# Comparative analysis on the selection of number of clusters in community detection

Tatsuro Kawamoto[1] and Yoshiyuki Kabashima[2]

[1]*Artificial Intelligence Research Center, National Institute of Advanced Industrial Science and Technology,*
*2-3-26 Aomi, Koto-ku, Tokyo, Japan*
[2]*Department of Mathematical and Computing Science, Tokyo Institute of Technology,*
*W8-45, 2-12-1 Ookayma, Meguro-ku, Tokyo, Japan*

We conduct a comparative analysis on various estimates of the number of clusters in community detection. An exhaustive comparison requires testing of all possible combinations of frameworks, algorithms, and assessment criteria. In this paper we focus on the framework based on a stochastic block model, and investigate the performance of greedy algorithms, statistical inference, and spectral methods. For the assessment criteria, we consider modularity, map equation, Bethe free energy, prediction errors, and isolated eigenvalues. From the analysis, the tendency of overfit and underfit that the assessment criteria and algorithms have becomes apparent. In addition, we propose that the alluvial diagram is a suitable tool to visualize statistical inference results and can be useful to determine the number of clusters.

## I. INTRODUCTION

Community detection is a coarse graining process for networks. Whereas the original dataset, given as a network, possesses information that is quite rich, it is often too microscopic to have its important structures interpreted. For better interpretability, a community detection algorithm summarizes (i.e., clustering or partitioning) the dataset by aggregating the vertices and edges of densely connected components. That is, the detailed relational information of similar vertices is discarded, while preserving an important macroscopic structure. A set of aggregated vertices is regarded as a cluster, or a community.

A straightforward approach, or a framework, for community detection is to optimize an objective function that evaluates the quality of clustering. Another popular approach is based on statistical inference and considers a generative model of a network. It is often formulated using the so-called stochastic block model [1–3] as a generative model, which is a random graph with a modular structure. Therefore, the community structure can be inferred by fitting the network to the model. While these two approaches may seem very different, the former can sometimes be formulated as a limiting case (zero-temperature limit in physics terminology) of the latter, and in this paper it is mainly explained in terms of the latter approach.

In regards to these frameworks, a number of algorithms have been proposed, such as greedy algorithms [4–6], spectral methods [7–10], and inference algorithms such as expectation-maximization (EM) algorithms [11–15] and Monte Carlo methods [16–18], to name a few. In this paper, the following are considered: Louvain method [5] and Infomap [6] for the greedy algorithms, the modularity matrix [8,19] and nonbacktracking matrix [9] for the matrices in the spectral methods, and the EM algorithm with belief propagation (BP) [15,20] for the inference algorithm.

When community detection is performed, the number of clusters $q^*$ needs to be determined. In other words, the

complexity of the model, i.e., the model selection, needs to be specified. This process consists of determining the partition that describes the modular structure most efficiently, or to evaluate whether the obtained partition is statistically significant. Just as the quality of the $q$-way partition varies depending on the very definition of a cluster, the appropriateness of the number of clusters also varies depending on the principle followed. Meanwhile, it is difficult to decide on which principle to apply, given a dataset. Therefore, it is important to investigate the typical behavior and biases of each criterion. For example, some criteria may behave very differently from others in some cases, or some criteria may be more sensitive to the accuracy of a particular algorithm. Moreover, the dangers of underfit and overfit are often not symmetric. In the case of community detection, it is often safer to underfit than to overfit, because the former only implies a different level of coarse graining, while the latter implies the detection of fictitious small clusters.

In this paper, a comparative analysis of various criteria that estimate $q^*$ is conducted. This analysis is distinct from other comparative analyses in the following sense. Whereas the performance of the community detection completely depends on the framework (or objective function), algorithm, and assessment criterion used, it is often the case that a specific combination of them is employed in a benchmark test. For example, Infomap, a greedy algorithm for the map equation [6,21], is a popular algorithm and frequently appears in benchmark tests. However, when the performance of a certain objective function or an assessment criterion is compared with the map equation, it is fair to use a common algorithm. For this reason, the performance of various assessment criteria using the same statistical inference algorithm is compared. In addition, the performance of the same assessment criterion on the same dataset using different algorithms is investigated. Therefore, when a criterion is ill behaved, it can be argued whether it is due to the criterion itself or the algorithm used.

As can be observed below, sometimes, the validation curves of assessment criteria change very gradually. In such a case,

it is not easy to determine the plausible value of $q^*$ from the assessment criteria, and a finer inspection of the partition actually obtained is required. For this purpose, a visualization technique, called the alluvial diagram [22], is proposed as a suitable tool, not only because of the way the network is partitioned but also because it allows the significance level to be evaluated from an inference algorithm.

The remainder of the paper is organized as follows. First, stochastic block models are defined in Sec. II to set the basic framework. Second, the algorithms used to determine the cluster assignments and for the estimate of $q^*$ in Sec. III are introduced. Third, the assessment criteria of the number of clusters $q^*$ is explained in Sec. IV. Then, in Sec. V, the results of the comparative analyses are shown. In Sec. VI, how the alluvial diagram helps determine the number of clusters is explained. Finally, Sec. VII is devoted to the summary and discussion.

## II. STOCHASTIC BLOCK MODELS

Community detection based on a stochastic block model is considered to be the basic framework. The sets of vertices and edges are denoted as $V$ and $E$, respectively. Their respective cardinalities are referred to as $N$ and $L$.

### A. Standard stochastic block model

The most standard version of stochastic block models is constructed as follows. We first consider a set of vertices $V$ without edges. For each vertex, we randomly specify the cluster assignment $\sigma_i \in \{1, \ldots, q\}$, where $i$ is the index of a vertex and the number of clusters $q$ is given as an input. The probability of the cluster size can also be specified. It is a prior distribution of the cluster assignments and is expressed by a multinomial distribution $\prod_i \gamma_{\sigma_i}$. Then, the edges are generated randomly according to the vertex pair's cluster assignment, where the connection probability is specified by an element of the $q \times q$ affinity matrix $\boldsymbol{\omega}$; the edge probability distribution of a vertex pair is given by the Bernoulli distribution. Thus, the likelihood of the stochastic block model is given as

$$p(A,\boldsymbol{\sigma}|\boldsymbol{\omega},\boldsymbol{\gamma},q) = p(A|\boldsymbol{\sigma},\boldsymbol{\omega},q)p(\boldsymbol{\sigma}|\boldsymbol{\gamma},q)$$
$$= \prod_i \gamma_{\sigma_i} \prod_{i<j} \omega_{\sigma_i\sigma_j}^{A_{ij}} \left(1 - \omega_{\sigma_i\sigma_j}\right)^{1-A_{ij}}. \quad (1)$$

When a higher connection probability is provided for pairs of vertices with the same cluster assignment (as compared to pairs of vertices with different cluster assignments), i.e., $\omega_{\sigma\sigma} > \omega_{\sigma\sigma'}$ ($\sigma \neq \sigma'$), a set of random graphs with a community structure can be generated. Generating the stochastic block model is a forward problem, and community detection is its inverse problem, i.e., the inference of $\boldsymbol{\gamma}$, $\boldsymbol{\omega}$, and $\boldsymbol{\sigma}$.

### B. Degree-corrected stochastic block model with restricted model-parameter space

Whereas the standard stochastic block model is very flexible, it is often not suitable to fit real-world networks, mainly because it can only have a binomial degree distribution, which is not true in many datasets. To resolve this problem, the

so-called degree-corrected stochastic block model [3] was proposed. Following Ref. [3], the Bernoulli distribution for the edge probability of each vertex pair was approximated with the Poisson distribution, which is justified when the network is sparse. In this model, it is assumed that the mean of the Poisson distribution depends on the degrees of the vertex pair as well as on the affinity matrix. Hence, for a given affinity matrix $\boldsymbol{\omega}$ and the number of clusters $q$, the likelihood of the degree-corrected stochastic block model is given as

$$p(A,\boldsymbol{\sigma}|\boldsymbol{\omega},q) = p(A|\boldsymbol{\sigma},\boldsymbol{\omega},q)p(\boldsymbol{\sigma}|q)$$
$$= \prod_{i<j} \left(d_i \omega_{\sigma_i\sigma_j} d_j\right)^{A_{ij}} e^{-d_i \omega_{\sigma_i\sigma_j} d_j}, \quad (2)$$

where $d_i$ is the degree of vertex $i$. Here and hereafter, the uniform prior distribution for $p(\boldsymbol{\sigma})$ was considered for simplicity. Moreover, we neglected the existence of self-loops, which is also justified when the network is sparse. The log likelihood reads

$$\log p(A,\boldsymbol{\sigma}|\boldsymbol{\omega}) = \sum_{i<j} \left[ A_{ij} \log \left(d_i \omega_{\sigma_i\sigma_j} d_j\right) - d_i \omega_{\sigma_i\sigma_j} d_j \right], \quad (3)$$

where we neglected a constant term.

While the stochastic block model of Eq. (3) is able to express various modular structures, hereafter, we restrict our interest to the community structure. The affinity matrix $\boldsymbol{\omega}$ is then restricted to the form

$$\omega_{\sigma\sigma'} = \begin{cases} \omega_{\text{in}} & (\sigma = \sigma'), \\ \omega_{\text{out}} & (\sigma \neq \sigma'). \end{cases} \quad (4)$$

In other words, only whether a vertex pair is assigned to the same cluster or not is distinguished. This restriction to the inference algorithm using BP was proposed in Ref. [23]. One of the reasons why this restriction is employed is because it can considerably reduce the computational cost, so that performance on large networks with many clusters can be evaluated. Another reason is because some assessment criteria compared in this paper are specialized to the community structure, and their generalizations to general modular structures are not known.

The log likelihood Eq. (3) is then simplified to

$$\log p(A,\boldsymbol{\sigma}|\boldsymbol{\omega}) = \sum_{i<j} \delta_{\sigma_i\sigma_j} \left[ A_{ij} \log \frac{\omega_{\text{in}}}{\omega_{\text{out}}} - (\omega_{\text{in}} - \omega_{\text{out}})d_i d_j \right]$$
$$+ \sum_{i<j} [A_{ij} \log(\omega_{\text{out}} d_i d_j) - \omega_{\text{out}} d_i d_j]. \quad (5)$$

Note that the second sum does not depend on the cluster assignment $\boldsymbol{\sigma}$. We consider this stochastic block model for community detection.

## III. COMMUNITY DETECTION ALGORITHMS

### A. Statistical inference

The goal of community detection is to determine the set of cluster assignments $\boldsymbol{\sigma}$; it is a hidden variable, and when the model parameter $\boldsymbol{\omega}$ is learned for a given number of clusters $q$ fitting the network for the stochastic block model is carried out by maximizing the marginal log likelihood or, equivalently,

minimizing the free energy:

$$f(\boldsymbol{\omega}, q) = -\log \sum_{\boldsymbol{\sigma}} p(A, \boldsymbol{\sigma} | \boldsymbol{\omega}, q). \tag{6}$$

Unfortunately, this optimization problem is computationally difficult in general, and a number of approximate methods have been proposed in the literature. The EM algorithm is employed in this paper, which is a popular method for fitting the stochastic block model. To obtain the minimum of the free energy, the EM algorithm iteratively optimizes the distribution of the hidden variable $\boldsymbol{\sigma}$ with a fixed model parameter $\boldsymbol{\omega}$ (E step) and the optimization of $\boldsymbol{\omega}$ with a fixed distribution of $\boldsymbol{\sigma}$ (M step). For the E step, we use the BP algorithm which will be explained later in this section. Thus, we obtain the probability distribution of the cluster assignment for each vertex, such that Eq. (6) is expected to be minimized. Hereafter, we often omit the number of clusters $q$ in the argument, which is always given as an input; we try various values of $q$ for model assessment.

When the affinity matrix $\boldsymbol{\omega}$ is fixed as a constant in the E step, the free energy reads

$$f(\boldsymbol{\omega}, q) = \text{const} - \log \sum_{\boldsymbol{\sigma}} e^{2L\beta Q(\boldsymbol{\sigma})}, \tag{7}$$

where

$$Q(\boldsymbol{\sigma}) = \frac{1}{2L} \sum_{i<j} \delta_{\sigma_i \sigma_j} \left[ A_{ij} - \alpha \frac{d_i d_j}{2L} \right], \tag{8}$$

$$\alpha = \frac{2L}{\beta}(\omega_{\text{in}} - \omega_{\text{out}}), \quad \beta = \log \frac{\omega_{\text{in}}}{\omega_{\text{out}}} \tag{9}$$

are the modularity function $Q(\boldsymbol{\sigma})$, resolution parameter $\alpha$ [24], and inverse temperature $\beta$, respectively. This indicates that modularity maximization can be regarded as a special case of the inference using the stochastic block model; the partition with the maximum modularity coincides with the result of the statistical inference when the entropic effect is ignored, or $\beta \to \infty$. This is known as the maximum *a posteriori* (MAP) estimate [25]. The connection between likelihood maximization and modularity maximization was first discussed in Ref. [26] for $q = 2$ in the context of spectral graph partitioning; the above relation was pointed out in Ref. [23], which discusses a finite temperature formulation of the modularity maximization. It is known that $\beta$ also plays the role of a resolution parameter [27] that controls the typical scale of clusters.

