Check for updates

J. Appl. Probab. **63,** 484–500 (2026)
*doi:*10.1017/jpr.2025.10038

LOCALIZED GEOMETRY DETECTION IN SCALE-FREE RANDOM GRAPHS

GIANMARCO BET,* *Università degli Studi di Firenze*
RICCARDO MICHIELAN,** *Gran Sasso Science Institute*
CLARA STEGEHUIS,*** *University of Twente*

Abstract

We consider the problem of detecting whether a power-law inhomogeneous random graph contains a geometric community, and we frame this as a hypothesis-testing problem. More precisely, we assume that we are given a sample from an unknown distribution on the space of graphs on n vertices. Under the null hypothesis, the sample originates from the inhomogeneous random graph with a heavy-tailed degree sequence. Under the alternative hypothesis, $k = o(n)$ vertices are given spatial locations and connect following the *geometric* inhomogeneous random graph connection rule. The remaining $n - k$ vertices follow the inhomogeneous random graph connection rule. We propose a simple and efficient test based on counting normalized triangles to differentiate between the two hypotheses. We prove that our test correctly detects the presence of the community with high probability as $n \to \infty$, and identifies large-degree vertices of the community with high probability.

Keywords: Community detection; network geometry; scale-free graphs; weighted triangles

2020 Mathematics Subject Classification: Primary 60F05; 62G10; 62G30
Secondary 05C80

1. Introduction

Random graphs provide a unified framework to model many complex systems in biology, computer science, and sociology, as well as numerous other sciences. Random graphs are particularly useful as *null models* to determine if some observed real-world network deviates from its expected structure in a statistically significant way. In this context, it has been widely observed that real-world networks share two defining features: heavy-tailed degree sequences and large clustering [16, 34]. Neither of these features are reproduced by the classical Erdös–Rényi random graph model, which makes this an unsatisfactory null model for most applications. Consequently, alternative models have been developed to match the degree sequence and clustering observed in real-world networks. The so-called *inhomogeneous random graph* (IRG) [14] is a popular generalization of the Erdös–Rényi random graph obtained

Received 18 July 2025; accepted 18 July 2025.
* Postal address: Dipartimento di Matematica e Informatica 'Ulisse Dini', Università degli Studi di Firenze, Italy. Email: gianmarco.bet@unifi.it
** Postal address: Gran Sasso Science Institute, L'Aquila, Italy. Email: riccardo.michielan@gssi.it
*** Postal address: Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, Netherlands. Email: s.stegehuis@utwente.nl

© The Author(s), 2025. Published by Cambridge University Press on behalf of Applied Probability Trust. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution, and reproduction in any medium, provided the original work is properly cited.

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

by assigning weights to nodes, and connecting two nodes with a probability that is proportional to the products of their weights. This way, the IRG can reproduce an arbitrary degree sequence, but still has low clustering.

A popular method to obtain a random graph with a large clustering is to embed the vertices in a metric space (such as the sphere or the torus) and connect them with probabilities proportional to their distances [26]. Indeed, the presence of distances makes two neighbors of a given vertex likely to be close by and therefore connected as well, due to the triangle inequality. By embedding the vertices of the IRG in a torus, we obtain the so-called *geometric inhomogeneous random graph* (GIRG) [12]. This model creates realistic networks with heavy-tailed degree sequences, as well as high clustering, and has been very successful in embedding real-world network data into a geometric space [9, 18]. Furthermore, this random graph model is a very natural model when the vertices have features, such as locations, interests, or other properties, and vertices with similar features are more likely to connect.

However, these types of random graph model assume that all nodes are spread uniformly across the geometric space, while real-world network embeddings often show a more clustered geometric space instead [8, 9], causing a community structure. It is then of great practical interest, first, to establish if these communities are present, and, second, to identify them. Perhaps surprisingly, the early literature on the latter problem did not address the former [20, 32, 33]. In fact, often the focus of the community detection literature lies on algorithms to extract a community structure from given networks, regardless of whether the structure is actually present. These algorithms are usually tested on random graph models with a known community structure. One such example is the stochastic block model [22], which has received considerable attention due to its mathematical tractability. However, this comes at the expense of unrealistic assumptions, such as very large communities (of the same order as the graph size) and homogeneous degree distributions. To overcome this, Arias-Castro and Verzelen [4, 38] considered the problem of detecting the presence of a small community in an Erdös–Rényi random graph. They found the region in the parameter space where (almost sure) detection is impossible, and gave tests that are able to detect the community outside this region. These results were later generalized to the IRG in [7]. However, none of these results address the realistic graph case with heavy-tailed degree sequences.

In this paper, we take a further step toward obtaining detection results for realistic networks by detecting communities in *realistic random graph models with both heterogeneity and geometry*. More precisely, our null model is the IRG with a heavy-tailed degree sequence, and the community, if present, is obtained by embedding a small number of the total nodes within a torus and connecting them according to the usual GIRG connection probabilities. Due to the geometric nature of the community it contains many triangles, as opposed to the tree-like nature of an IRG-based community. This realistic feature of the community allows us to develop efficient testing methods for the testing as well as the identification of the community.

More precisely, when the community is indeed present in a graph of size n, $n - k$ nodes of the network form connections with each other on a non-geometric basis. The other k nodes have a position in some geometric space, and nearby nodes are more likely to connect. This geometric setting creates a subgraph with many triangles, and can therefore be thought of as a community in the network. This geometric structure is less restrictive than planting a clique, and more realistic than a dense inhomogeneous random graph as a community. Furthermore, the fact that the planted structure is geometric allows for efficient, triangle-based tests to detect and identify the structure. To the best of our knowledge, this type of planted structure has not been considered before.

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

Our contributions are the following:

- We provide a statistical test to detect the presence of a planted geometric community in *realistic random graph models*. Unlike other detection tests for dense subgraph detection [7, 38], the test works for heterogeneous degrees and sparse networks. This method is triangle-based, making it efficient in implementation. Rather than using standard triangle counts, which may not be able to differentiate between geometric and non-geometric networks, the statistic weights the triangles based on their evidence for geometry.
- We provide a statistical method to identify the largest-degree vertices of the planted geometric community. This method is also triangle-based, and is therefore efficient to implement. We show that this method achieves almost exact recovery among all high-degree vertices.
- We provide a method to infer the size of the planted geometric community. This method uses the largest-degree identified vertices of the planted geometric community to obtain an estimate for the community size based on the convergence of order statistics.
- We show numerically that the combination of these tests leads to accurate identification of the planted geometric community. Furthermore, these tests can be performed computationally in only $O(n^{3/2})$ time [27].

