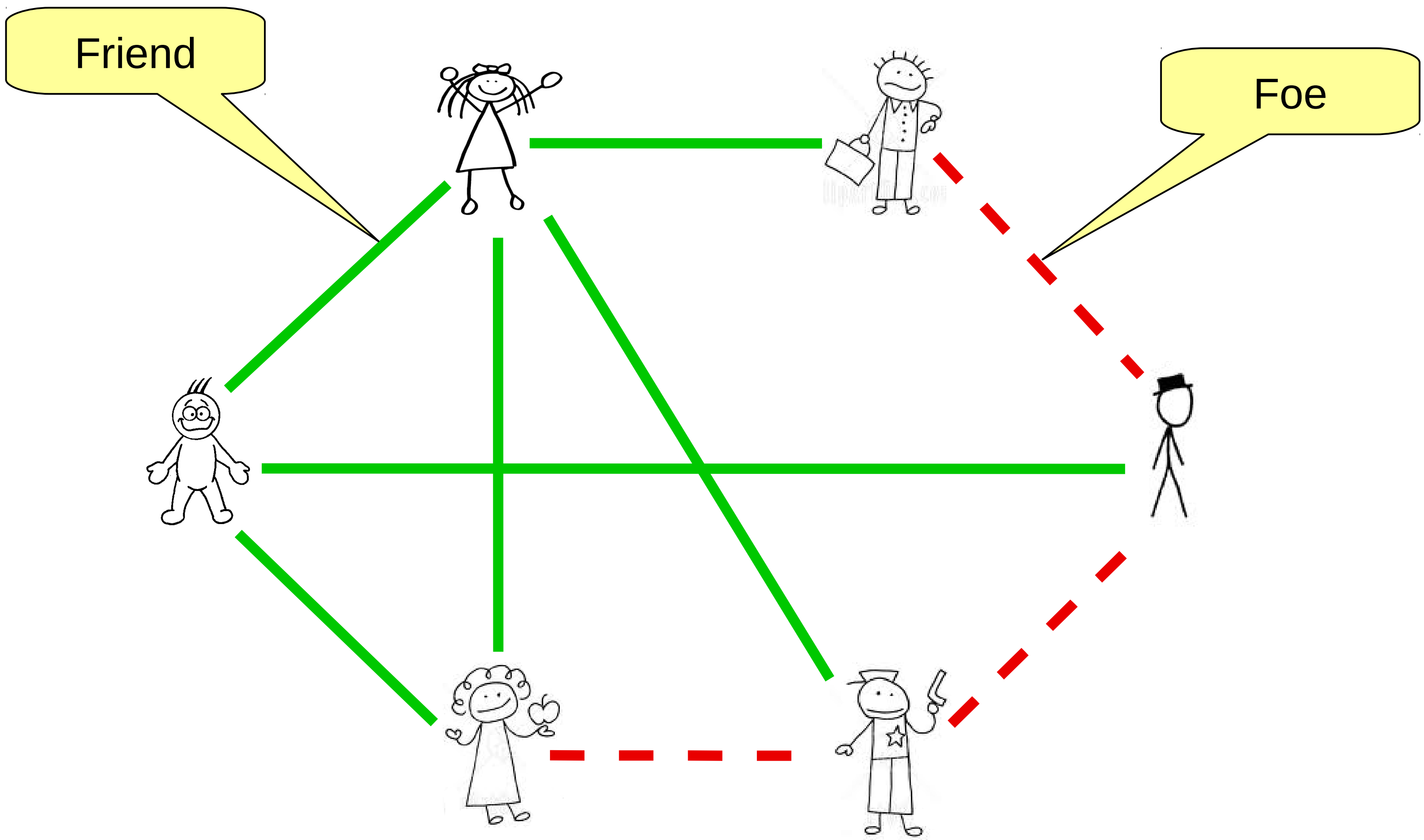


Trust Network

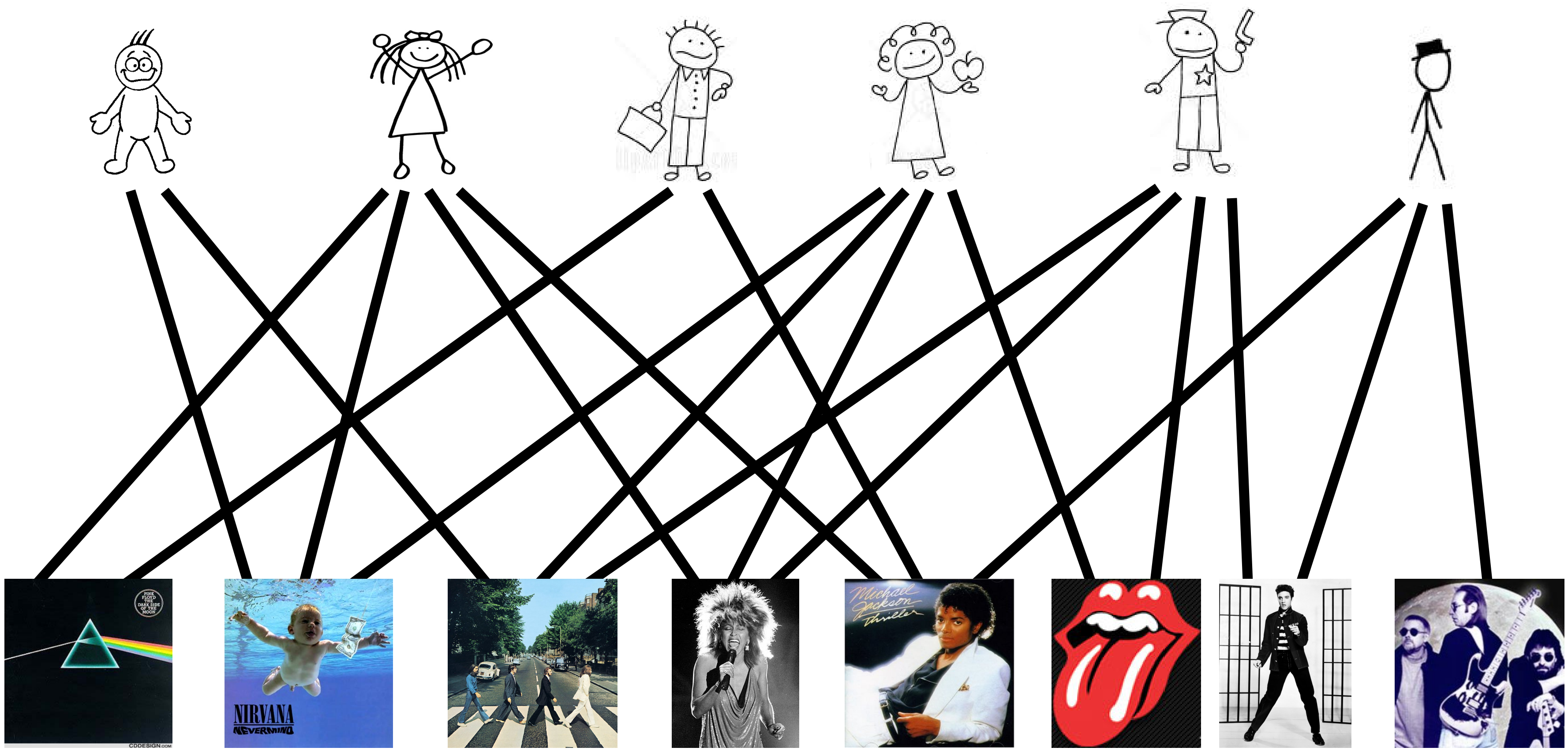


Signed Social Network

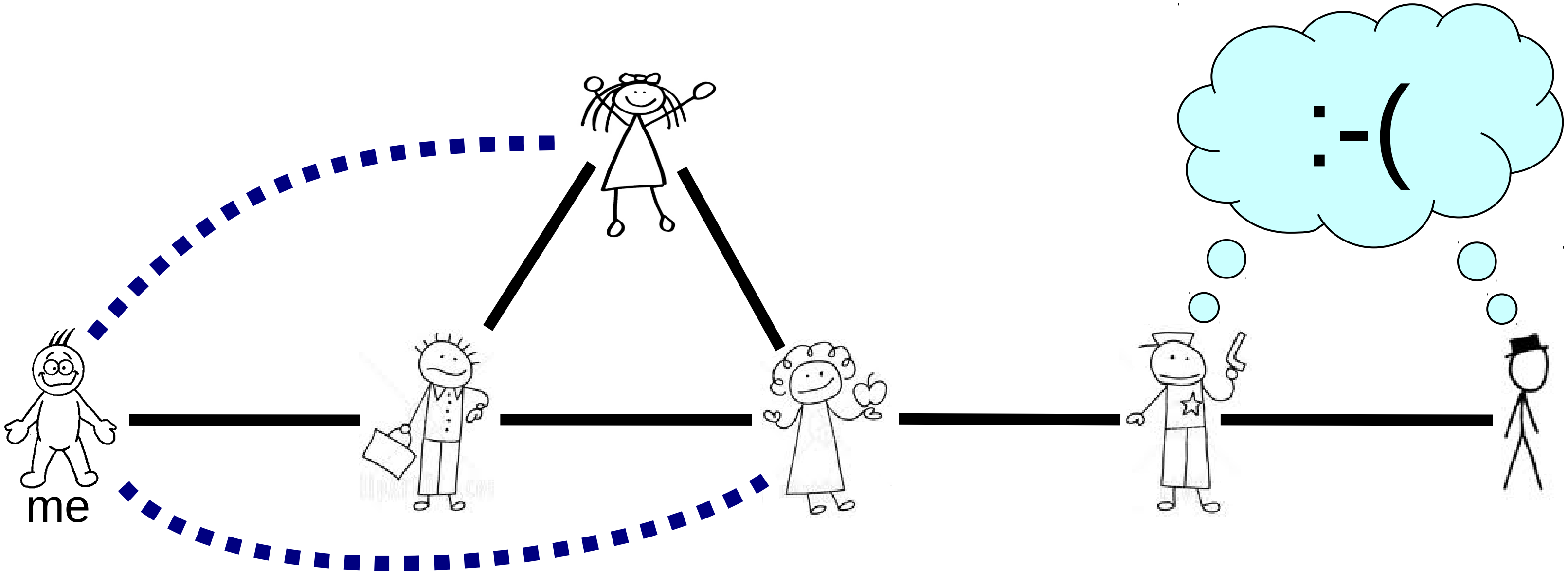
Slashdot



Interaction Network



Recommender Systems



Predict who I will add as friend next

Facebook's algorithm: find friends-of-friends

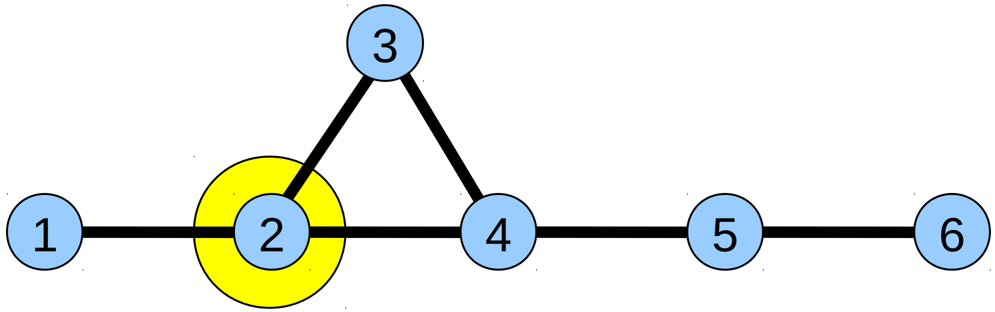
→ Problem: Rest of the network is ignored!

Outline

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