In Refs. [23,27], $\alpha$ is set to unity and $\beta$ is treated as an input parameter, which corresponds to fitting a network with a fixed affinity matrix. However, it is more natural to learn them instead. The learning of $\omega_{\text{in}}$ and $\omega_{\text{out}}$ can be carried out in a straightforward manner. They are obtained as the values that minimize the free energy, Eq. (7). The derivatives with respect to the model parameters [28] yield

$$\hat{\omega}_{\text{in}} = \frac{\sum_{(i,j)\in E} \langle \delta_{\sigma_i \sigma_j} \rangle}{\sum_{i<j} d_i d_j \langle \delta_{\sigma_i \sigma_j} \rangle}, \tag{10}$$

$$\hat{\omega}_{\text{out}} = \frac{\sum_{(i,j)\in E} \left(1 - \langle \delta_{\sigma_i \sigma_j} \rangle\right)}{\sum_{i<j} d_i d_j \left(1 - \langle \delta_{\sigma_i \sigma_j} \rangle\right)}, \tag{11}$$

where $\delta_{\sigma \sigma'}$ is the Kronecker delta, $\langle \cdots \rangle$ is the average with respect to the current estimate of the posterior distribution

$p(\boldsymbol{\sigma}|A, \boldsymbol{\omega})$, and the hat notation indicates the estimated quantity. Let $n_\sigma = \sum_i \langle \delta_{\sigma \sigma_i} \rangle$ denote the number of nodes within cluster $\sigma$. As mentioned in Ref. [29], if we assume that $p(\boldsymbol{\sigma}|A, \boldsymbol{\omega})$ is the distribution that prevents $n_\sigma$ from fluctuating significantly, i.e., $\langle n_\sigma^2 \rangle \approx \langle n_\sigma \rangle^2$,

$$\sum_{i<j} d_i d_j \langle \delta_{\sigma_i \sigma_j} \rangle \approx \frac{1}{2} \sum_\sigma \left\langle \sum_i d_i \delta_{\sigma \sigma_i} \right\rangle^2. \tag{12}$$

We also assumed that the overcounting for $i = j$ in the sum is negligible. Then, Eqs. (10) and (11) can be approximated as

$$\hat{\omega}_{\text{in}} = 2 \frac{\sum_\sigma \sum_{(i,j)\in E} \langle \delta_{\sigma \sigma_i} \delta_{\sigma \sigma_j} \rangle}{\sum_\sigma \left( \sum_i d_i \langle \delta_{\sigma \sigma_i} \rangle \right)^2}, \tag{13}$$

$$\hat{\omega}_{\text{out}} = 2 \frac{L - \sum_\sigma \sum_{(i,j)\in E} \langle \delta_{\sigma \sigma_i} \delta_{\sigma \sigma_j} \rangle}{(2L)^2 - \sum_\sigma \left( \sum_i d_i \langle \delta_{\sigma \sigma_i} \rangle \right)^2}. \tag{14}$$

Note that the update of model parameters only costs linear time; therefore, it is not a bottleneck in the algorithm.

To evaluate the probability of cluster assignment for each vertex, BP is used (see Refs. [13,20,23,30] for details). The marginal probability $\psi_{\sigma_i}^i$ of vertex $i$'s cluster assignment $\sigma_i$ can be obtained by marginalizing the likelihood Eq. (2). Using the tree approximation, which is valid for sparse networks, it can be expressed as

$$\psi_{\sigma_i}^i = \frac{1}{Z^i} \prod_{k \in \partial i} \left[ \sum_{\sigma_k} \psi_{\sigma_k}^{k \to i} e^{\beta \delta_{\sigma_i \sigma_k}} \right]$$

$$\times \prod_{\ell \notin i \cup \partial i} \left[ \sum_{\sigma_\ell} \psi_{\sigma_\ell}^{\ell \to i} e^{-\alpha \beta \frac{d_i d_\ell}{2L} \delta_{\sigma_i \sigma_\ell}} \right]$$

$$= \frac{1}{Z^i} \prod_{k \in \partial i} \left[ 1 + \psi_{\sigma_i}^{k \to i} (e^\beta - 1) \right]$$

$$\times \prod_{\ell \notin i \cup \partial i} \left[ 1 + \psi_{\sigma_i}^{\ell \to i} \left( e^{-\alpha \beta \frac{d_i d_\ell}{2L}} - 1 \right) \right], \tag{15}$$

where $\partial i$ denotes the neighboring vertices of $i$. In Eq. (15), $\psi_{\sigma_i}^{k \to i}$ indicates the cavity bias from vertex $k$ to vertex $i$, that is, the marginal probability of $k$ without the marginalization from $i$, and $Z^i$ is the normalization factor. Assuming that $\alpha \beta d_i d_\ell / 2L = O(N^{-1})$, we can further approximate $\psi_{\sigma_i}^i$ and $\psi_{\sigma_i}^{i \to j}$ as

$$\psi_{\sigma_i}^i \approx \frac{1}{Z^i} e^{-\frac{\alpha \beta}{2L} d_i \theta_{\sigma_i}} \prod_{k \in \partial i} \left[ 1 + \psi_{\sigma_i}^{k \to i} (e^\beta - 1) \right], \tag{16}$$

$$\psi_{\sigma_i}^{i \to j} \approx \frac{1}{Z^{i \to j}} e^{-\frac{\alpha \beta}{2L} d_i \theta_{\sigma_i}} \prod_{k \in \partial i \setminus j} \left[ 1 + \psi_{\sigma_i}^{k \to i} (e^\beta - 1) \right], \tag{17}$$

respectively, where $\theta_\sigma \equiv \sum_\ell d_\ell \psi_\sigma^\ell$. The cavity bias $\psi_{\sigma_i}^{i \to j}$ is normalized by $Z^{i \to j}$ (see Ref. [23] for details). Using these quantities, we can estimate the elements in Eqs. (13) and (14) as

$$\sum_i d_i \langle \delta_{\sigma \sigma_i} \rangle = \sum_i d_i \psi_\sigma^i = \theta_\sigma, \tag{18}$$

$$\sum_\sigma \langle \delta_{\sigma\sigma_i} \delta_{\sigma\sigma_j} \rangle = \sum_\sigma \frac{1}{Z^{ij}} \psi_\sigma^{i\to j} d_i \omega_{\mathrm{in}} d_j \psi_\sigma^{j\to i}$$

$$= \frac{\omega_{\mathrm{in}} \sum_\sigma \psi_\sigma^{i\to j} \psi_\sigma^{j\to i}}{(\omega_{\mathrm{in}} - \omega_{\mathrm{out}}) \sum_\sigma \psi_\sigma^{i\to j} \psi_\sigma^{j\to i} + \omega_{\mathrm{out}}}$$

$$\text{for } (i,j) \in E. \tag{19}$$

It should be noted that the iteration of Eqs. (13), (14), (16), and (17) does not minimize the free energy Eq. (6) itself, but minimizes its approximated quantity called the Bethe free energy. We show the specific form of the Bethe free energy in Sec. IV A.

The critical values of $\beta$ for the stochastic block model have been discussed in Refs. [23,27]. There are three phases of state, depending on the value of $\beta$ and the strength of the community structure: the retrieval phase, paramagnetic phase, and spin-glass phase. In the retrieval phase, the fixed point of BP with the minimum Bethe free energy correctly indicates the community structure. In the paramagnetic phase, BP converges to the so-called factorized state as the minimum of the Bethe free energy. In the factorized state, for any vertex $i$, the marginal probability distribution of the cluster assignment $\psi_\sigma^i$ has a uniform distribution, $\psi_\sigma^i = 1/q$. In other words, any vertex has an equal probability of joining any cluster. Therefore, the resulting partition does not exhibit any community structure. Finally, the spin-glass phase is the phase in which BP typically does not converge. This is also the case in which the statistically significant community structure cannot be retrieved. In the case of the standard stochastic block model with equal size clusters, for a given number of clusters $q^*$, the critical value of $\beta$ between the paramagnetic phase and the spin-glass phase obtained by the stability of the factorized state against a random perturbation is

$$\beta^* = \log\left(\frac{q^*}{\sqrt{c}-1} + 1\right), \tag{20}$$

where $c$ is the average degree. The lower bound estimate of $\beta$ that prevents BP from going into the paramagnetic phase is given by

$$\beta_0 = \log\left(\frac{q^*}{c-1} + 1\right). \tag{21}$$

In practice, it cannot be uniquely determined whether BP belongs to the retrieval phase, paramagnetic phase, or spin-glass phase, because real-world networks do not precisely emulate the stochastic block model. However, they work as the reference values of $\beta$ to obtain an intuition regarding which phase BP belongs to. In Ref. [23], it is suggested that $\beta = \beta^*$ should be used as an input, because BP is expected to belong to the retrieval phase with this value.

The effect due to the absence of model-parameter learning can be interpreted as follows. Given that the model only distinguishes whether a pair of vertices is in the same cluster or not, the specific values of $\omega_{\mathrm{in}}$ and $\omega_{\mathrm{out}}$ may not be so crucial for the resulting cluster assignment. Conversely, when other statistical quantities such as likelihood or cross-validation errors are considered, erroneous model-parameter estimates may cause a large bias. As we observe in Sec. V, the results of the criteria that depend only on cluster assignments (e.g., modularity and minimum description length of the map equation) are not very

sensitive to model-parameter learning, while the criteria that utilize the model parameters (e.g., the Bethe free energy and cross-validation errors) are ill behaved without learning.

### B. Greedy algorithms

In the previous section, the free energy minimization based on the stochastic block model has been considered. In the limit of $\beta \to \infty$, it reduces to the maximization of the modularity function $Q(\boldsymbol{\sigma})$ in Eq. (8), or the energy minimization. In this case, the probability distribution with respect to $\boldsymbol{\sigma}$ is no longer considered, and our goal here is to find the best cluster assignment for each vertex.

While a number of algorithms have been proposed in the literature, perhaps greedy algorithms, such as the Louvain method [5], are the most widely used in practice. Another greedy algorithm for community detection that we analyze is the Infomap [6], which optimizes the map equation [6] (see Sec. IV C for the details of the map equation). In such algorithms, we assign a unique cluster label for each vertex at the beginning, i.e., $q = N$, and merge and update their assignments as referring to the neighboring vertices to achieve a higher or lower value of the objective function, e.g., $Q(\boldsymbol{\sigma})$. Note that the number of clusters is also determined automatically during the optimization process. Although these greedy algorithms are extremely fast, as we will observe in Sec. V A, they tend to largely overfit when the algorithm is trapped in a local extremum of the energy landscape. The situation is very severe particularly when the landscape is glassy [31].

### C. Spectral methods

Other commonly used algorithms are spectral methods. In this section, the focus is on the case of $q = 2$ (bisection). Let us consider the case of modularity maximization. While maximizing $Q(\boldsymbol{\sigma})$ is originally a discrete optimization problem, the assignments $\boldsymbol{\sigma}$ are relaxed to a real vector $\boldsymbol{x} \in \mathbb{R}^N$ with a spherical normalization constraint, i.e.,

$$\max_{\boldsymbol{x}} \ Q(\boldsymbol{x}) \quad \sum_i x_i^2 = N. \tag{22}$$

Here, $Q(\boldsymbol{x}) = \boldsymbol{x}^\top \mathcal{Q} \boldsymbol{x}$ and $\mathcal{Q}$ is a matrix the element of which is given as

$$\mathcal{Q}_{ij} = A_{ij} - \alpha \frac{d_i d_j}{2L}. \tag{23}$$

This matrix is known as the modularity matrix [8,19]. If Eq. (22) is rewritten using the Lagrange multiplier, an eigenvalue problem is obtained with respect to $\mathcal{Q}$ and the leading eigenvector is expected to be correlated to the optimum assignments. Although $\mathcal{Q}$ is a dense matrix, because it is due to the rank 1 matrix of the second term in Eq. (23), its leading eigenvalues and eigenvectors can be computed efficiently using the power iteration [32].

A more classical version of a spectral method is the one based on the minimization of an objective function called the normalized cut [33]. The normalized cut $f_{\mathrm{Ncut}}$ is defined as, for $\sigma_i \in \{1, 2\}$,

$$f_{\mathrm{Ncut}}(\boldsymbol{\sigma}) \propto \frac{\sum_{i,j} A_{ij} \delta_{\sigma_i,1} \delta_{\sigma_j,2}}{\left(\sum_{i(\sigma_i=1)} d_i\right)\left(\sum_{j(\sigma_j=2)} d_j\right)}. \tag{24}$$
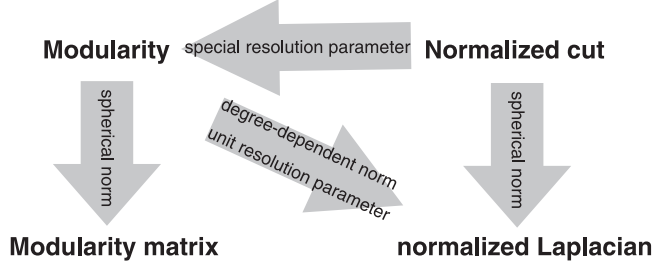
FIG. 1. Relationships with the modularity function, normalized cut, and the corresponding spectral methods in the case of bisection.

Analogous to the case of modularity, the continuous relaxation with the spherical normalization constraint yields the eigenvalue problem with respect to the normalized Laplacian $\mathcal{L}$, which is defined as

$$\mathcal{L} = I - D^{-1/2}AD^{-1/2}, \tag{25}$$

where $I$ is the identity matrix and $D^k \equiv \text{diag}\{d_1^k, \ldots, d_N^k\}$ (see Ref. [7] for details).

The optimizations of the normalized cut and modularity look different. However, it is known that minimizing the normalized cut $f_{\text{Ncut}}(\boldsymbol{\sigma})$ is equivalent to maximizing modularity $Q(\boldsymbol{\sigma})$ with a special choice of the resolution parameter $\alpha$ [34] at the level of discrete optimization. Moreover, when the problem is relaxed to a continuous optimization, it is also possible to formulate the modularity maximization as the eigenvalue problem of the normalized Laplacian [26]; it can be done by imposing a degree-dependent normalization constraint instead of the spherical normalization constraint and setting the resolution parameter $\alpha$ to unity. These relationships are summarized in Fig. 1.

In general, the leading $q$ eigenvectors are expected to be correlated to the optimum $q$-way partition. Thus, to determine the assignment of each vertex, those eigenvectors have to be rounded, e.g., using the K-means method. However, the focus of this paper is only on the eigenvalues, because they are sufficient to estimate the number of clusters.

While the above two spectral methods are based on energy minimization, a spectral method related to the Bethe free energy minimization was proposed in Ref. [9]. The matrix that appears in this method is called the nonbacktracking matrix $\mathcal{B}$, and is derived from the linear stability analysis of the BP algorithm that minimizes the Bethe free energy. The nonbacktracking matrix $\mathcal{B}$ is not a symmetric matrix, and its specific form is given as

$$\mathcal{B} = \begin{pmatrix} 0 & D-I \\ -I & A \end{pmatrix}. \tag{26}$$

A symmetric variant of the nonbacktracking matrix is also proposed in Ref. [10].

