

APLICAÇÃO DE GRAFOS E ALGORITMOS DE AGRUPAMENTO PARA A DISTRIBUIÇÃO DE AUXÍLIOS ESTUDANTIS

2022012333 - CAIO HENRIQUE PESSOA DE LIMA

2022006612 - GUILHERME CEZAR DE SOUZA

2022014937 - HELLEN CRISTINA ANDRADE
NAPOLEAO

2022010320 - LUIS EDUARDO DAMASCENO

2022004912 - OTAVIO HENRIQUE RODRIGUES DE
OLIVEIRA

SMAC03 - GRAFOS

Prof. Rafael Frinhani



INSTITUTO DE
MATEMÁTICA E
COMPUTAÇÃO

UNIFEI - Itajubá



Aplicação de Grafos e Algoritmos de Agrupamento para a Distribuição de Auxílios Estudantis

1 Introdução

Este projeto visa explorar as políticas e práticas implementadas pela Universidade Federal de Itajubá (UNIFEI), para promover a inclusão e diversidade, enfocando especificamente em seu programa de auxílio estudantil, que foi desenhado para auxiliar estudantes em situação de vulnerabilidade. Contudo, a eficiência na distribuição dessas bolsas de estudo é um desafio devido a baixa disponibilidade de assistentes sociais e ao volume de candidatos. O objetivo deste projeto é otimizar o processo de classificação e agrupamento dos candidatos por meio da teoria e métodos de grafos, auxiliando os assistentes sociais nas suas tomadas de decisão e facilitando uma distribuição mais ágil e justa do programa de auxílio estudantil.

1.1 Cenário de Estudo

A Universidade Federal de Itajubá é uma instituição pública de ensino superior que tem como um de seus pilares a inclusão social e a diversidade. Para isso, a universidade conta com a Diretoria de Assuntos Estudantis (DAE), um órgão vinculado à Pró-Reitoria de Graduação e é responsável pela execução do Programa de Assistência Estudantil (PAE), ofertada pelo Programa Nacional de Assistência Estudantil (PNAES), que é uma política do Ministério da Educação que visa auxiliar a permanência de jovens de baixa renda matriculados em cursos de graduação presencial das instituições federais de ensino superior, ao compreender ações que objetivam viabilizar a igualdade de oportunidades entre todos os estudantes e contribuir para a melhoria do desempenho acadêmico, a partir de medidas que buscam combater situações de repetência e evasão.

O auxílio estudantil da DAE é um programa de assistência estudantil oferecido pela Diretoria de Assuntos Estudantis da Universidade Federal de Itajubá e tem como objetivo ampliar e democratizar o acesso e permanência dos discentes da UNIFEI em situação de vulnerabilidade socioeconômica, oferecendo bolsas, apoio pedagógico, recursos tecnológicos, apoio psicológico e acesso ao restaurante universitário. Atuando assim de forma a minimizar os efeitos das desigualdades sociais, reduzir as taxas de evasão e retenção e contribuir para o desempenho acadêmico dos discentes, nos termos dos artigos 1º, 2º e 5º do Decreto nº 7.234 de 19 de julho de 2010 (Programa Nacional de Assistência Estudantil).

Uma das dificuldades enfrentadas pela Diretoria de Assistência Estudantil (DAE), é o problema de distribuição de bolsas do auxílio estudantil, que envolve a análise dos dados socioeconômicos dos candidatos, bem como a verificação da documentação exigida para comprovar a situação de vulnerabilidade. Esse processo é realizado pela Diretoria de Assisten-

cia Estudantil (DAE) da instituição de ensino, que deve avaliar cada caso individualmente e definir o valor e o tipo de auxílio a ser concedido. No entanto, esse processo é demorado e burocrático, podendo levar de 10 a 15 dias para a análise preliminar, mais 3 dias para a análise dos recursos dos candidatos que não concordarem com o resultado.

1.2 Objetivos do Projeto

O propósito primordial deste projeto é conceber uma solução eficaz para a distribuição de auxílios estudantis. A necessidade de tal sistema é impulsionada pelo aumento expressivo na solicitação de assistência financeira por parte dos discentes em situação de vulnerabilidade e pela limitação dos métodos utilizados atualmente.

A solução proposta é fundamentada na aplicação de grafos e algoritmos de agrupamento, modularidade e distância euclidiana para classificar os discentes em grupos distintos, com base em critérios pré-estabelecidos pela Diretoria de Assuntos Estudantis (DAE), órgão responsável pela distribuição dos auxílios. Estes critérios podem englobar, mas não estão restritos a, renda familiar, moradia familiar, moradia do aluno, patrimônio, escolaridade do aluno e dos pais, entre outros.

Os grafos serão empregados para representar a rede de discentes, onde cada vértice simboliza um discente e as arestas representam as relações entre eles, baseadas nos critérios estipulados. O algoritmo de agrupamento, por sua vez, será aplicado para segmentar essa rede em grupos de discentes.