1.1. Related literature

Our work lies at the intersection of two rich lines of research: community detection and what might be referred to as *structure detection*. In the former setting, one is given a sample from a known random graph model and the task is to determine if there is a statistically unlikely dense subgraph, and possibly to identify it. In this context, the *planted clique problem* has received considerable attention as a testbed for community detection algorithms. In this model, a large network of size n is generated according to some mechanism, and a small clique of size k might be planted in it [2, 39]. The seminal works [4, 38] form a stepping stone toward more realistic dense communities. Their null model is the (respectively dense, sparse) Erdös–Rényi random graph, and, when present, the community is a small subset of vertices with larger connection probability than in the null model. See also [21]. Further generalizing this work, [7] focuses on detecting a dense subgraph in an inhomogeneous random graph. More precisely, their null model is the inhomogeneous random graph, and in the alternative hypothesis, the connection probabilities of a small subset of nodes C are increased by a multiplicative factor $\rho_C > 1$. Crucially, their approach requires precise control of the inhomogeneity of the graph and does not work, for example, for heavy-tailed degree distributions. Therefore, the difference between our work and [7] is two-fold. First, we consider the case of power-law vertex weights, which is more attractive from a modeling point of view. Second, the planted structure is a community by virtue of the underlying geometrical structure, rather than by tuning an additional model parameter of a tree-like graph. [6] tackles the opposite problem to ours, namely, detecting mean-field effects in a geometric random graph model. More precisely, their null model is a geometric random graph, and in the alternative hypothesis, a small subset of vertices connects with every other vertex according to independent and identically distributed Bernoulli random variables. They provide detection thresholds, as well as asymptotically powerful tests.

On the other hand, in the setting of structure detection, we are given a sample from an unknown random graph model, and the task is to determine if the sample originates from a mean-field model or a structured (e.g. geometric) model. In [13] (see also [15]), the null model

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

is the Erdös–Rényi graph, and the alternative model is a high-dimensional geometric random graph. For recent progress on this problem, see [10]. [17] proposes a test based on small subgraph counts to distinguish between the Erdös–Rényi graph and a general class of structured models that includes the stochastic block model and the configuration model. More recently, [11] proposes a test to distinguish between a mean-field model and Gibbs models, and [29] proposes a test to distinguish between a power-law random graph with and without geometry. See also [19] for two-sample hypothesis testing for inhomogeneous random graphs. [23] proposes the so-called SCORE algorithm for community detection on the degree-corrected block model. One of their main ideas is to overcome the statistical issues caused by the heterogeneity of the degree distribution by constructing test statistics that are properly normalized so as to cancel out the effects of vertex weights. In a similar spirit, [24] considers an inhomogeneous random graph with community and proposes a normalized test based on short paths and short cycles to detect the presence of more than one community. Our work here is also graphlet-based (triangles in this case), but rather than taking all triangles as equal, we weight the triangles based on the inhomogeneity of the network degrees. This provides a robust statistic to infer communities in heavy-tailed networks.

1.2. Structure of the paper

The rest of the paper is structured as follows. In Section 2 we explain the model and the hypotheses for our tests. In Section 3 we provide the tests that we propose, and state our main results on their accuracy, followed by a discussion in Section 4. We finally prove our main results on detecting the presence of a geometric structure in Section 5, and our results on the identification of the geometric structure in Section 6.

1.3. Notation

We adopt the standard notation of a statistical testing problem. The null hypothesis will be denoted by H_0, and the alternative hypothesis by H_1. When operating under H_0, i.e. assuming the null hypothesis holds, the probability of some event E will be denoted by $P_0(E) := \mathbb{P}(E \mid H_0)$. We denote the expected value and the variance with respect to this probability measure by E_0 and Var_0, respectively. On the other hand, when H_1 is assumed to hold, we will similarly use the notation P_1, E_1, Var_1. Throughout the paper we make use of the standard Bachmann–Landau notation. We write $f(n) = o(g(n))$ if $\lim_{n\to\infty} f(n)/g(n) = 0$, $f(n) = O(g(n))$ if $\limsup_{n\to\infty} f(n)/g(n) < \infty$, and $f(n) = \Omega(g(n))$ if $g(n) = O(f(n))$. Finally, we say that a sequence of events $\{E_n\}_{n\geq 1}$ happens with high probability if $\lim_{n\to\infty} \mathbb{P}(E_n) = 1$.

2. Model

We now formulate the problem of community detection in a graph as a hypothesis-testing problem. We are given a single sample of a simple graph $G = (V, E)$, where $V = [n] := \{1, \ldots, n\}$ is the set of nodes, and $E \subseteq \{(i, j) \in V \times V : i < j\}$ are the edges. Note that, by assumption, G does not contain self-loops and multiple edges.

2.1. Null model

Under the null hypothesis H_0, the graph G is a sample of the IRG model, which is defined as follows [14]. To each vertex $i \in V$ we assign a weight w_i, and $F_n(x) = (1/n) \sum_{i\in V} \mathbf{1}_{\{w_i \leq x\}}$ denotes the empirical cumulative weight distribution. F_n can also be seen as the cumulative

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

weight distribution of a uniformly chosen vertex in the graph. We require the weight sequence to satisfy the following assumption.

Assumption 1. *There exist* $\tau \in (2, 3)$ *and* $C, w_0 > 0$ *such that, for all* $x \geq w_0$,

$$1 - F_n(x) = Cx^{1-\tau}(1 + o(1)).$$

Given the weight sequence $\{w_i\}_{i \in V}$, any edge (i, j) is present with probability

$$p_{ij} = p(w_i, w_j) := \min\left(\frac{w_i w_j}{\mu n}, 1\right), \tag{1}$$

independently of all other edges, where $\mu = w_0(\tau - 1)/(\tau - 2)$. In the Supplementary Material we prove that when n is large, μ is asymptotically equal to the average weight.

2.2. Alternative model

Under the alternative hypothesis H_1, k of the vertices form a community. Without loss of generality, we assume these are $V_C := \{1, \ldots, k\} \subset V$. For convenience, we write $V_I := V \setminus V_C$, and we call the elements of V_I *type-A* vertices, while we call the elements of V_C *type-B* vertices. Let us now define the geometric community more precisely. Let $\mathcal{X} = \mathbb{R}^d/\mathbb{Z}^d$ be the d-dimensional torus. We endow $\mathcal{X}$ with the norm

$$\|x - y\| = \sup_{i=1,\ldots,d} \min\{|x(i) - y(i)|, |1 - (x(i) - y(i))|\}, \tag{2}$$

where $x = (x(1), \ldots, x(d))$ and $y = (y(1), \ldots, y(d))$ are elements of $\mathcal{X}$. Note that this is the usual infinity norm compatible with the torus structure. To each vertex $i \in V_C$ we assign a (random) position x_i in the torus $\mathcal{X}$. Formally, $(\mathrm{x}_i)_{i \in V_C}$ is a sequence of random variables distributed uniformly over $\mathcal{X}$, and we will denote by $(x_i)_{i \in V_C}$ a realization of such random sequence. Again, we assign to each vertex $i \in V$ a weight w_i, where $(w_i)_{i \in V}$ is a sequence satisfying Assumption 1. Additionally, defining the empirical cumulative distribution of the vertex weights in the geometric community as $F_k(x) = (1/k) \sum_{i \in V_C} \mathbf{1}_{\{w_i \leq x\}}$, we will also require that $F_k(x)$ has a power-law tail.

Assumption 2. *Let* τ, C, w_0 *be as in Assumption 1. Then, for all* $x \geq w_0$,

$$1 - F_k(x) = Cx^{1-\tau}(1 + o(1)).$$

Under H_1, any edge $(i, j) \in V_I \times V_I$ is present independently of all other edges with probability as in (1). Instead, if $(i, j) \in V_C \times V_I$,

$$p_{ij} = p(w_i, w_j) := \frac{1}{1 + C_1} \min\left(\frac{w_i w_j}{\mu n}, 1\right). \tag{3}$$

That is, pairs with at least one type-A vertex connect with probability determined by the weights of the two endpoints, similarly to under H_0, but with a correction factor $1/(1 + C_1)$ if the other endpoint is a type-B vertex.

