

Multiplex reconstruction with partial information

Daniel Kaiser,^{1,*} Siddharth Patwardhan,^{1,*} and Filippo Radicchi¹

¹*Center for Complex Networks and Systems Research,
Luddy School of Informatics, Computing, and Engineering,
Indiana University, Bloomington, Indiana 47408, USA*

A multiplex is a collection of network layers, each representing a specific type of edges. This appears to be a genuine representation for many real-world systems. However, due to a variety of potential factors, such as limited budget and equipment, or physical impossibility, multiplex data can be difficult to observe directly. Often, only partial information on the layer structure of the system is available, whereas the remaining information is in the form of a single-layer network. In this work, we face the problem of reconstructing the hidden multiplex structure of an aggregated network from partial information. We propose a relatively simple reconstruction algorithm relying on the degree-corrected stochastic block model. The algorithm is characterized by a computational time that grows linearly with the network size. We perform a systematic study of reconstruction problems for both synthetic and real-world multiplex networks. We show that the ability of the proposed method to solve the reconstruction problem is affected by the heterogeneity of the individual layers and the similarity among the layers. On real-world networks, we observe that the accuracy of the reconstruction saturates quickly as the amount of available information increases. In genetic interaction and scientific collaboration multiplexes for example, we find that 10% of ground-truth information yields 70% accuracy, while 30% information allows for more than 90% accuracy.

I. INTRODUCTION

Networks have emerged as powerful modeling frameworks for relational data over the past few decades, boasting an impressive body of supporting research and methodologies [1]. However, it has become clear recently that networks in their simplest realization are incapable of properly modeling multi-dimensional relational data [2–5]. The literature is indeed plenty with examples of how ignoring the multi-dimensionality of systems in network modeling leads to fundamental errors in the characterization of both their structural and dynamical properties, see Refs. [2, 4, 6–18] among others.

A multiplex is probably the simplest network representation of a multi-relational system [2, 3, 19–21]. A multiplex is a collection of single-layer networks sharing common nodes; each layer of a multiplex captures a different type or flavor of pairwise interaction among nodes. This is a convenient and meaningful representation for many real-world systems, including social [22, 23] and biological systems [4, 24, 25].

Even if the system under study is truly a multiplex, data about its topology are often available in an aggregated, single-layer format. As a matter of fact, obtaining precise information about the flavor of all edges in a real multiplex could be prohibitively expensive, too time intensive, or even physically impossible. In this respect, there is an apparent need for tools to infer full, true multiplex structure from an associated single-layer network. In this paper, we refer to this classification problem as the multiplex reconstruction problem (MRP, see Methods for its definition).

Only a few attempts to solve the MRP exist in the literature. In Ref. [26], Bagrow and Lehmann consider the MRP in the context of temporal networks, i.e., multiplex networks where layers correspond to different snapshots of the same network

at different instants of time. They obtain good prediction accuracy leveraging a sparsity-enforced lasso regression technique. The method requires full knowledge not only of the aggregated network topology, but also of the node degrees in the individual layers. Zhang *et al.* consider the MRP on two-layer multiplex networks [27]. Their solution of the MRP corresponds to the maximization of the edge clustering coefficient of the individual layers, obtained via a simulated-annealing-like algorithm. The algorithm is applicable only to small and sufficiently dense multiplex networks, as for example networks representing trades of different commodities among countries. Finally, Wu *et al.* interpret the MRP as a statistical inference problem based on the hypothesis that network layers are instances of the configuration model [28]. They develop an expectation-maximization algorithm for solving the MRP. The algorithm can be trained on partial knowledge of the ground-truth topology of a multiplex to make predictions about the unobserved edges. The method allows to perform prediction and classification of edges of a multiplex network, and the classification task includes the possibility of an edge to belong simultaneously to multiple layers. Wu *et al.* also provide a theoretical analysis of the MRP, relating the accuracy of reconstruction of their method to a metric of multiplex's entropy derived under the ansatz of the configuration model.

The above three articles mark, to the best of our knowledge, the only literature directly addressing the MRP. Some existing literature deals with the related problem of determining whether the multiplex framework is indeed required to properly model an observed single-layer network. For example, Lacasa *et al.* show that diffusion properties of a single-layer network can be used to judge whether the network is truly represented by a single layer or if instead it is better represented by a (hidden) multiplex structure [29]. Similar goals are pursued by Santoro and Nicosia with an approach that leverages Kolmogorov's complexity [16]. Finally, Valles-Catalá *et al.* develop a method to determine the reliability of links in multiplex networks [30]. The method relies on the generalization

* These authors contributed equally to this work.

of the stochastic block model from single-layer to multi-layer networks. Link reliability consists in determining whether an observed (or unobserved) edge exists or not. Valles-Catalá *et al.* show that accounting for the hidden multiplexity with their generalized stochastic block model indeed increases the ability to predict the existence and/or non-existence of edges. However, they do not directly focus on the MRP, i.e., the classification of observed edges in types or flavors.

In this paper, we introduce an algorithm able to approximate MRP solutions. The computational complexity of the algorithm grows linearly with the total number of edges in the multiplex network. The algorithm assumes that the network layers are instances of the degree-corrected stochastic block model [31]. This fact allows us to write a very simple expression for the probability of an observed edge to belong to a specific layer instead of another in terms of the degree of the nodes in the layers and the community structure of the layers. The latter ingredients are learned from the partial observation of the ground-truth structure of the multiplex network. Given that the two main ingredients of the classifier are the degree sequence and the community structure of the layers we named it as degree- and community-based classifier. From the systematic study of the MRP on synthetic multiplex networks, we show that the accuracy of the classification task is strongly influenced by both the heterogeneity of the individual network layers and the correlation among the structure of the layers. From the analysis of real-world networks, we show that the accuracy of the method saturates pretty quickly as the amount of partial information available on the system increases. Roughly, having 30% partial knowledge yields 90% accuracy.

II. RESULTS

In this paper, we consider the multiplex reconstruction problem (MRP) on multiplex networks composed of two layers only. We work under the assumption that all edges in the multiplex network are known. Depending on the experimental setting considered, we may have access to some level of ground-truth information. We remark that in our formulation of the problem edges can belong to one layer only. The above assumptions lead to a formulation of the MRP as a binary classification task. We systematically study the MRP on both synthetic and real networks. Details on the methods, networks and experimental settings are reported in the Methods section.

A. Reconstruction with degree and community information

We begin our analysis under a peculiar experimental setting. Network layers are instances of the configuration model [32]. Except from the pre-imposed degree sequence, network layers are completely random so that no community structure characterizes the networks. We work with power-law degree distributions with tunable degree exponent. We assume that the degree-based (D) classifier has full knowledge of the degree sequences of both layers composing the multi-

plex network. Except for that, the D classifier is completely uninformed about the ground-truth topology of the system, thus the test set is composed of all edges in the network.

We use this experimental setting to understand how difficult the MRP is on synthetic multiplex networks with variable (i) degree heterogeneity at the level of individual layers and (ii) degree-degree correlation among layers. In particular, we note that network layers are generated according to a model that is compatible with the assumption made at the basis of the D classifier, thus results from this set of experiments should represent a reliable proxy for the intrinsic difficulty of the MRP.

As the results of Fig. 1 show, we find that the performance of our method to solve the MRP decreases as the degree exponent of the pre-imposed degree distribution increases. Essentially, solving the MRP is easier on multiplex networks with heterogeneous degree sequences than on multiplex networks with homogeneous degree sequences. However, it is also important that the two layers are sufficiently diverse one from the other. If nodes have the same exact degrees on both layers, the performance of the D classifier is identical to that of random guessing (Fig. 1a). As the degree-degree correlation of the layers decreases, performance improves. In particular, we find that the best performance is achieved for maximally anticorrelated degree sequences (Fig. 1b).

Second, we study the performance of the community-based (C) classifier on network layers generated according to the Lancichinetti-Fortunato-Radicchi (LFR) model [33]. We use the LFR model to generate network layers with built-in community structure; we further control for the amount of degree and community-structure correlation between the layers of the multiplex. We test the performance of the classifier on these networks when informed about their ground-truth community structure, but no information on their degree sequence is used. As the results of Fig. 1c indicate, the performance of the C classifier increases as the strength of the community structure increases; also, performance increases as the correlation between the community structure of the layers decreases.

