Synthetic Network Data

Modeling Social-Graph Theories, & Applications

Tutors:
Jan Bachmann, and Lisette Espín-Noboa

ECML PKDD 2024 VILNIUS

9am - 1pm

bit.ly/snma-2024

Who are we?

Complexity Science Hub

CEU CENTRAL EUROPEAN UNIVERSITY

Jan Bachmann

- **PhD Student**
  Central European University (CEU) &
  Complexity Science Hub (CSH)

- **Masters in Computer Science**
  RWTH Aachen

- **Research**
  Network inequalities, science of
  science, computational social science

  www.mannbach.de

Lisette Espín-Noboa

- **Postdoc**
  Complexity Science Hub (CSH) &
  Central European University (CEU)

- **PhD. in Computer Science**
  University of Koblenz-Landau

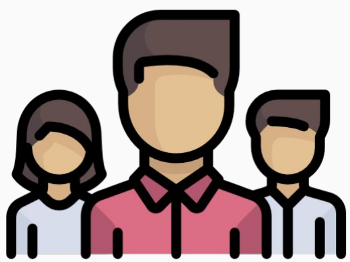
- **Research**
  Network fairness, social network analysis, algorithmic auditing, computational social science

  www.lisetteespin.info

Social networks in the era of big data

Before

Field observations and surveys



“Designed” data covering few people
in small geographical areas

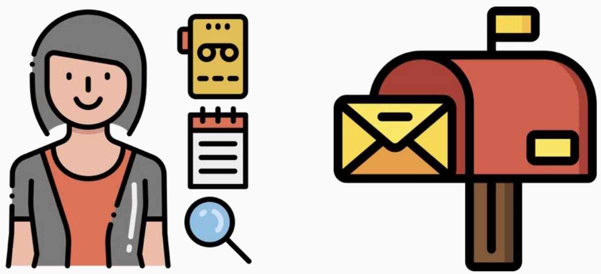
Now

Digital footprints from
social media, phones, online surveys

“Organic” data covering almost
the entire world

Social networks in the era of big data and machine learning

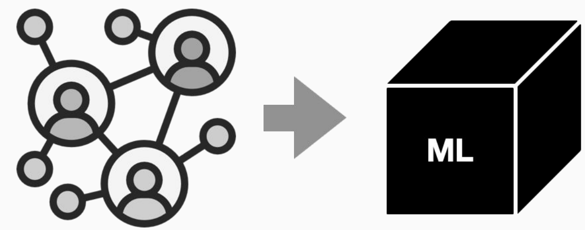
Problem #1



Traditional methods do not scale

We need new tools to characterize edge formation

Problem #2



ML algorithms are not transparent

classification

ranking

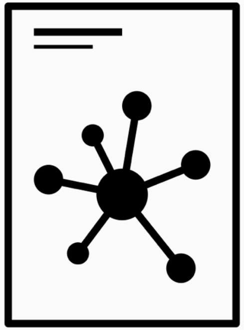
ML algorithms need to be interpretable and explainable

Why do we need synthetic network data?