In Sec. IV D, we will discuss the properties of the spectra of the above matrices and their use as the assessment criteria of the number of clusters.

## IV. ASSESSMENT CRITERIA OF THE NUMBER OF CLUSTERS

In this section, we explain the assessment criteria of the number of clusters $q^*$. To determine it using an algorithm in which $q$ is given as an input, we assess the quality of the clustering based on a criterion, as we sweep the value of $q$. It should be noted that, although the input value of the number of clusters, namely, the maximum number of clusters that the vertices can be assigned to, is $q$, the resulting partition might have less than $q$ clusters.

### A. Bethe free energy

In the present framework of statistical inference, the most natural assessment is to measure the free energy and observe its saturation as an increment of the number of clusters $q$. When the network is generated by a block model with $q^*$ clusters, the marginal likelihood will not be increased for $q > q^*$. Therefore, we expect that a parsimonious number of clusters can be selected from the saturation of the free energy.

As mentioned above, the algorithm using BP does not minimize the free energy itself. Instead, it minimizes the Bethe free energy as an approximated quantity, and it can be written in terms of the cavity bias $\psi_\sigma^{i \to j}$ and affinity matrix $\boldsymbol{\omega}$ as follows:

$$f_{\text{Bethe}} = -\frac{1}{\beta N} \left( \sum_i \log Z^i - \sum_{(i,j)\in E} \log Z^{ij} - \sum_{(i,j)\notin E} \log \tilde{Z}^{ij} \right), \tag{27}$$

where

$$Z^i = \sum_\sigma e^{-d_i h_{\sigma_i}} \prod_{k\in\partial i} \left[ \sum_{\sigma_k} \psi_{\sigma_k}^{k\to i} d_k \omega_{\sigma_k\sigma_i} d_i \right], \tag{28}$$

$$Z^{ij} = \sum_{\sigma\sigma'} \psi_\sigma^{i\to j} d_i \omega_{\sigma\sigma'} d_j \psi_{\sigma'}^{j\to i} \quad \text{for } (i,j)\in E, \tag{29}$$

$$\tilde{Z}^{ij} = \sum_{\sigma\sigma'} \psi_\sigma^{i\to j}(1 - d_i\omega_{\sigma\sigma'}d_j)\psi_{\sigma'}^{j\to i} \quad \text{for } (i,j)\notin E, \tag{30}$$

$$h_\sigma = \sum_k d_k \sum_{\sigma_k} \psi_{\sigma_k}^k \omega_{\sigma_k\sigma}. \tag{31}$$

Some simple algebra shows that the Bethe free energy here $f_{\text{Bethe}}$ is related to the Bethe free energy in Ref. [23] (which we refer to as $f_{\text{Bethe}}^{\text{mod}}$) as

$$f_{\text{Bethe}} = f_{\text{Bethe}}^{\text{mod}} + C(\omega_{\text{out}}), \tag{32}$$

$$C(\omega_{\text{out}}) = -\frac{c}{2\beta}\left( 2L\omega_{\text{out}} + \log\omega_{\text{out}} + \frac{1}{L}\sum_i d_i\log d_i \right). \tag{33}$$

### B. Modularity

While modularity appeared as an objective function with a fixed $q$ in Sec. III A, it was originally defined as an assessment criterion of the number of clusters [35]. In modularity, the strength of a community structure is measured by comparing the actual network and a randomized network in each cluster. Although the performance of modularity is not considered state

of the art, it has been extensively studied and used as a baseline in many benchmark tests.

Precisely speaking, while the sum is taken over every vertex pair $(i, j)$ $(i < j)$ in Eq. (8), the sum is taken over all possible combinations of vertices (including the case $i = j$) in the original definition, although this does not cause a qualitative difference unless the self-loops are significant. The modularity function of Eq. (8) with the partition obtained by free energy minimization is sometimes distinguished as the retrieval modularity. However, it is referred to as *modularity* in this paper for simplicity. The partition is selected with a maximum modularity, or the parsimonious one among the partitions with a high modularity.

From the viewpoint of statistical inference, the modularity maximization corresponds to a maximum likelihood estimate; i.e., there is no penalty term. In principle, it can still assess the number of clusters because the degrees of freedom of the affinity matrix $\omega$ are restricted as in Eq. (4), and thus the model with a larger $q$ does not contain the model with a smaller $q$ as a subset. In addition, if we tune the resolution parameter $\alpha$, the likelihood varies, and the optimum value $q^*$ changes.

### C. The map equation

Another popular criterion is the map equation [6,21], in which the strength of the community structure is measured in terms of the minimum description length of a random walker. The map equation encodes the moves of a random walker on a given network using multiple codebooks. Specifically, it considers a codebook that encodes moves between clusters, as well as codebooks that encode moves within each cluster. Given that the codewords of different codebooks can be overlapped, a proper assignment of clusters will compress the description length of a random walker. Moreover, by using the codebooks of superclusters, i.e., the clusters of clusters, its hierarchical extension can be performed naturally. The map equation also has an interesting feature in that it allows for the consideration of flow information, e.g., the directedness of edges, although we do not address this point in this paper (see Refs. [6,21] for more details).

The excellent performance of the map equation and its greedy implementation (Infomap) has been shown in numerous articles. As with modularity, one can use the minimum description length of the map equation for model assessment only and perform community detection based on another objective function. It should be noted that the characterization of a cluster in the map equation is not equivalent to that of the stochastic block model. However, when densely connected components exist in a network, the minimum description length of a random walker is further compressed by clustering them; thus, it is expected that an optimal partition in the sense of modularity is also a good partition in the sense of the map equation.

It is also debatable whether we should consider the hierarchical nature of the map equation [21]. The map equation is naturally formulated as a hierarchical clustering, and the fundamental two-level method can be regarded as a truncation of the general multilevel method. Nevertheless, we measure the minimum description length of the two-level method and compare it with other model assessment criteria, because it is not always possible to measure the minimum description
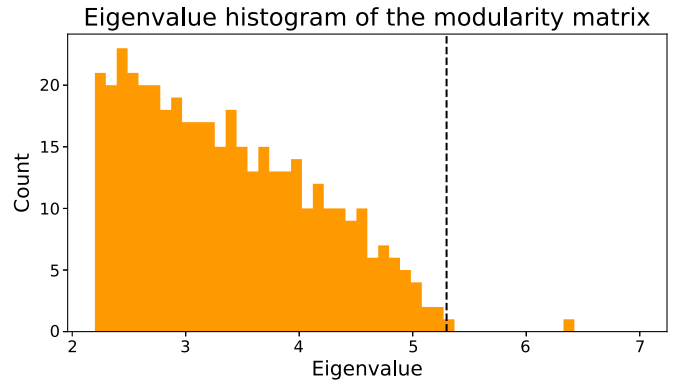


FIG. 2. A part of the histogram of the eigenvalues of the modularity matrix for the standard stochastic block model. The network has $N = 2000$ with two equally sized planted clusters and an average degree equal to 6. The dashed line indicates the estimate of the boundary of the spectral band calculated by the result in Ref. [34].

length in the sense of the multilevel map equation. Whereas the multilevel map equation assumes a hierarchical structure, for example, in the case of partitions using the inference algorithm considered in this paper, each partition with different values of $q$ is independent and is not constrained to constitute a hierarchical structure.

### D. Spectral band

The spectra of matrices $\mathcal{Q}$, $\mathcal{L}$, and $\mathcal{B}$, in Sec. III C can be used to estimate the number of clusters. In the case of a uniform random graph, in the infinite graph size limit, the spectrum of a corresponding matrix exhibits a nonzero spectral density within a finite range. In other words, the spectral band can be observed, as exemplified in Fig. 2; it is often referred to as the *semicircle law* [36] in the case of a symmetric matrix. The spectral band stems purely from the random nature of a network, and if a characteristic structure in a network exists the eigenvalues outside of the spectral band, i.e., isolated eigenvalues, will be observed. As we mentioned in Sec. III C, because the leading eigenvectors are expected to be correlated to the optimum partition, the number of statistically significant clusters can be estimated by counting the isolated eigenvalues.

While it is sometimes possible to evaluate the boundary of the spectral band by visual inspection, it is not trivial and it is preferable to have its estimate. The estimate of the boundary of the spectral band of the modularity matrix $\mathcal{Q}$ was first derived in Ref. [37]; the estimate was then generalized for random networks with arbitrary expected degrees [38] and arbitrary degree sequences [39]. These results, however, assume that the average degree is sufficiently large. A result that is applicable to sparse networks is derived in Ref. [34]; although it is still a mean-field result, it yields the estimate for a random network with an arbitrary degree sequence and it is exact when the network is regular.

Although the boundary estimate of the spectral band of the normalized Laplacian $\mathcal{L}$ is also possible using the mean-field approximation [40], it is known that $\mathcal{L}$ seriously suffers from the emergence of localized eigenvectors; those localized eigenvectors consist of a few elements with very large values and most of the elements are close to zero. These localized

eigenvectors are not correlated to the optimum partition and can deteriorate the estimate of the number of clusters. For this reason, we do not pursue the assessment using the spectrum of $\mathcal{L}$ in this paper.

On the other hand, the nonbacktracking matrix $\mathcal{B}$ tends to avoid the emergence of the localized eigenvectors, and its spectral band has a clear boundary at $\sqrt{\rho(\mathcal{B})}$ [41], where $\rho(\mathcal{B})$ is the spectral radius of $\mathcal{B}$. As seen in Sec. V B, the assessment using the nonbacktracking matrix performs well in many cases. Note that, however, the nonbacktracking matrix is not completely free from the localized eigenvectors [42,43].

### E. Prediction errors

Finally, we explain the cross-validation estimates of prediction errors, which are also useful to estimate the number of clusters (see Ref. [15] for the detailed derivations). Although evaluating cross-validation errors is computationally demanding in general, the leave-one-out cross-validation (LOOCV) is an exceptional case and the corresponding errors can be obtained efficiently using the result of BP [15].

We consider four types of cross-validation errors: the Bayes prediction error, Gibbs prediction error, MAP estimate of the Gibbs prediction error, and Gibbs training error. We refer to $A^{\backslash(i,j)}$ as the adjacency matrix without the knowledge of $A_{ij}$. Given $A^{\backslash(i,j)}$, the cluster assignment probability of $i$ and $j$ is

$$p(\sigma_i, \sigma_j | A^{\backslash(i,j)}) = \psi_\sigma^{i \to j} \psi_{\sigma'}^{j \to i}. \tag{34}$$

Then, the prediction probability $\hat{p}(A_{ij} = 1 | A^{\backslash(i,j)})$ that $i$ and $j$ are connected is

$$\hat{p}(A_{ij} = 1 | A^{\backslash(i,j)})$$
$$= \sum_{\sigma_i, \sigma_j} \hat{p}(A_{ij} = 1 | \sigma_i, \sigma_j) p(\sigma_i, \sigma_j | A^{\backslash(i,j)})$$
$$= \sum_{\sigma\sigma'} \psi_\sigma^{i \to j} d_i \omega_{\sigma\sigma'} d_j \psi_{\sigma'}^{j \to i} = Z^{ij}. \tag{35}$$

Note that the two-point partition function $Z^{ij}$ is the normalization factor in Eq. (19) and is not equivalent to the two-point partition function defined in Ref. [23], which does not have a

probabilistic interpretation. The cross-entropy error function with respect to $\hat{p}(A_{ij}|A^{\backslash(i,j)})$, which is referred to as the Bayes prediction error of LOOCV, $E_{\text{Bayes}}$, is

$$E_{\text{Bayes}}(q) = -\frac{1}{L} \sum_{i<j} \{A_{ij} \log \hat{p}(A_{ij} = 1 | A^{\backslash(i,j)})$$
$$+ (1 - A_{ij}) \log[1 - \hat{p}(A_{ij} = 1 | A^{\backslash(i,j)})]\}. \tag{36}$$

Using the fact that $\omega_{\sigma\sigma'} = O(N^{-1})$, it can be approximated as

$$E_{\text{Bayes}}(q) \simeq 1 - \frac{1}{L} \sum_{(i,j) \in E} \log Z^{ij}, \tag{37}$$

where we neglected the $O(N^{-1})$ term. The Bayes prediction error $E_{\text{Bayes}}$ should be the appropriate choice for assessing models in terms of the predictability of edges when the network is generated by the stochastic block model. However, this is often not the case. Hence, the Gibbs prediction error $E_{\text{Gibbs}}$ is considered, which is a rough estimate of the prediction error compared to $E_{\text{Bayes}}$. While the probability with respect to $\sigma_i$ and $\sigma_j$ is marginalized when the cross-entropy error function is measured in $E_{\text{Bayes}}$, a specific choice is made regarding $\sigma_i$ and $\sigma_j$ first, and the average is taken later in $E_{\text{Gibbs}}$. Thus, we have

$$E_{\text{Gibbs}}(q) \simeq 1 - \frac{1}{L} \sum_{(i,j) \in E} \sum_{\sigma_i, \sigma_j} p(\sigma_i, \sigma_j | A^{\backslash(i,j)})$$
$$\times \log[\hat{p}(A_{ij} = 1 | \sigma_i, \sigma_j)]$$
$$= 1 - \frac{1}{L} \sum_{(i,j) \in E} \sum_{\sigma_i, \sigma_j} \psi_{\sigma_i}^{i \to j} \psi_{\sigma_j}^{j \to i} \log\left(d_i \omega_{\sigma_i \sigma_j} d_j\right)$$
$$= -\frac{\beta}{L} \sum_{(i,j) \in E} \sum_\sigma \psi_\sigma^{i \to j} \psi_\sigma^{j \to i} - \log \omega_{\text{out}} + \text{const}, \tag{38}$$

where we again neglected the $O(N^{-1})$ term. By replacing $\psi_\sigma^{i \to j}$ with the delta function that has a peak at $\operatorname{argmax}_\sigma \psi_\sigma^{i \to j}$, the MAP estimate of the Gibbs prediction error is obtained, which is referred to as $E_{\text{MAP}}$.