Finally, we apply the degree- and community-based (DC) classifier to the LFR multiplex networks. We find that the combination of the two ingredients leads to a significant boost in classification performance (Fig. 1d). Decreasing correlation leads to a visible improvement in classification performance only if the community structure of the layers is strong enough (Fig. 1d).

B. Reconstruction with partial edge information

The experimental settings considered in the previous section are useful to understand intrinsic properties of the MRP. However, the settings are not very representative for applications in the real world. For example, it appears as unrealistic to have full and exact knowledge of the layer-wise community structure of the multiplex, but no information about the topology of the multiplex. From now on, we work using a standard experimental setting where we assume that the edges in the multiplex are randomly divided in a training set and a test set. The relative size of one set over the other is the main control

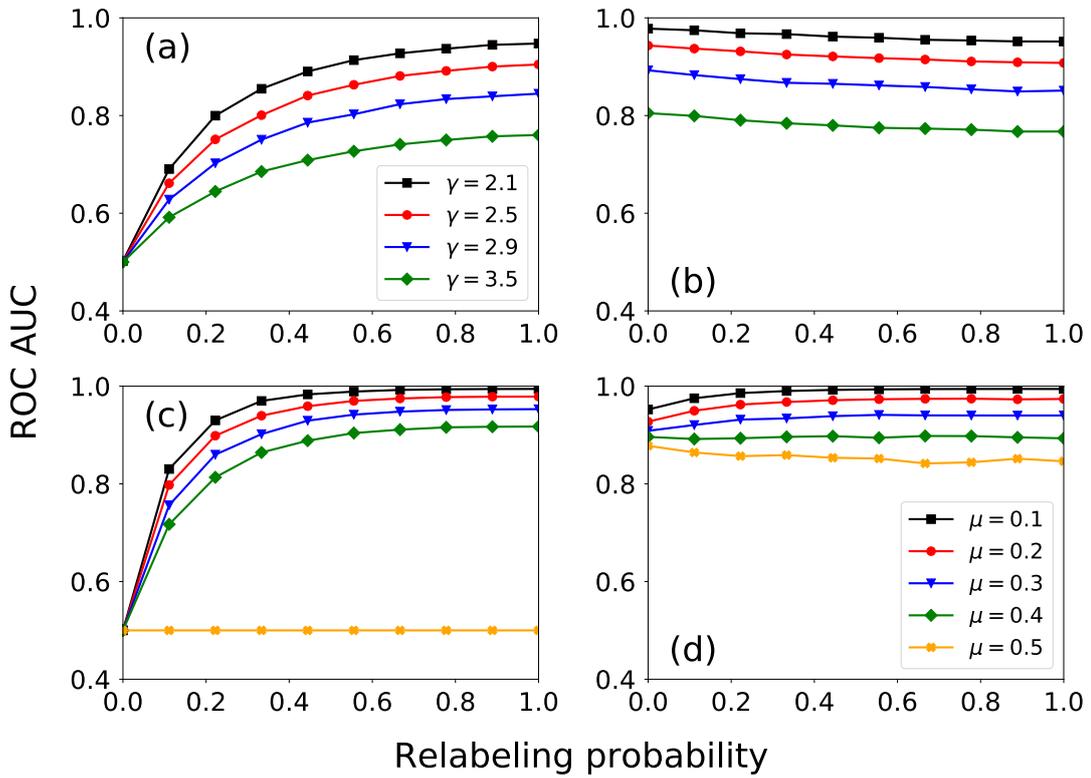


Figure 1. **Reconstruction of synthetic multiplex networks.** (a) We consider random multiplex networks with power-law degree distribution $P(k) \sim k^{-\gamma}$ composed of $N = 100,000$ nodes. Minimum degree is $k_{\min} = 3$; maximum degree is $k_{\max} = \sqrt{N}$ if $\gamma \leq 3$, and $k_{\max} = N^{1/(\gamma-1)}$ if $\gamma \geq 3$. Different curves are obtained for different γ values. We plot the classification metric ROC AUC of the degree-based (D) classifier as a function of the probability relabeling nodes; we relabel nodes to control for degree-degree correlation (see Methods). Degrees are positively correlated when the relabeling probability is zero; they become progressively uncorrelated as the relabeling probability increases. Results are obtained over a single realization of the model. We do not observe substantial fluctuations from run to run. (b) Same as in panel (a), but for multiplex networks with anti-correlated degree sequences. The sequences are maximally anti-correlated for a null relabeling probability, and they become uncorrelated as the relabeling probability increases. (c) We consider the multiplex networks with pre-imposed community structure and power-law degree distribution $P(k) \sim k^{-\gamma}$. Graphs with $N = 10,000$ nodes are generated according to the Lancichinetti-Fortunato-Radicchi (LFR) model, see Methods for details. We set degree exponent $\gamma = 2.1$, maximum degree $k_{\max} = \sqrt{N}$, and average degree $\langle k \rangle = 20$. Communities have size obeying a power-law distribution with exponent $\tau = 1.0$. The strength of the community structure is determined by the mixing parameter μ . We plot the ROC AUC of the community-based (C) classifier as a function of the probability relabeling the nodes of the multiplex. The classifier is unaware of the ground-truth degree sequences of the network layers. Different curves are obtained for different values of the mixing parameter μ controlling for the strength of the community structure of the network layers. The community structure of the two layers is correlated when no nodes are relabeled; correlation decreases as the relabeling probability increases. (d) We plot the ROC AUC of the degree- and community-based (DC) classifier as a function of the relabeling probability for synthetic multiplex networks with pre-imposed community structure. The networks are the same networks as in panel (c). In this case, however, the classifier takes also advantage of the knowledge of the ground-truth degree sequence of the network layers.

parameter of our experiments. The classifier is trained from the knowledge of the ground-truth flavor of the edges within the training set. Performance is measured as the ROC AUC of the binary classification task concerning edges belonging to the test set.

We consider the MRP on synthetic multiplex networks, see Fig. 2. In this set of experiment, the degree sequences and community structures of the layers are uncorrelated. Similarly to what reported in the previous section, we find that the accuracy of the DC classifier grows as the heterogeneity of the degree distribution (Fig. 2a) and the strength of the community structure (Fig. 2b and Fig. 2c) increase. Depending on whether the network layers have or have not modular struc-

ture, the classification is enhanced if the classifier takes or does not take advantage of its community-based component (Fig. 2d). If the graph has not community structure, the DC classifier may also display lower performance than the simple D classifier. In all the experiments considered in Fig. 2, the accuracy of the classifier saturates quite quickly with the amount of partial information used to train the classifier. The performance of the classifier saturates as the size of the training set increases. These general observations on the performance of the DC classifier to solve the MRP on synthetic networks are only mildly affected by the average degree of the network and its size. Specifically, we find that the performance of the classifier is almost unaffected by the average degree of multiplex