Finally, any edge $(i, j) \in V_C \times V_C$ is present independently of all other edges with probability

$$p_{ij} = p(w_i, w_j, x_i, x_j) := \frac{1}{1 + C_1} \min\left(\frac{w_i w_j}{\mu k \|x_i - x_j\|^d}, 1\right)^{\gamma} \tag{4}$$

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

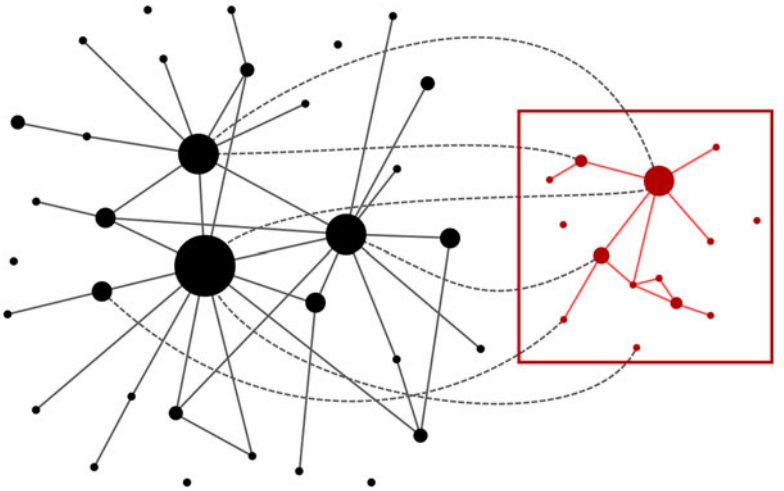


FIGURE 1. Visualization of the geometric community in the alternative model. Black and red dots represent type-A and type-B vertices, respectively, and their sizes grow with vertex weights.

for some $\gamma \in (1, \infty]$. This is a geometric connection probability on k vertices similar to the GIRG model [12], multiplied by the factor $1/(1 + C_1) \in (0, 1)$. The correct choice for the correction factor is $C_1 := (1 + (\gamma - 1)^{-1})2^d$, and it will be discussed below. By convention, the choice $\gamma = \infty$ corresponds to the *threshold* connection rule, i.e. $p_{ij} = 1/(1 + C_1)$ if $\|x_i - x_j\|^d \le w_i w_j/(\mu k)$, and $p_{ij} = 0$ otherwise. Thus, these k type-B vertices form connections based on their weights as well as their positions. In particular, the closer x_i and x_j, the more likely they are to connect. The triangle inequality also ensures that a connection between type-B nodes i and j and i and k makes it more likely for an edge between j and k to be present as well. Thus, the type-B vertices are likely to be more clustered than the type-A vertices. Note that an alternative interpretation for the connection rule (4) is that it is the GIRG connection probability on n vertices [12], where positions of vertices V_C are sampled uniformly over the (shrinking) torus $[0, \lceil k/n \rceil]^d$. In Figure 1 we offer a visualization of the graph model introduced above, under the alternative hypothesis H_1.

2.3. Sources of randomness

Observe that under H_1, two sources of randomness are present: the position sequence $(x_i)_{i \in V_C}$, and the random independent connections between vertices. Given a network sample, the positions that generated the network community are usually unknown, and we only observe the network connections. Thus, we assume that we do not know the positional vectors of the community. However, when a given network is a realization of an inhomogeneous random graph or a geometric inhomogeneous random graph, degrees are mixed-Poisson distributed. In particular, the degree of a vertex i in the network is $d_i \sim \text{Poisson}(w_i)$, so that d_i is close to w_i, with high probability when $w_i \gg 1$; see, e.g., [12] and [37, Appendix C]. Therefore, we assume that the weight sequence is known, as it is possible to infer it from the degree distribution of the observed network.

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

2.4. Correction factor

In the IRG, any vertex i has expected degree $w_i(1+o(1))$. On the other hand, the random graph formed under H_1, without the correction factor $1/(1+C_1)$ in (3) and (4), would introduce a bias on the expected degree. Therefore, a simple check on the degree distribution would be sufficient to determine if a random graph has been sampled from H_0 or H_1. With the correction factor of $1/(1+C_1)$, the expected degree of any vertex is $w_i(1+o(1))$ under both H_0 and H_1, as proved in the Supplementary Material, which excludes a trivial detection test.

2.5. Model choice for the community

The presence of a network community can often be attributed to the embedding of its nodes in a hidden metric space [8, 35]. The interpretation for real-world networks is that such a spatial arrangement enables vertices with similar structural or functional characteristics to cluster together, naturally forming a distinct community within the network. Our choice for H_1 follows this logic, as vertices in V_C are embedded in the torus $\mathcal{X}$. Because edge connections are drawn according to the probabilities (3) and (4), for any vertex $v \in V_C$, the proportion of connections between v and the community V_C among the total number of neighbors of v is positive (on average) and equal to $C_1/(1+C_1)$. Thus, the vertices in V_C are tightly connected.

3. Main results

In this section we describe our main results regarding the detection and the identification of the geometric community. First, let us introduce a few important notions. A *test* ψ is a mapping from G to $\{0, 1\}$. Here, $\psi(G)=1$ indicates that the null hypothesis H_0 is rejected and the graph contains a planted geometric community, and $\psi(G)=0$ otherwise. The *risk* of such a test is defined as $R(\psi) := P_0(\psi(G)=1) + P_1(\psi(G)=0)$. Our goal is to distinguish H_0 and H_1 when the graph size n is large. Formally, a sequence of tests $(\psi_n)_{n\geq 1}$ is said to be *asymptotically powerful* when it has vanishing risk, i.e. $\lim_{n\to\infty} R(\psi_n)=0$. Such a sequence of tests identifies the underlying model correctly in the limit of $n\to\infty$.

3.1. Detection

In this section we first describe an asymptotically powerful test for planted geometric community detection, the *weighted triangle test*. We will use the shorthand notation $\{i, j, k\} = \triangle$ to mean $\{(i, j), (j, k), (k, i)\} \subseteq E$. The test uses the *weighted triangles* statistic

$$W(G) := \sum_{a,b,c\in V} \frac{1}{w_a w_b w_c} \mathbf{1}_{\{\{a,b,c\}=\triangle\}}. \tag{5}$$

Thus, each triangle is given a weight that is inversely proportional to the product of the weights of its vertices. In this way, *W discounts* the triangles formed by high-weight vertices. Indeed, triangles between high-weight vertices are likely to be formed in geometric as well as in non-geometric random graphs. Therefore, standard triangle counts are not even able to distinguish between power-law geometric graphs and inhomogeneous random graphs [29], and we need more advanced triangle-based statistics. The main distinction is given by the triangles formed between low-degree vertices, which are unlikely in non-geometric models. The weighted triangle test rejects H_0 when $W(G)$ is larger than some threshold $f(n)$. Formally, the weighted triangle test ψ_W is defined as

$$\psi_W(G) = \mathbf{1}_{\{W(G)\geq f(n)\}}. \tag{6}$$

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

The next result shows that there is significant freedom in the choice of $f(n)$, while still having an asymptotically powerful test.

