Computing (2024) 106:2675–2705
https://doi.org/10.1007/s00607-024-01303-z

REGULAR PAPER

Quickcent: a fast and frugal heuristic for harmonic centrality estimation on scale-free networks

Francisco Plana[1] · Andrés Abeliuk[1,2] · Jorge Pérez[3]

Received: 14 March 2023 / Accepted: 31 May 2024 / Published online: 8 June 2024

Abstract

We present a simple and quick method to approximate network centrality indexes. Our approach, called *QuickCent*, is inspired by so-called *fast and frugal* heuristics, which are heuristics initially proposed to model some human decision and inference processes. The centrality index that we estimate is the *harmonic* centrality, which is a measure based on shortest-path distances, so infeasible to compute on large networks. We compare *QuickCent* with known machine learning algorithms on synthetic network datasets, and some empirical networks. Our experiments show that *QuickCent* can make estimates that are competitive in accuracy with the best alternative methods tested, either on synthetic scale-free networks or empirical networks. QuickCent has the feature of achieving low error variance estimates, even with a small training set. Moreover, *QuickCent* is comparable in efficiency—accuracy and time cost—to more complex methods. We discuss and provide some insight into how QuickCent exploits the fact that in some networks, such as those generated by preferential attachment, local density measures such as the in-degree, can be a good proxy for the size of the network region to which a node has access, opening up the possibility of approximating expensive indices based on size such as the harmonic centrality. This same fact may explain some evidence we provide that QuickCent would have a superior performance on empirical information networks, such as citations or the internet. Our initial results show that simple heuristics are a promising line of research in the context of network measure estimations.

Keywords Centrality measure · Complex networks · Power-law distribution · Degree

✉ Francisco Plana
fplana@dim.uchile.cl

Jorge Pérez
https://www.cero.ai/

[1] Department of Computer Science, Universidad de Chile, Beauchef 851, Santiago, Chile

[2] National Center for Artificial Intelligence (CENIA), Vicuña Mackenna 4860, Macul, Chile

[3] cero.ai, Providencia, Chile

Mathematics Subject Classification 91D30 · 05C82

1 Introduction

1.1 Heuristics are a model of cognitive processes

Some models based on heuristics have been proposed to account for cognitive mechanisms [1], which assume that, though these heuristics are used at a lesser computational cost, they sacrifice accuracy and lead to systematic errors. This viewpoint has been challenged by the so-called *Fast and frugal* heuristics [2], which are simple heuristics initially proposed to model some human decision and inference processes. They have shown that very simple human-inspired methods, by relying on statistical patterns of the data, can reach accurate results, in some cases even better than methods based on more information or complex computations [2, 3]. Due to these features, fast and frugal heuristics have been applied in problems different from their original motivation, including medical decision-making [4], predicting the outcomes of sport matches [5] and geographic profiling [6].

1.2 The problem of centrality computation

In this paper, we provide an example of the usefulness of one of these simple heuristics for estimating the centrality index in a network. Originally, these indexes were proposed as a measure of the importance of a node given by its location in a social network [7], but today find diverse applications such as identifying influential spreaders on epidemic propagation [8], early adopters of innovation [9], or prediction of diseases from cortical networks [10]. We chose to estimate the *harmonic centrality* index [11], a sound measure which, for example, is a competitive ranking function for the relevance of web queries results [12]. It also satisfies a set of necessary axioms that any centrality should meet [12], namely that nodes belonging to large groups are important (*size* axiom); that nodes with a denser neighborhood, i.e. with more connections, are more important (*density* axiom); and that the importance increases with the addition of an arc (*score-monotonicity* axiom). Consider a directed graph $G = (V, A)$, with V the set of nodes and A the set of arcs or edges. Formally, let $d_G(y, x)$ be the length of the shortest path from node y to x in the digraph G. The harmonic centrality of x is computed as

$$H_G(x) = \sum_{y \in V, y \neq x} \frac{1}{d_G(y, x)}, \tag{1}$$

which has the nice property of managing unreachable nodes in a clean way.

Besides its good properties, to compute the harmonic centrality for all nodes in a network we need first to solve the all-pairs shortest-path problem. Notice that by the total number of pairs of nodes, there is an intrinsic lower bound of $| V |^2$ for computing this centrality, and $O(| V |^2)$ is already a huge constraint for modern networks.

There has been a lot of work on optimizing the computation of all-pairs shortest-paths for weighted networks [13–15] but even under strict constraints on the structure of the networks [15] this computation is unfeasible for networks with a large number of nodes, usually needing time $O(\mid A \mid \cdot \mid V \mid)$. Thus, in order to use harmonic centrality in practice we need ways of estimating or approximating it.

Though there are few centrality indexes satisfying the three axioms [12], some simple measures can be built that do satisfy them. One way of doing this, is by taking the simple product of a density measure, such as the in-degree, with a size measure, such as the number of weakly reachable nodes [12]. While the in-degree is cheap to compute, many times stored as an attribute so accessible in constant time, size measures have a higher time complexity. For example, the number of reachable nodes, for each node, can be computed from the condensation digraph of strongly connected components, which may give, in the worst case, a total time complexity of $O(\mid A \mid \cdot \mid V \mid + \mid V \mid^2)$. In this paper, we explore whether expensive indexes, sensitive to either density and size, such as the harmonic centrality, may be approximated by cheap local density measures such as the in-degree.

1.3 Our proposal

Our proposed method, called *QuickCent*, is a modification of QuickEst [16], a heuristic proposed to represent the cognitive processes underlying the human estimation of magnitudes. Since a wide range of natural and human-made phenomena can be modeled as a power-law distribution [17], QuickCent is designed to work on these kinds of distributions. QuickCent can be considered as a generalization of QuickEst, in the sense that, although in this work we focus on centrality approximation, it proposes a general procedure to regress a variable on a predictor when some assumptions are met. QuickCent is a very simple heuristic based on sequences of *binary clues* associated with nodes in a network; the value of a clue is an indicator of the presence or absence of an attribute signal of greater centrality for a node. The method finds the first clue with value 0 (absence), and it outputs an estimate according to this clue. All the clues used in QuickCent are based on the in-degree of the node, thus QuickCent can be seen as a method to regress a variable (harmonic centrality) that correlates with a predictor variable (in-degree) that is cheaper to compute.

This paper extends previous work by some of the authors [18], mainly by adding the study of networks defying the heuristics assumptions (Sect. 4.4) and the performance over empirical networks (Sect. 4.5). Besides that, some technical implementation details (Sect. 3.4), together with the use of the in-degree instead of the degree (Sect. 3.1), the examples (Examples 1, 2, 3) and the insight into why harmonic centrality can be estimated with in-degree (Sects. 4.5.1 and 4.5.3), are new. From suggestive evidence in Sect. 4.5.3, it seems to be that QuickCent is better suited to directed networks, rather than to undirected ones as in our previous work [18].

1.4 Related work

In the existing literature, methods for estimating centrality measures have led to two distinct research directions: one that focuses on the ranking of nodes based on their centrality measure without calculating the centrality values for all nodes, and another that is dedicated to approximating the exact centrality values themselves.

Previous research focusing on node ranking is based on the observation that real-world scale-free networks exhibit a sigmoid pattern in the reverse ranking of closeness centrality. Leveraging this observation, heuristic methods for closeness ranking have been introduced that approximate the sigmoid curve by approximating the closeness centrality of nodes positioned at the minimum, middle, and maximum ranks in the network and using it to get the rank of specific nodes based on their explicit closeness centrality [19, 20]. Using a similar approach, Saxena et al. [21, 22] estimate the degree rank of a node and the variance of the rank estimate based on the degree of a node, exploiting the power-law degree distribution characteristic of scale-free networks. The method estimates the necessary network parameters of the power-law distribution, which yields a functional form for the probability of a node having degree k, which is used to estimate the node rank.

Next, we present research on approximating node centrality measures. Several centrality measures are based on the shortest paths between pairs of nodes, such as harmonic centrality (see Eq. 1). For large networks, computing shortest paths is intractable, since the computation is quadratic in the number of nodes. One line of work for estimating centrality measures based on shortest path distances focuses on estimating the distances used to compute the centrality measures. Rattigan et al. [23] introduced the Network Structure Index (NSI) by annotating nodes and mapping pairs of node annotations to estimate their shortest distance. Pfeffer et al. [24] proposed an approach that computes the shortest paths from all nodes in a network while constraining the path distances, which they call k-measure-based centrality measures that can approximate traditional centrality measures.

Another line of work is based on sampling techniques for estimating centrality measures, where one computes exact centrality values for a predefined set of sampled nodes, and then estimates the centrality values for the remaining nodes by extrapolating the contributions of individual nodes in the sample [25, 26]. Chan et al. [27] adapted the sample approximation methods for networks that exhibit modularity, using the community structure of the network to approximate the betweenness centrality measure. Cohen et al. [28] proposed a sampling method in which the closeness centrality of a node is estimated using its distance to the sampled nodes. Their algorithm provides a relative error guarantee. Chechik et al. [29] developed a weighted sampling method that offers statistically guaranteed estimates for all nodes.

Our approach exhibits resemblances to previous methods that rank nodes by leveraging the statistical regularities observed in scale-free networks, particularly the power-law degree distribution. Nonetheless, our method diverges by directly computing the actual centrality values of nodes, as opposed to focusing solely on their rankings.

1.5 Results and future work

Our method is able to generate accurate estimates even if trained with a small proportion—lesser than 10%—of the dataset. We compare QuickCent with three standard machine learning algorithms trained with the same predictor variable over synthetic data and empirical networks. Our results show that QuickCent is comparable in accuracy to the best-competing methods tested, and has the lowest error variance. Moreover, the time cost of QuickCent is in the middle range compared to the other methods, even though we developed a naive version of QuickCent.

QuickCent has particularly good accuracy on synthetic preferential attachment networks, and we discuss how this may be due to the exploitation that higher degree nodes are more likely to be found because more paths lead to them, opening up the possibility of approximating expensive size-based measures such as harmonic centrality. We also show suggestive evidence that empirical information networks, such as citations or the internet, which can be well approximated by the preferential attachment growth mechanism [30–32], would provide an optimal context for this heuristics.

Working in the future with more general notions of local density [33, 34] may serve to extend the validity of the heuristics for more general networks. The results of this paper are a proof of concept to illustrate the potential of using methods based on simple heuristics to estimate some network measures. Whether or not these heuristics provide a realistic model of human cognition, is a wide problem [35] which is out of the scope of this work.