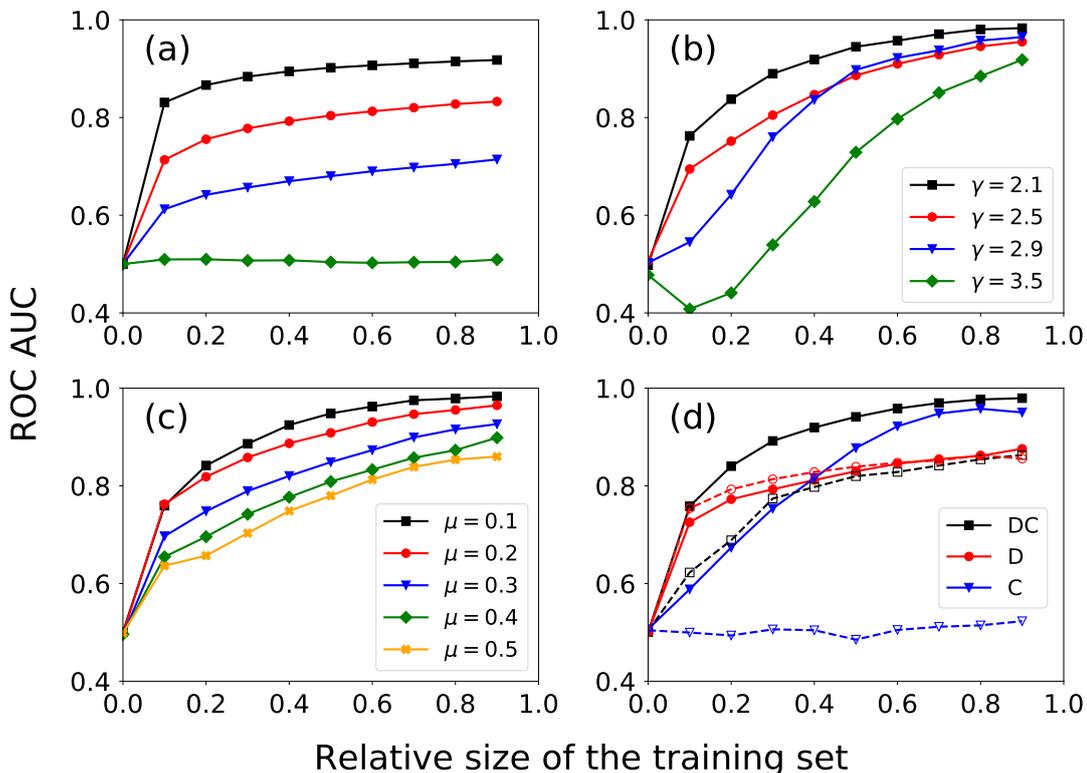


Figure 2. **Reconstruction of synthetic multiplex networks with partial edge information.** (a) We plot the performance metric ROC AUC of the degree-based (D) classifier as a function of the relative size of the training set. Multiplex networks have size $N = 100,000$ and power-law degree distribution $P(k) \sim k^{-\gamma}$. We display results for different γ values. We impose minimum degree $k_{\min} = 3$ for all γ values, while the maximum degree is $k_{\max} = \sqrt{N}$ for $\gamma \leq 3$ and $k_{\max} = N^{1/(\gamma-1)}$ for $\gamma > 3$. The degree sequences of the layers are uncorrelated. (b) We plot the ROC AUC of the degree- and community-based (DC) classifier as a function of the relative size of the training set. Tests are performed on synthetic multiplex networks with pre-imposed community structure constructed according to the LFR model with $N = 10,000$, $\langle k \rangle = 5.0$, and $k_{\max} = \sqrt{N}$ for $\gamma \leq 3$ and $k_{\max} = N^{1/(\gamma-1)}$ for $\gamma > 3$. Community power-law exponent is $\tau = 1.0$, and mixing parameter is $\mu = 0.1$. The degree sequences and community structures of the layers are uncorrelated. (c) Same as in (b), but for fixed degree exponent $\gamma = 2.1$ and different μ values. (d) We compare the performance of the different classifiers on synthetic graph with built-in community structure (filled symbols / full curves) and without community structure (empty symbols / dashed curves). We plot the performance metric of the DC classifier, the D classifier and the community-based (C) classifier. Graphs with community structure are constructed using the LFR multiplex model with $N = 10,000$, $\gamma = 2.1$, $k_{\max} = \sqrt{N}$, $\langle k \rangle = 5.0$, $\tau = 1.0$, and $\mu = 0.1$. We use the configuration model with $N = 10,000$, $\gamma = 2.1$, $k_{\min} = 3$, $k_{\max} = \sqrt{N}$ for the graphs with no community structure. The degree sequences and community structures of the layers are uncorrelated.

network, see Fig. 5a. Also, we find that, as the system size is increased, the performance of the reconstruction algorithm mildly increases, see Figs. 5b and 5c.

C. Time complexity of the reconstruction algorithms

The proposed DC classifier generates solutions of the MRP in a time that grows almost linearly with the total number of edges in the multiplex. The classifier benefits from the scalability of the Louvain algorithm, i.e., the method used to detect communities in the network layers [34]. We stress that the DC classifier can leverage any community detection method to produce solutions of the MRP; however, the time complexity of the classifier may be dramatically increased by that of the community detection method. The fact that the DC method is able to reconstruct multiplex networks in linear time is verified in Fig. 3. There, we systematically apply the DC method

in the reconstruction of the topology of multiplex networks composed of synthetic network layers of variable size. By construction, these networks are sparse so that the total number of edges is proportional to the total number of nodes. In the same spirit as in the analysis of Fig. 2, we use the DC classifier to generate solutions of the MRP for a given amount of partial information. Please note that, even networks that do not have pre-imposed community structure are still analyzed according to the full pipeline of the DC classifier. The quasi-linear scaling of the time of computation is apparent from our results.

D. Applications to real-world multiplex networks

Finally, we study the MRP with partial edge information on real-world multiplex networks, see Tab. I for the complete list of datasets analyzed. In Fig. 4, we report results for four

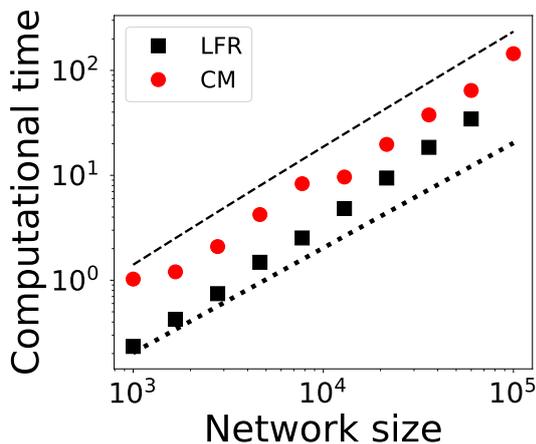


Figure 3. **Computational complexity of algorithms for multiplex reconstruction.** We generate multiplex networks with variable size N . Tests are performed on synthetic multiplex networks with pre-imposed community structure constructed according to the LFR model with $\gamma = 2.1$, $\tau = 1.0$, $k_{\max} = \sqrt{N}$, $\langle k \rangle = 5.0$, and $\mu = 0.1$. Tests are also performed on networks without community structure that were generated using the configuration model (CM) with $\gamma = 2.1$, $k_{\min} = 3$ and $k_{\max} = \sqrt{N}$. No correlation at the level of degree sequence and/or community structure is present between the layers of the multiplex. Given a network, we provide 50% of partial information to the degree- and community-based (DC) classifier and measure the time required by the algorithm to reconstruct the multiplex. Computational time is measured in seconds. Simulations were run on a Intel(R) Xeon(R) CPU E5-2690 v4 @ 2.60GHz. As guidelines, we display lines denoting the scalings N (dotted) and $N \log N$ (dashed).

datasets. Results for three additional datasets can be found in Fig. 6. Although some datasets include information about multiplex networks with more than two layers, we study only multiplexes formed by two layers at a time. Overall, we find a mixed behavior of the DC classifier. There are cases where the classifier is able to correctly identify the flavor of a large portion of edges even if the size of the training set is small. This fact happens for multiplex networks representing scientific collaborations (Fig. 4a) and genetic interactions in organisms (Figs. 4b and 6). In other situations instead, as for example for a connectome multiplex network (Figs. 4c) and for a multiplex transportation network (Figs. 4d), the performance of the classifier is not exceptional.

We compare the performance of the DC classifier with a suitably modified version of the classifier by Wu *et al.* [28], see Methods for details. The classifier by Wu *et al.* relies on the configuration model, and the classification of edges is obtained via the expectation-maximization algorithm. No information on the network community structure is used by the method. The classifier by Wu *et al.* excels on the London multiplex transportation network (Fig. 4d). It produces comparable performance to the one of the DC classifier in all other multiplex networks, except for those characterized by layers with sufficiently strong but uncorrelated community structure, as for example the genetic interaction multiplex of the *Drosophila Melanogaster* (Fig. 4a). The strength of the

community structure of a network layer is measured in terms of modularity, see Tab. I for modularity values of the best partitions for the various multiplex networks. In the same table, we also report the normalized mutual information between the best partitions of the two layers, which provides us with a quantitative proxy to judge the level of similarity between pairwise community structures.

