Selecting Clustering Algorithms for Identity-By-Descent Mapping

Ruhollah Shemirani †1, Gillian M Belbin[1], Keith Burghardt[2], Kristina Lerman[2],Christy L Avery[3], Eimear E Kenny[1], Christopher R Gignoux[4], José Luis Ambite[2]

[1] *Institute for Genomic Health, Icahn School of Medicine at Mount Sinai, New York, NY , USA*
[2] *Information Sciences Institute, University of Southern California, Marina Del Rey, CA, USA*
[3] *Department of Epidemiology, University of North Carolina, Chapel Hill, NC, USA*
[4] *Colorado Center for Personalized Medicine, University of Colorado Anschutz Medical Campus, Aurora, CO , USA*
† *E-mail: ruhollah.shemirani@mssm.edu*

Groups of distantly related individuals who share a short segment of their genome identical-by-descent (IBD) can provide insights about rare traits and diseases in massive biobanks using IBD mapping. Clustering algorithms play an important role in finding these groups accurately and at scale. We set out to analyze the fitness of commonly used, fast and scalable clustering algorithms for IBD mapping applications. We designed a realistic benchmark for local IBD graphs and utilized it to compare the statistical power of clustering algorithms via simulating 2.3 million clusters across 850 experiments. We found Infomap and Markov Clustering (MCL) community detection methods to have high statistical power in most of the scenarios. They yield a 30% increase in power compared to the current state-of-art approach, with a 3 orders of magnitude lower runtime. We also found that standard clustering metrics, such as modularity, cannot predict statistical power of algorithms in IBD mapping applications. We extend our findings to real datasets by analyzing the Population Architecture using Genomics and Epidemiology (PAGE) Study dataset with 51,000 samples and 2 million shared segments on Chromosome 1, resulting in the extraction of 39 million local IBD clusters. We demonstrate the power of our approach by recovering signals of rare genetic variation in the Whole-Exome Sequence data of 200,000 individuals in the UK Biobank. We provide an efficient implementation to enable clustering at scale for IBD mapping for various populations and scenarios.

Supplementary Information: The code, along with supplementary methods and figures are available at https://github.com/roohy/localIBDClustering

Keywords: Clustering; Community Detection; Identity-By-Descent; Comparative Analysis; Genome-wide Association Studies; Benchmark; Clustering Metrics.

1. Background

Finding structure in networks, known as community detection, or clustering, has a wide range of biomedical applications.[1–3] Recently, clustering algorithms have been applied in the context of Identity-By-Descent (IBD) mapping[4,5] as an alternative approach for rare variant association testing that leverages genotype data in the absence of directly observed variation for genomic discovery. This method relies on shared haplotypes along the genome co-inherited identically from a recent common ancestor and utilizes them as the basis for association testing, under the assumption that the haplotypes may co-harbour recently arisen rare variation not directly captured on genotyping arrays. In this process, as illustrated in Figure 1, the chromosome is first divided into consecutive windows. For each window, a graph of IBD sharing is generated, which we refer to as a local IBD graph. In these graphs, samples are represented as nodes and IBD sharing is represented by edges connecting the respective nodes carrying the shared haplotype. False-positive and false-negative edges, artifacts of errors in genotyping, phasing, and IBD estimation, add noise to these graphs. Clustering algorithm are used to refine them and consecutively the IBD information they represent. IBD sharing groups can then be tested for phenotype enrichment. In a study of individuals from the United Kingdom, Gusev et al.[4] found that, empirically, IBD mapping can yield up to forty times more statistical power than standard genome-wide association analyses (GWAS) in tagging rare genetic variation through recovering known and novel associations with binary phenotypes, especially in founder populations. Browning et al.[6] also replicated the results of a GWAS study via IBD mapping. Kenny et al.[7] used IBD mapping to fine-map known associations with plasma plant sterol levels in an isolated founder island population in Kosrae. Finally, Belbin et al.[8] identified the source of a common collagen disease in the Puerto Rican population of BioMe biobank.

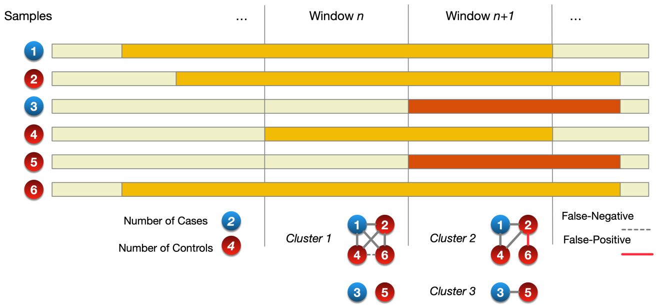

Fig. 1. A general schema of the IBD mapping process that can help identify shared haplotypes carrying rare causal variants. Haplotypes of the same color are inherited from the same ancestor.