Theorem 1. *Let $f(n)$ be a function such that $f(n) \to \infty$ as $n \to \infty$ and $f(n) = o(k)$. Then, the weighted triangle test is asymptotically powerful.*

Theorem 1 shows that it is possible to detect the presence of any geometric subset as long as it grows (arbitrarily slowly) in n. Still, the test statistic (6) relies on knowledge of a lower bound on the geometric size k when choosing the threshold $f(n)$, as Theorem 1 requires $f(n) = o(k)$; in Section 3.3 we present a method to overcome this problem.

Based on the same techniques used to prove Theorem 1, we can design a level-α test as follows. Let $\alpha \in (0, 1)$ and define the test $\bar{\psi}_W(G) = \mathbf{1}_{\{W(G) \geq 1 + 1/\sqrt{\alpha\mu^3}\}}$. Using the results in Section 5, it is possible to show that $\lim_{n\to\infty} P_0(\bar{\psi}_W(G) = 1) \leq \alpha$. In other words, without prior knowledge of the potential community size, it is possible to bound the type-I error below any desired level $\alpha \in (0, 1)$.

3.2. Identification

We now focus on the problem of identifying the geometric vertices under H_1. When a test rejects H_0, the following goal is to identify the vertices that are part of the planted geometric part. To this end, let $\hat{V}_C \subseteq V$ be an estimator for the set of geometric vertices. We assume that the size of the planted geometric community, k, is known. To measure the performance of an estimator of the geometric vertices, we use the risk function

$$R_{\mathrm{id}}(\hat{V}_C) := \mathbb{E}_{V_C}\left[\frac{|\hat{V}_C \triangle V_C|}{2|V_C|}\right],$$

where $\hat{V}_C \triangle V_C := ((V \setminus \hat{V}_C) \cap V_C) \cup (\hat{V}_C \cap (V \setminus V_C))$ denotes the symmetric difference between $\hat{V}_C$ and V_C, and $\mathbb{E}_{V_C}$ denotes the expected value given the knowledge of the set V_C. Note that $|\hat{V}_C \triangle V_C| \leq 2|V_C|$ when we assume that the community size is known and $\hat{V}_C$ outputs exactly k vertices, so that in that case $R_{\mathrm{id}} \in [0, 1]$. We say that a method achieves *almost exact recovery* when $R_{\mathrm{id}}(\hat{V}_C) \to 0$, and *partial recovery* when $R_{\mathrm{id}}(\hat{V}_C) \to c$ for $c \in (0, 1)$. In other words, a test achieves almost exact recovery if the number of misclassified vertices is negligible compared to the community size; a test achieves partial recovery when it identifies a positive proportion of the vertices in the community. We refer to Abbe's monograph [1] for a precise definition of different recovery notions. To obtain an estimator for the set of geometric vertices, we construct a test statistic $T\colon V \to \{0, 1\}$ such that $T(i) = 1$ if node $i \in V$ is estimated to be in the community, and $T(i) = 0$ otherwise.

Low-weight vertices in a GIRG have degree zero with positive probability, and zero-degree type-A and type-B vertices cannot be identified. This strongly suggests that in our setting, where $O(n)$ vertices have a weight of order $O(1)$, almost exact recovery cannot be achieved. In fact, even partial recovery is difficult because low-weight vertices are a non-vanishing fraction of all the vertices. We therefore focus on achieving almost exact recovery among the graph induced by all high-weighted vertices.

For the purpose of identification, we propose the test statistic $T\colon V \to \{0, 1\}$

$$T(a) := \mathbf{1}_{\{W(a) > n/(w_a\sqrt{\log n})\}}, \tag{7}$$

where

$$W(a) := \frac{n}{w_a^2} \sum_{b,c \in V} \frac{1}{w_b w_c} \mathbf{1}_{\{\{a,b,c\} = \triangle\}}. \tag{8}$$

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

We next show that when $w_a = w_a(n) \to \infty$ this test leads to vanishing type-I and type-II errors.

Theorem 2. *The test T(a) achieves almost exact recovery among the set of all vertices a with weight* $w_a \gg \log(n)$. *Formally, setting* $\hat{V}_C := \{a \in V\colon T(a) = 1\}$, *we get*

$$\lim_{n\to\infty} \mathbb{E}_{V_C}\left[\frac{\left|\hat{V}_C^{t_n} \triangle V_C^{t_n}\right|}{2\left|V_C^{t_n}\right|} \mid \left|V_C^{t_n}\right| > 0\right] = 0$$

as long as

$$(n\log(n)/k)^{1/\tau} \ll t_n \ll k^{1/(\tau-1)}, \tag{9}$$

where, for any $U \subseteq V$, $U^h := U \cap \{a \in V\colon w_a \geq h\}$.

Note that while Theorem 2 uses knowledge of k in the threshold for the weights on which the risk tends to 0, this threshold can be avoided by choosing $t_n \gg (n\log(n))^{1/\tau}$, as $k \geq 1$. In this case, we only need to know whether the upper limit also holds, i.e. whether $k \gg (n\log(n))^{(\tau-1)/\tau}$, and we only need a sufficiently large lower bound on k.

3.3. Estimating the community size

Next, we tackle a different issue, namely inferring the size of the planted geometric community under H_1. We will make crucial use of the fact that the estimator for the community introduced above identifies high-degree vertices exactly in the $n \to \infty$ limit by Theorem 2.

Let $X_{(1)}, X_{(2)}, \ldots$ denote the order statistics of the weights of the vertices of the geometric part that are identified by Theorem 2. That is, $X_{(1)}$ is the vertex of the geometric part with the highest degree. Thus, we take the m highest-weight vertices that are identified by the node-based test as being part of the GIRG as input. Denote the weights of these vertices by $X_{(1)}$, $X_{(2)}$, and so on. As an estimator of the community size k we propose $\hat{k}_m := mX_{(m)}^{\tau-1}$.

Theorem 3. *Assume that* $k \gg (n\log(n))^{(\tau-1)/(2\tau-1)}$ *and* $m \in \mathbb{N}$. *Then, as* $n \to \infty$, $\hat{k}_m/k \xrightarrow{\mathbb{P}} 1$.

Proof. Let $X_{(1)}, X_{(2)}, \ldots$ denote the order statistics of the weights of the vertices of the GIRG part. By [36, Eq. (4.17)], as $k \to \infty$,

$$\frac{X_{(s)}}{(k/s)^{1/(\tau-1)}} \xrightarrow{\mathbb{P}} 1.$$

Now, the m type-B vertices with the highest weight can be identified correctly with probability tending to 1 as long as there exists some t_n satisfying (9). Such a sequence exists as long as $k \gg (n\log(n))^{(\tau-1)/(2\tau-1)}$. Then, $\hat{k} = X_{(m)}^{\tau-1} m/k \xrightarrow{\mathbb{P}} 1$. □

In Theorem 1, the threshold function $f(n)$ is determined based on knowledge of the community size, k. However, Theorems 2 and 3 offer a way to confirm the existence of a geometric community without requiring prior knowledge of k. The process involves three steps: first, identifying the vertex $X_{(1)}$ with the highest weight in the graph for which the test T in (7) is successful; second, computing $\hat{k}_1 = X(1)^{\tau-1}$; and finally, defining $f(n) = \sqrt{\hat{k}_1}$ before applying Theorem 1.

3.4. Numerical results

We present here some numerical experiments to illustrate the finite-sample performance of our tests.