Take into account the whole network

Adjacency Matrix



$\mathbf{A}_{ij} = 1$ when i and j are connected

$\mathbf{A}_{ij} = 0$ when i and j are not connected

$\mathbf{A}$ is square and symmetric

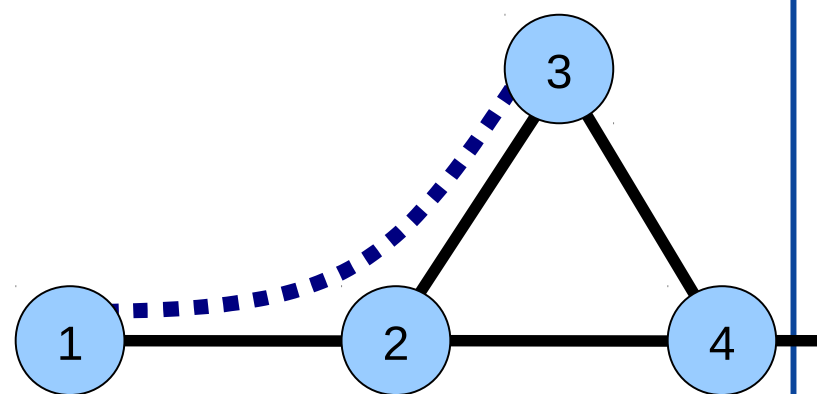
$$\mathbf{A} = \begin{array}{c|cccccc|} & 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 2 & 1 & 0 & 1 & 1 & 0 & 0 \\ 3 & 0 & 1 & 0 & 1 & 0 & 0 \\ 4 & 0 & 1 & 1 & 0 & 1 & 0 \\ 5 & 0 & 0 & 0 & 1 & 0 & 1 \\ 6 & 0 & 0 & 0 & 0 & 1 & 0 \end{array}$$