III. DISCUSSION

In this paper, we studied the multiplex reconstruction problem (MRP), i.e., the identification of the flavour of edges in multiplex networks composed of two layers.

First, we characterized facets of the reconstruction problem. We showed that the intrinsic difficulty of the MRP depends on the level of heterogeneity of the individual layers, in the sense that multiplex networks with broad degree distributions are easier to reconstruct than multiplex networks with homogeneous degree distributions. If layers are characterized by community structure, then the stronger such a structure is the easier the MRP is. However, within-layer structural diversity is not the only important ingredient that determines the hardness of the MRP. The type and strength of structural correlation among layers is essential too. We showed that multiplex networks that have non-correlated or anticorrelated layers, at the level of degree sequences and/or community structure, can be reconstructed quite well. On the other hand, positively correlated layers do not allow for an easy reconstruction.

Second, we presented a new algorithm for solving the MRP. The method classifies edges based on the assumption that the network layers of the multiplex are independent instances of the degree-corrected stochastic block model. In our experiments, the classifier is trained on a portion of edges whose ground-truth layer is revealed. Its performance is measured in terms of the ability of classification of the remaining edges whose ground-truth layer is not revealed. We systematically studied the performance of the classifier on both synthetic and real multiplex networks as a function of the relative size of the training set of edges. We found that the performance of the classifier saturates quite quickly as the amount of information used to train the classifier increases. Roughly, 30% of known edges are sufficient to let the classifier reach 90% performance. There are, however, also multiplex networks that are not easily reconstructed by our method.

The present work can be extended in multiple directions. The generalization of the MRP to multiplex networks composed of more than two layers is one of these directions. Considering mechanisms of aggregation different from the exclusive OR that we addressed in this paper is another potentially relevant direction. Also, the two main ingredients of the reconstruction algorithm developed in this paper can be used in more sophisticated, maybe more effective ways. For example, one could think of treating the MRP as a maximum likelihood problem where the network layers of the multiplex are fitted by degree-corrected stochastic block models, and edges's flavors are treated as the tunable parame-

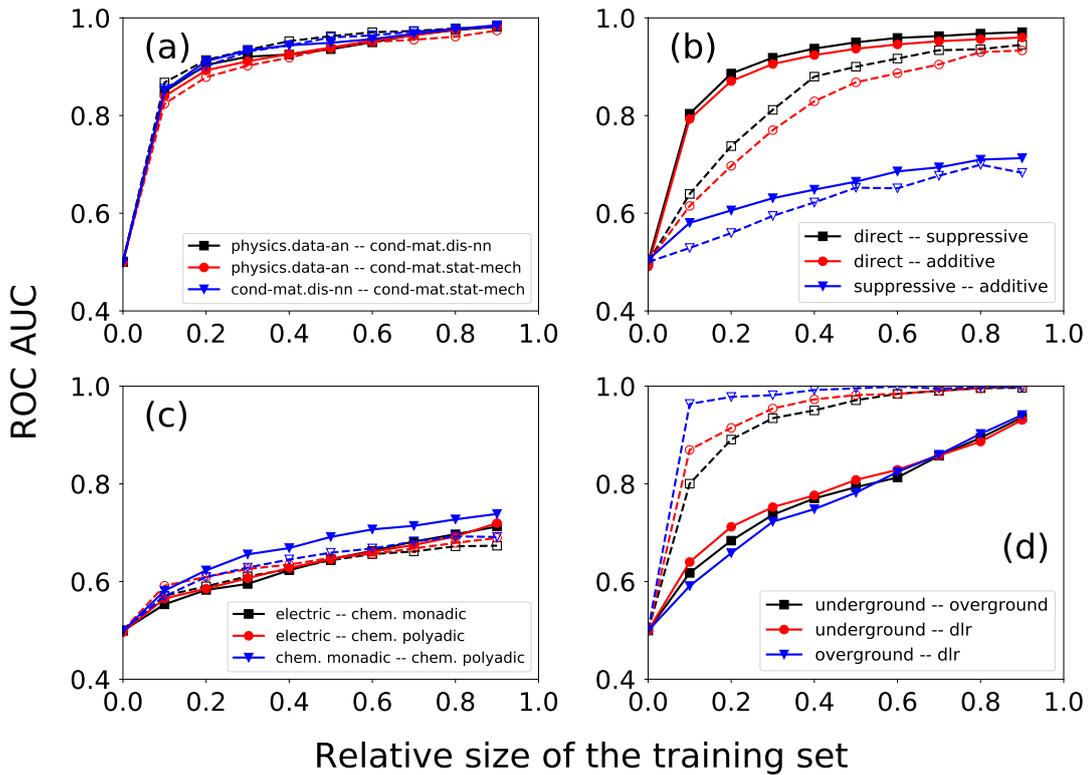


Figure 4. **Reconstruction of real-world multiplex networks.** (a) ROC AUC as a function of the relative size of the training set. We report results for the arXiv multiplex collaboration network [35]. We compare the results achieved with the degree- and community-based (DC, filled symbols / full curves) classifier with those of the Wu *et al.*'s classifier (empty symbols / dashed curves) [28]. Results are averaged over 10 realizations. (b) Same as in panel (a), but for the genetic interactions multiplex network of the *Drosophila Melanogaster* [36]. Results are averaged over 100 realizations. (c) Same as in panel (a), but for the *Caenorhabditis Elegans* multiplex connectome [37]. (d) Same as in panel (a), but for the London multiplex transportation network [38]. Results are averaged over 100 realizations.

ters of the fit. Such an extension could also involve the use of more sophisticated models such as the multi-layer degree-corrected stochastic block model [30]. Potential solutions of the MRP could be then obtained by means of likelihood maximization using standard optimization techniques such as the expectation-maximization algorithm or simulated annealing.

ACKNOWLEDGMENTS

The authors thank G. Bianconi for comments on the manuscript. S.P. and F.R. acknowledge support by the Army Research Office (W911NF-21-1-0194); F.R. acknowledges support by the the Air Force Office of Scientific Research (FA9550-21-1-0446). The funders had no role in study design, data collection and analysis, decision to publish, or any opinions, findings, and conclusions or recommendations expressed in the manuscript.

IV. METHODS

A. The multiplex reconstruction problem

We consider multiplex networks composed of two layers, namely α and β . The graph representing layer α is denoted by $\mathcal{G}^{(\alpha)} = \{\mathcal{N}^{(\alpha)}, \mathcal{E}^{(\alpha)}\}$, where $\mathcal{N}^{(\alpha)}$ is the set of nodes of the layer, and $\mathcal{E}^{(\alpha)}$ is the set of its edges. The same notation is used for layer β . Being a multiplex, we assume that $\mathcal{N}^{(\alpha)} = \mathcal{N}^{(\beta)}$, and we indicate the size of the network with $N = |\mathcal{N}^{(\alpha)}| = |\mathcal{N}^{(\beta)}|$. Further, we exclude the possibility that the edge (i, j) belongs simultaneously to both layers, meaning that $\mathcal{E}^{(\alpha)} \cap \mathcal{E}^{(\beta)} = \emptyset$.

We indicate the degree sequence of the layer α with $\vec{k}^{(\alpha)} = (k_1^{(\alpha)}, k_2^{(\alpha)}, \dots, k_N^{(\alpha)})$, where $k_i^{(\alpha)}$ is the degree of node i in layer α defined as

$$k_i^{(\alpha)} = \sum_{(r,s) \in \mathcal{E}^{(\alpha)}} \delta(i, r) + \delta(i, s). \quad (1)$$

In the above expression, $\delta(x, y)$ is the Kronecker delta function, i.e., $\delta(x, y) = 1$ if $x = y$ and $\delta(x, y) = 0$ if $x \neq y$. A similar expression is used to relate the degree sequence $\vec{k}^{(\beta)}$ to the set $\mathcal{E}^{(\beta)}$.

The community memberships of the nodes in the two layers are denoted respectively as $\vec{\sigma}^{(\alpha)}$ and $\vec{\sigma}^{(\beta)}$. Community structure is either *a priori* known based on the generative network model of the multiplex layers, or inferred by the Louvain algorithm [34].