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

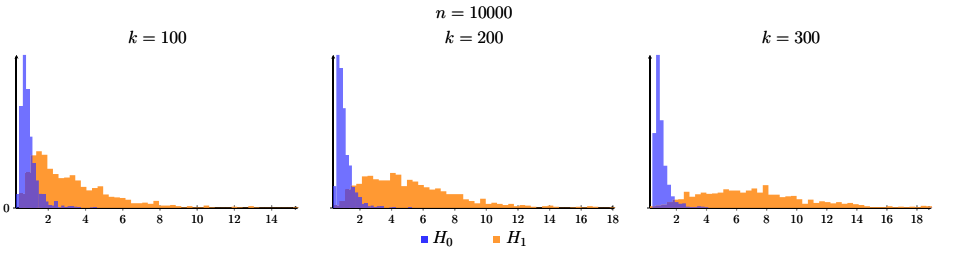
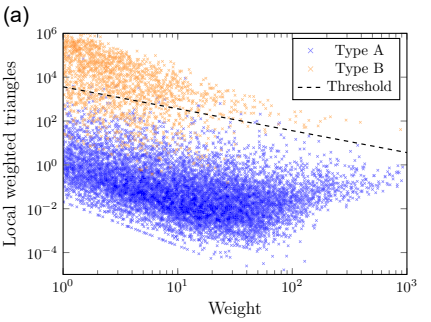
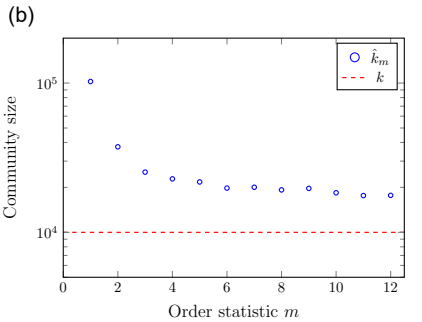


FIGURE 2. Histogram of the value of W for 10^4 sample graphs generated under H_0 (blue) and H_1 (orange) using $\tau = 2.5$, $C = 1$, $w_0 = 1$, $d = 2$, and $\gamma = 5$.



Points are values of the local weighted triangles statistic $W(a)$, for all type-A and type-B vertices, against their weight.



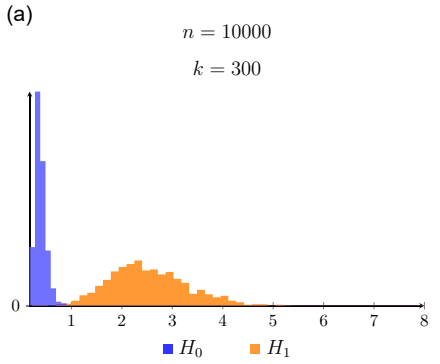
Blue dots are the average of $\hat{k}_m$ over 15 simulations. The red line is the true size of the community, k.

FIGURE 3. Identification of geometric vertices and estimate of the community size under H_1 using the parameters $n = 10^6$, $k = 10^4$, $\tau = 2.5$, $d = 1$, $\gamma = 5$, $C = 1$, and $w_0 = 1$.

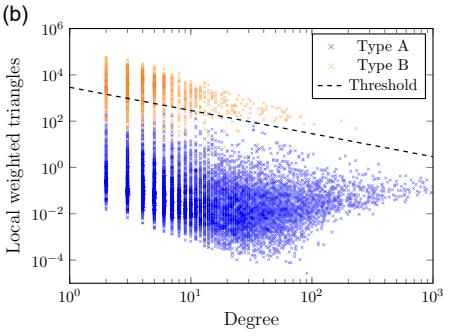
In Figure 2 we compare the histogram of W from (5) evaluated over multiple samples of the null and alternative models. In both cases, the models are generated on $n = 10^4$ vertices, and the size of the geometric community varies between $k = 100, 200, 300$. Under H_0, W is highly concentrated, while under H_1, the typical value of W is larger and increases in k. This is consistent with Propositions 1 and 2, and shows that our test ψ_W can work with high accuracy for finite samples. Furthermore, the larger k is, the better the separation between H_0 and H_1 in terms of W.

Figure 3(a) illustrates the performance of our identification test (7), which plots the value of the local weighted-triangle statistic $W(a)$ for each vertex a for a single sample of H_1. Figure 4(a) shows that the clouds of coordinates $(w_a, W(a))$ of type-B vertices separate well from the cloud formed by type-A vertices. The dotted line in Figure 3(a) is the curve $y = Cn/(x\sqrt{\log n})$, where here C is a constant value tuned to the parameters of the model. According to Theorem 2, all but a small fraction of type-B vertices with large weights lie above the dotted line, whereas the type-A vertices are located below. Our simulations confirm this. We also observe that a large proportion of the vertices with low weights is correctly identified. This suggests that partial recovery might still be achieved for the entire community by ignoring the vertices whose local weighted-triangle statistic equals zero, even though Theorem 2 only works for high-degree vertices.

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press



Histogram of the value of W for 10^4 sample graphs generated under H_0 (blue) and H_1 (orange) using $\tau = 2.5$, $C = 1$, $w_0 = 1$, $d = 2$, and $\gamma = 5$.



Values of the local weighted-triangle statistic $W(a)$ for type-A and type-B vertices against their degrees, using the parameters $n = 10^6$, $k = 10^4$, $\tau = 2.5$, $d = 1$, $\gamma = 5$, $C = 1$, and $w_0 = 1$.

FIGURE 4. Detection of the geometric community and identification of its vertices when using degrees as a proxy for vertex weights.

In Figure 3(b) we show a practical application of Theorem 3. The blue dots are the average values of the resulting estimates of k, obtained from 15 simulations of the model H_1. Theorem 3 shows that $\hat{k}_m$ converges in probability to the real size of the geometric community k as $n \to \infty$. This convergence cannot be observed in Figure 3(b), where the size of the graph is fixed ($n = 10^6$). Nonetheless, the figure shows that $(\hat{k}_m)_m \geq 1$ are able to approximate k quite well, even for moderate values of m.

Lastly, in Figure 4 we observe how well our detection and identification tests perform when we use degrees as a proxy for the unknown weights in (5) and (8). Figure 4 shows that our detection and identification tests still perform well, demonstrating that our tests also work well with observable statistics.

4. Discussion

4.1. Computational complexity

As our test is triangle-based, it only requires a triangle enumeration for all vertices. This can be done in $O(n^{1+(1/\tau)})$ [28] or $O(n^{3/2})$ [27] time, or in time $O(n \log(n))$ for a good approximation [5], providing an extremely efficient method for detecting and identifying geometry.

4.2. Graph sparsity

The null and alternative models compared in this paper are sparse random graphs, where the average degree is fixed as the graph size increases. It is easily checked that our results fail in a denser regime, namely when $\mu \to \infty$, since weighted triangles lose their statistical test significance. Instead of using the weighted triangles introduced in this paper, in a denser regime it could be more appropriate to test for geometry via *signed triangles* [25], which are counted after centering the adjacency matrix. Signed triangles have been used to distinguish

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

Erdös–Rényi from high-dimensional geometric random graphs, and they were shown to perform better than pure triangle counts since they are much less correlated with the edge count in the graph [13].