The Gibbs training error $E_{\text{training}}$ can be derived in the same manner. In $E_{\text{training}}$, we include the information of $A_{ij}$ for the probability with respect to $\sigma_i$ and $\sigma_j$. Thus, we have

$$E_{\text{training}}(q) \simeq 1 - \frac{1}{L} \sum_{(i,j) \in E} \sum_{\sigma_i, \sigma_j} p(\sigma_i, \sigma_j | A) \log[\hat{p}(A_{ij} = 1 | \sigma_i, \sigma_j)] = 1 - \frac{1}{L} \sum_{(i,j) \in E} \sum_{\sigma_i, \sigma_j} \frac{\psi_{\sigma_i}^{i \to j} d_i \omega_{\sigma_i \sigma_j} d_j \psi_{\sigma_j}^{j \to i}}{Z^{ij}} \log\left(d_i \omega_{\sigma_i \sigma_j} d_j\right)$$
$$= \text{const} - \frac{1}{L} \sum_{(i,j) \in E} \frac{\omega_{\text{out}} \log \omega_{\text{out}} + \beta[\omega_{\text{out}} + (\alpha/2L) \log \omega_{\text{out}}] \sum_\sigma \psi_\sigma^{i \to j} \psi_\sigma^{j \to i}}{\omega_{\text{out}} + (\alpha\beta/2L) \sum_\sigma \psi_\sigma^{i \to j} \psi_\sigma^{j \to i}}. \tag{39}$$

Again, the $O(N^{-1})$ term was neglected.

Note that the complexity of computing the Bethe free energy and the cross-validation errors is considerably reduced by restricting the parameter space of the stochastic block model. While the stochastic block model required a computation of $O(q^2 L)$ in the general case, it is $O(L)$ with the restriction: Eq. (4).

## V. COMPARATIVE ANALYSIS

In this section, a comparative analysis of the assessment of the number of clusters was conducted using synthetic and real-world networks. For the synthetic networks, the planted number of clusters is denoted as $q_{\text{planted}}$.

### A. Assessment using the greedy algorithms

The performance of the greedy algorithms was first examined on the basis of the standard stochastic block model. Figures 3(a) and 3(b) show the number of clusters detected using the Louvain method and the two-level Infomap, respectively. The horizontal axes represent the strength of the community structure $\omega_{\text{out}}/\omega_{\text{in}} \equiv \epsilon$. The Louvain method is a hierarchical clustering algorithm that aims to optimize modularity, while the (two-level) Infomap is a nonhierarchical clustering algorithm that aims to optimize the map equation. For the implementation, we used Ref. [44] for the Louvain method and Ref. [45] for the Infomap.

All instances considered in this section have $q_{\text{planted}} = 2$. Given that the stochastic block model is exactly the model assumed in the inference algorithm, the assessment by the Bethe free energy and some of the prediction errors are known to be very accurate [15,46,47], even when the planted modular structure is very weak.

When the average degree is sufficiently high and the community structure is strong (i.e., $\epsilon \sim 0$), both algorithms correctly detect two clusters. However, when the networks are sparse and the community structure is weak (i.e., $\epsilon \gg 0$), those algorithms tend to largely overfit. Moreover, as shown in Figs. 3(c) and 3(d), the detected number of clusters increases as the network becomes larger. A nonhierarchical clustering algorithm for modularity [48,49] and the multilevel Infomap [21] were also tested. Although the tendency that the hierarchical clusterings slightly prevent overfitting was confirmed, significant differences were not observed.

It should be noted that detecting too many clusters does not readily imply the failure of the algorithm. For example, when the result consists of a few large clusters and many very small clusters, significant clusters can be extracted via a visual inspection. This is actually the case for instances with strong community structures. Otherwise, the sizes of clusters can be broadly distributed, and such a visual inspection may fail. Such situations are exemplified in Fig. 4 for the Infomap.

As a reference to the comparative analysis of the latter sections, we list the results of the greedy algorithms on synthetic and real-world networks in Fig. 5. The descriptions of the networks can be found in Sec. V B and the references therein.

### B. Assessment using the inference algorithm

In this section, the performance of various assessment criteria based on the statistical inference algorithm that were explained in Sec. III A is analyzed. First, the performance on synthetic networks, called the LFR network [50], is analyzed. The LFR network is a random graph model that has a power-law degree distribution and, as a planted structure, a power-law cluster size distribution. The parameters of the LFR networks considered are listed in Table I. Although the LFR network is often analyzed as a random graph that emulates typical
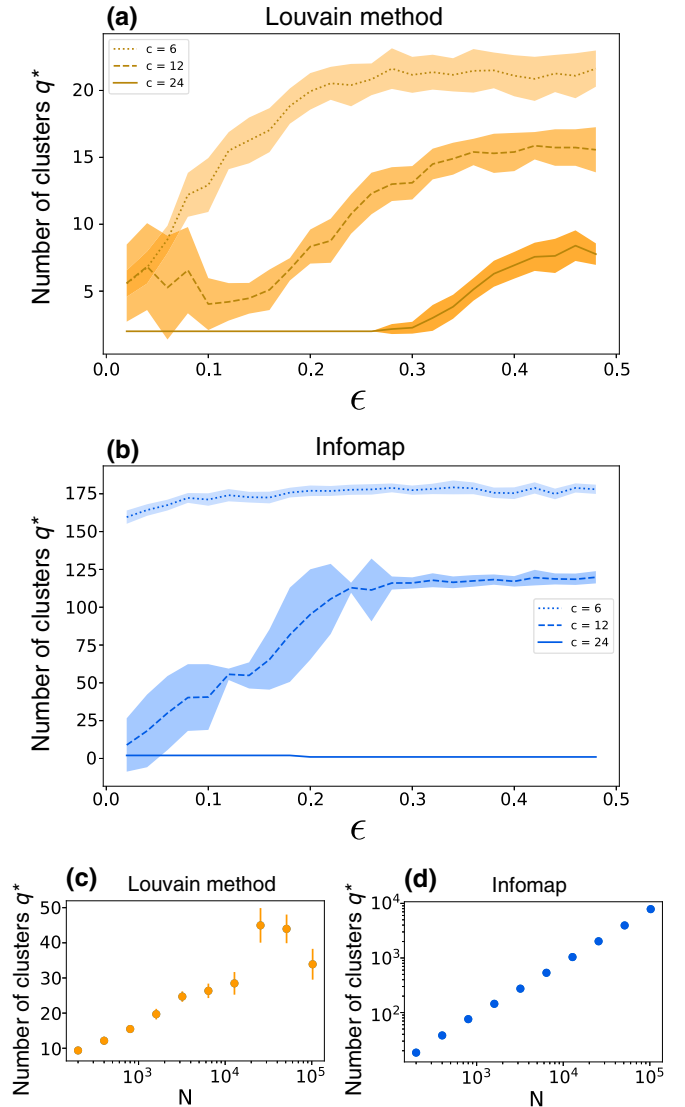


FIG. 3. Numbers of clusters $q^*$ detected by the (a) Louvain method and (b) Infomap for instances of the standard stochastic block model. Each network has two equally sized clusters as a planted structure, and the networks are generated for various values of $\epsilon = \omega_{\text{out}}/\omega_{\text{in}}$, i.e., the strength of community structure. The algorithms were executed 30 times for each network; the resulting number of clusters fluctuates depending on the initial cluster assignments, and the shaded regions show the standard deviations from the mean value. In each case, the stochastic block models of $N = 2\,000$ with the average degrees $c = 6$, 12, and 24 are evaluated. The bottom figures show the $N$ dependence on the estimated number of clusters $q^*$ for the (c) Louvain method and (d) Infomap; the planted structure is of two equally sized clusters with $c = 6$ and $\epsilon = 0.5$, and the experimental procedure is the same as in Figs. 3(a) and 3(b). The error bars indicate the standard deviations.

real-world networks, in this paper it is not argued whether the parameter set investigated is "realistic" or not. In fact, it is not obvious whether the LFR network really emulates typical real-world networks, because, as can be seen from Fig. 4, a broad cluster size distribution can be obtained fictitiously.

Figure 6 shows the result for an instance with a strong community structure (small mixing parameter $\mu$, in terms of
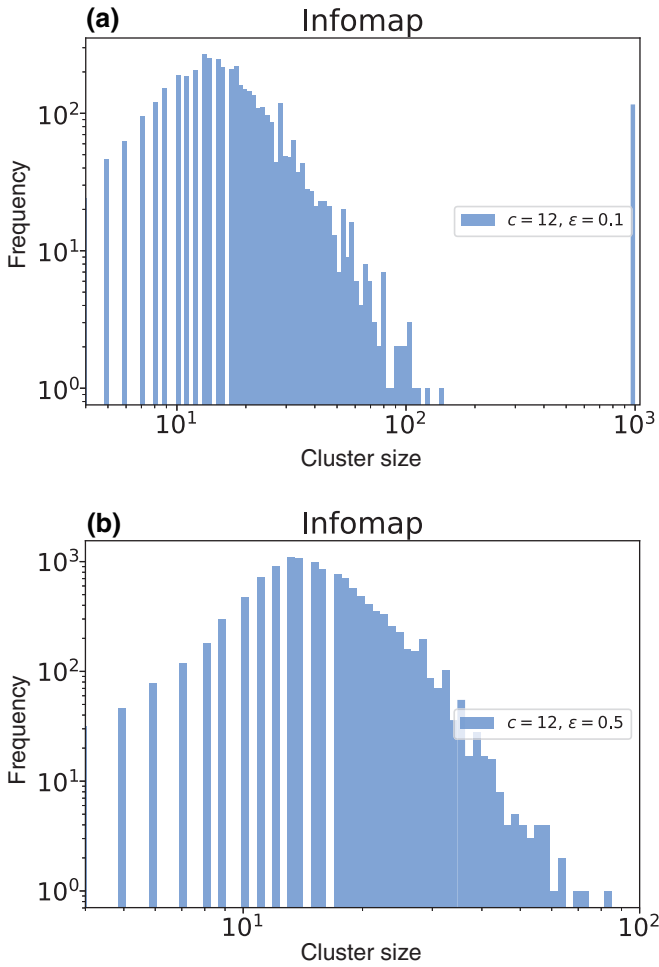
FIG. 4. Histograms of the cluster size distributions detected by the Infomap for the same stochastic block models as in Fig. 3 with (a) $c = 12$ and $\epsilon = 0.1$ and (b) $c = 12$ and $\epsilon = 0.5$. The algorithm is run 100 times on the same network and the results show their cumulative frequencies. When the community structure is strong ($\epsilon = 0.1$), one of the planted clusters seems to be detected. Conversely, only small clusters that are broadly distributed are detected when $\epsilon = 0.5$. Note that the specific form of the cluster size distribution possibly depends on the details of the algorithm.

the LFR network). Although the network has vertices with very large degrees, the cluster sizes are set to be almost equal. For this network, all the criteria support values close to $q_{\text{planted}}$, although the criterion based on the nonbacktracking matrix slightly overfits.

While the values of the model parameters are learned in Fig. 6, we show the result for the same network without the model-parameter learning in Fig. 7. As we can see from Figs. 6 and 7, modularity and the map equation behave similarly in both cases. As we mentioned at the end of Sec. III A, this may be due to the fact that the cluster assignment is not very sensitive to specific values of model parameters, at least when the network has a strong community structure. Conversely, the performance of the Bethe free energy and cross-validation errors change qualitatively, indicating that the learning step cannot be skipped. Note that skipping the learning step does not necessarily mean that it is computationally more efficient.

With an incorrect choice of the affinity matrix $\omega$, it will be more difficult to fit the network. It turns out that BP requires more sweeps until convergence. Therefore, it is more beneficial to optimize the model parameters. The rest of the results in this paper are generated according to the algorithm with model-parameter learning.

The LFR networks with weak community structures are shown in Fig. 8. Figures 8(a)–8(c) represent the results for networks with narrowly peaked planted cluster size distributions. Conversely, Figs. 8(d)–8(f) represent the result for networks with broad planted cluster size distributions. Although it is difficult to thoroughly examine the effect of cluster size distribution, we can at least confirm that the performance of the present algorithm and assessment criteria are not very sensitive to the cluster size distribution.

In the case of sparse networks such that the average degree is of $O(1)$, if the planted community structure is too weak, it becomes fundamentally impossible to retrieve the planted structure. In other words, the network becomes statistically impossible to distinguish from a uniform random graph. The critical strength of the community structure is called the detectability threshold, or the detectability limit [13,51,52]. In terms of the spectral method, it is the case that the leading eigenvalues are buried in the spectral band. In terms of other assessment criteria, the slope of a validation curve becomes flat, or the value at $q = 1$ becomes the minimum. In the case of the stochastic block model with equally sized clusters, this threshold is given by the value of $\epsilon$ that corresponds to $\beta^*$ in Eq. (20), and the paramagnetic phase will be observed beyond the detectability threshold.

For the network in Fig. 8(a), all the criteria we consider behave reasonably, supporting the values close to $q_{\text{planted}}$. For the network in Fig. 8(b), other than the Gibbs prediction error and its MAP estimate, the assessment criteria still support the values near $q_{\text{planted}}$. Indeed, in the case of the stochastic block model, it is known that the Gibbs prediction error tends to underfit near the detectability threshold [15]. Although the value of the (information-theoretic) detectability threshold for the LFR network is not known, the network in Fig. 8(c) may be beyond the detectability threshold. The Gibbs prediction error and Bayes prediction error are minimized or saturated already at $q = 1$ (not shown in the figure). The Bethe free energy exhibits a monotonic behavior, while the values of other criteria behave violently; this implies that the landscapes of the objective functions are glassy.

More importantly, while we observed in Figs. 3 and 5 that the estimates by modularity and the map equation largely overfit when the greedy algorithm is used, the results in Fig. 8 indicate that those criteria behave reasonably when statistical inference is used. Therefore, the shortcomings that we observed in Fig. 3 were not the flaw of the criteria themselves, but of the greedy algorithms and the MAP estimate (i.e., $\beta \to \infty$) framework. In fact, when the assumed model is correct, it is known that the MAP estimate overfits compared to the estimate with the optimum inverse temperature $\beta$ [23,51,53]. The contribution of this paper is that it confirms that the overfitting occurs with the greedy algorithms near the detectability threshold.

