

NET 4103/7431 HOMEWORK
NETWORK SCIENCE AND GRAPH LEARNING
Vincent Gauthier
vincent.gauthier@telecom-sudparis.eu

Credits: Some of these problems have been adapted from the following classes:

- **Prof. Aaron Clauset** Network Analysis and Modeling, 2014
- **Prof. Mason A. Porter** *Networks*, University of Oxford, 2016

SUBMITTING ANSWERS: Prepare answers to your homework in a single PDF/DOCX file and submit it via Moodle.

SUBMITTING CODE: Provide a link to a github or gitlab¹ repository containing all code assignment. Failure to submit your code will result in reduction of all points for that question from your project score.

Background Information

Facebook100 is a dataset of Facebook friendship connections from each of the 100 US universities during a single-time snapshot in fall 2005 (when one had to have a .edu account to be on Facebook).

You can download the dataset from the following link:

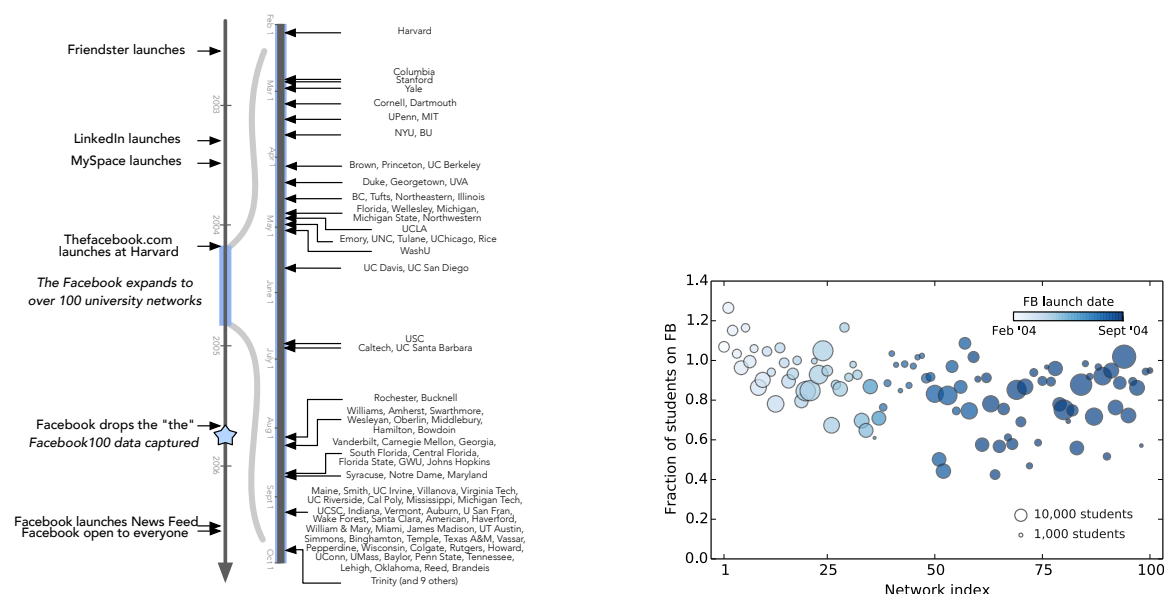
<https://partage.imt.fr/index.php/s/rrwgdsjWj9ZFor>

Facebook history (this text is extracted from the sec. 2 of [1])

Today, Facebook is a large and sophisticated social media system, claiming more than 1.44 billion monthly active users (as of March 2015). In 2005, however, Facebook was a very different kind of online social network, in a correspondingly different social media landscape.

