Carnegie Mellon

Large Graph Mining – Patterns and Tools

Christos Faloutsos

CMU

Carnegie Mellon

Thank you!

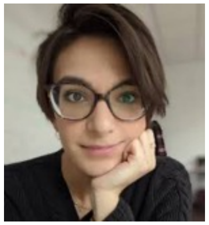
- Puja Das

Thank you!

- Meng-Chieh Jeremy Lee (CMU)

- Robson Cordeiro (CMU)

- Catalina Vajiac (CMU)



- Karish Grover (CMU)

Slides for semester course

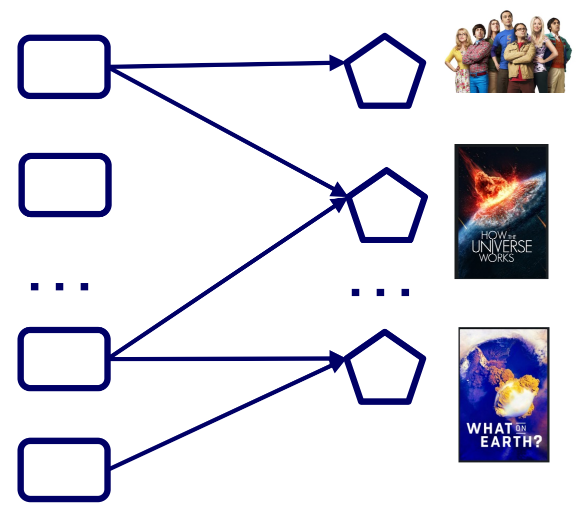
- Fractals and power laws (4 lectures)
- Text mining
- **Matrices, SVD and tensors** (5 lectures)
- **Graph mining** (6 lectures)
- Time series, Fourier, wavelets, & forecasting (4 lectures)
- https://www.cs.cmu.edu/~christos/courses/989.F23/schedule.html

Roadmap

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
- Part#2: Graph Mining – (semi-)supervised
- Part#3: Time-evolving graphs
- Part#4: Explanations
- Conclusions

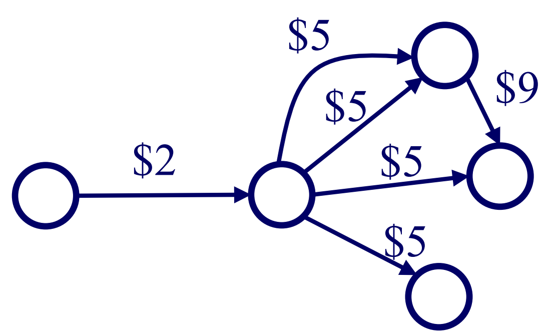
Graphs – why should we care?

Customers Movies



(source, destination, timestamp, duration)

Graphs – why should we care?



(source, destination, timestamp, $amount)

Graphs - why should we care?

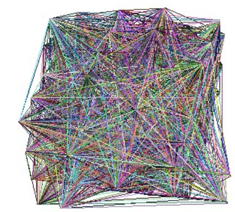
>$10B; ~1B users

Graphs - why should we care?

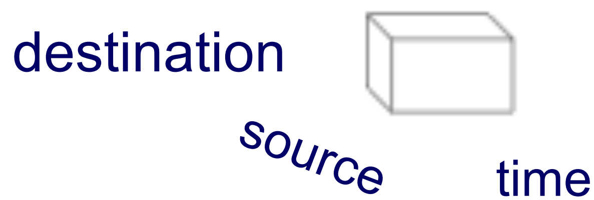
- web-log ('blog') news propagation
- computer network security: email/IP traffic and anomaly/intrusion detection
- Recommendation systems
- ....

- Many-to-many db relationship -> graph

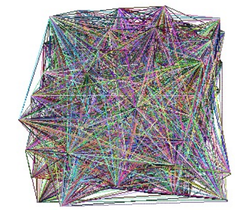
Motivating problems



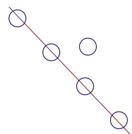
- P1: patterns? Fraud detection?
- P2: Propagation
- P3: patterns in time-evolving graphs / tensors



Motivating problems



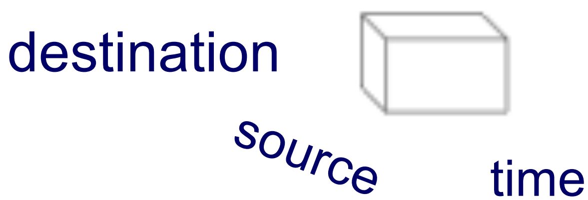
- P1: patterns? Fraud detection?

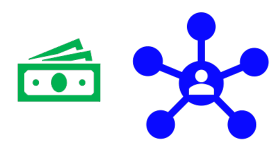


Patterns

anomalies

- P2: Propagation
- P3: patterns in time-evolving graphs / tensors





‘Recipe’ Structure:

- Problem definition
- Short answer/solution
- LONG answer – details
- Conclusion/short-answer

Roadmap

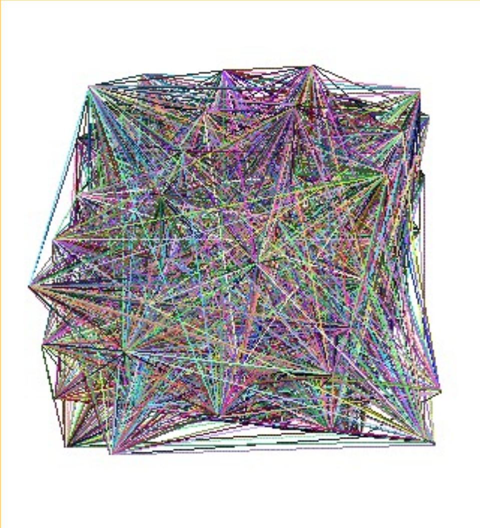
- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
- Part#2: Graph Mining – (semi-)supervised
- Part#3: Time-evolving graphs
- Part#4: Explanations
- Conclusions

Roadmap (detailed)

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
  - 1.1 Patterns
  - 1.2 Anomalies
  - 1.3 Money laundering detection
- Part#2: Graph Mining – (semi-)supervised
- …

Problem

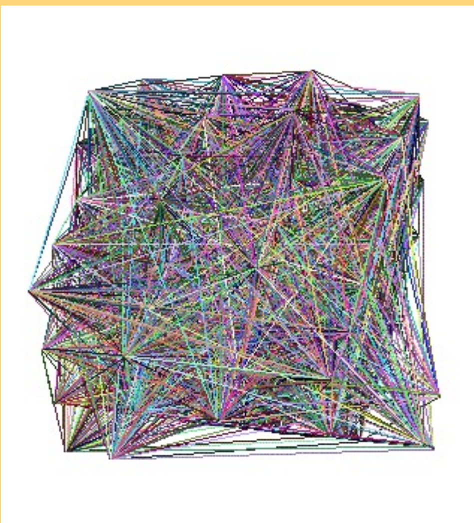
Given:



Find patterns ('what is normal')

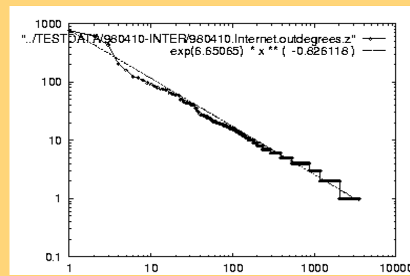
Solution(s)

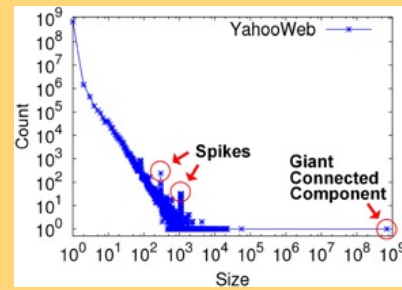
Given:

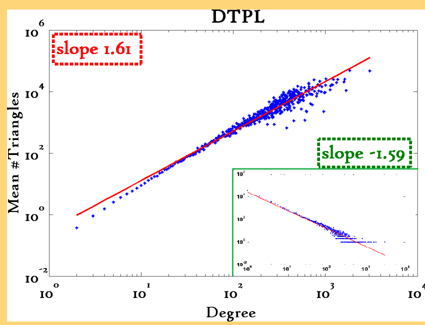


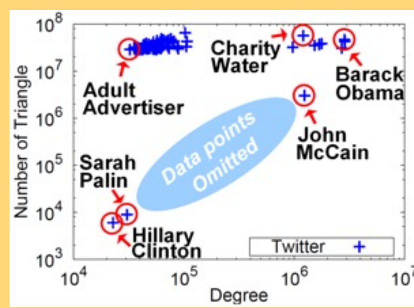
Find patterns ('what is normal')

6-degrees



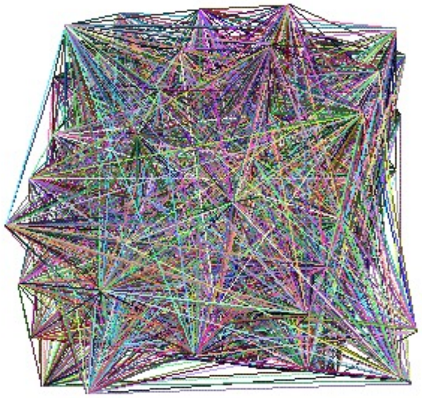




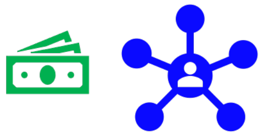


...

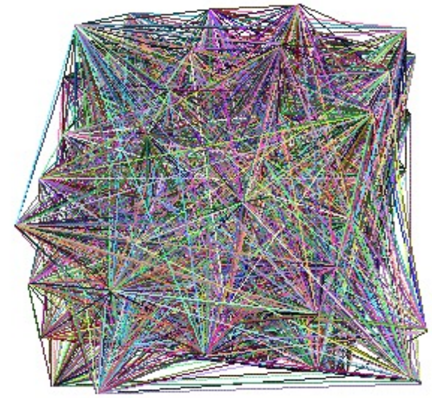
Laws and patterns



- Q1: Are real graphs random?



Laws and patterns

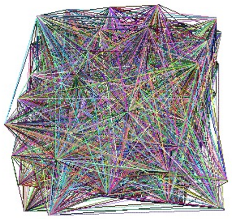


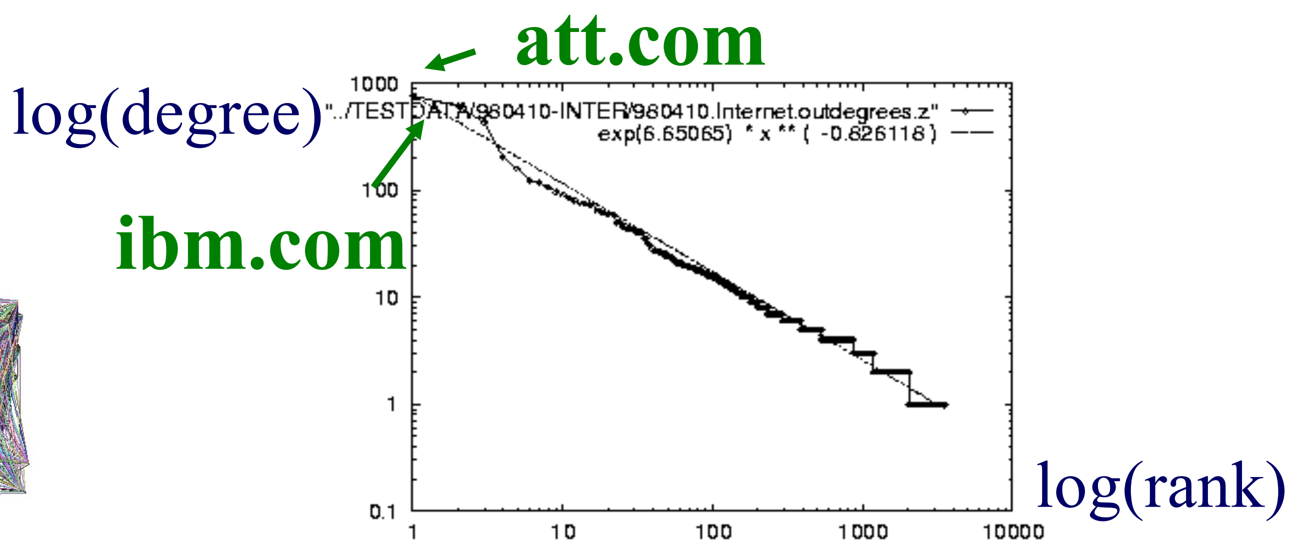
- Q1: Are real graphs random?
- A1: NO!!
  - Diameter ('6 degrees'; 'Kevin Bacon')
  - in- and out- degree distributions
  - other (surprising) patterns
- So, let's look at the data

Solution# S.1

- Power law in the degree distribution [Faloutsos x 3 SIGCOMM99]

internet domains

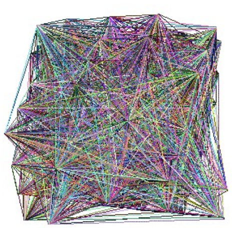


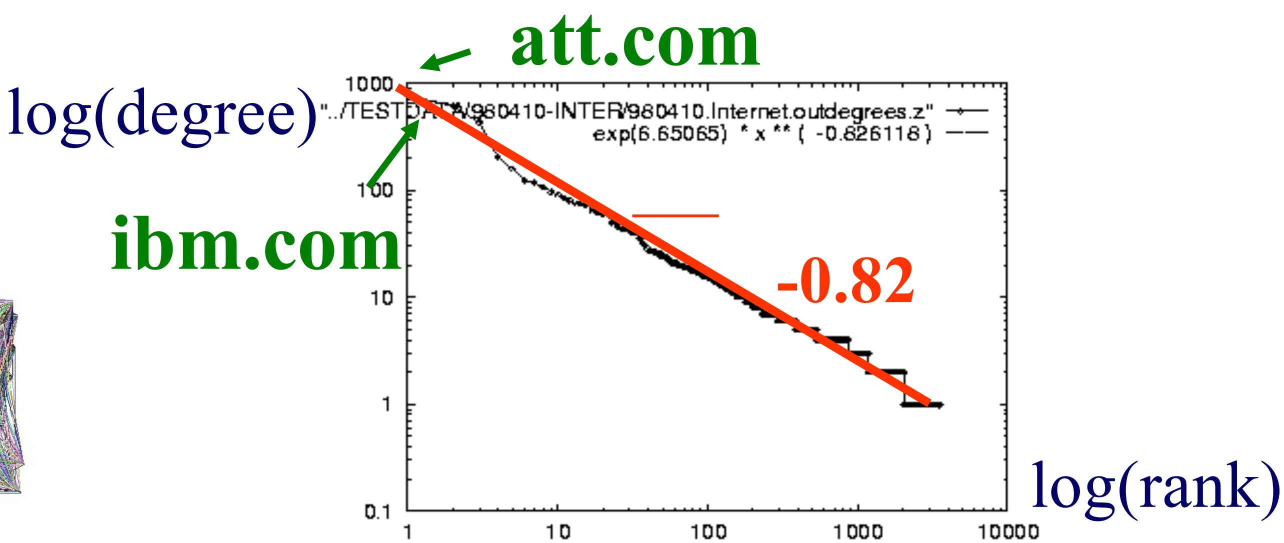


Solution# S.1

- Power law in the degree distribution [Faloutsos x 3 SIGCOMM99]

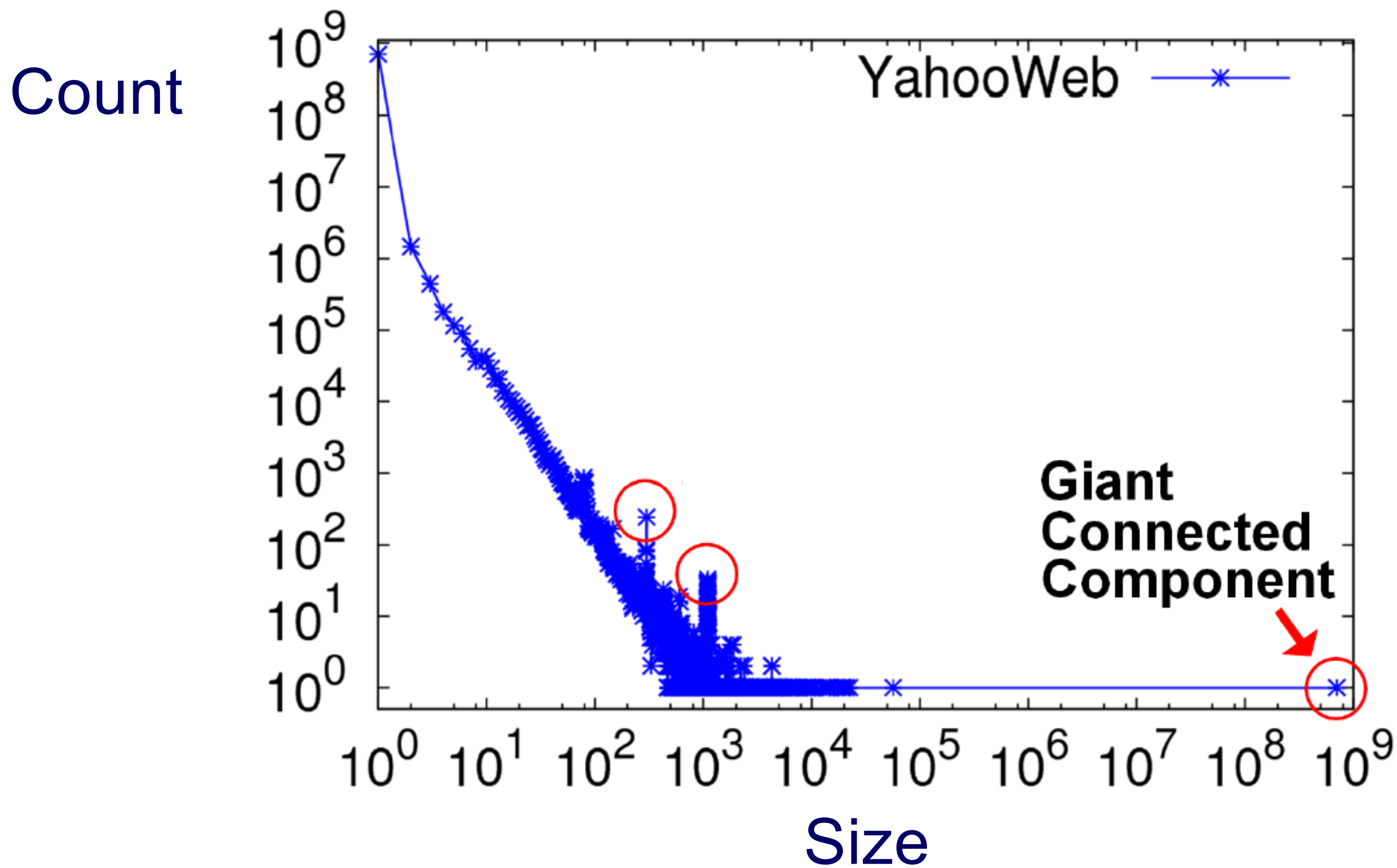
internet domains





S2: connected component sizes

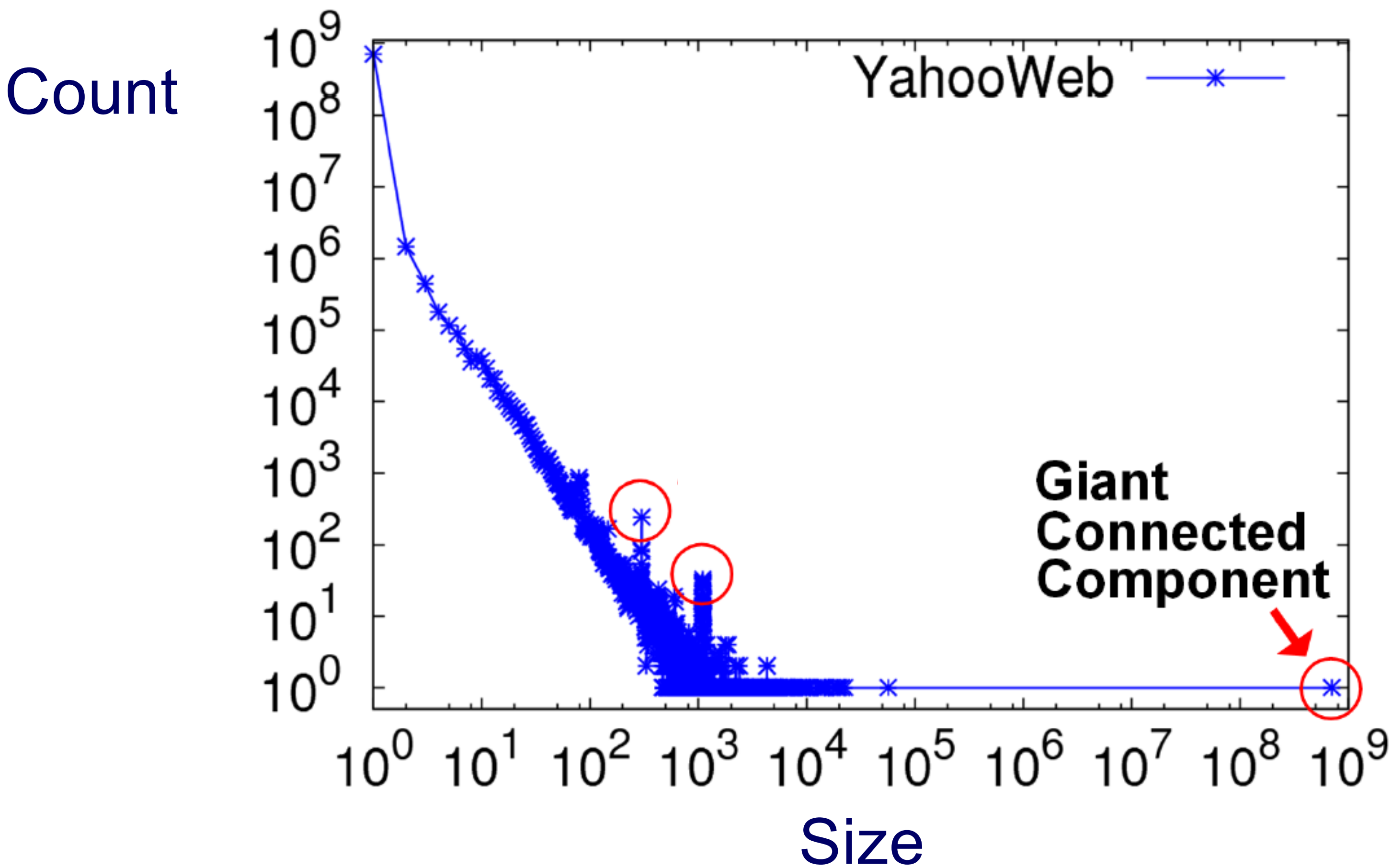
- Connected Components – 4 observations:



