Chapter 7

Community Validation Metrics and Evaluation Methodologies

This chapter focuses on community evaluation. Explores properties of social community formation for designing validation metrics in order to ensure accuracy alternatively through quality measure. Studies relative inclination of community detection algorithms towards accuracy. Exploits quantile-quantile plot and linear regression analysis in order to deal with output variation of community detection algorithms.

7.1 Introduction

Evaluation of community detection algorithms is important for ensuring both accuracy and quality of identified communities. Various metrics and methodologies have been developed for effective analysis of communities. Existing community evaluation confronts with three major issues: 1) how to assure accuracy if ground truth communities are not available, 2) how to deal with the trade-off between accuracy and quality if ground truth

communities are available, and 3) how to compare the performance if output of algorithms vary in different executions. In this chapter, aforementioned issues are addressed from the perspective of designing metrics and evaluation methodologies.

On the ground of evaluation, both quality and accuracy are important for communities. Quality of communities is measured by considering the connectivity among community members, whereas accuracy of communities is measured by comparing members of communities with ground truth. Real-world networks mostly do not have ground truths so accuracy cannot be measured for those networks. However, quality can be measured easily since it does not require ground truth. Thus, it will be advantageous if accuracy can be ensured alternatively via quality measure. Exploring this idea, a set of three quality metrics is proposed. Unlike most of the existing quality metrics that are developed based on dense connectivity, the proposed metrics are designed based on two properties of social community formation: *unification* and *isolation*. Competency of the proposed metrics in dealing with accuracy is analyzed on both real-world networks and synthetic networks. Moreover, these metrics satisfy all of the six quality related properties suggested by Van Laarhoven and Marchiori [196].

As mentioned above, measuring quality incorporates edges, while measuring accuracy involves node labels. This fundamental difference between the two measures has led to the trade-off between accuracy and quality. Trade-off between quality and accuracy is a major issue during performance evaluation of community detection algorithms. A framework is proposed to analyze Relative Inclination Towards accuracy (RITA) of a set of community detection algorithms. The framework of RITA analysis utilizes MCDM [12, 103, 219] technique to accumulate indications of various metrics into single score. RITA analysis involves both quality metrics and accuracy metrics in order to determine inclination of algorithms. It expresses how likely an algorithm will identify accurate communities in

comparison to other algorithms. RITA analysis is simple, it just requires to visually inspect the trend and height of the curves representing inclination of different algorithms. Effectiveness of RITA analysis in indicating relative inclination of community detection algorithms towards accuracy is analyzed on various real-world networks. With RITA analysis, the trade-off between quality metrics and accuracy metrics disappears during evaluation of communities.

Community detection algorithms that are of random nature identify different community structures in different execution of algorithms. Comparing performance of those community detection algorithms is challenging. Generally, various metric values obtained for identified communities are considered to compare the performance of algorithms. Existing analysis methods such as non-parametric analysis, where mean, median or standard deviation are computed against different metrics can only give overall information regarding distribution of the metric values. However, if the communities are good (or bad) then computing mean, median or standard deviation cannot express how good (or bad) are the communities in comparison to other algorithms. A visual analysis methodology is proposed to deal with *output variation* of community detection algorithms during evaluation. Key features of this methodology are as follows.

- Solutions obtained with different algorithms (in terms of specific metric) are considered to compare performance of algorithms.
- Proposed methodology utilizes the concepts of quantile-quantile plot and simple regression analysis. Quantile-quantile plot ensures involvement of each individual solution in the evaluation process.
- With quantile-quantile plot, three kinds of dominance is defined to express how good or bad are the solutions of one algorithm compared to other algorithms.

- To accumulate overall dominance of all points in the quantile-quantile plot and linear regression analysis is performed. Then regression line dominance and shifting mechanism is developed, and incorporated into analysis methodology.

- With proposed methodology, algorithms are compared in terms of dominance of one algorithm over other algorithms. The dominance of an algorithm is determined simply by observing the angle between regression line and neutral line and the position of intersection of regression line and neutral line.

7.2 Proposed Quality Metrics

In this section, the unification and isolation properties of social community formation through social interactions are discussed. Also, the designing of proposed quality metrics incorporating these two properties is explained.

7.2.1 Social Community Formation

In social network, personal interest is one of the stimulators for the interaction and communication between two persons. Those who share common or interrelated personal interest, have the greater tendency to communicate with each other [140]. Social connectivity grows based on the foundation of such social interaction [1, 28, 92]. Engagements with the interactions among different persons determine how strongly or loosely they will be connected or have relationship with each other. Such notion in the relationship is referred as *mutual interest* of both persons induced by their personal interests (detailed in chapter 3). Higher mutual interest yields stronger relationship or connection. The strength of such connections among the persons is represented graphically by assigning relative weights and persons are represented as objects. Strongly connected persons with each

other forms group or community by pursuing the notion of unification of persons with common personal interest. On the other hand, weakly connected persons or the persons having different personal interest detach from other persons by pursuing the notion of isolation. Similarly, group of persons having common personal interest also follows isolation from the persons having different personal interest. Thus, formation of community in a network requires to follow both unification and isolation properties. These two properties are stated in the context of communities as follows:

Unification: Unification is a property that unites the members of smaller communities into one community. Two communities are unified into single community if members are significantly connected.

Isolation: Isolation is a property that isolates the members of community from rest of the network and integrates the members of the community. This means that connectivity of community with rest of the network should be less and connectivity within the community should be high.

Extrapolating these two properties, a set of three quality metrics is defined to evaluate the communities predicted by any community detection algorithm. Designing of these metrics are discussed in the next section.

7.2.2 Unifiability

The unification property implies that there has a tendency to unify multiple communities into single community. Unifiability is the measure of such tendency for a community as a whole instead of single node. It incorporates the notion of unification of any community with other communities by considering the connections of all members of the community as a whole. Intuitively, the members of two different communities will have connections

at individual level if the network is connected. The collective strength of such connections will give the strength of the relationship between two communities. To unify two communities, the strength of relationship between the two has to be higher than their total strength of relationships with all communities. Therefore, Unifiability of community C_i with respect to another community C_j is measured as follows:

$$\text{Unifiability}(C_i, C_j) = \frac{\sum_{u \in C_i, v \in C_j} \delta(u,v)}{\sum_{u \in C_i, v \notin C_i} \delta(u,v) + \sum_{u \notin C_j, v \in C_j} \delta(u,v) - \sum_{u \in C_i, v \in C_j} \delta(u,v)} \quad (7.1)$$

where, $\delta(u,v)$ represents the strength of connection between any two nodes u and v. Numerator of the above equation is the total strength of common connections between communities C_i and C_j, and denominator is the total strength of external connections of both communities C_i and C_j. In the above definition, Unifiability for any community is measured with respect to another community, which is the ratio of connection strength between the two communities to the total strength of connections associated with all other communities for both the communities. For unweighted graph, strengths of connections are considered as 1. Hence, Unifiability of any community for unweighted graph is considered as a ratio of the number of connections between the two communities to the total number of connections that associated to both the communities with other communities. Thus, Unifiability is expressed in terms of connections for unweighted graph as follows:

$$\text{Unifiability}(C_i, C_j) = \frac{\{(u,v) | u \in C_i \ \& \ v \in C_j\}}{\{\{(u,x),(v,y)\} | u \in C_i, v \in C_j, x \notin C_i, y \notin C_i, y \notin C_j\}} \quad (7.2)$$

where, u and v are any two nodes of communities C_i and C_j respectively, x is the any neighbor node of community C_i, y is the any neighbor node of community C_j excluding

the nodes that belong to community C_i, the pairs (u,x) and (v,y) represents connection between nodes u to x and v to y respectively. Unifiability of community C_i with respect to all k communities of community structure C is obtained as follows:

$$\text{Unifiability}(C_i) = \sum_{j=1}^{k} \text{Unifiability}(C_i, C_j) \tag{7.3}$$

Unifiability of all communities together will represent Unifiability of the community structure. The overall effect of Unifiability corresponding to each community is obtained by averaging. For any predicted community structure C of graph G, that comprises k communities within it, Average Unifiability (AVU) is computed as follows:

$$Q_{AVU}(G,C) = \frac{1}{k} \sum_{i=1}^{k} \text{Unifiability}(C_i) \tag{7.4}$$

The range of AVU is established in the following theorem.

Theorem 7.1. *Range of AVU is [0, 1] for any connected network.*

Proof of Theorem 7.1 is given in Appendix A.1.

7.2.3 Isolability

Isolability of a community is defined by incorporating the notion of isolation and connection strength among the nodes of the network. In this context, nodes are examined by considering the strength of connections to measure overall ability of any community to isolate itself from rest of the network. It is named as Isolability because this measure determines the ability of any community to isolate itself from rest of the network. Isolability of any community C_i is defined as follows:

$$\text{Isolability}(C_i) = \frac{\sum_{u \in_{C_i} v} \delta(u,v)}{\sum_{u \in_{C_i} v} \delta(u,v) + \sum_{u \in C_i, v \notin C_i} \delta(u,v)} \tag{7.5}$$

where, u and v are any two nodes and $\delta(u,v)$ represents the strength of connection between nodes u and v. In the above definition, Isolability for any community is measured, which is the ratio of connection strength within the community to the total strength of connections associated with the community. For unweighted graph, strengths of connections are considered as 1. Hence, Isolability of any community for unweighted graph is considered as a ratio of the number of connections within a community to the total number of connections associated with that community. This instance is very similar to Relative Density measure [169], where the ratio of internal and the total node degree has been considered instead of connectivity. Simplified Relative Density appears same ratio as Isolability when all connection strengths are considered as 1. Hence, Relative Density can be treated as a special case of Isolability. Isolability of any community C_i for unweighted graph is defined simply in terms of connections as follows:

$$\text{Isolability}(C_i) = \frac{\{(u,v) | u \in_{C_i} v\}}{\{\{(u,v);(u,w)\} | u \in_{C_i} v \;\&\; w \notin C_i\}} \tag{7.6}$$

where, u and v are any two nodes belong to the cluster C_i, w is the node outside the community, the pairs (u,v) and (u,w) represents connection between nodes u to v and u to w respectively.

Measuring Isolability ensures higher connectivity among all nodes within the community. Similarly, Isolability of all communities present in the predicted community structure is measured relative to each other. Averaging Isolability of all communities yield overall connectivity of nodes corresponding to their respective communities. For any predicted

community structure C of graph G, that comprises k communities within it, Average Isolability (AVI) is computed as follows:

$$Q_{AVI}(G,C) = \frac{1}{k}\sum_{i=1}^{k} \text{Isolability}(C_i) \tag{7.7}$$

The range of AVI is established in the following theorem.

Theorem 7.2. *Range of AVI is [0, 1] for any connected network.*

Theorem 7.3. *In connected network, it is not possible to attain value 1 for both AVI and AVU at the same time.*

Proofs of Theorem 7.2 and Theorem 7.3 are given in Appendix A.1.

7.2.4 Balanced Isolability and Unifiability

AVI should be high as it indicates the ability of community to isolate itself from rest of the network. On the contrary, AVU should be low as it indicates the ability of community to unify itself with other communities. Communities should able to isolate themselves from rest of the network as much as possible. At the same time, the ability a community to unify with other communities has to be as less as possible. Therefore, AVI and 1/AVU have to be maximized to indicate good communities. Both are combined by giving equal weightage to each to obtain a single value. In that case, harmonic mean of AVI and 1/AVU will result in balanced effect of both. The notion of balancing is inspired by well established accuracy metric F-measure that combines the effect of precision and recall by taking harmonic mean of two. Both precision and recall have to be maximum for accurate clusters, which are balanced with harmonic mean. The balancing of AVI and 1/AVU is referred as Average of Unifiability and Isolability (AUI). The AUI for any community structure C of graph G is expressed as follows:

$$Q_{AUI}(G,C) = \frac{2}{\frac{1}{1/Q_{AVU}(G,C)} + \frac{1}{Q_{AVI}(G,C)}} \tag{7.8}$$

$$= \frac{2 \times Q_{AVI}(G,C)}{1 + Q_{AVU}(G,C) \times Q_{AVI}(G,C)} \tag{7.9}$$

The harmonic mean of AVI and 1/AVU gives the flexibility to equal contribution of both AVI and 1/AVU in maximizing the final value. Hence, AUI gives a balanced effect of both AVI and AVU. The range of AUI is established in following theorem.