While the results in Figs. 8(d) and 8(f) are similar to those in Figs. 8(a)–8(c), the result in Fig. 8(e) is qualitatively different.
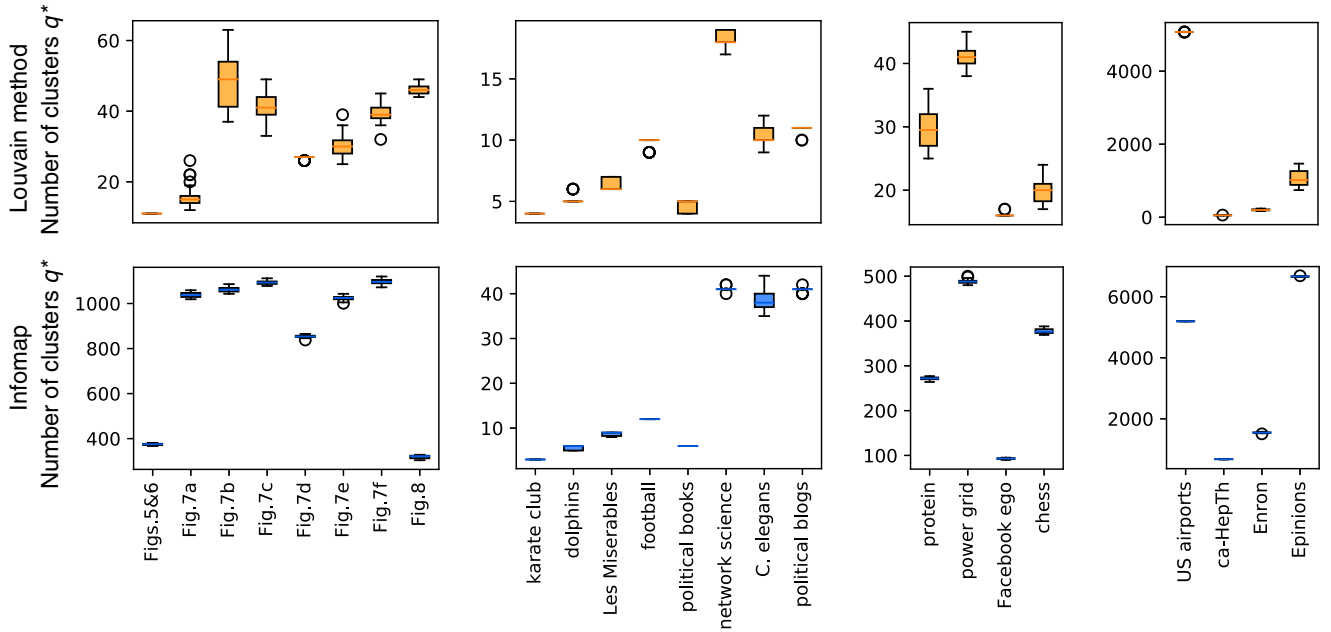
FIG. 5. Box plots of the estimates of the number of clusters $q^*$ using the Louvain method (top) and the two-level Infomap (bottom) on the synthetic and real-world networks that are analyzed in Secs. V B and V C. The algorithms were executed 30 times for each network. Each box represents the range from the upper quantile to the lower quantile, the line in the box represents the median, whiskers from the box represent the upper and lower extremes, and circles represent the outliers, which are significantly far from the upper and lower quantiles.

As the input values of $q$ are increased, at some point, BP converged to the factorized state [54]; as a result, the estimated value of $\beta$ jumps, and some prediction errors become constant. Convergence to the factorized state is a desirable feature of BP; it implies that BP has reached the detectability threshold and that there is no significant structure. Note, however, that it is often difficult to determine whether BP is actually in the paramagnetic phase or the retrieval phase. Given that the factorized state always exists as a fixed point of BP in the

TABLE I. Parameters of the LFR networks. All networks have $N \simeq 10^4$. The average degree $c$, maximum degree $d_{\max}$, size of the smallest cluster $N_{\min}^{\sigma}$, and size of the largest cluster $N_{\min}^{\sigma}$ are the realized values, while the mixing parameter $\mu$, minus the exponent of the degree distribution $\tau_1$, and minus the exponent of the planted cluster size distribution $\tau_2$ are the input values. The planted cluster size distribution is controlled by $N_{\min}^{\sigma}$ and $N_{\max}^{\sigma}$, while $\tau_2$ is fixed. A comparative analysis for different values of $\tau_2$ would be difficult because a required graph size $N$ becomes extremely large for a slight change of $\tau_2$. The planted number of clusters $q_{\text{planted}}$ and the estimates using the modularity matrix $q_{\text{mod}}^*$ and the nonbacktracking matrix $q_{\text{NBT}}^*$ are indicated in the last three columns.

| Figure | $c$ | $d_{\max}$ | $\mu$ | $\tau_1$ | $\tau_2$ | $N_{\min}^{\sigma}$ | $N_{\max}^{\sigma}$ | $q_{\text{planted}}$ | $q_{\text{mod}}^*$ | $q_{\text{NBT}}^*$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 6 and 7 | 9.9 | 960 | 0.1 | 2 | 1 | 991 | 1064 | 11 | 8 | 15 |
| 8(a) | 6.5 | 100 | 0.3 | 2 | 1 | 633 | 946 | 13 | 12 | 12 |
| 8(b) | 6.5 | 100 | 0.5 | 2 | 1 | 633 | 946 | 13 | 8 | 13 |
| 8(c) | 6.5 | 100 | 0.8 | 2 | 1 | 633 | 946 | 13 | 1 | 1 |
| 8(d) | 6.5 | 100 | 0.3 | 2 | 1 | 110 | 796 | 27 | 21 | 27 |
| 8(e) | 6.5 | 100 | 0.5 | 2 | 1 | 110 | 796 | 27 | 12 | 25 |
| 8(f) | 6.6 | 98 | 0.8 | 2 | 1 | 119 | 965 | 23 | 1 | 1 |
| 9 | 7.4 | 991 | 0.3 | 1.8 | 1 | 122 | 887 | 26 | 2 | 27 |

retrieval phase, it is possible that BP is trapped in a local minimum of the Bethe free energy, while the correct initial state would converge to the global minimum.

As the final example using the LFR network, consider the case in which the assessment seems to fail because of the EM algorithm. Consider a network that has a broad degree distribution as in Figs. 6 and 7, in addition to a weak community structure and a broad planted cluster size distribution. As shown in Fig. 9, while the spectrum of the nonbacktracking matrix exhibits the estimate of $q^*$ near $q_{\text{planted}}$, such estimates cannot be obtained via the other criteria, because BP converges to the factorized state at $q = 12$, although the values of the criteria significantly vary when they reach this value. In this case, we can hardly conclude that there are no statistically significant structures beyond $q = 12$, and it is more natural to conclude that the BP converged to a local minimum of the Bethe free energy. Note that, even if we accept the estimate of $q^* = 27$, we cannot obtain a result with 27 clusters; recall that the input value of $q$ is the maximum number of clusters allowed, and the actual number of clusters that can be obtained is much less than 27. Readers might wonder what factor dominates the performance of the EM algorithm in the LFR network. Although the degree distribution seems to be an important factor, because there are many model parameters in the LFR network, it is difficult to precisely determine parameter dependencies experimentally. Note that a thorough investigation of the phase space of a particular model is not the goal of this paper. Instead, we investigate generic behaviors in community detection.

Let us next examine the performance of the assessment criteria on real-world networks. The basic information about each dataset is listed in Table II and the results of the assessment by each criterion are shown in Figs. 10(a)–10(d),
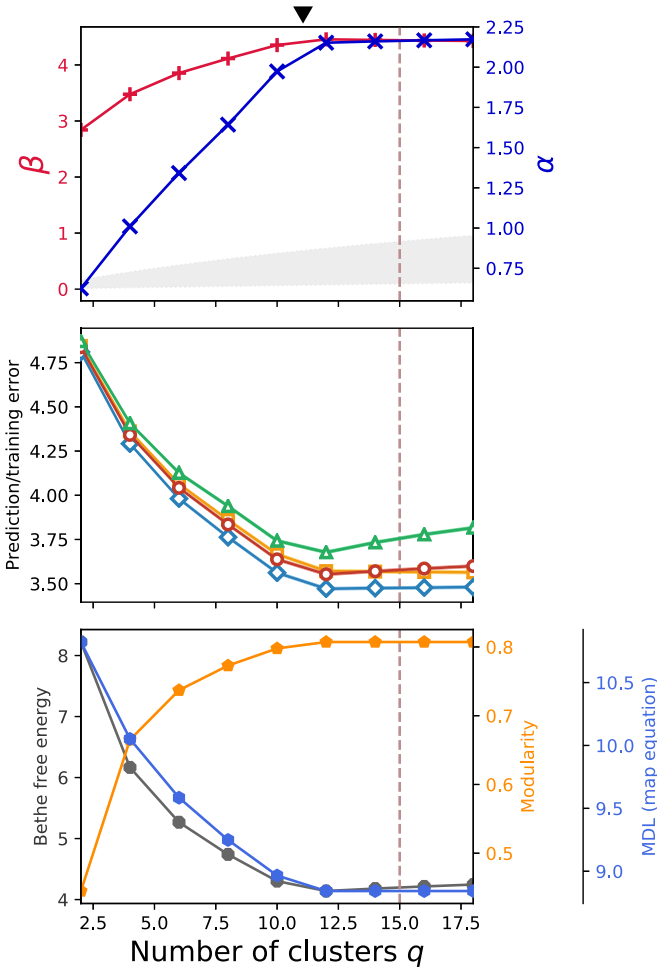
FIG. 6. Assessment of various criteria with respect to given inputs of $q$ for a LFR network. The top panel indicates the model parameters $\alpha$ (blue cross) and $\beta$ (red plus), where the shaded region indicates the region of $\beta$ between Eqs. (20) and (21). The middle panel indicates the Bayes prediction errors $E_{\text{Bayes}}$ (red circles), Gibbs prediction errors $E_{\text{Gibbs}}$ (green triangles), Gibbs training errors $E_{\text{training}}$ (blue diamonds), and MAP estimates $E_{\text{MAP}}$ of $E_{\text{Gibbs}}$ (yellow squares). The bottom panel indicates the modularity (yellow pentagon), map equation (blue hexagon), and Bethe free energy (gray octagon). In each panel, the number of clusters selected by the spectral method of the nonbacktracking matrix is indicated by a vertical dashed line. The planted number of clusters $q_{\text{planted}}$ is indicated with a filled triangle at the top of the figure.

11(e)–11(h), 12(i)–12(l), and 13(m)–13(p). For some networks, the inference algorithm does converge to the factorized state at some point as the input value of $q$ is increased; as far as we investigated, in many cases, the convergence to this trivial BP fixed point either supports a plausible value of $q^*$ or does not affect the assessment.

It is known that the Bethe free energy tends to largely overfit for real-world networks [13,15] when an affinity matrix of full degrees of freedom is used. However, with a restricted affinity matrix, the assessment by the Bethe free energy does not seem to be very wrong.

Unlike the cases of synthetic networks, the behaviors of the assessment criteria are often very different from each other. For example, modularity tends to support a relatively small value
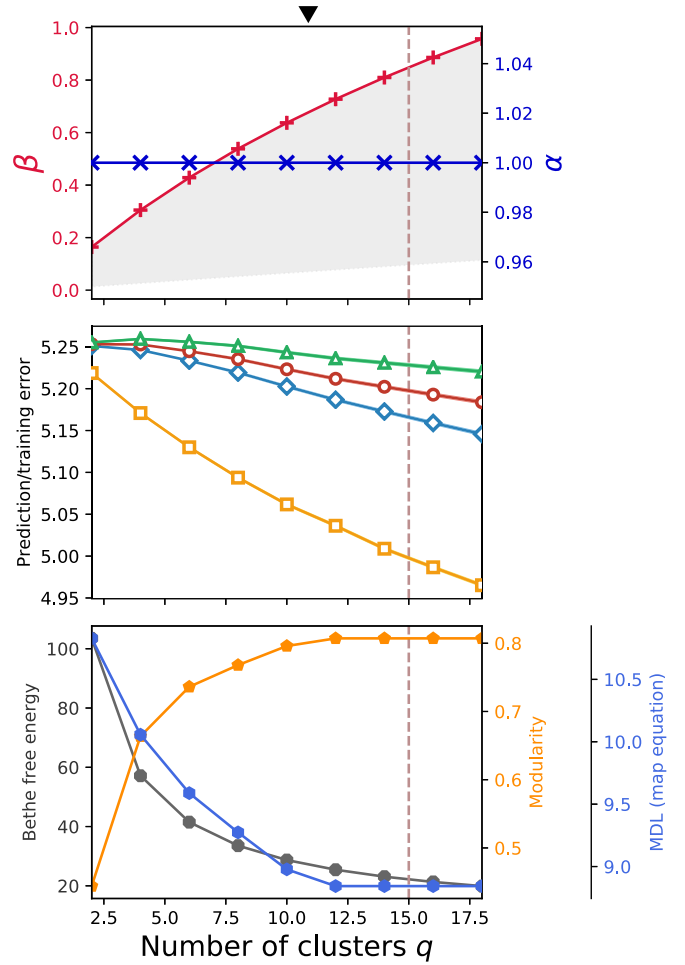
FIG. 7. Result of the same LFR network as Fig. 6. Here, statistical inference is performed without the model-parameter learning; the model parameters are fixed to $\alpha = 1$ and $\beta = \beta^*$.

for $q^*$, while the map equation tends to support a relatively large value. Assessment by the Bethe free energy and prediction errors can be close to either of them, and we cannot determine a similarity tendency. Note again that the inference algorithm does not optimize the minimum description length of the map equation; the partition is obtained such that the marginal likelihood is maximized and the minimum description length is measured only as a criterion for the assessment of the number of clusters. Another way to utilize the assessment by the minimum description length is to check whether the resolution limit [73,74] affects the result. The estimates for the number of clusters $q^*$ by modularity and by the map equation can differ considerably when many small clusters exist, because their resolution limits are very different [74]. When both modularity and the map equation support the same number of clusters $q^*$, it implies that the network is resolution limit-free [75].