We define the multiplex reconstruction problem (MRP) as the binary classification of the individual edges in the multiplex layers, i.e., predicting whether the generic edge (i, j) belongs to either $\mathcal{E}^{(\alpha)}$ or $\mathcal{E}^{(\beta)}$.

We perform the classification using partial knowledge of the ground-truth multiplex network. We consider two experimental settings:

1. We assume to have full knowledge about some of the features of the individual nodes, i.e., degrees and/or community memberships, but no direct information about the sets $\mathcal{E}^{(\alpha)}$ and $\mathcal{E}^{(\beta)}$. We train a classifier using the available information and apply it to the classification of all edges $(i, j) \in \mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}$.
2. We assume to have information about a portion of the network edges, namely the training sets $\mathcal{E}_{\text{train}}^{(\alpha)} \subseteq \mathcal{E}^{(\alpha)}$ and $\mathcal{E}_{\text{train}}^{(\beta)} \subseteq \mathcal{E}^{(\beta)}$, respectively. We train a classifier on these sets and we use it to classify all edges $(i, j) \in \mathcal{E}_{\text{test}}^{(\alpha)} \cup \mathcal{E}_{\text{test}}^{(\beta)}$, with $\mathcal{E}_{\text{test}}^{(\alpha)} = \mathcal{E}^{(\alpha)} \setminus \mathcal{E}_{\text{train}}^{(\alpha)}$ and $\mathcal{E}_{\text{test}}^{(\beta)} = \mathcal{E}^{(\beta)} \setminus \mathcal{E}_{\text{train}}^{(\beta)}$.

In both the above experimental settings, we measure the performance of the classifier using the area under the receiver operating characteristic curve (ROC AUC), i.e., a standard metric in binary classification tasks.

B. Degree-based classifier

In the first experimental setting, we assume to have complete knowledge of the degree sequences $\vec{k}^{(\alpha)}$ and $\vec{k}^{(\beta)}$. We use a straightforward prediction model, where we pretend that the networks of layers α and β are random networks generated according to the configuration model with prescribed degree sequences $\vec{k}^{(\alpha)}$ and $\vec{k}^{(\beta)}$, respectively [32]. Note that the network topology of each layer is assumed to be generated independently. According to our classifier, the edge $(i, j) \in \mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}$ belongs to layer α with probability

$$P\left[(i, j) \in \mathcal{E}^{(\alpha)} \mid \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}\right] = \frac{k_i^{(\alpha)} k_j^{(\alpha)}}{k_i^{(\alpha)} k_j^{(\alpha)} + k_i^{(\beta)} k_j^{(\beta)}}. \quad (2)$$

Clearly, we have that

$$P\left[(i, j) \in \mathcal{E}^{(\beta)} \mid \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}\right] = 1 - P\left[(i, j) \in \mathcal{E}^{(\alpha)} \mid \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}\right]. \quad (3)$$

In the second experimental setting, we first use Eq. (1) to estimate the degree sequences $\vec{k}_{\text{train}}^{(\alpha)}$ and $\vec{k}_{\text{train}}^{(\beta)}$ from the sets $(\mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}) \setminus \mathcal{E}_{\text{train}}^{(\beta)}$ and $(\mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}) \setminus \mathcal{E}_{\text{train}}^{(\alpha)}$, respectively. Essentially, we pretend that a layer is formed by all edges in the aggregate less those edges that we know for sure they belong to

the other layer. Then, we apply the classifier of Eq. (2), where $\vec{k}_{\text{train}}^{(\alpha)}$ and $\vec{k}_{\text{train}}^{(\beta)}$ respectively replace the unknown ground-truth vectors $\vec{k}^{(\alpha)}$ and $\vec{k}^{(\beta)}$, to the edges in the test set $\mathcal{E}_{\text{test}}^{(\alpha)} \cup \mathcal{E}_{\text{test}}^{(\beta)}$.

C. Community-based classifier

In the first experimental setting, we assume to have complete knowledge of the community memberships $\vec{\sigma}^{(\alpha)}$ and $\vec{\sigma}^{(\beta)}$. We use a straightforward prediction model, where we pretend that the networks of layers α and β are random networks generated according to the homogeneous stochastic block model with prescribed community memberships $\vec{\sigma}^{(\alpha)}$ and $\vec{\sigma}^{(\beta)}$, respectively. Note that the network topology of each layer is assumed to be generated independently. According to our classifier, the edge $(i, j) \in \mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}$ belongs to layer α with probability

$$P\left[(i, j) \in \mathcal{E}^{(\alpha)} \mid \vec{\sigma}^{(\alpha)}, \vec{\sigma}^{(\beta)}\right] = \frac{C(\sigma_i^{(\alpha)}, \sigma_j^{(\alpha)})}{C(\sigma_i^{(\alpha)}, \sigma_j^{(\alpha)}) + C(\sigma_i^{(\beta)}, \sigma_j^{(\beta)})}. \quad (4)$$

Clearly, we have that

$$P\left[(i, j) \in \mathcal{E}^{(\beta)} \mid \vec{\sigma}^{(\alpha)}, \vec{\sigma}^{(\beta)}\right] = 1 - P\left[(i, j) \in \mathcal{E}^{(\alpha)} \mid \vec{\sigma}^{(\alpha)}, \vec{\sigma}^{(\beta)}\right]. \quad (5)$$

In Eq. (4), $C(\sigma_i, \sigma_j)$ represents the propensity that two nodes with community memberships σ_i and σ_j are connected. For simplicity, we assume that such a propensity can be written as

$$C(\sigma_i, \sigma_j) = \nu [1 - \delta(\sigma_i, \sigma_j)] + (1 - \nu) \delta(\sigma_i, \sigma_j), \quad (6)$$

with $0 \leq \nu \leq 1$. Please note that the value of the parameter ν is known to the classifier.

The assumption of Eq. (6) is similar to the one underlying the Lancichinetti-Fortunato-Radicchi (LFR) model [33], with the caveat $\nu \approx \mu$, where μ is the mixing parameter of the LFR model.

In the second experimental setting, we apply the Louvain community detection algorithm to the network formed by the edges in the set $(\mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}) \setminus \mathcal{E}_{\text{train}}^{(\beta)}$ to infer the community structure $\vec{\sigma}_{\text{train}}^{(\alpha)}$ of layer α . A similar procedure is used to infer the vector $\vec{\sigma}_{\text{train}}^{(\beta)}$. Finally, we use the classifier of Eq. (4), where $\vec{\sigma}_{\text{train}}^{(\alpha)}$ and $\vec{\sigma}_{\text{train}}^{(\beta)}$ are used in place of $\vec{\sigma}^{(\alpha)}$ and $\vec{\sigma}^{(\beta)}$, respectively. The parameter ν of Eq. (6) is estimated as

$$\nu_{\text{train}} = 1 - \frac{R_{\text{train}}^{(\alpha)} + R_{\text{train}}^{(\beta)}}{\left|(\mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}) \setminus \mathcal{E}_{\text{train}}^{(\beta)}\right| + \left|(\mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}) \setminus \mathcal{E}_{\text{train}}^{(\alpha)}\right|}, \quad (7)$$

where

$$R_{\text{train}}^{(\alpha)} = \sum_{(i, j) \in (\mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}) \setminus \mathcal{E}_{\text{train}}^{(\beta)}} \delta\left[(\vec{\sigma}_{\text{train}}^{(\alpha)})_i, (\vec{\sigma}_{\text{train}}^{(\alpha)})_j\right].$$

A similar definition is used for $R_{\text{train}}^{(\beta)}$.