Theorem 7.4. *Range of AUI will be [0, 2] for connected network.*

Proof of Theorem 7.4 is given in Appendix A.1.

With the Theorem 7.4, we have AUI values ranged from 0 to 2. However, in general, metric values are normalized to 1. Since, AUI can have maximum value 2 so AUI is normalized to 1 by simply dividing it by 2. Therefore, Average Normalized Unifiability and Isolability (ANUI) is expressed as follows:

$$Q_{ANUI}(G,C) = \frac{Q_{AUI}(G,C)}{2} \tag{7.10}$$

$$= \frac{Q_{AVI}(G,C)}{1 + Q_{AVU}(G,C) \times Q_{AVI}(G,C)} \tag{7.11}$$

The range of ANUI is established in following theorem.

Theorem 7.5. *Range of ANUI will be [0, 1] for any connected network.*

Proof of Theorem 7.5 is simple, since ANUI is half of AUI.

7.3 Relative Inclination Towards Accuracy

In this section, the MCDM process and the techniques utilized for measuring RITA of community detection algorithms are explained. Measuring RITA incorporates both kinds of metrics (i.e. accuracy and quality) and generates scores according to customized weights assigned to each category.

7.3.1 MCDM Process

The process takes a decision matrix as input, where rows represent multiple criterion (or metrics) and columns represent multiple alternatives (or algorithms). Thus, any entry M_{ij} in the decision matrix $M_{m\times n}$ is the value corresponding to the criterion i of alternative j, where m and n are the numbers of criterion and alternatives respectively. The relative scores of all the alternatives are the output of the process. Higher score indicates better alternative.

Numerous techniques [12, 103, 219] have been developed for generating relative scores of alternatives based on the values of different criterion. The technique for order preference by similarity to ideal solution (TOPSIS) [91] is considered for generating relative scores. TOPSIS has the privilege for assigning customized weights to each criteria. In TOPSIS, the decision matrix $M_{m\times n}$ is normalized for each criteria. An ideal alternative is prepared constituting best values of each criteria and a negative-ideal alternative prepared constituting worst values of each criteria from the normalized decision matrix $\hat{M}_{m\times n}$. TOPSIS generates score for each alternative by minimizing separation with the ideal alternative and maximizing separation with the negative-ideal alternative.

7.3.2 RITA Framework

Framework for measuring RITA is presented in the Algorithm 7.1. The framework is an extension of MCDM process, where TOPSIS technique is used for evaluating community detection algorithms. A decision matrix is prepared with all the algorithms (alternatives) and metrics considered for evaluation. Both accuracy and quality metrics are used in the evaluation process. The percentage of weight assigned to each category is customized. The weight is distributed equally among different metrics of each category.

Weights are assigned to each category in relative to other. Relative weight in the sense that if one category gets $X\%$ weights then other category gets $(100-X)\%$ of total weights. Weights assigned to each category is equally distributed over all metrics belong to that category. It means, if accuracy metrics are assigned weight ω_A and m number of metrics of this category are considered for RITA then weights are distributed to each accuracy metric a as follows:

$$\omega_a = \frac{1}{m} \times \omega_A \tag{7.12}$$

Similarly, ω_Q amounts of weight is distributed to all quality metrics, and each quality metric q of n quality metrics gets weight as follows:

$$\omega_q = \frac{1}{n} \times \omega_Q \tag{7.13}$$

where, $\omega_A + \omega_Q = \omega_T$ is total weight. Thus, if ω_A is $X\%$ of ω_T then ω_Q will be $(100-X)\%$ of ω_T. If $\omega_T = 1$, by incrementing X with d amount will raise ω_A to $(\omega_A + 0.01 \times d)$ and ω_Q will reduce to $(\omega_Q - 0.01 \times d)$.

Algorithm 7.1: RITA Procedure($M_{m\times n}, s, d, e, T, r$)

1: Input: $M_{m\times n}$, s : total starting weights, d : difference between two consecutive total weights, e : total ending weights, T : types of metrics i.e. {accuracy, quality}, r : percentage of weight for accuracy metric
2: Output: O : relative scores of alternatives, P : total percentages of accuracy metric
3: $a \leftarrow$ number of accuracy metrics
4: $q \leftarrow$ number of quality metrics
5: $W \leftarrow$ weights of all metrics
6: $y \leftarrow s$
7: $j \leftarrow 1$
8: **while** $y \leq e$ **do**
9: **for all** $M_{ij} \in M$ **do**
10: **if** $T_i = accuracy$ **then**
11: $W_i \leftarrow \frac{r\times y}{a}$
12: **else**
13: $W_i \leftarrow \frac{(1-r)\times y}{q}$
14: **end if**
15: **end for**
16: $P(j) \leftarrow y$ //single value of y
17: $O\{j\} \leftarrow TOPSIS(M_{m\times n}, W)$ //list of values returned by TOPSIS
18: $y \leftarrow y + d$
19: $j \leftarrow j + 1$
20: **end while**
21: **return** O, P

The relative weight assignment is done with the Algorithm 7.1 as follows. The algorithm is initialized with starting total weight s and ending total weight e. Initial percentage of weight for accuracy metric is set as r and change in percentage is set as d. The decision matrix $M_{m\times n}$ contains the value corresponding to the metric i of algorithm j, where m and n are the numbers of metrics and algorithms respectively. The T contains the type of each metric i.e. whether the metric is accuracy or quality metric. The relative weight to each metric is assigned with lines 9-15. Each iteration current total weight y is increased by d. The TOPSIS process generates relative scores of different community detection algorithms for the relative weight assignment to different metrics. At each iteration, corresponding to each set up of weight assignment TOPSIS process generates a set of scores for different algorithms. Thus, for k such set ups obtained with gradual increment or

decrement of $d\%$ we will get k set of scores. Thus, the Algorithm 7.1 will return P and O, where P contains the k percentages of weights assigned to accuracy metrics and O contains the relative scores of community detection algorithms corresponding to each of the k percentages in P.

Plotting of these k sets of scores corresponding to k percentages in 2-dimensional plot will express the bias of an algorithm towards quality or accuracy. The percentage of weights assigned to accuracy metrics is considered in x-axis for 2-dimensional plotting. This plot presents how an algorithm is inclined towards accuracy. If an algorithm is not inclined towards accuracy then it means that the algorithm is biased towards quality. How much an algorithm is biased towards quality relative to other algorithm is easily visualized in the plotting of O versus P obtained from the Algorithm 7.1.

7.4 Visual Analysis Methodology

In this section, visual analysis methodology for comparing multiple outputs of algorithms is detailed. Explained point dominance, regression line dominance and regression line shifting mechanism. Derived some important properties, and developed various comparison methodologies.

7.4.1 Point Dominance

Consider two sets of data $D_a = \{a_i | i = 1,2,3,...,r\}$ and $D_c = \{c_i | i = 1,2,3,...,r\}$. The set whose dominance is to be evaluated is referred as *actor*, while the set with respect to whom the dominance is evaluated is referred as *competitor*. Equal number of items are considered in both actor and competitor. Quantile-quantile plot is used, which is simply a plot of sorted D_a versus D_c. The x-axis is specified for plotting actor and y-axis for

plotting competitor. Each point (a_i, c_i) in the plot is a pair of $a_i \in D_a$ and $c_i \in D_c$. Point dominance at any point (a_i, c_i) is defined in reference to a neutral line.

Definition 7.6 (Neutral Line (NL))**.** A line $Y = mX + c$ is referred as NL if it goes through the origin (i.e. intercept $c = 0$) and gradient $m = 1$. The line is neutral in the sense that, at every point (x, y) in the NL $x = y$.

Larger values imply higher the dominance [1]. In reference to the NL, three kinds of possible dominance are defined at any point (a_i, c_i) in the quantile-quantile plot as follows:

Non-dominance: Dominance of actor over competitor at any point (a_i, c_i) that *lies on* the NL is referred as non-dominance since $a_i = c_i$ i.e. neither a_i dominates c_i nor c_i dominates a_i.

Actor-dominance: Dominance of actor over competitor at any point (a_i, c_i) that *lies above* the NL is referred as actor-dominance since $a_i > c_i$ i.e. a_i dominates c_i.

Competitor-dominance: Dominance of actor over competitor at any point (a_i, c_i) that *lies below* the NL is referred as actor-dominance since $a_i < c_i$ i.e. a_i is dominated by c_i.

In Figure 7.1, points P_1, P_2 and P_3 of the quantile-quantile plot are examples of non-dominance, actor-dominance and competitor-dominance respectively.

7.4.2 Regression Line Dominance

Simple Linear Regression (SLR) analysis derives linear relationship between two variables of the bivariate data set: explanatory and dependent variables. Point dominance

[1] Later on, smaller values are considered to have higher the dominance when deal with minimization problems.

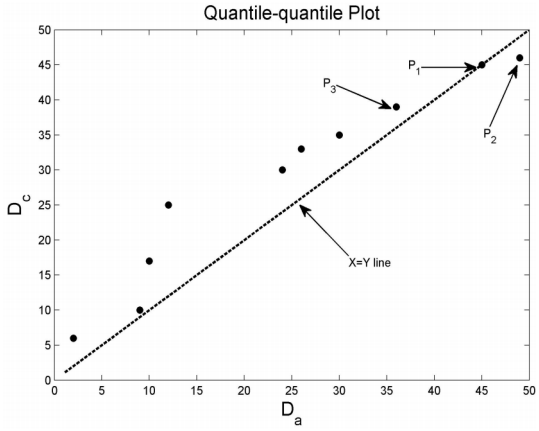


Figure 7.1: Point dominance in quantile-quantile plot.

discussed in the previous subsection incorporates two sets of data, where actor and competitor sets are presented in a quantile-quantile plot. With SLR analysis on the data points in quantile-quantile plot accumulates the point dominance by simply deriving linear relationship between actor set and competitor set. Actor set is considered as explanatory and competitor set is treated as dependent variable. Regression line (RL) is the representative for the relationship drawn between two variables with SLR.

Definition 7.7 (Regression Line (RL))**.** A linear regression line or simply regression line has an equation of the form $Y = mX + c$, where X is the explanatory variable and Y is the dependent variable. The gradient of the line m, and the intercept c are computed during SLR analysis of variables X and Y.

The RL obtained with SLR analysis on quantile-quantile plot preserves the overall dominance of actor set over competitor set. If point dominance changes, RL shifts accordingly to a new position. Various instances of RLs with incremental changing in the portions of D_a and D_c are shown in Figure 7.2. These instances of RLs are obtained by changing some of the points from actor dominant to competitor dominant. One can notice that with the rotational shifting of RL, intersecting point on NL also moves ahead towards end.

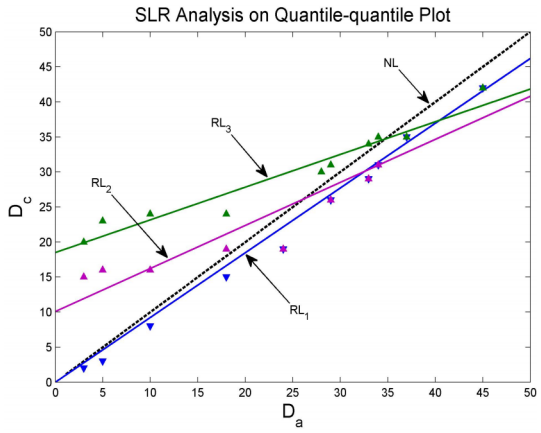


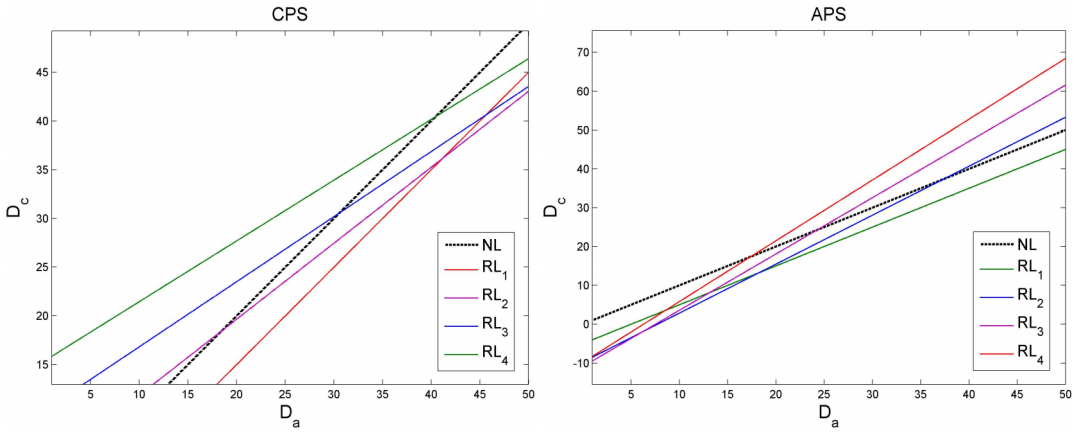
Figure 7.2: Regression Line dominance with Simple Linear Regression (SLR) analysis on quantile-quantile plot.