### C. Assessment using the spectral methods

Finally, we examine the performance of the assessment of the number of clusters using the spectral methods that we explained in Secs. III C and IV D. The results of the estimates using the modularity matrix $q_{\text{mod}}^*$ and nonbacktracking matrix
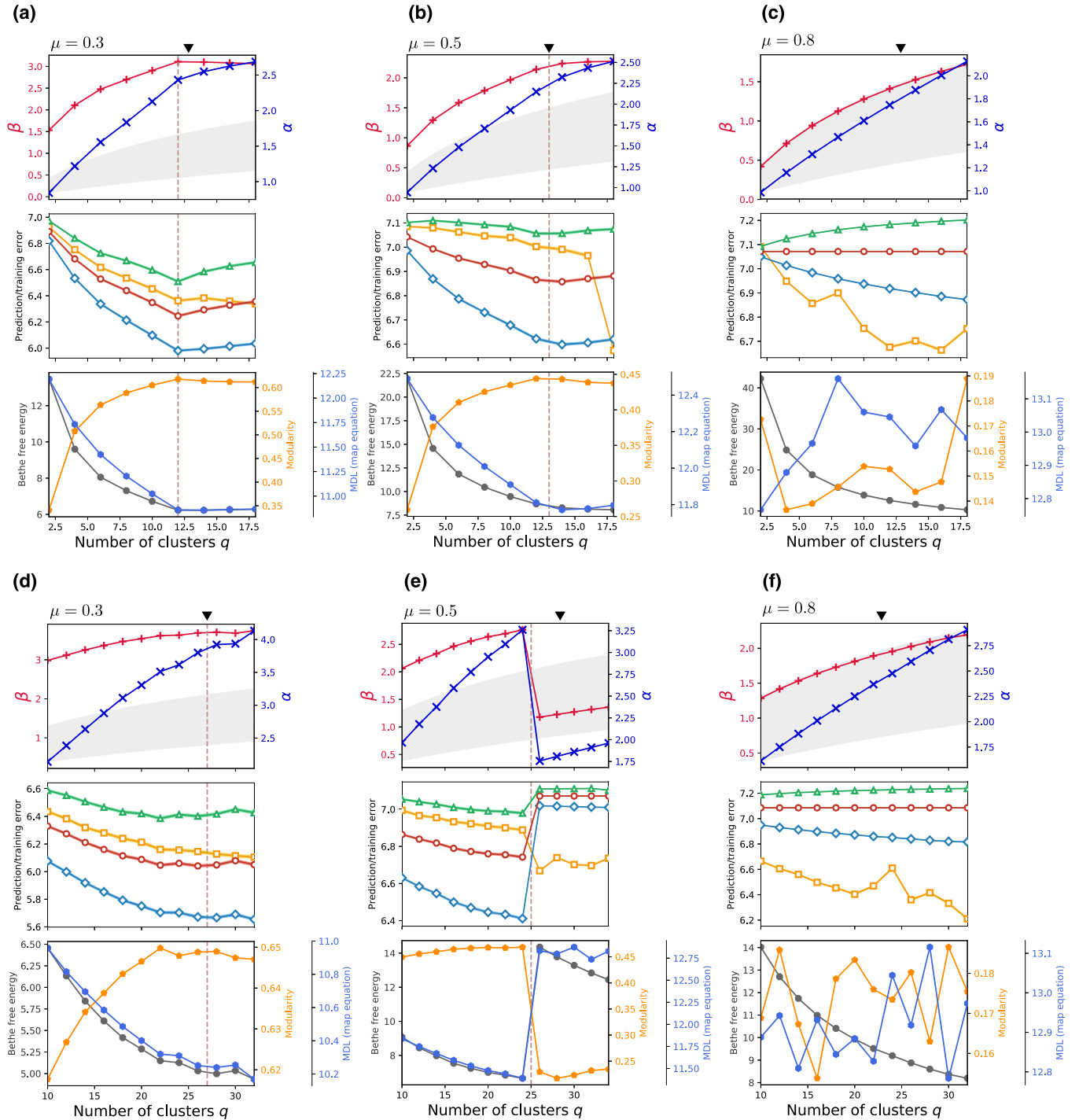
FIG. 8. Assessment of various criteria with respect to given inputs of $q$ for several LFR networks. They are plotted in the same manner as in Figs. 6 and 7.

$q_{NBT}^*$ are listed in Tables I and II. The estimates using the nonbacktracking matrix are also shown in Figs. 6–13 as dashed lines.

Note that, because the leading eigenvalues can be quickly computed for sparse networks, the assessment using the spectral method can be conducted most easily. Overall, the assessment using the modularity matrix tends to underfit, while the assessment using the nonbacktracking matrix tends to overfit, compared to the other criteria. Furthermore, for

real-world networks, it is often the case that the spectral band of the modularity matrix cannot be clearly observed. Therefore, in many cases, it is also difficult to visually assess the number of clusters from the spectral density.

The assessment using the nonbacktracking matrix is often very accurate in the sense that it coincides with the planted value $q_{planted}$ of an LFR network. The analysis with various values of $\mu$ was also analyzed in Ref. [76]. It is difficult to analyze what exactly causes overfitting in the cases of the real-
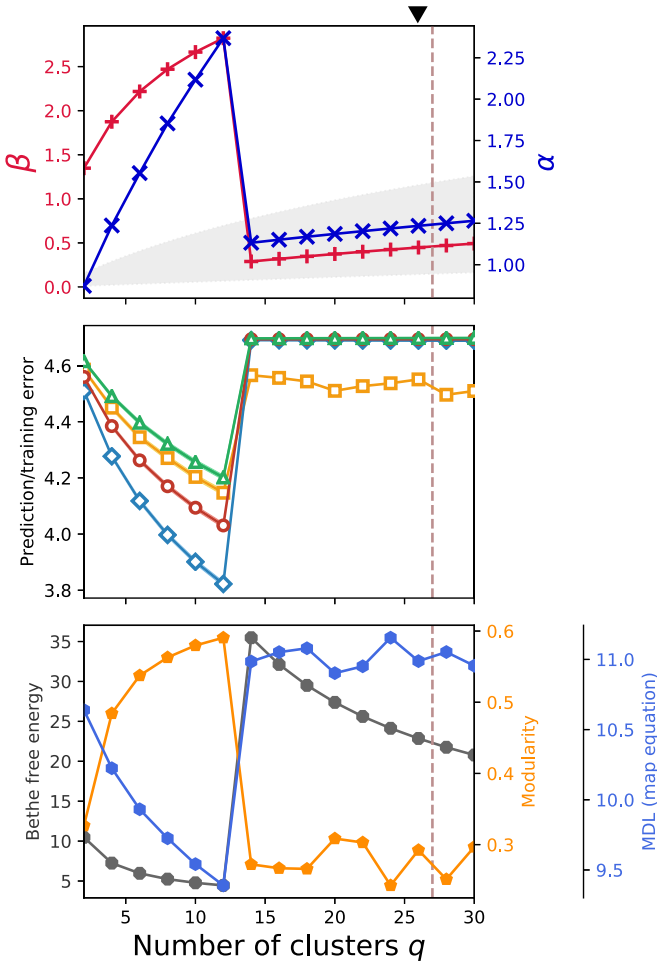
FIG. 9. Assessment of various criteria with respect to given inputs of $q$ for a LFR network. They are plotted in the same manner as in Figs. 6–8.

world networks; one of the possibilities is that the community structure may not be the only structure that contributes to the eigenvalues outside of the spectral band, and those unknown structures cause overfitting when we focus on community detection.

## VI. ASSESSMENT THROUGH VISUALIZATION

As we have observed, the validation curves of the criteria are often gradually saturated, particularly when the community structure is weak. In such a case, a criterion supports a range of values for $q^*$, instead of a single plausible value. Therefore, a finer inspection is required as a final step, if one wishes to determine the most plausible value of $q^*$.

Visualizing how a network is partitioned for each input value of $q$ can be helpful for the assessment of the number of clusters. The alluvial diagram [22] is a suitable tool for this purpose. It was originally introduced as a diagram to indicate the time evolution of a community structure. Here, we visualize the change in the partition for different values of $q$, rather than the change in the partition over time. In the alluvial diagram, the results of community detection are aligned horizontally. For each partition, the set of vertices in the

TABLE II. Estimated number of clusters of real-world networks using the spectra of the modularity matrix $q^*_{\mathrm{mod}}$ and nonbacktracking matrix $q^*_{\mathrm{NBT}}$. Multiedges, self-loops, and the direction and weights of edges are neglected in all networks.

| Dataset | $N = |V|$ | $L = |E|$ | $q^*_{\mathrm{mod}}$ | $q^*_{\mathrm{NBT}}$ |
|---|---|---|---|---|
| Karate club [55] | 34 | 78 | 1 | 2 |
| Dolphins [56] | 62 | 159 | 2 | 2 |
| Les Miserables [57] | 77 | 254 | 2 | 4 |
| Football [58] | 115 | 613 | 10 | 10 |
| Political books [19] | 105 | 441 | 2 | 3 |
| Network science [8] | 379 | 914 | 4 | 21 |
| C. elegans [59] | 453 | 2 025 | 1 | 5 |
| Political blogs [60] | 1 222 | 16 714 | 3 | 8 |
| Protein [61–63] | 2 738 | 6 007 | 1 | 14 |
| Power grid [64] | 4 941 | 6 594 | 10 | 25 |
| Facebook ego [65] | 4 039 | 88 234 | 15 | 55 |
| Chess [63,66,67] | 7 115 | 55 779 | 22 | 45 |
| U.S. airports [68] | 7 976 | 15 677 | 7 | 17 |
| ca-HepTh [69] | 8 638 | 24 806 | 33 | 82 |
| Enron [70,71] | 33 696 | 180 811 | 9 | 93 |
| Epinions [72] | 75 888 | 405 739 | 6 | 202 |

same cluster is expressed as a vertical bundle. Then, the same vertices in different partitions are smoothly connected. The alluvial diagram can be generated at Ref. [77] using .smap files.

The alluvial diagram also uses color tone to express the significance of the cluster assignment; the vertices with insignificant assignments are expressed by faint colors. We assess that the cluster assignment of vertex $i$ is not significant if max $\psi^i_\sigma$ is less than a threshold. Here, we regard max $\psi^i_\sigma > 0.9$ as a significant assignment.

The alluvial diagrams of four real-world networks are shown in Fig. 14. As we can see from the *political books* and *political blogs* networks, the actual partition may be kept the same as we increase the input value of $q$. We can confirm that the partition with $q = 3$ stably exists in the political books network and the partition with $q = 4$ is also consistent with the partitions with fewer clusters; i.e., it is a refinement of the partition with $q = 3$, and the highlighted clusters in the middle belong to different clusters in the partition with $q = 2$. For the political blogs network, although modularity and the map equation support $q = 3$ or 4, in the end, we can confirm that the dominant structure does not change from the partition with $q = 2$.