4.3. Iterative procedure

Our identification procedure identifies high-weight vertices correctly with high probability. We believe that this identification can serve as a starting point for determining the lower-weight vertices with non-trivial probability. Given the identification of the high-weight neighbors of a low-weight vertex, we can compute the likelihood of this vertex being part of the geometric structure or not, and identify it based on the highest likelihood. After this, we can again assess this likelihood with these updated identifications and iteratively improve the estimated geometric subset. Such a procedure has been proven to work for the stochastic block model [40], and using such procedures in this setting also seems promising.

4.4. Improving $\hat{k}$

While Theorem 3 shows that our estimator of k is unbiased in the large-network limit, for finite values of n, $\hat{k}$ overestimates k, as shown in Figure 3b. This is because the test is based on the identified geometric vertices of Theorem 2. In the large-network limit, these are all identified correctly. However, for finite n, some vertices may be misclassified as geometric or non-geometric. Since k is small compared to n, most misclassifications are non-geometric vertices that are misclassified as geometric. These misidentified vertices, therefore, make the inferred order statistics of the geometric vertices higher, leading to an overestimation of k. Improving this estimate for finite n is consequently an interesting line of further research. For example, we could use information on the expected number of misclassified vertices to improve the test.

4.5. Rescaling the box sizes

In our model, we rescaled the GIRG connection probability (4) with the size of the community k. This is equivalent to sampling the locations of the vertices in the community in a shrinking torus, and not rescaling the connection probability. Another natural choice is rescaling the connection probability within the community as

$$p_{ij} = \min\left(\frac{w_i w_j}{\mu n \|x_i - x_j\|^d}, 1\right)^{\gamma}.$$

However, this connection probability leads to a *sparse community*, where most community members will be disconnected from other community members. As the name suggests, in the sparse community scenario the average number of connections between a given vertex in the community and other community vertices decreases roughly as k/n. Because of this, we believe that our assumption of a *localized community* hypothesis of (4) is the more realistic scenario. Still, our detection methods also apply to the sparse community setting as long as $k \gg \sqrt{n}$. The thresholds for identifying such a sparse community are unknown, however, and would be an interesting point for further research to investigate the theoretical limits of our methods.

4.6. Different geometries

Under H_1, the positions of community vertices are sampled uniformly from the torus $\mathcal{X}$ endowed with the infinity norm. It may be interesting to investigate to what extent the results

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

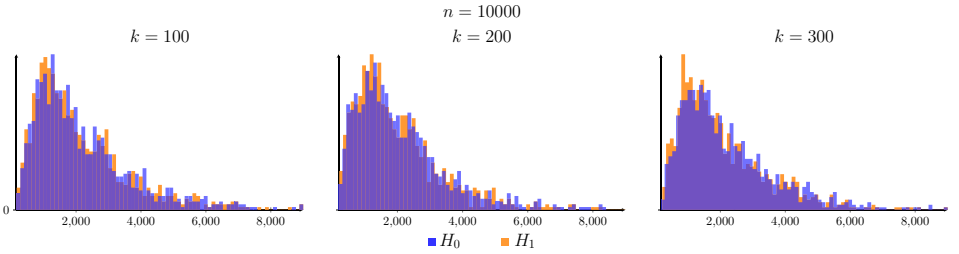


FIGURE 5. Histogram of the number of triangles for 10^4 sample graphs generated under H_0 (blue) and H_1 (orange) using $\tau = 2.5$, $C = 1$, $w_0 = 1$, $d = 2$, and $\gamma = 5$.

presented in this paper hold in a different space. For instance, other choices may be the d-dimensional sphere S^d or the cube $[0, 1]^d$. In both cases we do not expect the results to differ qualitatively, as both the sphere and the cube are locally diffeomorphic to $\mathcal{X}$. Furthermore, using $\|\cdot\|$ in (2) as the infinity norm on $\mathcal{X}$ is not restrictive, as all norms are equivalent in finite-dimensional vector spaces. However, we believe that metrics that are not induced by any norm may lead to different results.

4.7. Discounting triangles on high-weighted vertices

In (5) and (8), we introduced the statistics W and $W(a)$ to detect and identify the geometric community rather than the simpler statistics Δ and $\Delta(a)$, i.e. the triangle counts and the number of triangles containing a. Unfortunately, simple triangle counts may already fail to distinguish geometric and non-geometric graphs, as shown in Figure 5 and proved in [29]. On the other hand, our statistic W mainly counts triangles formed on low-weighted vertices. This quantity is intuitively small in the IRG model, where vertices with low weight mostly connect to vertices with high weights. Instead, a geometric community structure enforces triangles among low-weighted close-by vertices, making W an ideal statistic to detect geometric communities.

4.8. Other subgraph count statistics

Our analysis focuses on counting triangles while excluding those formed around high-weighted vertices. We believe that extending similar statistical measures to larger cliques would show that under the null model (H_0) they remain bounded, whereas in the alternative model (H_1) these statistics grow linearly with k; see [31]. Thus, incorporating additional clique patterns would not improve the performance of our test significantly. Furthermore, identifying larger motifs within a network poses computational challenges. While clique patterns may not improve our approach, subgraph-counting statistics that assign different weights to patterns could potentially lead to better performance; see [30].

4.9. Achieving partial/almost exact recovery

In Theorem 2 we show that the test $T(a)$ achieves almost exact recovery among the set of vertices with weight $w_a \gg \log(n)$. The question is: can we get a better result using a different local test, and achieve either partial or almost exact recovery for all vertices? The answer is no. Triangle-based statistics fail, since under H_1 there is a positive proportion of vertices with degree at most one. In particular, a positive proportion of vertices is isolated. Therefore, no local test can determine if an isolated vertex is part of the geometric community. However,

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

designing a suitable test to achieve partial or almost exact recovery among all non-isolated vertices is an interesting open problem.

4.10. Different connectivity kernels

This paper examines models based on the Chung–Lu type connection probability with a multiplicative kernel, as defined in (1), (3), and (4). This choice ensures that the models exhibit a scale-free structure with a heavy-tailed degree distribution, a common characteristic of real-world networks. In contrast, generic connectivity kernels $p_{ij} = f(w_i, w_j)$ do not inherently possess this property without additional adjustments. Still, it would be interesting to investigate what types of statistics would distinguish geometric regions for other kernels. A key aspect of our methodology is a precise understanding of the triangle distribution. In multiplicative kernel models, triangles predominantly form among nodes with degrees of order $\sqrt{n}$ or higher, while geometric communities in these models generate a substantial number of triangles among low-degree nodes. Because triangle formation patterns are central to our analysis, extending this approach to general kernels would require a deeper exploration of how triangles emerge in different connectivity structures.

5. Detection: Proofs

In this section we prove Theorem 1. The proof relies on the application of the second moment method. To apply Chebyshev's inequality, we first provide upper and lower bounds for $E[W]$ and $\mathrm{Var}\,(W)$, under the null and alternative hypotheses.

Proposition 1. (W under H_0.) *Under H_0, the expected value and the variance of W are*

$$\mathbb{E}_0[W] = 1 + o(1), \qquad \mathrm{Var}_0\,(W) \le \frac{1}{\mu^3}(1 + o(1)).$$

Proposition 2. (W under H_1). *Assume $k \equiv k(n) \to \infty$ as $n \to \infty$. Under H_1, the expected value and the variance of W are*

$$\mathbb{E}_1[W] = \Omega(k), \qquad \mathrm{Var}_1\,(W) = O(k).$$

The proofs of Propositions 1 and 2 can be found in the Supplementary Material.