Hence at any instance, the dominance relation of D_a and D_c is determined by the angle and position of intersecting point between the RL and NL.

7.4.3 Regression Line Shifting

Depending on the dominance of actor set over competitor set, the shifting of RL is divided into two categories. If the shifting of RL implies increment in dominance of actor set over competitor set, it as referred as positive shift and, otherwise it referred as negative shift.

Positive Shift: Positive shifting of RL is achieved in two ways as discussed below. The discussions above have interpreted larger value implies higher dominance. However, for minimization problems, smaller value implies higher dominance so dominance is interpreted oppositely. Therefore, RL_1 in Figure 7.3a means worst since RL is below the NL and it is parallel to NL. RL_2 is obtained by changing some points in starting portion from actor dominant to competitor dominant i.e. some values in actor set have been improved. Hence, the shifting of RL from RL_1 to RL_2 is considered as positive. This rotational shifting is in clockwise direction so it is referred as Clockwise Positive Shift (CPS). Similarly,



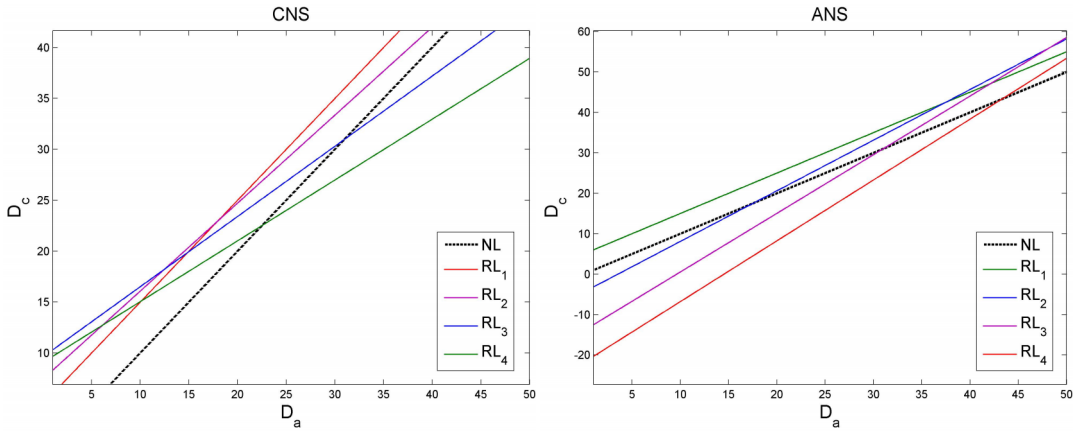
(a) Clockwise positive shift. (b) Anticlockwise positive shift.

Figure 7.3: Positive shifting of RL.

Note: Here, NL is the neutral line. RL_1, RL_2, RL_3 and RL_4 are the various instances of RL after successive incremental modification of (a) ***first*** m values of D_a and D_c, and (b) ***last*** m values of D_a and D_c.

if points of last portion are changed instead of starting it will also improve the values of actor set. However, shifting of RL in this case will be in anticlockwise direction as shown in Figure 7.3b, so it is referred as Anticlockwise Positive Shift (APS).

Negative Shift: Similar to positive shift, negative shift is achieved by changing dominance of points. In this case, RL above the NL and parallel to NL indicates all points are competitor dominant. Now, RL_1 in in Figure 7.4a will shift to RL_2 if some points from the last portion are changed to actor dominant. However, in the context of minimization problems, actually values of D_c that are dominated by D_a i.e. actor dominated means values of actor are worst. Hence, shifting from RL_1 to RL_2 indicates some best values (large) of actor set becomes worst i.e. negative of affect on actor set. This shifting is in clockwise direction so it is referred as Clockwise Negative Shift (CNS). Similarly, if points of starting portion are changed instead of last portion it will also degrade the values



(a) Clockwise negative shift. (b) Anticlockwise negative shift.

Figure 7.4: Negative shifting of RL.

Note: Here, NL is the neutral line. RL_1, RL_2, RL_3 and RL_4 are the various instances of RL after successive incremental modification of (a) ***last*** m values of D_a and D_c, and (b) ***first*** m values of D_a and D_c.

of actor set. However, shifting of RL in this case will be in anticlockwise direction as shown in Figure 7.4b, so it is referred as Anticlockwise Negative Shift (ANS).

Following theorems derive ranges of angle between RL and NL in the context of both kinds of shifting discussed above. The area that is covered either by positive or by negative shift of RL in reference to NL is established.

Theorem 7.8. *For minimization problems, the CPS or CNS can always have greater than 315 degree angle between RL and NL .*

Theorem 7.9. *For minimization problems, the APS or ANS can always have less than 45 degree angle between RL and NL.*

Theorem 7.10. *The maximum positive or negative rotational shift of RL is 90 degree and rotational area covers half of 1st quadrant and half of 4th quadrant with respect to NL.*

Proofs of Theorem 7.8, Theorem 7.9 and Theorem 7.10 are given in Appendix A.2.

7.4.4 Comparative Analysis Methodology

This section elaborates the detail about implication of regression line shifting mechanism with respect to NL for comparing performance of algorithms. With the same notion as discussed above, the algorithm whose performance has to be evaluated is termed as *acting algorithm*, and with respect to whom the performance of acting algorithm is evaluated is referred as *competing algorithm* . To evaluate performance of acting algorithm over competing algorithm based on solution quality, best solutions generated by the algorithms in multiple trials are considered. Dominance of of acting algorithm over competing algorithms is emphasized with clockwise and anticlockwise shifting of RL with respect to NL. Solution quality of algorithms are evaluated in following three ways. First, the one-to-one comparison, where performance of a acting algorithm is compared with only one competing algorithm at a time. Second, the one-to-many comparison, where performance of a acting algorithm is compared with multiple competing algorithms. At the last, many-to-many comparison, where ranking of acting algorithm is done based on performance with other competing algorithms. These three comparisons are explained below in the context of Evolutionary Optimization Algorithm (EOA)s by assuming that the objective function is a minimization problem.

One-to-one Comparison

To compare performance of an acting algorithm A over competing algorithm C on the basis of best solutions obtained over t trials are used to generate RL by incorporating the methodology illustrated above. Note that single RL is generated for this instance. In the above discussion, it has already been noticed that the same instance can be attained through either positive shift or negative shift. Thus, it is unpredictable for a single instance of RL whether the instance is attained through positive shift or negative shift. However,

Theorem 7.9 suggests that the angle between RL and NL in anticlockwise shifting (i.e. APS or ANS) is less than equal to 45 degree. Again, with Theorem 7.8 the angle is greater than equal to 315 degree in clockwise shifting (i.e. CPS or CNS). Therefore, simply by visualizing the angle it can be determined whether it is a clockwise shift or anticlockwise shift. From the perspective of performance, if detected shift is clockwise then smaller angle indicates acting algorithm A performs better than competing algorithm C. If detected shift is anticlockwise then larger angle indicates better performance of A. Along with the angle between RL and NL, another significant aspect of visual inspection is the position of intersection RL on NL. For clockwise shift, intersecting point is towards the end indicates better performance of A. While the intersecting point towards the beginning indicates better performance of A for anticlockwise shift.

One-to-many Comparison

The one-to-many comparison plays important role in deciding whether an algorithm is better than all other algorithms. Particularly, when a new algorithm is introduced, it has to be compared with existing state-of-the-art algorithms to prove its superiority. Consider any new algorithm A i.e. the acting algorithm, whose performance has to be evaluated with respect to other n competing algorithms $C_1, C_2, C_3, .., C_n$. As discussed above, best solutions obtained over t trials on an objective function are plotted in quantile-quantile plot for each (A, C_i) pair, and performed SLR analysis. Thus, there will be n different RLs for all pairs of acting and competing algorithms. Each RL represents dominance of acting algorithm A over respective competing algorithm C_i. To decide on dominance of acting algorithm A over any competing algorithm C_i, the intersecting point of RL on NL and angle with NL is noted. Depending on the significance of these two observations for the RL (discussed earlier), the performance of the acting algorithm A with respect to any competing algorithm C_i is decided.

Many-to-many Comparison

Comparing performance of n algorithms with each other or ranking them is a challenging task in the sense that performance of all algorithms vary with respect to other algorithms. One algorithm may perform better with respect to r algorithms and perform worst with respect to remaining $(n-r-1)$ algorithms. In this way, different sets of best performing and worst performing algorithms for each algorithm are obtained. Therefore, it is difficult to determine from the multiple possibilities that which of the algorithm exactly performs best or worst. Proposed methodology provides a very simplistic way to tackled this task.

The many-to-many comparison of n algorithms simply imply n one-to-many comparisons, where each of the algorithm is considered as acting algorithm, and remaining algorithms are competing algorithms. On the other hand, one-to-many comparison of n algorithms imply n one-to-one comparison. When one-to-one comparison is performed, the influence of competing algorithm in the dominance relationship of acting algorithm through the RL is actually determined. In that sense, to get combined influence of all n competing algorithms the influence of all algorithms has to be added. Thus, one-to-many comparison of n algorithms reduced to single one-to-one comparison, where one algorithm is considered as acting algorithm, and remaining $n-1$ algorithms are collectively considered as one single competing algorithm. Therefore, we have n one-to-one comparisons instead of one-to-many comparisons for many-to-many comparisons of n algorithms. For each of the n one-to-one comparisons, one of the n algorithms is considered as acting algorithm, and remaining $n-1$ algorithms are collectively considered as one comparing algorithm. The collective influence of all $n-1$ competing algorithms is obtained as follows. Consider n algorithms $A_1, A_2, A_3, .., A_n$ and suppose, A_m is the acting algorithm and all remaining $A_i, i \neq m$ algorithms are competing algorithm. Consider $D = \{D_i | i = 1, 2, .., n\}$ comprising the sets of best solutions of all algorithms for t is the number of trials, where $D_i = \{d_{ij} | j = 1, 2, .., t\}$ arranged in non-decreasing order. Influence of all $n-1$ competing

algorithms with respect to acting algorithm A_m is computed as follows:

$$D_c = \left\{ \sum_{i=1 \& i \neq m}^{i=n} d_{ij} | j = 1,2,..,t \right\} \tag{7.14}$$

The set D_c represents the collective influences of all $n-1$ algorithms and it is considered as single competing algorithm. For the acting algorithm A_m, the set of best solutions for t is the number of trials $D_m = \{d_{mj} | j = 1,2,..,t\}$. The D_m and D_c are plotted in quantile-quantile plot, and with SLR analysis obtained a RL. Besides this RL for A_m, there will be $n-1$ RLs for remaining $n-1$ algorithms.

With this methodology, the RL of best algorithm automatically acquires highest position in terms of RL position. Similarly, the RL of second best algorithm acquires the position below the best and so on up to the RL of worst algorithm. The acquiring of positions of RL depending on algorithm's performance can be understood with the following example. For instance, if algorithm A_m is the best algorithm that means most values in D_m will be dominated by the values of $D_i, i = \{1,2,..,n \ \& \ i \neq m\}$. This means that all D_i contains larger values than D_m and hence, their sum D_c will definitely contain the set of highest values. The, values in D_c will produce highest domination to D_m, attaining highest possible position of RL.