1.4B nodes
6B edges

S2: connected component sizes

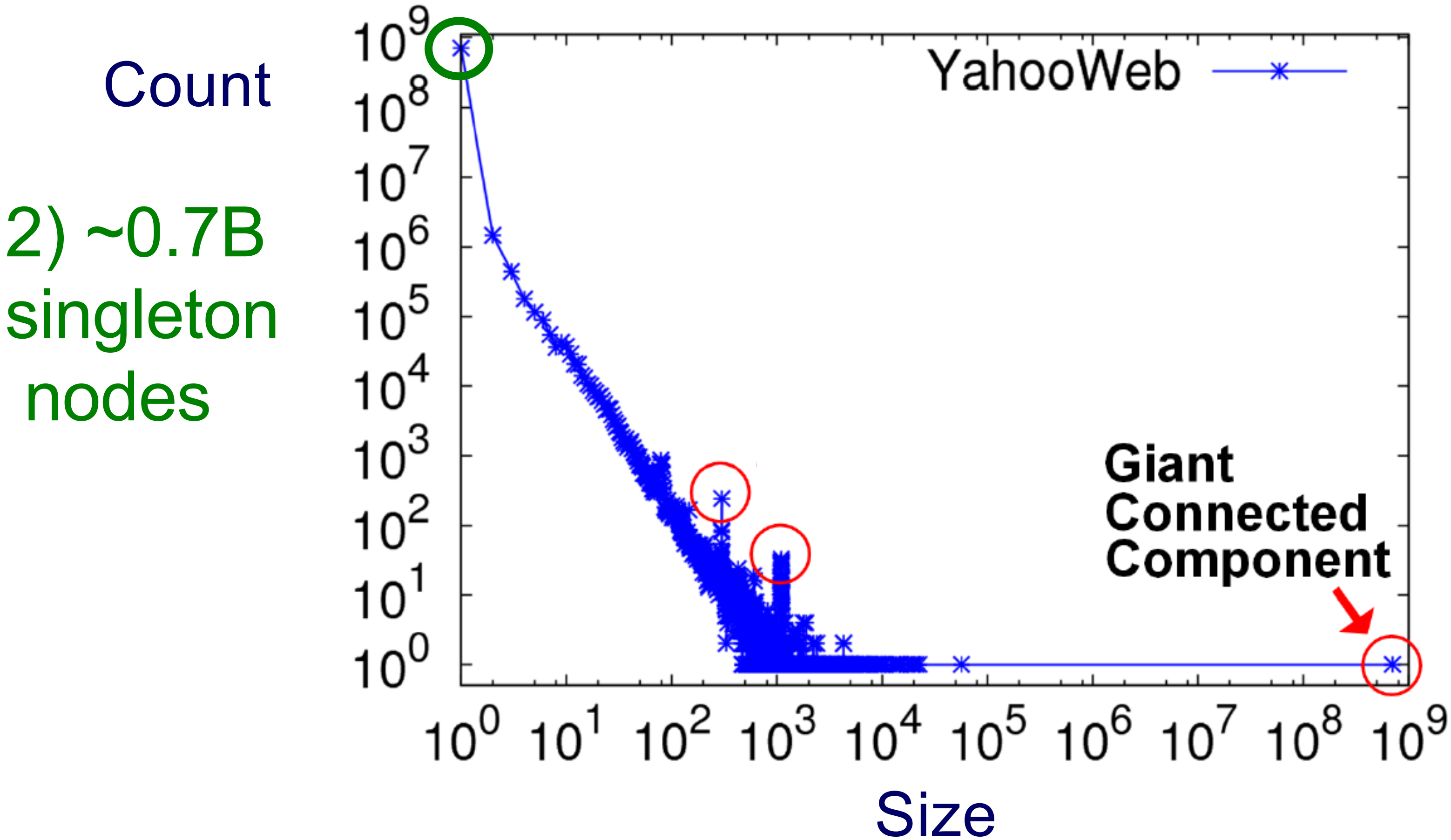
- Connected Components



1) 10K x larger than next

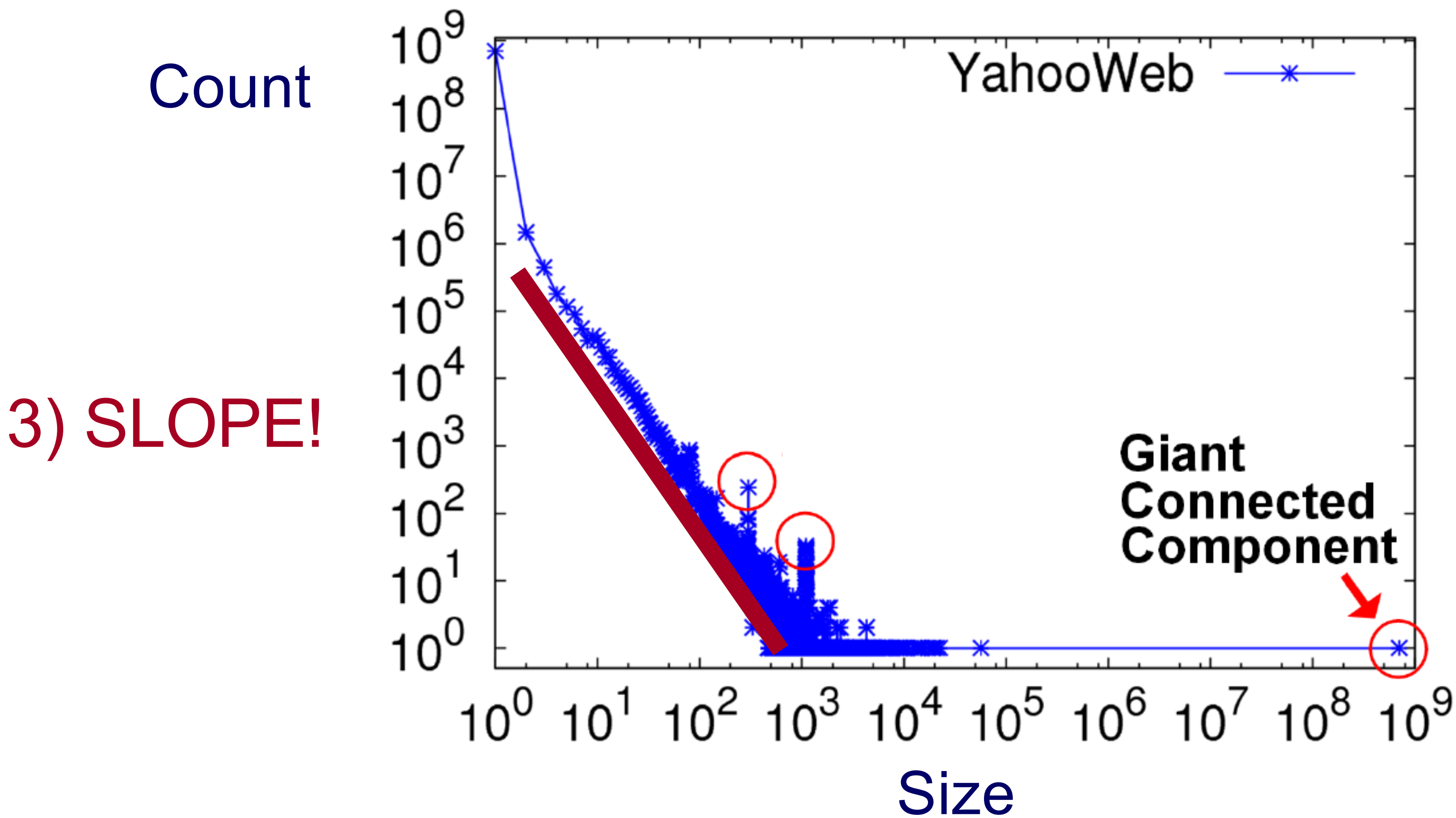
S2: connected component sizes

- Connected Components



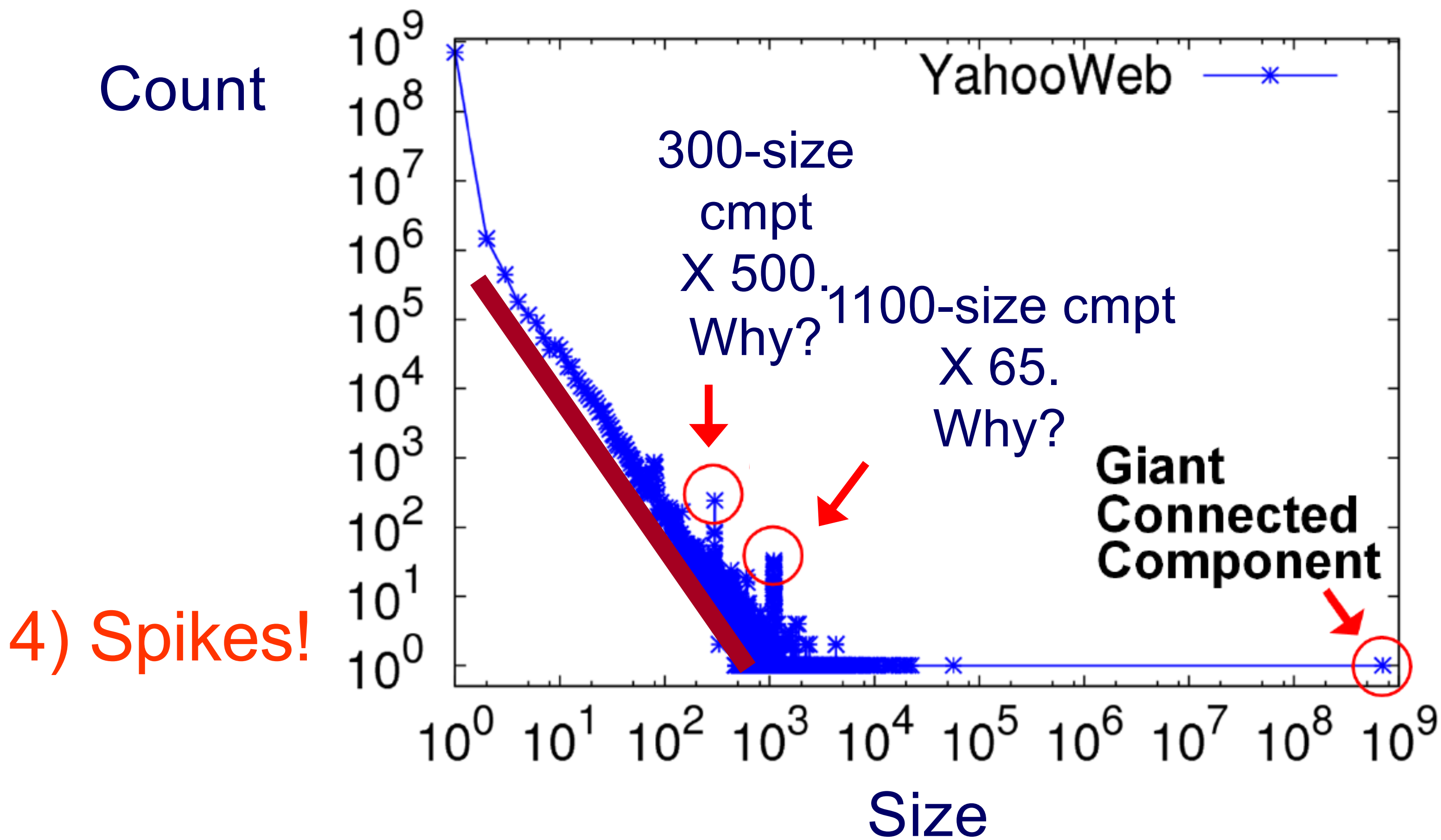
S2: connected component sizes

- Connected Components



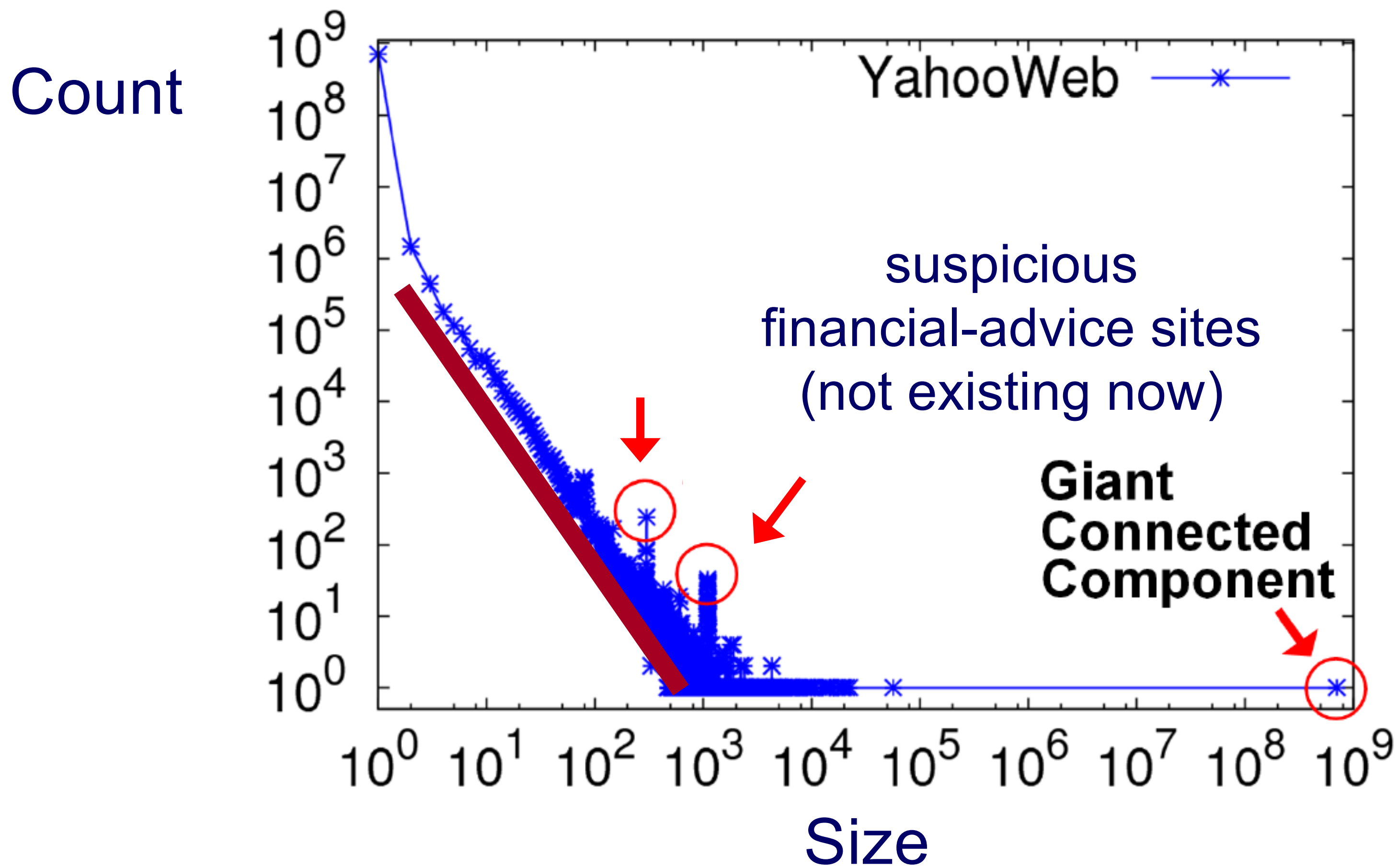
S2: connected component sizes

- Connected Components



S2: connected component sizes

- Connected Components

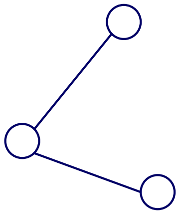


Roadmap (detailed)

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
  - 1.1 Patterns (degree, conn-comp, triangles)
  - 1.2 Anomalies
- Part#2: Graph Mining – (semi-)supervised
- …

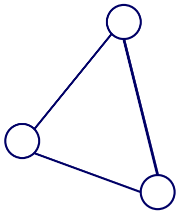
CarnegieMellon

Solution# S.3: Triangle 'Laws'

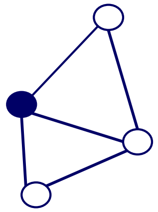


- Real social networks have a lot of triangles

Solution# S.3: Triangle 'Laws'

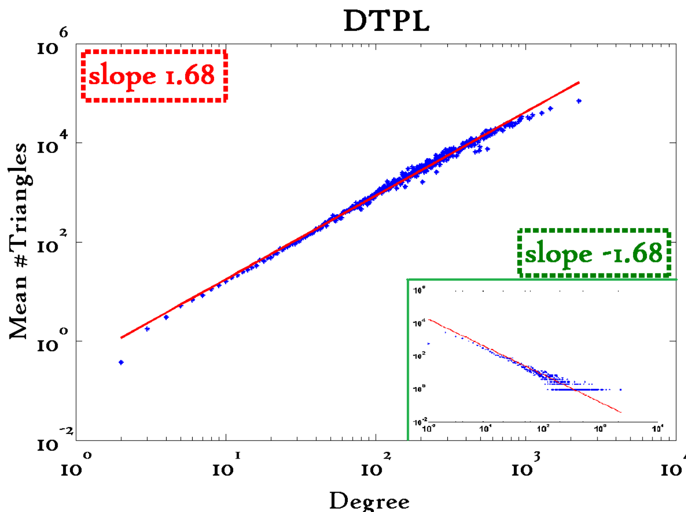


- Real social networks have a lot of triangles
  - Friends of friends are friends
- Any patterns?
  - 2x the friends, 2x the triangles ?

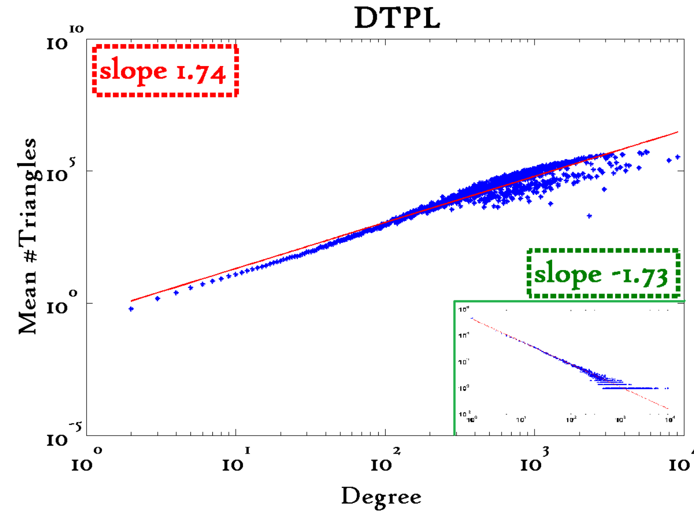


Triangle Law: #S.3
[Tsourakakis ICDM 2008]

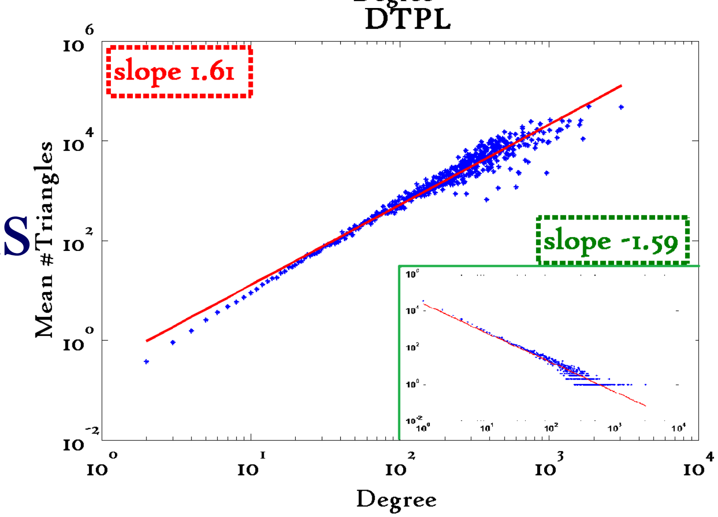
Reuters

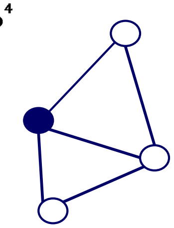


SN



Epinions





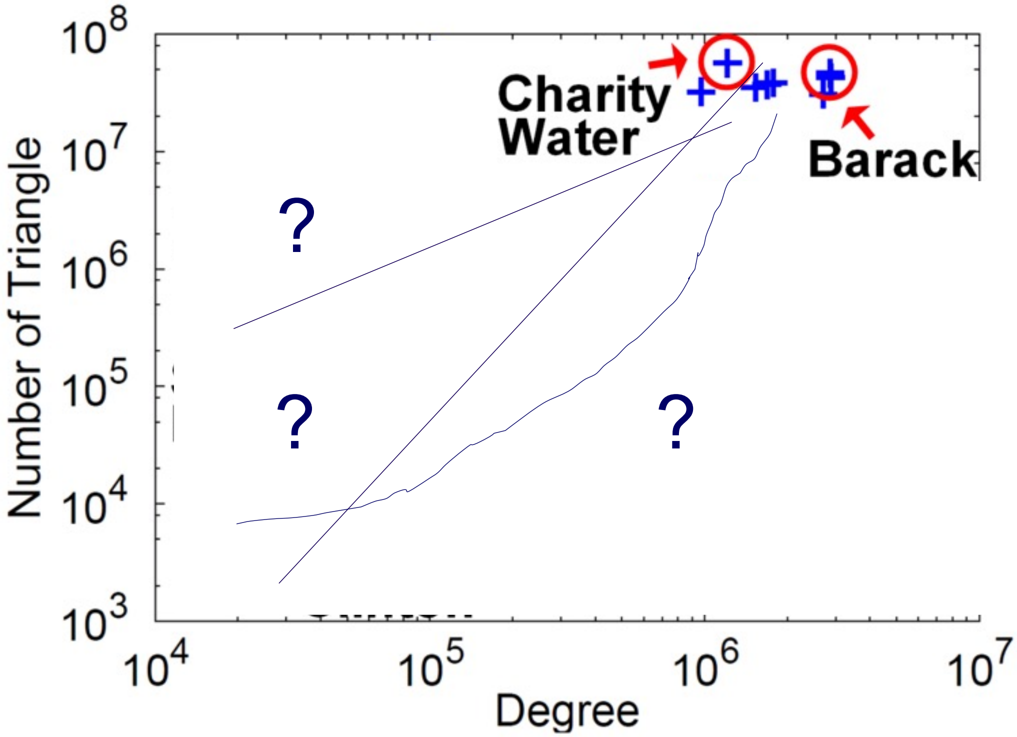
X-axis: degree

Y-axis: mean # triangles

n friends -> ~$n^{1.6}$ triangles

Carnegie Mellon

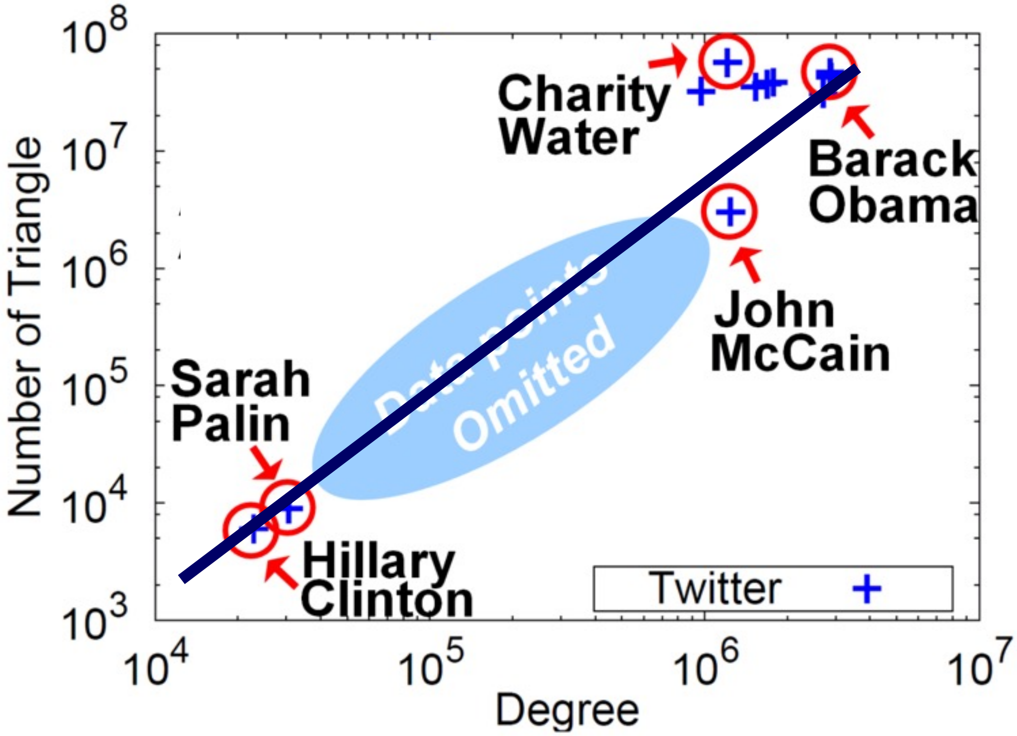
Triangle counting for large graphs?



Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

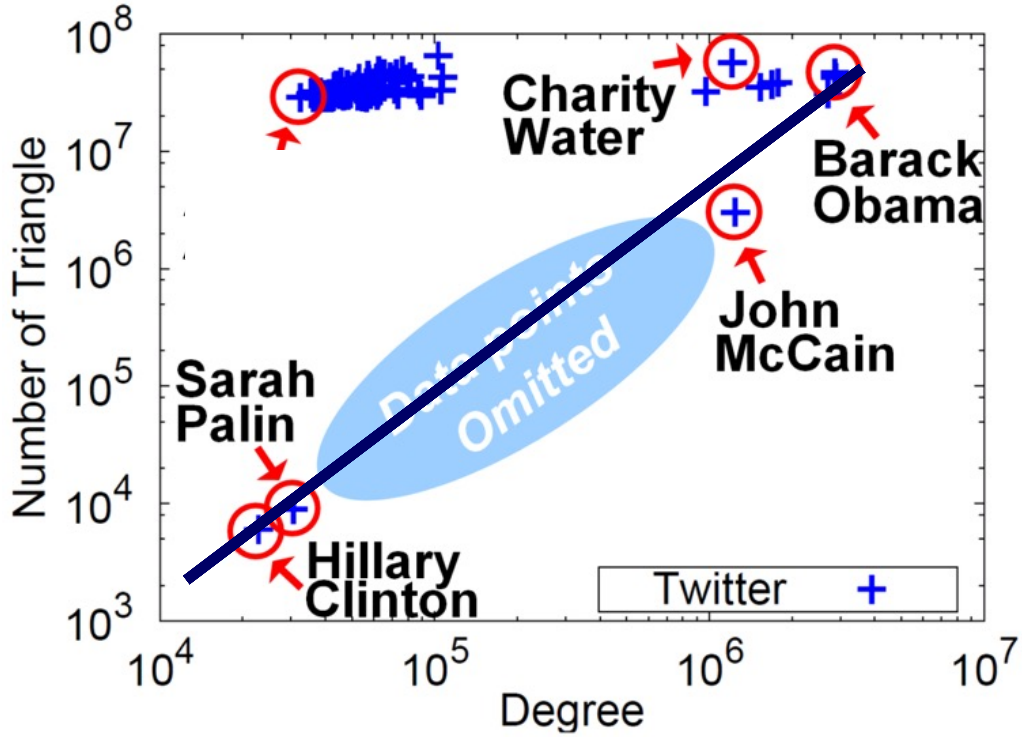
Triangle counting for large graphs?



Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

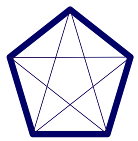
Triangle counting for large graphs?

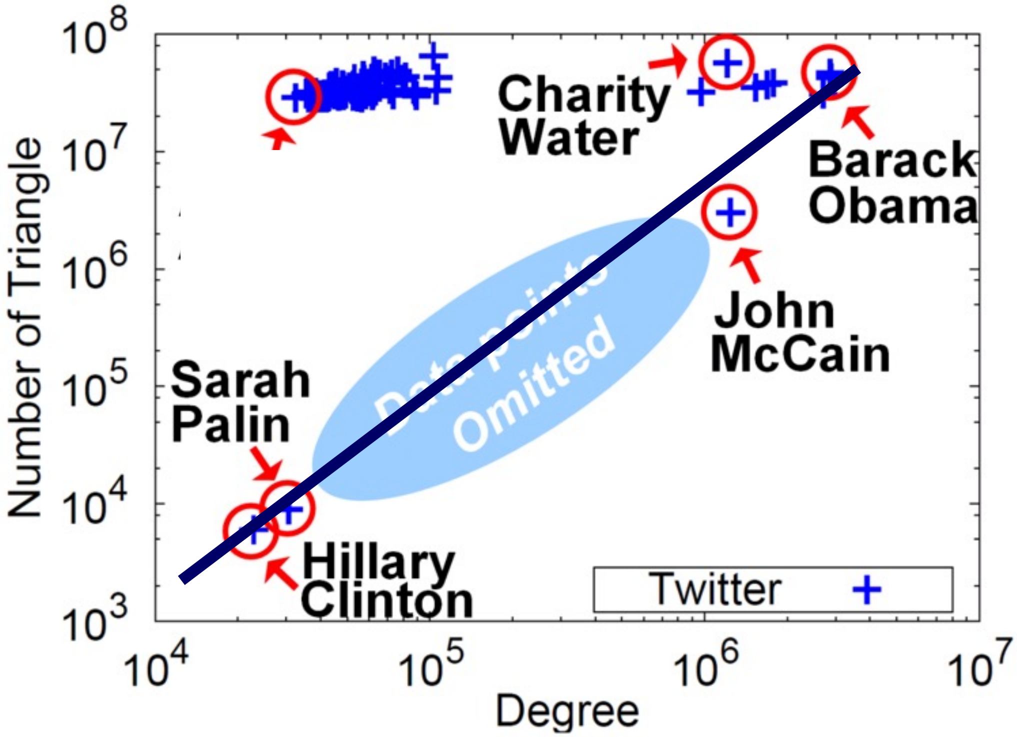


Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

Triangle counting for large graphs?

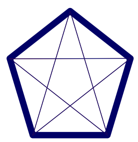


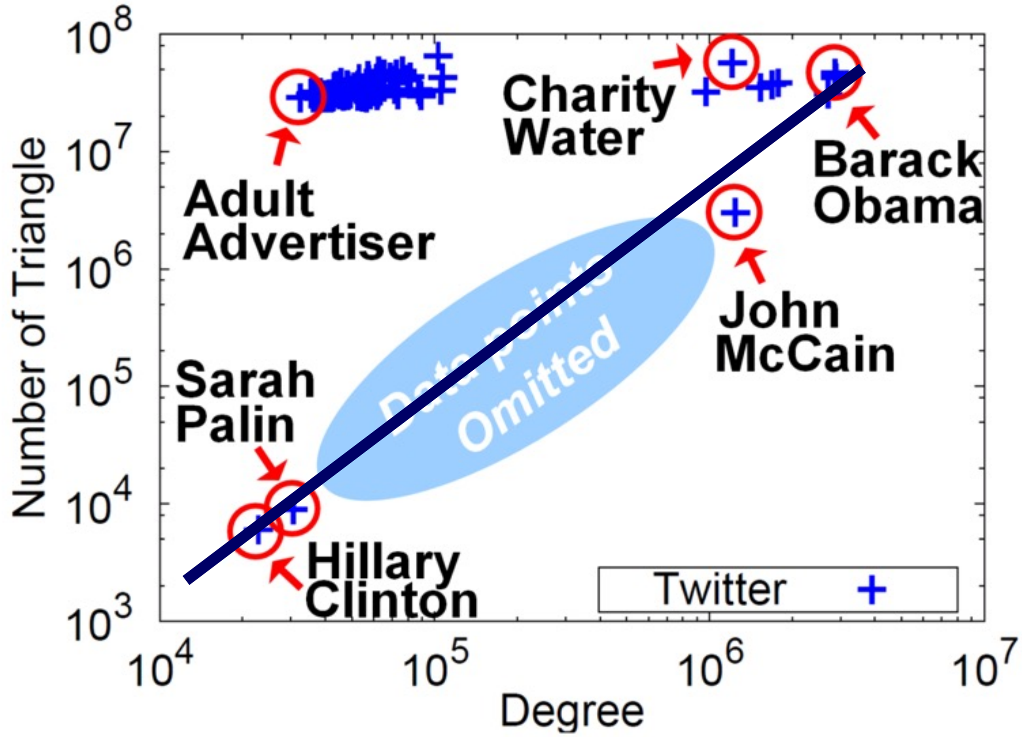


Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD'11]

Triangle counting for large graphs?





Anomalous nodes in Twitter(~ 3 billion edges)

[U Kang, Brendan Meeder, +, PAKDD’11]

MORE Graph Patterns

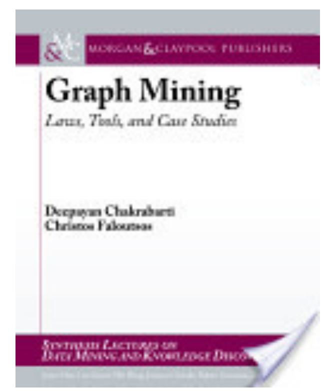
	Unweighted	Weighted
Static	L01. Power-law degree distribution [Faloutsos et al. \`99, Kleinberg et al. \`99, Chakrabarti et al. \`04, Newman \`04] L02. Triangle Power Law (TPL) [Tsourakakis \`08] L03. Eigenvalue Power Law (EPL) [Siganos et al. \`03] L04. Community structure [Flake et al. \`02, Girvan and Newman \`02]	L10. Snapshot Power Law (SPL) [McGlohon et al. \`08]
Dynamic	L05. Densification Power Law (DPL) [Leskovec et al. \`05] L06. Small and shrinking diameter [Albert and Barabási \`99, Leskovec et al. \`05] L07. Constant size 2nd and 3rd connected components [McGlohon et al. \`08] L08. Principal Eigenvalue Power Law (λ_1PL) [Akoglu et al. \`08] L09. Bursty/self-similar edge/weight additions [Gomez and Santonja \`98, Gribble et al. \`98, Crovella and	L11. Weight Power Law (WPL) [McGlohon et al. \`08]

RTG: A Recursive Realistic Graph Generator using Random Typing Leman Akoglu and Christos Faloutsos. *PKDD*'09.