1.6 Structure of the paper

The rest of this paper is structured as follows. We begin in Sect. 2 by introducing the general mechanism of QuickCent, while Sect. 3 presents our concrete implementation. In Sect. 4, we present and discuss the results of our proposal, including the comparison with other machine learning methods either on synthetic or empirical networks. Finally, Sects. 5 and 6 give possible avenues for future work and conclusions.

2 The QuickCent heuristic

In this section, we give a general abstract overview of our proposal, which we call QuickCent. The setting for QuickCent is as follows: the input is a network $G = (V, A)$ and we want to get an accurate estimate of the value of some centrality function $f_C : V \longrightarrow \mathbb{R}$. That is, for every $v \in V$, we want to compute a value $\tilde{f}_v$ that is an estimation of $f_C(v)$. We next explain the general abstract idea of the components of QuickCent.

Analogously to QuickEst [16], our QuickCent method relies on vectors of *n binary clues*. We associate to every node $v \in V$ a vector $\mathbf{x}_v = (x_v^1, x_v^2, \dots, x_v^n) \in \{0,1\}^n$. The intuition is that the value of the i–th component (clue) x_v^i is an indicator of the presence ($x_v^i = 1$) or absence ($x_v^i = 0$) of an attribute signal of greater centrality for node v. Our method also considers the following $n+1$ sets of nodes:

$$\begin{aligned} S_1 &= \{v \in V \mid x_v^1 = 0\} \\ S_i &= \{v \in V \mid x_v^i = 0 \text{ and } x_v^{i-1} = 1\} \quad (2 \le i \le n) \\ S_{n+1} &= \{v \in V \mid x_v^n = 1\} \end{aligned} \tag{2}$$

That is, S_i corresponds to nodes that do not have the i–th attribute while having the previous one. For each one of the sets S_i, with $1 \le i \le n+1$, our method needs a quantity $\bar{f}_i$ which is a summary statistic of the centrality distribution of the nodes in set S_i. QuickCent must ensure that successive clues are associated with higher centrality values, thus we will have that

$$\bar{f}_1 < \bar{f}_2 < \cdots < \bar{f}_n < \bar{f}_{n+1}. \tag{3}$$

With the previous ingredients, the general estimation procedure corresponds to the following simple rule.

General QuickCent heuristic: For node v, we iterate over the n clues, considering every value x_v^i. When we find the first i verifying that $x_v^i = 0$, we stop and output the value $\bar{f}_i$. If node v is such that $x_v^i = 1$ for every $i \in \{1, \dots, n\}$, we output $\bar{f}_{n+1}$.

Example 1 This is a simple example where we assume complete knowledge of the centrality values of all nodes. Let us consider the following network in Fig. 1 of size 25 obtained as a random instance of linear preferential attachment, defined in Sect. 3.2. Table 1 displays only the non-zero values of in-degree and harmonic centrality in this network. A reasonable way to aggregate these values is to consider four sets $S_i, i = 1,2,3,4$, with the following binary clues, $x_v^i = 1$ ($i = 1,2,3$) if and only if $\deg^{\text{in}}(v) > d_i$, with $d_1 = 0$, $d_2 = 3$ and $d_3 = 4$. With this choice, for simple centrality approximation it is natural to take, for example, the median of harmonic centrality on every set S_i as summary statistics, $\bar{f}_1 = 0$, $\bar{f}_2 = 1$, $\bar{f}_3 = 4.666$ and $\bar{f}_4 = 15.75$.

QuickCent provides a simple stopping rule: for each node, the search is finalized when the first clue with value 0 is found. In this sense, the heuristic is *frugal*, given that in many cases it can output an estimate without using all the available information. If our input is a network in which the vast majority of nodes have similar and small centrality values the procedure is likely to stop the search early and give an estimate quickly. An example of this behavior is exhibited by the scale-free or power-law distributions, where most values are small, but rare unbounded fluctuations may appear, see Sect. 3.2. The assumption of a power-law distribution has a reasonable plausibility since this distribution has a pervasive presence in many natural and human-made phenomena [17], although there has been some recent controversy on this topic [36, 37].

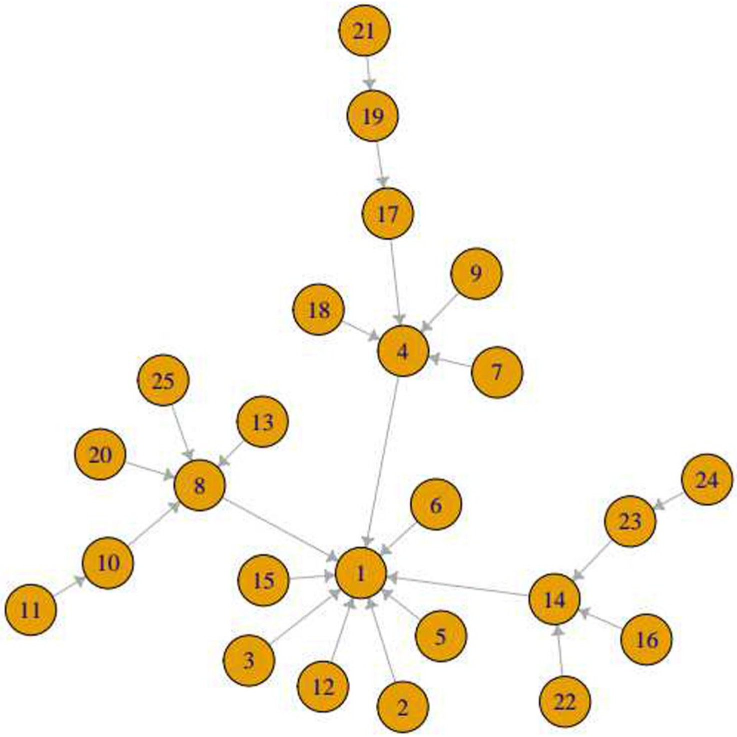


Fig. 1 A network randomly generated with linear preferential attachment

Table 1 In-degree and harmonic centrality values for each node of the network from Fig. 1

Node	1	4	8	10	14	17	19	23
In-degree	9	4	4	1	3	1	1	1
Harmonic	15.750	4.833	4.500	1.000	3.500	1.500	1.000	1.000
QC100	13.429	2.973	2.973	1.309	1.309	1.309	1.309	1.309
QC70	6.531	2.197	2.197	1.214	1.214	1.214	1.214	1.214

Nodes that do not appear here have a zero in-degree and centrality. The last two rows correspond to QuickCent models described in Example 3. The number of decimal places is truncated to three with respect to the source

Up to this point, QuickCent remains similar to QuickEst, whose details can be revised elsewhere [16]. The most critical aspects that distinguish QuickCent from QuickEst, as well as a specification of each part of the heuristic, are presented in the next section.

3 A QuickCent implementation

In this section, we propose an instantiation of our general QuickCent method, including a way to compute the clues x_v^i for every node v based on its in-degree in Sect. 3.1, and an efficient way to compute the summary statistic $\bar{f}_i$ of the centrality for every set S_i in Sect. 3.4. Section 3.3 makes explicit the assumptions that QuickCent requires to be a *ecologically rational heuristic* [16], i.e. the proper network conditions that ensure a successful application of the heuristic, including that the centrality has a power-law distribution. Necessary concepts of this distribution are introduced in Sect. 3.2.

3.1 Using the in-degree for the clues

Our approach for computing the binary clues is to employ a proxy variable related to the centrality by means of a monotonic function which ensures that Eq. (3) holds, with a proxy far cheaper to compute than the actual centrality. Our proxy variable is the in-degree of the node, that is, the number of incoming neighbors of the node, a basic network property many times stored as a node attribute (thus accessible in $O(1)$ time). The intuition for this proxy is that greater in-degree will likely be associated with shorter distances, which likely increases the harmonic centrality. The in-degree can itself be considered as a centrality measure [12]. For a node v we denote by $\deg^{\text{in}}(v)$ its in-degree.

Now, starting from a set of proportions $\{p_i\}_{i=1}^n$, where $0 \leq \cdots \leq p_i \leq p_{i+1} \leq \cdots \leq 1$, we can get the respective *quantile degree values* $\{d_i\}_{i=1}^n$. That is, if F is the cumulative distribution function (CDF) for the in-degree, then $d_i = F^{-1}(p_i)$ for each $i = 1, \ldots, n$. Then, we define the i–th clue for node v as

$$x_v^i = 1 \text{ if and only if } \deg^{\text{in}}(v) > d_i. \tag{4}$$

With this definition, the sets S_i are

$$\begin{aligned} S_1 &= \{v \in V \mid \deg^{\text{in}}(v) \leq d_1\}, \\ S_i &= \{v \in V \mid d_{i-1} < \deg^{\text{in}}(v) \leq d_i\} \quad (2 \leq i \leq n), \\ S_{n+1} &= \{v \in V \mid d_n < \deg^{\text{in}}(v)\}. \end{aligned} \tag{5}$$

Example 2 This type of clues where already used in Example 1. The quantile degree values $d = \{0, 3, 4\}$ used there to split the in-degree values shown in Table 1 from the network in Fig. 1, with the format of the sets shown in Eq. (5), can be obtained via the inverse of the in-degree CDF applied to the set of proportions $p = \{0.68, 0.84, 0.96\}$.

Finally, we propose to compute $\bar{f}_i$ analytically as the median of each S_i based on estimating the parameters of a power-law distribution, as explained in Sect. 3.4. The required background on this distribution is in the next subsection.