7.5 Theoretical Analysis of AVI, AVU and ANUI Metrics

Theoretical analysis is performed to prove theoretically that proposed metrics fulfill necessary criterion of being quality measure for communities. Six axioms or properties suggested by Van Laarhoven and Marchiori are considered for the axiomatic analysis. They have showed in their recent work [196] that the aforementioned six properties are enough

for being a good quality metric. Competency of AVI, AVU and ANUI in perspective of all six properties are proved theoretically. These properties are defined as follows:

First property is about independence of node identity. Quality Metric has to be dependent only on strengths of connections, not on the identity of nodes. It has to remain constant for a community structure even if nodes of actual graph are permuted.

Definition 7.11 (Permutation invariance)**.** A quality metric Q is permutation invariant if for all graphs $G = (V,E)$ and all isomorphisms $f : \{u \to u' | u \in V, u' \in V\}$ of G, it satisfies $Q(G,C) = Q(f(G), f(C))$. I.e. for any community structure of a graph, quality metric should remain same for all isomorphic forms of the graph.

Quality of communities has to be unaltered when distances or connection strengths are scaled uniformly. This implies that if quality metric for a community structure indicates good quality it should retain that indication even if distances are scaled.

Definition 7.12 (Scale invariance)**.** A quality metric Q is scale invariant if for all graphs $G = (V,E)$, all pairs of community structures C and D of G and all constants $\alpha > 0$, $Q(G,C) \leq Q(G,D)$ if and only if $Q(\alpha G,C) \leq Q(\alpha G,D)$. Here, $\alpha G \Rightarrow \delta(u,v) \mapsto \alpha\delta(u,v)$, $\forall (u,v) \in E$. This means all connection strengths of G are scaled with α factor.

Quality metric has to be non decreasing under monotonic consistent improvement. Strengthening internal connections of communities or weakening external connections of communities is referred as consistent improvement. For all such consistent improvements quality metric has to be non decreasing.

Definition 7.13 (Monotonicity)**.** A quality metric Q is monotonic if for all graphs $G = (V,E)$, all community structure C of G, all $\beta \geq 0$ and all consistent improvements $G \pm \beta$ of G, it satisfies $Q(G \pm \beta, C) \geq Q(G,C)$. Here, $G \pm \beta \Rightarrow \delta(u,v) + \beta \geq \delta(u,v)$ whenever $u \in_{C_i} v$ and $\delta(u,v) - \beta \leq \delta(u,v)$ whenever $u \notin_{C_i} v$ for all $C_i \in C$.

Local changes to a graph should have only local effect to the community structure. That means contribution of a single community to total quality should only depend on nodes within the community and neighbors of the community.

Definition 7.14 (Locality)**.** A quality metric Q is local if for all graphs $G = (V,E)$, all community structure C of G, all communities $C_i \in C$ and all changes G^* associated to C_i, it satisfies $Q(C_i,C) + Q(C \backslash C_i,C) = Q(G,C)$ if and only if $Q(C_i^*,C) + Q(C \backslash C_i,C) = Q(G^*,C)$. Here, $G^* \Rightarrow$ all kinds of changes associated to C_i. C_i^* represents all kind the affections in C_i.

Quality metrics should have the ability to make any community structure optimal by revising strengths of the connections. Main objective is to rule out trivial quality metrics. This is done by acquiring richness.

Definition 7.15 (Richness)**.** A quality metric Q is rich if for all graphs $G = (V,E)$ and all community structures C, there exist a graph $G' = (V,E')$ such that $argmax_C Q(G,C) = Q(G',C)$ and $\delta(u,v) \mapsto \delta'(u,v)$, where $\delta(u,v)$ is the strength of $(u,v) \in E$ and $\delta'(u,v)$ is the strength of $(u,v) \in E'$.

Last property is about continuity. Small change in the graph has to cause a smaller impact on quality metric. Quality metrics should not be affected significantly for any smaller modification to the actual graph.

Definition 7.16 (Continuity)**.** A quality metric Q is continuous if for all graphs $G = (V,E)$ and $\beta > 0$ there exist a $\gamma > 0$ such that for all graphs $G' = (V,E')$, if $\delta(u,v) - \beta < \delta'(u,v) < \delta(u,v) + \beta$ for some connections $(u,v) \in E$, then $Q(G',C) - \gamma < Q(G,C) < Q(G',C) + \gamma$ for all community structures C of G. Here, $\delta(u,v)$ and $\delta'(u,v)$ are the strengths of connection (u,v) in E and E' respectively.

Following theorems show that AVI, AVU and ANUI satisfy all of the six properties discussed above.

Theorem 7.17. *Quality metrics AVI, AVU and ANUI are permutation invariant.*

Theorem 7.18. *Quality metrics AVI, AVU and ANUI are scale invariant.*

Theorem 7.19. *Quality metrics AVI, AVU and ANUI are monotonic.*

Theorem 7.20. *Quality metrics AVI, AVU and ANUI are local.*

Theorem 7.21. *Quality metrics AVI, AVU and ANUI are rich.*

Theorem 7.22. *Quality metrics AVI, AVU and ANUI are continuous.*

Proofs of these Theorems are given in Appendix A.3.

7.6 Empirical Analysis of Proposed Metrics

7.6.1 Experimental Setup

The competency of proposed metrics is analyzed by comparing with four accuracy metrics NMI [184], ARI [157], Purity [126] and F-measure, and two quality metrics Modularity [143] and Coverage [21]. Four community detection algorithms HC-PIN [203], LICOD [97], Random Walk (RW) [183] and SCAN [216] are considered to obtain community structures. Three LFR benchmark graphs [109] with 5000, 6000 and 7000 nodes are considered, which are termed as LFR5K, LFR6K and LFR7K respectively. These graphs are generated with mixing parameter $\mu = 0.1$. Three real-world networks: Football [143], Karate [218] and GR-QC [112] are considered.

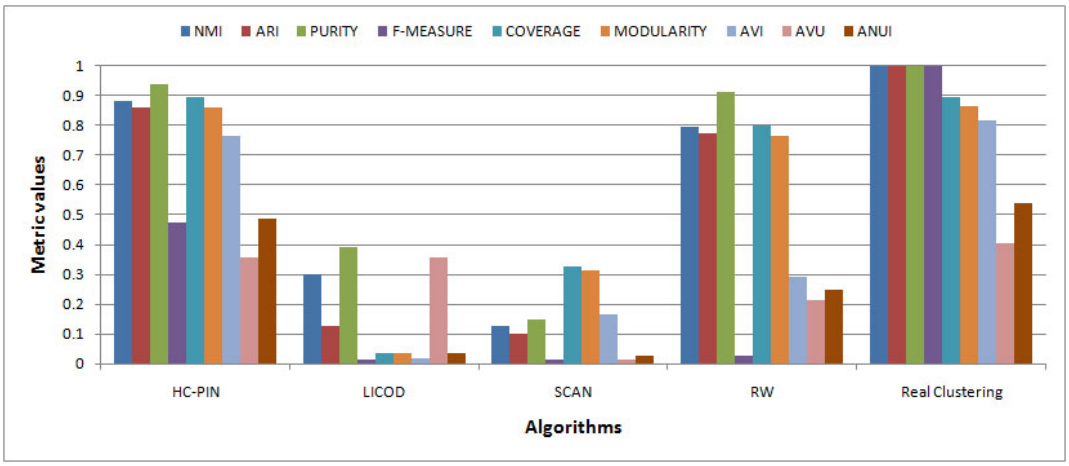
7.6.2 Metric Competitiveness Analysis

The objective of this analysis is to show that proposed metrics have the ability to ensure quality as well as accuracy. Quality and accuracy are two different aspects of the same

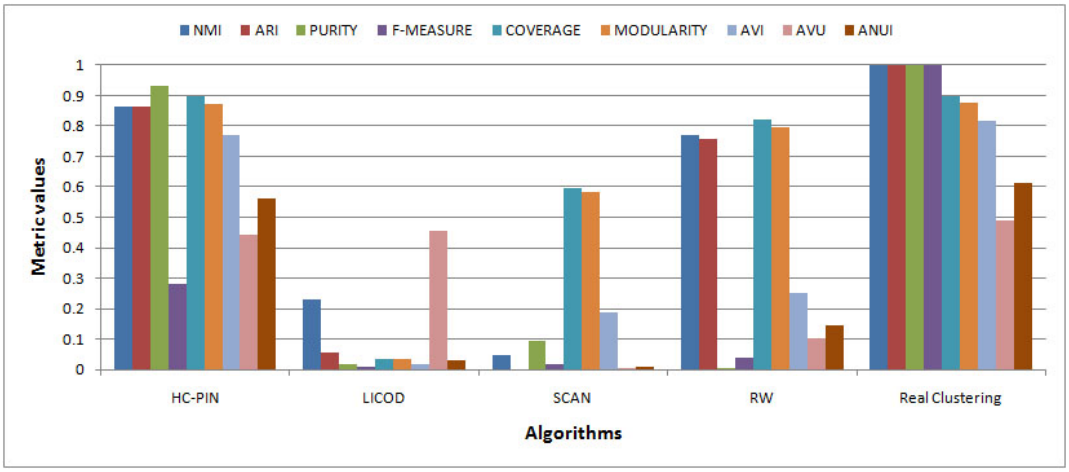
problem. Each aspect needs to be handled separately without loosing the significance of both. To verify the indications of any new metric require well-established quality and accuracy metrics. If both well-established quality and well-established accuracy metrics' indications are justified with the indications of proposed metrics then only one can say that the proposed metrics ensure both quality and accuracy. Hence, first indication of accuracy metrics are noted and compared with indications of proposed metrics. Then compared those indications that of quality metrics.

LFR Benchmark Graphs

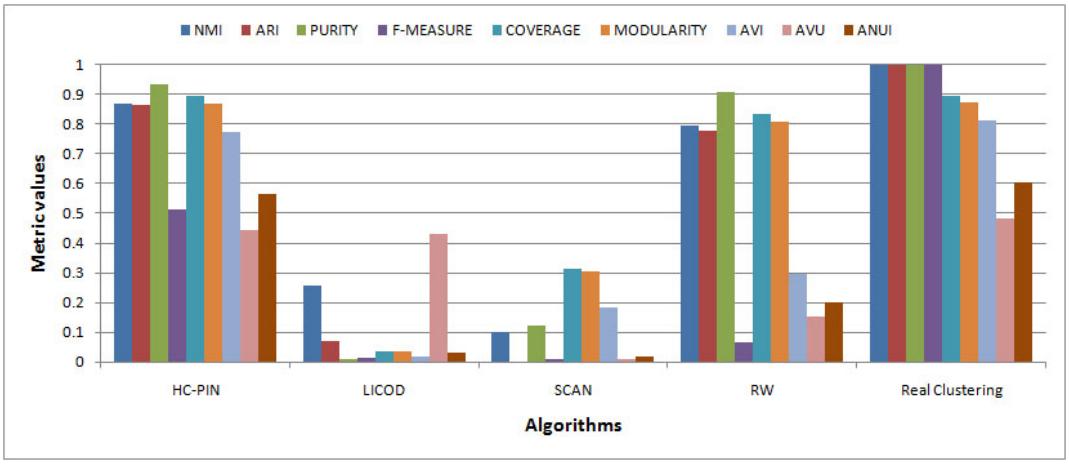
Metrics' values obtained for the communities predicted by different algorithms is verified with the metrics values obtained for real communities. Correctness of any metric and competitiveness with other metrics is examined through such verification. Considering accuracy indications of metrics NMI, ARI, Purity and F-measure as reference, the competitiveness of proposed metrics in indicating accuracy is compared with Modularity and Coverage. Results obtained on LFR benchmark graph are presented in Figure 7.5. All the accuracy metrics show highest value on LFR graphs as these metrics are evaluated over real communities. As indicated by NMI, ARI and Purity values, HC-PIN has been predicted as most accurate communities. F-measure values contradict level of accuracy indicated by NMI, ARI and Purity on all algorithms. This trade-off is very clear in case of RW, where F-measure as well as Purity show poor accuracy. Despite of such trade-offs Modularity and Coverage show very high values for the communities predicted by both HC-PIN and RW. Specifically, for the communities identified by RW on the LFR-6K graph, Modularity and Coverage show very high values, where Purity shows almost zero accuracy. This indicates Modularity and Coverage are incapable of dealing with the accuracy of communities. On the contrary, proposed metric ANUI can deal with such trade-off very smoothly. Considering the specific case of RW, one can notice the lower values of