There has been a plethora of innovations in clustering techniques, due to their increased importance.[9–13] New clustering methods have been proposed to address the size of social networks and internet hosts, which have grown to many millions of nodes in the past decade;[14–16] or to find new community structures that reflect the underlying data more accurately.[17,18] The emergence of large biobanks necessitates the employment of such new clustering techniques in the context of IBD-mapping. Yet, it remains unclear how advancements in community detection methods translate to this process, where the unique structural properties of local IBD graphs does not resemble that of common graphs analyzed in other fields of study. In this manuscript, we address this problem in three main aspects. First, we conduct a thorough analysis of the characteristics of local IBD graphs, and design a novel benchmark that realistically represents them. Second, we conduct a translational study of clustering metrics to IBD mapping related metrics to investigate their efficacy. Third, and most importantly, we evaluate both the power and scalability of common clustering algorithms in large datasets using both our benchmark and real data. By combining these aspects, we propose a methodical approach to find the most powerful algorithm for any datasets.

2. Methods

2.1. *Characterization of the Local IBD Graphs*

Common benchmark graphs such as those introduced by Lancichinetti et al.,[19] and Girvan and Newman,[18] are used to evaluate clustering methods in a variety of fields.[20] However, they should not be used to simulate local IBD graphs, mainly due to the properties of the local IBD relationships that generates these graphs. The topology of a graph that represents a relation between entities of a set is dictated by the properties of the relation. Local IBD relation is transitive. Thus, under ideal conditions, the local IBD relation can be represented as disjointed sets or cliques. In practice, false-positive and false-negative edges obfuscate these cliques, necessitating a graph representation. The goal in the clustering of local IBD graphs is to recover these well-defined cliques.

Noisy transitivity of local IBD relations results in uncommon graph properties. We look at the "small-world" property as an example.[21,22] This property cannot be calculated for local IBD graphs since, even before clustering, they are highly disconnected. For example, the local IBD graphs of chromosome 1 in the "Population Architecture using Genomics and Epidemiology" (PAGE) dataset[23] each have 13,961 connected components on average across 7952 local IBD graphs tested on chromosome 1, with an average of 3.74 nodes per connected component. In contrast, common benchmarking algorithms often generate a single connected component.[19]

Cluster size distribution is another area of difference between local IBD graphs and others. The LFR benchmark[19] only supports cluster size distributions that follow the power law. Estimating the local IBD cluster sizes using power law results in unrealistically low numbers of small clusters. A fitted power law distribution[24] underestimated the number of cluster sizes for clusters with less than thirteen members in PAGE dataset by a factor of ten (χ^2 p-value $= 3 \times 10^{-14}$). Cluster size distribution affects the statistical power of Louvain and Leiden clustering algorithms.[25] Thus, using power law distributions (as is common in graph benchmarking[19]) for our simulations would result in an erroneous evaluation of the fitness of clustering algorithms to recover local IBD communities.In the supplementary methods section 2.1, we describe our clique-centric benchmark that takes the specific properties of local IBD clusters into account and simulates phenotype for power analysis.

2.2. *Metrics*

Clustering metrics help analyze various properties of the recovered clusters that are either related to the inherent features of the clusters, such as the density of connections in the clusters, or their concordance with the true structure of the graph, such as the number of nodes that are in the same clusters as they are in the ground truth. We call the first group feature-based metrics in this manuscript to distinguish them from metrics that are based on ground truth. For local IBD clustering, it is important to calculate how much the results reflect the true structure of the cliques underneath the noise and errors. We studied 4 metrics based on ground truth, along with 6 feature-based metrics, since ground truth is often not available for real datasets. A full list and description of metrics is available in supplementary methods section 2.2.

2.3. *Clustering Methods*

We analyze five algorithms in three categories based on their methodology: Highly Connected Subgraphs(HCS)–the clustering algorithm used by DASH,[4] Louvain,[26] Leiden,[27] Infomap,[28] and Markov Clustering Algorithm (MCL).[29] Detailed description of these algorithms is available in supplementary method section 2.3. Every tested algorithm, except for HCS, is scalable to large datasets,[30] and can analyze our largest simulated dataset with 11,000 clusters in less than 5 minutes on average on our workstation running CentOS Linux release 7.4.1708 with 128 GB of memory and Intel® Xeon® Processors E5-2695 v2 (2.4 GHz) on a single thread.

3. Results

3.1. *Performance on Simulated Data*

Using our benchmark algorithm, we generated 750 graphs with a range of cluster counts, false-positive, false-negative rates, and phenotype prevalences, described in Supplementary Methods section 2.1.1, that added up to a total of 2,274,500 clusters with more than 6 million nodes across all simulated experiments. Our results show that this benchmark simulates the disjointedness of local IBD graphs, unlike the LFR algorithm (Supplementary Figure 2).