MORE Graph Patterns

	Unweighted	Weighted
Static	L01. Power-law degree distribution [Faloutsos et al. `99, Kleinberg et al. `99, Chakrabarti et al. `04, Newman `04] L02. Triangle Power Law (TPL) [Tsourakakis `08] L03. Eigenvalue Power Law (EPL) [Siganos et al. `03] L04. Community structure [Flake et al. `02, Girvan and Newman `02]	L10. Snapshot Power Law (SPL) [McGlohon et al. `08]
Dynamic	L05. Densification Power Law (DPL) [Leskovec et al. `05] L06. Small and shrinking diameter [Albert and Barabási `99, Leskovec et al. `05] L07. Constant size 2nd and 3rd connected components [McGlohon et al. `08] L08. Principal Eigenvalue Power Law (λ_1PL) [Akoglu et al. `08] L09. Bursty/self-similar edge/weight additions [Gomez and Santonja `98, Gribble et al. `98, Crovella and Bestavros `99, McGlohon et al. `08]	L11. Weight Power Law (WPL) [McGlohon et al. `08]

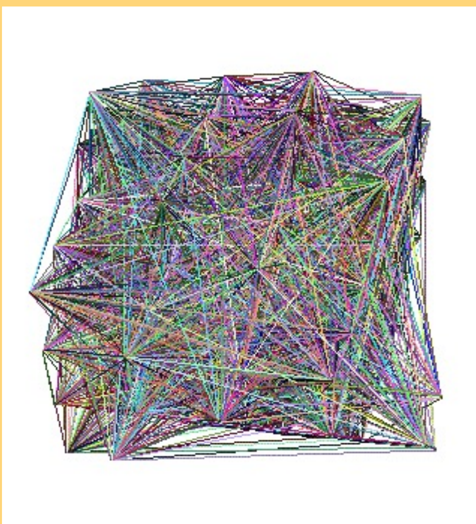
- Mary McGlohon, Leman Akoglu, Christos Faloutsos. *Statistical Properties of Social Networks.* in "Social Network Data Analytics" (Ed.: Charu Aggarwal)



- Deepayan Chakrabarti and Christos Faloutsos, *Graph Mining: Laws, Tools, and Case Studies* Oct. 2012, Springer.

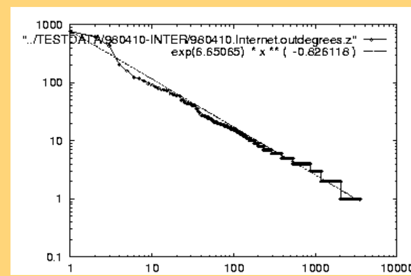
Solution(s)

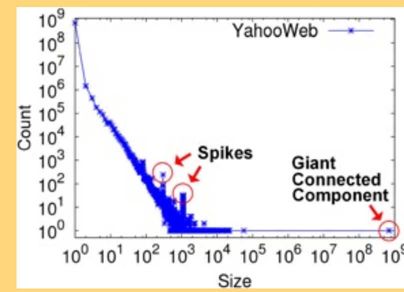
Given:

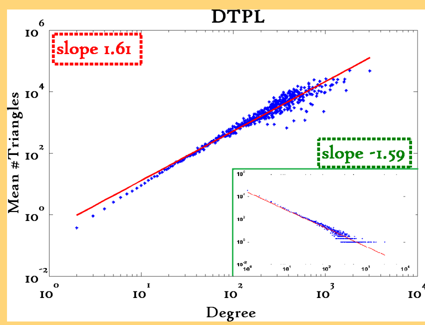


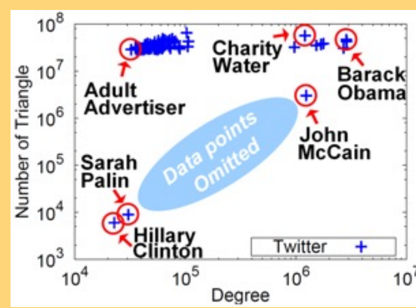
Find patterns ('what is normal')

6-degrees









...

Roadmap (detailed)

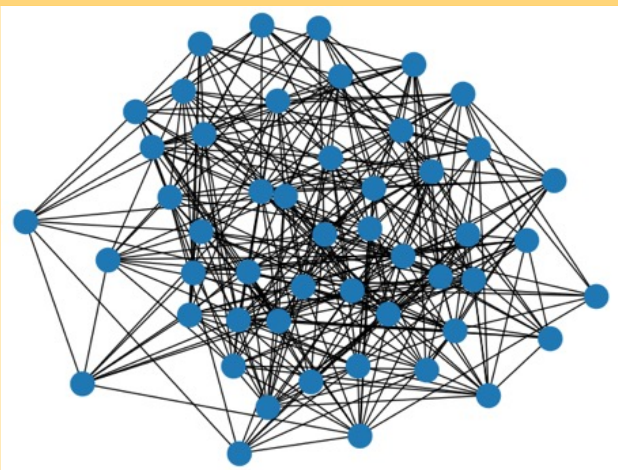
- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
  - 1.1 Patterns
  - 1.2 Anomalies

Patterns  anomalies

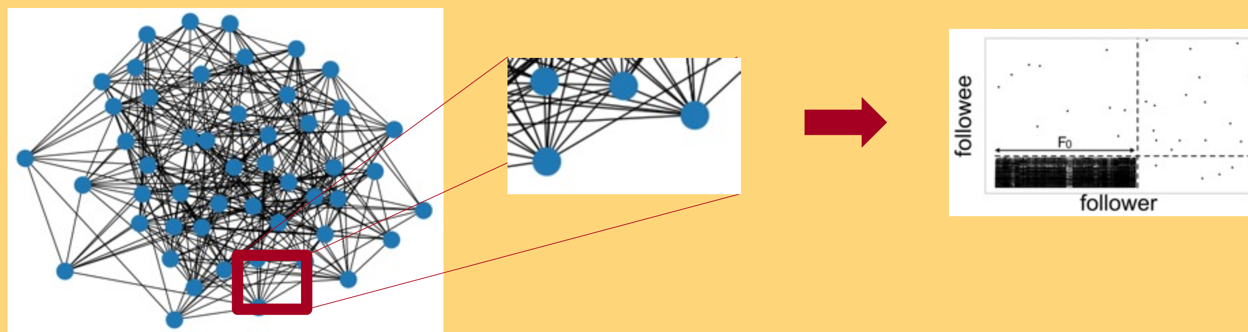
- Part#2: Graph Mining – (semi-)supervised
- …

Problem

Given:

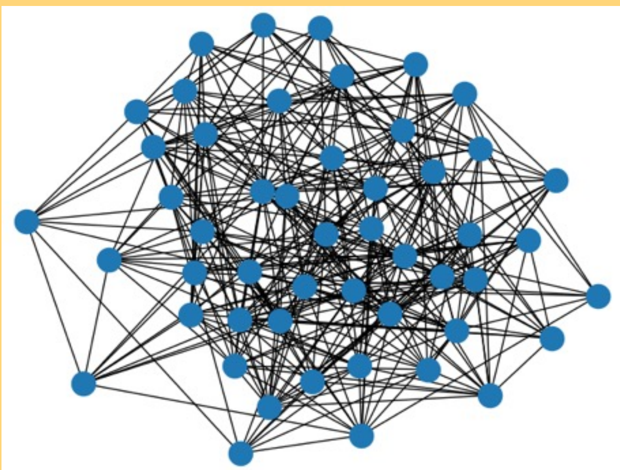


Find: suspicious sub-graphs



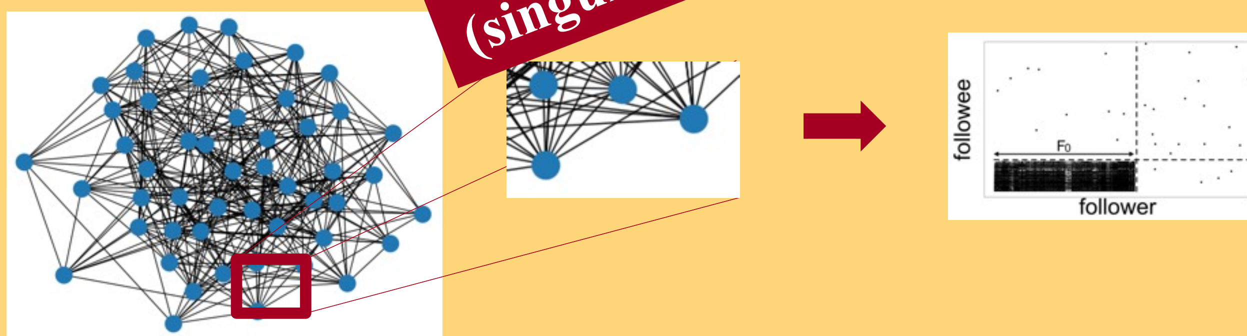
Solution

Given:



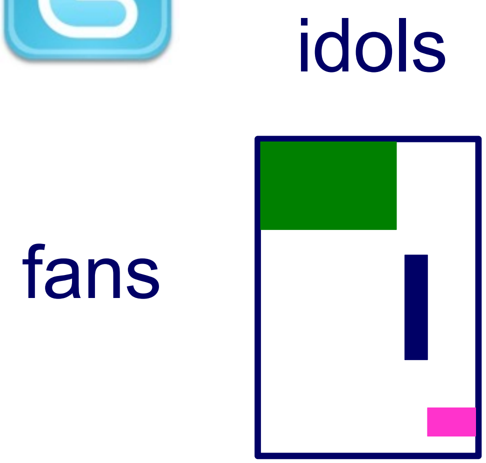
Find: suspicious sub-graphs

SVD
(singular value decomposition)

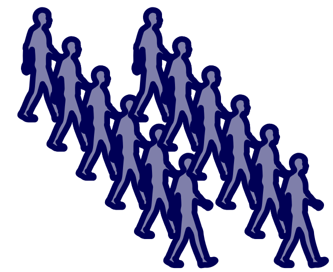


How to find 'suspicious' groups?

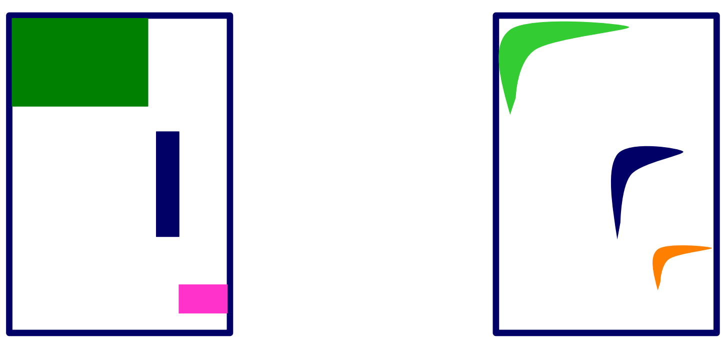
- 'blocks' are normal, right?



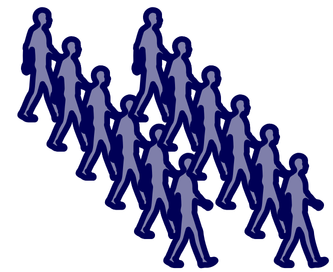
Except that:



- ‘blocks’ are normal, right?
- ‘hyperbolic’ communities are more realistic [Araujo+, PKDD’14]

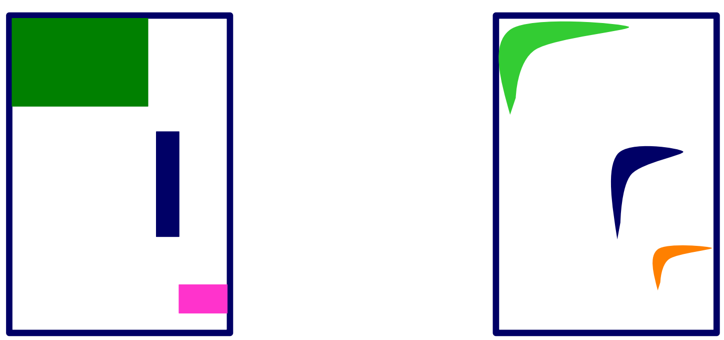


Except that:

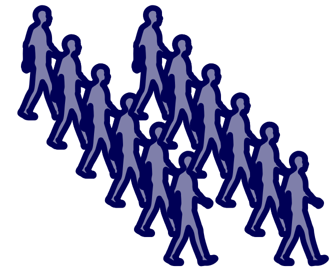


- ‘blocks’ are usually suspicious
- ‘hyperbolic’ communities are more realistic [Araujo+, PKDD’14]

Q: Can we spot blocks, easily?



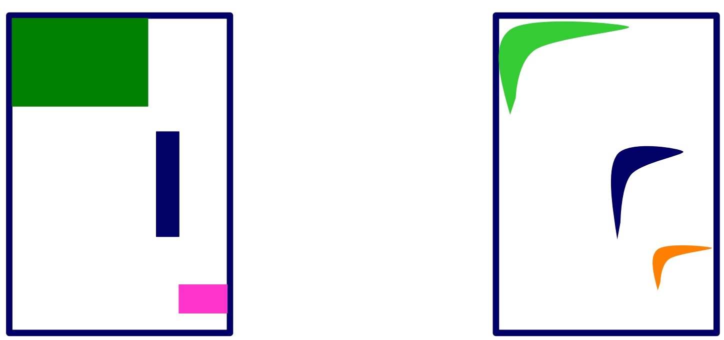
Except that:



- ‘blocks’ are usually suspicious
- ‘hyperbolic’ communities are more realistic [Araujo+, PKDD’14]

Q: Can we spot blocks, easily?

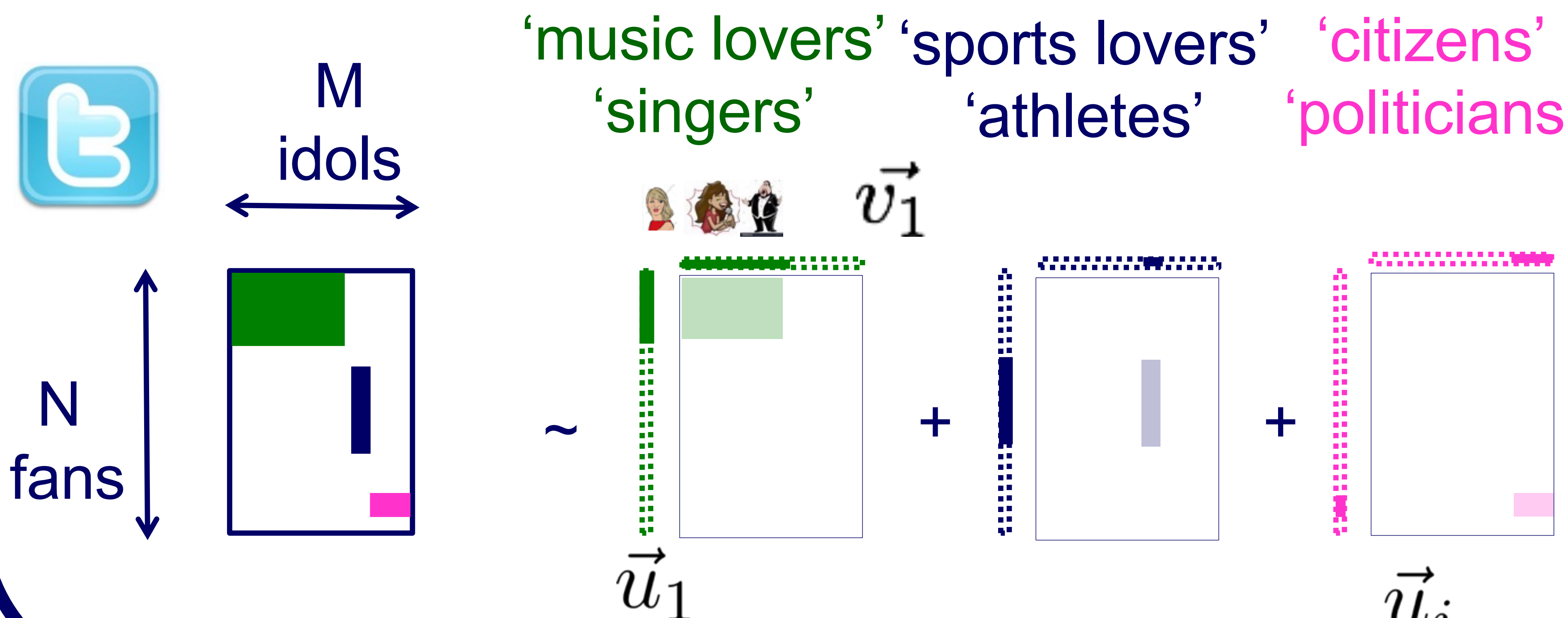
A: Silver bullet: SVD!



DETAILS

Crush intro to SVD

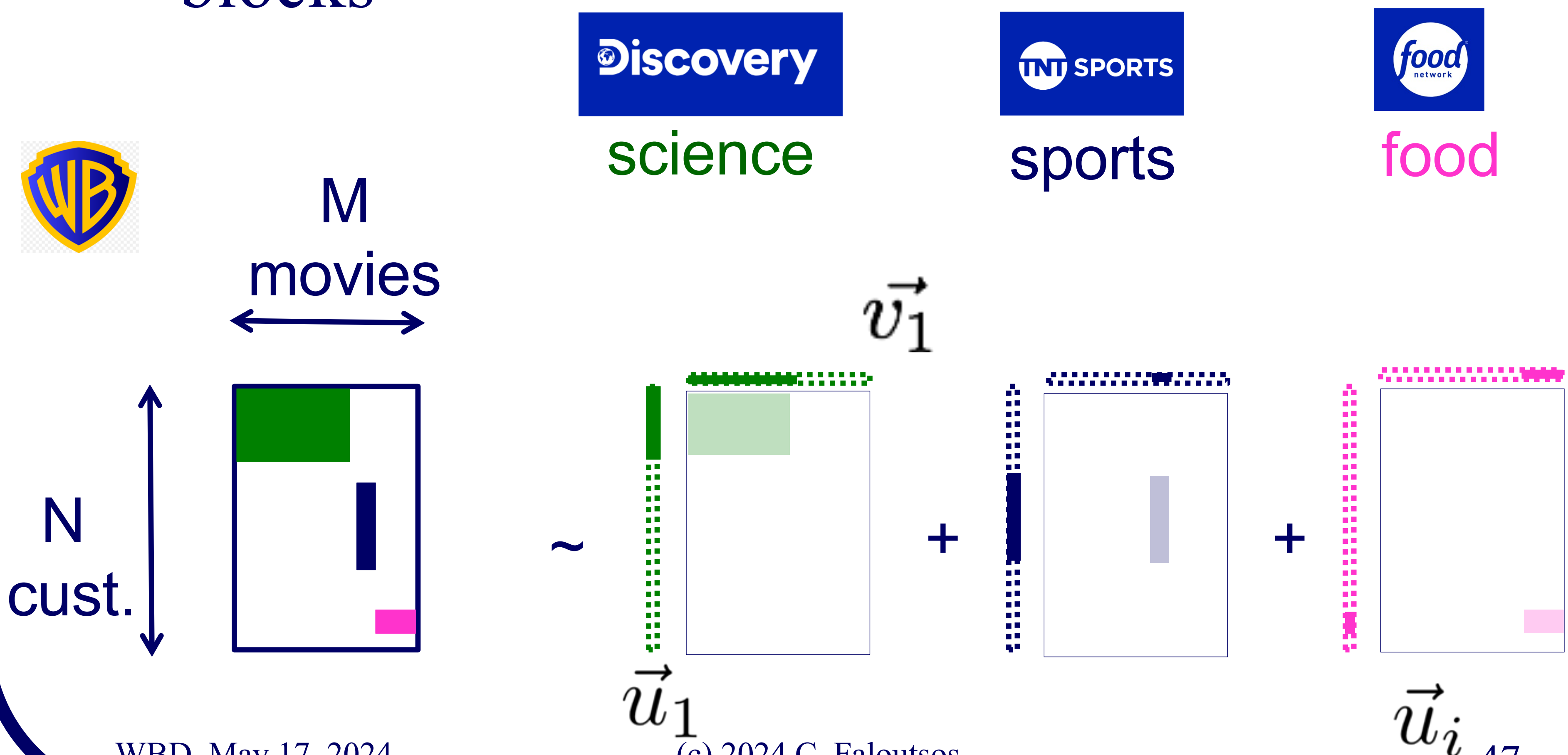
- Recall: (SVD) matrix factorization: finds blocks



DETAILS

Crush intro to SVD

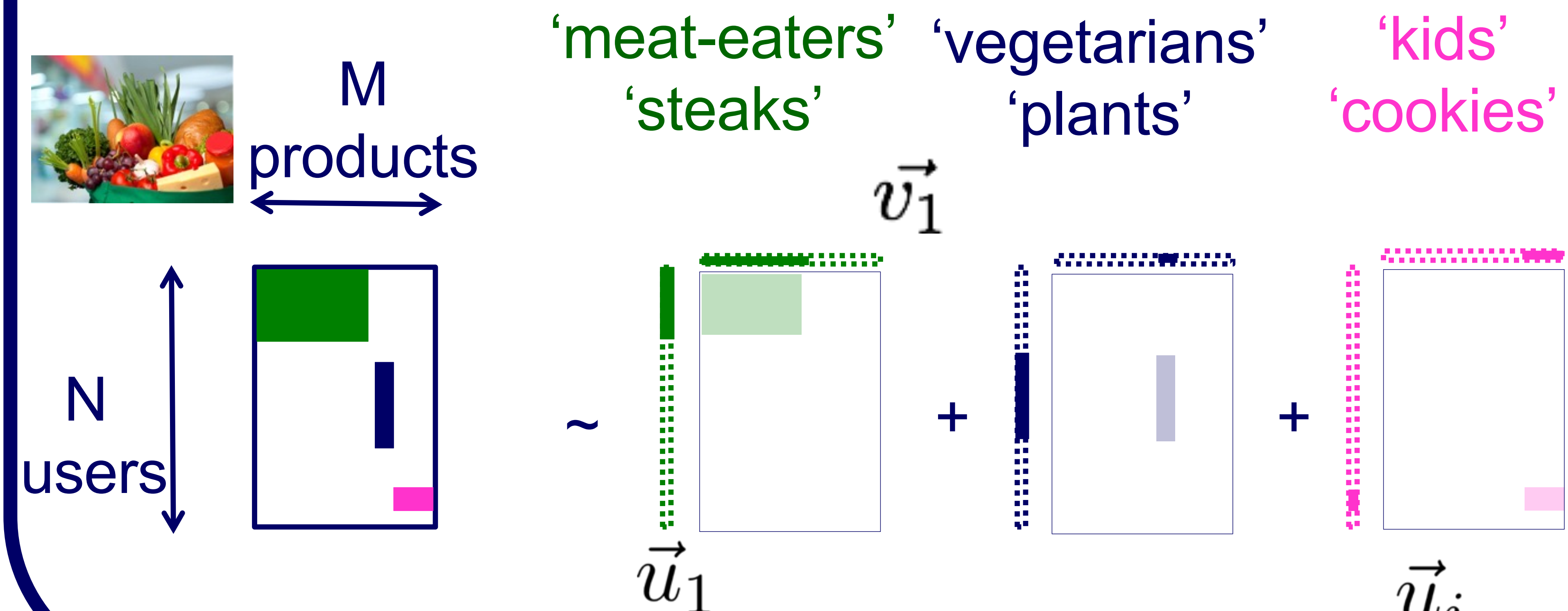
- Recall: (SVD) matrix factorization: finds blocks



DETAILS

Crush intro to SVD

- Recall: (SVD) matrix factorization: finds blocks

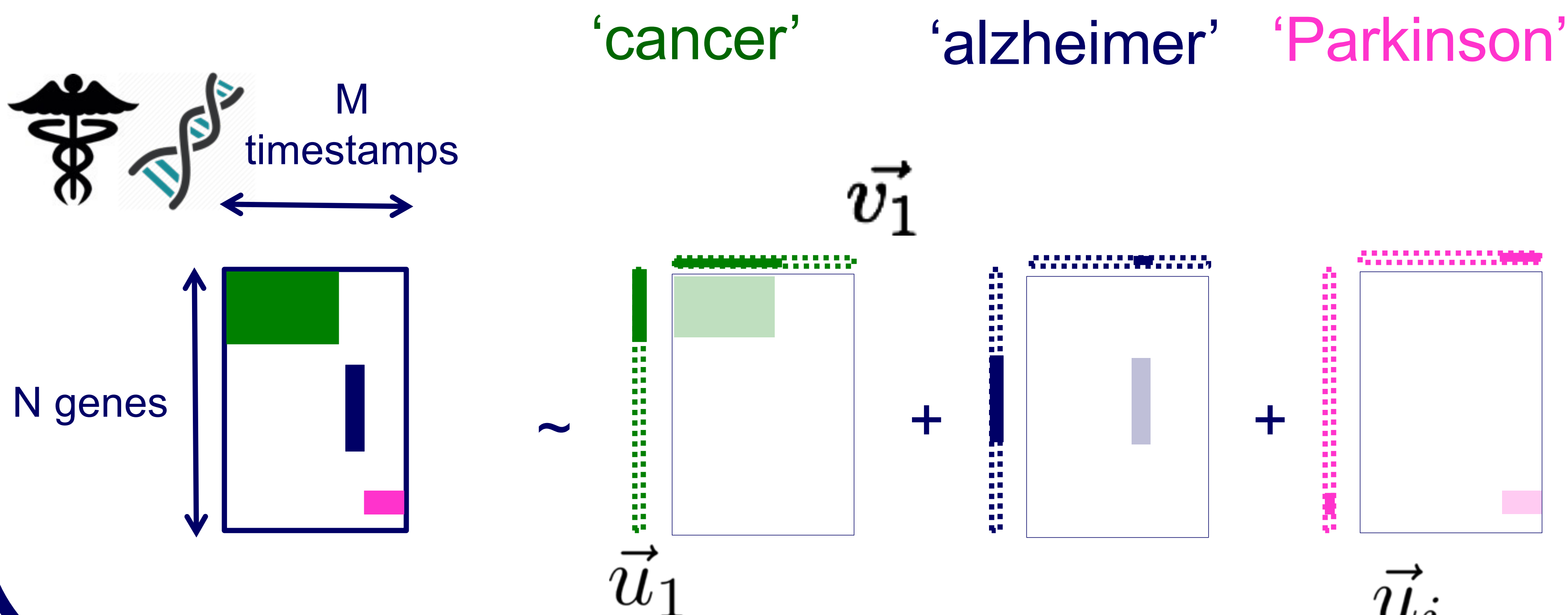


Carnegie Mellon

DETAILS

Crush intro to SVD

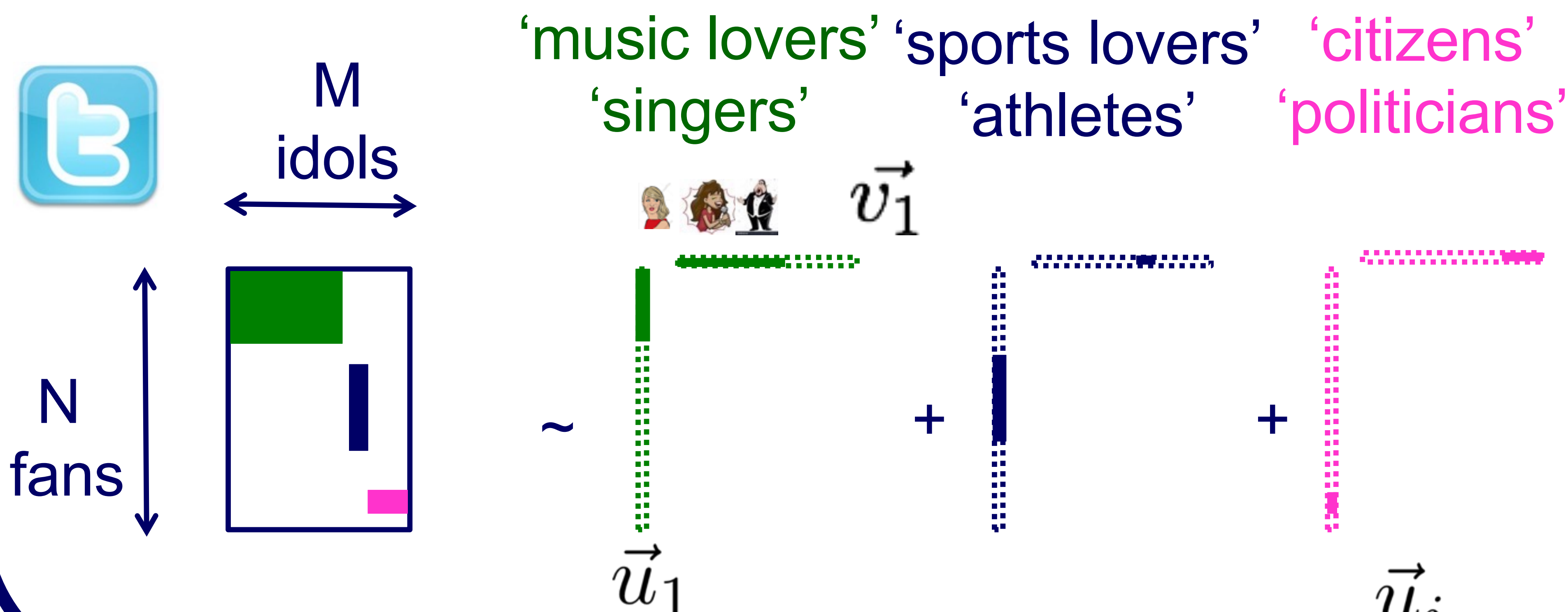
- Recall: (SVD) matrix factorization: finds blocks