(a) LFR-5K

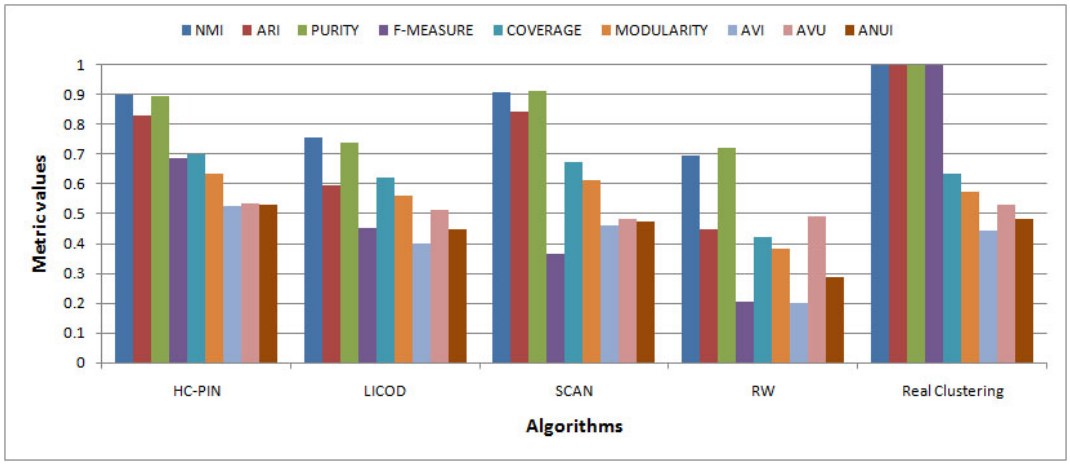


(b) LFR-6K

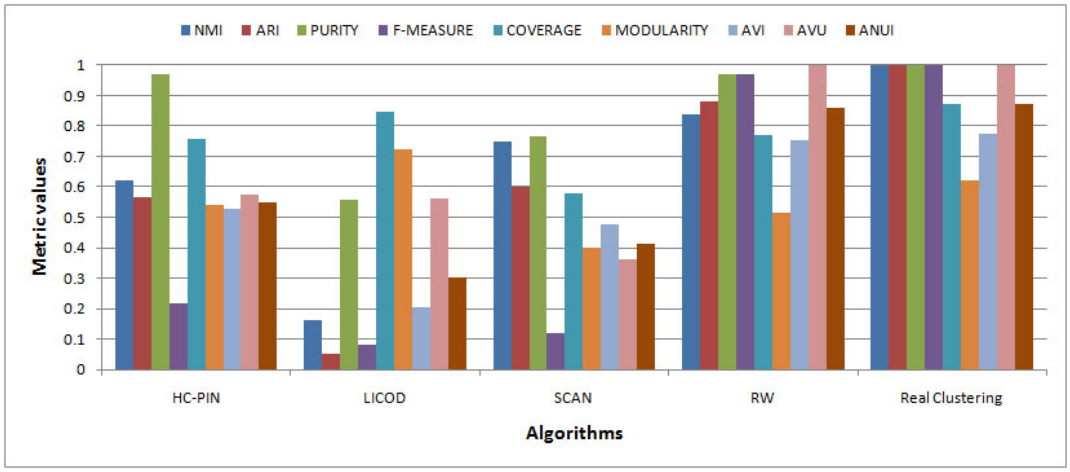


(c) LFR-7K

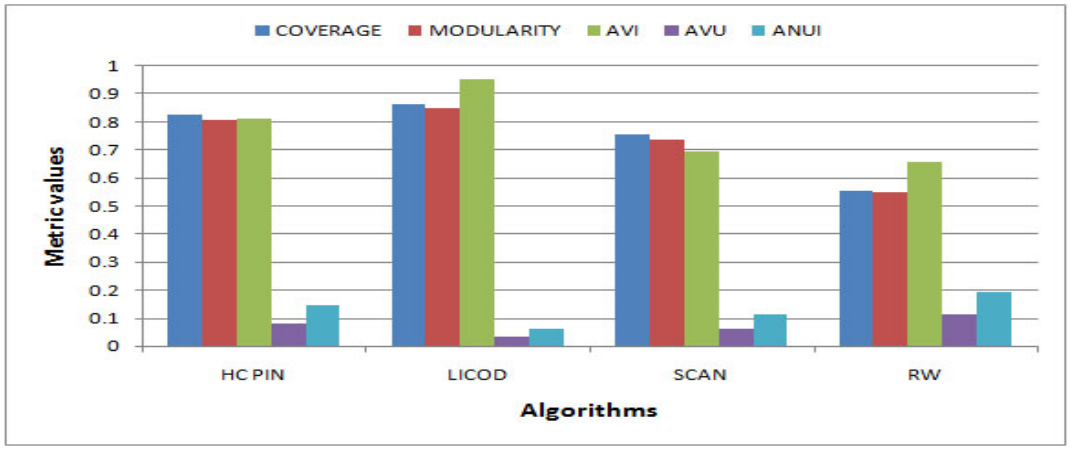
Figure 7.5: Comparison of effectiveness of AVI, AVU and ANUI with other metrics on LFR benchmark graphs. NMI, ARI, Purity and F-measure are accuracy metrics.



(a) Football



(b) Karate



(c) GR-QC

Figure 7.6: Comparison of effectiveness of AVI, AVU and ANUI with other metrics on real-world network.

ANUI. For the case of HC-PIN also, ANUI values are lower than real clustering which indicates predicted clustering is not exactly same as real clustering. For the case of SCAN, Modularity and Coverage show comparatively higher values, whereas all accuracy metric have indicated level of accuracy almost equal to zero. ANUI indicated low accuracy level for SCAN on all of the three LFR graphs and also for LICOD it shows quite reasonable values in perspective of accuracy.

To get more insight about the ability of ANUI to deal with the accuracy, Isolability and Unifiability of communities are analyzed. Considering the AVI values, one can notice higher values of AVI for real clustering on all the LFR graphs. Like Modularity and Coverage, AVI is also very much dependent on internal connections within the community. Summarily, good communities are supposed to have higher connectivity within the community. Hence, one can notice higher values of all three metrics. However as mentioned above, Modularity and Coverage are unable to deal with accuracy. Clearly, AVI shows higher values for HC-PIN but much lower values than real community structure for LICOD, SCAN and RW. On the contrary, Modularity and Coverage show comparatively higher values for HC-PIN, SCAN and RW. Clearly, AVU values indicate that for all LFR graphs, the clusters are quite likely to unify with other communities. Interestingly, AVU shows high values for LICOD on all of the three LFR graphs. This happens because communities detected are very inaccurate and AVI values are very low. It means that internal connectivity within the community is less and apparently external connectivity become high. Obviously, more external connectivity will result high AVU value. In this context, HC-PIN is far better than LICOD in terms of accuracy. Hence, both AVI and AVU have significant role in dealing with accuracy and better understanding of the community structure. Solely, ANUI can give overall value that is enough for evaluating communities whereas, ANUI along with AVI and AVU values can avail better understanding of the communities. Significance of AVI, AVU and ANUI values are summarized in Table 7.1.

Table 7.1: Combination of AVI, AVU and ANUI with indication of clustering quality and accuracy.

AVI	AVU	ANUI	Clustering
High	Low	Low	Bad
High	Moderate	High	Good
High	High	High	Good
Low	High	Low	Bad
Moderate	High	High	Good
Low	Low	-	Not possible
Moderate	Moderate	Moderate	Good

Real-world Network

Results obtained on real-world networks are presented in Figure 7.6. Real community structure for both Football and Karate networks show highest accuracy metric values as these metrics are evaluated over the real communities. Quality of real communities for Football network is not that high, but real communities for Karate network show very high quality. For Football network, HC-PIN produces most accurate clustering as indicated by accuracy metric values. Accuracy of LICOD and SCAN are also quite high, whereas RW produces least accurate communities. ANUI, Modularity and Coverage also indicated the same. For Karate network, RW produces most accurate communities. Accordingly, one can notice highest ANUI value and it is almost same as real communities. Accuracy of HC-PIN and SCAN are also high but much lower than the real communities. Accuracy metrics indicate that LICOD has produced least accurate communities. Modularity and Coverage fail to detect such an inaccurate community structure but ANUI has clearly indicated that LICOD has produced least accurate communities. Modularity and Coverage values obtained for the communities predicted by LICOD have showed their incapability to deal with the accuracy.

For GR-QC network, the real community structure is unavailable so we have to depend only on the values of quality metrics to ensure accuracy. In the case of LFR graphs as

well as on Football and Karate network, the worst performance of LICOD has been noticed both in terms of quality and accuracy. Moreover in the above discussion, about incapability of Modularity and Coverage to ensure accuracy (example inaccuracy of LICOD) is also noticed. In the light of those observations, indication by Modularity and Coverage as LICOD detecting the best communities in GR-QC network is questionable. On the contrary, the inclination of ANUI towards accuracy has already been notice in the above discussion. Hence, indication of ANUI as worst communities predicted by LICOD is quite believable. High accuracy of HC-PIN and RW also been observed above. Thus, indication of ANUI as HC-PIN and RW better than LICOD and SCAN is also believable to some extent. It can also be asserted easily that the communities of HC-PIN and RW are more accurate by observing simply the ANUI values without knowing the real communities of GR-QC.

7.6.3 Metric Characteristics

From the above discussions, algorithms HC-PIN, LICOD, SCAN and RW can be ranked in terms of their community accuracy as 1, 4, 3 and 2 respectively. With this intuition, the characteristics of proposed metrics are analyzed. Characteristics on LFR graphs are presented in the Figure 7.7. Clearly, Cumulative Density Function (CDF)s of ANUI for most of the algorithms are following closely with respect to that of the real communities. Although, CDF of ANUI for RW is following that of real communities but it is not closely following. Hence, it can be asserted that characteristics of ANUI for accurately predicted communities by any algorithm will be same as characteristics of ANUI for real communities. Now considering the CDFs of AVI, one can notice that CDFs of most algorithms are following, but except HC-PIN, they are far from the CDF of AVI for real communities. Therefore, the characteristics of AVI is considered same as ANUI. On the contrary, characteristics of AVU is similar to that of the real communities but AVU alone cannot