Facebook launched at Harvard University on February 4th, 2004 (Fig. 1a) under the name "thefacebook.com", at a time when the dominant online social networks were Friendster and MySpace. A host of other online college "facebook" such as CUcommunity, CampusNetwork, and CollegeFacebook were also emerging, in addition to efforts by individual universities to move their student directories onto the Web. Facebook initially limited registration to users affiliated with a sanctioned but growing list of colleges, starting with Harvard. Facebooks popularity spread quickly and by the time of the Facebook100 snapshot (September 2005), Facebook had dropped the "the" in its name, opened to over 800 colleges, and had just begun opening itself to high school students. By December 2005, Facebooks user base numbered 6 million, compared to 20 million for

¹<https://gitlabens.imtbs-tsp.eu/>



(a) Key milestone in the early history of Facebook
(Figure extracted from [1])

(b) Fraction of undergraduate that adopted Facebook vs. network index

Figure 1

Friendster and over 22 million users for MySpace. In September 2006, Facebook opened to all persons over the age of 13.

Description of the network dataset

The Facebook100 dataset [2] contains an anonymized snapshot of the friendship connections among $n = 1,208,316$ users affiliated with the first 100 colleges admitted to Facebook, all located in the United States. This comprises a total of $m = 93,969,074$ friendship edges (unweighted and undirected) between users within each separate college. Each vertex is associated with an array of social variables representing the persons status (undergraduate, graduate student, summer student, faculty, staff, or alumni), dorm (if any), major (if any), gender (M or F), and graduation year. Across all networks, only 0.03% of status values are missing. Other variables have slightly higher missing rates (gender: 5.6%; graduation year: 9.8%). Dorm and major have higher rates still, which is likely related to off-campus living and undeclared majors. The completeness of these data reflects the pervading social norms surrounding data privacy expectations in 2005, and possibly a selective bias toward users who disliked the default setting of sharing all information within the college network.

For nearly all colleges, alumni made up about 10-25% of users, a number that increased with the age of the network. Vertices labeled as faculty, staff or students who were not regular undergraduates (graduate students and summer students) made up on average 4.1% of each population.

Each college network includes an index variable that gives its ordinal position of when it joined Facebook: Harvard is 1 and Trinity College is 100 (Fig. 1a). For each network, we acquired college-level variables (enrollment, public vs. private, semester vs. quarter

calendar) from the Integrated Postsecondary Education Data System (IPEDS) provided by the U.S. Department of Education. Full-time undergraduate enrollment from 2007, the earliest date for which data are fully available, was used as a proxy for 2005 enrollment.

By dividing the number of undergraduate accounts in each college network by reported enrollment, we can estimate the fraction of students in each network who were on Facebook, a measure of service adoption. In some cases, the estimated ratio exceeds 1.0 as a result of either errors in our enrollment numbers, part-time students on Facebook who were not counted as full-time enrolled, or multiple/fake accounts at the few colleges that allowed students to control multiple email aliases and circumvent Facebooks initial limits on access.

Question 1: Reading

Read the following documents [3, 2, 1]

Question 2: Social Network Analysis with the Facebook100 Dataset

The smallest network (Caltech) has 762 nodes in the largest connected component (LCC), and the largest has more than 40000 nodes in the LCC.

Lets use three networks from the FB100: Caltech (with 762 nodes in the LCC), MIT (which has 6402 nodes in the LCC), and Johns Hopkins (which has 5157 nodes in the LCC).

- For these three networks plot the degree distribution for each of the three networks that you downloaded. What are you able to conclude from these degree distributions?
- Compute the global clustering coefficient and mean local clustering coefficient for each of the two networks. In addition compute the edge density of each network. Should either of these networks be construed as sparse?
- For each network, also draw a scatter plot of the degree versus local clustering coefficient. Based on these calculations as well as your previous ones, are you able to draw any conclusions about any similarities or differences between the tree networks? What other observations can you make?

Question 3: Social Network Analysis with the Facebook100 Dataset

In this question we expect you will perform the computation on all the available graph in the dataset.

- Of the FB100 networks, investigate the assortativity patterns for three vertex attributes: (i) student/faculty status, (ii) major, (iii) vertex degree, and (iiii) dorm. Treat these networks as simple graphs in your analysis.