DETAILS

Crush intro to SVD

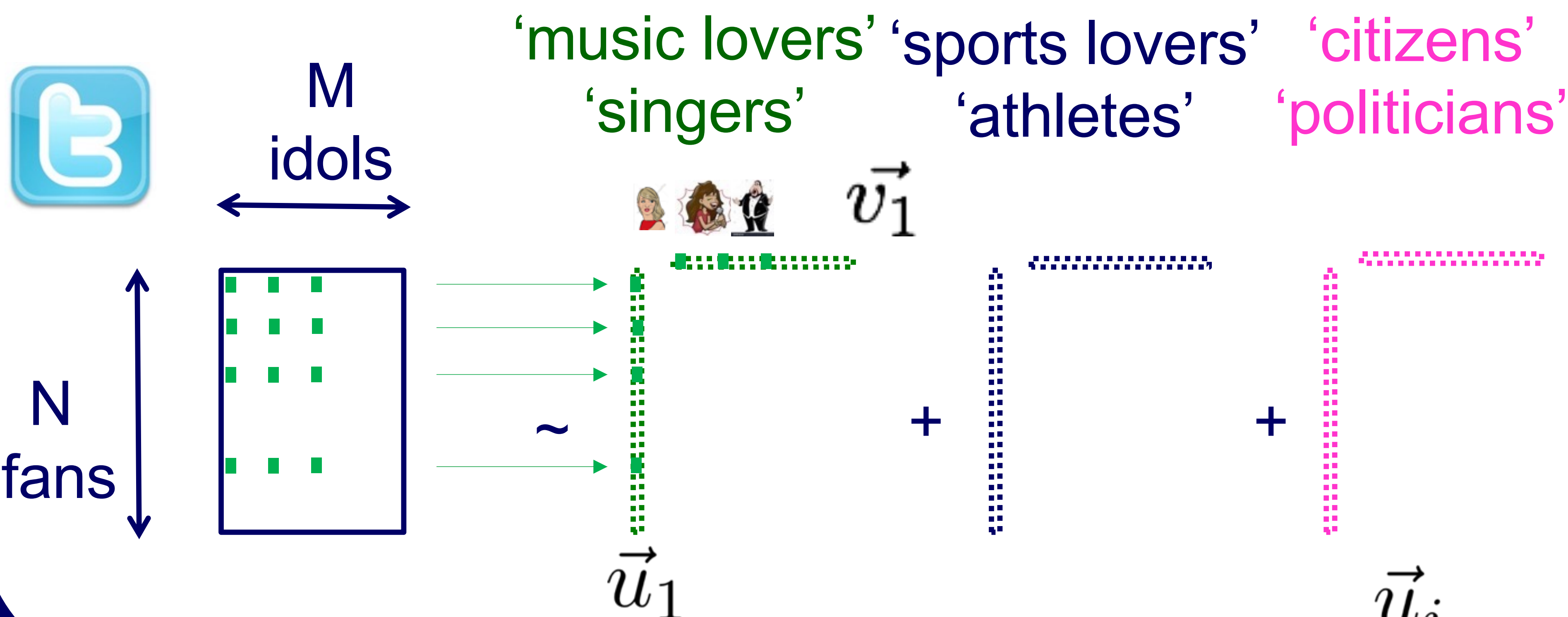
- Recall: (SVD) matrix factorization: finds blocks



DETAILS

Crush intro to SVD

- Recall: (SVD) matrix factorization: finds blocks **Even if shuffled!**



Carnegie Mellon

Inferring Strange Behavior from Connectivity Pattern in Social Networks

PAKDD'14



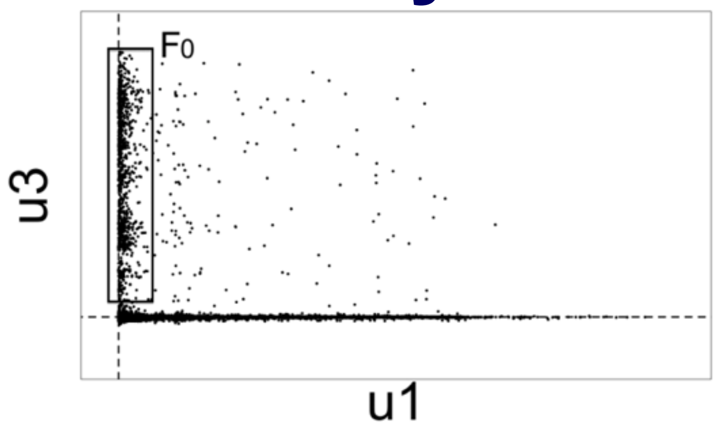
Meng Jiang, Peng Cui, Shiqiang Yang (Tsinghua)
Alex Beutel, Christos Faloutsos (CMU)

Dataset

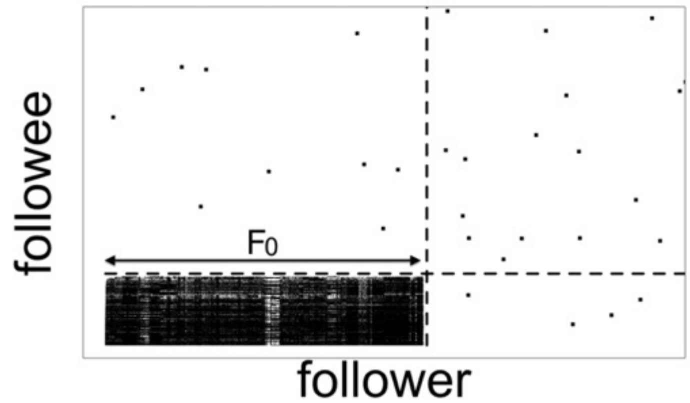
- Tencent Weibo
- 117 million nodes (with profile and UGC data)
- 3.33 billion directed edges

Real Data

"Rays"

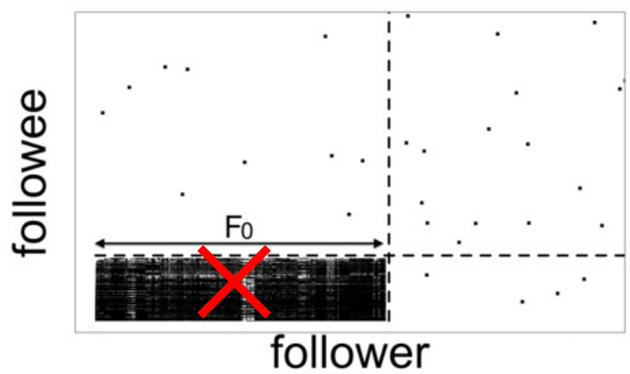


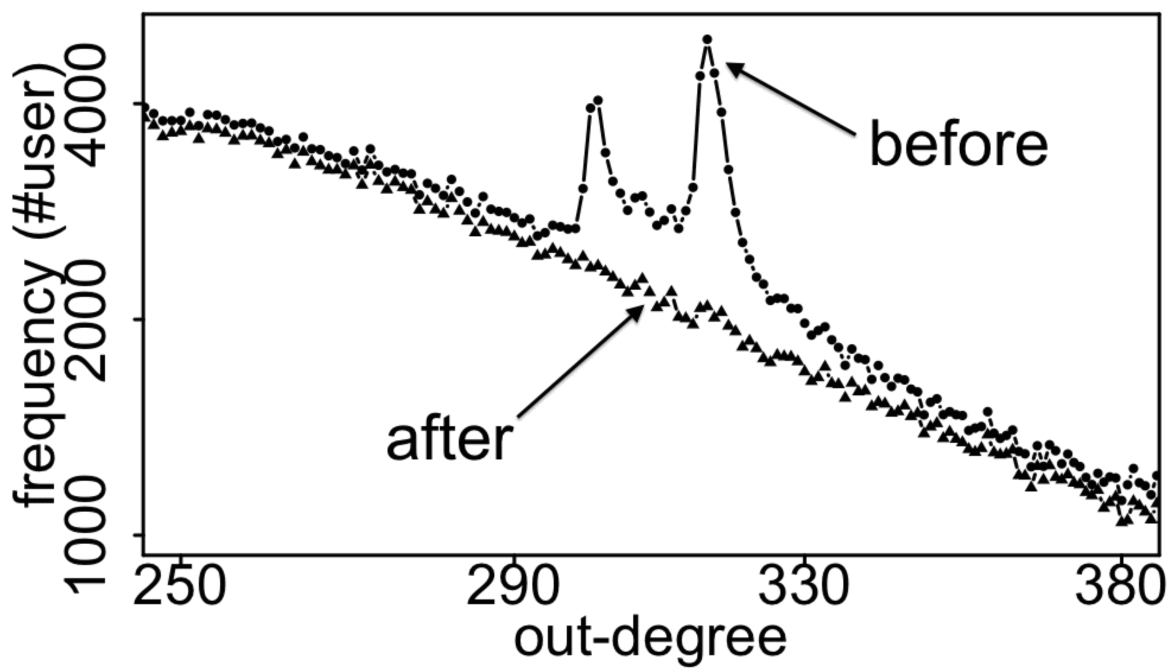
"Block"



Real Data

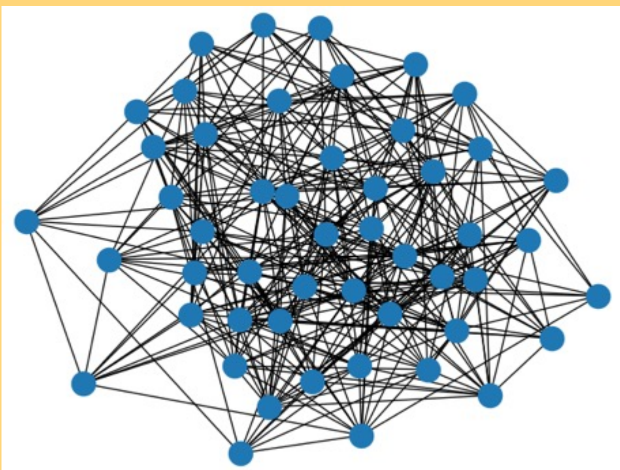
- Spikes on the out-degree distribution





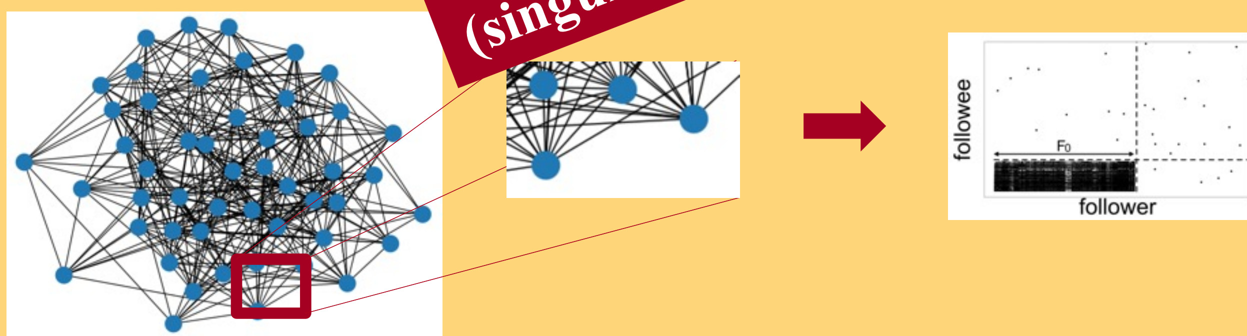
Problem

Given:



Find: suspicious sub-graphs

SVD
(singular value decomposition)



Roadmap

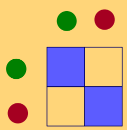
- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
- Part#2: Graph Mining – (semi-)supervised
- Part#3: Time-evolving graphs
- Part#4: Explanations
- Conclusions

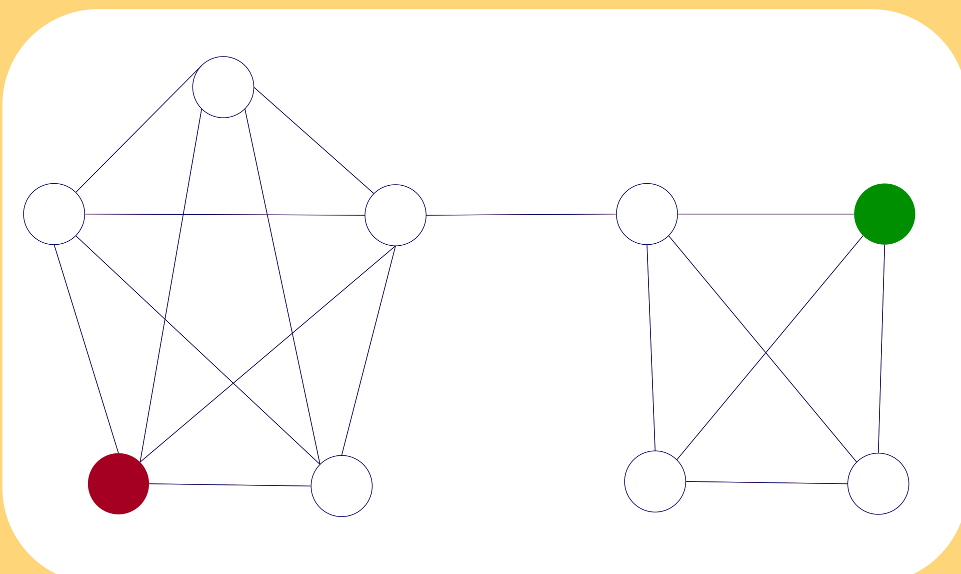
Roadmap

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
- Part#2: Graph Mining – (semi-)supervised
  - 2.1. success stories
  - 2.2. the gory details
- Part#3: Time-evolving graphs
- …

Problem

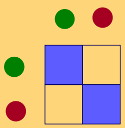
- What color, for the rest?
  - Given homophily (/heterophily etc)?

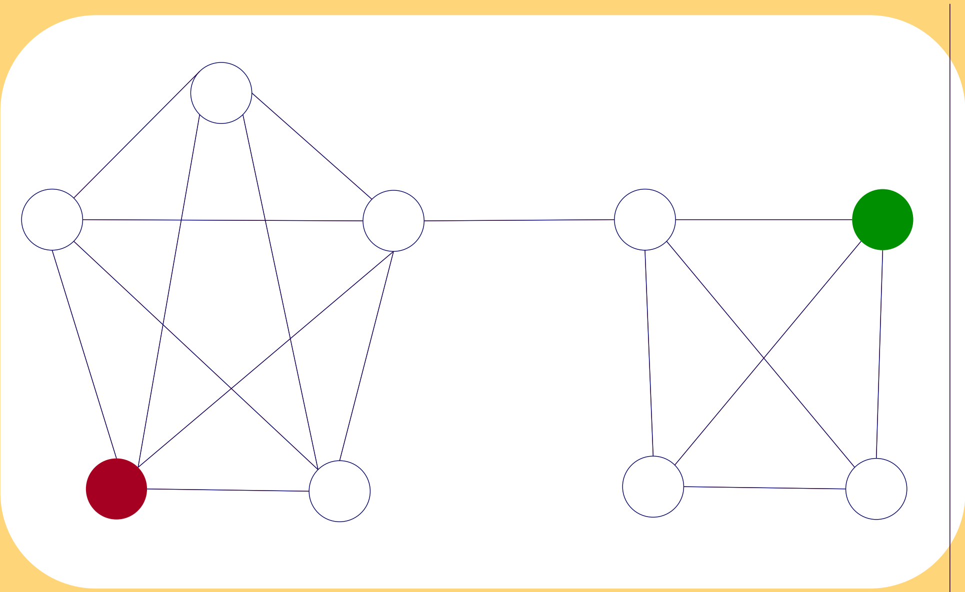




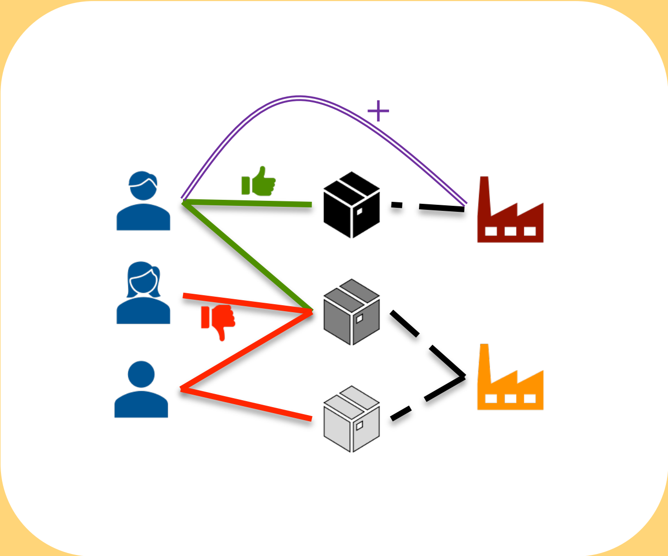
Short answer:

- What color, for the rest?
- A: Belief Propagation ('zooBP')





www.cs.cmu.edu/~deswaran/code/zoobp.zip



Carnegie Mellon

Unifying Guilt-by-Association Approaches: Theorems and Fast Algorithms

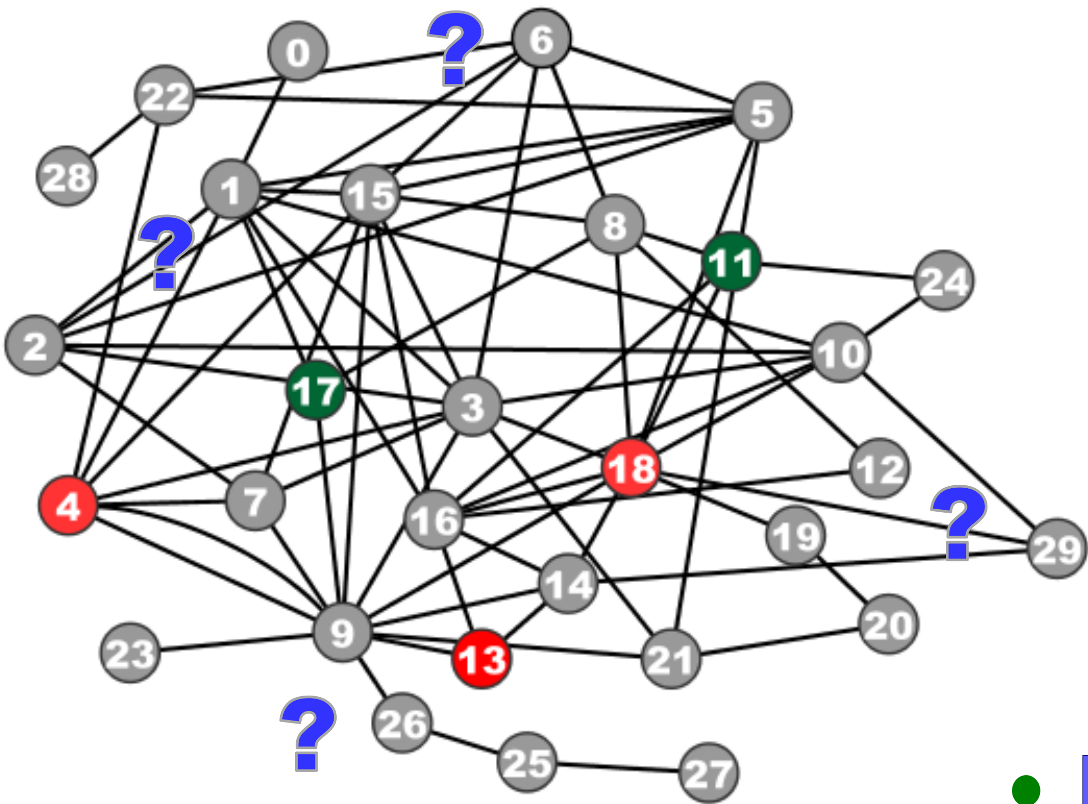
Danai Koutra
U Kang
Hsing-Kuo Kenneth Pao

Tai-You Ke
Duen Horng (Polo) Chau
Christos Faloutsos

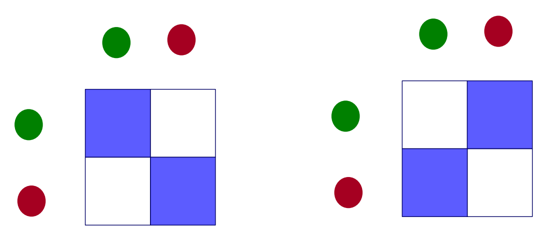
ECML PKDD, 5-9 September 2011, Athens, Greece

Carnegie Mellon

Problem Definition: GBA techniques



Given: Graph; &
few labeled nodes
Find: labels of rest
(assuming network effects)



Are they related?

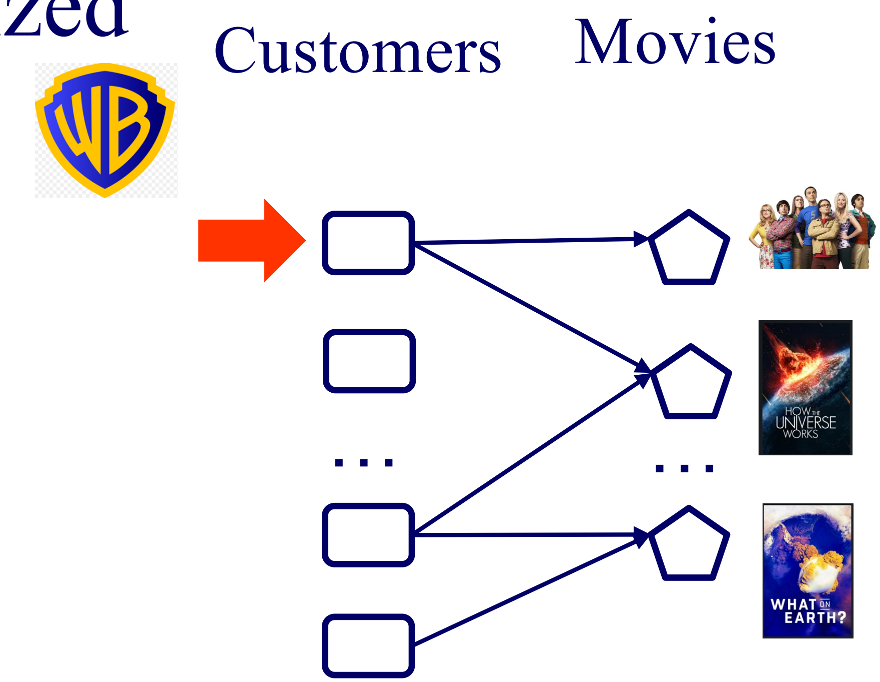
- RWR (Random Walk with Restarts)
  - google's pageRank ('*if my friends are important, I'm important, too*')
- SSL (Semi-supervised learning)
  - minimize the differences among neighbors
- BP (Belief propagation)
  - send messages to neighbors, on what you believe about them

Are they related? YES!

- RWR (Random Walk with Restarts)
  - google's pageRank ('*if my friends are important, I'm important, too*')
- SSL (Semi-supervised learning)
  - minimize the differences among neighbors
- BP (Belief propagation)
  - send messages to neighbors, on what you believe about them

Rec. sys <-> GBA <-> RWR

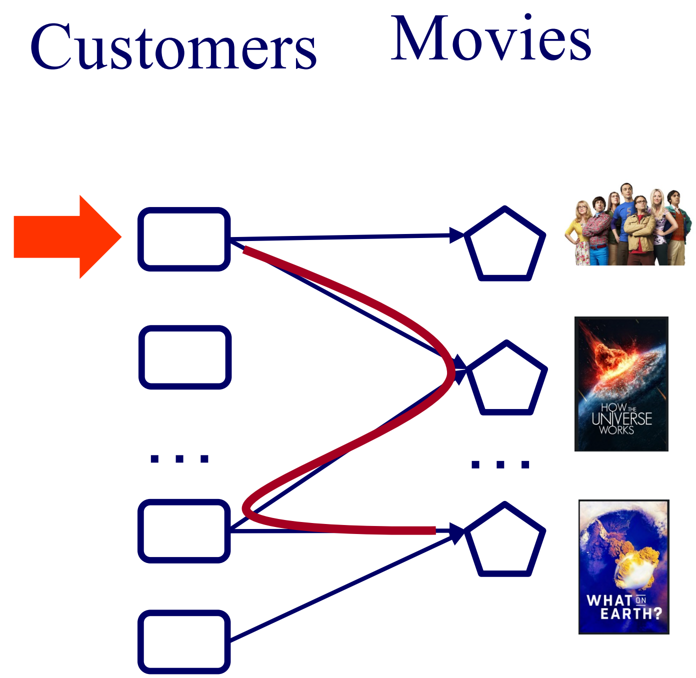
- RWR = PPR (Personalized PageRank)



- Pixie [Eksombatchai+, 2017]

Rec. sys <-> GBA <-> RWR

- RWR = PPR (Personalized PageRank)



- Pixie [Eksombatchai+, 2017]

Carnegie Mellon

Correspondence of Methods

Method	Matrix		Unknow n		known
RWR	$[I - c\ AD^{-1}]$	$\times$	x	$=$	$(1\text{-}c)y$
SSL	$[I + a(D - A)]$	$\times$	x	$=$	y
FABP	$[I + a\ D - c'A]$	$\times$	b_h	$=$	φ_h

$$\begin{bmatrix}1 & & \\ & 1 & \\ & & 1\end{bmatrix} \quad \begin{bmatrix}d1 & & \\ & d2 & \\ & & d3\end{bmatrix} \quad \begin{bmatrix}0 & 1 & 0\\ 1 & 0 & 1\\ 0 & 1 & 0\end{bmatrix} \qquad \begin{bmatrix}?\end{bmatrix} \qquad \begin{bmatrix}0\\ 1\\ 1\end{bmatrix}$$

adjacency matrix

final labels/ beliefs

prior labels/ beliefs

BP vs. Linearized BP

DETAILS

Original [Yedidia+]:

Our proposal:

Belief Propagation

$$m_{ij}(x_j) \leftarrow \sum_{x_i} \phi_i(x_i) \cdot \psi_{ij}(x_i, x_j) \cdot \prod_{n \in N(i) \setminus j} m_{ni}(x_i)$$

$$b_i(x_i) \leftarrow \cdot \phi_i(x_i) \cdot \prod_{j \in N(i)} m_{ij}(x_i)$$

Linearized BP

BP is approximated by

$$[\mathbf{I} + a\mathbf{D} - c'\mathbf{A}]\, \mathbf{b}_h = \phi_h$$

$$\begin{bmatrix} 1 & & \\ & 1 & \\ & & 1 \end{bmatrix} \quad \begin{bmatrix} d1 & & \\ & d2 & \\ & & d3 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} ? \end{bmatrix} \quad \begin{bmatrix} 0 \\ -10^{-2} \\ 10^{-2} \end{bmatrix}$$

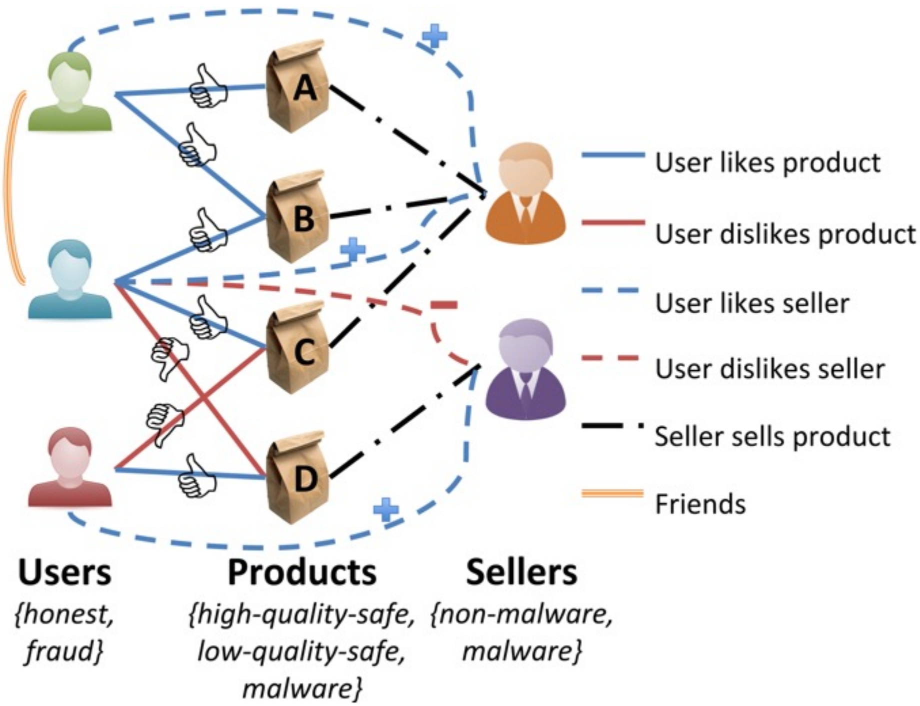
non-linear

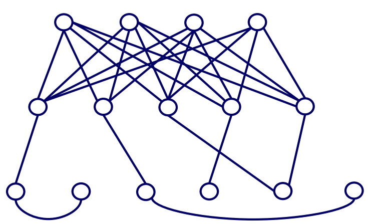
linear

- ✓ Closed-form formula?
- ✓ Convergence?