Baseline: Friend of a Friend Model

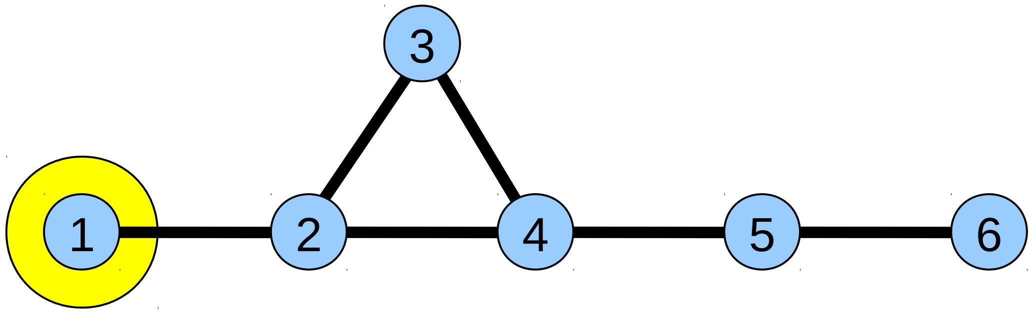
Count the number of ways a person can be found as the friend of a friend

Matrix product $\mathbf{A}\mathbf{A} = \mathbf{A}^2$

$$\mathbf{A}^2 = \begin{vmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{vmatrix}^2 = \begin{vmatrix} 1 & 0 & \boxed{1} & 1 & 0 & 0 \\ 0 & 3 & 1 & 1 & 1 & 0 \\ 1 & 1 & 2 & 1 & 1 & 0 \\ 1 & 1 & 1 & 3 & 0 & 1 \\ 0 & 1 & 1 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{vmatrix}$$



Friend of a Friend of a Friend



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

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

Problem: $\mathbf{A}^3$ is not sparse!

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

UNIVERSITÄT KOBLENZ · LANDAU

Eigenvalue Decomposition

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

where

$\mathbf{U}$ are the eigenvectors $\mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{I}$

$\mathbf{\Lambda}$ are the eigenvalues $\mathbf{\Lambda}_{ij} = 0$ when $i \neq j$

Jérôme Kunegis
kunegis@uni-koblenz.de

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

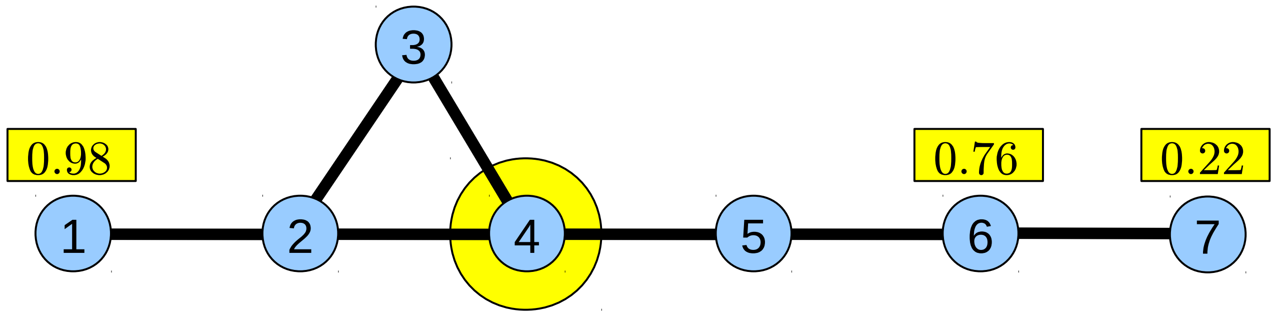
$$\mathbf{A}^3 = \mathbf{U}\mathbf{\Lambda}\boxed{\mathbf{U}^{\mathrm{T}}\,\mathbf{U}}\mathbf{\Lambda}\boxed{\mathbf{U}^{\mathrm{T}}\,\mathbf{U}}\mathbf{\Lambda}\mathbf{U}^{\mathrm{T}} = \mathbf{U}\boxed{\mathbf{\Lambda}^3}\mathbf{U}^{\mathrm{T}}$$

Exploit $\mathbf{U}$ and $\mathbf{\Lambda}$:

- $\boxed{\mathbf{U}^{\mathrm{T}}\mathbf{U}} = \mathbf{I}$ because $\mathbf{U}$ contains eigenvectors
- $(\boxed{\mathbf{\Lambda}^k})_{ii} = \mathbf{\Lambda}_{ii}{}^k$ because $\mathbf{\Lambda}$ contains eigenvalues

Result: Just cube all eigenvalues!

Matrix Exponential



$$\exp(\mathbf{A}) = \mathbf{I} + \mathbf{A} + 1/2\ \mathbf{A}^2 + 1/6\ \mathbf{A}^3 + \ldots$$

$$\exp\begin{vmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{vmatrix} = \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \begin{vmatrix} 1.66 & 1.72 & 0.93 & 0.98 & 0.28 & 0.06 & 0.01 \\ 1.72 & 3.57 & 2.70 & 2.93 & 1.04 & 0.29 & 0.06 \\ 0.93 & 2.70 & 2.86 & 2.71 & 0.99 & 0.28 & 0.06 \\ 0.98 & 2.93 & 2.71 & 3.63 & 1.95 & 0.76 & 0.22 \\ 0.28 & 1.04 & 0.99 & 1.95 & 2.35 & 1.59 & 0.64 \\ 0.06 & 0.29 & 0.28 & 0.76 & 1.59 & 2.23 & 1.38 \\ 0.01 & 0.06 & 0.06 & 0.22 & 0.64 & 1.38 & 1.59 \end{vmatrix} \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array}$$

Jérôme Kunegis
kunegis@uni-koblenz.de

Spectral Transformations

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

Friend of a friend

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

Friend of a friend of a friend

$$\exp(\mathbf{A}) = \mathbf{U}\boxed{\exp(\mathbf{\Lambda})}\mathbf{U}^\mathrm{T}$$

Matrix exponential

…are link prediction functions!

Jérôme Kunegis
kunegis@uni-koblenz.de

Outline

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

Why does it work?