In the case of the *protein* network, for any choice of $q$, only a fraction of vertices belong to a specific cluster significantly. In other words, the network does not have a global community structure. Whereas the vertices with insignificant assignments exhibit a randomlike behavior, the vertices with significant assignments roughly exhibit a hierarchical structure. According to the Gibbs prediction error $E_{\mathrm{Gibbs}}$ and its MAP estimate $E_{\mathrm{MAP}}$ in Fig. 12(i), the partitions with $q = 3$ or 4 are supported. Although we cannot clearly see a qualitative difference in the alluvial diagram, if we look carefully, from the partition with $q \geqslant 5$, we can observe that a small fraction of vertices with significant assignments start to exhibit a nonhierarchical structure.
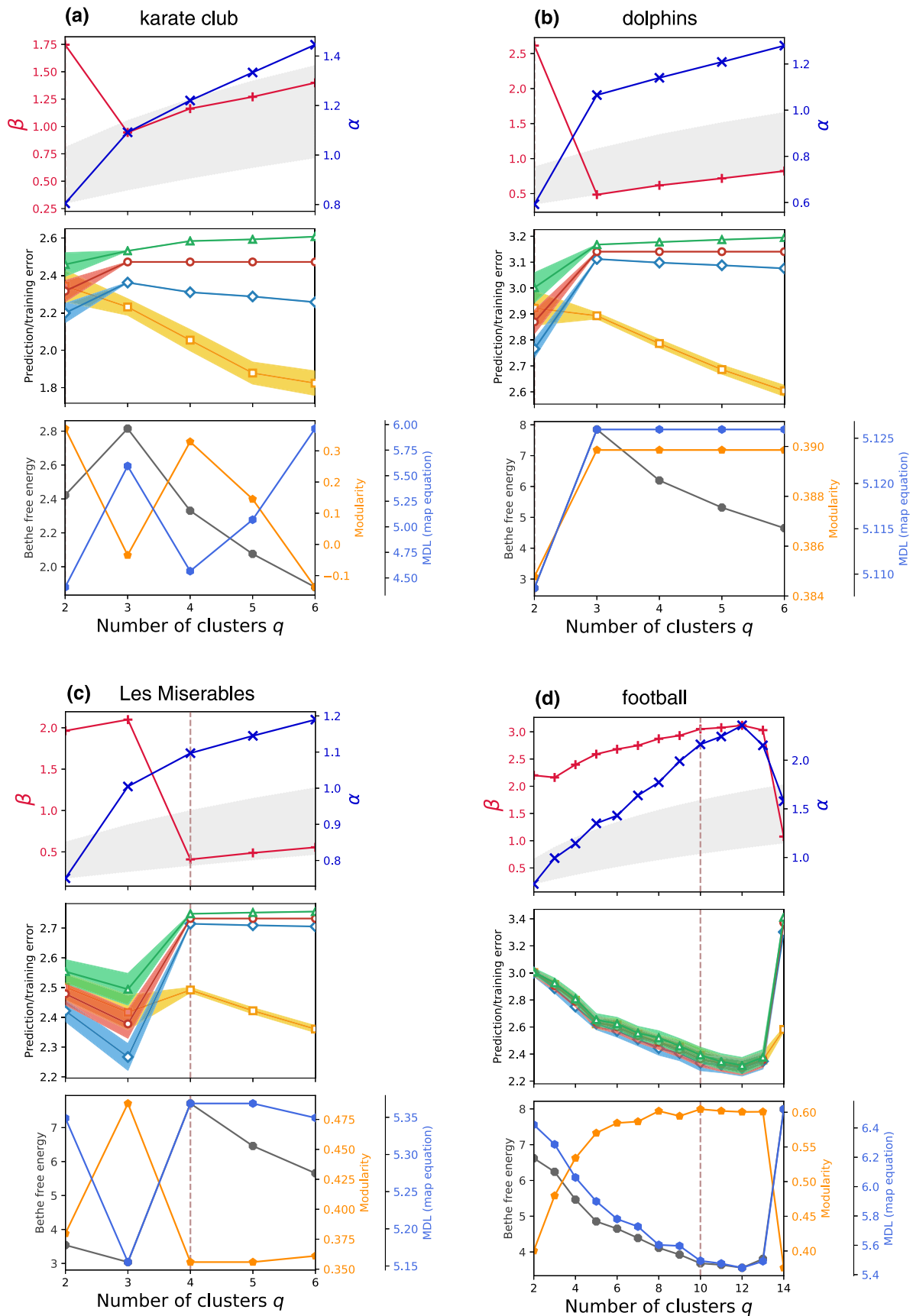
FIG. 10. Assessment of various criteria with respect to given inputs of $q$ for several real-world networks. They are plotted in the same manner as in Figs. 6–9. The shaded parts in the cross-validation error plot indicate the standard errors.
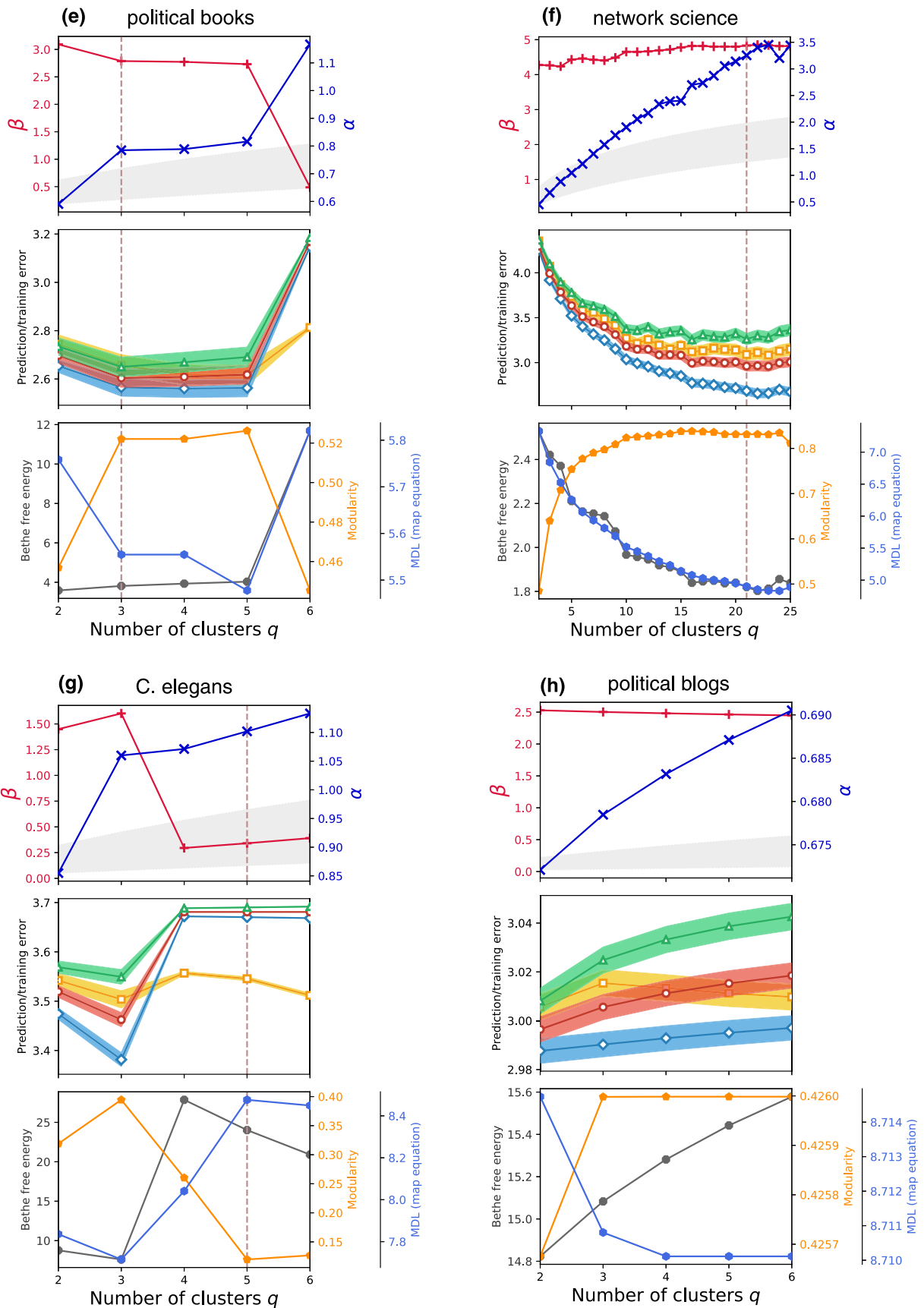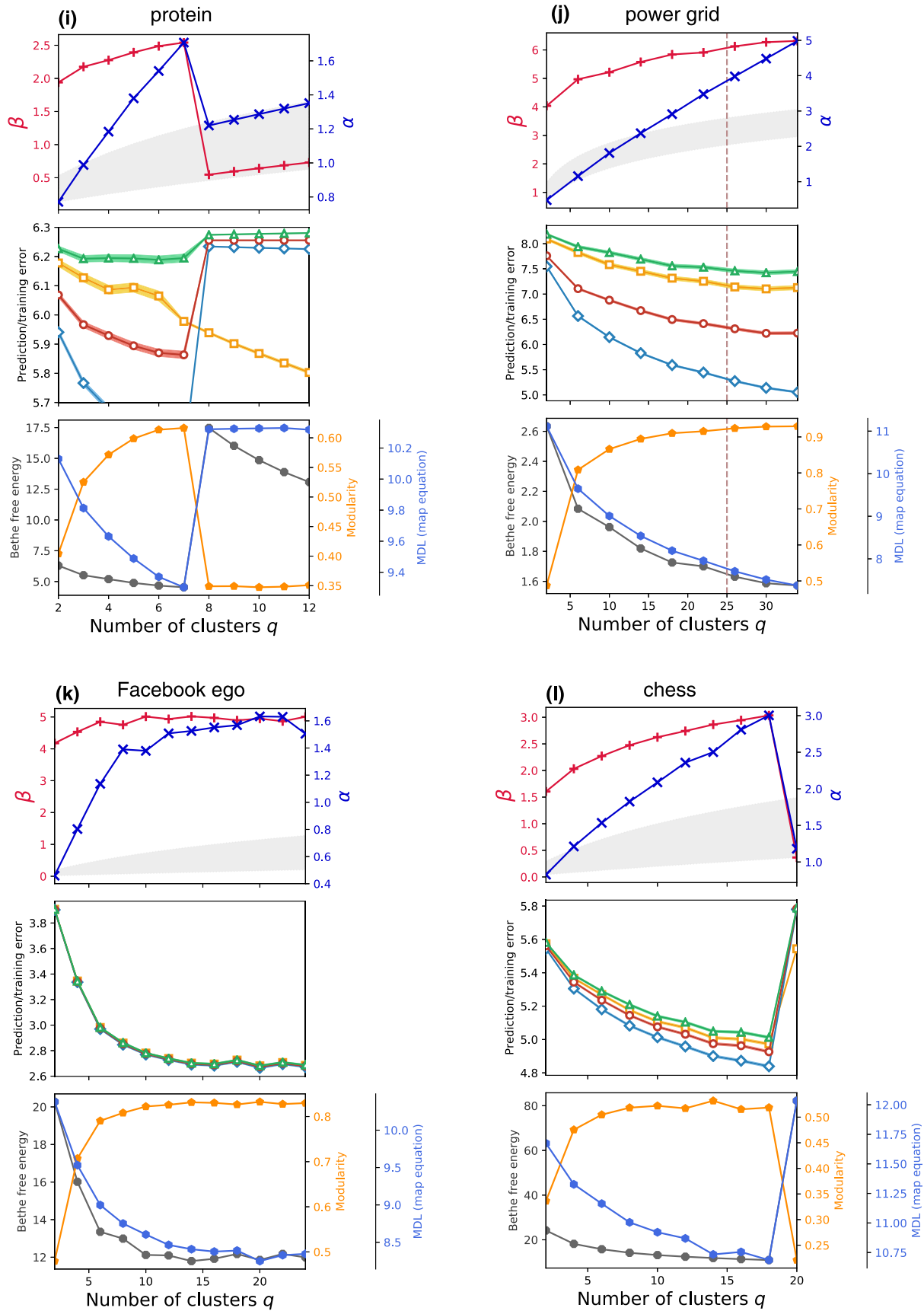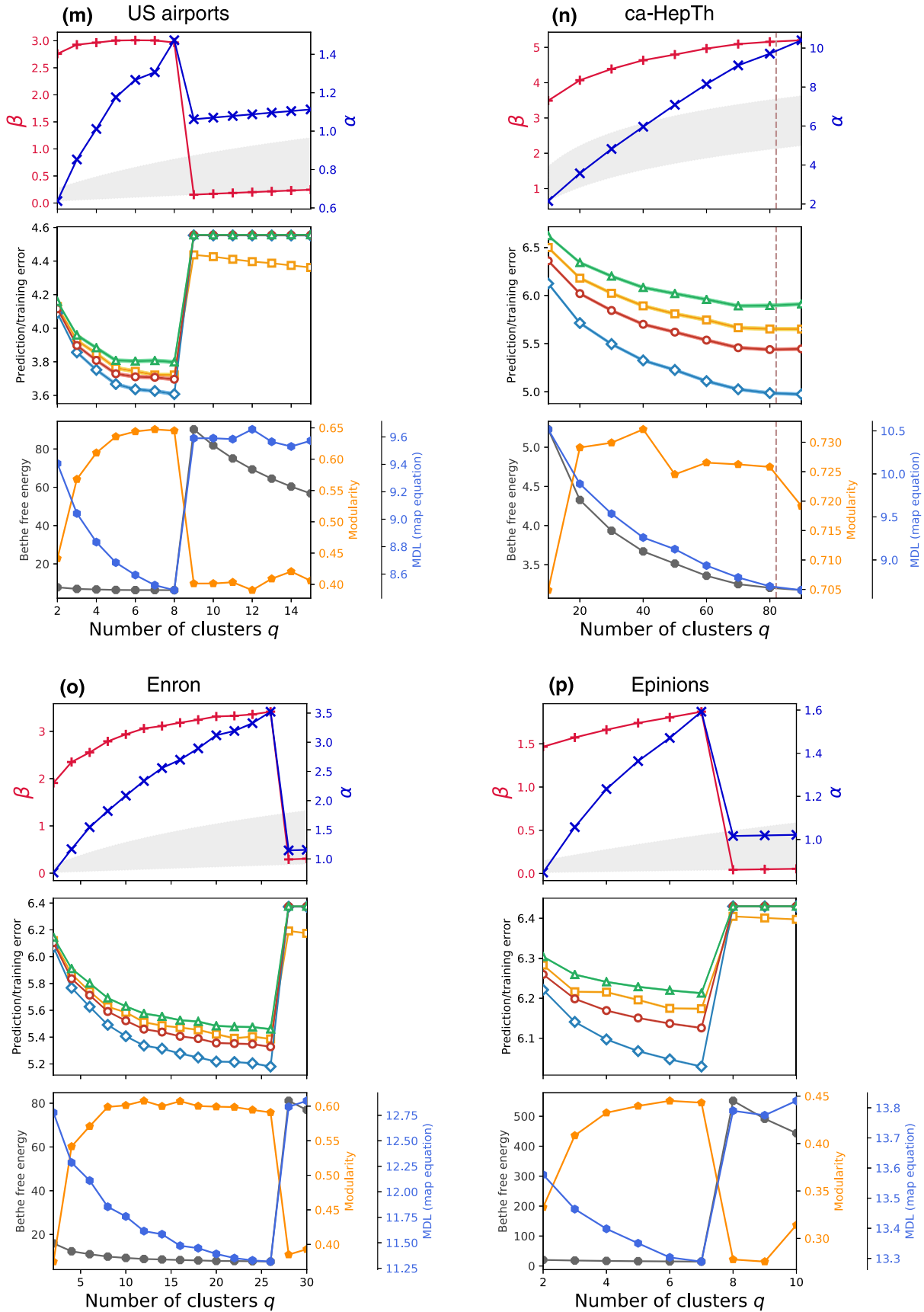
FIG. 11. Assessment of various criteria with respect to given inputs of $q$ for several real-world networks. They are plotted in the same manner as in Figs. 6–9. The shaded parts in the cross-validation error plot indicate the standard errors.

FIG. 12. Assessment of various criteria with respect to given inputs of $q$ for several real-world networks. They are plotted in the same manner as in Figs. 6–9. The shaded parts in the cross-validation error plot indicate the standard errors.