3.2 Definition, synthetic model and parameter specification of the power-law distribution

A random variable follows a power-law when its probability density function is given by an expression of the form $p(x) = Kx^{-\alpha}$ where α is called the *exponent* of the power-law, and K is a *normalization constant* depending on α. Few real-world distributions follow a power-law on their whole range; many times the power-law behavior is observed only for higher values, in whose case it is said that the distribution "has a power-law tail". In analytic terms, since this density function diverges when x goes to 0, there has to be a lower limit $x_{\min}$ from which the power-law holds. That is, $x_{\min}$ is a value that satisfies

$$\int_{x_{\min}}^{\infty} Kx^{-\alpha} dx = 1. \tag{6}$$

Moreover, from (6) it is simple to solve that $K = (\alpha - 1)(x_{\min})^{\alpha-1}$. The ℓ–th moment of x is given by

$$\langle x^{\ell} \rangle = \int_{x_{min}}^{\infty} y^{\ell} p(y) dy = \frac{\alpha - 1}{\alpha - 1 - \ell} x_{min}^{\ell}. \tag{7}$$

This expression is well defined for $\ell < \alpha - 1$. In particular, the second moment $\langle x^2 \rangle$ diverges when the exponent $\alpha \leq 3$, and the first moment, the mean, diverges for exponents $\alpha \leq 2$. These features are reflected in the high heterogeneity of values sampled from several real-world distributions, which is the reason they are also referred to as *scale-free* [17]. One of the most known synthetic models engendering networks with power-law degree is preferential attachment (PA). The PA hypothesis states that the rate $\Pi(k)$ in which a k–degree node creates new links is an increasing linear function of k. Suppose the rate $\Pi(k)$ has the following general form,

$$\Pi(k_i) = \frac{k_i^{\beta}}{\sum_j k_j^{\beta}} = C(t) k_i^{\beta}. \tag{8}$$

Krapivsky et al. [38] prove that for $\beta = 1$, or linear PA, this model reduces to the usual power-law BA graph [30] with exponent 3. In the sublinear case, $\beta < 1$, the degree distribution follows a stretched exponential, that is, the bias favoring more connected nodes is weaker, which produces an exponential cutoff that limits the size of hubs. On the other hand, for a superlinear attachment $\beta > 1$, a single node becomes central and connects to nearly all other nodes.

In our work, we need to estimate the α parameter of a power-law distribution. A simple and reliable way to estimate α from a sample $\{x_i\}_{i=1}^{m}$ of m observations from a power-law distribution is to employ the maximum likelihood estimator (MLE) which in this case is given by the formula [17]

$$\hat{\alpha} = 1 + m\left(\sum_{i=1}^{m} \ln \frac{x_i}{x_{\min}}\right)^{-1}. \tag{9}$$

As it is apparent from the previous formula, there are at least two aspects that impact the quality of the estimate $\hat{\alpha}$: the number of observations (m in the formula above), and, in case we do not know the exact value of $x_{\min}$, the estimate $\hat{x}_{\min}$ that we use for it. It is clear how to improve in the first case: we just use more data points. Estimating $x_{\min}$ is a bit more involved. One possible way of computing an estimate for $x_{\min}$ is to visually inspect the log-log plot for the point where the CDF starts to look like a straight line. However, this method is imprecise and highly sensitive to noise [39]. A better method is the one proposed by Clauset et al. [40], which selects the $\hat{x}_{\min}$ that makes the distributions of the empirical data and its fitted power-law model as similar as possible above $\hat{x}_{\min}$, that is, where the fit model is well defined. This similarity, or distance between two probability distributions, could be implemented through the Kolmogorov–Smirnov statistic (KS), whose expression, in this case, is the following

$$D(x_{\min}) = \max_{x \geq x_{\min}} \mid S(x) - P(x) \mid \tag{10}$$

where $S(x)$ is the CDF of the data for observations with a value greater than $x_{\min}$, and $P(x)$ is the CDF of the power-law model that best fits the data (for example, the MLE estimation (9)) in the region $x \geq x_{\min}$. Finally, $\hat{x}_{\min}$ corresponds to the value $x_{\min}$ that minimizes $D(x_{\min})$.

We estimated $x_{\min}$ with the bootstrap method implemented by the *poweRlaw* R package [41], where several samples $x_{\min}$ are drawn and that minimizing $D(x_{\min})$ is selected. We noticed in our experiments that, with high frequency, this method selects $x_{\min}$ as a point with a high value, that is, a $x_{\min}$ value that discards a high portion of the distribution. We have taken the heuristic approach of limiting the search space by an upper bound given by the percentile 20 of the distribution of positive centrality values,[1] since we have seen for many datasets this is enough to span the point where the log-log plot of the complementary ECDF starts to behave like a straight line. Other authors giving implementations of this method have also noticed the difficulties when estimating ${x_{\min}}^{2}$. This method has a statistical consistency that has been proved only for some heavy-tailed models [42]. There are alternative methods to optimize the KS statistic that perform, for example, a grid search over a predefined set of exponent values for each possible $x_{\min}$ that, however, have been claimed to present many drawbacks [43].

Finally, the goodness-of-fit of the fitted power-law models is assessed by means of the test proposed by Clauset et al. [39]. This test produces a p-value p, computed

[1] This is the domain where the power-law fit can be computed.

[2] The commented code from https://github.com/keflavich/plfit says: ...*“The MLE for the power-law alpha is very easy to derive given knowledge of the lowest value at which a power law holds, but that point is difficult to derive and must be acquired iteratively.”*

via a Monte Carlo procedure, which estimates the probability that the KS distance (Eq. (10)) for any random sample is larger than the distance d of a given fit. Thus, if p is close to 1, the fit is acceptable since the difference between the empirical data and the model fit can be explained by random fluctuations; otherwise, if p is close to 0, the model is not an appropriate fit to the data [39].

3.3 The assumptions of QuickCent

The first assumption is the existence of a non-decreasing function g relating the in-degree and the centrality.[3] If there exists a function g satisfying this condition, then the quantiles in the centrality side are equivalent to the application of g on the same degree quantiles [44]. With this result, the quantile proportions can be specified according to characteristics of the centrality distribution, as it is explained in Sect. 3.4. In practice, and even more so considering that the in-degree is a discrete variable while the centrality is continuous, the object g is a relation rather than a function. More formally, let $\{C_i\}_{i=1}^n$ be the set of quantile centrality values associated to the proportions $\{p_i\}_{i=1}^n$ that were used to compute the quantile degree values $\{d_i\}_{i=1}^n$ (see Eq. (4)). Given the above assumption about g, we can rewrite the sets S_i as follows:

$$\begin{aligned} S_1 &= \{v \in V \mid g(\deg^{\text{in}}(v)) \leq C_1\} \\ S_i &= \{v \in V \mid C_{i-1} < g(\deg^{\text{in}}(v)) \leq C_i\} \quad (2 \leq i \leq n) \\ S_{n+1} &= \{v \in V \mid C_n < g(\deg^{\text{in}}(v))\} \end{aligned} \tag{11}$$

The second assumption is that the centrality index that we want to estimate follows a power-law distribution. We add this assumption motivated by the pervasive presence of this distribution in many natural and human-made phenomena [17], as well as the argument that QuickEst would have a *negative bias* [16], in the sense that it is a negative clue (or absent attribute) that stops this heuristic. Thus, a distribution such as the power law where most values are small (with mostly negative clues) and only a few high values exist (with mostly positive clues), would provide an optimal context for the performance of QuickEst. This is consistent with the finding that this heuristic predicts well the estimation behavior by some people on this kind of data [45]. As we next show, the assumption of the power-law distribution will allow us to use some particular properties to approximate the values $\{C_i\}_{i=1}^n$ used in the rewriting above, and then use them to efficiently compute the statistics $\{\bar{f}_i\}_{i=1}^{n+1}$ for every set S_i. In Sect. 4.4, we show some experiments to argue that these two assumptions of the heuristic are key to ensuring its competitive accuracy.

3.4 Putting all the pieces together

Let $D = (V, A)$ be our input network, and recall that we are assuming that the centrality that we want to estimate for D follows a power-law distribution. Let $\hat{\alpha}$ be the estimate of the exponent from Eq. (9), and $\hat{x}_{\min}$ be the estimate of the lower limit

[3] It is required an additional assumption—left continuity—which can be consulted at Hosseini [44].

obtained by (10), which have been computed by considering a set of m nodes in V and their (real) centrality values. With all these pieces, we can compute the values $\{C_i\}_{i=1}^n$ associated to the proportions $\{p_i\}_{i=1}^n$ easily by using the equation

$$\int_{\hat{x}_{\min}}^{C_i} Kx^{-\hat{\alpha}}dx = p_i \tag{12}$$

from which we get that

$$C_i = \hat{x}_{\min} \cdot (1 - p_i)^{\frac{1}{1-\hat{\alpha}}}. \tag{13}$$

Now, in order to compute the summary statistics $\{\bar{f}_i\}_{i=1}^{n+1}$, we will use the median of every set S_i. This median can be computed as follows. Given that we rewrote S_i as the set of centrality values x such that $C_{i-1} \leq x \leq C_i$, then the median md_i of S_i must verify

$$\int_{md_i}^{C_i} Kx^{-\hat{\alpha}}dx = \frac{1}{2}\int_{C_{i-1}}^{C_i} Kx^{-\hat{\alpha}}dx \tag{14}$$

from which we obtain that

$$md_i = \left(\frac{(C_{i-1})^{1-\hat{\alpha}} + (C_i)^{1-\hat{\alpha}}}{2}\right)^{\frac{1}{1-\hat{\alpha}}} = \bar{f}_i \quad (2 \leq i \leq n) \tag{15}$$

Moreover, since the extreme points of the distribution are $x_{\min}$ (estimated as $\hat{x}_{\min}$) and ∞, the two remaining statistics $\bar{f}_1$ and $\bar{f}_{n+1}$ are computed as

$$\bar{f}_1 = \left(\frac{(C_1)^{1-\hat{\alpha}} + (\hat{x}_{\min})^{1-\hat{\alpha}}}{2}\right)^{\frac{1}{1-\hat{\alpha}}} \tag{16}$$

and

$$\bar{f}_{n+1} = 2^{\frac{1}{\hat{\alpha}-1}} \cdot C_n \tag{17}$$

We stress that with these formulas we compute the summary statistic $\bar{f}_i$ for each set S_i just by knowing the values $\{C_i\}_{i=1}^n$, which are computed by using only the values $\hat{\alpha}$, $\hat{x}_{\min}$, and the underlying vector of proportions $\{p_i\}_{i=1}^n$. We choose this last element as the quantile probability values that produced equidistant points on the range of $\{\log(h(v)) \mid v \in V, h(v) \geq \hat{x}_{\min}\}$, that is, the set of vertices where the power-law is well defined for the harmonic centrality. Logarithmic binning is chosen to gauge the tail of the power-law distribution with higher frequency. The length n of the vector of proportions required to construct the clues (see Eq. (4)) was chosen after pilot testing on each type of distribution. See SI Section 2 for more details on the length of this vector. The election of this vector is a way of adapting QuickCent to distinct centrality distributions.