Looking at Real Facebook Data

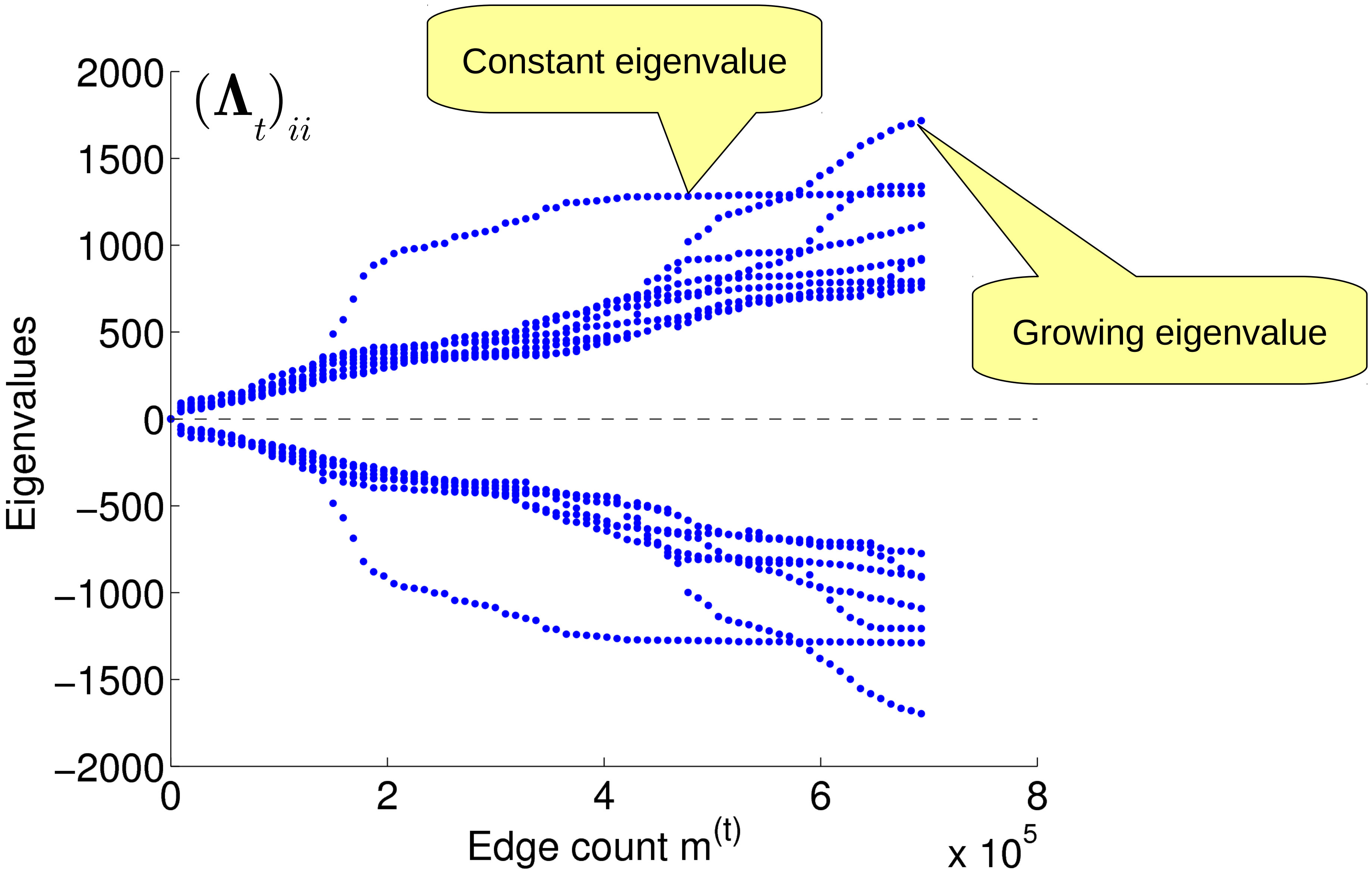
Dataset: Facebook New Orleans (Viswanath et al. 2009)

$63{,}731$ persons
$1{,}545{,}686$ friendship links with **creation dates**

Adjacency matrix $\mathbf{A}_t$ at time t $(t = 1 \ldots 75)$

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

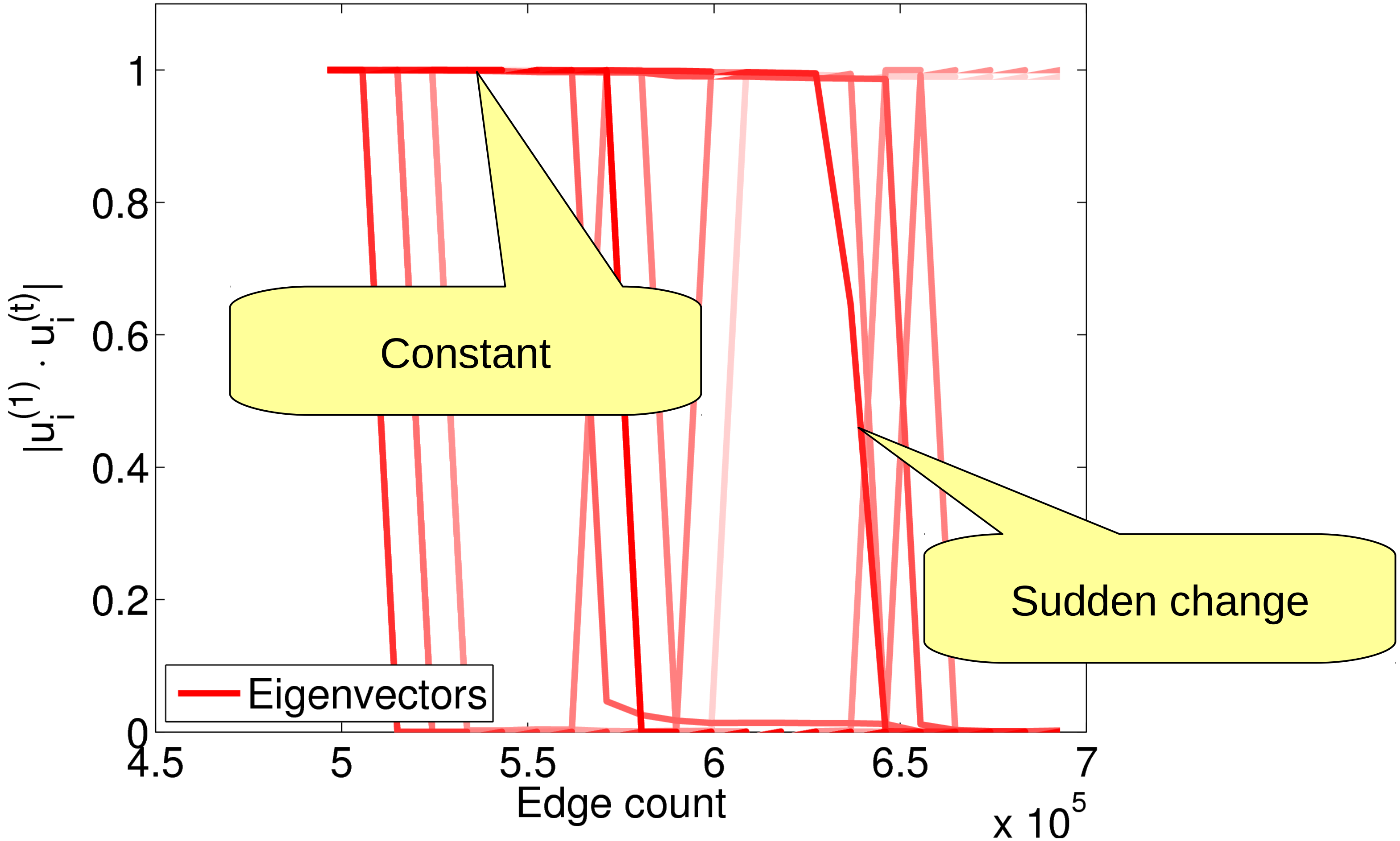
Evolution of Eigenvalues



Jérôme Kunegis
kunegis@uni-koblenz.de

Eigenvector Evolution

Cosine similarity between $(\mathbf{U}_t)_{\bullet i}$ and $(\mathbf{U}_{t+x})_{\bullet i}$