Carnegie Mellon

Problem: e-commerce ratings fraud

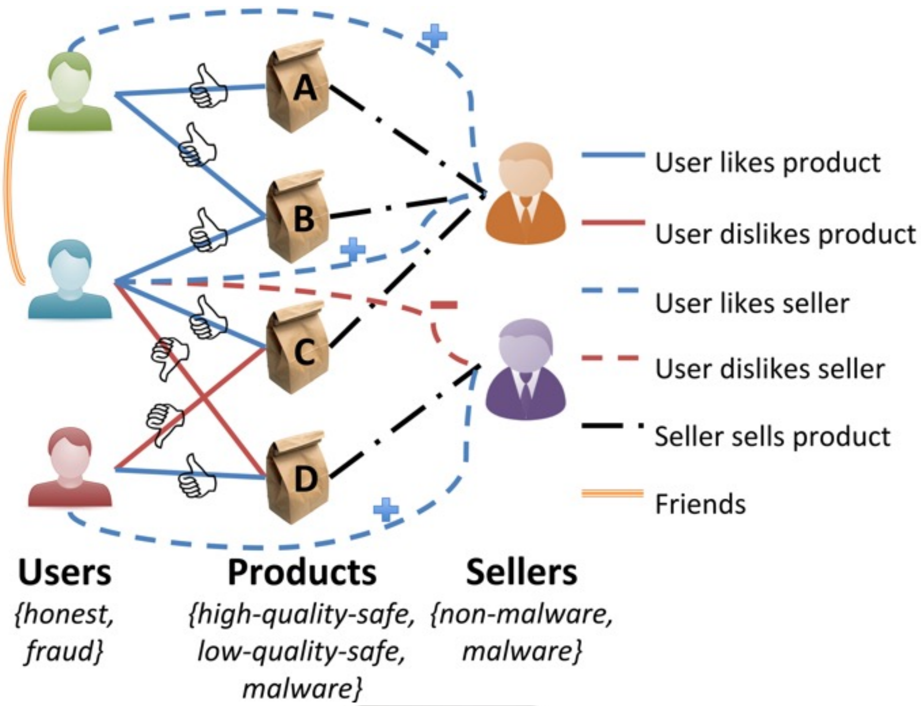




- **Given** a **heterogeneous** graph on users, products, sellers and positive/negative ratings with "seed labels"
- **Find** the top *k* most fraudulent users, products and sellers

Carnegie Mellon

Problem: e-commerce ratings fraud



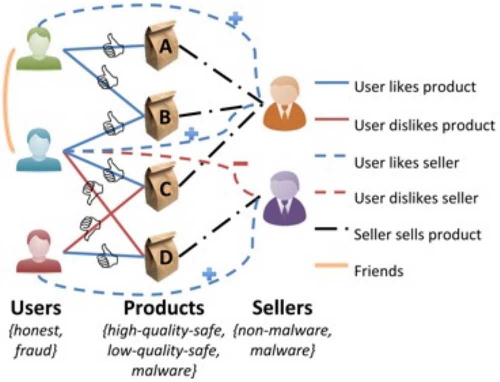
- **Given** a heterogeneous graph on users, products, sellers and positive/negative ratings with "seed labels"
- **Find** the top *k* most fraudulent users, products and sellers

Dhivya Eswaran, Stephan Günnemann, Christos Faloutsos, Disha Makhija, Mohit Kumar, "*ZooBP:* Belief Propagation for Heterogeneous Networks", VLDB 2017

Carnegie Mellon

DETAILS

Problem: e-commerce ratings fraud



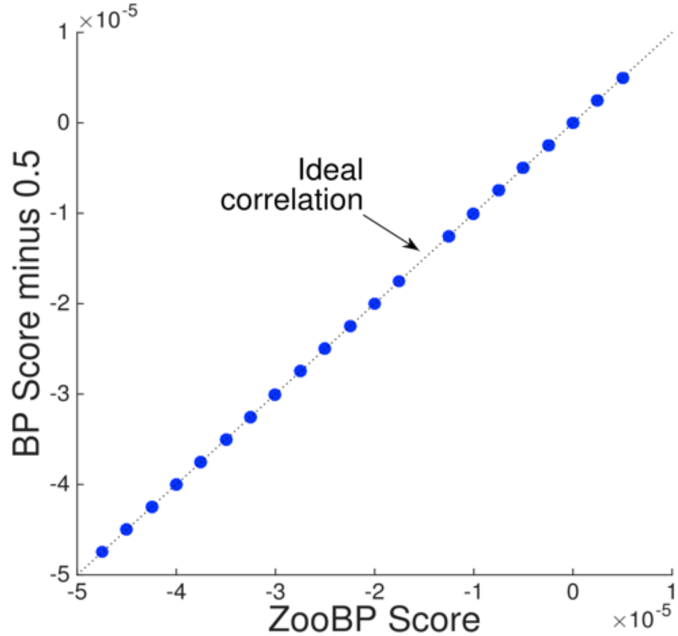
Theorem 1 (ZooBP). *If* $\mathbf{b}, \mathbf{e}, \mathbf{P}, \mathbf{Q}$ *are constructed as described above, the linear equation system approximating the final node beliefs given by BP is:*

$$\mathbf{b} = \mathbf{e} + (\mathbf{P} - \mathbf{Q})\mathbf{b} \qquad (\textit{ZooBP}) \qquad (10)$$

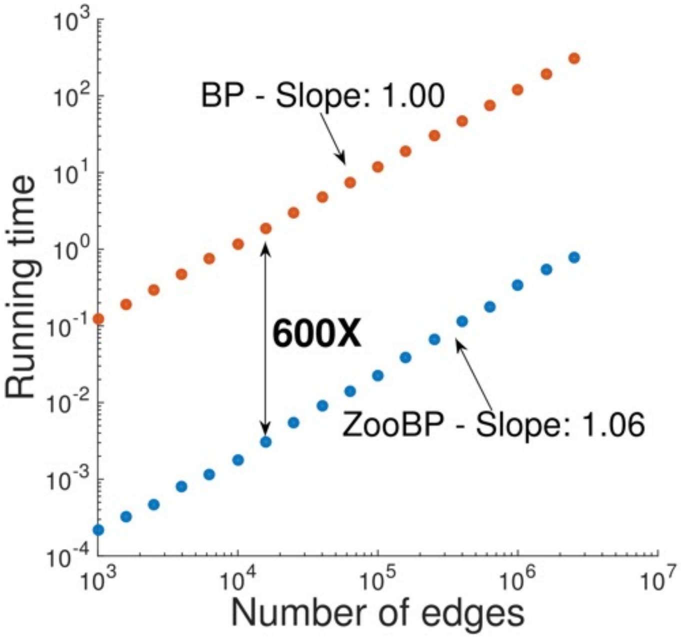
Dhivya Eswaran, Stephan Günnemann, Christos Faloutsos, Disha Makhija, Mohit Kumar, "*ZooBP:* Belief Propagation for Heterogeneous Networks", VLDB 2017

ZooBP: features

Fast; convergence guarantees.



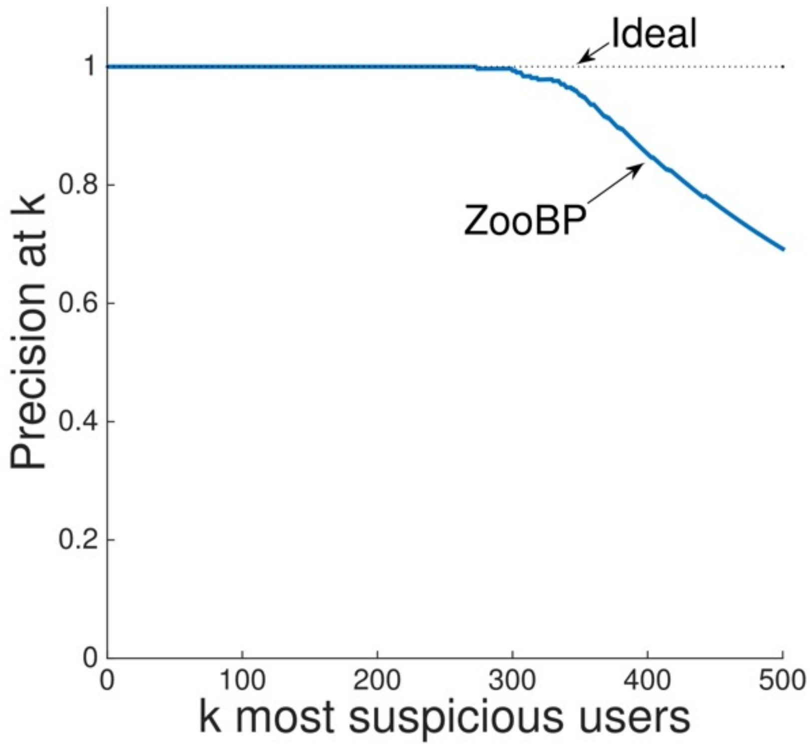
Near-perfect accuracy



linear in graph size

<u>Dhivya Eswaran</u>, Stephan Günnemann, Christos Faloutsos, Disha Makhija, Mohit Kumar, "*ZooBP:* Belief Propagation for Heterogeneous Networks", VLDB 2017

ZooBP in the real world

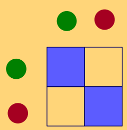


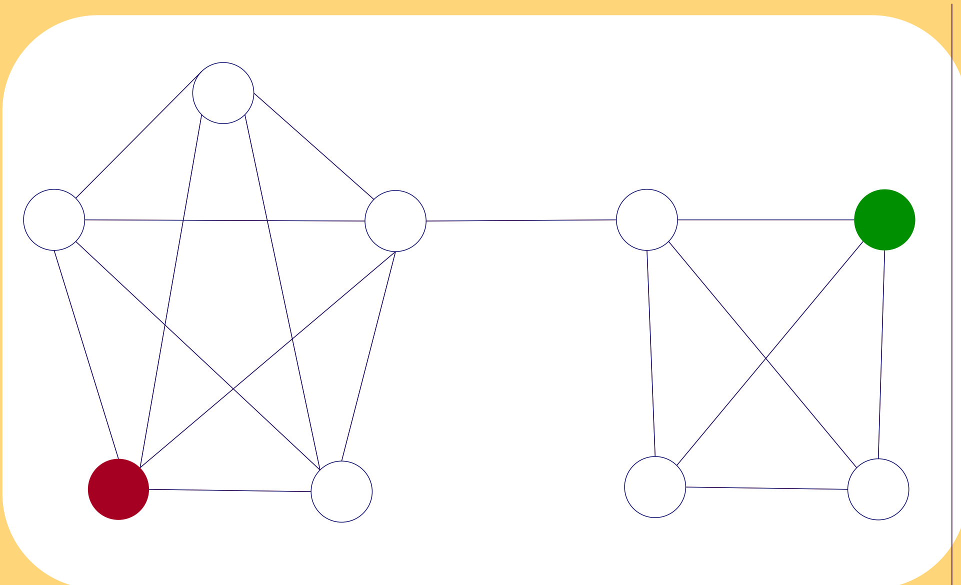
- Near 100% precision on top 300 users (Flipkart)
- Flagged users: suspicious
  - 400 ratings in 1 sec
  - 5000 good ratings and no bad ratings

<u>Dhivya Eswaran</u>, Stephan Günnemann, Christos Faloutsos, Disha Makhija, Mohit Kumar, "*ZooBP:* Belief Propagation for Heterogeneous Networks", VLDB 2017

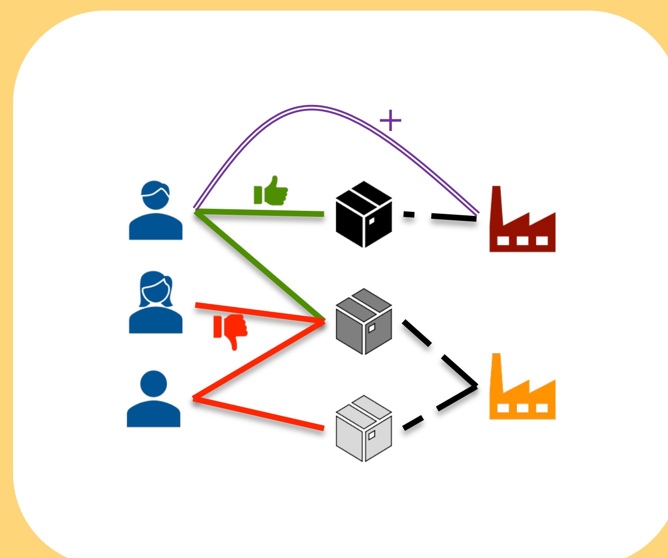
Short answer:

- What color, for the rest?
- A: Belief Propagation ('zooBP')





www.cs.cmu.edu/~deswaran/code/zoobp.zip



Roadmap

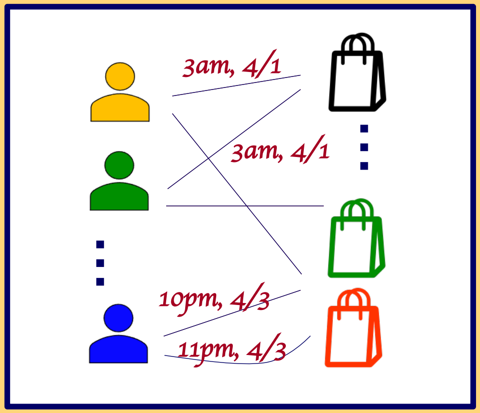
- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
- Part#2: Graph Mining – (semi-)supervised
- Part#3: Time-evolving graphs
- Part#4: Explanations
- Conclusions

Roadmap

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
- Part#2: Graph Mining – (semi-)supervised
- Part#3: Time-evolving graphs
  - 3.1. Tensors
  - 3.2. inter-arrival times
- …

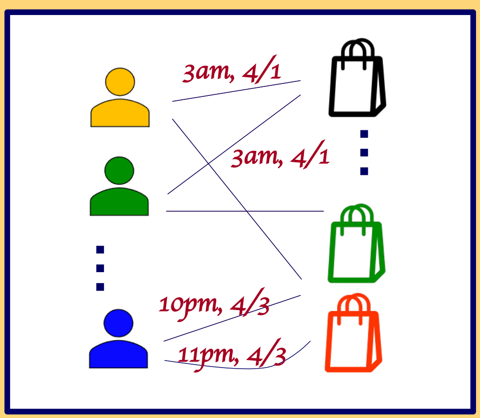
Problem

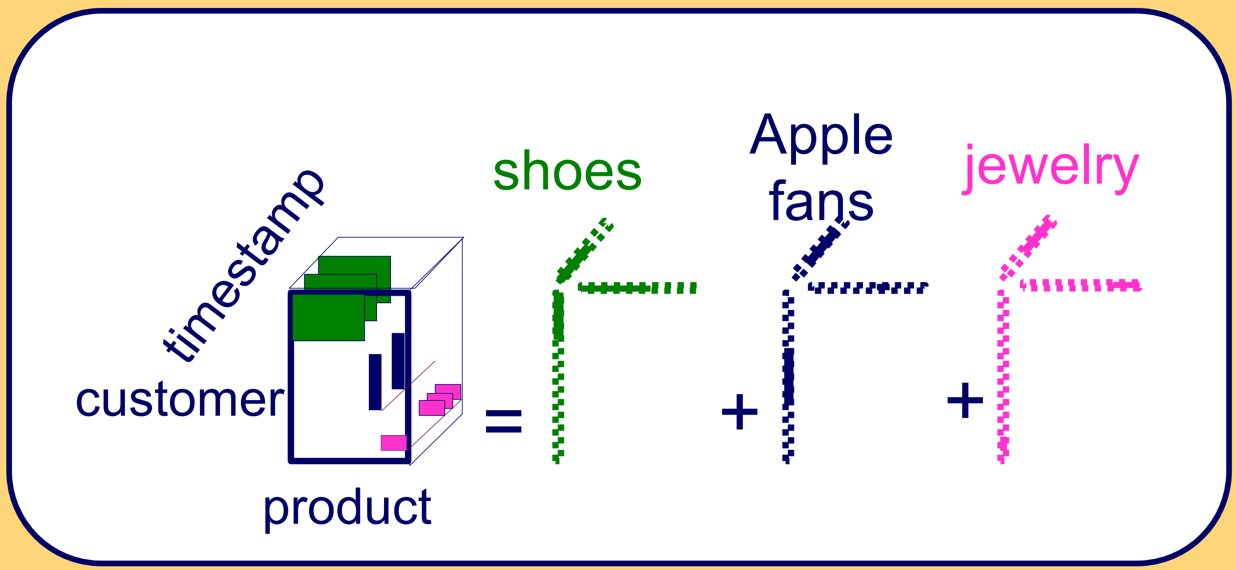
- Patterns/anomalies in time-evolving graphs?



Short answer:

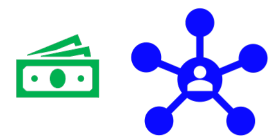
- Patterns/anomalies in time-evolving graphs?
- PARAFAC tensor decomposition

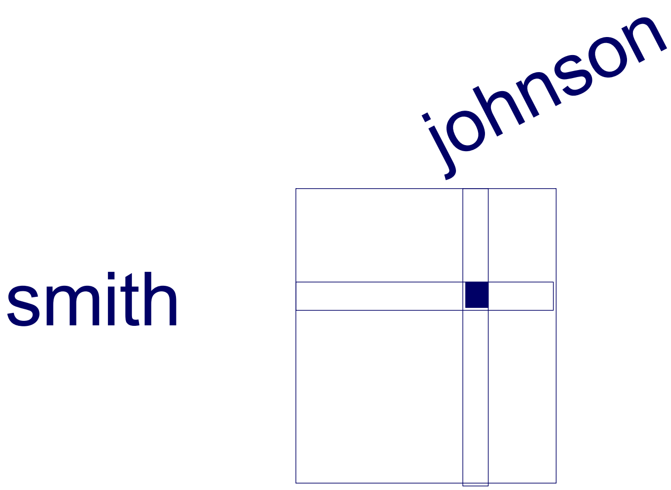




Graphs over time -> tensors!

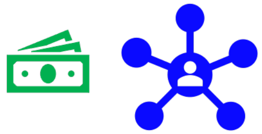
- Problem:
  - Given who calls whom, and **when**
  - Find patterns / anomalies

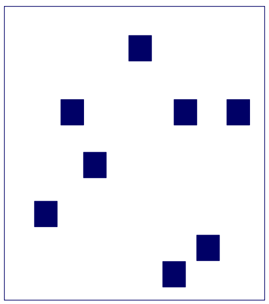




Graphs over time -> tensors!

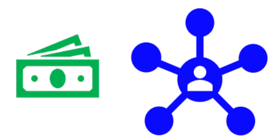
- Problem:
  - Given who calls whom, and **when**
  - Find patterns / anomalies

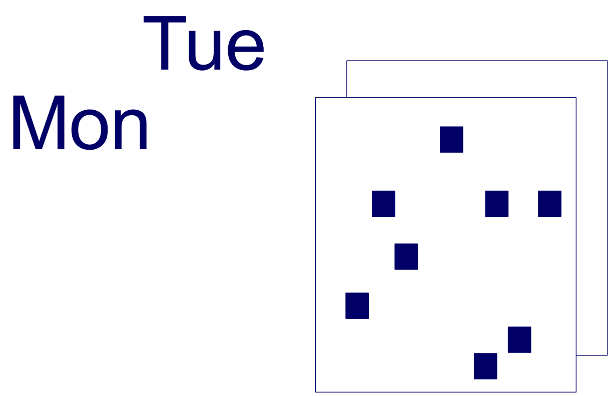




Graphs over time -> tensors!

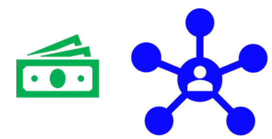
- Problem:
  - Given who calls whom, and **when**
  - Find patterns / anomalies

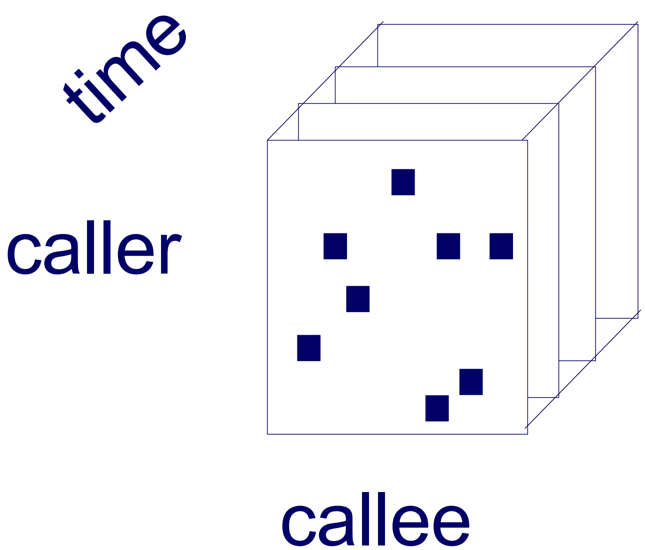




Graphs over time -> tensors!

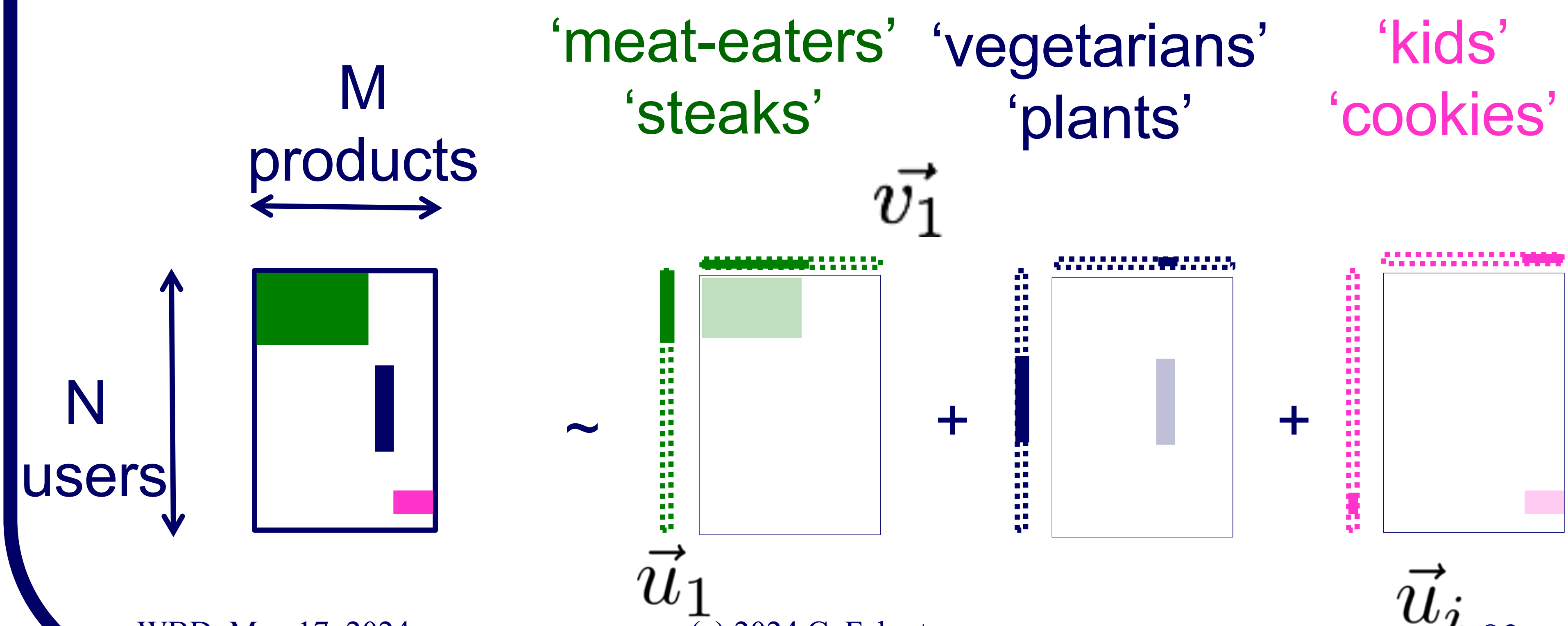
- Problem:
  - Given who calls whom, and **when**
  - Find patterns / anomalies





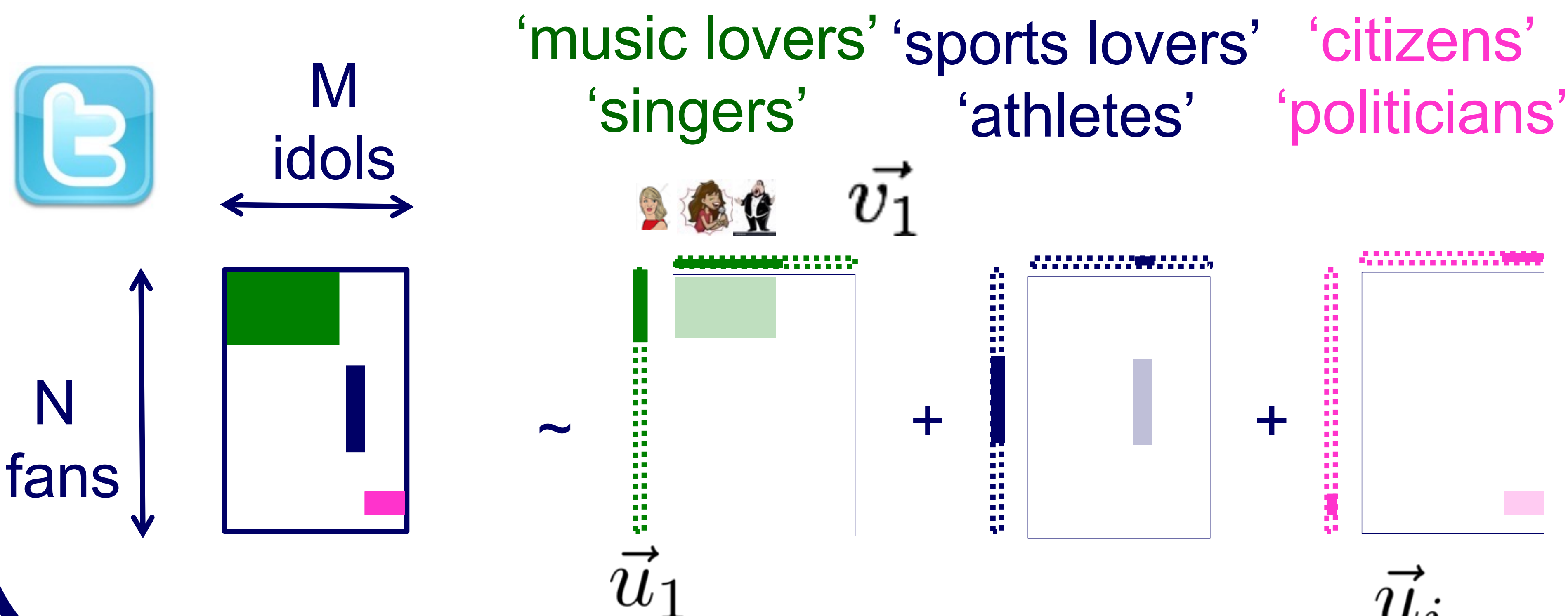
Answer : tensor factorization

- Recall: (SVD) matrix factorization: finds blocks



Crush intro to SVD

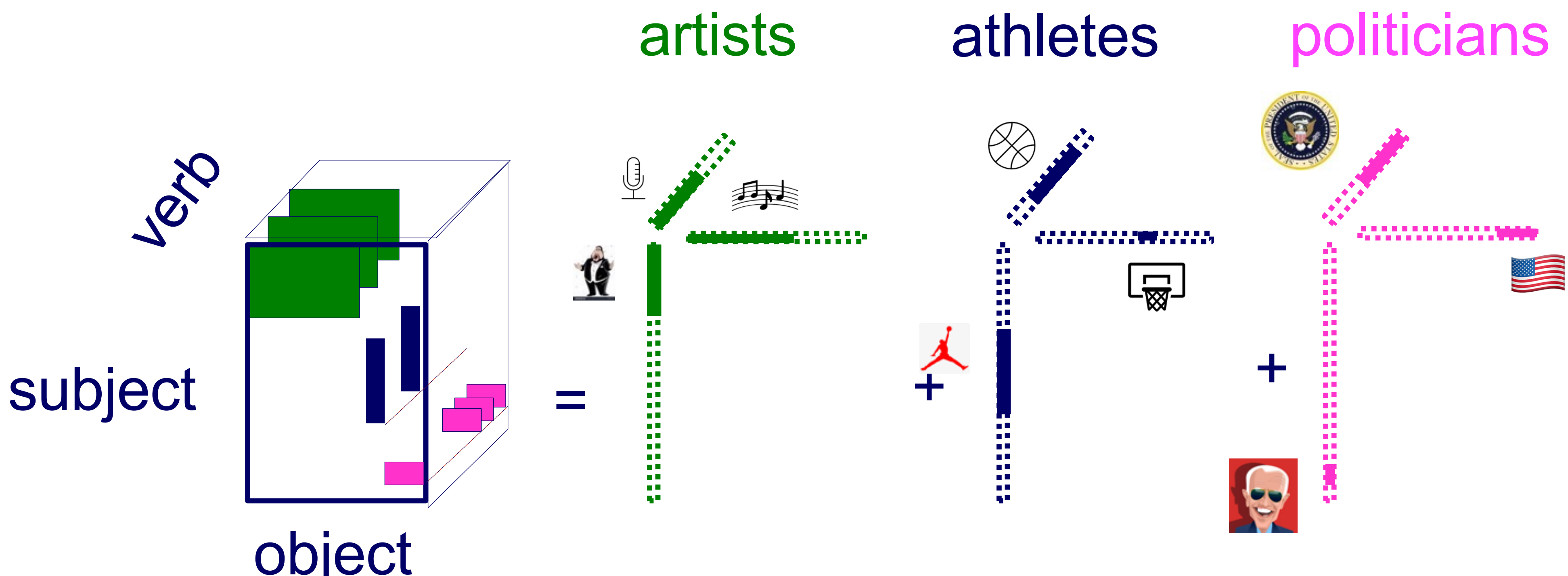
- Recall: (SVD) matrix factorization: finds blocks