D. Degree- and community-based classifier

In the first experimental setting, we assume to have complete knowledge of the degree sequences $\vec{k}^{(\alpha)}$ and $\vec{k}^{(\beta)}$ of the network layers, and of the community memberships $\vec{\sigma}^{(\alpha)}$ and $\vec{\sigma}^{(\beta)}$. We pretend that the network layers are instances of the degree-corrected stochastic block model [31]. Also in this case, we assume that the network topology of each layer is generated independently of the other. According to this degree- and community-based classifier, the edge $(i, j) \in \mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}$ belongs to layer α with probability

$$P\left[(i, j) \in \mathcal{E}^{(\alpha)} | \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}, \vec{\sigma}^{(\alpha)}, \vec{\sigma}^{(\beta)}\right] = \frac{Q k_i^{(\alpha)} k_j^{(\alpha)} C(\sigma_i^{(\alpha)}, \sigma_j^{(\alpha)})}{Q k_i^{(\alpha)} k_j^{(\alpha)} C(\sigma_i^{(\alpha)}, \sigma_j^{(\alpha)}) + Q k_i^{(\beta)} k_j^{(\beta)} C(\sigma_i^{(\beta)}, \sigma_j^{(\beta)})}. \quad (8)$$

In the above equation, Q is a normalization constant so that

$$P\left[(i, j) \in \mathcal{E}^{(\beta)} | \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}, \vec{\sigma}^{(\alpha)}, \vec{\sigma}^{(\beta)}\right] = 1 - P\left[(i, j) \in \mathcal{E}^{(\alpha)} | \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}, \vec{\sigma}^{(\alpha)}, \vec{\sigma}^{(\beta)}\right]. \quad (9)$$

In Eq. (8), $C(\sigma_i, \sigma_j)$ represents the propensity that two nodes with community memberships σ_i and σ_j are connected. For simplicity, we assume that such a propensity can be written as in Eq. (6), and that the value of the parameter μ is known.

In the second experimental setting, we pretend that layer α is formed by all edges in the set $(\mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}) \setminus \mathcal{E}_{\text{train}}^{(\beta)}$, and layer β is composed of all edges in the set $(\mathcal{E}^{(\alpha)} \cup \mathcal{E}^{(\beta)}) \setminus \mathcal{E}_{\text{train}}^{(\alpha)}$. We then use Eq. (1) to estimate the degree sequences $\vec{k}_{\text{train}}^{(\alpha)}$ and $\vec{k}_{\text{train}}^{(\beta)}$, and the Louvain community detection algorithm to infer the layer community structures $\vec{\sigma}_{\text{train}}^{(\alpha)}$ and $\vec{\sigma}_{\text{train}}^{(\beta)}$. Finally, we use the classifier of Eq. (8), where $\vec{k}_{\text{train}}^{(\alpha)}$ and $\vec{k}_{\text{train}}^{(\beta)}$ respectively replace the unknown ground-truth vectors $\vec{k}^{(\alpha)}$ and $\vec{k}^{(\beta)}$, and $\vec{\sigma}_{\text{train}}^{(\alpha)}$ and $\vec{\sigma}_{\text{train}}^{(\beta)}$ are used in place of $\vec{\sigma}^{(\alpha)}$ and $\vec{\sigma}^{(\beta)}$, respectively. The parameter ν_{train} of Eq. (6) is estimated with Eq. (7).

E. Modified Wu *et al.* classifier

The original algorithm by Wu *et al.* is conceived to solve a more complicated problem than the one considered here [28]. Specifically, it aims at predicting the existence of an edge, and if the edge exists, it aims at classifying the flavor of the edge. In the classification part of the problem, an edge can belong to one layer, the other, or both of them.

We modify the Wu *et al.* algorithm to solve the classification problem considered in this paper. We use this classifier only in the second experimental setting, where partial information on the multiplex edges is provided with the training sets $\mathcal{E}_{\text{train}}^{(\alpha)}$ and $\mathcal{E}_{\text{train}}^{(\beta)}$. As in its original formulation, also here the classifier takes advantage of the expectation-maximization (EM) algorithm. First, we define the known degree of node i in layer α as

$$z_i^{(\alpha)} = \sum_{(r,s) \in \mathcal{E}_{\text{train}}^{(\alpha)}} \delta(i, r) + \delta(i, s). \quad (10)$$

A similar equation is used to define $z_i^{(\beta)}$. Then, we initialize $P[(i, j) \in \mathcal{E}_{\text{test}}^{(\alpha)} | \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}] = P[(i, j) \in \mathcal{E}_{\text{test}}^{(\beta)} | \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}] = 1/2$ for all edges in the test set. We then iterate the following equations:

$$k_i^{(\alpha)} = z_i^{(\alpha)} + \sum_{(r,s) \in \mathcal{E}_{\text{test}}^{(\alpha)}} [\delta(i, r) + \delta(i, s)] P[(r, s) \in \mathcal{E}_{\text{test}}^{(\alpha)} | \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}] \quad (11)$$

and

$$P[(i, j) \in \mathcal{E}_{\text{test}}^{(\alpha)} | \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}] = \frac{k_i^{(\alpha)} k_j^{(\alpha)}}{k_i^{(\alpha)} k_j^{(\alpha)} + k_i^{(\beta)} k_j^{(\beta)}}. \quad (12)$$

We use a similar expression as of Eq.(11) for $k_i^{(\beta)}$. Note that $P[(i, j) \in \mathcal{E}_{\text{test}}^{(\alpha)} | \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}] = 1 - P[(i, j) \in \mathcal{E}_{\text{test}}^{(\beta)} | \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}]$. At the end of each iteration, we check for convergence by comparing the predicted values of the probabilities $P[(i, j) \in \mathcal{E}_{\text{test}}^{(\alpha)} | \vec{k}^{(\alpha)}, \vec{k}^{(\beta)}]$ with those predicted in the previous iteration of the algorithm. In our tests, the condition for convergence is satisfied if the difference between consecutive values of the estimated probabilities is smaller than $\epsilon = 10^{-3}$ for all edges in the test set.

F. Multiplex networks

Synthetic multiplex networks

We begin by generating a degree sequence from the power-law distribution $P(k) \sim k^{-\gamma}$ for $k \in [3, k_{\text{max}}]$, and $P(k) = 0$ otherwise. As prescribed in Ref. [39], we set $k_{\text{max}} = \sqrt{N}$ if $2 < \gamma \leq 3$, and $k_{\text{max}} = N^{1/(\gamma-1)}$ for $\gamma > 3$. We consider various values of the degree exponent γ .

To generate positively correlated degree sequences, we sort the obtained degree sequence and use it for layer α , i.e., $k_1^{(\alpha)} \leq k_2^{(\alpha)} \leq \dots \leq k_N^{(\alpha)}$. The degree sequence of layer β is a copy of the one of layer α , i.e., $k_i^{(\beta)} = k_i^{(\alpha)}$ for all $i = 1, \dots, N$. This condition creates perfectly correlated degree sequences. To decrease correlation between the degree sequences of the layers, we swap the labels of random pairs of nodes in layer β . The level of correlation is dependent on the fraction of pairs that undergo swapping. If enough pairs of node labels are swapped, the two degree sequences are completely uncorrelated. The resulting degree sequences are then independently used to generate the network layers of the multiplex using the configuration model [32].

To obtain anticorrelated degree sequences, we start from a sorted degree sequence for layer α . For layer β , we use $k_i^{(\beta)} = k_{N-i}^{(\alpha)}$ for all $i = 1, \dots, N$. This fact ensures that the two degree sequences are maximally anticorrelated. Also here, we swap the labels of a certain fraction of pairs of nodes in layer β to decrease the level of correlation between the two degree sequences. The resulting degree sequences are then independently used to generate the network layers of the multiplex using the configuration model [32].

To study the effect of correlation between the layerwise community structures on the multiplex reconstruction problem we generate network layers using the LFR model [33]. We begin by generating an instance of the LFR model with given set of parameters. We fix the value of the community size power-law exponent $\tau = 1.0$ and the average degree $\langle k \rangle = 5$ or $\langle k \rangle = 20$. We vary the degree exponent γ . We do not impose any constraint on the size and number of communities. We consider various values of the mixing parameter μ . The same network instance is used as topology for both network layers α and β . To reduce the edge overlap between the layers to a negligible value without altering the correlation of the layers' community structure, we swap at random the labels of all pairs of nodes within the same community in layer β . To reduce the correlation among the structure of the network layers, we swap the labels for a random fraction of pairs of nodes in layer β .