The last element we introduced in our procedure, is the use of an additional quantile centrality value $C_0 = \hat{x}_{\min}$, with the goal of spanning the centrality values

$h(v) < \hat{x}_{\min}$ with greater accuracy. Since for this range of the vertex set the power-law distribution is no longer valid, the representative statistic $\bar{f}_0$ we have used is simply the empirical median of the harmonic centrality in the set of nodes v such that $\deg^{\text{in}}(v) \leq g^{-1}(\hat{x}_{\min})$. With this element, it turns out that, if we use a proportions vector $\{p_i\}_{i=1}^{n}$ of length n, the total number of medians $\{\bar{f}_i\}_{i=0}^{n+1}$ is $(n+2)$. In the code provided to produce the analyses of this paper [46], this element is optional (and activated by setting **rm**=True or **rms**=True). All the results in this paper were obtained with this centrality quantile and median.

Example 3 We continue revisiting Example 1. The power-law exponent $\hat{\alpha}(1)$ (Eq. 9) that turns out from the fit to the whole set of centrality values and by fixing $x_{\min} = 1$, is 2.067. The set of proportions shown in Example 2 comes from evaluating the centrality CDF on the set of points $\{1, 2.506, 6.283\}$, which correspond to $x_{\min}$ and two points ($n = 2$) that in logarithmic scale turn out to be equidistant to the minimum and maximum of the set $\{\log(h(v)) \mid v \in V, h(v) \geq \hat{x}_{\min}\}$, the (log) centrality domain of the given network where the power law is valid. From these parameters and the expressions shown in this section, one can get the medians required by QuickCent to make estimates. These can be examined in Table 1, corresponding to the model **QC100**, which has a MAE (mean absolute error) over the whole digraph of $3.606e-01$. A more interesting case may be computed when $\hat{\alpha}(1)$ is derived from a random sample of the centrality distribution. For example, by taking a sample without replacement of size 70% one may get an exponent estimate of $\hat{\alpha}(1) = 2.477$, which has a MAE of $6.948e-01$ and QuickCent estimates that can be examined in the model **QC70** in Table 1.

This completes all the ingredients for our instantiation of QuickCent, as we have the values for the clues $(x_v^1, x_v^2, \ldots, x_v^n)$ computed from the in-degree of the node v, plus the values $\{d_i\}_{i=1}^{n}$ as shown in Eq. (4), and also the summary statistics $\{\bar{f}_i\}_{i=0}^{n+1}$ for each set S_i, which are the two pieces needed to apply the heuristic.

4 Results

In the present section, we show the results of applying *QuickCent* on synthetic data and empirical networks, and we compare it with alternative procedures for centrality estimation. We first show the comparison of QuickCent with other methods when applied on randomly generated linear PA networks, considering accuracy and time measurements in Sects. 4.2 and 4.3. The experiments show a great accuracy of QuickCent on these networks, together with intermediate elapsed times. Section 4.4 reviews the output of QuickCent on null network models where its accuracy is not as good relative to other methods, with the aim of showing that the two assumptions of QuickCent presented in Sect. 3.3 are jointly required as a necessary condition for the competitive performance of this heuristics. The same benchmark presented for the synthetic case was applied to the empirical datasets, and the results are shown in Sect. 4.5. These last experiments confirm the trends observed before regarding the importance of the QuickCent assumptions for its performance, particularly the monotonic map from in-degree to harmonic centrality, and show how this insight in turn explains the superior performance on linear PA networks, and moreover, on empirical digraphs well approximated by this mechanism such as information networks. The experiments to check the fulfillment of QuickCent assumptions by the different networks are shown in Sections 1, 3, and 4 from the Supplementary Information (SI) document.

4.1 Experiments specifications

In all of our experiments, we consider harmonic centrality as the target to estimate. The number of nodes chosen for the synthetic networks experiments is 10, 000 and 1000 for the null models, with the aim of accelerating the bootstrap computations to check the assumptions of QuickCent on each network. In the case of empirical networks, since the goal is to test an agnostic set of networks, the range of the 39 network sizes is widened to a minimum of 32 and a maximum of 146005 nodes.

The norm that we employed to summarize the error committed on each node is the mean absolute error (MAE), and also the relative error in Sect. 4.5. It is reasonable to use MAE when the summary statistic chosen for each centrality interval is the median, since the conditional median is the solution to the regression problem for the *Minkowski* loss with $\mathcal{L}_1$ norm [47]. On the other hand, the relative error, which is the ratio between the absolute error deviation and the actual harmonic centrality, is used with the goal of obtaining a metric that is valid for distinct networks, with possibly distinct sizes and error magnitudes. For obvious reasons, the relative error is only computed for those nodes where the harmonic centrality is strictly greater than zero.

Regarding the estimate of the lower limit of the distribution, in Sects. 4.2 and 4.4.1, the fixed value $\hat{x}_{\min} = 1$ was used to simplify the calculations, since there is a good fit of this power-law model to the harmonic distributions in these experiments. On the other hand, Sects. 4.4.2 and 4.5 where more general networks were tested, used the procedure explained in Sect. 3.2 to estimate the lower limit. For general networks, it should be more accurate to use the fitted value of $\hat{x}_{\min}$ than a fixed value, although this depends on the variability range existing on the values less than $\hat{x}_{\min}$, which can introduce potentially large contributions to the estimation error.

Finally, all the experiments were performed on the R language [48] with igraph library [49] for graph manipulation, and ggplot2 library [50] to produce the plots. In each one of the plots displaying the error distributions, there is a boxplot showing the MAE distribution for each regression method. Each boxplot goes from the 25th-percentile to the 75-th percentile, with a length known as the *inter-quartile range* (IQR). The line inside the box indicates the median and the rhombus indicates the mean. The whiskers start from the edge of the box and cover until the furthest point within 1.5 times the IQR. Any data point beyond the whisker ends is considered an

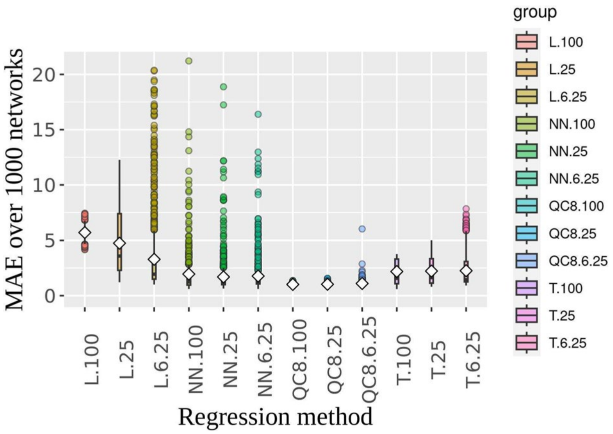


Fig. 2 Benchmark with other ML methods for linear PA digraph instances and training sizes 100%, 25% and 6.25%. The number after the dot of each method corresponds to the size of the training set. The number 8 after QC corresponds to the length of the proportions vector. For display purposes, the vertical limit of the plot has been set to 21, since with this value the points of all MAE distributions are contained in the plot, except those of the neural network models, which have outliers well beyond the magnitude of the other methods' outliers (max(NN.100) = 252.163, max(NN.25) = 116.302, max(NN.6.25) = 87.064)

outlier, and it is drawn as a dot. Finally, all of the experiments were carried out in a Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz, with 24 cores, and 29 GB RAM.

4.2 Experiments with synthetic networks

In this section, we compare the performance of known existing regression methods with QuickCent. Specifically, whether it can deliver reasonable estimates, in relation to alternative solutions for the same task. This is not a trivial matter, considering that QuickCent is designed to do little computational work of parameter estimation and output production, possibly with limited training data, while common alternative machine learning (ML) methods generally perform more complex computations. For a fair comparison, all other methods use only the in-degree as an explanatory variable. In rigor, QuickCent is able to produce the estimates only from the binary clues, without using the in-degree. The competing methods considered are linear regression (denoted by L in plots), a regression tree (T) [51, 52] and a neural network (NN) [53], which are representatives of some of the most known machine learning algorithms. We used *Weka* [52] and the *RWeka* R interface [54] to implement T and NN using default parameters.

The results of this experiment are shown in Fig. 2 with training sizes of 100%, 25% and 6.25%, where the test set is always the entire digraph. In the figure, it can be seen the remarkable result that QuickCent (QC) produces the lowest MAE errors of all the methods, either in terms of the IQR length, the mean, median, and outliers. This performance is due to the fact that linear PA networks do meet the assumptions

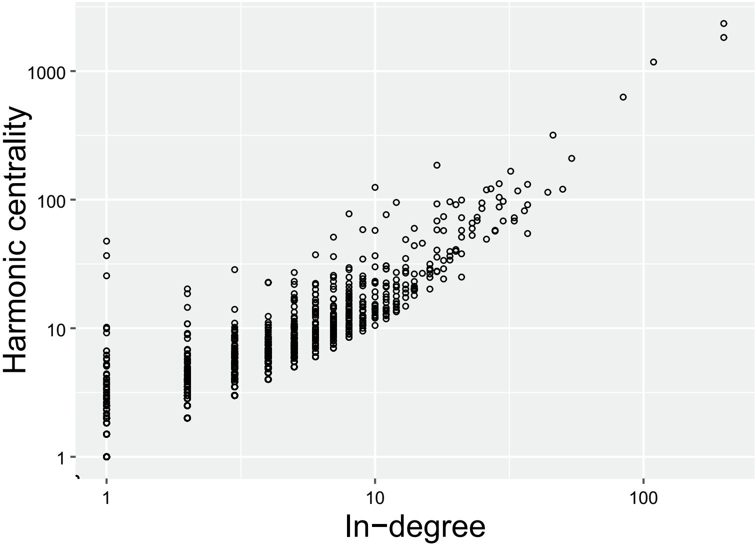


Fig. 3 Scatterplot of in-degree versus harmonic centrality for a random instance of linear preferential attachment network. The axes are in logarithmic (base 10) scale

of the QC heuristic. These networks are known as a paradigmatic case of power-law networks [30]. On the other hand, Fig. 3 shows the relation between in-degree and harmonic centrality for a random linear preferential attachment network, showing that the approximation of assuming that it behaves as a monotonic function is justified. In the SI Section 1 there are additional experiments to validate these assumptions for linear PA networks, as well as for the other regimes of the PA model.