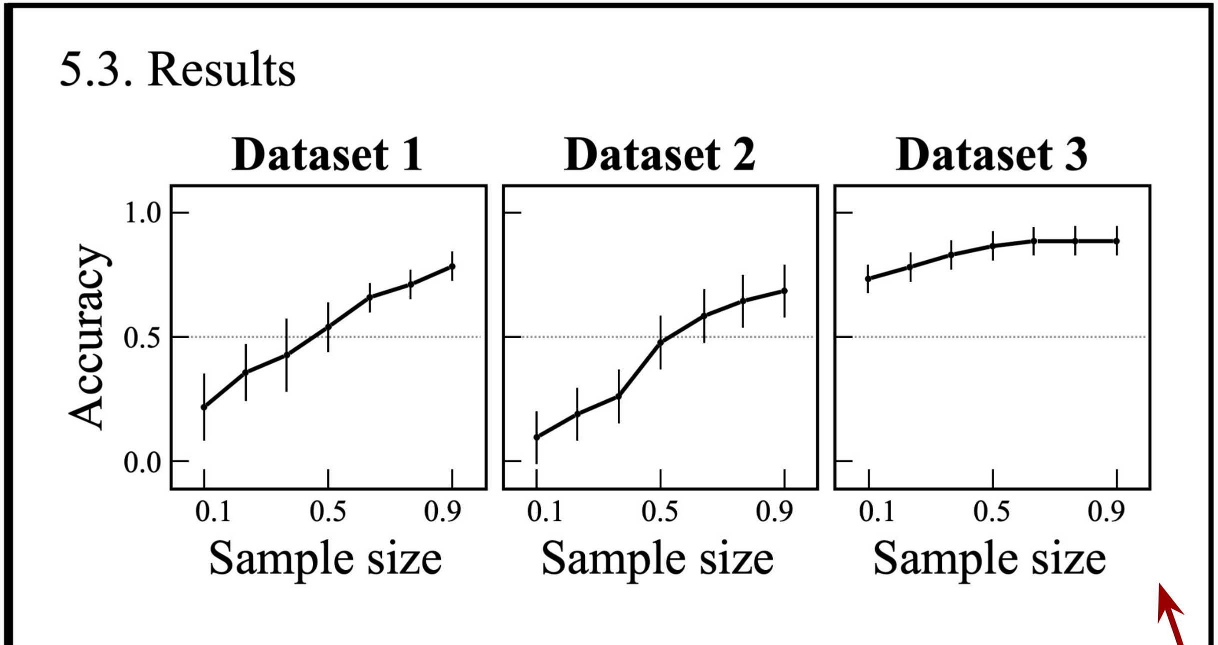
To audit algorithms, and make them interpretable!

image source: https://insights.dice.com/2019/09/25/networking-building-contacts-effective

The CS approach: Evaluating performance on real-world data



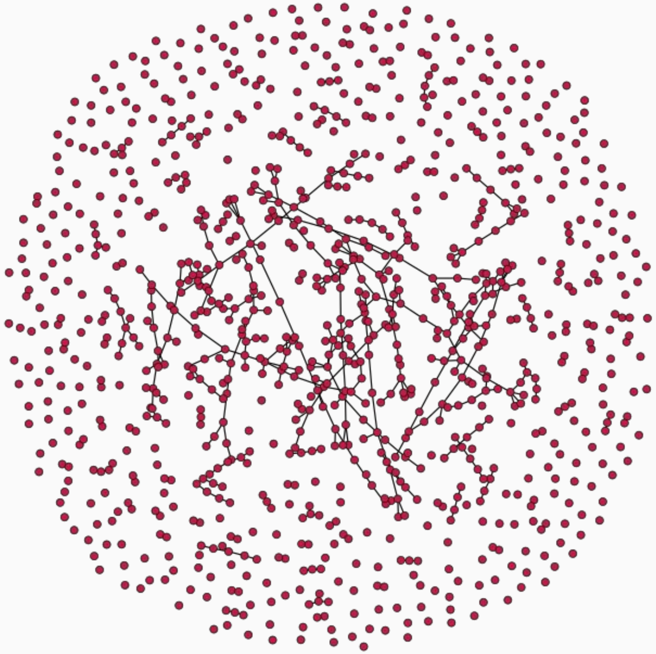
Smith et al.
Novel **node classification algorithm** outperforms state-of-the-art algorithm X.
Top-tier Venue (2024).



Evaluating your algorithms on benchmark datasets is NOT enough if we want to understand the WHY of their outcomes!