Carnegie Mellon

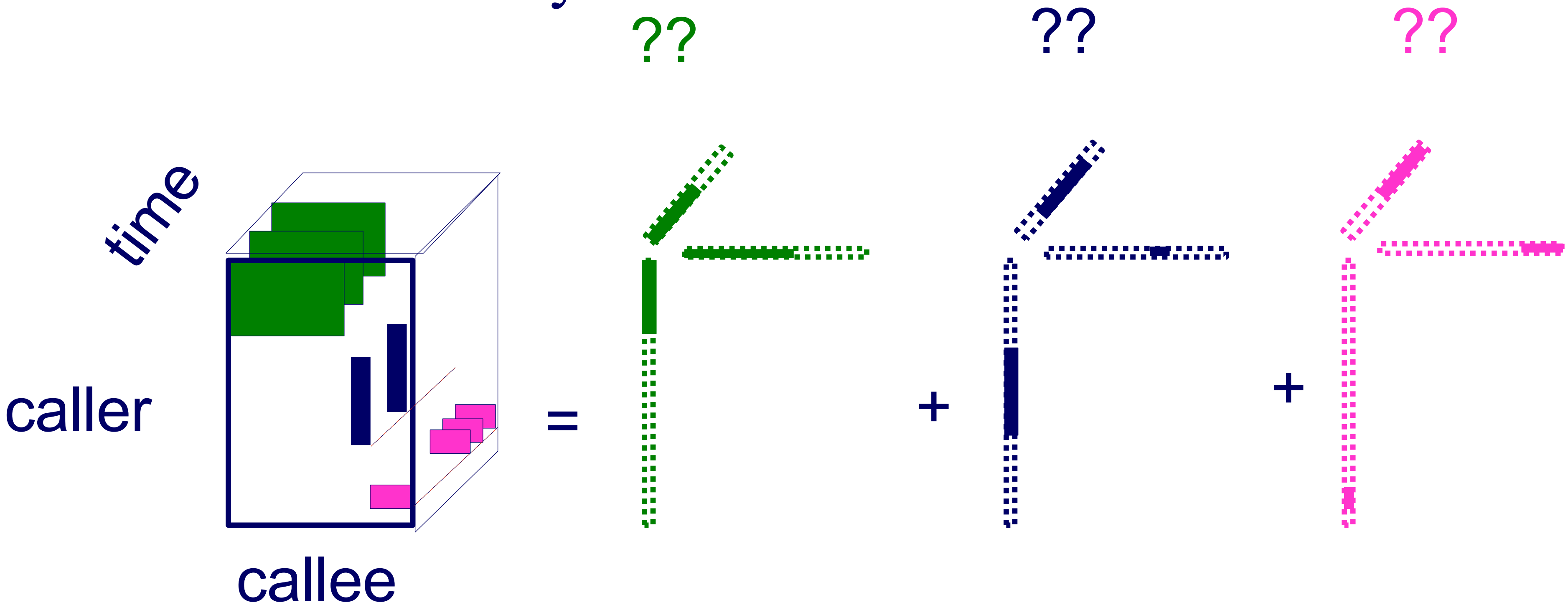
Answer: tensor factorization

- PARAFAC decomposition



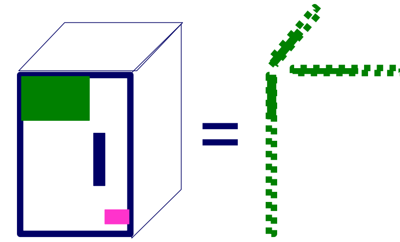
Answer: tensor factorization

- PARAFAC decomposition
- Results for who-calls-whom-when
  - 4M x 15 days



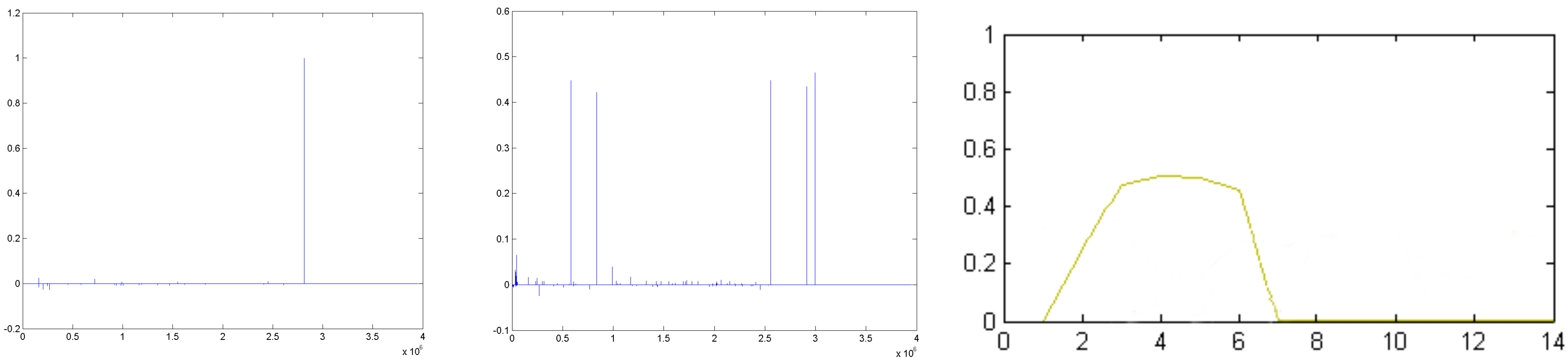
Carnegie Mellon

Anomaly detection in time-evolving graphs



- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks

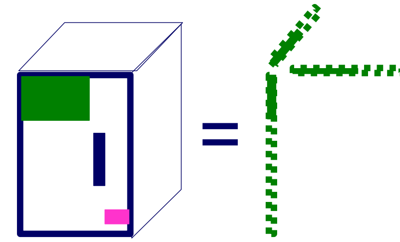
1 caller | 5 receivers | 4 days of activity



~200 calls to EACH receiver on EACH day!

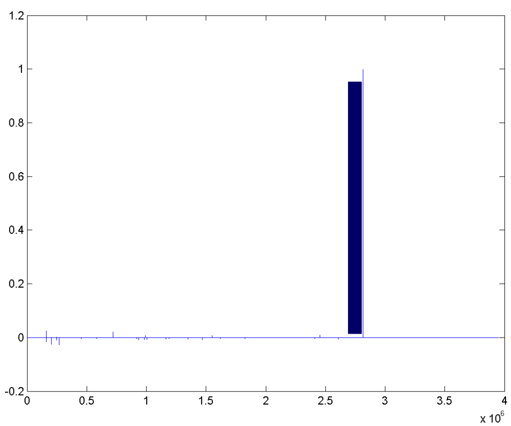
Carnegie Mellon

Anomaly detection in time-evolving graphs

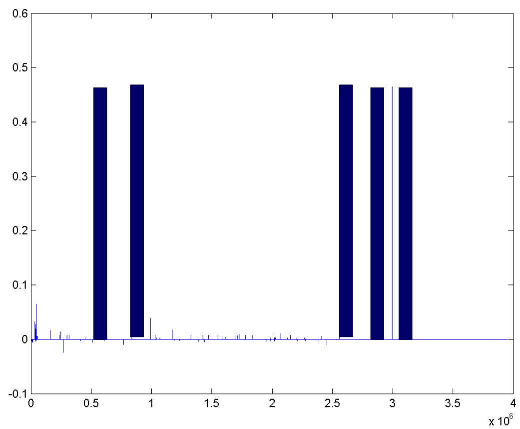


- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks

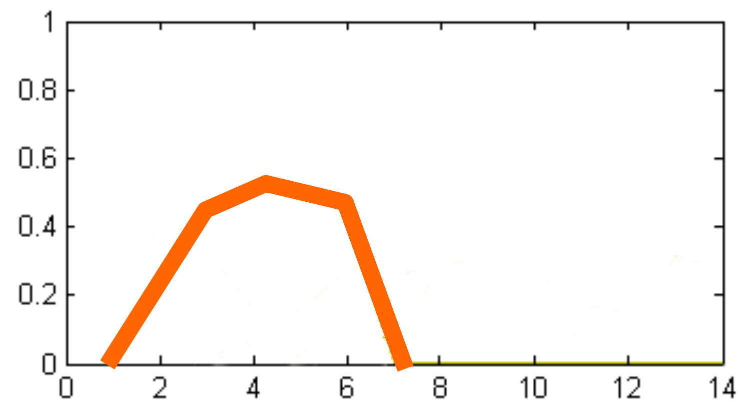
1 caller



5 receivers

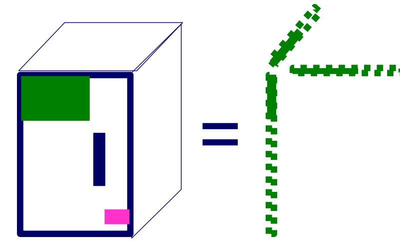


4 days of activity



~200 calls to EACH receiver on EACH day!

Anomaly detection in time-evolving graphs



- Anomalous communities in phone call data:
  - European country, 4M clients, data over 2 weeks

Miguel Araujo, Spiros Papadimitriou, Stephan Günnemann, Christos Faloutsos, Prithwish Basu, Ananthram Swami, Evangelos Papalexakis, Danai Koutra. *Com2: Fast Automatic Discovery of Temporal (Comet) Communities*. PAKDD 2014, Tainan, Taiwan.

Roadmap

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
- Part#2: Graph Mining – (semi-)supervised
- Part#3: Time-evolving graphs
  - 3.1. Tensors
  - 3.2. inter-arrival times
- …

Carnegie Mellon

Universidade de São Paulo

筑波大学
University of Tsukuba

Carnegie Mellon University

KDD 2015 – Sydney, Australia

RSC: Mining and Modeling Temporal Activity in Social Media

Alceu F. Costa* Yuto Yamaguchi Agma J. M. Traina

Caetano Traina Jr. Christos Faloutsos

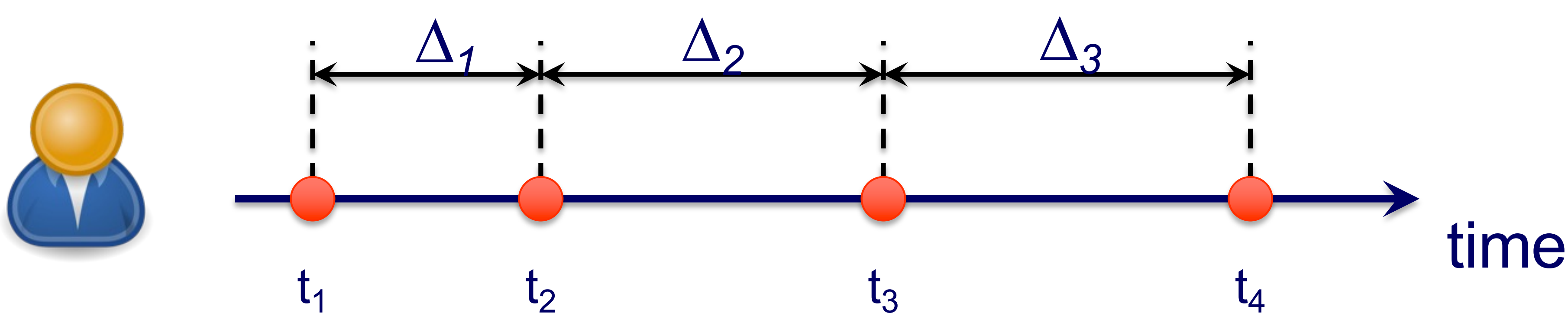
*alceufc@icmc.usp.br

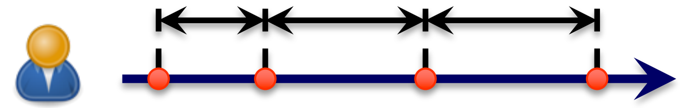
Pattern Mining: Datasets

Reddit Dataset	Twitter Dataset
Time-stamp from comments	Time-stamp from tweets
21,198 users	6,790 users
20 Million time-stamps	16 Million time-stamps

For each user we have:

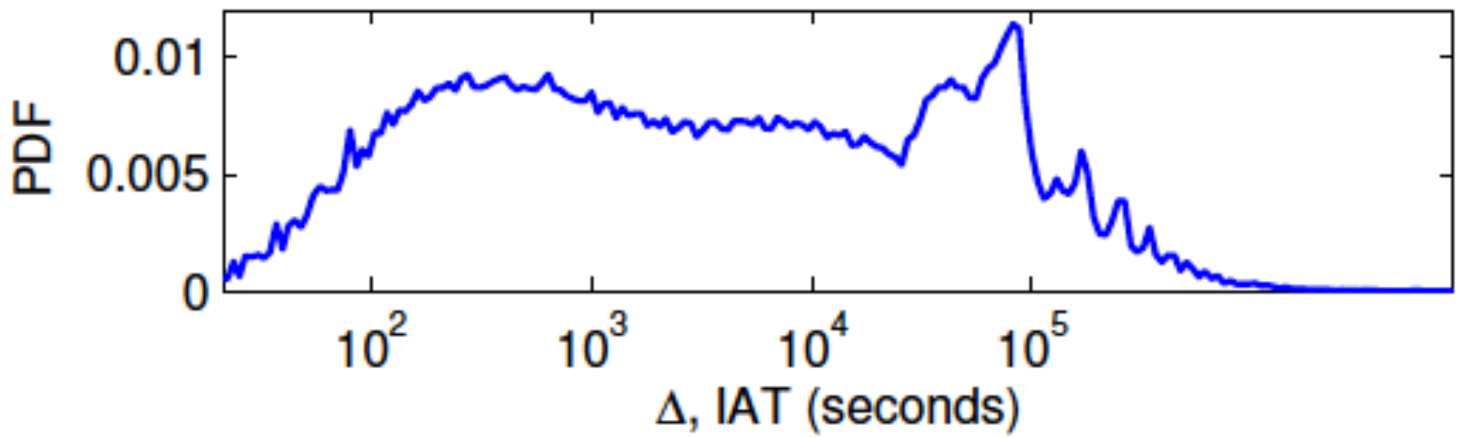
- Sequence of postings time-stamps: $\boldsymbol{T} = (t_1, t_2, t_3, \ldots)$
- Inter-arrival times (IAT) of postings: $(\Delta_1, \Delta_2, \Delta_3, \ldots)$



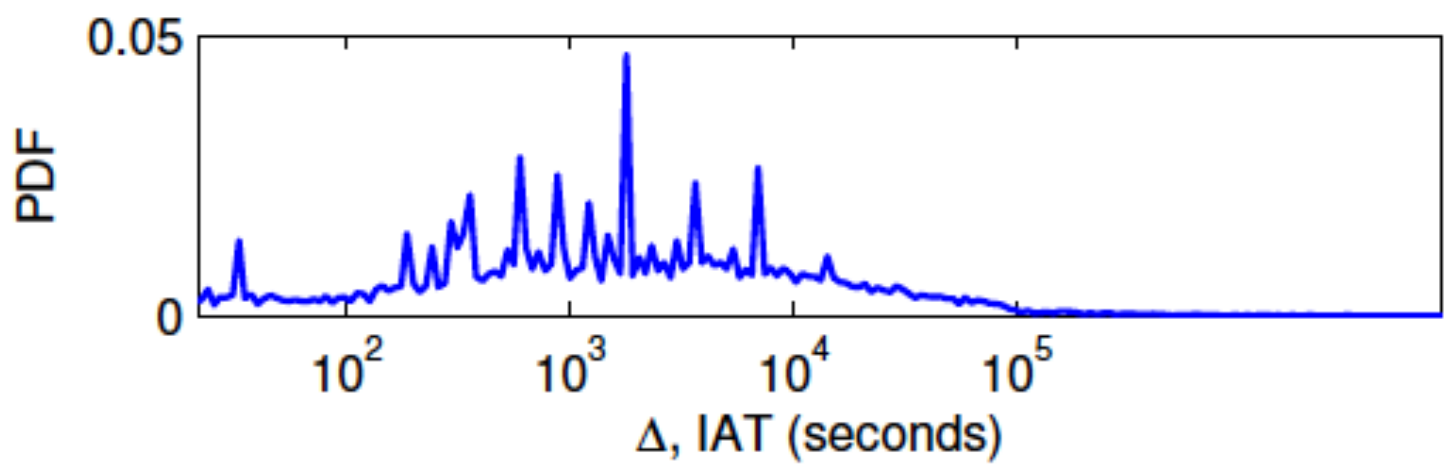


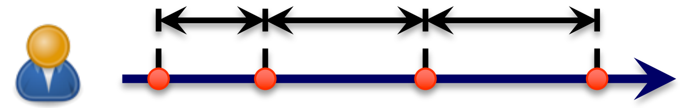
Human? Robots?

linear

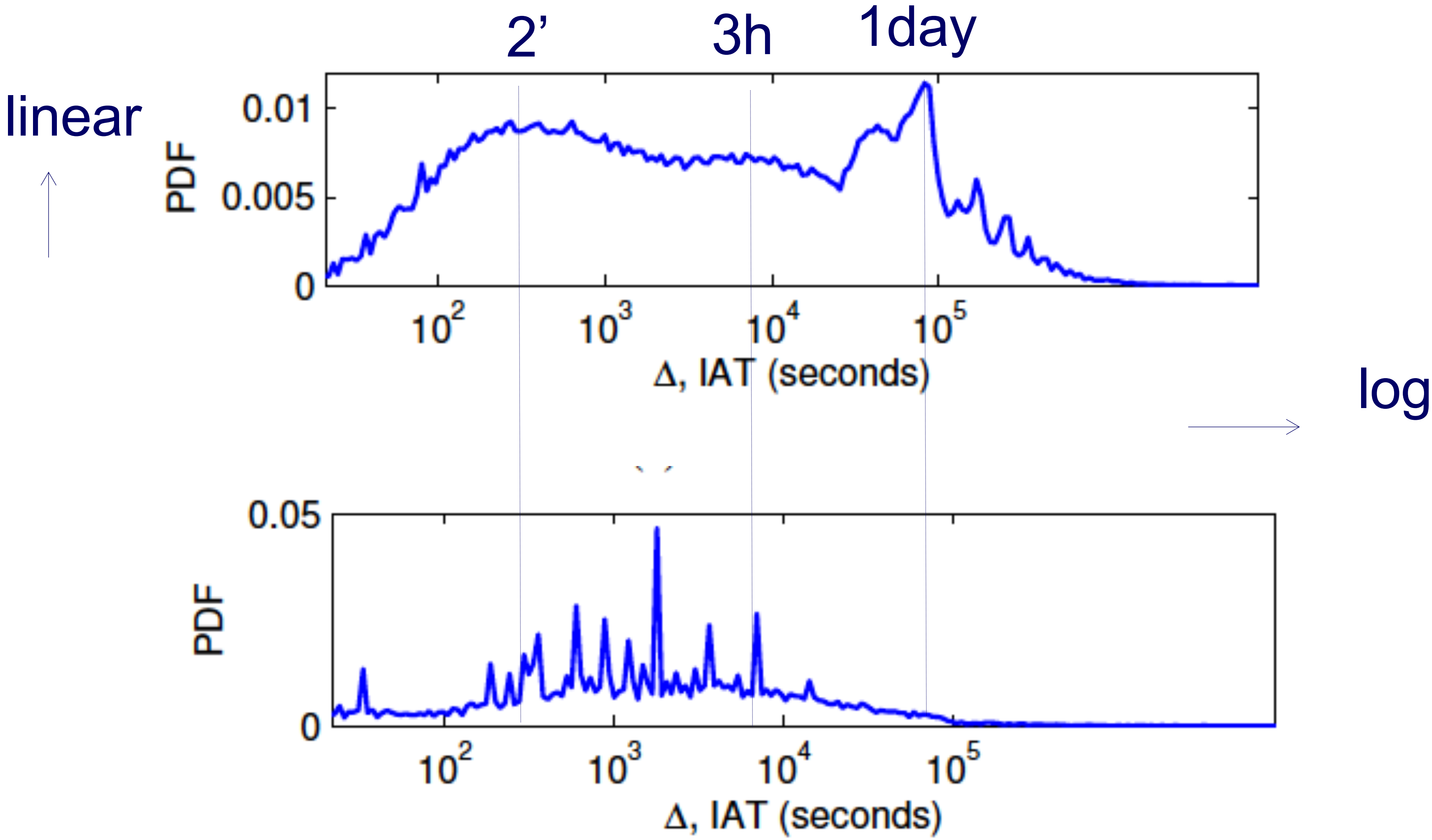


log





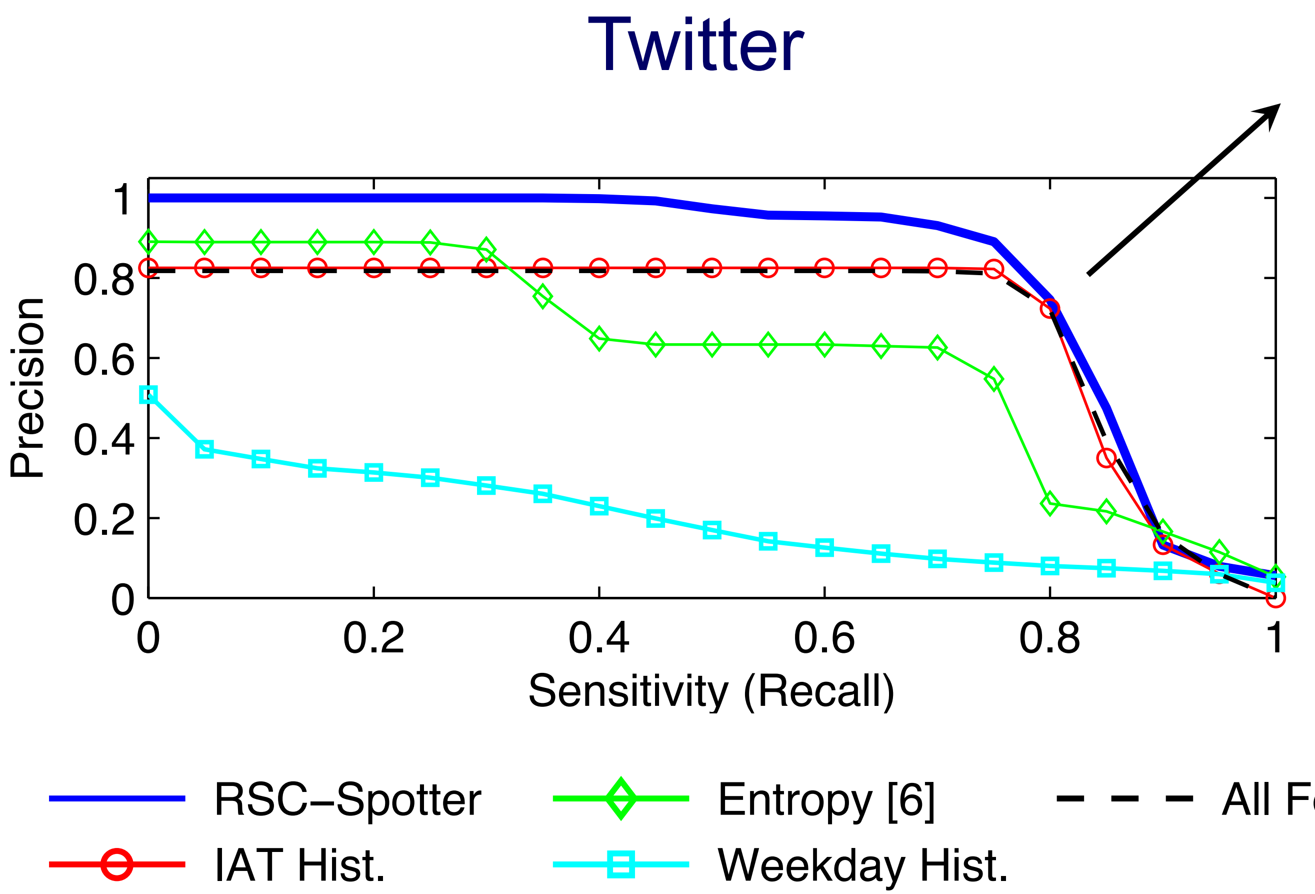
Human? Robots?



Experiments: Can RSC-Spotter Detect Bots?

Precision vs. Sensitivity Curves

Good performance: curve close to the top



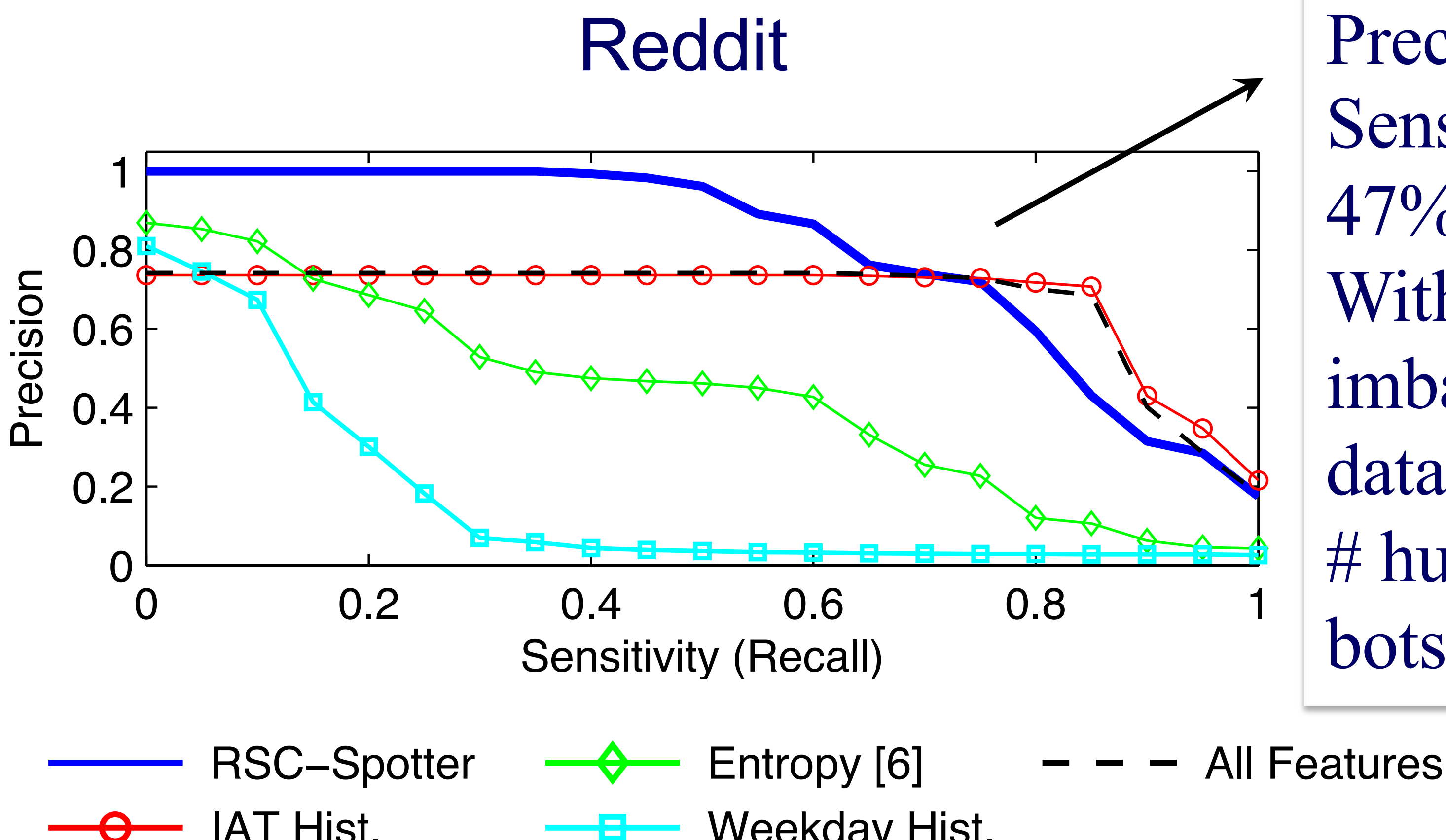
Precision > 94%
Sensitivity > 70%

With strongly imbalanced datasets
humans >> # bots

Experiments: Can RSC-Spotter Detect Bots?