Thus, the main takeaway is that QC, when its assumptions are fulfilled, is able to produce estimates at the same level as much more complex ML methods, with likely lower variance. This fact is consistent with the argument given by Brighton and Gigerenzer [55] claiming that the benefits of simple heuristics are largely due to their low variance. The argument relies on the decomposition of the (mean squared) error into *bias*, the difference between the average prediction over all data sets and the desired regression function, and *variance*, the extent to which the estimates for individual datasets vary around their average [47]. Thus, along the range of the bias-variance trade-off of models, simple heuristics are relatively rigid models with high bias and low variance, avoiding the potential overfitting of more complex models.

By examining the contrast of the outliers between the distinct training sizes, it can be noticed that QC suffers the least impact from scarce data. In the case of L and NN, they share the unexpected pattern of having errors that are lower for the training sizes that do not span the whole network. Since there are only a few large values in the entire graph, when the training sample gets smaller, the sample values have a better linear fit, in comparison to larger samples. Therefore, a linear model adjusted to some small sample provides a good fit to the small-to-moderate centrality size nodes, which is the case for most of the nodes. This also explains the presence of higher errors and outliers on the smaller training sizes for the

Table 2 Mean and standard deviation of elapsed time in milliseconds over 1000 linear PA digraphs of size 10000 nodes, for each of the 4 machine learning methods

	Mean 6.25	S.D. 6.25	Mean 25	S.D. 25	Mean 100	S.D. 100
L	2.5	0.7	2.9	0.8	5	0.94
QC	131.2	21.2	133	29.6	127	29.35
T	32.1	11.7	32.7	8.9	38	5.80
NN	98.7	12.1	284.1	21.4	1007	57.24

The number after the name of the statistic (mean or standard deviation) corresponds to the size of the training size expressed as a percentage of the total vertex set

linear regression. On the other hand, the behavior of the regression tree is more similar to that of QuickCent.

4.3 Time measurements

The time cost is a critical feature of any approximation method because it measures the tradeoff between accuracy and cost. Elapsed time measurements were taken in the experiments shown in Sect. 4.2, and the results are displayed in Table 2. These times consider the training and inference time for each method, without including any centrality computation.

From the table, we can see that the elapsed time of QC is in the middle range of the compared methods. The linear regression has the lowest times, around two orders of magnitude faster than QuickCent, and one order of magnitude smaller than the regression tree time, and the neural network has the highest elapsed time. Note also that there is no significant time difference between the distinct training sizes for QC. This can be explained by the fact that the differences in sample sizes only affect the number of terms in the sum in Eq. (9) when estimating the exponent $\hat{\alpha}$, and summing a list of values is an extremely simple and quick procedure.

These elapsed times from Table 2, as well as the results from Sect. 4.2, do not include the estimation of $\hat{x}_{\min}$, since the fixed value of 1 was used. If we do estimate $\hat{x}_{\min}$ on the same set of 1000 linear PA digraphs of size 10000 nodes, the mean and standard deviation of the elapsed times are 1011 and 70 ms. For 1000 PA networks with exponent $\beta = 0.5$ (Eq. 8) of the same size, the mean and s.d. are 1312 and 77 milliseconds, and for 1000 PA networks with exponent $\beta = 1.5$ of the same size, the mean and s.d. are 74 and 32 milliseconds. These times show that, for general networks where it could be necessary to estimate $\hat{x}_{\min}$, these computations may add up a considerable computational overhead to those performed by QuickCent, which, however, are dependant on the type of network under consideration. More statistics regarding these elapsed times can be reviewed in Tables 1, 2, and 3 from Section 1 in the SI document.

Based on these results, QuickCent is among the lowest time complexity methods tested, and it also may present a competitive execution time for very large networks given its rather invariant elapsed time, having a time execution pattern similar to that of the regression tree. Among the computations that QuickCent performs, the most expensive ones correspond to the selection problem of finding the median of the lowest centrality values (Sect. 3.4), plus the quantile degree values (Sect. 3.1). The procedure used to compute the proportions vector (Sect. 3.4) also relies on solving the selection problem (of the maximum of the set of centrality values) and sorting the centrality values set (to find the proportions). Even the estimation of $\hat{x}_{\min}$ relies on the selection problem of computing the maximum (Eq. 10) of a set of values. Selection and sorting may be solved in expected and worst-case linear time [56] on the input size, that is, linear on the network size $\mathcal{O}(\mid V \mid)$. In contrast to the highly optimized R implementations for L, T, and NN, we considered only a naive implementation of QuickCent without, for example, architectural considerations. With these improvements such as using more appropriate data structures, these times could still be improved. We left as future work the construction of an optimized implementation for QuickCent.

4.4 Networks defying QuickCent assumptions

Up to this point, we have mainly seen examples of networks where QuickCent exhibits quite good performance compared to competing regression methods. In order to give a full account of QuickCent capabilities and its *ecological rationality* [16], one should also have an idea of the networks where its accuracy deteriorates. To accomplish this, we will look at the two assumptions of QuickCent, namely, the power-law distribution of the centrality, and its monotonic map with the in-degree, to show that they are jointly required as a necessary condition for the competitive performance of the heuristic. Our approach here is to work with two null network models, each acting as a negation of the conjunction of the two assumptions, which provide strong evidence for this claim.

4.4.1 Response to the loss of the monotonic map

Our first null model is a scale-free network built by preferential attachment, just as in the previous experiments, but after a *degree-preserving randomization* [57] of the initial network, which is simply a random reshuffling of arcs that keeps the in- and out-degree of each node constant. The aim is, on the one hand, to break the structure of degree correlations found in preferential attachment networks [58, 59], which may be a factor favoring a monotonic relationship between in-degree and harmonic centrality, and on the other hand, to maintain a power-law distribution for the harmonic centrality by preserving the degree sequence of nodes in the network. This last feature does not ensure that the harmonic distribution is a power-law, since randomization also affects this property. However, the experiments performed to check the assumptions of QuickCent on the randomized networks, shown in SI Section 3, confirm that these networks do satisfy them. Finally, Fig. 4 shows the impact of randomization on each regression method. This is an experiment where 1000 PA

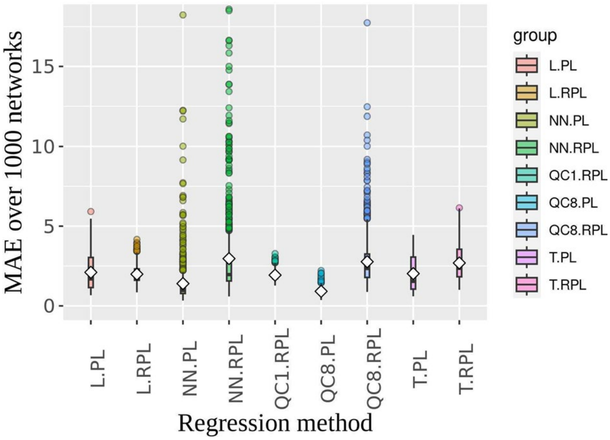


Fig. 4 Effect of randomization on different ML methods using 30% of the training size. Each boxplot group is labeled with the name of the ML method, a dot, and the type of network on which the estimates are made ('PL' for the initial PA network, 'RPL' for the network after randomization). QC8 corresponds to QuickCent with a proportion vector of length 8, and analogously for QC1. For display purposes, the vertical limit of the plot has been set to 18, since with this value the points of all MAE distributions are contained in the plot, except those of the neural network models, which have outliers beyond the magnitude of the other methods' outliers (max(NN.PL)=33.177, max(NN.RPL)= 39.718)

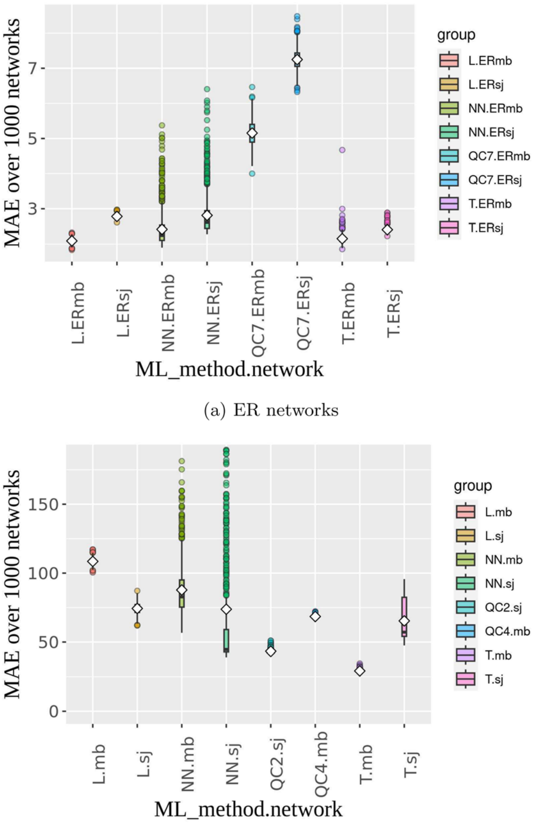
Table 3 General description of the two empirical control networks

Name	N	$\bar{\text{deg}}^{\text{in}}$	Corr	Arc meaning	Refs.
moreno_blogs	990	19.21	0.872	Blog hyperlink	[64]
subelj_jung-j	2208	62.81	0.808	Software dependency	[65]

The fields in the table are the dataset name, the number of nodes with positive in-degree (N), the mean in-degree of nodes with positive in-degree ($\bar{\text{deg}}^{\text{in}}$), the Spearman correlation between the positive values of in-degree and harmonic centrality (Corr), the meaning of the arcs, and the original reference. The name corresponds to the *Internal name*field in the KONECT database. To access the site to download the dataset, append the internal name to the link http://konect.cc/networks/

networks (exponent 1) of size 1000 were created, and the four ML methods used in Sect. 4.2 were trained on each network with samples of size 30% of the total nodeset, using only the in-degree as the predictor variable for the harmonic centrality. The same procedure was run on each network after applying degree-preserving randomization on 10000 pairs of arcs. The plot shows that the randomization has a similar impact on the performance loss of each method, which is an expected result due to the fact that the only source of information used by each method, the in-degree, becomes less reliable due to the weaker association with harmonic centrality thanks to the arc randomization. Since QuickCent was the most accurate of the methods tested on the initial PA networks, it appears to be also one of the most affected methods by randomization.