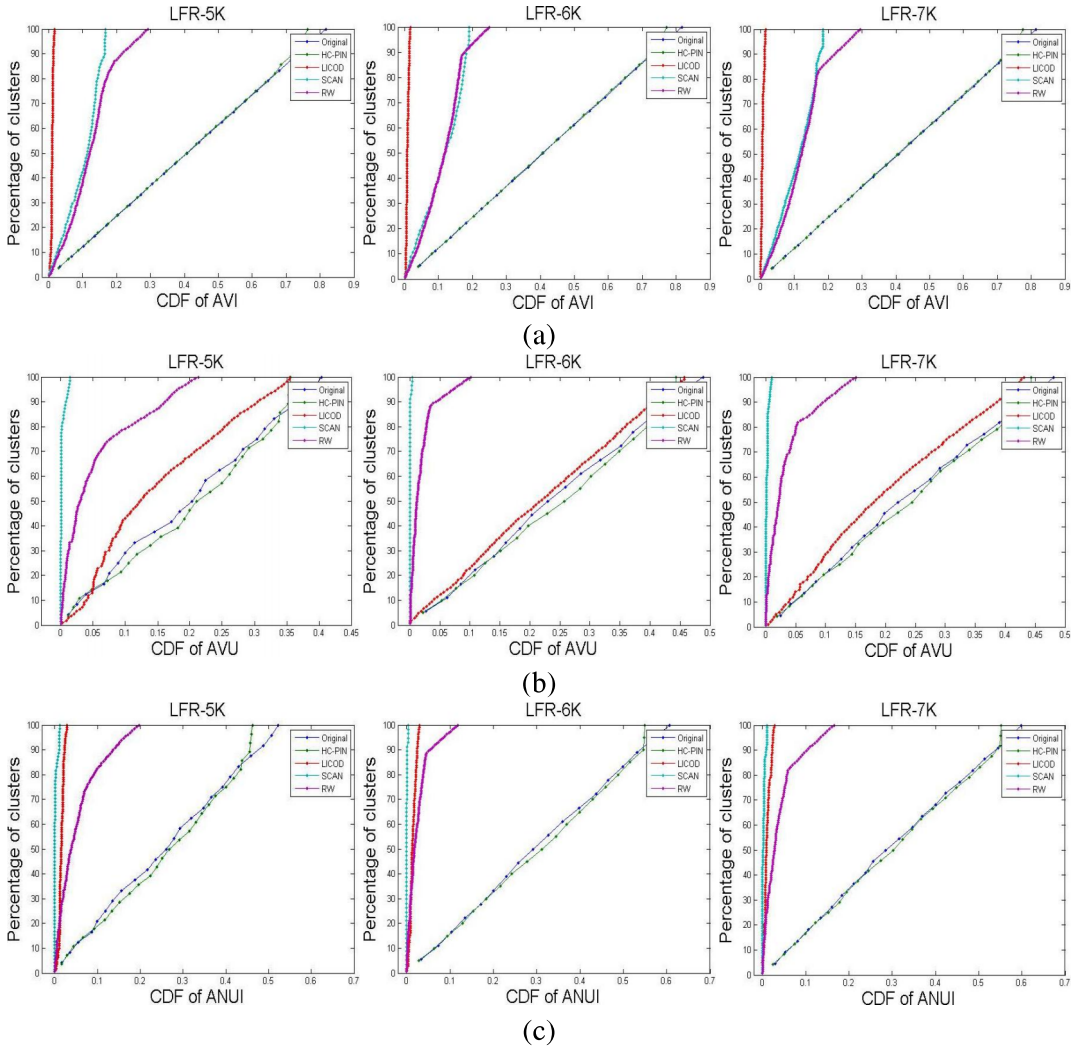


Figure 7.7: Characteristics of AVI, AVU and ANUI on LFR benchmark graphs. Characteristics of AVI, AVU and ANUI are shown in (a), (b) and (c) respectively.

determine accuracy. For example, communities detected by LICOD on LFR graphs show similar AVU characteristics as real communities but that does imply accuracy (see in Figure 7.5 AVU values of LICOD are same as real communities but accuracy metric values are very low). However, if AVU is considered along with AVI, it can indicate accuracy. ANUI reflects combined effect of both AVI and AVU so one can notice CDF of ANUI for LICOD is not following the real communities.

CDFs of all three metrics for real communities of LFR graphs increase almost linearly

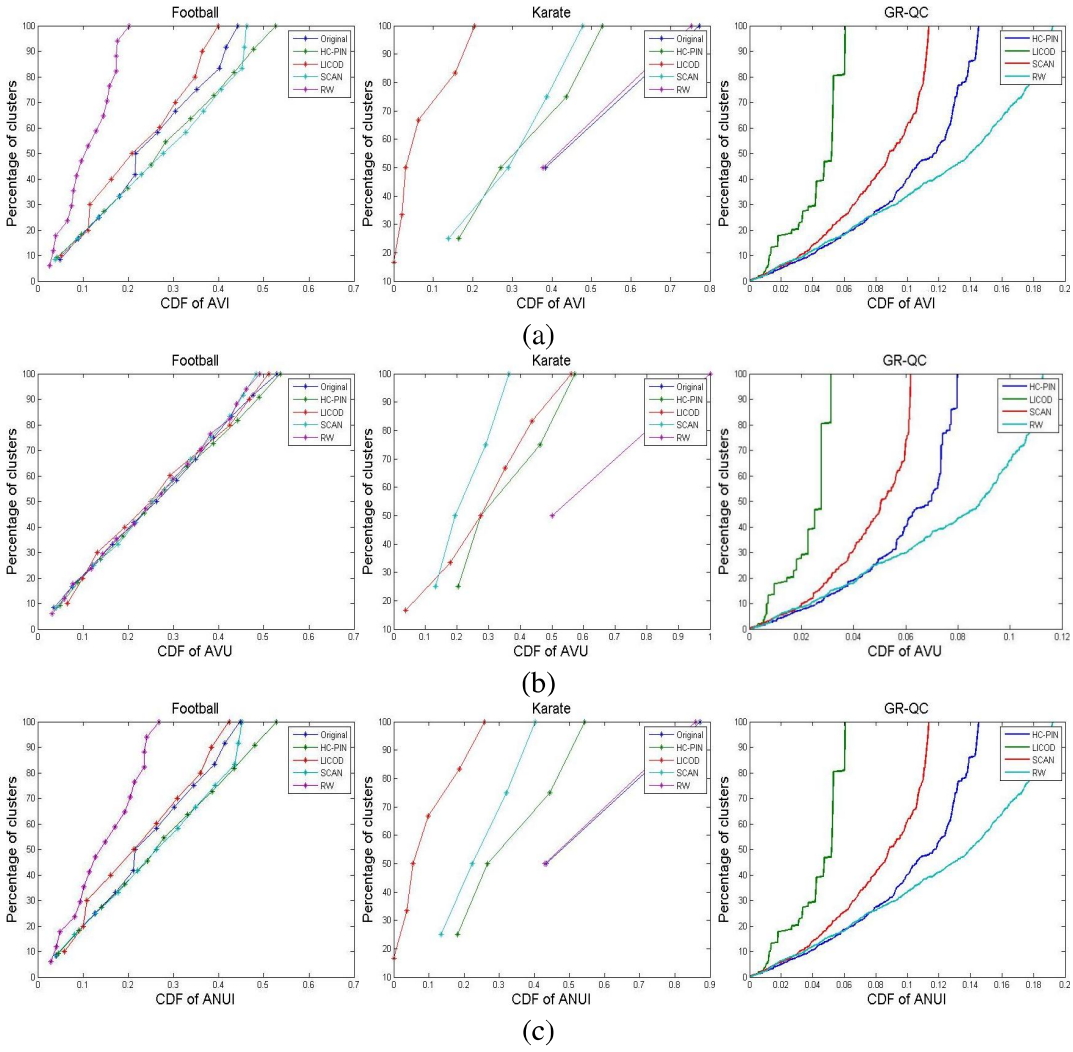


Figure 7.8: Characteristics of AVI, AVU and ANUI on real-world networks. Characteristics of AVI, AVU and ANUI are shown in (a), (b) and (c) respectively.

with the increment in percentage of communities they cover. Such linear increment can also be observed for real communities of Football and Karate networks as shown in Figure 7.8. During competitiveness analysis in Figure 7.6 it was noticed that RW detects very accurate communities on Karate network. Accordingly, one can notice CDFs of all three metrics for RW are aligned with real communities of Karate network. With such insight knowledge, the indication regarding accuracy of different communities of GR-QC

can be determined by simply visualizing linearity of AVI, AVU and ANUI characteristics. Clearly, characteristics of all three metrics for RW are more linear. This indicates accuracy of clustering produced by RW on GR-GC network is higher than other three algorithms. Similarly, one can assert that accuracy of LICOD on GR-QC is least among all four algorithms.

7.7 Analysis using Proposed Evaluation Methodologies

In this section, proposed analysis methodologies: 1) RITA analysis and 2) visual analysis with regression line dominance are demonstrated on real-world networks and benchmark optimization functions.

7.7.1 RITA Analysis

Experimental Setup: Six community detection algorithms: FastU [17], HC-PIN [203], LeadF [173], LICOD [97], RandW [183] and SCAN [216] are considered to analyze the RITA of each algorithm relative to other algorithms. Five accuracy metrics are considered for measuring accuracy of communities, which include popularly used NMI [184], ARI [157], F-measure, Purity [126] and Entropy [225]. Four quality metrics are considered for evaluating quality of communities, which include Modularity [143], Coverage [21], ExtD [169] and proposed AVI (see above subsection 7.2.3 for detail). Four networks Dolphin [122], Football [61], Karate [218] and Strike [132] with known ground truth communities are considered for analysis.

Weight Assignment: Lower limit of weights 25% and upper limit of weights 75%. Thus, initially accuracy and quality will have weights $\omega_A = 0.25$ and $\omega_Q = 0.75$ respectively. Each accuracy and quality metrics will get 0.25/5 = 0.05 and 0.75/4 = 0.1875 respectively. Weights assigned to both accuracy and quality are equally distributed over respective metrics. Equally distributed in the sense that there are five accuracy metrics, and say weight assigned for all is 75% of total weight 1 i.e. $\omega_A = 0.75$, which is equally distributed over all five accuracy metrics. Hence, each metrics will get weight $\omega_a = \frac{0.75}{5} = 0.15$ to contribute in ranking. Since, $\omega_A = 0.75$, apparently weights assigned to quality metrics becomes $\omega_Q = 0.75$. Four quality metrics will get weight $\omega_q = \frac{0.25}{4} = 0.0625$ each to contribute in the score. Summation of all weights (value only) assigned to different metrics is $\omega_T = 1$. Accuracy weights are varied from 25% to 75% and apparently quality weights varied from 75% to 25%. The value of W is incremented with difference of $d = 5\%$, which resulted 11 different MCDM scores for each algorithm.

Result Interpretation: Plotting of MCDM scores and percentage of accuracy in 2-dimensional plot is presented in Figure 7.9. The curves obtained for all algorithms are relative to performance of other algorithms. Interpretation of results are made as follows. When accuracy given 25% weightage, apparently quality contribution becomes 75% so quality metrics will have more impact on the score. Similarly, towards the end 75% weightage is given to accuracy contribution so accuracy will have more impact on the score. If the curve representing an algorithm always remains at high level and above other, it means the RITA of that algorithm is high. If accuracy is more stronger i.e. if the algorithm likely to produce highly accurate communities then only curve will tend to increase and remains high all the time, otherwise curve will decrease or remains at low. If the curve is decreasing then it means the algorithm is not inclined towards the accuracy i.e. it is likely that the algorithm will produce inaccurate communities. Therefore, if curve is decreasing, the

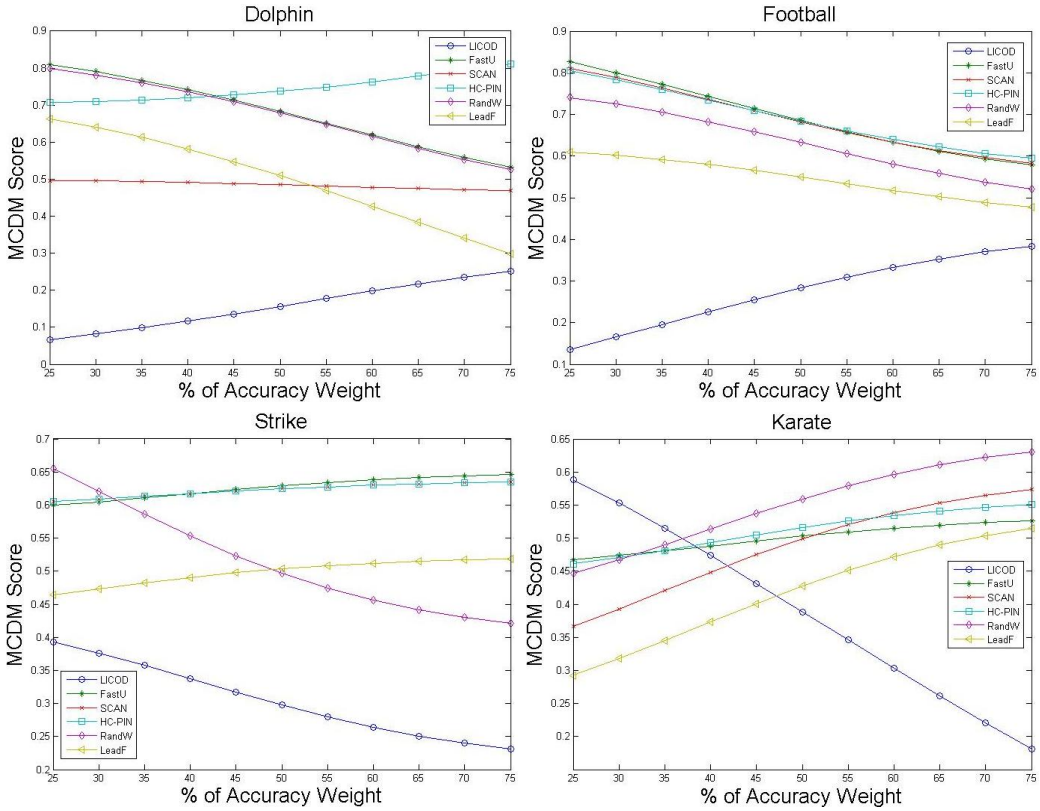


Figure 7.9: Tendency of algorithms' relative inclination towards accuracy (RITA).

algorithm is termed as biased towards quality. Significance of higher scores in terms of RITA and necessary conditions are summarized in Table 7.2. Low score itself indicates that RITA of algorithm is low.