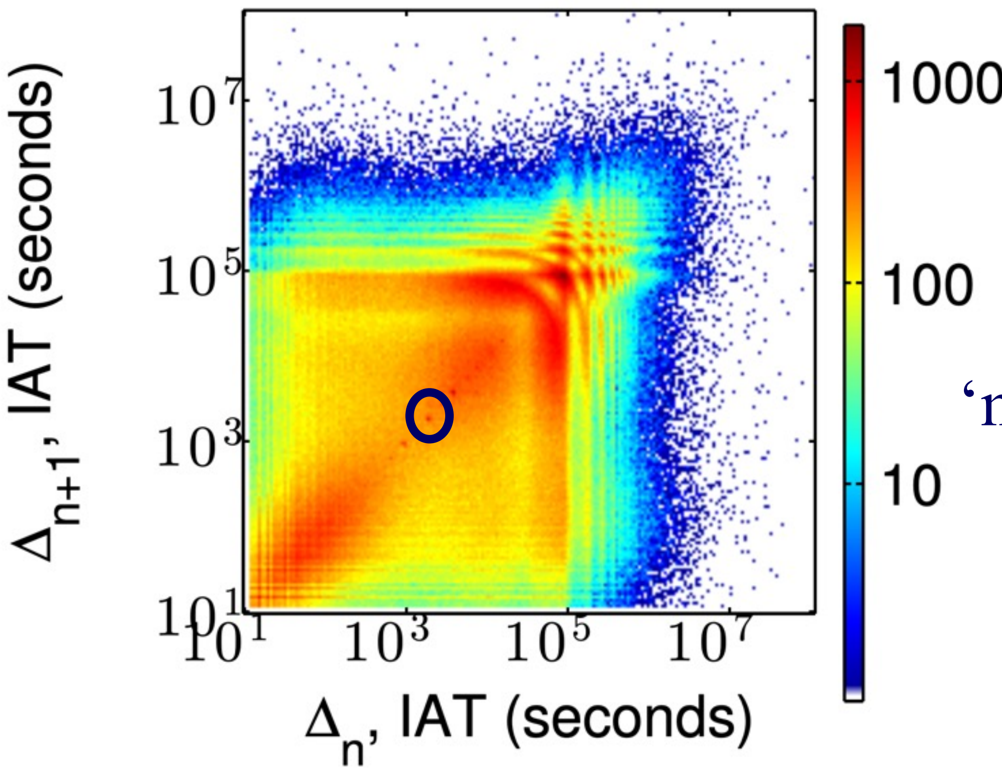
Precision vs. Sensitivity Curves

Good performance: curve close to the top



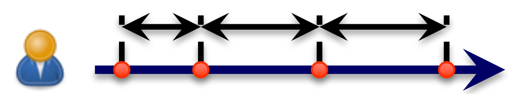
Precision > 96%
Sensitivity > 47%
With strongly imbalanced datasets
humans >> # bots

'Delay map'



'machine-gun' behavior

(b) Twitter



Roadmap

- Introduction – Motivation
- Part#1: Graph Mining – unsupe
- Part#2: Graph Mining – (semi-)supervised
- Part#3: Time-evolving graphs
  - 3.1. Tensors
  - 3.2. inter-arrival times
  - 3.3. Forecasting
- …

AutoGluon TS

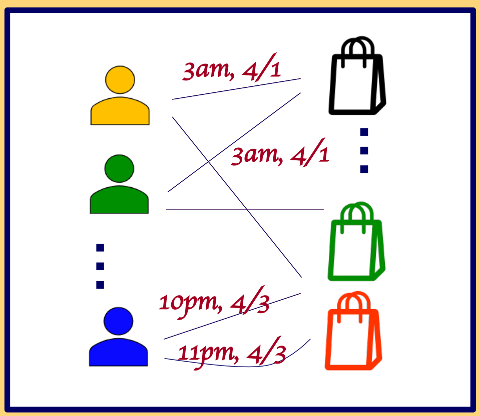
- https://auto.gluon.ai/stable/tutorials/timeseries/index.html

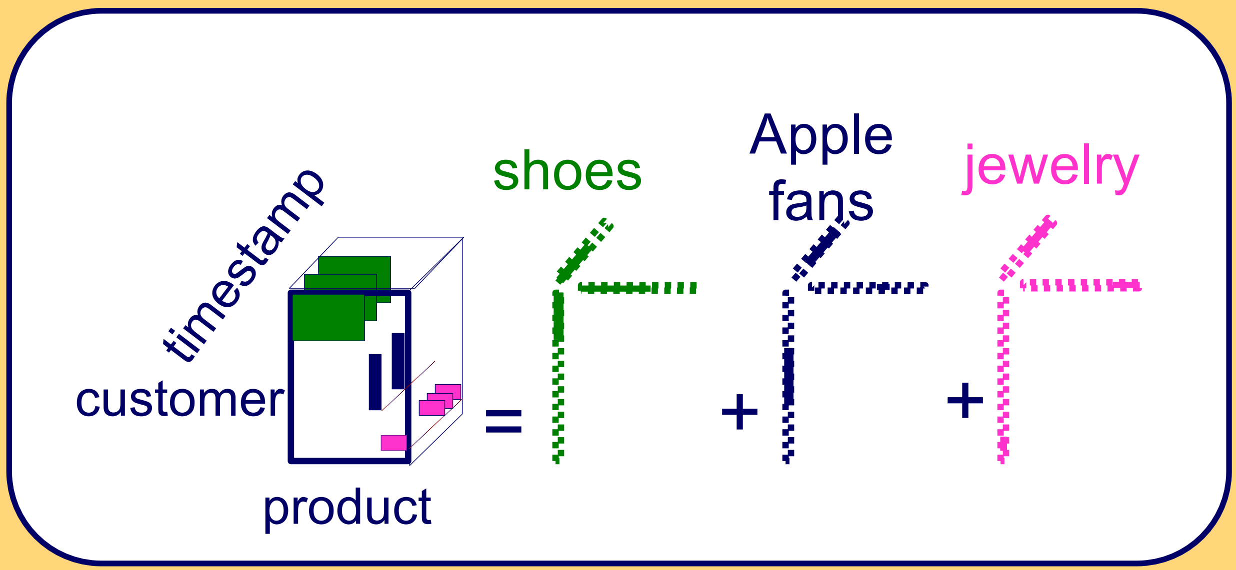
```
from autogluon.timeseries import *
fit()
```

AutoGluon

Short answer:

- Patterns/anomalies in time-evolving graphs?
- PARAFAC tensor decomposition





Roadmap

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
- Part#2: Graph Mining – (semi-)supervised
- Part#3: Time-evolving graphs
- Part#4: Explanations / Visualization
- Conclusions

Carnegie Mellon

TgraphSpot: Fast and Effective Anomaly Detection for Time-Evolving Graphs

IEEE BigData, 2022

Mirela Cazzolato[1,2], Saranya Vijayakumar[1], Xinyi Zheng[1], Namyong Park[1], Meng-Chieh Lee[1], Pedro Fidalgo[3,4], Bruno Lages[3], Agma J. M. Traina[2], Christos Faloutsos[1]

Carnegie Mellon

USP

iscte INSTITUTO UNIVERSITÁRIO DE LISBOA

Open source:

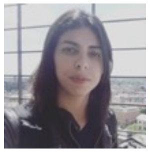
`https://github.com/mtcazzolato/tgraph-spot`

Video: https://youtu.be/jI1adN-BQuo?t=1537

Carnegie Mellon

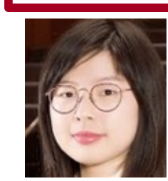
Authors

Carnegie Mellon USP iscte INSTITUTO UNIVERSITÁRIO DE LISBOA

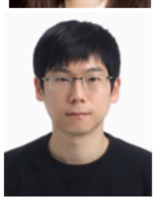


Mirela Cazzolato

Saranya Vijayakumar



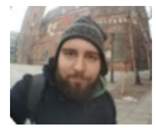
Xinyi Zheng



Namyong Park

Meng-Chieh Jeremy Lee

Pedro Fidalgo

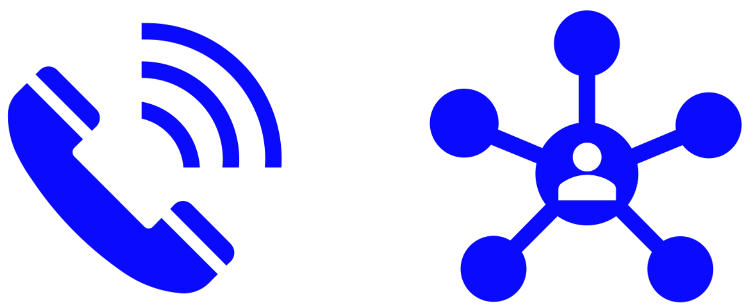


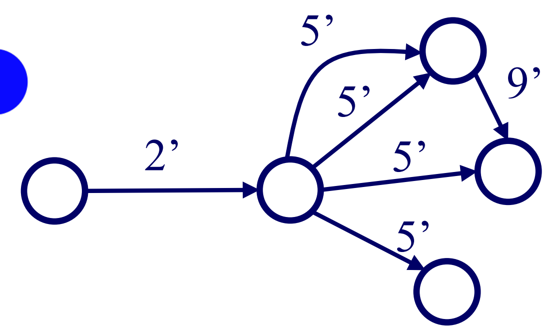
Bruno Lages

Agma Traina

Christos Faloutsos

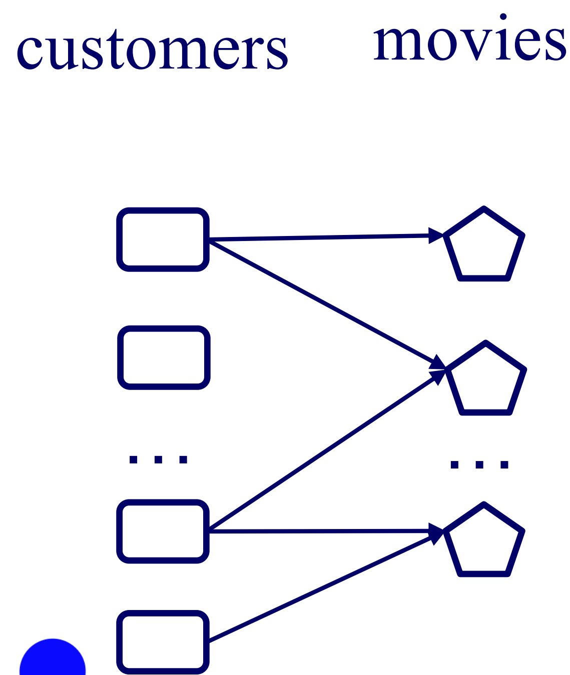
Problem definition

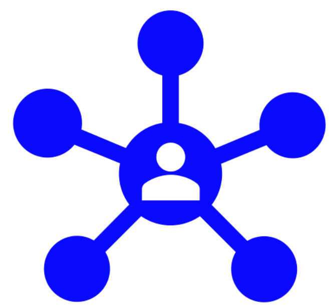


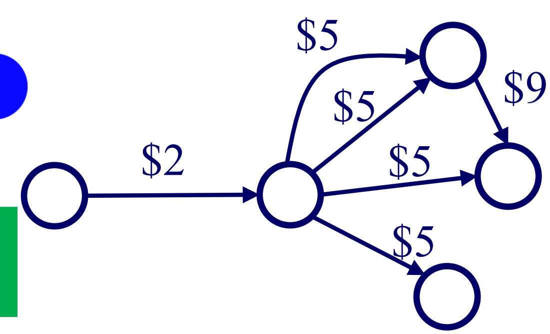


(source, destination, timestamp, duration)

Problem definition





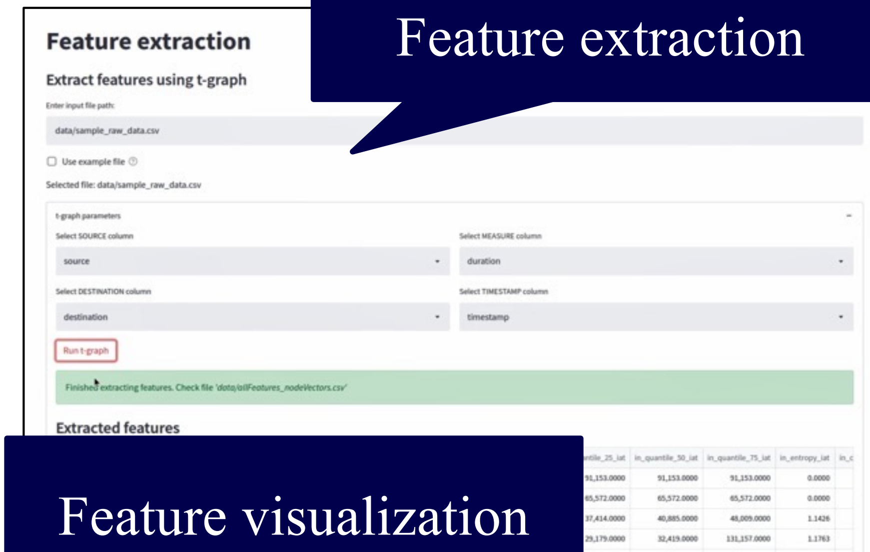


(source, destination, timestamp, $amount)

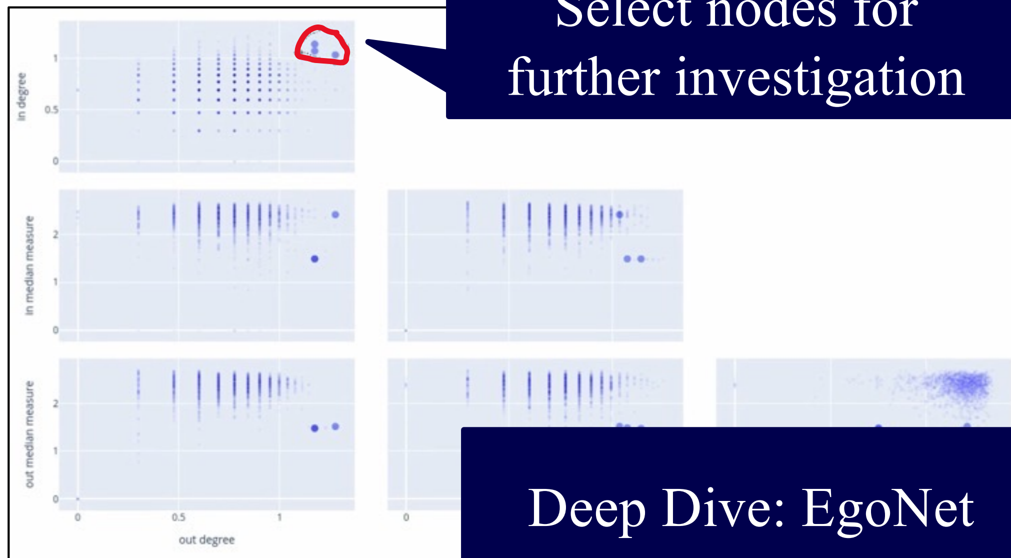
Carnegie Mellon

System Overview - current

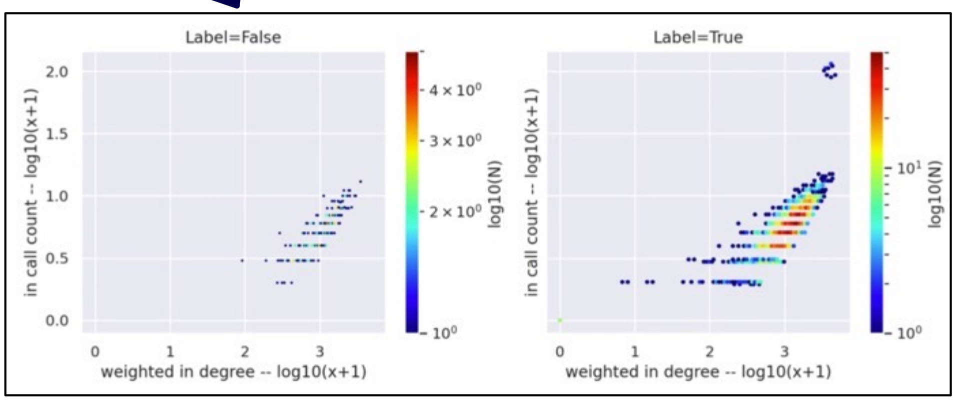
Feature extraction



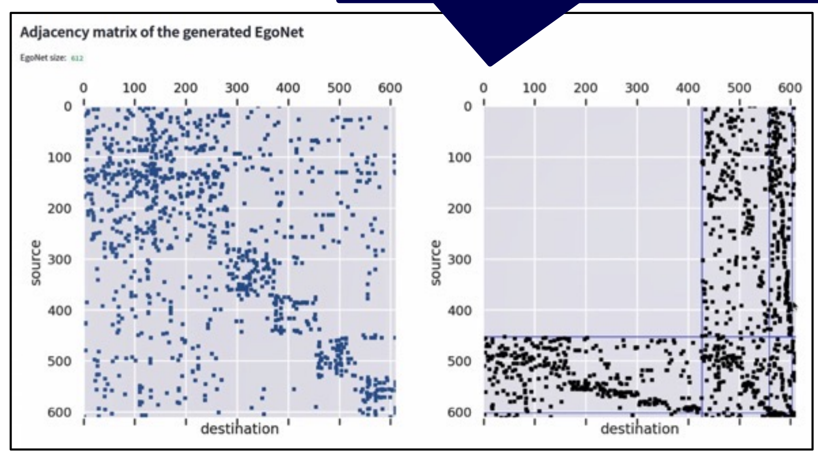
Select nodes for further investigation



Feature visualization



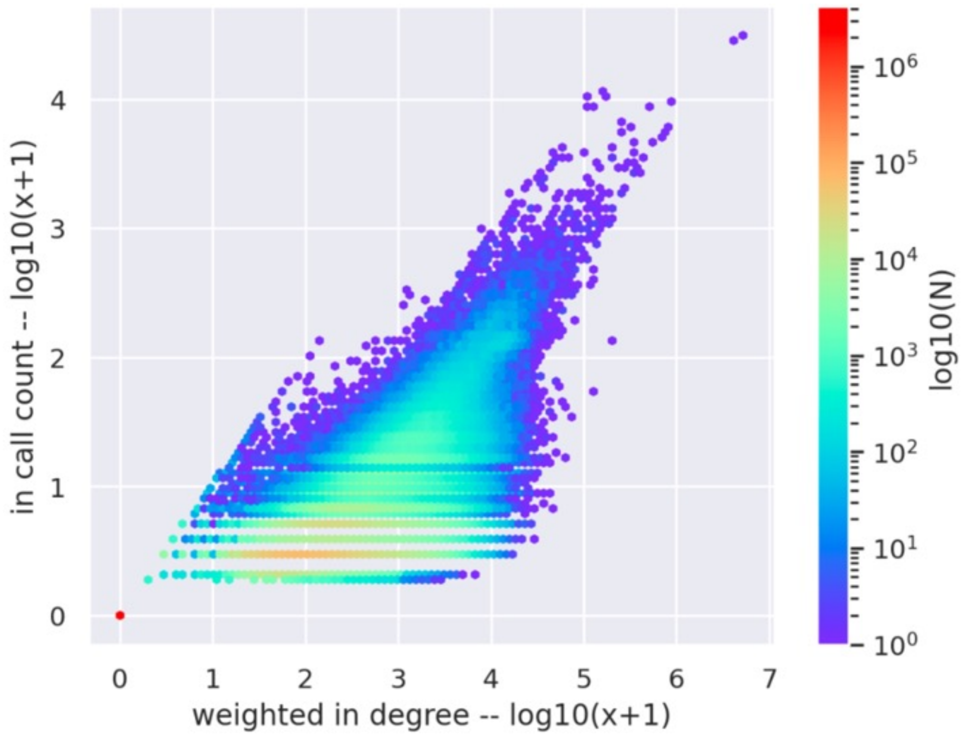
Deep Dive: EgoNet



Video: https://youtu.be/jI1adN-BQuo?t=1537

Discovery #1

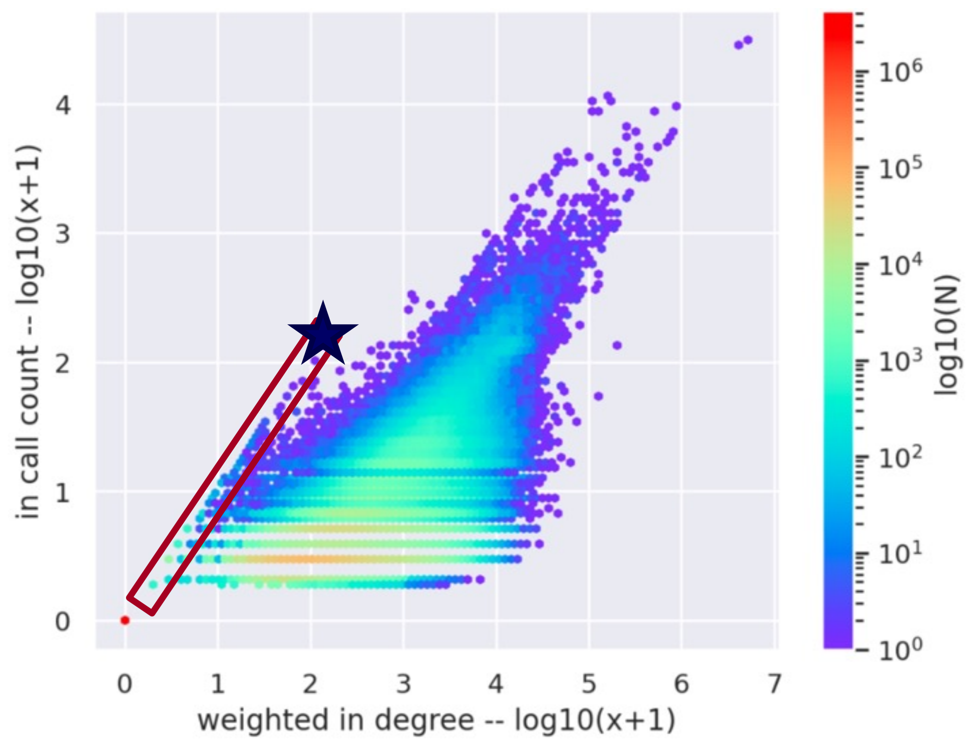
in-degree



Weighted in-degree (= in-seconds)

Discovery #1

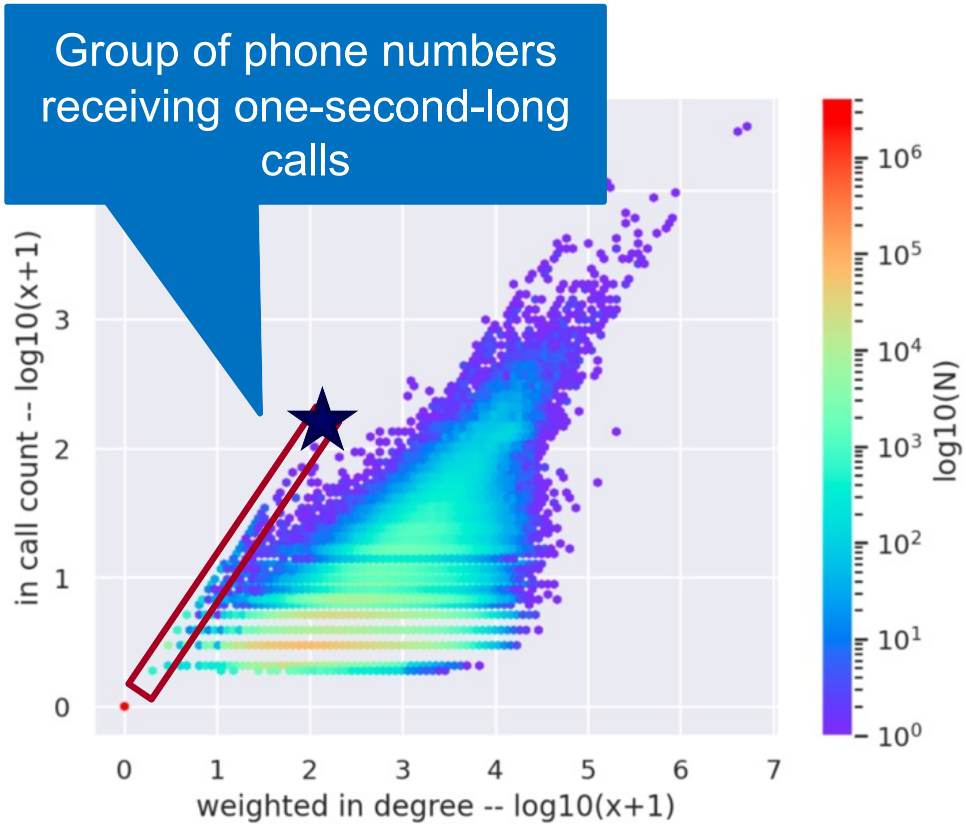
in-degree



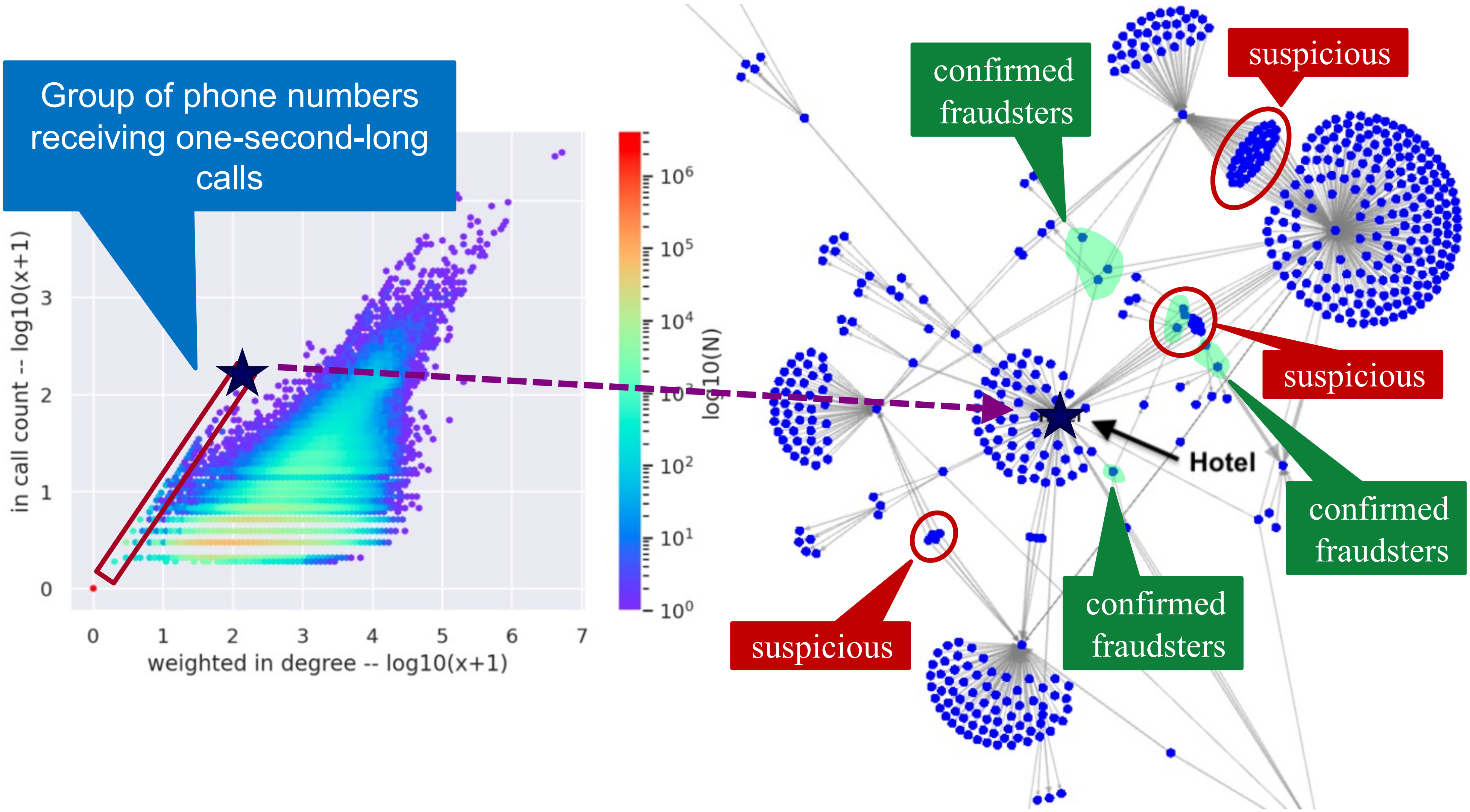
Weighted in-degree (= in-seconds)