3.1.1. *Clustering Metrics*

We ran the clustering algorithms on the simulated datasets. We then calculated the scores achieved by every method for each metric. We calculated the Pearson correlation coefficients and R^2 scores[31] between metrics to see whether, and to what degree, each clustering metric is associated with statistical power. The results are displayed in Figure 2 and Supplementary Figure 1.

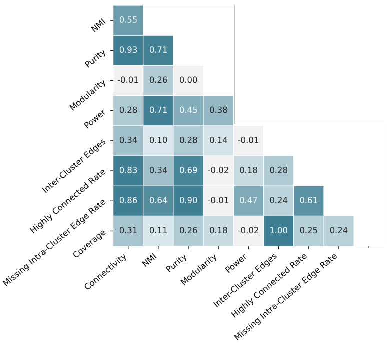

Fig. 2. R^2 scores among clustering metrics across all simulations.

Among all metrics, AMI has the highest concordance with statistical power, explaining 79% of the variation of the power score. Among the feature-based metrics, missing intra-cluster edge rate has the highest R^2 score of 29% with statistical power, while highly connected rate had the lowest score. While generating denser subgraphs with less missing edges is important to gain power, focusing solely on the density and ignoring coverage will counter those effects, resulting in lower power. Modularity showed a weak association with statistical power ($R^2 = 0.14$) compared to missing intra-cluster edge rate. This suggest that partitioning a graph into highly modular subgraphs (through optimizing modularity) does not necessarily result in clusters that represent the true IBD communities in the underlying population. While optimizing modularity is advantageous in finding large non-clique-like communities,[32] local IBD graphs are both clique-like and often smaller in scale. This high percentage of small cliques results in a discordance between modularity and power scores. If instead of a realistic cluster size distribution, we use a uniform distribution (resulting in a higher number of large clusters), the R^2 score for modularity and statistical power rises to 0.34 (from 0.15) and the gap between modularity/power and AMI/power R^2 scores decreases from 0.63 with realistic distribution, to 0.49 with the uniform distribution. At the same time AMI/power R^2 score increases only slightly to 0.83, compared to the 100% increase in modularity/power R^2 score.

The observed discordance between modularity and power in our experiments can also be explained through the concept of ”resolution-limit” in modularity optimization, i.e., the inability of modularity optimizing methods in detecting fine-grained clusters. Fortunato and Barthelemy found that the modularity score for a clustering is not only dependant on the structure of the graph, but also on the expected maximum possible modularity of any random graph with the same number of edges, as modularity optimization fails to capture clusters that have an order of magnitude fewer edges compared to the total number of edges in the graph.[25] This results in smaller clusters getting collapsed into other clusters via the optimization process. Small, clique-like structure of local IBD graphs intensifies the effects of this phenomenon on the performance of modularity metrics and methods optimizing it.

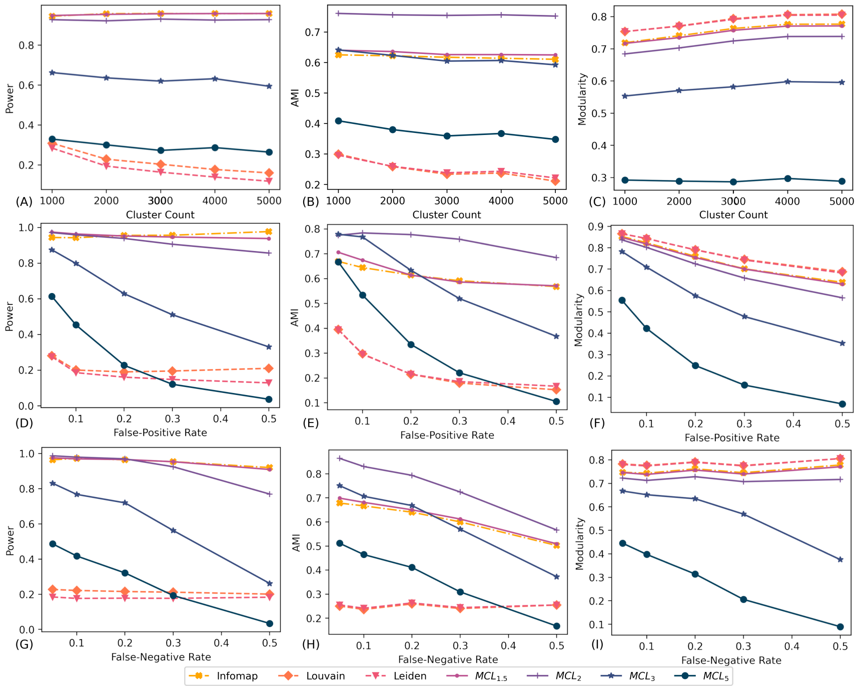

Fig. 3. The effects of number of simulated clusters, false-positives, and false-negatives edges on the performance of algorithms in terms of (A) power, (B) AMI, and (C) modularity.