Fig. 5 Effect of centrality distribution on different ML methods using 30 % of training size. Each boxplot group is labeled with the name of the ML method, a dot, and the type of network on which the estimates are made (‘mb’ for moreno_blogs, ‘sj’ for subelj_jung-j, ‘ERmb’ for the ER digraph created with the parameters of moreno_blogs, and analogously for ‘ERsj’). The number after ‘QC’ is the length of the vector of proportions used by that method, corresponding to the best accuracy for the respective network. For display reasons, the vertical limit of the control networks plot has been set at 184, since with this value the points of all MAE distributions are contained in the plot, except that of NN.sj which has a maximum value of 999.578



(a) ER networks

(b) moreno_blogs and subelj_jung-j

4.4.2 Response to the loss of the power-law distribution of centrality

Our second null model is the directed Erdös-Rényi graph model [60–62], and is chosen with the aim of gauging the impact of losing the power-law distribution of the centrality while maintaining the monotonic map from in-degree to centrality. This model is known to have a Poisson degree distribution [61], a behavior very different from a heavy-tailed distribution. According to our simulations shown in SI Section 4, this model turns out to be ideal for our purposes, since we have chosen connection probabilities that ensure a unimodal distribution for centrality and a strong correlation with in-degree, i.e. with a mean in-degree greater than 1 [62]. To get a fair control on the performance of QuickCent, we have taken two empirical digraphs that satisfy the given condition of the mean in-degree, with node sets of size near 1000, just to accelerate the bootstrap p-value computations. The networks are extracted from the KONECT database [63],[4] and their meta-data is shown in

[4] http://konect.cc/

Table 3. The fields N and $\overline{\deg}^{\text{in}}$ given in this table, are used to determine the network size and the connection probability used to instantiate the respective ER digraphs from the identity $\overline{\deg}^{\text{in}} = p \cdot (N - 1)$.

Finally, in Fig. 5 we can see the results of an experiment analogous to the one with the first null model, that is, there are 1000 iterations where the same four ML methods were trained on each network, two ER graphs with the two connection probabilities and sizes given by the two empirical/control networks, with random samples of size 30% of the total node-set, using only the in-degree. Since the unimodal distribution of ER digraphs is very different from a power-law, in this experiment we have taken the approach of using the parameter $\hat{x}_{\min}$ estimated by the method reviewed in Sect. 3.2, instead of a fixed lower limit as in the previous experiments. Now, by comparing the two plots in Fig. 5, one can observe a noticeable difference in the behavior of QuickCent in the two cases. While QuickCent achieves an average accuracy relative to other regression methods on the control networks with centrality distributions that are more or less close to heavy-tailed, on ER digraphs with similar characteristics to the controls, QuickCent consistently performs worse than other methods. The performance of QuickCent in these plots corresponds to the best possible for each network as a function of the length of the proportions vector, denoted by the number after ‘QC’. These results reveal the critical importance of the centrality distribution of the dataset for the proper functioning of QuickCent. On the other hand, all of the methods exhibit better performance on the ER digraphs than on the corresponding control network, probably due to less heterogeneity in the values to be predicted on the former.

4.5 Experiments with empirical networks

In this section, we present the performance of QuickCent on some real network datasets, either by itself or in comparison to other machine learning methods. The aim is to assess these heuristics in more diverse contexts, as well as to study some of the patterns reviewed in previous sections.

We selected 39 datasets, all of them extracted from the KONECT network database [63],[5] a public online database of more than one thousand network datasets. The criteria for selecting the networks were to select 2–3 networks of distinct magnitude orders of size, from every network category of the database where the networks are unipartite rather than bipartite, with the goal of using a simple framework that spans all the networks reviewed in this work. Each category corresponds to networks coming from distinct fields, and hence, representing different systems. General descriptors of these datasets, as well as several statistics we have computed, are displayed in Table 4. There, we see that we have a diverse set of networks, with distinct edge meanings and sizes ranging from 32 to 146005 nodes.

In the following subsections, we present the most important patterns we have found by studying the correlations among some columns of this table, together with

[5] http://konect.cc/

Table 4 General description of the 39 empirical network datasets

Name	Dir.	N	m	Edge meaning	CHD	GOF	mRE	sdRE	percMED
asoiaf	N	796	32629	Co-appearance	0.705	0.3	217.838	0.115	0.818
cit-HepPh	Y	34546	421578	Citation	0.649	1	7415.077	1	0.871
dblp-cite	Y	12590	49759	Citation	0.643	1	1104.165	0.626	0.636
dimacs10-polblogs	Y	1224	33430	Blog link	0.941	0	392.85	0.718	0.838
eat	Y	23132	511764	Word association	0.851	0.245	2195.6	19.99	0.534
ego-gplus	Y	23628	39242	User friendship	0.537	1	44.412	0.747	0.875
elec	Y	7118	103675	User vote	0.906	1	1671.194	0.271	0.575
epinions	Y	131828	841372	User trust	0.65	0.025	31164.584	153.671	0.416
facebook-wosn-wall	Y	46952	876993	Wall post	0.767	1	11549.749	69.761	0.435
foodweb-baydry	Y	128	2137	Food exchange	0.911	1	60.17	0.237	0.336
foodweb-baywet	Y	128	2106	Food exchange	0.91	1	59.748	0.235	0.332
linux	Y	30837	213954	File inclusion	0.623	0	12882.653	4.054	0.271
loc-brightkite_edges	N	58228	214078	User friendship	0.666	1	16142.611	199.388	0.542
maayan-figeys	Y	2239	6,452	Protein interaction	0.81	1	25.463	0.448	0.669
maayan-foodweb	Y	183	2494	Food source	0.94	1	106.912	0.584	0.755
maayan-Stelzl	Y	1706	6207	Protein interaction	0.681	1	373.046	14.489	0.632
moreno_health	Y	2539	12969	Friendship	0.8	1	354.625	5.919	0.82
moreno_hens	Y	32	496	Hen dominance	0.993	1	11.138	0.111	0.531
moreno_innovation	Y	241	1098	Trust	0.493	0.005	34.488	0.903	0.72
moreno_mac	Y	62	1187	Macaque dominance	0.961	1	18.486	0.761	0.639
moreno_names	N	1773	16401	Noun co-ocurrence	0.75	0.2	573.391	8.269	0.554
moreno_oz	Y	217	2672	Friendship	0.926	1	75.55	0.07	0.736
opsahl-powergrid	N	4941	6594	Power supply	0.319	1	320.151	0.132	0.82
p2p-Gnutella05	Y	8846	31839	Host connection	0.72	1	702.237	7.564	0.802
p2p-Gnutella31	Y	62586	147892	Host connection	0.485	0.025	2084.554	30.504	0.812

Table 4 (continued)

Name	Dir.	N	m	Edge meaning	CHD	GOF	mRE	sdRE	percMED
reactome	N	6327	147547	Protein interaction	0.759	1	2429.303	25.694	0.784
subelj_cora	Y	23166	91500	Citation	0.741	0	3821.564	0.899	0.507
subelj_jdk	Y	6434	150985	Class dependency	0.779	1	5599.494	1.649	0.222
subelj_jung-j	Y	6120	138706	Class dependency	0.808	1	5332.473	1.726	0.232
sx-mathoverflow	Y	24818	506550	User comment	0.785	1	4580.59	21.272	0.378
tntp-ChicagoRegional	Y	12982	39018	Road	0.15	1	353.439	0.133	0.863
topology	N	34761	171403	Internet AS connection	0.529	1	12345.65	0.104	0.48
twin	N	14274	20573	City twinning	0.587	0	1614.989	220.383	0.685
wiki_talk_gl	Y	8097	63809	User message	0.216	0	390.079	27.95	0.458
wiki_talk_sv	Y	120833	598066	User message	−0.041	0	6255.868	1.661	0.865
wikipedia_link_gan	Y	9189	176051	Article wikilink	0.616	0.755	3412.184	3.341	0.524
wikipedia_link_mg	Y	125916	1025610	Article wikilink	0.321	1	84490.496	0.954	0.39
wikisigned-k2	Y	138592	740397	User interaction	0.558	0	18733.197	312.286	0.489
wordnet-words	N	146005	656999	Word relationship	0.485	1	32367.14	93.729	0.811

The fields in the table are the dataset name, whether the network is directed, the number of nodes (N), the number of edges (m), the meaning of the edges, the Spearman correlation between the logarithm of the in-degree and the logarithm of the harmonic centrality (CHD), the goodness-of-fit (GOF) p-value of the fitted power-law model, the mean (across network nodes) of relative errors (mRE), the standard deviation (across network nodes) of relative errors (sdRE), and the percentile of the median of QuickCent MAE within the distribution of MAE of all the 4 machine learning methods shown in Sect. 4.2 (percMED). The name corresponds to the *Internal name* field in the KONECT database. GOF is computed along 200 bootstrap iterations where the lower limit search space is that from Sect. 3.2. mRE and sdRE are the means over 200 QuickCent fits associated to distinct random seeds, each one built from a training set of size of 25%. A similar procedure was used to compute percMED, where the training size used was 12.5%. Several models with distinct training sizes were executed, which can be reviewed with the code supplied [46] and do not change the result trends reported in this section. All QuickCent models reported in this table were obtained with a length of the proportions vector equal to 2, except for network wikipedia_link_gan that used a length of 1 their discussion. Section 4.5.1 shows how the QuickCent assumptions impact differently on the central tendency and variability of relative errors of the heuristic, and discusses how preferential attachment may leverage this. Section 4.5.2 studies the factors affecting the performance of QuickCent relative to other methods, showing suggestive evidence regarding the key role of the monotonic map of in-degree and harmonic centrality for general networks. Finally, Sect. 4.5.3 shows suggestive evidence that information networks such as citations or the internet would provide an optimal context for QuickCent, and how this is related to the preferential attachment mechanism.