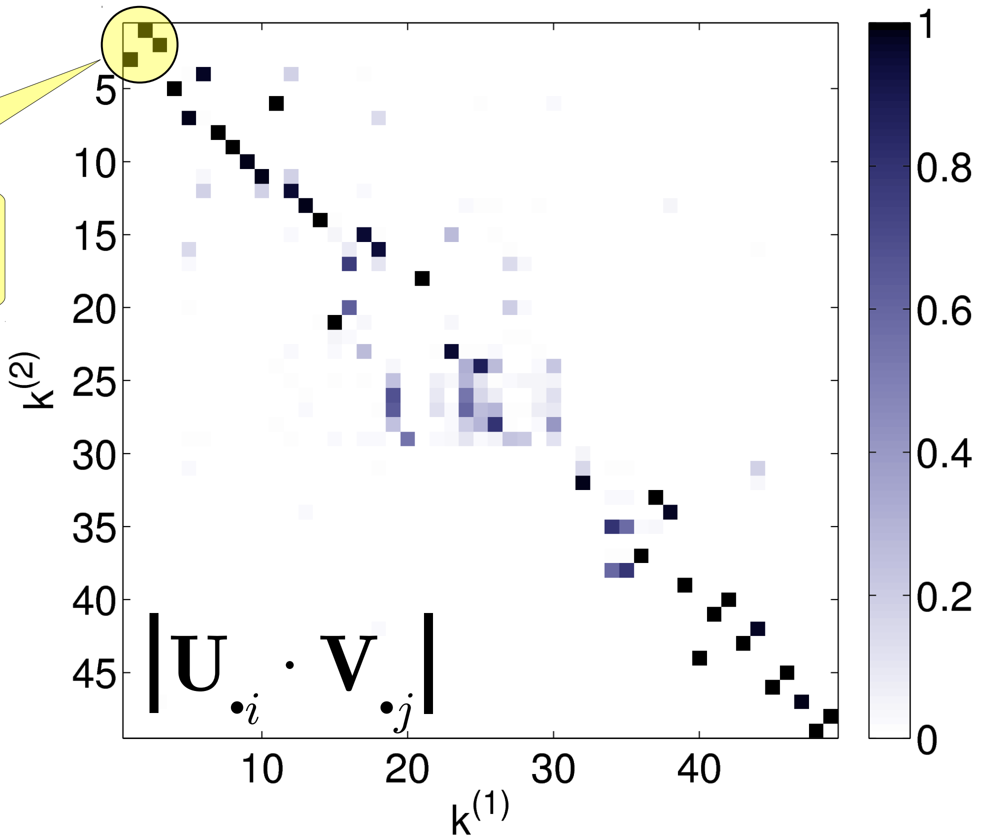
Jérôme Kunegis
kunegis@uni-koblenz.de

Eigenvector Permutation

Time split: old edges $\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathrm{T}}$

new edges $\mathbf{B} = \mathbf{V}\mathbf{D}\mathbf{V}^{\mathrm{T}}$

Eigenvectors permute

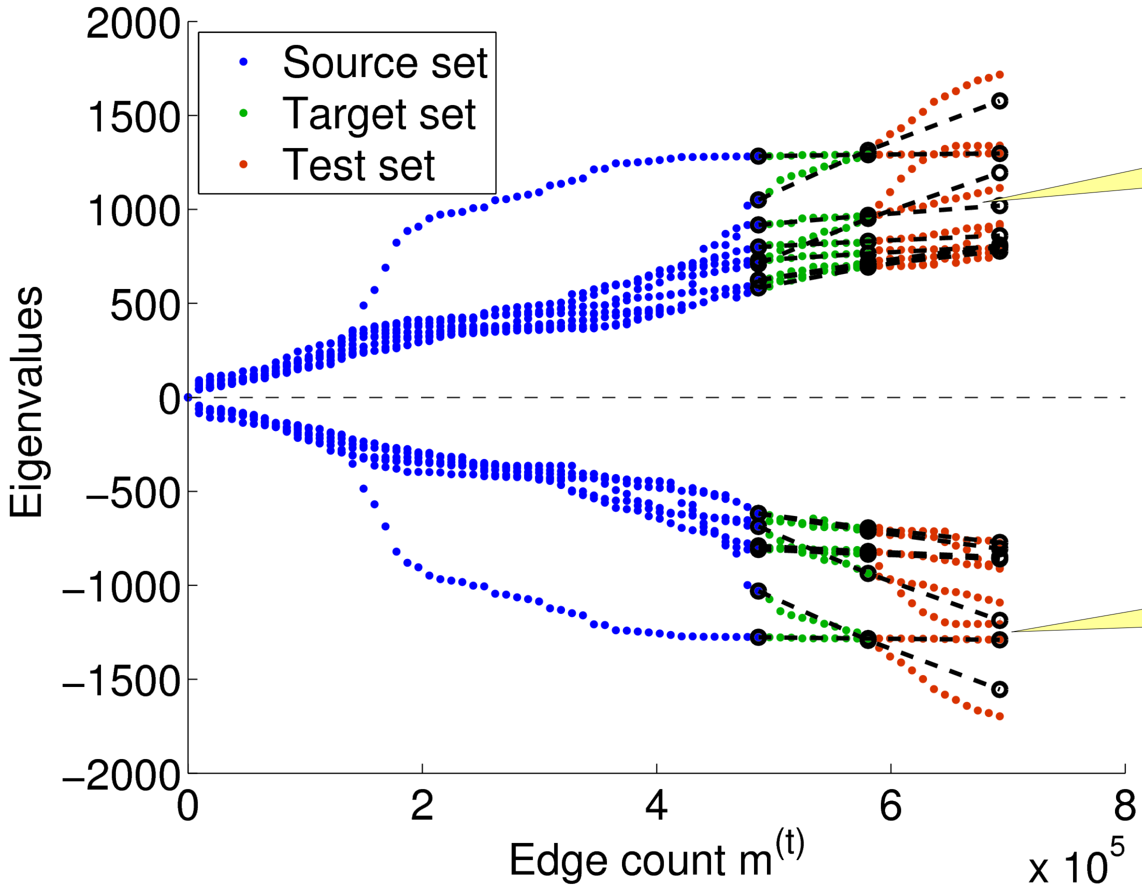


Jérôme Kunegis
kunegis@uni-koblenz.de

1. Algebraic Link Prediction
2. Spectral Transformations
3. **Learning Link Prediction**
   a) Learning by Extrapolation
   b) Learning by Curve Fitting

What spectral transformation is best?