Our results show that purity is unfit for our IBD clustering purposes. Regardless of the true underlying structure, a more granular clustering always yields a higher purity score. MCL_5, a clustering approach that has the fifth best performance in statistical power (Figure 3A) repeatedly gains the highest purity score, due to over-clustering, suggesting that purity score in the absence of others can be misleading and uninformative.

While AMI score is the best indicator of statistical power among the metrics we tested, due to the effects of smaller clusters (with less than 10 nodes), its concordance with statistical power is imperfect. As further demonstrated by the performances of MCL_2 and MCL_3 in Figure 3, compared to statistical power (Figure 3A), the gap between MCL_3 and top performing methods is less pronounced for the AMI scores (Figure 3B). Moreover, MCL_2 performance increases and surpasses the performance of Infomap and $MCL_{1.5}$ in terms of AMI score compared to the statistical power. The same issue, together with a high baseline, severely affects the performance of NMI as well. Compared to AMI scores, the gap in the NMI scores of MCL algorithms and Infomap is even less pronounced (Supplementary Figure 6).

Another disadvantage of the AMI metric is its reliance on the existence of ground truth data. However, in the absence of the true clustering information, our experiments show that none of feature-based metrics can be used to accurately predict statistical power. We look at missing intra-cluster edge rate as an example due to its higher R^2 score. Methods that yield the highest and lowest score in this metric (Leiden and MCL_5) both perform poorly in terms of statistical power, suggesting a lack of rank preservation in these metrics.

3.1.2. *Clustering Algorithms*

Table 1 shows the average score of clustering algorithms for every metric across all of the simulated datasets. Infomap received the highest average statistical power score, followed closely by MCL, while Louvain and Leiden got the lowest score (See Supplementary Figures 3,4, and 5).

As expected, Louvain and Leiden algorithms yield the most modular clustering results; followed by Infomap. In terms of conforming to the ground truth (purity, power, and AMI/NMI scores), however, Louvain and Leiden achieve a much lower score than MCL and Infomap; further corroborating our analysis of resolution limit in the previous section. As a result of resolution limit, Louvain and Leiden were unable to find smaller communities in our simulations. Greedy modularity optimization tends to merge lightly connected subgraphs into clusters. Although clusters of any size can be affected, those with fewer internal edges than $\sqrt{2E}$, with E as the total number of edges in the graph get merged frequently.[25] For example, the average number of edges for a graph with 2,000 clusters in our experiments is 62,007, which means any pairs of clusters that have a combined edge count smaller than $\sqrt{2 \times 62,007} = 352$ have a high chance of being merged by Louvain and Leiden if they are connected by a single edge, as it increases the modularity score. The vast majority of IBD clusters have less than 352 individuals.

This threshold for resolution limit grows at a faster rate compared to the number of large clusters (Supplementary Figure 9A). In other words, the average number of subgraphs that are larger than this threshold decreases as the total number of clusters increases. The approximate threshold for resolution limit grew from 227 to 744 as cluster count was increased from 1,000 to 10,000 clusters. At the same time, the percentage of clusters larger than the resolution limit threshold decreased from 23.4% to 0.8%. This effect also causes the modularity optimizing algorithms to have an improved modularity score as the number of clusters grows while their statistical power decreases.

We further analyze the distribution of connectivity scores achieved by the algorithms across all of our simulations in Supplementary Figure 7. The average percentage of nodes that were connected to at least half of the other members of their cluster, extracted by Louvain and Leiden, was 13% and 12%, respectively. The same average for MCL_2 was 78%, indicating that Louvain and Leiden merge more cliques together compared to other methods.

Resolution limit has another disadvantage; the dependence of accuracy on the overall edge count and not on the individual clusters[25] causes implications for local IBD clustering; where a variety of cluster size distributions exist for the same total edge count. For example, in the PAGE study dataset, the average number of edges per cluster for local IBD graphs that only include samples from Puerto Rican and African American populations is 96.8±12.7 and 1.6±0.1 respectively. Thus, the statistical power of Louvain and Leiden is subject to change between the two populations, even in the same dataset. The average number of nodes per cluster in the ground truth was 3.6 (std=0.2), the average number of nodes per clusters found by Louvain and MCL_5 were 197.7 (std=212.5) and 3.0 (std=1.7), respectively (See Supplementart Figure 10).