We analyze several real-world multiplex networks. Even if some dataset consist of more than two layers, our tests are performed considering two layers at a time and all possible pairings of the layers. For each combination of the layers, edges shared by both layers are deleted and no information on their existence is considered in the MRP. Then, the set of nodes in the corresponding multiplex is given by the union of the sets of nodes of the two individual layers. Details on the datasets analyzed in the paper are reported in Table I. In the table, we report also values of the modularity valide for the best partition detected by the Louvain algorithm on each layer. We further measure the normalized mutual information between the layer-wise partitions to assess their similarity.

APPENDIX

We find that the performance of the proposed classifier improves as the size of the network increases, whereas remains almost identical as the average degree of the network is varied. We report results of this type of analysis in Fig. 5.

The analysis of three additional real-world multiplex networks is summarized in Fig. 6.

-
- [1] M. E. J. Newman, "Network structure from rich but noisy data," *Nature Physics* 2018 14:6 **14**, 542–545 (2018).
 - [2] Ginestra Bianconi, *Multilayer Networks*, Vol. 1 (Oxford University Press, 2018).
 - [3] S. Boccaletti, G. Bianconi, R. Criado, C. I. del Genio, J. Gómez-Gardeñes, M. Romance, I. Sendiña-Nadal, Z. Wang, and M. Zanin, "The structure and dynamics of multilayer networks," *Physics Reports* **544**, 1–122 (2014), arXiv:1407.0742.
 - [4] Manlio De Domenico, Vincenzo Nicosia, Alexandre Arenas, and Vito Latora, "Structural reducibility of multilayer networks," *Nature Communications* **6**, 1–9 (2015).
 - [5] E. Vasilyeva, A. Kozlov, K. Alfaro-Bittner, D. Musatov, A. M. Raigorodskii, M. Perc, and S. Boccaletti, "Multilayer representation of collaboration networks with higher-order interactions," *Scientific Reports* **11** (2021), 10.1038/S41598-021-85133-5.
 - [6] Filippo Radicchi and Alex Arenas, "Abrupt transition in the structural formation of interconnected networks," *Nature Physics* **9**, 717–720 (2013).
 - [7] Sergio Gomez, Albert Diaz-Guilera, Jesus Gomez-Gardenes, Conrad J Perez-Vicente, Yamir Moreno, and Alex Arenas, "Diffusion dynamics on multiplex networks," *Physical review letters* **110**, 028701 (2013).
 - [8] Federico Battiston, Vincenzo Nicosia, Mario Chavez, and Vito Latora, "Multilayer motif analysis of brain networks," *Chaos: An Interdisciplinary Journal of Nonlinear Science* **27**, 047404 (2017), publisher: American Institute of Physics.
 - [9] Sergey V. Buldyrev, Roni Parshani, Gerald Paul, H. Eugene Stanley, and Shlomo Havlin, "Catastrophic cascade of failures in interdependent networks," *Nature* **464**, 1025–1028 (2010), number: 7291 Publisher: Nature Publishing Group.
 - [10] Alessio Cardillo, Jesús Gómez-Gardeñes, Massimiliano Zanin, Miguel Romance, David Papo, Francisco del Pozo, and Stefano Boccaletti, "Emergence of network features from multiplexity," *Scientific Reports* 2013 3:1 **3**, 1–6 (2013).
 - [11] Manlio De Domenico, Albert Solé-Ribalta, Elisa Omodei, Sergio Gómez, and Alex Arenas, "Ranking in interconnected multilayer networks reveals versatile nodes," *Nature Communications* **6** (2015), 10.1038/ncomms7868.
 - [12] Manlio De Domenico, Clara Granell, Mason A. Porter, and Alex Arenas, "The physics of spreading processes in multilayer networks," *Nature Physics* **12**, 901–906 (2016), number: 10 Publisher: Nature Publishing Group.
 - [13] Marina Diakonova, Vincenzo Nicosia, Vito Latora, and Maxi San Miguel, "Irreducibility of multilayer network dynamics: the case of the voter model," *New Journal of Physics* **18**, 023010 (2016), publisher: IOP Publishing.
 - [14] James P. Gleeson, Kevin P. O'Sullivan, Raquel A. Baños, and Yamir Moreno, "Effects of Network Structure, Competition and Memory Time on Social Spreading Phenomena," *Physical Review X* **6**, 021019 (2016), publisher: American Physical Society.
 - [15] Guanwen Zeng, Daqing Li, Shengmin Guo, Liang Gao, Ziyou Gao, H. Eugene Stanley, and Shlomo Havlin, "Switch between critical percolation modes in city traffic dynamics," *Proceedings of the National Academy of Sciences of the United States of America* **116**, 23–28 (2019), arXiv:1709.03134.
 - [16] Andrea Santoro and Vincenzo Nicosia, "Algorithmic complexity of multiplex networks," *Phys. Rev. X* **10**, 021069 (2020).
 - [17] Saeed Osat, Filippo Radicchi, and Fragkiskos Papadopoulos, "k -core structure of real multiplex networks," *Physical Review Research* **2** (2020), 10.1103/PHYSREVRSEARCH.2.023176/FIGURES/1/THUMBNAIL.
 - [18] Saeed Osat, Ali Faqeeh, and Filippo Radicchi, "Optimal perco-

Dataset	α	β	N	$ \mathcal{E}^{(\alpha)} $	$ \mathcal{E}^{(\beta)} $	$Q^{(\alpha)}$	$Q^{(\beta)}$	NMI
arXiv [35]	physics.data-an	cond-mat.dis-nn	7,187	11,929	4,785	0.89	0.96	0.77
	physics.data-an	cond-mat.stat-mech	5,963	13,326	1,423	0.86	0.98	0.75
	cond-mat.dis-nn	cond-mat.stat-mech	4,342	6,839	2,080	0.91	0.97	0.78
D. Melanogaster [36]	direct	suppressive	7,519	23,911	1,798	0.45	0.64	0.49
	direct	additive	7,486	23,928	1,376	0.46	0.67	0.48
	suppressive	additive	1,005	1,395	956	0.66	0.73	0.59
C. Elegans [37]	electric	chem. monadic	273	406	777	0.67	0.51	0.37
	electric	chem. polyadic	277	355	1,541	0.70	0.44	0.38
	chem. monadic	chem. polyadic	273	258	1,073	0.70	0.40	0.37
London [38]	underground	overground	321	301	72	0.82	0.78	0.67
	underground	dlr	311	312	46	0.83	0.70	0.69
	overground	dlr	126	83	46	0.77	0.69	0.70
S. Pombe [36]	phys. assoc.	dir. interact.	2,577	1,038	6,622	0.87	0.51	0.51
	phys. assoc.	colocalization	3,009	1,431	30,752	0.84	0.23	0.44
	dir. interact.	colocalization	3,782	6,940	30,677	0.52	0.23	0.40
H. Sapiens [36]	phys. assoc.	dir. interact.	17,415	37,361	72,004	0.53	0.41	0.32
	phys. assoc.	colocalization	15,254	82,050	16,189	0.42	0.64	0.35
	dir. interact.	colocalization	3,782	6,940	30,677	0.52	0.23	0.40
R. Norvegicus [36]	phys. assoc.	dir. interact.	2,520	2,610	832	0.70	0.93	0.63
	phys. assoc.	colocalization	2,065	2,757	88	0.71	0.91	0.59
	dir. interact.	colocalization	1,074	991	100	0.92	0.91	0.78

Table I. **Real-world multiplex networks.** From left to right, we report the name of the dataset and reference where the dataset was introduced, the name of the network layers that compose the multiplex, the number of nodes within the two layers, the number of edges for each of the two layers, the modularity of the best Louvain partition of each layer, the normalized mutual information between the two partitions of the two layers.