Extrapolate the growth of the spectrum



Good when growth is irregular

Potential problem: overfitting

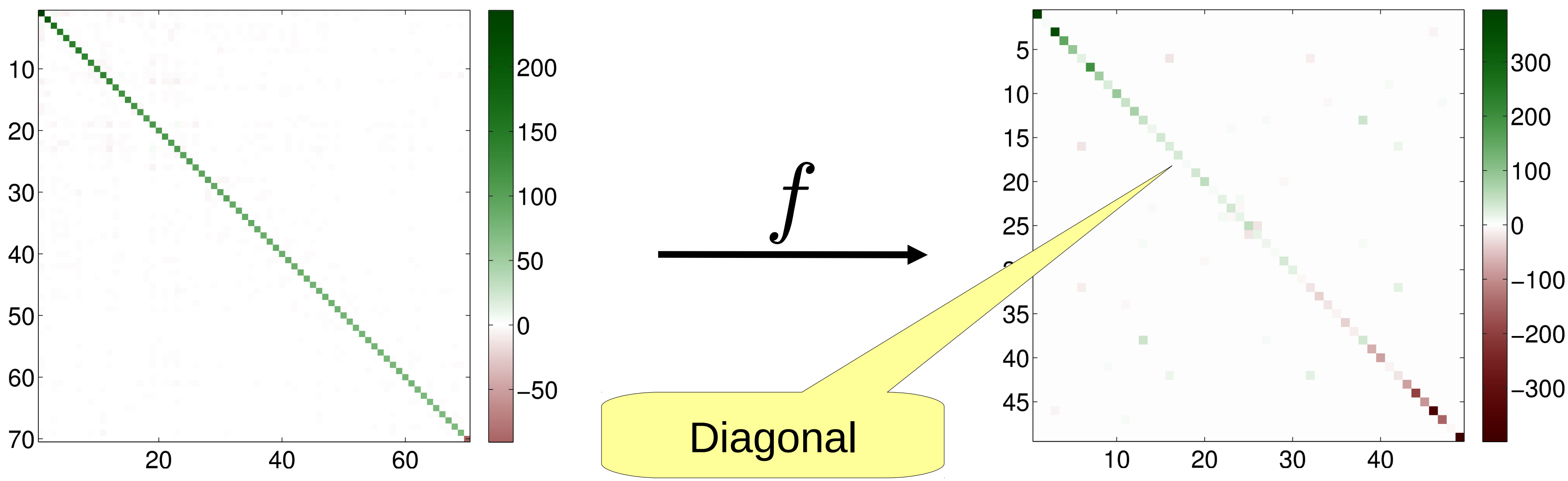
Jérôme Kunegis
kunegis@uni-koblenz.de

b) Learning by Curve Fitting

$$\mathbf{A} \xrightarrow{\quad f \quad} \mathbf{B}$$

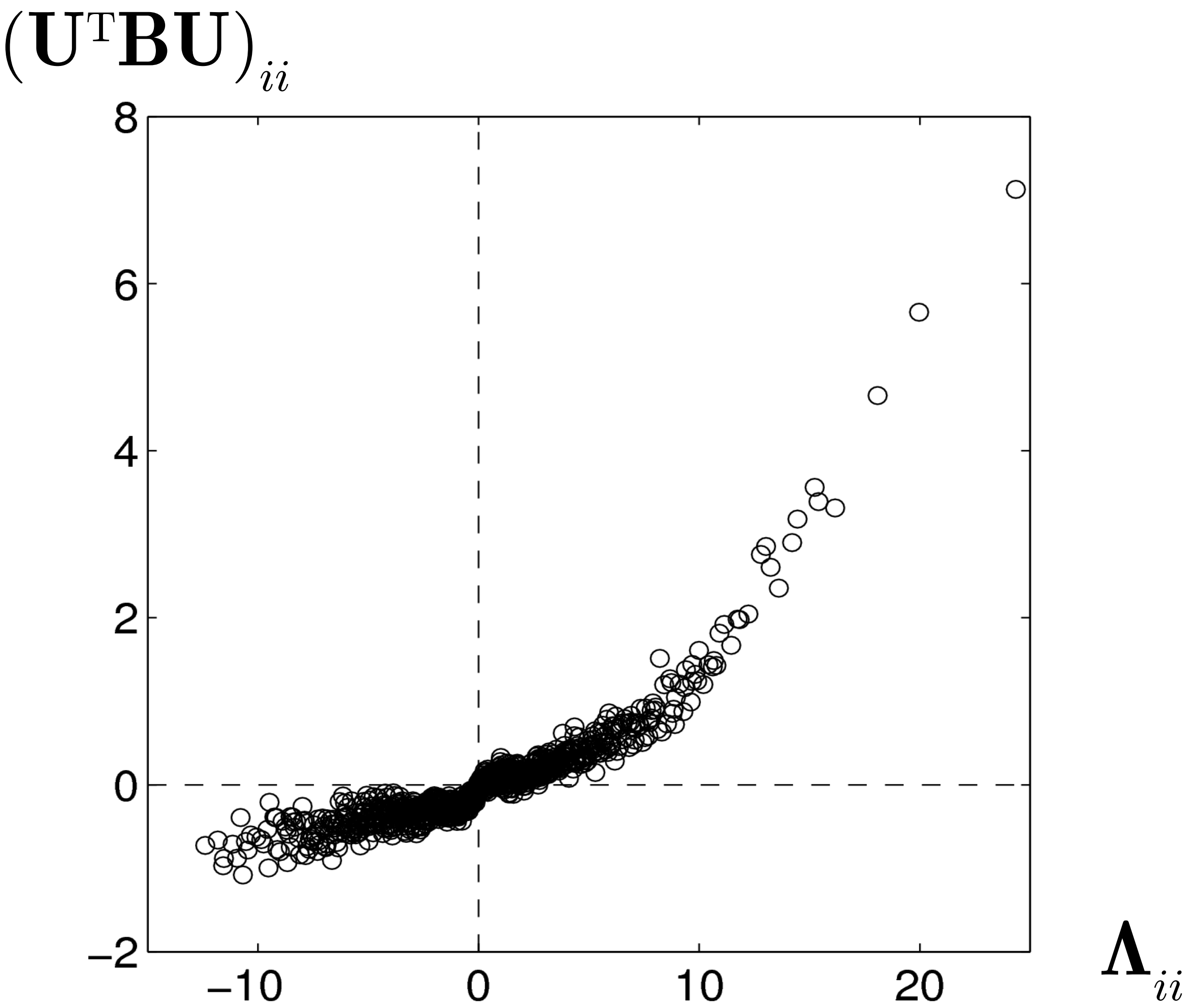
$$\mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathrm{T}} \longrightarrow \mathbf{B}$$

$$\mathbf{\Lambda} \longrightarrow \boxed{\mathbf{U}^{\mathrm{T}}}\mathbf{B}\boxed{\mathbf{U}}$$