The Effects Of False-Positive Edges

Our experiments show that the supremacy of the Infomap, $MCL_{1.5}$, and MCL_2 performances over other methods is stable for false-positive rates ranging from 5% to 50% of the total num-

Table 1. Average scores (with standard error) of clustering algorithms across our experiments. Overall, MCL$_2$, Infomap, and MCL$_{1.5}$ yielded the best performances. Modularity optimizing methods had a much lower power.

		Methods						
Metrics		Infomap	Louvain	Leiden	MCL$_{1.5}$	MCL$_2$	MCL$_3$	MCL$_5$
Connectivity	Mean	41.52%	13.03%	12.28%	49.34%	78.78%	90.58%	**94.03%**
	Error	14.20	8.79	9.07	15.63	13.31	8.31	5.32
AMI	Mean	61.77%	24.81%	25.18%	63.05%	**75.60%**	61.36%	37.26%
	Error	8.62	12.04	11.65	9.51	12.69	22.12	26.02
Purity	Mean	63.24%	23.58%	23.03%	67.58%	86.85%	92.57%	**94.41%**
	Error	7.62	11.84	12.12	8.77	6.76	6.51	6.01
Modularity	Mean	75.52%	**78.64%**	78.48%	75.02%	71.76%	57.98%	29.07%
	Error	13.52	11.99	12.17	13.19	14.32	21.14	24.39
Power	Mean	**95.49%**	21.54%	17.98%	95.47%	92.61%	62.85%	29.05%
	Error	5.84	10.36	10.98	3.69	12.19	31.64	30.97
ICE*	Mean	14.26%	**8.70%**	9.10%	14.64%	17.91%	32.66%	66.35%
	Error	10.61	6.77	7.45	9.80	11.66	21.84	27.19
HCR**	Mean	27.40%	19.53%	15.95%	33.80%	82.50%	**95.15%**	91.67%
	Error	12.27	19.35	16.42	17.11	19.47	9.56	23.31
MICER***	Mean	37.92%	94.33%	94.09%	36.44%	25.51%	22.17%	**14.65%**
	Error	13.93	6.24	6.12	14.64	16.20	13.84	8.04
Coverage	Mean	85.74%	**91.30%**	90.90%	85.36%	82.09%	67.34%	33.65%
	Error	10.61	6.77	7.45	9.80	11.66	21.84	27.19
NMI	Mean	91.72%	58.89%	59.65%	92.02%	**95.26%**	94.10%	91.70%
	Error	3.58	19.47	19.19	2.90	2.89	3.36	3.58

*: Inter-Cluster Edges **:Highly Connected Rate ***: Missing Intra-Cluster Edges Rate

ber of edges. Figure 3 illustrates the effects of false-positives on the performance of algorithms in three metrics. High rates of false-positive edges were simulated to simplify detection and comparison of performance patterns. They do not happen in our real data experiments regularly since iLASH, our IBD estimation algorithm, has a low false-positive rate.[33] The statistical power of Infomap and $MCL_{1.5}$ stays stable as the number of false-positives grows (Figure 3 D). The power of MCL_2 slightly decreases as the rate of false-positives is increased above 30%. However, it stays above 0.9. This suggests that these methods do not: (1) break the clusters into smaller ones, and (2) mix them together as a results of their false-positive connections to each other. This is not true for other clustering methods as their power seemingly converges to a minimum value that is determined by the large clusters that are less structurally affected by the higher rates of false-positive edges. In case of modularity optimizing methods, the lower bound is also affected by the resolution limit. Increasing the number of edges in the graph (by adding false-positive edges), thus has a twofold effect on Louvain and Leiden merging pairs of loosely connected clusters.

AMI score trends slightly differ from power, primarily due to a more pronounced effect of smaller clusters. $MCL_{1.5}$ and Infomap yield less stable results. While MCL_3 and MCL_5 have a similar performance to the top performing methods with a false-positive rate of 5%, their performance declines with higher intensity, resulting in the same pattern as their power score.

False-Negatives Edges

As shown in Figure 3G, the effects of false-negative edges on the power of the algorithms is less pronounced than that of the false-positives edges. While false-negative edges have an adverse effect on the power of MCL and Infomap, they do not affect the power of Louvain and Leiden significantly. Resolution limit works slightly in favor of Louvain and Leiden here. Still, even the lowest power scores of $MCL_{1.5}$, MCL_2, and Infomap, at a false-negative rate of 50%, is 70% higher than the scores of Louvain and Leiden. The effects of false-negative edges on modularity of the graph are also eviden in the modularity score. While their power score decreased, the top performing algorithms gained higher modularity scores. This is the opposite of what happened when the number of false-positives edges grew; causing modularity to have a higher correlation with power and AMI.