For each vertex attribute, make a scatter plot showing the assortativity versus network size n , with log-linear axes for all 100 networks, and a histogram or density plot showing the distribution of assortativity values. In both figures, include a line indicating no assortativity. Briefly discuss the degree to which vertices do or do not exhibit assortative mixing on each attribute, and speculate about what kind of processes or tendencies in the formation of Facebook friendships might produce this kind of pattern. For example, below are figures for assortativity by gender on these networks. The distribution of points spans

the line of no assortativity, with some values nearly as far below 0 as others are above 0. However, the gender attributes do appear to be slightly assortative in these social networks: although all values are within 6% in either direction of 0, the mean assortativity is 0.02, which is slightly above 0. This suggests a slight amount of homophily by gender (like links with like) in the way people friend each other on Facebook, although the tendency is very weak. In some schools, we see a slight tendency for heterophily (like links with dislike), as one might expect if the networks reflected heteronormative dating relationships.

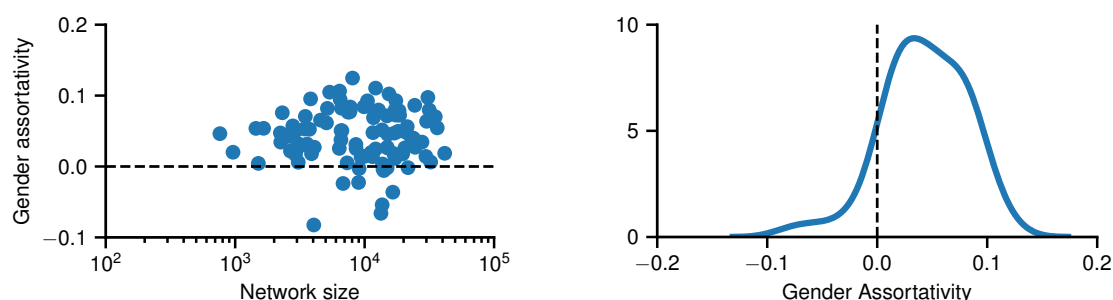


Figure 2: a) Gender assortativity as function of the networks size. b) Gender assortativity distribution.

Question 4: Link prediction

In this question we expect you will compute the link prediction algorithms on a large set of graphs (> 10).

- (a) Read the following documents [4].
- (b) Implement the following link prediction metrics: *common neighbors*, *jaccard*, *Adamic/Adar*. We use the *scikit-learn*² API as an example for our implementation of the link prediction metrics. Please use the implementation (in listing. 1) as an example. Your implementation should inherit from the class *LinkPrediction* defined in listing. 1.
- (c) Evaluating a link predictor:
 1. Select graph $G_{fb}(V, E)$ in the Facebook100 dataset
 2. randomly remove a given fraction $f \in [0.05, 0.1, 0.15, 0.2]$ of edges $E_{removed}$ from the original graph G_{fb} .
 3. For each node pair in the graph $|V| \times |V|$, for each node pair compute the link predictor metrics of interest p , these are the predicted "friendship" $E_{predict}$.
 4. Sort in decreasing order of confidence as a function p from the node pair $E_{predict}$ and then we take the first n pairs of nodes $E_{predict}^{(n)}$.
 5. Compute the size of the intersection between the edge set of removed edges and the edge set of predicted node $|E_{removed} \cap E_{predict}^{(n)}|$.