4.5.1 Ecological rationality of QuickCent

The merit of QuickCent is analyzed by studying its relative errors, both the mean of relative errors within a network (mRE), and their standard deviation (sdRE). The first index reflects the general volume of errors in relation to the actual centrality values, while the second is their uniformity across nodes in the network. We have found that mRE has a Spearman correlation of −0.397 with *p*-value 0.013 to CHD, which in turn is the Spearman correlation of the (log) harmonic centrality to the (log) degree. That is, the better the monotonic relationship between degree and harmonic, the smaller the mean of relative errors, which may be interpreted as the role of the first assumption of QuickCent. On the other hand, sdRE presents a Spearman correlation of −0.355 with *p*-value 0.026 to GOF, the goodness-of-fit *p*-value of the power-law fit. This means that, the better the power-law fit, or as explained in Sect. 3.2, the greater GOF is, the more uniform the relative error is within a network. As seen in Sect. 4.2 from the performance of linear models, it is not trivial to get a uniform performance across the entire spectrum of a power-law, and it makes perfect sense that this is achievable by models with a good fit. The significance of these correlations is important, and the fact that they are obtained on an arbitrary selection of empirical networks, speaks that the adequacy or the ecological rationality of QuickCent is a structural feature of it.

These results may open the discussion on why QuickCent has the best performance on preferential attachment networks (Sect. 4.2). The simulations we display in SI Section 1 suggest a clear scale-free distribution of the harmonic centrality on these networks, which may be considered another result of this work. We do not know of any result describing the behavior of harmonic centrality on digraphs as, for example, those known for PageRank and the in-degree [66]. However, the previous correlations suggest that the relationship between in-degree and harmonic may have a stronger effect on the performance. There is converging evidence showing that preferential attachment, which in its usual formulation requires global information about the current degree distribution, can be the outcome of link-creation processes guided by the local network structure, such as a random walk adding new links to neighbors of connected nodes, or in simple words, meeting friends of friends [32, 67]. The reason is that the mechanism of choosing a neighbor of a connected node makes those higher-degree nodes more likely to be chosen by the random walk, which in turn makes more paths lead to them. That is, the local density could indeed reflect the access to larger parts of the network. Of course, preferential attachment is not the only mechanism capable of producing scale-free networks [68–70], and the distinct generative mechanisms may engender or not, a stronger relationship between density and size in the resulting network. This insight may be the reason why the monotonic relationship between harmonic centrality and in-degree is more apparent in the preferential attachment model than in some empirical networks.

4.5.2 Competitiveness of QuickCent

Up to this point, we have studied the performance of QuickCent by itself. In order to conclude its performance on empirical networks in comparison to other machine learning methods, we have proposed to compute the percentile of QuickCent MAE median, with respect to the distribution of errors of all the 4 methods shown in Sect. 4.2. This distribution works as a rough approximation of the error any (randomly sampled) learning method would make on this task of approximating the harmonic centrality by using knowledge of the in-degree. Thus, a low percentile of the QuickCent MAE median signals a competitive performance of these heuristics in relation to other methods, and vice versa.

We have obtained that the Spearman correlation of percMED to GHD is −0.221 with a *p*-value of 0.176, while its correlation to GOF is 0.015 with a *p*-value of 0.929. From these values, one can see that the key variable giving a competitive performance to QuickCent is the monotonic map from in-degree to harmonic. The differential influence of GHD versus GOF may be an expression of the fact that, since power-law distributions may suit several possible distributions, the QuickCent assumption of the monotonic map imposes a stronger requirement on the input datasets. Finally, the fact that the structure of QuickCent is reflected in its performance relative to other methods, may be an expression of the *no-free-lunch* theorem, which states on one of its formulations that at root, how well any algorithm performs is determined by how well it is aligned with the distribution that governs the problems on which that algorithm is run, rather than the operation of the algorithm itself [71].

4.5.3 Ideal networks for QuickCent

The insight described before about the monotonic map between the in-degree and the harmonic centrality on networks generated by preferential attachment (PA), as well as the importance of this assumption for the QuickCent advantage, nurtures the conjecture that QuickCent may be best suited to empirical networks better described by PA growth. Such networks are, for example, information networks such as the Internet or citations, more than pure social networks such as friendships [31, 36, 67]. There is evidence that, if one assumes that some nodes to form links are found uniformly at random, while others are found by searching locally through the current structure of the network, it turns out that the more pure social networks appear to be governed largely by random meetings, while others like the World Wide Web and citation networks involve much more network-based link formation [67].

We have performed a basic test of this hypothesis by building an indicator variable that is equal to 1 if a network of our dataset is an information network and 0 otherwise. It is not straightforward to define whether a given network is informational or not. We have taken the simple approach of discriminating the networks based on the network categories supplied by the very KONECT database. Thus, we consider information networks those belonging to the categories Software (subelj_jdk, subelj_jung-j, linux), Hyperlink (wikipedia_link_gan, wikipedia_link_mg, dimacs10-polblogs), Computer (p2p-Gnutella05, p2p-Gnutella31, topology), Citation (cit-HepPh, dblp-cite, subelj_cora) and Online contact (elec, wikisigned-k2, sx-mathoverflow). With these definitions, information networks exhibit a mean of percMED of 0.535, and 0.651 for non-information networks. The Kruskal–Wallis p-value, of the null hypothesis that the location parameters of the distribution are the same in each group, is 0.094. Better or worse p-values may be obtained by playing with categories that are in the limit of being considered informational. For example, if communication networks (facebook-wosn-wall, wiki_talk_gl, wiki_talk_sv) are also informational, now the same results are 0.544, 0.661 and 0.081. While this hypothesis would require more datasets to be confirmed, these results deliver suggestive evidence that this conjecture is correct.

Finally, we have also examined if there are any QuickCent performance differences between directed and non-directed networks. If we create an indicator vector whose *ith*–coordinate is equal to 1 iff the respective *ith*-network is non-directed, we get that the mRE mean is 8251.384 for non-directed networks, and 6622.454 for directed networks, with a Kruskal–Wallis p-value of 0.135. This suggests an overall better performance of QuickCent on directed networks, which is the framework adopted in this work, that is, using the in-degree, in contrast to our previous related work where the degree was used [18]. This feature, together with changes in some technical details of the heuristics (see Sect. 3.4), may explain the better performance on linear PA networks obtained in the current work, in comparison to the previous work. The indicator vector we defined previously also presents a Spearman correlation to CHD of −0.248 with a p-value of 0.128, pointing to the pattern that the monotonic map from degree to harmonic may be weaker in non-directed networks, in relation to directed ones. However, in this dataset of 39 networks, the number of directed and non-directed networks are not balanced, and these conclusions should be taken with care.

5 Future work

Applying QuickCent to other types of networks, such as bipartite, or centrality measures is not a direct task, since, depending on the type of network considered, degree and centrality may be strongly or weakly related. We plan to address these extensions in future work, where research on possible improvements achievable by tuning the vector of proportions may be addressed. One possible line of research is to formulate the problem of finding the proportion quantiles as that of obtaining an optimal quantizer [72]. There is some resemblance between our problem of finding the quantiles minimizing the error with respect to some distribution and that of finding the optimal thresholds of a piecewise constant function minimizing the distortion error of reproducing a continuous signal by a discrete set of points. On the other hand, future work should deal with extensions and flexibility of the clues employed, trying other clues or new ways to integrate different clues. The idea raised in our work of using a local density measure to approximate expensive size-based centrality indices could be generalized to be valid on more general networks, for example, by using a more general notion of local density than the in-degree, such as a modified degree measure to ensure minimum overlapping between spreading regions [33], or spreading indices based on the degree and neighbors' degree [34]. Some of these ideas may add more size-based information to the local density measures.

In the literature, there is previous work specifically tailored to centrality estimation using ML methods, but for other centrality indices beyond harmonic centrality. In particular Brandes and Pich study specific estimations for *closeness* and *betweenness* centrality [26]. It would be interesting to compare our method with the one proposed by Brandes and Pich [26], but this would amount to changing and adapting their method to harmonic centrality. This kind of approach may benefit from exploiting the kind of network patterns that QuickCent leverages. It also would be interesting to study different centrality indices in a QuickCent-like method, according to their axiomatic characterization of centrality [12], that is, taking into account their differential sensitivity to local density or size, and how this impacts the estimation performance against the network type where the approximation is applied. We leave this adaptation and further comparison as future work.

6 Conclusion

The results of this paper are a proof of concept to illustrate the potential of using methods based on very simple heuristics to estimate some network centrality measures. Our approach shares similarities with prior methods that rank nodes by utilizing statistical regularities found in scale-free networks, but differs in that it calculates the actual centrality values of nodes, rather than their rankings. Our results show that QuickCent is comparable in accuracy to the best-competing methods tested, with the lowest error variance, even when trained on a small proportion of the dataset, and all of this at intermediate time cost relative to the other methods using a naive implementation. We give some insight into how QuickCent exploits the fact that in some networks, such as those generated by preferential attachment, local density measures, such as the in-degree, can be a good proxy for the size of the network region to which a node has access, opening up the possibility of approximating expensive indices based on size such as the harmonic centrality. This same fact may explain some evidence we provide that QuickCent would have superior performance on empirical information networks, such as citations or the internet.

7 Supplementary information

This article has an accompanying supplementary document, which includes the experiments for checking the fulfillment of assumptions of the heuristics by the distinct networks shown in the Results section. It also shows the results of experiments on the robustness of QuickCent on the distinct regimes of the Preferential Attachment network model.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00607-024-01303-z.

Author Contributions All authors contributed to the study conceptualization, formal analysis, methodology, resources, validation, and writing. F.P. additionally performed data curation, investigation, visualization, and software implementation. J.P. and A.A. contributed supervision and project administration. F.P. and A.A. contributed with funding acquisition.

Funding This work was supported in part by the National Center for Artificial Intelligence CENIA [FB210017, Basal ANID], and by the National Agency of R+D ANID doctoral studies scholarship [CONICYT-PCHA/Doctorado Nacional/2016-21161085]. The funding sources played no role in the research.

Data availability All datasets and codes used in this work are publicly available under the following DOI and private link (for review purposes). If the paper is accepted, the repository produces a new public link, and the DOI is published. 10.6084/m9.figshare.25055234 https://figshare.com/s/eeb68dab2554021ceab9

Declarations

Conflict of interest The authors have no Conflict of interest to declare that are relevant to the content of this article.