Runtimes

Supplementary Figure 11 displays the average amount of time (in seconds) each method took in our experiments to analyze a dataset as the number of clusters in the dataset grew. The runtime for all methods seem to grow quadratically with respect to the number of simulated clusters. Louvain and Leiden were the fastest methods, analyzing datasets with 5,000 clusters in 0.9 and 0.6 of a second, respectively. Infomap, took 191 seconds on average for the same number of clusters, while $MCL_{1.5}$ and MCL_2, took 30 and 15 seconds on average, respectively.

Highly Connected Subgraphs

The DASH algorithm has been a standard tool for IBD mapping in recent years.[4] DASH requires the fine-tunning of two parameters based on the IBD inference performance. This raises a challenge as we do not always have such information *a priori*. Moreover, as the oldest clustering method that we analyzed, it does not scale to the size of our experiments. We ran

HCS, and the other four algorithms, on a set of 750 small graphs, with cluster counts ranging from 100 to 500. While other algorithms took less than half a second on average to analyze graphs with 100 clusters, HCS took 81.6. This number grew quadratically to 5595 seconds to analyze graphs with 500 clusters (Supplementary Figure 11). For the same number of clusters, MCL_2 analysis took only 1 second on average. Our simulations of smaller datasets showed that HCS has a lower statistical power compared to that of Infomap and MCL. The average statistical power of HCS algorithm in these experiments was 0.23 while the top performing algorithm, Infomap had an average score of 0.92.

Performance on Real Data

We next used the PAGE study dataset to compare the algorithms wth real data. First, we ran iLASH over the chromosome 1 genotype data to estimate IBD and generated local IBD graphs using the output. Out of the resulting 8,447 local IBD graphs, we randomly chose 800 ($\sim$ 10%) to cluster using every algorithm. We then calculated the feature-based metric scores of the results. The real dataset results further demonstrate the effects of the resolution limit on Louvain and Leiden. In every population, the two algorithms returned the lowest percentages of node connectivity and highly-connected subgraphs, not able to detect false-positive edges. An inflated percentage of missing intra-cluster edges further proves this. Their total clustering of the PAGE data on chromosome 1 requires 43% additional edges in order to turn all the clusters to cliques, compared to $MCL_{1.5}$ (top performing method in the simulations) which requires 10% less edges. MCL_5 requires only 19.7% additional edges to achieve the same task, 24% less than Louvain and Leiden.

The score gap between Infomap, $MCL_{1.5}$, and MCL_2 on feature-based clustering metrics decreases in the real datasets compared to the simulated ones. This can be partly explained by a lower false-positive rate demonstrated in the high coverage scores achieved by all the methods. To verify this, we trained a linear regressor based on the feature-based metric scores in our simulations to predict false-positive and false-negative rates of the graphs. The linear regressor could predict false-positive and false-negative rates in our simulated graphs with an average error of 2% (std=1%) and 1%(std=2%), respectively. We employed cross validation leaving 20% of the data for testing each round. Using the linear regression model, we estimated that, in our PAGE dataset, the false-positive rate is 2% (std< 1%), and the false-negative rate is 24% (std=3%). Focusing exclusively on the simulated graphs with false-positive and false-negative rates close to the ones estimated for the PAGE study dataset shows a clear superiority for MCL_2 in terms of statistical power. We simulated 100 graphs, each containing 11,000 clusters (the average number of clusters in a PAGE study dataset local IBD graph) and with realistic false-positive/false-negative rates we estimated. In these simulations, MCL_2 yielded the highest average statistical power score of 98.8%, followed by $MCL_{1.5}$ (98.6%), MCL_3 (97.6%) and Infomap (95.5%). Louvain and Leiden had the lowest score at 35%, considerably lower than the MCL methods (See Supplementary Figure 8).

We also calculated the ability of local IBD clusters in recovering rare variants in the Whole-Exome Sequence data obtained from 200,000 participants of UK Biobank and compared it to a set of randomly generated clusters of the same sizes. After extracting local IBD clusters in UK Biobank using our approach, for every rare variant, we tried to find a cluster covering its region that includes the highest number of the carriers of that variant and looked at the fractions of the number of carriers per allele counts. The results are shown in Figure 4. Local IBD clusters outperformed the random clusters by fully recovering 35% of doubletons and tripletons, while randomized clusters fully recovered only 0.01%. For variants with minor allele frequencies between 10-20, real clusters had an average recovery rate of 42% against 7% for the randomized clusters.