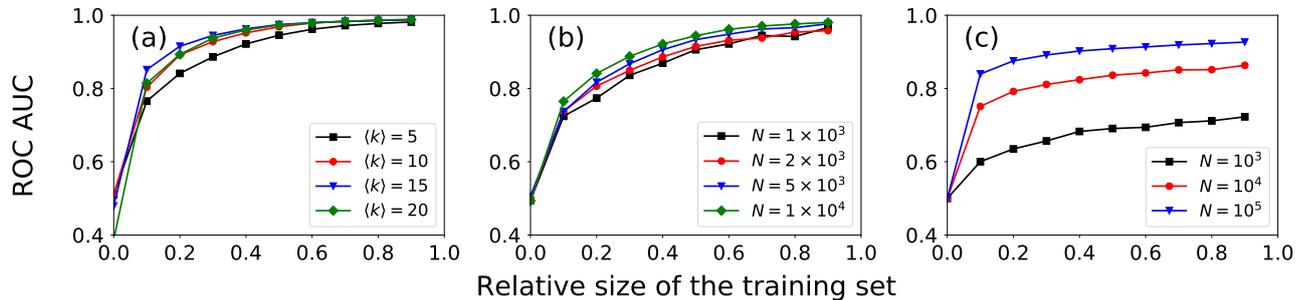


Figure 5. **Reconstruction of synthetic multiplex networks** (a) We display the ROC AUC of the degree- and community-based (DC) classifier as a function of the relative size of the training set. We consider multiplex networks with pre-imposed community structure with size $N = 10,000$ and power-law degree distribution $P(k) \sim k^{-\gamma}$, with $\gamma = 2.1$. The strength of the community structure is determined by the mixing parameter $\mu = 0.1$. Communities have size obeying a power-law distribution with exponent $\tau = 1.0$. Different curves represent different values of average degree $\langle k \rangle$. Maximum degree is $k_{\max} = \sqrt{N}$. (b) Same as in panel (a), but for networks with average degree $\langle k \rangle = 5$. Different curves are obtained for different N values. (c) We display the ROC AUC of the degree-based (D) classifier as a function of the relative size of the training set. Networks generated according to the configuration model with $\gamma = 2.1$, $k_{\min} = 3$, and $k_{\max} = \sqrt{N}$. Different curves are obtained for different N values.

- lation on multiplex networks,” Nature communications **8**, 1–7 (2017).
- [19] Kyu-Min Lee, Byungjoon Min, and Kwang-Il Goh, “Towards real-world complexity: an introduction to multiplex networks,” The European Physical Journal B **88**, 1–20 (2015).
- [20] Mikko Kivelä, Alex Arenas, Marc Barthélemy, James P Gleeson, Yamir Moreno, Mason A Porter, and Ernesto Estrada, “Multilayer networks,” Journal of Complex Networks **2**, 203–271 (2014).
- [21] Manlio De Domenico, Albert Solé-Ribalta, Emanuele Cozzo, Mikko Kivelä, Yamir Moreno, Mason A Porter, Sergio Gómez, and Alex Arenas, “Mathematical formulation of multilayer networks,” Physical Review X **3**, 041022 (2013).
- [22] Michael Szell, Renaud Lambiotte, and Stefan Thurner, “Multirelational organization of large-scale social networks in an online world,” Proceedings of the National Academy of Sciences **107**, 13636–13641 (2010).
- [23] Peter J Mucha, Thomas Richardson, Kevin Macon, Mason A

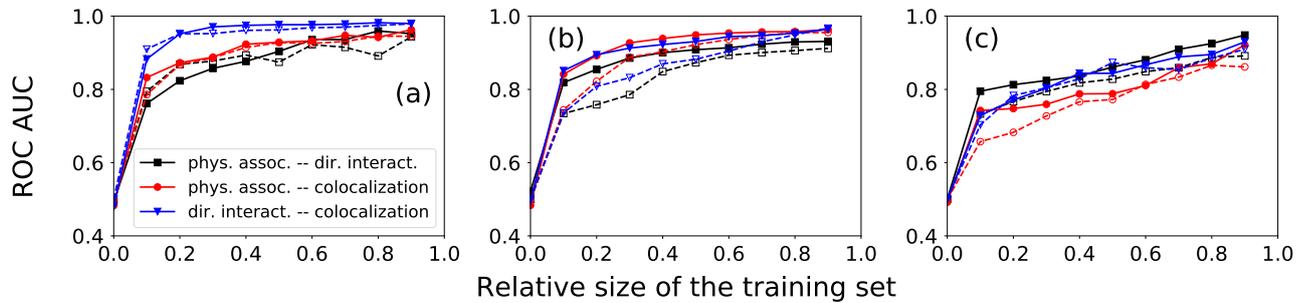


Figure 6. **Reconstruction of real-world multiplex networks** (a) ROC AUC as a function of the relative size of the training set. We report results for the *Saccharomyces Pombe* genetic multiplex network [36]. Results are averaged over 10 realizations. We compare the results achieved with the degree- and community-based (DC, filled symbols / full curves) classifier with those of the Wu *et al.*'s classifier (empty symbols / dashed curves) [28]. Results are averaged over 10 realizations. (b) Same as in panel (a), but for the genetic interactions multiplex network of the *Rattus Norvegicus* [36]. Results are averaged over 10 realizations. (c) Same as in panel (a), but for the genetic interactions multiplex network of the *Rattus Norvegicus* [36]. Results are averaged over 10 realizations.

Porter, and Jukka-Pekka Onnela, “Community structure in time-dependent, multiscale, and multiplex networks,” *science* **328**, 876–878 (2010).

- [24] Ed Bullmore and Olaf Sporns, “Complex brain networks: graph theoretical analysis of structural and functional systems,” *Nature reviews neuroscience* **10**, 186–198 (2009).
- [25] Sol Lim, Filippo Radicchi, Martijn P van den Heuvel, and Olaf Sporns, “Discordant attributes of structural and functional brain connectivity in a two-layer multiplex network,” *Scientific Reports* **9**, 1–13 (2019).
- [26] James P Bagrow and Sune Lehmann, “Recovering lost and absent information in temporal networks,” (2021).
- [27] Aobo Zhang, An Zeng, Ying Fan, and Zengru Di, “Detangling the multilayer structure from an aggregated network,” *New Journal of Physics* **23**, 073046 (2021).
- [28] Mincheng Wu, Jiming Chen, Shibo He, Youxian Sun, Shlomo Havlin, and Jianxi Gao, *Discrimination universally determines reconstruction of multiplex networks*, Tech. Rep. (2020) arXiv:2001.09809v1.
- [29] Lucas Lacasa, Inés P. Mariño, Joaquin Míguez, Vincenzo Nicosia, Édgar Roldán, Ana Lísica, Stephan W. Grill, and Jesús Gómez-Gardeñes, “Multiplex Decomposition of Non-Markovian Dynamics and the Hidden Layer Reconstruction Problem,” *Physical Review X* **8**, 031038 (2018).
- [30] Toni Vallès-Català, Francesco A Massucci, Roger Guimerà, and Marta Sales-Pardo, “Multilayer stochastic block models reveal the multilayer structure of complex networks,” *Physical Review X* **6**, 1–11 (2016), arXiv:1411.1098.
- [31] Brian Karrer and Mark EJ Newman, “Stochastic blockmodels and community structure in networks,” *Physical review E* **83**, 016107 (2011).
- [32] Michael Molloy and Bruce Reed, “A critical point for random graphs with a given degree sequence,” *Random structures & algorithms* **6**, 161–180 (1995).
- [33] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi, “Benchmark graphs for testing community detection algorithms,” *Physical review E* **78**, 046110 (2008).
- [34] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre, “Fast unfolding of communities in large networks,” *Journal of statistical mechanics: theory and experiment* **2008**, P10008 (2008).
- [35] Manlio De Domenico, Andrea Lancichinetti, Alex Arenas, and Martin Rosvall, “Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems,” *Physical Review X* **5**, 011027 (2015).
- [36] Manlio De Domenico, Vincenzo Nicosia, Alexandre Arenas, and Vito Latora, “Structural reducibility of multilayer networks,” *Nature communications* **6**, 1–9 (2015).
- [37] Manlio De Domenico, Mason A Porter, and Alex Arenas, “Muxviz: a tool for multilayer analysis and visualization of networks,” *Journal of Complex Networks* **3**, 159–176 (2015).
- [38] Manlio De Domenico, Albert Solé-Ribalta, Sergio Gómez, and Alex Arenas, “Navigability of interconnected networks under random failures,” *Proceedings of the National Academy of Sciences* **111**, 8351–8356 (2014).
- [39] Michele Catanzaro, Marián Boguná, and Romualdo Pastor-Satorras, “Generation of uncorrelated random scale-free networks,” *Physical review e* **71**, 027103 (2005).