²scikit-learn: <https://scikit-learn.org/stable/>

- (d) Choose a couple of graphs in the facebook100 dataset run and evaluate each link predictor on them, and conclude on the efficiency of the following metrics: *common neighbors*, *jaccard*, *Adamic/Adar*.

```

1 from abc import ABC
2 from abc import abstractmethod
3 import networkx as nx
4 import numpy as np
5 import progressbar
6
7 class LinkPrediction(ABC):
8     def __init__(self, graph):
9         """
10         Constructor
11
12         Parameters
13         -----
14         graph: Networkx graph
15         """
16         self.graph = graph
17         self.N = len(graph)
18
19     def neighbors(self, v):
20         """
21         Return the neighbors list of a node
22
23         Parameters
24         -----
25         v: int
26             node id
27
28         Return
29         -----
30         neighbors_list: python list
31         """
32         neighbors_list = self.graph.neighbors(v)
33         return list(neighbors_list)
34
35     @abstractmethod
36     def fit(self):
37         raise NotImplementedError("Fit must be implemented")
38
39 class CommonNeighbors(LinkPrediction):
40     def __init__(self, graph):
41         super(CommonNeighbors, self).__init__(graph)

```

Listing 1: Python implementation example of a Link prediction metric

Question 5: Find missing labels with the label propagation algorithms

In this question we expect you will compute the label propagation algorithm on a large set of graphs (> 10).

- Read the following document [5].
- Implement in python the label propagation algorithm [5], please consider pytorch³ and networkx⁴ for the development of your algorithm.

³<https://pytorch.org/>

⁴<https://networkx.org/>

- (c) Choose a network from The Facebook100 dataset and randomly select 10%, 20%, and 30% of of the node attributes of the network to be removed. Use the label propagation algorithm you implemented to recover the missing attributes. Perform this operation for each of the following attributes : "dorm", "major", "gender".

	fraction removed			
	0.1	0.2	0.3	0.4
<i>Duke</i>				
<i>Major</i>	0.282	0.265	0.255	0.241
<i>Dorm</i>	0.529	0.523	0.500	0.463
<i>Year</i>	0.913	0.905	0.898	0.891
<i>Gender</i>	0.675	0.684	0.682	0.679

Table 1: Accuracy of the label propagation algorithm

- (d) For each case of the following percentage of missing attributes: 10%, 20% and 30% and for each of the following attributes: the "dorm", "major", "gender" show the mean absolute error and accuracy score (as defined in eq. 1) of the label propagation algorithm as in the example provided in Table 1 for the Duke University Facebook network. Note we can use the formula eq. 1 for computing the *accuracy*. However, a better approach would have been to compute the F1-score [6].

$$accuracy(y, \hat{y}) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} \mathbf{1}(\hat{y}_i = y_i) \quad (1)$$

- (e) Conclude on the accuracy of the label propagation algorithm for different labels, could you explain why is there such difference in the accuracy between each type of label ?

References

- [1] Jacobs, A. Z., Way, S. F., Ugander, J. & Clauset, A. Assembling thefacebook: Using heterogeneity to understand online social network assembly. In *Proceedings of the ACM Web Science Conference, WebSci '15*, 18:1–18:10 (2015). URL <https://arxiv.org/abs/1503.06772>.
- [2] Traud, A. L., Kelsic, E. D., Mucha, P. J. & Porter, M. A. Comparing community structure to characteristics in online collegiate social networks. *SIAM Review* **53**, 526–543 (2011). URL <https://arxiv.org/abs/0809.0690>.

-
- [3] Traud, A. L., Mucha, P. J. & Porter, M. A. Social structure of facebook networks. *Physica A: Statistical Mechanics and its Applications* **391**, 4165 – 4180 (2012). URL <https://arxiv.org/abs/1102.2166>.
 - [4] Liben-Nowell, D. & Kleinberg, J. The link prediction problem for social networks. In *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, CIKM '03, 556–559 (2003). URL <https://www.cs.cornell.edu/home/kleinber/link-pred.pdf>.
 - [5] Bhagat, S., Cormode, G. & Muthukrishnan, S. Node Classification in Social Networks. In *Social Network Data Analytics*, 115–148 (Springer US, 2011). URL <https://arxiv.org/abs/1101.3291v1>.
 - [6] Mishra, A. Metrics to evaluate your machine learning algorithm (2018). URL <https://bit.ly/2JqbRrt>.