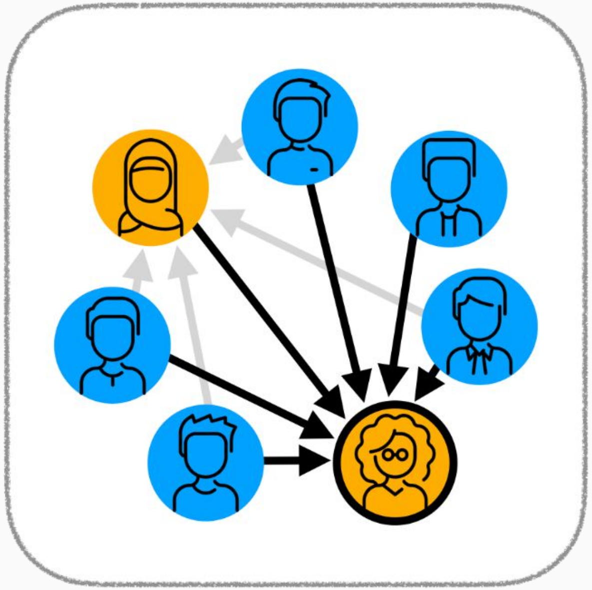
1. The larger the training sample, the better the accuracy
2. Accuracy “seems” to correlate w/ net. structure
3. It “seems” to work best for assortative & directed net.
4. What about other types of networks?

Erdős–Rényi model

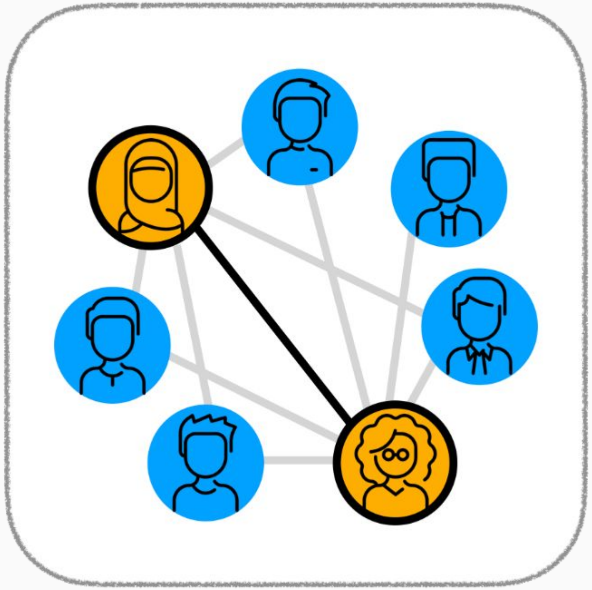


An Erdős–Rényi–Gilbert graph with 1000 vertices at the critical edge probability p=1/(n-1), showing a large component and many small ones

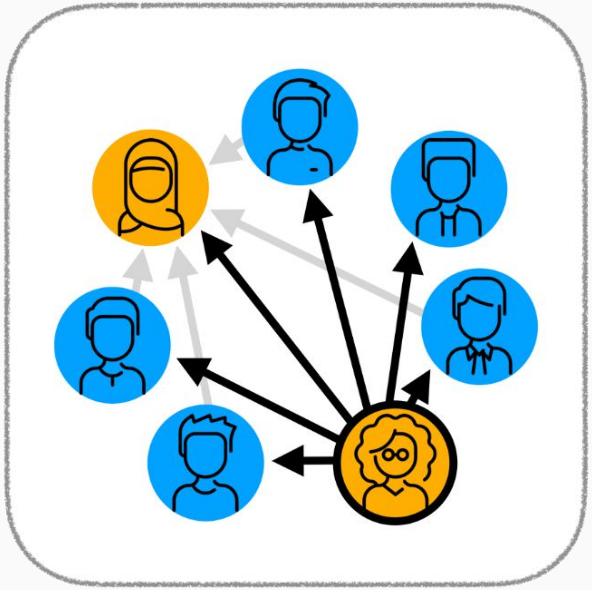
There exists multiple mechanisms of edge formation …



Preferential Attachment
(popularity)



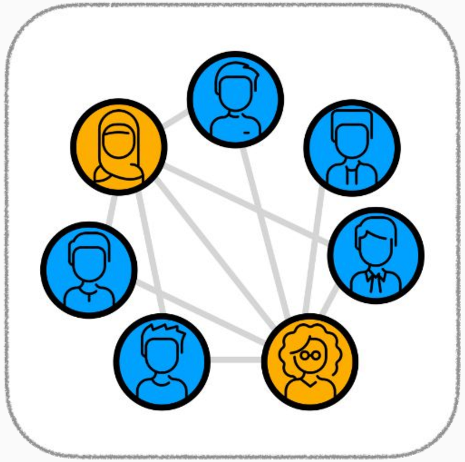
Homophily
(similarity)