Proof of Theorem 1. By Proposition 1 and Chebyshev's inequality,

$$P_0(W(G) > f(n)) \le \frac{\mathrm{Var}_0\,(W(G))}{f(n)^2} \to 0.$$

Furthermore, by Proposition 2, for n sufficiently large there exists a constant M such that

$$P_1(W(G) < f(n)) \le P_1(W(G) - \mathbb{E}[W(G)] < f(n) - Mk).$$

Then, again by Chebyshev's inequality,

$$P_1(W(G) < f(n)) \le \frac{\mathrm{Var}_1\,(W(G))}{(f(n) - Mk)^2} \to 0$$

by Proposition 2. □

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

6. Identification: Proofs

We will now show that, under the alternative hypothesis H_1, a positive fraction of high-degree vertices in the geometric community of the network can be distinguished through the localized triangle statistic. The result will follow, again, from Chebyshev's inequality. First, we need some results on the first and second moments of the localized statistic $W(a)$, for vertices inside or outside the geometric community.

Proposition 3. ($W(a)$ outside V_C.) *Suppose a is a type-A vertex. When $k = o(n)$,*

$$\mathbb{E}_1[W(a)] \leq \frac{1}{2\mu}(1 + o(1)).$$

Proposition 4. ($W(a)$ inside V_C.) *Suppose a is a type-B vertex. When $k = o(n)$ and $k \to \infty$ as $n \to \infty$, $\mathbb{E}_1[W(a)] = \Omega(n/w_a)$.*

Proposition 5. (Variance of $W(a)$.) *Suppose $k = o(n)$, and $k \to \infty$ as $n \to \infty$. For any a,* $\mathrm{Var}_1(W(a)) = O(n^2/w_a^3)$.

The proofs of Propositions 3, 4, and 5 can be found in the Supplementary Material. We are now ready to demonstrate the validity of our identification test.

Proof of Theorem 2. Assume that $w_a \gg \log(n)$.

We first calculate the type-I error. When a is type-A, from Propositions 3 and 5 we have $\mathbb{E}_1[W(a)] = O(1)$, $\mathrm{Var}(W(a)) = O(n^2/w_a^3)$. By Chebyshev's inequality,

$$P\left(W(a) > \frac{n}{w_a\sqrt{\log(n)}}\right) \leq \frac{\mathrm{Var}(W(a))}{((n/(w_a\sqrt{\log(n)}))(1 + o(1)))^2} = O\left(\frac{\log(n)}{w_a}\right). \tag{10}$$

Let $\tilde{n}$ denote the number of vertices with weight at least t_n, and $\tilde{k}$ the number of type-B vertices with weight at least t_n. For the type-II error, from Propositions 4 and 5 we have $\mathbb{E}_1[W(a)] = \Omega(n/w_a)$, $\mathrm{Var}(W(a)) = O(n^2/w_a^3)$ for any type-B vertex a. Then, by Chebyshev's inequality,

$$\begin{aligned} P\left(W(w_i) < \frac{n}{w_a\sqrt{\log(n)}}\right) &\leq \frac{\mathrm{Var}(W(a))}{((n/(w_a\sqrt{\log(n)})) - \mathbb{E}_1[W(a)])^2} \\ &= O\left(\frac{n^2/w_a^3}{(n/w_a)^2}\right) = O\left(\frac{1}{w_a}\right). \end{aligned} \tag{11}$$

Thus, the expected number of misclassified type-B vertices equals $\tilde{k}K_2(h\log(n))^{-\tau} = o(\tilde{k})$ for some $K_2 > 0$.

With weight threshold t_n for which we apply the test, as $\tilde{k}$ is binomial with mean $nt_n^{1-\tau}$, [3, Theorem A.1.4] yields that, for all $\varepsilon > 0$,

$$\mathbb{P}(\mathcal{E}_1) := \mathbb{P}\left(\tilde{k} < (1-\varepsilon)kt_n^{1-\tau}\right) \leq \exp\left(-kt_n^{1-\tau}((1-\varepsilon)\log(1-\varepsilon) + \varepsilon)\right) = \exp\left(-\tilde{\varepsilon}kt_n^{1-\tau}\right)$$

for some $\tilde{\varepsilon} > 0$, and

$$\mathbb{P}(\mathcal{E}_2) := \mathbb{P}\left(\tilde{n} < (1+\varepsilon)nt_n^{1-\tau}\right) \leq \exp\left(-nt_n^{1-\tau}((1+\varepsilon)\log(1-\varepsilon) - \varepsilon)\right) = \exp\left(-\hat{\varepsilon}nt_n^{1-\tau}\right)$$

for some $\hat{\varepsilon} > 0$. Thus, as $k < n$, $\mathbb{P}(\bar{\mathcal{E}}_1 \cap \bar{\mathcal{E}}_2) \geq 1 - \exp\left(-\zeta kt_n^{1-\tau}\right)$ for some $\zeta > 0$. By (10) with $w_0 = t_n$, on $\bar{\mathcal{E}}_2$, the expected number of misclassified type-A vertices equals

$$O\left(\sum_{a\in[n-k]:w_a>t_n} \frac{\log(n)}{w_a}\right) = O\left(\log(n)t_n^{-\tau}nt_n^{1-\tau}\right),$$

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

where $nt_n^{1-\tau}$ appears since on $\bar{\mathcal{E}}_2$ there are at most $(1+\varepsilon)nt_n^{1-\tau}$ vertices of type A with weight at least t_n. Furthermore, misclassified type-B vertices are $o(\tilde{k})$ by (11).

Now,

$$\begin{aligned}\mathbb{E}\left[\frac{|\hat{V}_C^{t_n}\triangle V_C^{t_n}|}{2|V_C^{t_n}|} \mid V_C^{t_n}\geq 1\right] &= \mathbb{P}(\mathcal{E}_1\cup\mathcal{E}_2)\mathbb{E}\left[\frac{|\hat{V}_C^{t_n}\triangle V_C^{t_n}|}{2|V_C^{t_n}|} \mid \mathcal{E}_1\cup\mathcal{E}_2,\, V_C^{t_n}\geq 1\right]\\ &\quad + \mathbb{P}(\bar{\mathcal{E}}_1\cap\bar{\mathcal{E}}_2)\mathbb{E}\left[\frac{|\hat{V}_C^{t_n}\triangle V_C^{t_n}|}{2|V_C^{t_n}|} \mid \bar{\mathcal{E}}_1\cap\bar{\mathcal{E}}_2\right]\\ &\leq n\exp\left(-\tilde{\varepsilon}kt_n^{1-\tau}\right) + \frac{O\big(\log(n)t_n^{1-2\tau}n\big)}{(1-\varepsilon)kt_n^{1-\tau}} = o(1), \qquad (12)\end{aligned}$$

as long as $(n\log(n)/k)^{1/\tau} \ll t_n \ll k^{1/(\tau-1)}$. □

Funding information

RM was partially supported by PNRR MUR project GAMING, 'Graph Algorithms and MinINg for Green agents' (PE0000013, CUP D13C24000430001).

Competing interests

There were no competing interests to declare that arose during the preparation or publication process of this article.