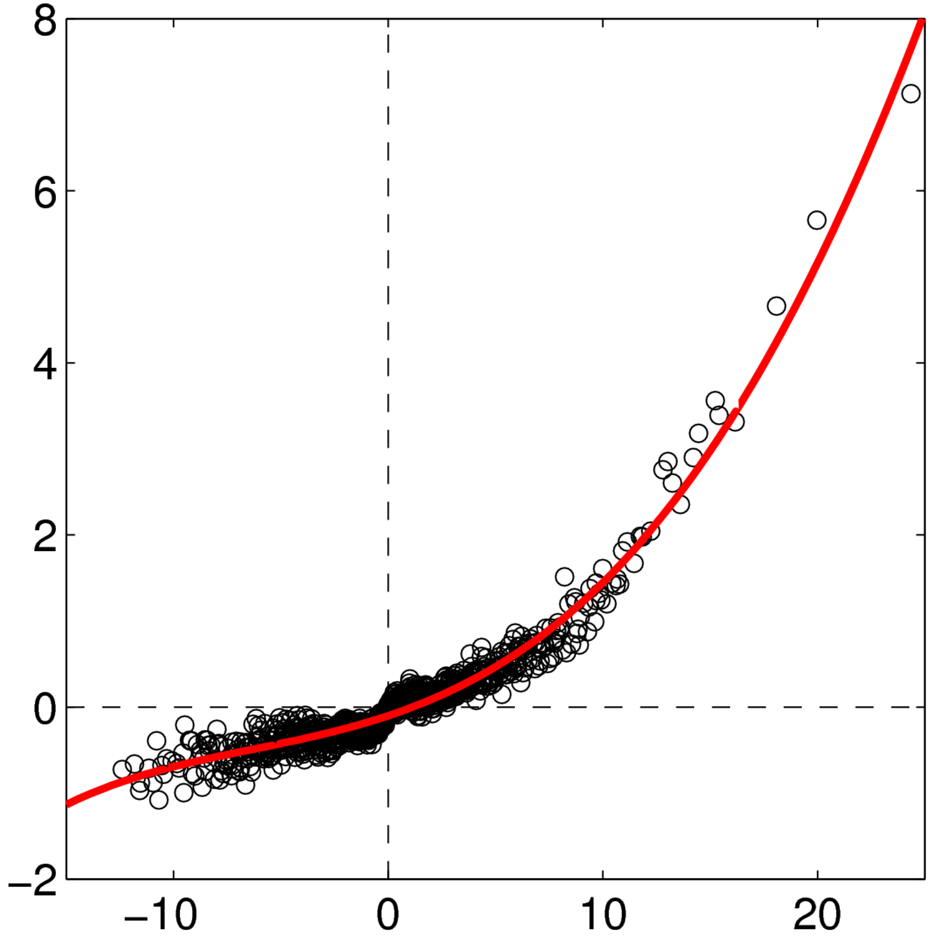
Jérôme Kunegis
kunegis@uni-koblenz.de

Curve Fitting

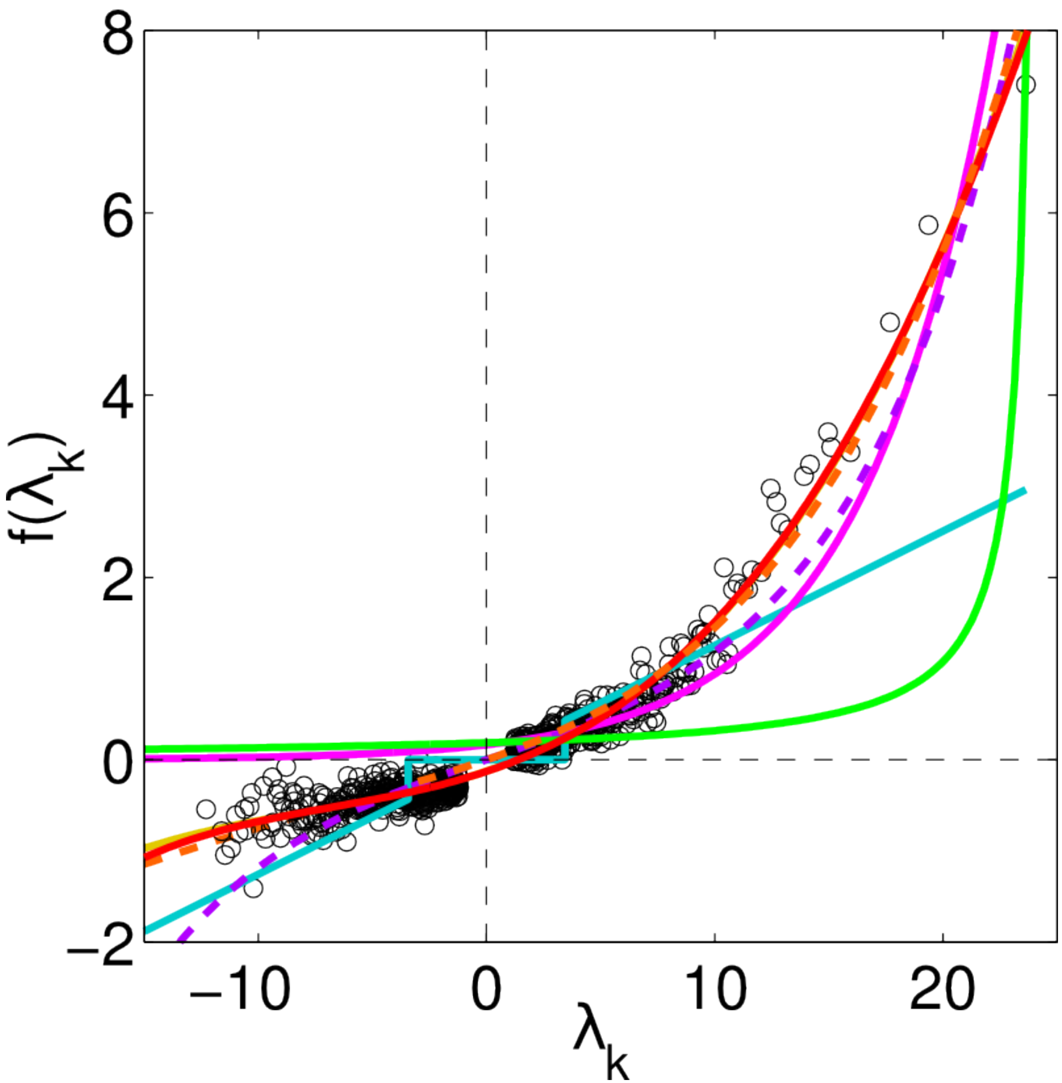


Polynomial Curve Fitting

Fit a polynomial $$a + bx + cx^2 + dx^3 + ex^4$$



Other Curves



Friend of a Friend

$a + bx + cx^2$

Polynomial

$a + bx + cx^2 + dx^3 + ex^4$

Nonnegative polynomial

$a + \ldots + hx^7 \quad a, \ldots, h \geq 0$

Matrix exponential

$b \exp(ax)$

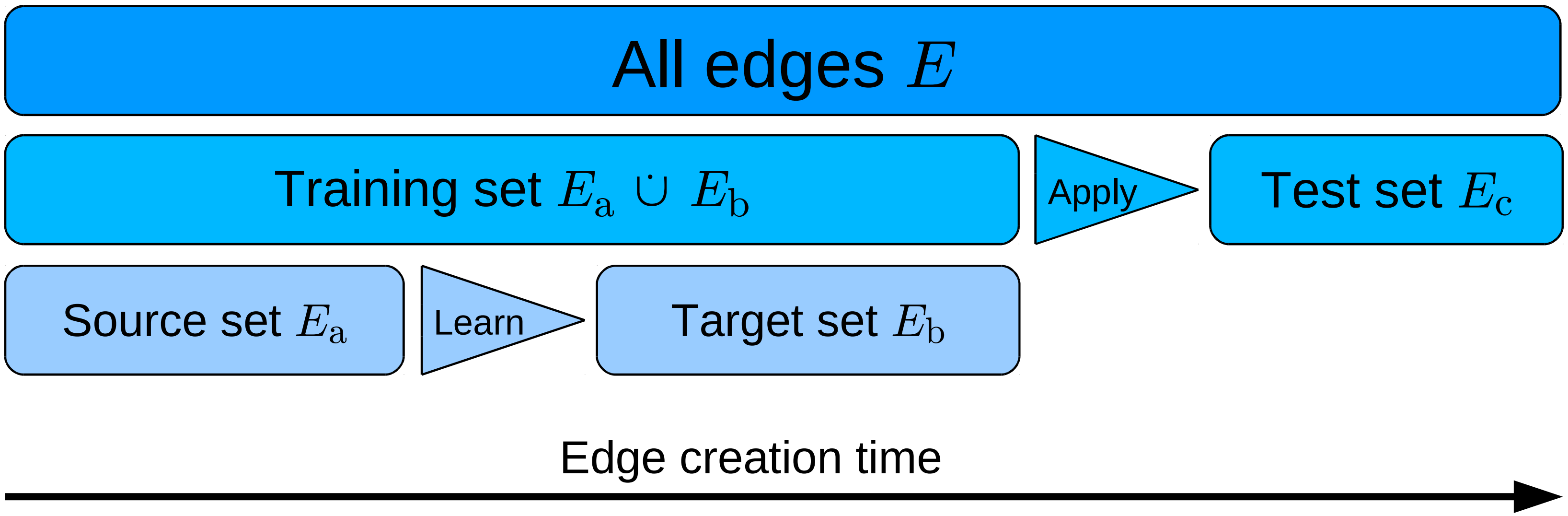
Neumann kernel

$b \,/\, (1 - ax)$

Rank reduction

ax if $|x| \geq x_0$, 0 otherwise

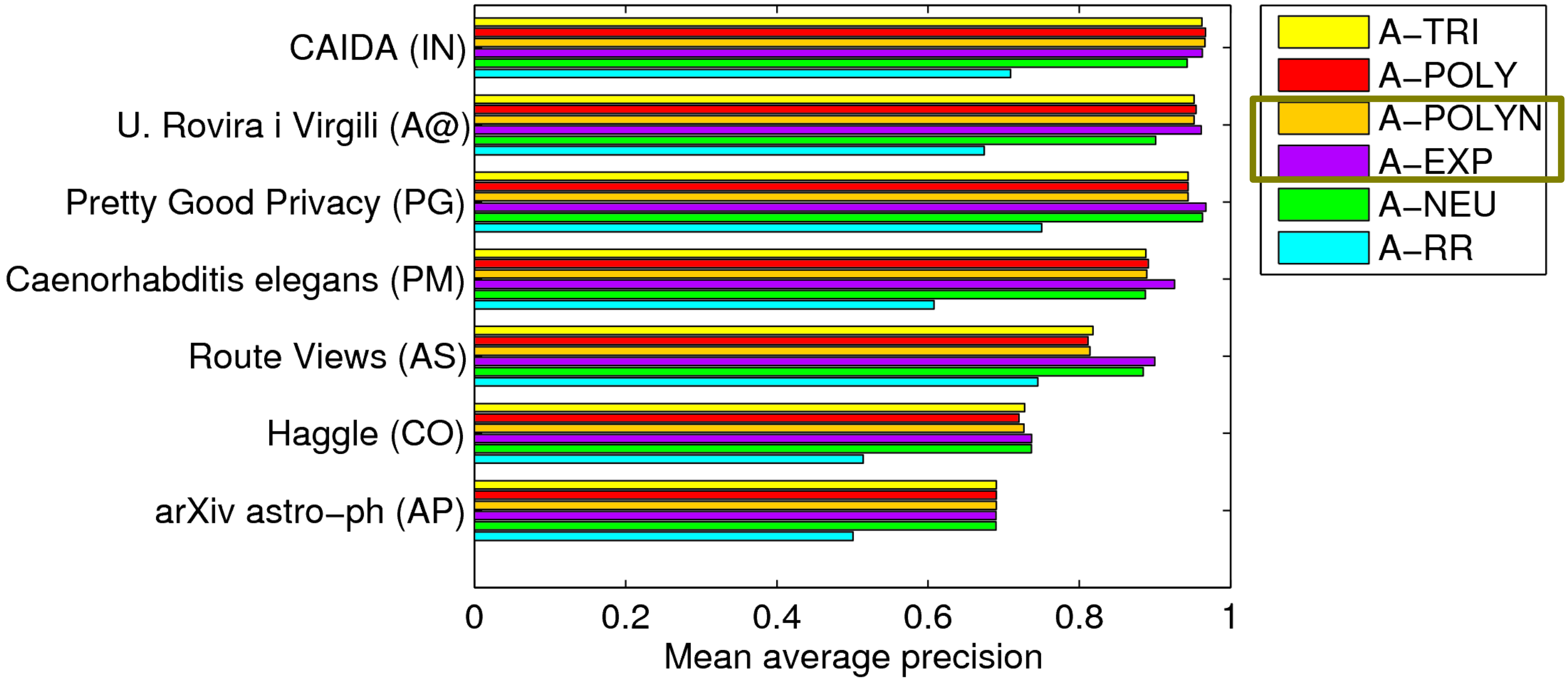
Evaluation Methodology



3-way split of edge set by edge creation time

Jérôme Kunegis
kunegis@uni-koblenz.de

Precision of link prediction (1 = perfect)



All datasets available at konect.uni-koblenz.de

Conclusion

- **Observation**
  - Eigenvalue change, eigenvectors are constant
- **Why?**
  - Graph kernels, triangle closing, the sum-over-paths model, rank reduction, etc.
- **Application to recommender systems**
  - By learning the spectral transformation for a given dataset

ACKNOWLEDGMENTS →

Thank You!

Selected Publications

The Slashdot Zoo: Mining a social network with negative edges
J. Kunegis, A. Lommatzsch and C. Bauckhage
In Proc. World Wide Web Conf., pp. 741–750, 2009.

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

Spectral analysis of signed graphs for clustering, prediction and visualization
J. Kunegis, S. Schmidt, A. Lommatzsch and J. Lerner
In Proc. SIAM Int. Conf. on Data Mining, pp. 559–570, 2010.

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

References

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.