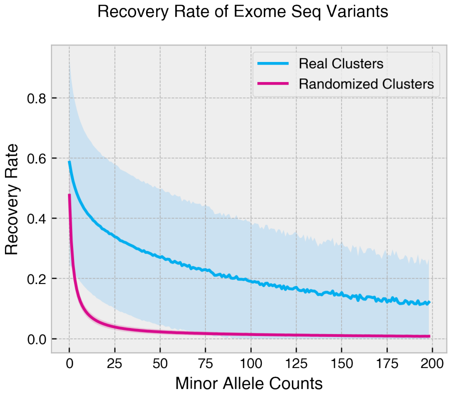

Fig. 4. The recovery rate of local IBD graphs when tagging rare genetic variants captured by whole-exome sequencing data in the UK Biobank compared to a null model with randomized clusters.

Discussion

We proposed a realistic approach to simulate local IBD graphs that addresses distinctive properties of such graphs. It provided us with a ground truth for analyzing a group of scalable clustering algorithms and common clustering metrics for the purpose of local IBD clustering for the first time. We demonstrated that available analyses on clustering algorithms and clustering metrics do not apply to local IBD graphs, further stressing the importance of our analysis. Common clustering metrics cannot be considered sufficient substitutes for power in IBD mapping.

As suggested by Emmons et al,[20] the definition and structure of communities under study should derive the decision on what clustering methods to use. Our real dataset analysis shows various populations may require specific clustering approaches. MCL_2 generally performed better than the other methods in our realistic experiments. However, various datasets and IBD estimation algorithms necessitate dataset specific simulations in order to find the fittest clustering algorithm. We found novel utility for feature-based clustering metrics by using them to enable realistic dataset-specific simulations of local IBD graphs. The simulations determine the fittest clustering algorithm in terms of statistical power.

We showed that both the cluster size distribution of IBD graphs, which is heavily skewed towards smaller clusters, and the size of the dataset could lead some clustering algorithms to aggregate groups of small clusters, specially methods that are based on greedy modularity optimization. Moreover, we found further evidence that the performance of greedy modularity optimizing methods is dependent on the size of the graph being analyzed, making them unpredictable. While IBD mapping can help us understand the genetic origins of some traits, its potential is bound by the capabilities of its clustering approach. Even slight clustering errors can negatively affect the accuracy due to the small size of the local IBD communities.

We plan to utilize our approach to conduct a large IBD mapping analysis in the UK Biobank dataset. We believe distinctive properties of UK Biobank, such as its size, and health record availability, together with power of IBD mapping will help us find novel genetic associations. We plan to add two functionalities to our benchmark algorithm. First, we aim to design a realistic approach to simulate edges weights for the graphs that represent IBD segments length, augmenting local IBD graphs with segment lengths as edge weights can help clustering methods (that support weights) detect false-positives more accurately. The longer the segment, the lower the probability of it being a false-positive edge. Second, we plan to simulate overlapping local IBD graphs, where a group of IBD graphs are merged and processed together to save computing resources. In order to reduce the number local IBD graphs to process, we can aggregate them in groups via dividing the chromosome into windows of static length (for example 0.5 cM). We aim to evaluate clustering algorithms' power in detecting overlapping communities in our benchmark. Simulating these two phenomena requires a genetic coalescence simulation that was outside the scope of the current manuscript.

References

1. E. Hartuv, A. O. Schmitt, J. Lange, S. Meier-Ewert, H. Lehrach and R. Shamir, An algorithm for clustering cdna fingerprints, *Genomics* **66**, 249 (2000).
2. E. Han, P. Carbonetto, R. E. Curtis, Y. Wang, J. M. Granka, J. Byrnes, K. Noto, A. R. Kermany, N. M. Myres, M. J. Barber *et al.*, Clustering of 770,000 genomes reveals post-colonial population structure of north america, *Nature communications* **8**, p. 14238 (2017).
3. G. M. Belbin, S. Wenric, S. Cullina, B. S. Glicksberg, A. Moscati, G. L. Wojcik, R. Shemirani, N. D. Beckmann, A. Cohain, E. P. Sorokin *et al.*, Towards a fine-scale population health monitoring system, *bioRxiv* , p. 780668 (2019).
4. A. Gusev, E. E. Kenny, J. K. Lowe, J. Salit, R. Saxena, S. Kathiresan, D. M. Altshuler, J. M. Friedman, J. L. Breslow and I. Pe'er, Dash: a method for identical-by-descent haplotype mapping uncovers association with recent variation, *AJHG* **88**, 706 (2011).
5. Y. Qian, B. L. Browning and S. R. Browning, Efficient clustering of identity-by-descent between multiple individuals, *Bioinformatics* **30**, 915 (2014).
6. S. R. Browning and E. A. Thompson, Detecting rare variant associations by identity-by-descent mapping in case-control studies, *Genetics* **190**, 1521 (2012).
7. E. E. Kenny, A. Gusev, K. Riegel, D. Lütjohann, J. K. Lowe, J. Salit, J. B. Maller, M. Stoffel, M. J. Daly, D. M. Altshuler *et al.*, Systematic haplotype analysis resolves a complex plasma plant sterol locus on the micronesian island of kosrae, *Proceedings of the National Academy of Sciences* **106**, 13886 (2009).
8. G. M. Belbin, J. Odgis, E. P. Sorokin, M.-C. Yee, S. Kohli, B. S. Glicksberg, C. R. Gignoux, G. L. Wojcik, T. Van Vleck, J. M. Jeff *et al.*, Genetic identification of a common collagen disease in puerto ricans via identity-by-descent mapping in a health system, *Elife* **6**, p. e25060 (2017).
9. S. Papadopoulos, Y. Kompatsiaris, A. Vakali and P. Spyridonos, Community detection in social media, *Data Mining and Knowledge Discovery* **24**, 515 (2012).
10. J. Shi and J. Malik, Normalized cuts and image segmentation, *IEEE Transactions on pattern analysis and machine intelligence* **22**, 888 (2000).
11. Association for Computational Linguistics, *Chinese whispers: an efficient graph clustering algorithm and its application to natural language processing problems* Jan 2006.