Supplementary material

The supplementary material for this article can be found at https://doi.org/10.1017/jpr.2025.10038

References

[1] Abbe, E. (2018). Community detection and stochastic block models: Recent developments. *J. Mach. Learn. Res.* **18**, 1–86.

[2] Alon, N., Krivelevich, M. and Sudakov, B. (1998). Finding a large hidden clique in a random graph. *Random Structures Algorithms* **13**, 457–466.

[3] Alon, N. and Spencer, J. H. (2016). *The Probabilistic Method*, 4th edn. John Wiley, Hoboken, NJ.

[4] Arias-Castro, E. and Verzelen, N. (2014). Community detection in dense random networks. *Ann. Statist.* **42**, 940–969.

[5] Becchetti, L., Boldi, P., Castillo, C. and Gionis, A. (2010). Efficient algorithms for large-scale local triangle counting. *ACM Trans. Knowledge Discovery from Data* **4**, 1–28.

[6] Bet, G., Bogerd, K., Castro, R. M. and van der Hofstad, R. (2021). Detecting a botnet in a network. *Math. Statist. Learn.* **3**, 315–343.

[7] Bogerd, K., Castro, R. M., van der Hofstad, R. and Verzelen, N. (2021). Detecting a planted community in an inhomogeneous random graph. *Bernoulli* **27**, 1159–1188.

[8] Boguñá, M., Bonamassa, I., Domenico, M. D., Havlin, S., Krioukov, D. and Serrano, M. Á. (2021). Network geometry. *Nature Rev. Phys.* **3**, 114–135.

[9] Boguñá, M., Papadopoulos, F. and Krioukov, D. (2010). Sustaining the internet with hyperbolic mapping. *Nature Commun.* **1**, 1–8.

[10] Brennan, M., Bresler, G. and Nagaraj, D. (2020). Phase transitions for detecting latent geometry in random graphs. *Prob. Theory Relat. Fields* **178**, 1215–1289.

[11] Bresler, G. and Nagaraj, D. (2018). Optimal single sample tests for structured versus unstructured network data. *Proc. Mach. Learn. Res.* 75, 1657–1690.

[12] Bringmann, K., Keusch, R. and Lengler, J. (2019). Geometric inhomogeneous random graphs. *Theoret. Comput. Sci.* **760**, 35–54.

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press

[13] Bubeck, S., Ding, J., Eldan, R. and Rácz, M. Z. (2016). Testing for high-dimensional geometry in random graphs. *Random Structures Algorithms* **49**, 503–532.
[14] Chung, F. and Lu, L. (2002). The average distances in random graphs with given expected degrees. *Proc. Nat. Acad. Sci. USA* 99, 15879–15882.
[15] Eldan, R. and Mikulincer, D. (2020). Information and dimensionality of anisotropic random geometric graphs. In *Geometric Aspects of Functional Analysis*, eds R. Eldan, B. Klartag, A. Litvak, and E. Milman. Springer, New York, pp. 273–324.
[16] Faloutsos, M., Faloutsos, P. and Faloutsos, C. (1999). On power-law relationships of the internet topology. *ACM SIGCOMM Comp. Commun. Rev.* **29**, 251–262.
[17] Gao, C. and Lafferty, J. (2017). Testing network structure using relations between small subgraph probabilities. Preprint, arXiv:1704.06742.
[18] García-Pérez, G., Boguñá, M., Allard, A. and Serrano, M. Á. (2016). The hidden hyperbolic geometry of international trade: World trade atlas 1870–2013. *Sci. Rep.* **6**, 33441.
[19] Ghoshdastidar, D., Gutzeit, M., Carpentier, A. and Von Luxburg, U. (2020). Two-sample hypothesis testing for inhomogeneous random graphs. *Ann. Statist.* **48**, 2208–2229.
[20] Girvan, M. and Newman, M. E. (2002). Community structure in social and biological networks. *Proc. Nat. Acad. Sci*. 99, 7821–7826.
[21] Hajek, B., Wu, Y. and Xu, J. (2015). Computational lower bounds for community detection on random graphs. *Proc. Mach. Learn. Res*. 40, 899–928.
[22] Holland, P. W., Laskey, K. B. and Leinhardt, S. (1983). Stochastic blockmodels: First steps. *Social Networks* **5**, 109–137.
[23] Jin, J. (2015). Fast community detection by score. *Ann. Statist.* **43**, 57–89.
[24] Jin, J., Ke, Z. and Luo, S. (2018). Network global testing by counting graphlets. *Proc. Mach. Learn. Res*. 80, 2333–2341.
[25] Kaur, G. and Röllin, A. (2021). Higher-order fluctuations in dense random graph models. *Electron. J. Prob.* **26**, 1–36.
[26] Krioukov, D., Papadopoulos, F., Kitsak, M., Vahdat, A. and Boguná, M. (2010). Hyperbolic geometry of complex networks. *Phys. Rev. E* **82**, 036106.
[27] Latapy, M. (2007). Practical algorithms for triangle computations in very large (sparse (power-law)) graphs. Available at https://www-complexnetworks.lip6.fr/~latapy/Publis/triangles_short.pdf.
[28] Latapy, M. (2008). Main-memory triangle computations for very large (sparse (power-law)) graphs. *J. Theoret. Comp. Sci.* **407**, 458–473.
[29] Michielan, R., Litvak, N. and Stegehuis, C. (2022). Detecting hyperbolic geometry in networks: Why triangles are not enough. *Phys. Rev. E* **106**, 054303.
[30] Michielan, R. and Stegehuis, C. (2025). Optimal network geometry detection for weak geometry. *Phys. Rev. E* **112**, 014314.
[31] Michielan, R., Stegehuis, C. and Walter, M. (2024). Optimal subgraphs in geometric scale-free random graphs. Preprint, arXiv:2404.14972.
[32] Newman, M. E. (2006). Modularity and community structure in networks. *Proc. Nat. Acad. Sci*. 103, 8577–8582.
[33] Newman, M. E. and Girvan, M. (2004). Finding and evaluating community structure in networks. *Phys. Rev. E* **69**, 026113.
[34] Newman, M. E. J. (2003). Properties of highly clustered networks. *Phys. Rev. E* **68**, 026121.
[35] Papadopoulos, F., Psomas, C. and Krioukov, D. (2015). Network mapping by replaying hyperbolic growth. *IEEE/ACM Trans. Networking* **23**, 198–211.
[36] Resnick, S. I. (2007). *Heavy-Tail Phenomena: Probabilistic and Statistical Modeling*. Springer, New York.
[37] Stegehuis, C., van der Hofstad, R., van Leeuwaarden, J. S. H. and Janssen, A. J. E. M. (2017). Clustering spectrum of scale-free networks. *Phys. Rev. E* **96**, 042309.
[38] Verzelen, N. and Arias-Castro, E. (2015). Community detection in sparse random networks. *Ann. Appl. Prob.* **25**, 3465–3510.
[39] Wu, Y. and Xu, J. (2021). Statistical problems with planted structures: Information-theoretical and computational limits. In *Information-Theoretic Methods in Data Science*, eds M. R. D. Rodrigues and Y. C. Eldar. Cambridge University Press, pp. 383–424.
[40] Yun, S.-Y. and Proutiere, A. (2014). Accurate community detection in the stochastic block model via spectral algorithms. Preprint, arXiv:1412.7335.

https://doi.org/10.1017/jpr.2025.10038 Published online by Cambridge University Press