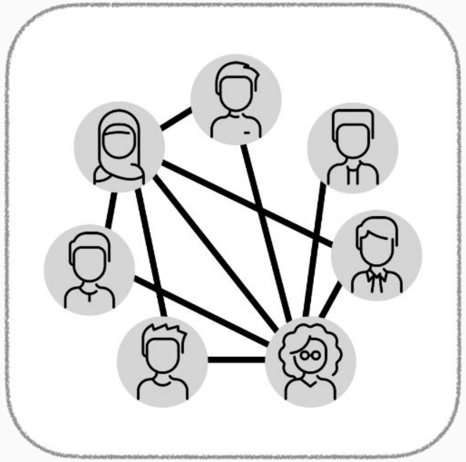
Activity
(outreach)

Merton, R. K. (1968). The Matthew effect in science. Science, 159(3810), 56-63.
Barabási, A. L., & Albert, R. (1999). Emergence of scaling in random networks. science, 286(5439), 509-512.
McPherson, M., et al, (2001). Birds of a feather: Homophily in social networks. Annual review of sociology, 27(1), 415-444.
Newman, M. E. (2003). Mixing patterns in networks. Physical review E, 67(2), 026126.
Perra, N., et al. (2012). Activity driven modeling of time varying networks. Scientific reports, 2(1), 469.

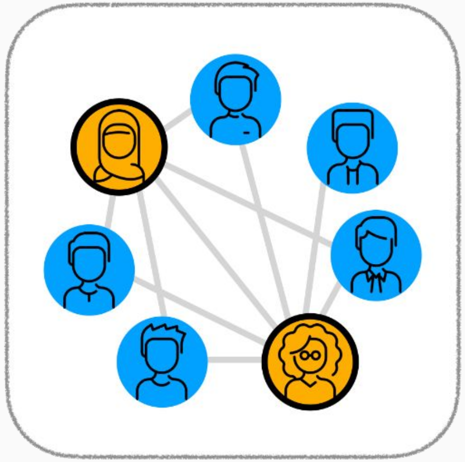
… And graphs can have different structures!



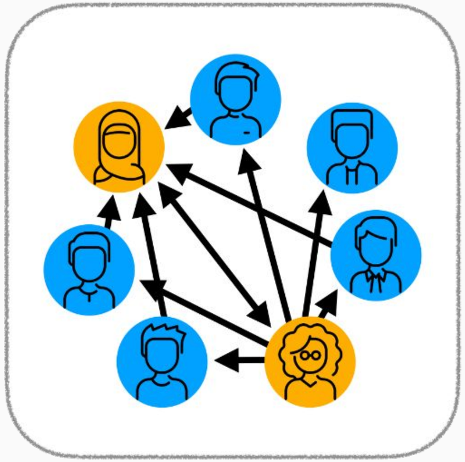
Network size
(number of nodes)



Density
(number of edges given number of nodes)



Fraction of minority
(class balance, aka. group size)



Directionality
(directed edges)

Merton, R. K. (1968). The Matthew effect in science. Science, 159(3810), 56-63.
Barabási, A. L., & Albert, R. (1999). Emergence of scaling in random networks. science, 286(5439), 509-512.
McPherson, M., et al, (2001). Birds of a feather: Homophily in social networks. Annual review of sociology, 27(1), 415-444.
Newman, M. E. (2003). Mixing patterns in networks. Physical review E, 67(2), 026126.
Perra, N., et al. (2012). Activity driven modeling of time varying networks. Scientific reports, 2(1), 469.