Result Analysis: With the intuitive interpretations discussed above, the results presented in the Figure 7.9 are analyzed as follows. The FastU, SCAN and HC-PIN algorithms show almost similar curves for all the networks. Initially, when accuracy contribution is given 25% weightage, FastU and HC-PIN show higher scores. An important fact to note here is as follows. Connectivity pattern of networks is different so it is obvious that the performance of same algorithm varies in different networks. However, if an algorithm is really competent in detecting accurate communities, it will do that irrespective

Table 7.2: Summary of high MCDM scores in the curve and meaning in terms of RITA and bias of algorithms.

Accuracy	Quality	Stronger	Curve	Score	High RITA	Biased
High	High	Accuracy	Increasing	High	Yes	No
High	High	Quality	Decreasing	High	No	Yes
High	Low	Accuracy	Increasing	High	Yes	No
Low	High	Quality	Decreasing	High	No	Yes
Low	Low	Accuracy	Increasing	Low	No	No
Low	Low	Quality	Decreasing	Low	No	Yes
High	Low	Accuracy	Increasing	Low	No	No
Low	High	Quality	Decreasing	Low	No	Yes

Table 7.3: Average of accuracy metrics and quality metrics values.

Algorithms	**Dolphin**		**Football**		**Karate**		**Strike**	
	Accuracy	Quality	Accuracy	Quality	Accuracy	Quality	Accuracy	Quality
LICOD	0.4741	0.2351	0.2607	0.0856	0.4808	0.5084	0.3411	0.4158
FastU	0.4325	0.4763	0.6795	0.4598	0.4958	0.4401	0.7622	0.5403
SCAN	0.3362	0.3819	0.6671	0.4399	0.4758	0.3488	0.9218	0.5681
HC-PIN	0.9490	0.5970	0.7965	0.4548	0.5219	0.4492	0.9218	0.5681
RandW	0.5647	0.3235	0.5128	0.4081	0.9201	0.5137	0.7159	0.4951
LeadF	0.2267	0.2838	0.4363	0.3661	0.4285	0.3022	0.4136	0.3944

of network. One can notice higher MCDM scores for HC-PIN in particular, which means the RITA of HC-PIN is high in comparison to other algorithms. On the contrary, LICOD is showing low RITA in all networks. However, in the cases of FastU, SCAN and RandW, RITAs are higher than LICOD and LeadF but they are biased towards the quality.

The proposed framework gives an easy platform for expressing relative inclination of an algorithm towards accuracy, which otherwise was very difficult. The advantage of proposed framework can be understood with the actual indications in metric values as follows. Performance of all six algorithms indicated by each of the accuracy and quality metrics are presented in Table 7.3. For ease of understanding, average values of accuracy metrics and quality metrics are considered. Clearly, both RandW and HC-PIN show high accuracy and quality metrics values in Dolphin network, accordingly one can notice high

RITAs. However, it is difficult to say with the results presented in Table 7.3 that HC-PIN is inclined towards accuracy but RandW is not. On the contrary, the inclination of HC-PIN is easily identified with proposed framework as in Figure 7.9. Similarly, all the four algorithms except LICOD and LeadF have higher average accuracy and quality values in Strike network, but it cannot be determined that HC-PIN, FastU and SCAN are inclined towards accuracy but RandW is not. Interestingly, despite of higher average accuracy and quality values in the Football network, all of the HC-PIN, FastU and SCAN are seems to be biased. On the other hand the LICOD algorithm, which has least average values, is seems to be inclined towards accuracy. Nevertheless, the MCDM scores are relatively low for LICOD in comparison to HC-PIN, FastU and SCAN. This indicates, higher RITA for HC-PIN, FastU and SCAN. In case of Karate network, all the algorithms show inclination towards accuracy except LICOD. However, only RandW appears to have highest RITA. High performance indicated by different metrics in the Table 7.3 can be easily verified with RITA indications in the Figure 7.9 and Table 7.2.

7.7.2 Visual Analysis

Solution Quality Comparison of EOAs

Experimental Setup: First 14 benchmark functions of CEC 2005 special session on real parameter optimization (see subsection 5.4.1 for detail) are considered for analysis. Three algorithms: PSO-TVIW [180], PSO-TVAC [158] and SADE [155] are considered for demonstrating comparisons methodologies. For one-to-one and one-to-many comparison, SADE is considered as the acting algorithm, whose performance has to be evaluated with respect to other two competing algorithms. For many-to-many comparison, each of the three algorithms is considered as acting algorithm (comparing algorithm) during its turn and other two are considered as competing algorithms (counter competitor).

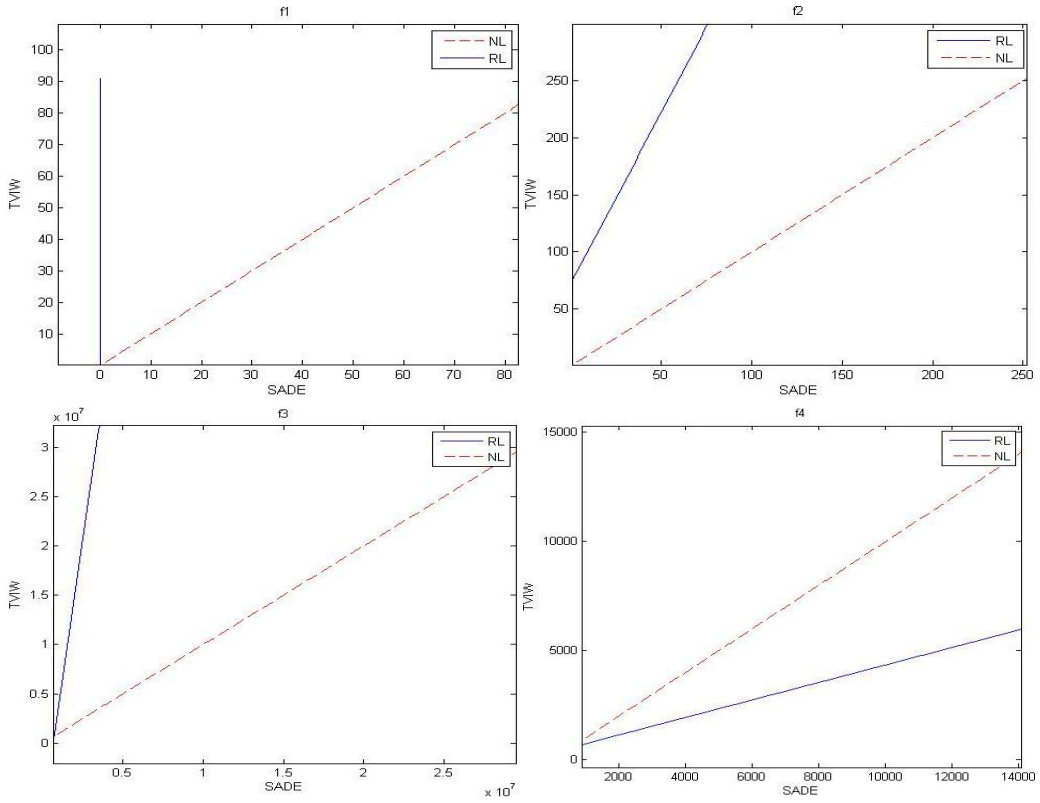


Figure 7.10: One-to-one solution quality comparison in CEC2005 functions f1-f4.

Demonstration of One-to-one Comparison: In Figure 7.10, results of one-to-one comparison of SADE with TVIW (shorted PSO-TVIW) and TVAC (shorted PSO-TVAC) are presented. The interpretation of visual results are discussed in section 7.4.4, particularly the significance of the angle between NL and RL, and the position of RL with respect to NL is detailed. The visual results are analyzed based on the angle and position of RL with reference to NL as follows. For f1-f3, the RL is above and produce high angle (nearly the maximum angle 45 degree) with the NL. This implies SADE outperforms over both TVIW and TVAC. Generally, TVAC performs better than TVIW and the SADE is one of the best performing DE variants [197]. Thus, better performance of SADE than TVAC justifies the poor performance than TVIW.

Demonstration of One-to-many Comparison: In Figure 7.11, visual results of the

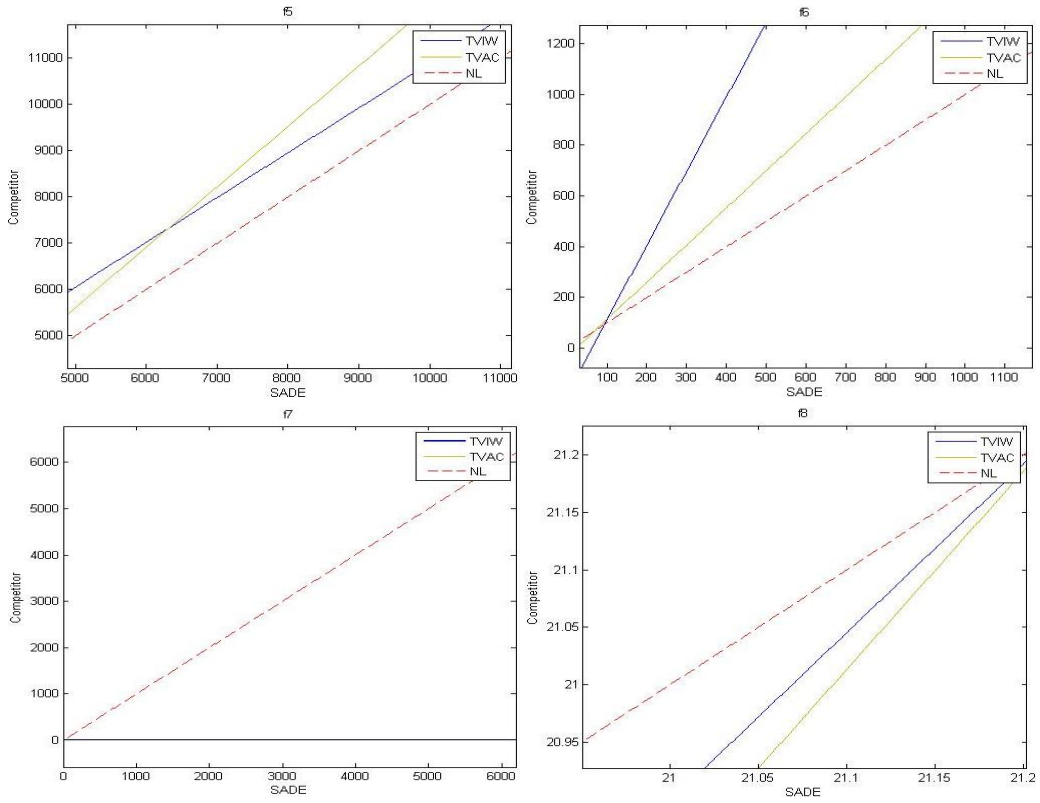


Figure 7.11: One-to-many solution quality comparison in CEC2005 functions f5-f8.

one-to-many comparison of SADE with respect to TVIW and TVAC are presented. Clearly, SADE's performance is better in f5, f6, f9, f10 and f12 as RL corresponding to TVIW and TVAC lies above producing higher angle with NL. In the visual results for f7, f8 and f11, RLs lies below the NL, which indicates poor solution quality of SADE in contrast to TVIW and TVAC. Clearly, one can notice the worst numeric values of SADE for f7, f8, and f11. However, overall SADE performs better than both TVIW and TVAC since five out of eight functions SADE performs better than both the algorithms.

Demonstration of Many-to-many Comparison: The visual results of many-to-many comparison i.e. ranking of SADE, TVIW and TVAC are presented in Figure 7.12. The RL representing SADE in f9, f10, f12 and f13 lies above the NL, and it is also lies above the RLs of both TVIW and TVAC. Therefore, SADE is ranked as best for these functions.