Discovery #1

100 in-calls
100 seconds



Discovery #1


Group of phone numbers receiving one-second-long calls
in call count -- log10(x+1)
weighted in degree -- log10(x+1)
log10(N)
10^0
10^1
10^2
10^4
10^5
10^6
0
1
2
3
4
5
6
7
confirmed fraudsters
suspicious
suspicious
Hotel
confirmed fraudsters
suspicious
confirmed fraudsters

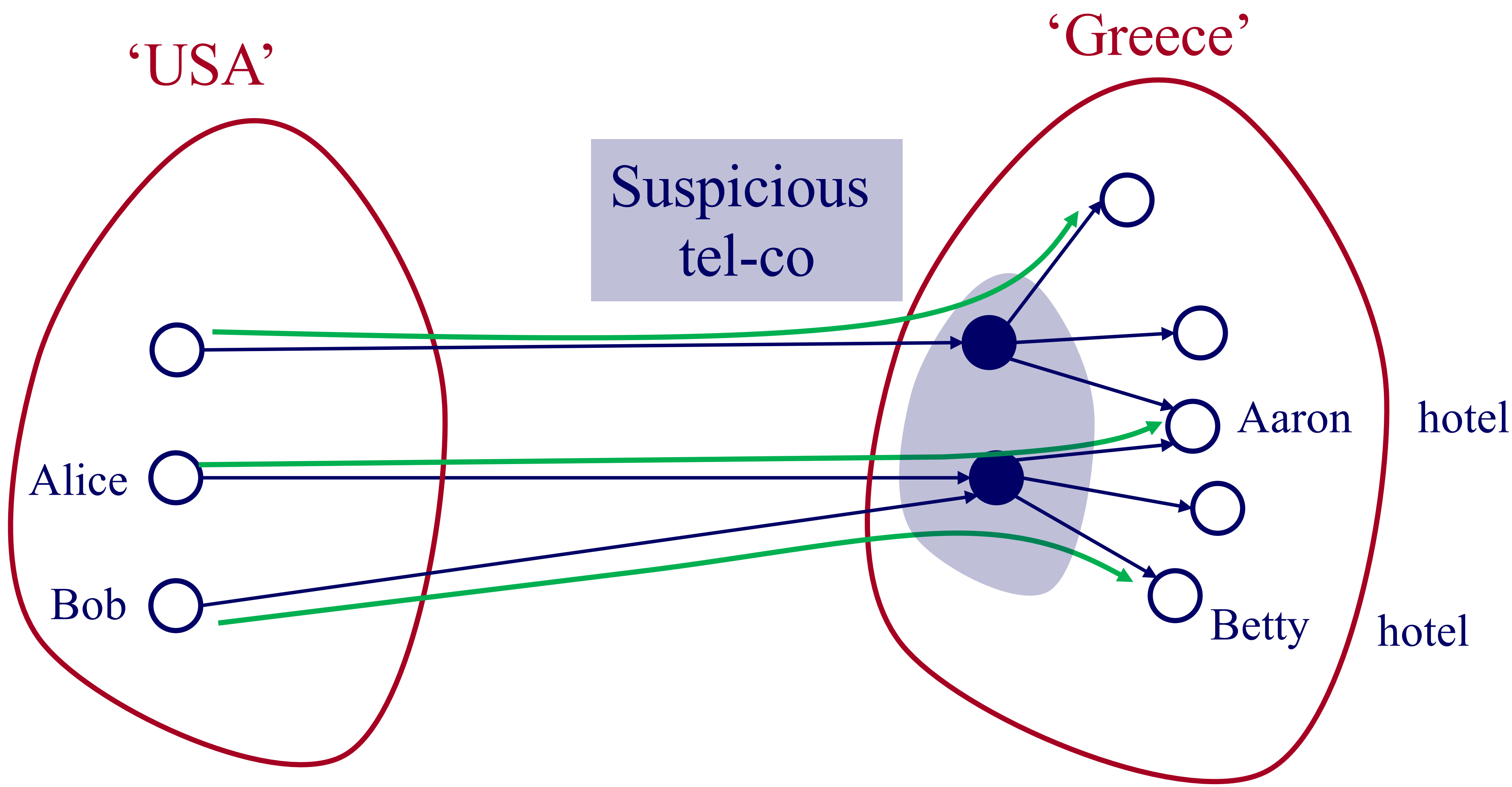
Q: Why?

- Q: Why would people call hotel-like numbers, for 1second?

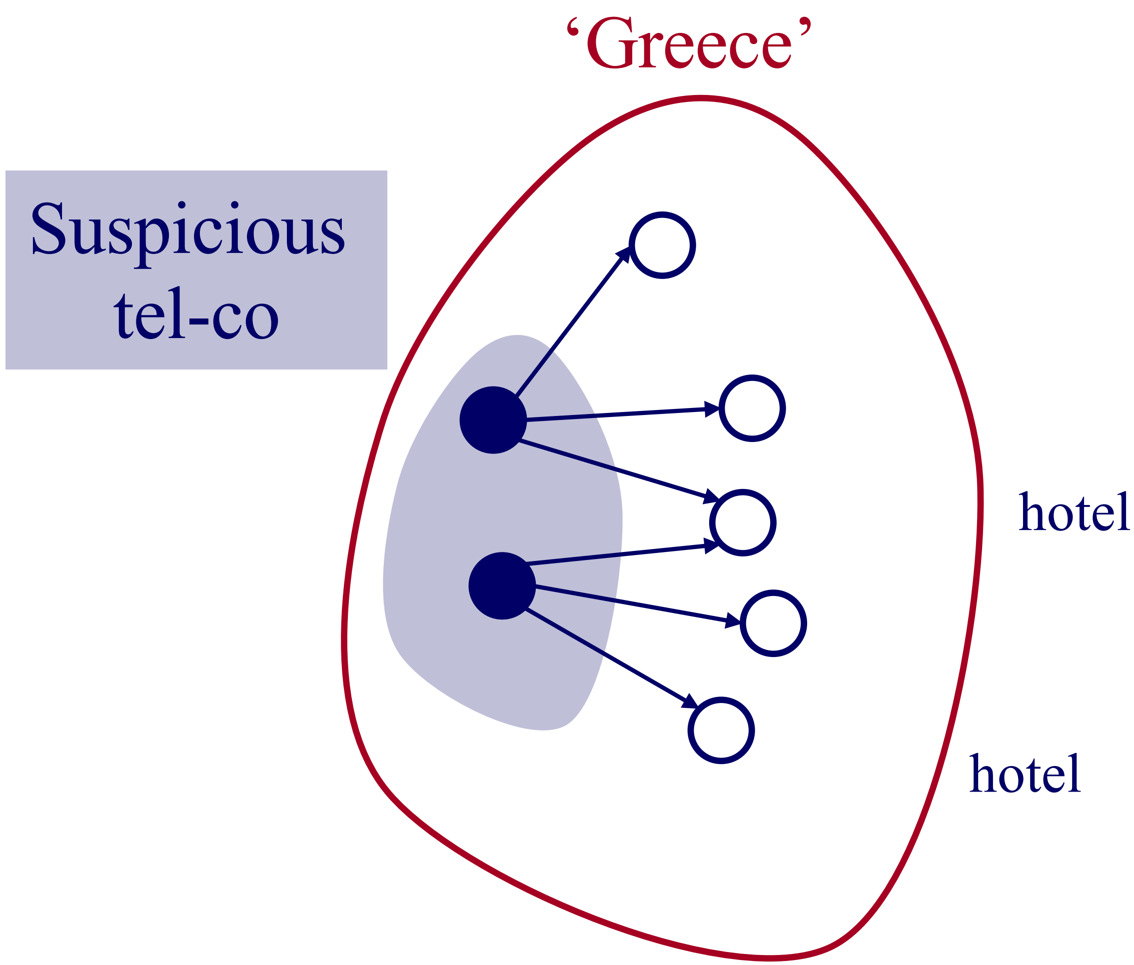
Q: Why?

- Q: Why would people call hotel-like numbers, for 1second?
- A: low quality/ low price, gray-area international carrier, that drops a lot of phonecalls

A: ‘international by-pass’



A: 'international by-pass'

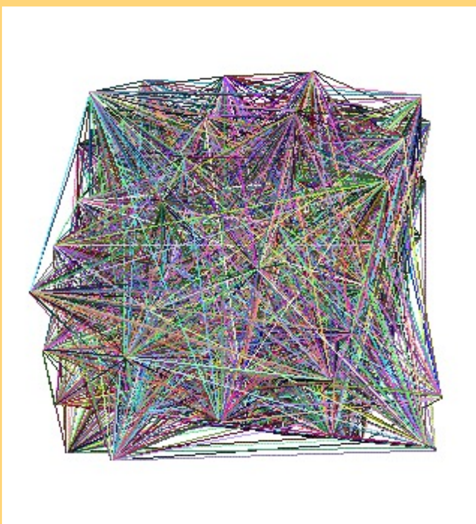


Roadmap

- Introduction – Motivation
  - Why study (big) graphs?
- Part#1: Graph Mining – unsupervised
- Part#2: Graph Mining – (semi-)supervised
- Part#3: Time-evolving graphs
- Part#4: Explanations
- Conclusions

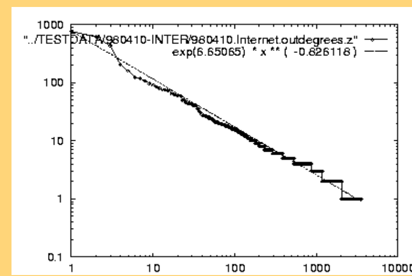
CONCLUSION#1: many patterns

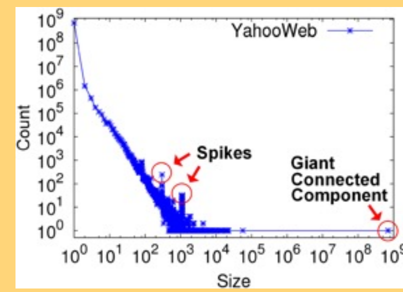
Given:

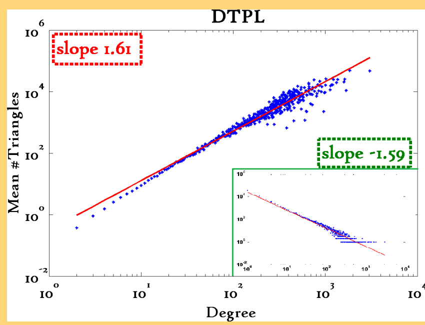


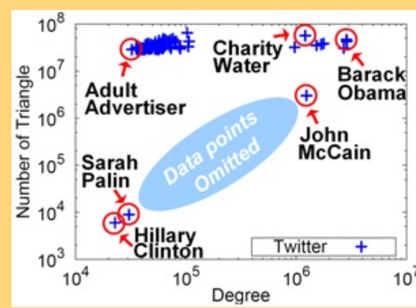
Find patterns ('what is normal')

6-degrees





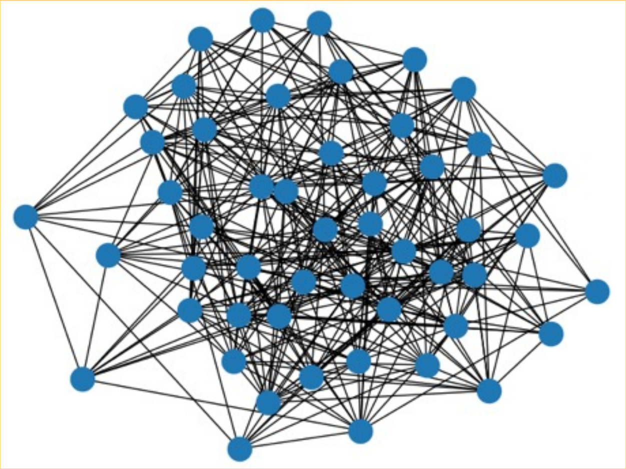




. . .

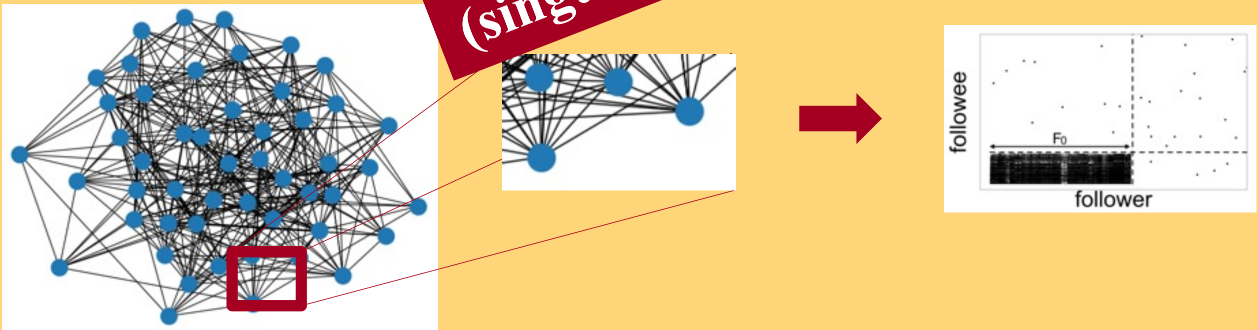
CONCLUSION#1’: Many tools

Given:



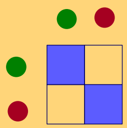
Find: suspicious sub-graphs

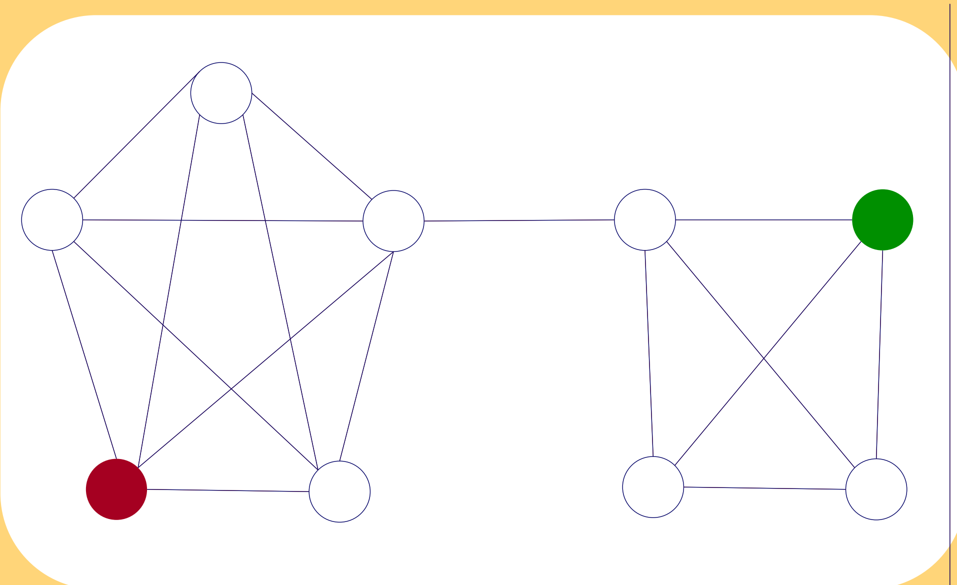
SVD
(singular value decomposition)



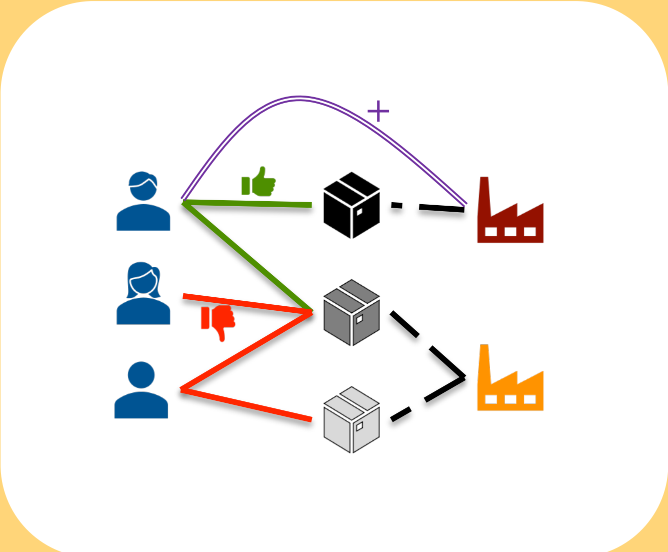
CONCLUSION#2: (zoo)BP

- What color, for the rest?
- A: Belief Propagation ('zooBP')



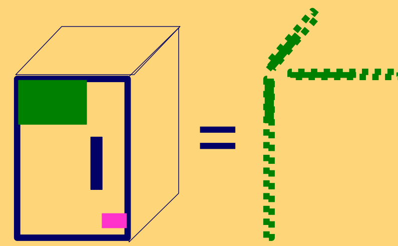


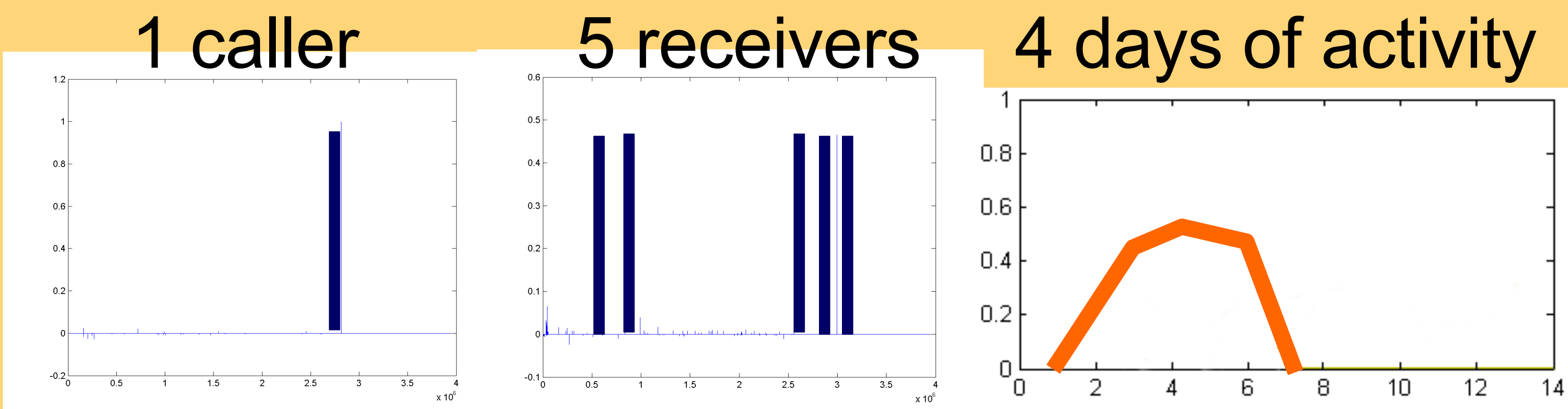
www.cs.cmu.edu/~deswaran/code/zoobp.zip



CONCLUSION#3 – tensors

- powerful tool

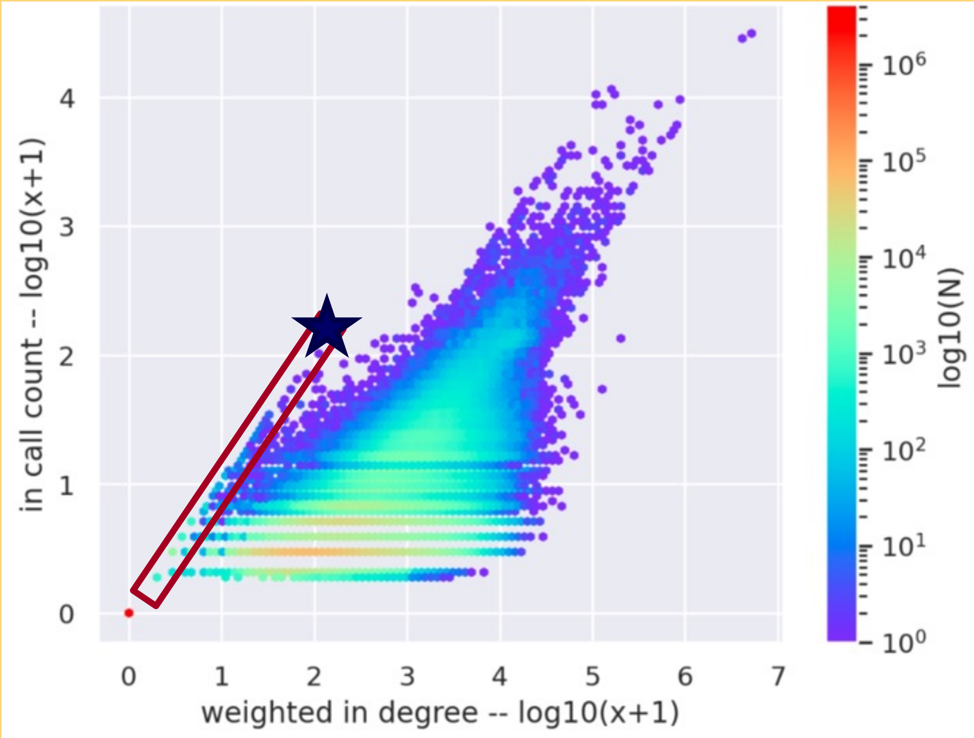




Carnegie Mellon

CONCLUSION#4 - visualization

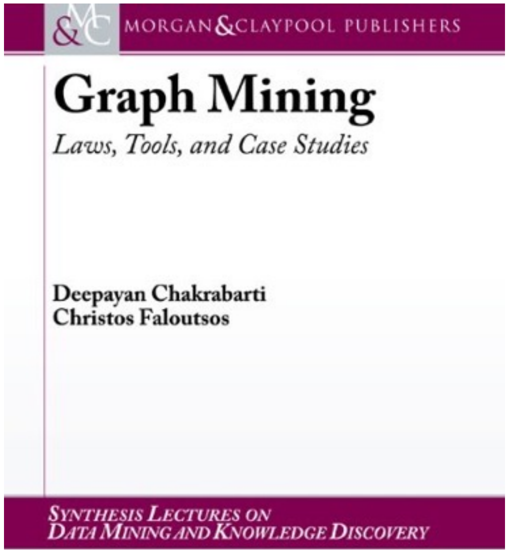
in-degree



Weighted in-degree (= in-seconds)

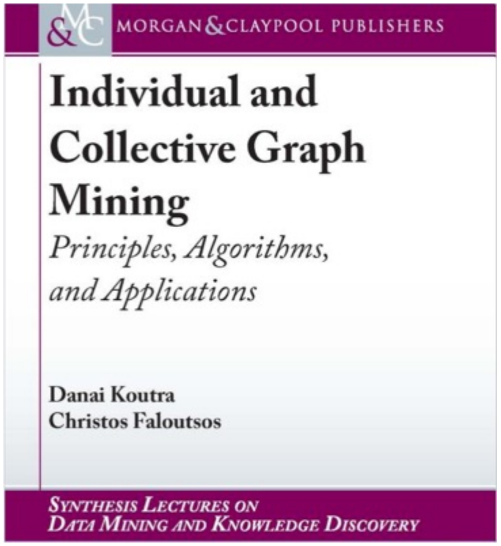
References

- D. Chakrabarti, C. Faloutsos: *Graph Mining – Laws, Tools and Case Studies*, Morgan Claypool 2012
- https://link.springer.com/book/10.1007/978-3-031-01903-6
- Earlier version – Survey



References

- Danai Koutra and Christos Faloutsos, *Individual and Collective Graph Mining: Principles, Algorithms, and Applications, Springer, 2017* https://link.springer.com/book/10.1007/978-3-031-01911-1

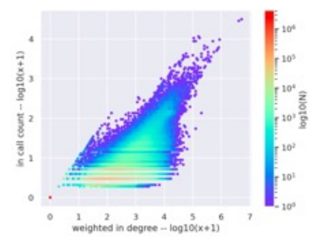


Carnegie Mellon

TgraphSpot: Fast and Effective Anomaly Detection for Time-Evolving Graphs

IEEE BigData, 2022

Mirela Cazzolato[1,2], Saranya Vijayakumar[1], Xinyi Zheng[1], Namyong Park[1], Meng-Chieh Lee[1], Pedro Fidalgo[3,4], Bruno Lages[3], Agma J. M. Traina[2], Christos Faloutsos[1]



Open source:

https://github.com/mtcazzolato/tgraph-spot

Video: https://youtu.be/jI1adN-BQuo?t=1537

AutoGluon TS

- https://auto.gluon.ai/stable/tutorials/timeseries/index.html

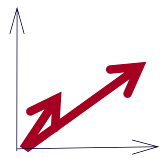
```
from autogluon.timeseries import *
fit()
```

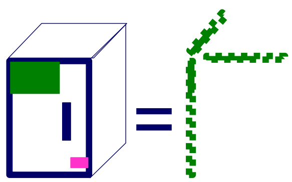
Slides for semester course

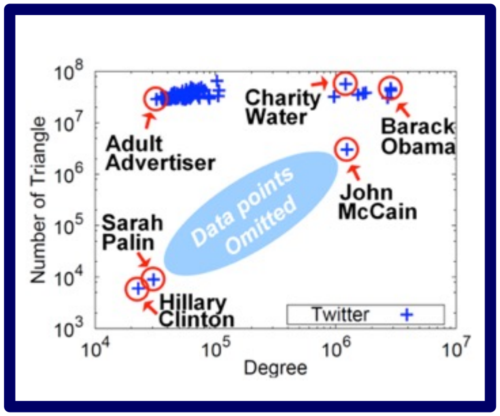
- https://www.cs.cmu.edu/~christos/courses/989.F23/schedule.html
- Fractals and power laws (4 lectures)
- Text mining
- **Matrices, SVD and tensors** (5 lectures)
- **Graph mining** (6 lectures)
- Time series, Fourier, wavelets, & forecasting (4 lectures)

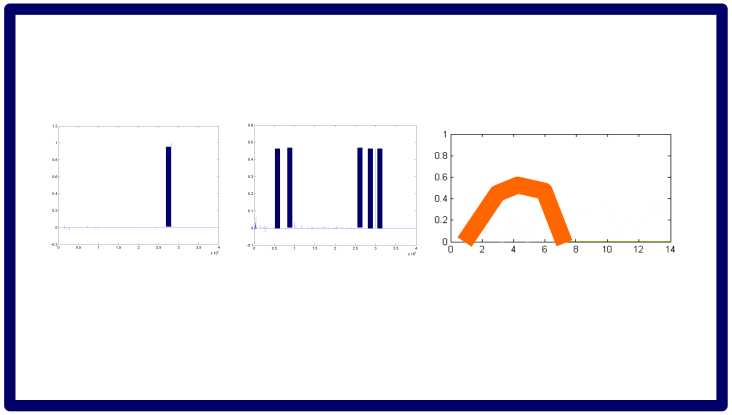
TAKE HOME MESSAGE:

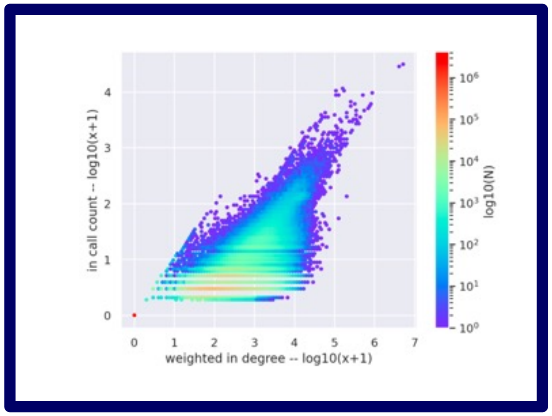
Cross-disciplinarity











Thank you!

christos@cs.cmu.edu