What today's tutorial is about

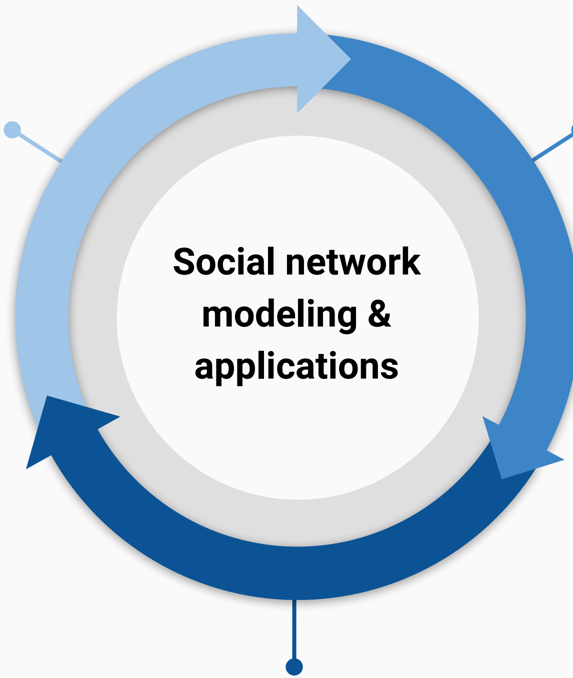
Does the outcome correlate with network structure?

Can network structure determine systemic biases?

What we will cover in this tutorial

PARADIGM 1

From social theories to models
(e.g., homophily, preferential attachment)



PARADIGM 2

From data to models
(e.g., model fitting, inference)

APPLICATIONS

Network inequalities & algorithmic bias
(i.e., ranking, interventions)

What you will learn today

1	How to generate more realistic synthetic graphs	• Useful when real networks are too big, and for data protection • How? Using `netin 2.0.0a1`
2	How to study social phenomena when no real network is available	• How do networks (edges) form? • How to write my own model?
3	To audit network-based algorithms	• What-if scenarios via simulations • To audit network-based algorithms ○ When does my model fail?

What this tutorial is NOT about

- Extensive review of all existing random network models
  - Here we focus on models that help replicating the most important properties and mechanisms found in real-world social networks such as preferential attachment, homophily, and triadic closure.
- Extensive review of network-based algorithms
  - Here we show the main ingredients of how to use synthetic data to audit your own algorithm. Due to time limitations we will cover only sampling biases and ranking inequalities. But the same logic applies to any other algorithm.
- Extensive review of network libraries
  - We will provide a list of common network-based libraries for Python and R, but for today we focus on the **`netin`** package. It bundles the concepts taught today and can be extended to run custom models.

Agenda

Friday, September 13

09:00 - 13:00 (EEST)

ECML PKDD 2024

Radisson Blu Hotel
Omikron

09:15 - 11:00	**Paradigm 1: From social theories to models** Tutor: Lisette Espín-Noboa & Jan Bachmann • Social theories • Network properties and structure • Network models • *Exercise 1*: **`netin`** Graph generation • *Exercise 2*: Auditing node rankings
11:00 - 11:20	**Coffee break**
11:20 - 12:50	**Paradigm 2: From data to models** Tutor: Jan Bachmann & Lisette Espín-Noboa • *Exercise 3*: Mitigating biased node rankings • Model selection (Bayesian) • *Exercise 4*: Model selection
12:50 - 13:00	**Challenges & open questions** Tutor: Lisette Espín-Noboa

Exercises

Please make sure your python environment is ready to go!

Note that if you prefer to run the exercises directly from Google Colab you can skip this info, but recall that you need a Google account.

1. Download and install conda
   conda.io/projects/conda/en/stable/user-guide/install/download.html

2. Create an environment with python 3.9
   `conda create -n "ecmlpkdd" python=3.9`

3. Activate your newly created conda environment
   `conda activate ecmlpkdd`

4. Clone the tutorial in your computer
   ```
   git clone
   ```
   [**https://github.com/snma-tutorial/ecmlpkdd2024.git**](https://github.com/snma-tutorial/ecmlpkdd2024.git)

5. Install the additional dependencies
   ```
   conda install pip
   pip install -r requirements.txt
   ```

Material

All required information is on the tutorial's website:

http://bit.ly/snma-2024