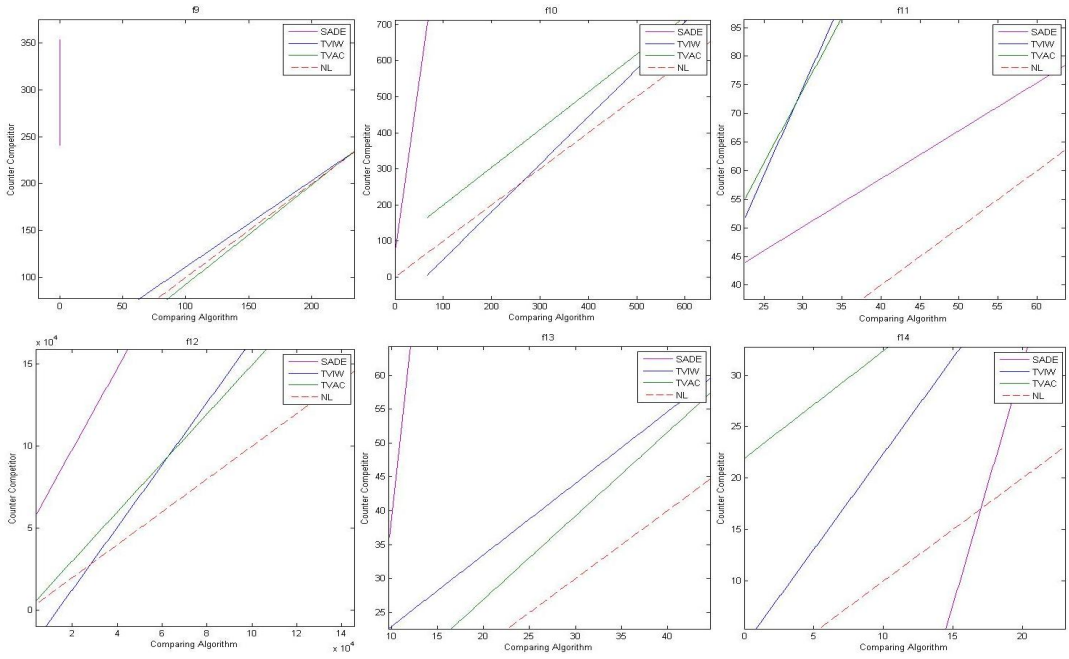


Figure 7.12: Many-to-many solution quality comparison in CEC2005 functions f9-f14.

Moreover, the angle between RL and NL for SADE is high in these functions, which indicates the SADE produces best solutions than TVIW and TVAC. Now, the next best would be one of the two i.e. either TVIW or TVAC for these functions. The functions f6, f10 and f12, TVAC is second best, while TVIW is second best for rest of the functions. In case of f11, TVIW is ranked as best since angle between RL and NL is largest. In case of f14, although SADE has largest angle between RL and NL, SADE is not the best because its RL is not above the NL and intersect almost the middle of RL and NL. In this case, TVAC is best since it's RL lies above all RLs and NL.

One-to-many Accuracy Comparison of Community Detection Algorithms

Four community detection algorithms: FuzAg (see chapter 4), GAFCD [185], FMM/H2 [186] and CFGC [63] are considered for analysis. To demonstrate one-to-many comparison, FuzAg algorithm is compared with GAFCD, FMM/H2 and CFGC in terms of an

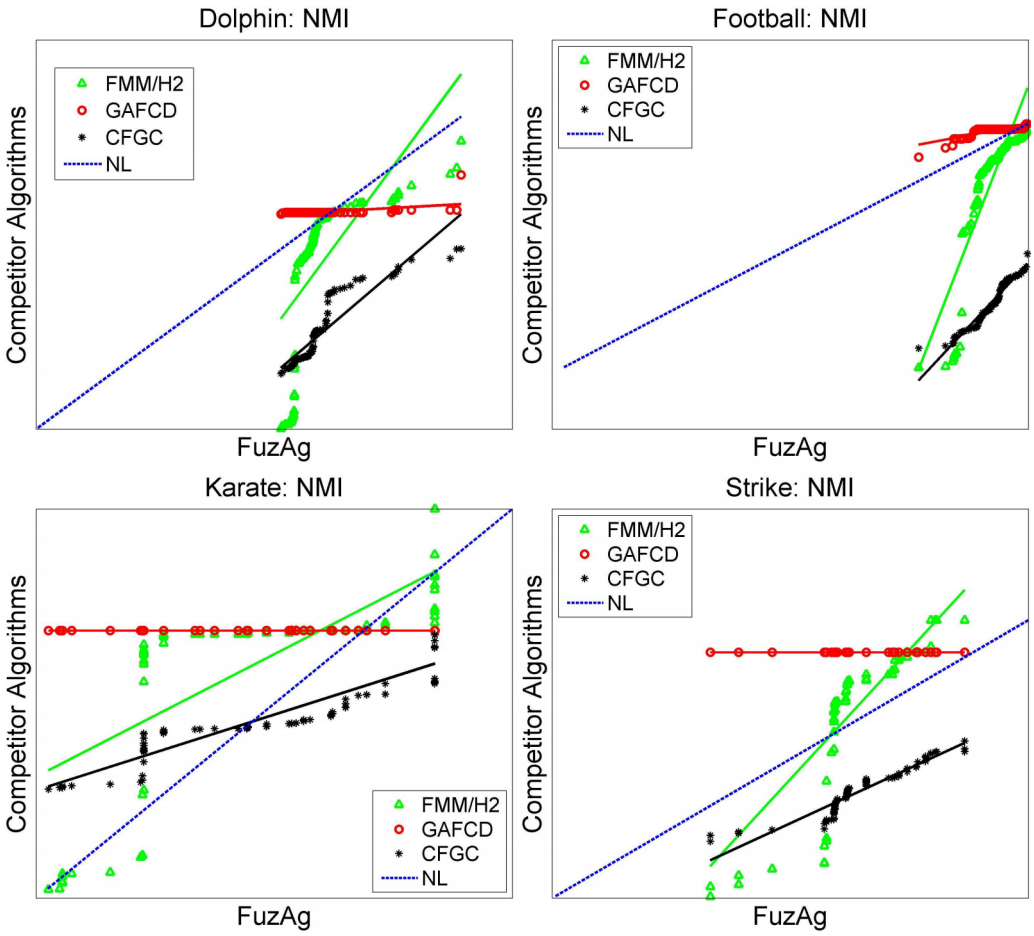


Figure 7.13: One-to-many comparison of community detection algorithms in terms of NMI values.

accuracy metric on four networks: Dolphin, Football, Karate and Strike. Accuracy metric NMI [184] is considered that compares the communities predicted by the algorithms with ground truth communities to determine accuracy of communities. NMI has to be high for good communities so the interpretation of results becomes opposite to the interpretations considered for EOAs above. Therefore, if RL is below the NL it would mean better performance of actor or comparing algorithm. Other interpretations such as intersection or angle between RL and NL remains same but only difference is large values are considered statistically more significant. Results presented in Figure 7.13 clearly indicate the performance of FuzAg is better than FMM/H2, GAFCD and CFGC in Dolphin and Football

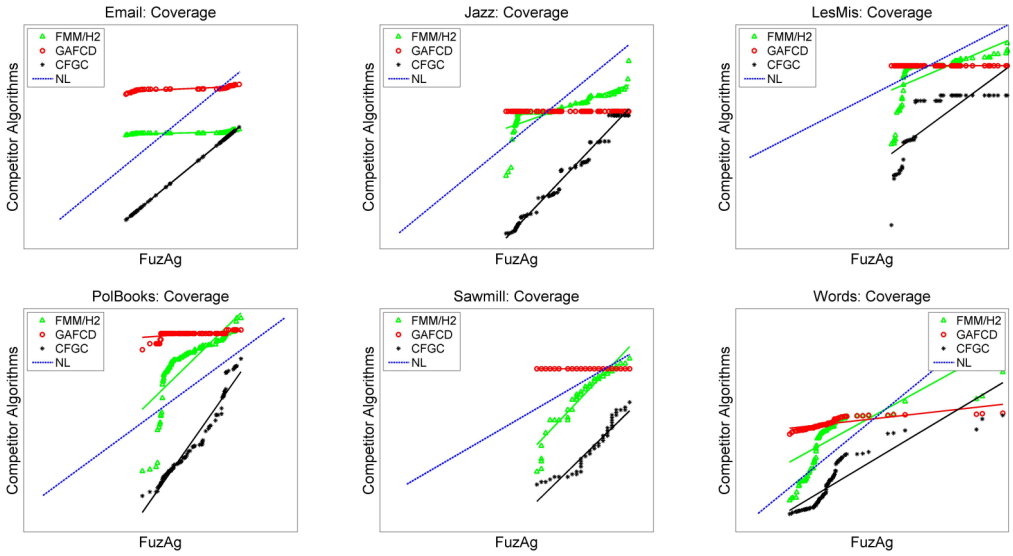


Figure 7.14: One-to-many comparison of community detection algorithms in terms of Coverage values.

networks since RL is below NL. We can also see most of the points in quantile-quantile plot are actor-dominant i.e. FuzAg dominates other algorithms. However, performance of FuzAg is poor in Karate network compared to other algorithms as indicated by the RL. FuzAg performs only better than CFGC in Strike network.

One-to-many Quality Comparison of Community Detection Algorithms

To demonstrate one-to-many quality comparison, FuzAg is compared with GAFCD, FMM/H2 and CFGC in terms of quality metric Coverage on six networks: Email, Jazz, LesMis, PolBooks, Sawmill and Words. The quality metric Coverage [21] considers connectivity pattern of the network to measure quality communities predicted by the algorithms. Like NMI, Coverage also has to be high for good communities so interpretation of results are same as NMI. Results presented in Figure 7.14 indicate FuzAg detects better quality communities than CFGC in all of the six networks. FuzAg performs better than both GAFCD and FMM/H2 in Jazz, LesMis, Sawmill and Words networks. In PolBooks

network performance of FuzAg is poor, while in Email network, performance of FuzAg is similar to that of GAFCD and FMM/H2.

7.8 Conclusion

In this chapter, three issues related to community evaluation are addressed. First three quality metrics namely AVI, AVU and ANUI are proposed. Metrics are designed based on the two properties of social community formation: unification and isolation. Second a framework for RITA analysis is designed to overcome the trade-off between quality metrics and accuracy metrics during evaluation of communities. The framework incorporates the concept of MCDM technique in background to generate relative scores of algorithms. Lastly, a visual analysis methodology is developed to deal with output variation of community detection algorithms and ease comparative performance evaluation. The methodology is designed based on the concepts of quantile-quantile plot and simple linear regression. Advantages of these proposals are highlighted below.

- Capability of each of the three metrics is analyzed individually while dealing with accuracy. Results demonstrate that both AVI and ANUI alone can indicate the level of accuracy of communities, but AVU alone cannot determine accuracy. However, ANUI along with AVI and AVU can give more insight knowledge about the community structure.

- Good quality communities can be identified by simply observing AVI and AVU values. Theoretically proved that AVI, AVU and ANUI satisfy all of the six quality related properties.

- Linearity in characteristics of AVI, AVU and ANUI can give the indication about accuracy. Such indication will be very helpful for determining accuracy of communities in the networks where ground truth in unavailable.

- The proposed framework for RITA analysis easily identifies relatively more inclined algorithms towards accuracy. Moreover, a very common problem i.e. the trade-off between accuracy and quality of the algorithms during evaluation is mitigated with the proposed framework.

- In visual analysis, quantile-quantile plot is used to define dominance of each point, which is nothing but comparison of good (or bad) solutions of one with good (or bad) solutions of other algorithms. Thus, with proposed methodology, it can be easily explained if an algorithm is performing better, actually, how good the solutions are compared to other algorithms.

- The quantile-quantile plot ensures involvement of each of the multiple outputs in the evaluation process by incorporating the notion of dominance. SLR analysis on the data points in quantile-quantile plot accumulates the point dominance by simply deriving linear relationship.

- The dominance of an algorithm is determined simply by observing the angle between RL and NL and position of the intersection of RL and NL.