12. F. Lamberti, A. Sanna and C. Demartini, A relation-based page rank algorithm for semantic web search engines, *IEEE Transactions on Knowledge and Data Engineering* **21**, 123 (2008).
13. Springer, *SIGNUM: A graph algorithm for terminology extraction* 2008.
14. *Metafac: community discovery via relational hypergraph factorization* 2009.
15. *Detecting strong ties using network motifs* 2017.
16. D. Camacho, A. Panizo-LLedot, G. Bello-Orgaz, A. Gonzalez-Pardo and E. Cambria, The four dimensions of social network analysis: An overview of research methods, applications, and software tools, *arXiv preprint arXiv:2002.09485* (2020).
17. G. Palla, I. Derényi, I. Farkas and T. Vicsek, Uncovering the overlapping community structure of complex networks in nature and society, *nature* **435**, 814 (2005).
18. M. Girvan and M. E. Newman, Community structure in social and biological networks, *Proceedings of the national academy of sciences* **99**, 7821 (2002).
19. A. Lancichinetti, S. Fortunato and F. Radicchi, Benchmark graphs for testing community detection algorithms, *Physical review E* **78**, p. 046110 (2008).
20. S. Emmons, S. Kobourov, M. Gallant and K. Börner, Analysis of network clustering algorithms and cluster quality metrics at scale, *PloS one* **11** (2016).
21. D. J. Watts and S. H. Strogatz, Collective dynamics of 'small-world'networks, *nature* **393**, p. 440 (1998).
22. A.-L. Barabási and R. Albert, Emergence of scaling in random networks, *science* **286**, 509 (1999).
23. G. L. Wojcik, M. Graff, K. K. Nishimura, R. Tao, J. Haessler, C. R. Gignoux, H. M. Highland, Y. M. Patel, E. P. Sorokin, C. L. Avery *et al.*, The page study: how genetic diversity improves our understanding of the architecture of complex traits, *bioRxiv* , p. 188094 (2018).
24. A. Clauset, C. R. Shalizi and M. E. Newman, Power-law distributions in empirical data, *SIAM review* **51**, 661 (2009).
25. S. Fortunato and M. Barthelemy, Resolution limit in community detection, *Proceedings of the national academy of sciences* **104**, 36 (2007).
26. V. D. Blondel, J.-L. Guillaume, R. Lambiotte and E. Lefebvre, Fast unfolding of communities in large networks, *Journal of statistical mechanics: theory and experiment* **2008**, p. P10008 (2008).
27. V. A. Traag, L. Waltman and N. J. van Eck, From louvain to leiden: guaranteeing well-connected communities, *Scientific reports* **9**, 1 (2019).
28. M. Rosvall and C. T. Bergstrom, Maps of random walks on complex networks reveal community structure, *Proceedings of the National Academy of Sciences* **105**, 1118 (2008).
29. S. V. Dongen, Graph clustering by flow simulation, PhD thesis, University of Utrecht Amsterdam, (Netherlands, 2000), pp. ix + 10.
30. A. Lancichinetti and S. Fortunato, Community detection algorithms: a comparative analysis, *Physical review E* **80**, p. 056117 (2009).
31. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research* **12**, 2825 (2011).
32. M. T. Schaub, J.-C. Delvenne, S. N. Yaliraki and M. Barahona, Markov dynamics as a zooming lens for multiscale community detection: non clique-like communities and the field-of-view limit, *PloS one* **7** (2012).
33. R. Shemirani, G. M. Belbin, C. L. Avery, E. E. Kenny, C. R. Gignoux and J. L. Ambite, Rapid detection of identity-by-descent tracts for mega-scale datasets, *Nat Comms* **12**, 1 (2021).