References

1. Tversky A, Kahneman D (1974) Judgment under uncertainty: heuristics and biases: biases in judgments reveal some heuristics of thinking under uncertainty. Science 185(4157):1124–1131
2. Gigerenzer, G., Todd, P.M., Group, A.R. (1999) Simple heuristics that make us smart. Oxford UP, New York
3. Katsikopoulos KV, Schooler LJ, Hertwig R (2010) The robust beauty of ordinary information. Psychol Rev 117(4):1259
4. Backlund LG, Bring J, Skånér Y, Strender L-E, Montgomery H (2009) Improving fast and frugal modeling in relation to regression analysis: test of 3 models for medical decision making. Med Decis Making 29(1):140–148
5. Scheibehenne B, Bröder A (2007) Predicting wimbledon 2005 tennis results by mere player name recognition. Int J Forecast 23(3):415–426
6. Snook B, Zito M, Bennell C, Taylor PJ (2005) On the complexity and accuracy of geographic profiling strategies. J Quant Criminol 21(1):1–26
7. Wasserman S, Faust K (1994) Social network analysis: methods and applications. Cambridge University Press, Cambridge
8. De Arruda GF, Barbieri AL, Rodriguez PM, Rodrigues FA, Moreno Y, da Fontoura Costa L (2014) Role of centrality for the identification of influential spreaders in complex networks. Phys Rev E 90(3):032812
9. Sziklai BR, Lengyel B (2022) Finding early adopters of innovation in social networks. Soc Netw Anal Min 13(1):4
10. de Arruda GF, da Fontoura Costa L, Schubert D, Rodrigues FA (2014) Structure and dynamics of functional networks in child-onset schizophrenia. Clin Neurophysiol 125(8):1589–1595
11. Marchiori M, Latora V (2000) Harmony in the small-world. Physica A 285(3–4):539–546
12. Boldi P, Vigna S (2014) Axioms for centrality. Internet Math 10(3–4):222–262
13. Pettie S, Ramachandran V (2002) Computing shortest paths with comparisons and additions. In: Proceedings of the thirteenth annual ACM-SIAM symposium on discrete algorithms, pp 267–276

14. Pettie S (2002) On the comparison-addition complexity of all-pairs shortest paths. In: International symposium on algorithms and computation. Springer, pp 32–43
15. Planken LR, de Weerdt MM, van der Krogt RP (2012) Computing all-pairs shortest paths by leveraging low treewidth. J Artif Intell Res 43:353–388
16. Hertwig R, Hoffrage U, Martignon L (1999) Quick estimation: letting the environment do the work. In: Simple heuristics that make us smart. Oxford University Press, pp 209–234
17. Newman ME (2005) Power laws, pareto distributions and zipf's law. Contemp Phys 46(5):323–351
18. Plana F, Pérez J (2018) Quickcent: a fast and frugal heuristic for centrality estimation on networks. In: 2018 IEEE/WIC/ACM international conference on web intelligence (WI). IEEE, pp 238–245
19. Saxena A, Gera R, Iyengar S (2017) Fast estimation of closeness centrality ranking. In: Proceedings of the 2017 IEEE/ACM international conference on advances in social networks analysis and mining, pp 80–85
20. Saxena A, Gera R, Iyengar S (2019) A heuristic approach to estimate nodes' closeness rank using the properties of real world networks. Soc Netw Anal Min 9(1):3
21. Saxena A, Malik V, Iyengar S (2015) Rank me thou shalln't compare me. arXiv:1511.09050
22. Saxena A, Malik V, Iyengar S (2015) Estimating the degree centrality ranking of a node. arXiv:1511.05732
23. Rattigan MJ, Maier M, Jensen D (2006) Using structure indices for efficient approximation of network properties. In: Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining, pp 357–366
24. Pfeffer J, Carley KM (2012) k-centralities: local approximations of global measures based on shortest paths. In: Proceedings of the 21st international conference on world wide web, pp 1043–1050
25. Eppstein D, Wang, J (2001) Fast approximation of centrality. In: Proceedings of the twelfth annual ACM-SIAM symposium on discrete algorithms. Society for Industrial and Applied Mathematics, pp 228–229
26. Brandes U, Pich C (2007) Centrality estimation in large networks. Int J Bifurc Chaos 17(07):2303–2318
27. Chan SY, Leung IX, Liò P (2009) Fast centrality approximation in modular networks. In: Proceedings of the 1st ACM international workshop on complex networks meet information & knowledge management. ACM, pp 31–38
28. Cohen E, Delling D, Pajor T, Werneck RF (2014) Computing classic closeness centrality, at scale. In: Proceedings of the second ACM conference on online social networks, pp 37–50
29. Chechik S, Cohen E, Kaplan H (2015) Average distance queries through weighted samples in graphs and metric spaces: high scalability with tight statistical guarantees. arXiv:1503.08528
30. Barabási A-L, Albert R (1999) Emergence of scaling in random networks. Science 286(5439):509–512
31. Jeong H, Néda Z, Barabási A-L (2003) Measuring preferential attachment in evolving networks. EPL (Europhys Lett) 61(4):567
32. Vázquez A (2003) Growing network with local rules: preferential attachment, clustering hierarchy, and degree correlations. Phys Rev E 67(5):056104
33. Kumar S, Lohia D, Pratap D, Krishna A, Panda B (2022) Mder: modified degree with exclusion ratio algorithm for influence maximisation in social networks. Computing 104(2):359–382
34. Berahmand K, Bouyer A, Samadi N (2019) A new local and multidimensional ranking measure to detect spreaders in social networks. Computing 101:1711–1733
35. Bröder A, Newell B (2008) Challenging some common beliefs: empirical work within the adaptive toolbox metaphor. Judgm Decis Mak 3(3):205
36. Broido AD, Clauset A (2019) Scale-free networks are rare. Nat Commun 10(1):1–10
37. Barabási A-L (2018) Love is all you need: Clauset's fruitless search for scale-free networks. Blog post available at https://www. barabasilab.com/post/love-is-all-you-need 20
38. Krapivsky PL, Redner S, Leyvraz F (2000) Connectivity of growing random networks. Phys Rev Lett 85(21):4629
39. Clauset A, Shalizi CR, Newman ME (2009) Power-law distributions in empirical data. SIAM Rev 51(4):661–703
40. Clauset A, Young M, Gleditsch KS (2007) On the frequency of severe terrorist events. J Conflict Resolut 51(1):58–87
41. Gillespie CS (2015) Fitting heavy tailed distributions: the poweRlaw package. J Stat Softw 64(2):1–16

42. Drees H, Janßen A, Resnick SI, Wang T (2020) On a minimum distance procedure for threshold selection in tail analysis. SIAM J Math Data Sci 2(1):75–102
43. Voitalov I, van der Hoorn P, van der Hofstad R, Krioukov D (2019) Scale-free networks well done. Phys Rev Res 1(3):033034
44. Hosseini R (2010) Quantiles equivariance. https://doi.org/10.48550/arXiv.1004.0533
45. von Helversen B, Rieskamp J (2008) The mapping model: a cognitive theory of quantitative estimation. J Exp Psychol Gen 137(1):73
46. Plana F (2024) Quickcent paper data and code figshare. https://doi.org/10.6084/m9.figshare.25055234
47. Bishop CM (2006) Pattern recognition and machine learning, 1st edn. Springer, New York
48. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2020). R Foundation for Statistical Computing. https://www.R-project.org/
49. Csardi G, Nepusz T (2006) The igraph software. Complex Syst 1695:1–9
50. Wickham H (2009) Ggplot2: elegant graphics for data analysis. Springer, New York
51. Quinlan JR, et al. (1992) Learning with continuous classes. In: 5th Australian joint conference on artificial intelligence, vol 92. Singapore, pp 343–348
52. Witten IH, Frank E (2005) Data mining: practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco
53. Rumelhart DE, McClelland JL, Group PR et al (1988) Parallel distributed processing, vol 1. MIT Press, Cambridge
54. Hornik K, Buchta C, Zeileis A (2009) Open-source machine learning: R meets Weka. Comput Stat 24(2):225–232. https://doi.org/10.1007/s00180-008-0119-7
55. Brighton H, Gigerenzer G (2015) The bias bias. J Bus Res 68(8):1772–1784
56. Cormen TH, Leiserson CE, Rivest RL, Stein C (2001) Introduction to algorithms, 2nd edn. MIT Press, Cambridge
57. Maslov S, Sneppen K (2002) Specificity and stability in topology of protein networks. Science 296(5569):910–913
58. Li L, Alderson D, Willinger W, Doyle J (2004) A first-principles approach to understanding the internet's router-level topology. In: ACM SIGCOMM computer communication review, vol 34. ACM, pp 3–14
59. Zhang L, Small M, Judd K (2015) Exactly scale-free scale-free networks. Physica A 433:182–197
60. Erdös P, Rényi A (1959) On random graphs, i. Publicationes Mathematicae (Debrecen) 6:290–297
61. Bollobás B (1981) Degree sequences of random graphs. Discret Math 33(1):1–19
62. Karp RM (1990) The transitive closure of a random digraph. Random Struct Algorithms 1(1):73–93
63. Kunegis J (2013) Konect: the koblenz network collection. In: Proceedings of the 22nd international conference on world wide web, pp 1343–1350
64. Adamic LA, Glance N (2005) The political blogosphere and the 2004 us election: divided they blog. In: Proceedings of the 3rd international workshop on link discovery, pp 36–43
65. Šubelj L, Bajec M (2012) Software systems through complex networks science: review, analysis and applications. In: Proceedings of the first international workshop on software mining, pp 9–16
66. Litvak N, Scheinhardt WR, Volkovich Y (2007) In-degree and pagerank: why do they follow similar power laws? Internet Math 4(2–3):175–198
67. Jackson MO, Rogers BW (2007) Meeting strangers and friends of friends: how random are social networks? Am Econ Rev 97(3):890–915
68. Krioukov D, Papadopoulos F, Kitsak M, Vahdat A, Boguná M (2010) Hyperbolic geometry of complex networks. Phys Rev E 82(3):036106
69. Zhou T, Yan G, Wang B-H (2005) Maximal planar networks with large clustering coefficient and power-law degree distribution. Phys Rev E 71(4):046141
70. Doye JP (2002) Network topology of a potential energy landscape: a static scale-free network. Phys Rev Lett 88(23):238701
71. Wolpert DH (2013) Ubiquity symposium: evolutionary computation and the processes of life: what the no free lunch theorems really mean: how to improve search algorithms. Ubiquity 2013(December):1–15
72. Gray RM, Neuhoff DL (1998) Quantization. IEEE Trans Inf Theory 44(6):2325–2383