FIG. 13. Assessment of various criteria with respect to given inputs of $q$ for several real-world networks. They are plotted in the same manner as in Figs. 6–9. The shaded parts in the cross-validation error plot indicate the standard errors.
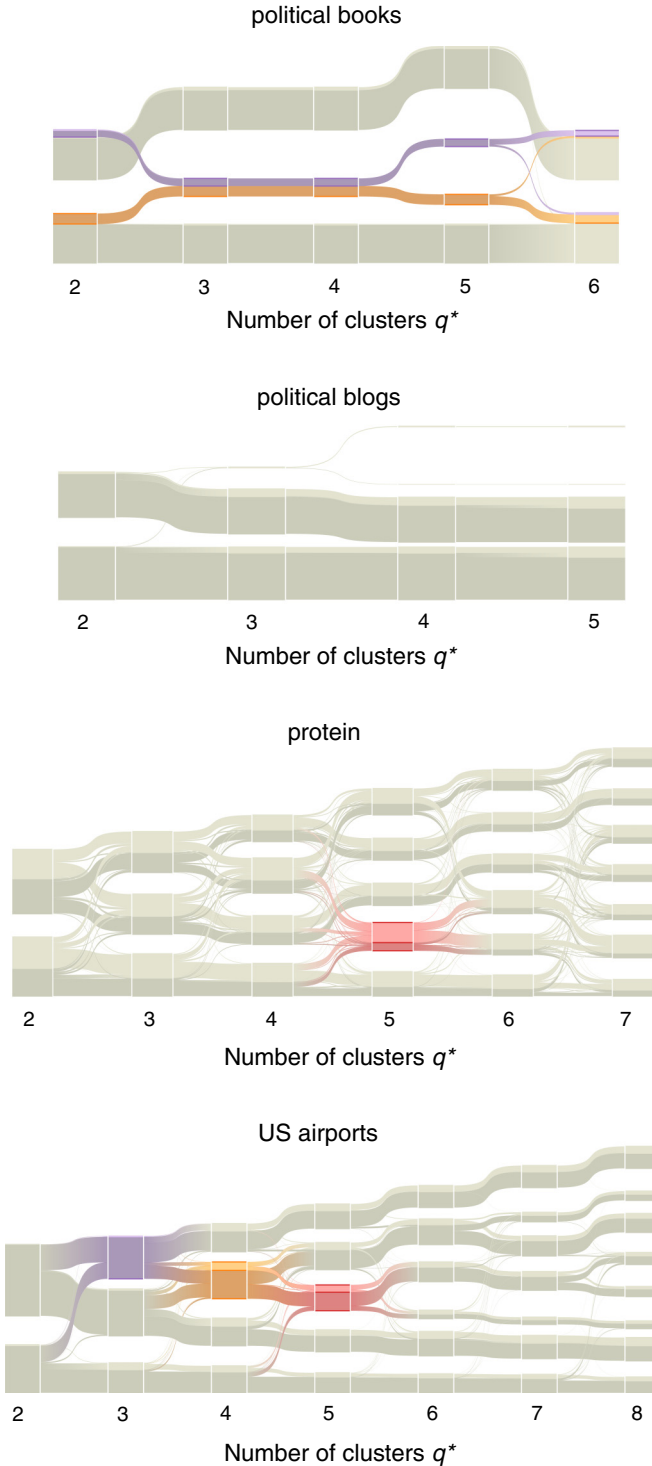
FIG. 14. Alluvial diagrams of the political books, political blogs, protein, and U.S. airport networks. Some clusters are highlighted.

The way the *U.S. airports* network is partitioned is also different from other networks. When we focus on the vertices with significant assignments, the resulting partitions do not constitute a hierarchical structure for small values of $q$, while they roughly do for large values of $q$. As various assessment criteria support the range of $5 \leqslant q^* \leqslant 8$, it seems to be plausible to select $q^*$ in this range.

## VII. SUMMARY AND DISCUSSION

We conducted a comparative analysis on the assessment of the number of clusters in community detection. Although we examined the performance of various algorithms and assessment criteria, an exhaustive analysis requires all possible combinations of the frameworks, algorithms, and assessment criteria. For example, an important missing part is the Monte Carlo method [16–18]. The Monte Carlo method usually incorporates the prior probability distribution with respect to the affinity matrix $\omega$; it plays the role of a penalty in the assessment of the number of clusters. Therefore, a comparative analysis including the Monte Carlo method would not only be a comparison of different algorithms but also a comparison of different frameworks. In a broader sense, we should note that community detection also possesses some other fundamental issues as discussed in Ref. [78].

We confirmed that the assessment using the EM algorithm with BP and the corresponding prediction errors also provide plausible estimates in various synthetic and real-world networks, while the greedy algorithms tend to largely overfit. Note that it is not trivial that the BP algorithm performs reasonably for real-world networks, because the emergence of the so-called hard phase [13] may deteriorate the performance when the planted number of clusters is large. Furthermore, the EM algorithm may be trapped in a local minimum depending on the choice of the initial condition of the model parameters.

We also observed that the estimate of $q^*$ using the modularity matrix tends to underfit, while the estimate using the nonbacktracking matrix tends to overfit. To the best of our knowledge, the assessment using the boundary of the spectral band of the modularity matrix has not been investigated in the literature.

Finally, we proposed the utilization of the alluvial diagram for the assessment of $q^*$. Although there is the obvious issue that it is not applicable to the networks with a very large $q^*$, when it is not the case, the alluvial diagram is very useful, particularly when the network does not clearly have a global community structure.

For the LFR networks and real-world networks, we do not know the number of clusters that are statistically significant. It may not coincide with the planted number of clusters or the number of clusters in the metadata. Therefore, we rely on the consistency among various criteria and algorithms for the plausibility of assessment. The tendency of overfit and underfit that we investigated in this paper represents the relative performance among the criteria and algorithms. Although there is no ground truth in a real-world network, estimating the number of clusters is a practical problem. At the end of the day, a practitioner selects a certain value (or a set of values) as $q^*$.

The code for the assessment using the modularity matrix is available at Ref. [32]. The code for the assessment using the other criteria in this paper, which can also generate .smap files, is available at Ref. [79].

[1] P. W. Holland, K. B. Laskey, and S. Leinhardt, Soc. Networks **5**, 109 (1983).

[2] Y. J. Wang and G. Y. Wong, J. Am. Stat. Assoc. **82**, 8 (1987).

[3] B. Karrer and M. E. J. Newman, Phys. Rev. E **83**, 016107 (2011).

[4] S. Fortunato, Phys. Rep. **486**, 75 (2010).

[5] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, J. Stat. Mech.: Theor. Exp. (2008) P10008.

[6] M. Rosvall and C. Bergstrom, Proc. Natl. Acad. Sci. USA **105**, 1118 (2008).

[7] U. Luxburg, Stat. Comput. **17**, 395 (2007).

[8] M. E. J. Newman, Phys. Rev. E **74**, 036104 (2006).

[9] F. Krzakala, C. Moore, E. Mossel, J. Neeman, A. Sly, L. Zdeborová, and P. Zhang, Proc. Natl. Acad. Sci. USA **110**, 20935 (2013).

[10] A. Saade, F. Krzakala, and L. Zdeborová, in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1* (MIT, Cambridge, MA, 2014), NIPS'14, pp. 406–414.

[11] J. J. Daudin, F. Picard, and S. Robin, Stat. Comput. **18**, 173 (2008).

[12] P. Latouche, E. Birmelé, and C. Ambroise, Statistical Modelling **12**, 93 (2012).

[13] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová, Phys. Rev. E **84**, 066106 (2011).

[14] K. Hayashi, T. Konishi, and T. Kawamoto, arXiv:1602.02256 (2016).

[15] T. Kawamoto and Y. Kabashima, Sci. Rep. **7**, 3327 (2017).

[16] K. Nowicki and T. A. B. Snijders, J. Am. Stat. Assoc. **96**, 1077 (2001).

[17] T. P. Peixoto, Phys. Rev. E **89**, 012804 (2014).

[18] M. E. J. Newman and G. Reinert, Phys. Rev. Lett. **117**, 078301 (2016).

[19] M. E. J. Newman, Proc. Natl. Acad. Sci. USA **103**, 8577 (2006).

[20] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová, Phys. Rev. Lett. **107**, 065701 (2011).

[21] M. Rosvall and C. T. Bergstrom, PloS One **6**, e18209 (2011).

[22] M. Rosvall and C. T. Bergstrom, PLoS One **5**, 1 (2010).

[23] P. Zhang and C. Moore, Proc. Natl. Acad. Sci. USA **111**, 18144 (2014).

[24] J. Reichardt and S. Bornholdt, Phys. Rev. E **74**, 016110 (2006).

[25] H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing: An Introduction* (Oxford University, New York, 2001).

[26] M. E. J. Newman, Phys. Rev. E **88**, 042822 (2013).

[27] C. Schülke and F. Ricci-Tersenghi, Phys. Rev. E **92**, 042804 (2015).

[28] A formal derivation of the EM algorithm is the one using the variational expression, e.g., Ref. [15].

[29] X. Zhang, T. Martin, and M. E. J. Newman, Phys. Rev. E **91**, 032803 (2015).

[30] M. Mézard and A. Montanari, *Information, Physics, and Computation* (Oxford University, New York, 2009).

[31] B. H. Good, Y.-A. de Montjoye, and A. Clauset, Phys. Rev. E **81**, 046106 (2010).

[32] https://github.com/tatsuro-kawamoto/modularity_eigenvalues.

[33] J. Shi and J. Malik, IEEE Trans. Pattern Anal. Mach. Intell. **22**, 888 (2000).

[34] T. Kawamoto and Y. Kabashima, Eur. Phys. Lett. **112**, 40007 (2015).

[35] M. E. J. Newman and M. Girvan, Phys. Rev. E **69**, 026113 (2004).

[36] M. L. Mehta, *Random Matrices*, 3rd ed. (Elsevier, Amsterdam, 2004).

[37] R. R. Nadakuditi and M. E. J. Newman, Phys. Rev. Lett. **108**, 188701 (2012).

[38] R. R. Nadakuditi and M. E. J. Newman, Phys. Rev. E **87**, 012803 (2013).

[39] X. Zhang, R. R. Nadakuditi, and M. E. J. Newman, Phys. Rev. E **89**, 042816 (2014).

[40] T. Kawamoto and Y. Kabashima, Phys. Rev. E **91**, 062803 (2015).

[41] A. Saade, F. Krzakala, and L. Zdeborov, Europhys. Lett. **107**, 50005 (2014).

[42] T. Kawamoto, J. Stat. Mech.: Theor. Exp. (2016) 023404.

[43] R. Pastor-Satorras and C. Castellano, Sci. Rep. **6**, 18847 (2016).

[44] https://github.com/vtraag/louvain-igraph.

[45] http://igraph.org/python/doc/igraph.Graph-class.html#community_infomap.

[46] E. Mossel, J. Neeman, and A. Sly, Probab. Theory Relat. Fields **162**, 431 (2014).

[47] L. Massoulié, Community Detection Thresholds and the Weak Ramanujan Property, in *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC'14)* (ACM, New York, NY, 2014), pp. 694–703.

[48] A. Clauset, M. E. J. Newman, and C. Moore, Phys. Rev. E **70**, 066111 (2004).

[49] http://igraph.org/python/doc/igraph.Graph-class.html#community_fastgreedy.

[50] A. Lancichinetti and S. Fortunato, Phys. Rev. E **80**, 056117 (2009).

[51] C. Moore, arXiv:1702.00467 (2017).

[52] Precisely speaking, the algorithmic detectability threshold that a certain algorithm fails to retrieve the planted structure is different from the information-theoretic detectability threshold that it is fundamentally impossible to retrieve the planted structure.

[53] T. P. Peixoto, arXiv:1705.10225 (2017).

[54] Of course, this can also be confirmed directly by observing the marginal distributions themselves.

[55] W. W. Zachary, Journal of Anthropological Research **33**, 452 (1977).

[56] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, Behav. Ecol. Sociobiol. **54**, 396 (2003).

[57] D. E. Knuth, *The Stanford GraphBase: A Platform for Combinatorial Computing* (Addison-Wesley, Reading, MA, 1993), Vol. 37.

[58] M. Girvan and M. E. J. Newman, Proc. Natl. Acad. Sci. USA **99**, 7821 (2002).

[59] J. Duch and A. Arenas, Phys. Rev. E **72**, 027104 (2005).

[60] L. A. Adamic and N. Glance, The Political Blogosphere and the 2004 U.S. Election: Divided They Blog, in *Proceedings of the Third International Workshop on Link Discovery* (ACM, New York, NY, 2005), pp. 36–43.

[61] J.-F. Rual, K. Venkatesan, T. Hao, T. Hirozane-Kishikawa, A. Dricot, N. Li, G. F. Berriz, F. D. Gibbons, M. Dreze, and N. Ayivi-Guedehoussou, Nature **437**, 1173 (2005).

[62] Human protein (vidal) network dataset—KONECT (2016), http://konect.uni-koblenz.de/networks/maayan-vidal.

[63] J. Kunegis, in *Proceedings of the 22nd International Conference on World Wide Web*, WWW '13 Companion (ACM, New York, 2013), pp. 1343–1350, http://userpages.uni-koblenz.de/kunegis/paper/kunegis-koblenz-network-collection.pdf.

[64] D. J. Watts and S. H. Strogatz, Nature (London) **393**, 440 (1998).

[65] J. Leskovec and J. J. Mcauley, in *Advances in Neural Information Processing Systems 25*, edited by F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Curran Associates, Inc., Nevada, 2012), pp. 539–547.

[66] Kaggle, Chess ratings—Elo versus the rest of the world, https://www.kaggle.com/c/chess/data/ (2010).

[67] Chess network dataset—KONECT (2016), http://konect.uni-koblenz.de/networks/chess.

[68] T. Opsahl, F. Agneessens, and J. Skvoretz, Soc. Networks **32**, 245 (2010).

[69] J. Leskovec, J. Kleinberg, and C. Faloutsos, ACM Trans. Knowl. Discov. Data **1** (2007).

[70] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, Internet Math. **6**, 29 (2009).

[71] B. Klimt and Y. Yang, Introducing the Enron Corpus, in Conference on Email and Anti-Spam, Mountain View, CA, USA (2004), https://www.researchgate.net/publication/220271805_Introducing_the_Enron_Corpus.

[72] M. Richardson, R. Agrawal, and P. Domingos, *Trust Management for the Semantic Web* (Springer, Berlin, 2003), pp. 351–368.

[73] S. Fortunato and M. Barthélemy, Proc. Natl. Acad. Sci. USA **104**, 36 (2007).

[74] T. Kawamoto and M. Rosvall, Phys. Rev. E **91**, 012809 (2015).

[75] Although the word "resolution limit-free" is sometimes used for an algorithm, it seems more appropriate to say that a network is resolution limit-free for a given set of algorithms.

[76] R. K. Darst, Z. Nussinov, and S. Fortunato, Phys. Rev. E **89**, 032809 (2014).

[77] http://www.mapequation.org/.

[78] L. Peel, D. B. Larremore, and A. Clauset, Sci. Adv. **3**, e1602548 (2017).

[79] https://github.com/tatsuro-kawamoto/graphBIX.