A expectativa é que essa metodologia permita uma distribuição mais equânime e transparente dos auxílios estudantis, assegurando que os recursos sejam direcionados para aqueles que demonstram maior necessidade de acordo com a análise empregada ao resultado provido pelo algoritmo. Adicionalmente, antecipa-se que a solução seja escalável e possa ser adaptada para atender a diferentes contextos e demandas.

Este artigo detalha a metodologia proposta, discutirá sua implementação, modelagem do problema, o tratamento de dados e apresentará uma avaliação de sua eficácia com base em estudos de caso e análises de dados. Por meio deste trabalho, espera-se contribuir para a literatura existente sobre a distribuição de auxílios estudantis e fornecer uma ferramenta valiosa para as instituições de ensino na gestão de distribuição de auxílios estudantis e de recursos de assistência financeira.

2 Referencial Teórico

Na resolução de problemas de redes de grande escala onde há uma grande quantidade de atributos diferentes, encontrar

uma similaridade de interação desses atributos entre os vértices da rede é necessária para realização de agrupamentos de dados para que haja melhor entendimento do grafo possibilitando uma solução. Em [Carvalho \(2018\)](#) é abordado métodos para resolução de problemas de geometria de distância utilizando matriz de distância euclidiana para definição de similaridade. "Dada uma matriz $D \in \mathbb{R}^{n \times n}$ de distâncias, D é uma matriz de distâncias Euclidianas (ou EDM) se para algum inteiro $k > 0$, existem vetores x_1, x_2, \dots, x_n " ([Carvalho, 2018](#), p. 29)

Isto se dá pela fórmula:

$$D_{ij} = ||X_i - X_j||^2, \quad \forall i, j \in \{1, \dots, n\} \quad (1)$$

A matriz de similaridade pode ser utilizada na clusterização hierárquica, "O ponto de partida da clusterização hierárquica é uma matriz de similaridade, cujos elementos x_{ij} indicam a distância do nó i em relação ao nó j . Na identificação da comunidade, a similaridade é extraída da posição relativa dos nós i e j dentro da rede." ([Barabási, 2013](#), p. np)

Como aponta [Aaron Clauset \(2004\)](#) a modularidade é utilizada em detecção de comunidades medindo a qualidade de cada partição onde este valor pode variar entre 0 a 1. Quanto maior for o valor encontrado melhor é a estrutura da comunidade, e quando o valor é 0 indica que esta comunidade se equipara a uma rede aleatória e não possui uma estrutura definida. Porém valores encontrados acima de 0,3 também é um indicativo de de uma estrutura de comunidade na rede. A definição de modularidade se dá por:

$$Q = \frac{1}{2m} \sum_{vm} \left[A_{vw} - \frac{k_v \cdot k_w}{2m} \right] \sigma(c_v, c_w) \quad (2)$$

Um algoritmo que pode ser utilizado para detecção de comunidade é o Fast Greedy, segundo [Pankaj Chejara \(2017\)](#) o algoritmo fast greedy considera inicialmente cada nó como uma comunidade e como se não possuíssem nenhuma aresta, conforme a combinação em pares dessas comunidades é realizada a verificação na mudança da modularidade (ΔQ) e o maior valor é guardado em um maxheap, caso haja uma aresta ligando estes dois elementos as comunidades são agrupadas.

3 Solução da Proposta

Nessa seção será apresentada a metodologia utilizada para a modelagem do problema, passando pelo dataset utilizado, o tratamento dos dados do dataset, criação da matriz de similaridade, o método de modelagem de grafo utilizado e por fim a solução proposta.

3.1 Dataset

O dataset fornecido para o projeto apresenta os estudantes que aplicaram para o programa de auxílio em 2018, contendo as respostas do questionário padrão de análise socioeconômica de 961 aplicantes, o qual foi visualizado e transformado dentro da plataforma Excel. As respostas coletadas são utilizadas como ponderação para a análise feita pela DAE, portanto nesse artigo seguiremos com essa metodologia utilizada por esse órgão.

Os atributos relacionados a cada estudante são resultados do questionário aplicado. Dessa forma, são 13 atributos

para cada aluno que contemplam informações sociais e econômicas. A partir de uma entrevista com um membro da Diretoria de Assistência Estudantil (DAE), foi explicado que dentre todos esses atributos coletados, são escolhidos apenas alguns deles para uma análise numérica, com o objetivo de selecionar os candidatos que têm prioridade econômica. Os demais atributos que têm caráter social serão analisados individualmente pelos assistentes da DAE.

Portanto, neste trabalho, prosseguiremos com o método de análise utilizado pela DAE.

3.2 Tratamento de Dados

Um dos processos mais importantes da modelagem de um problema em Grafos e de qualquer método de detecção de comunidades é o tratamento dos dados, pois a qualidade e a transparência desses dados tem impacto direto no resultado da análise, podendo levar a um resultado incoerente com a realidade ou enviesado. Portanto, será descrito o processo de tratamento de dados deste trabalho.

Foram selecionados quatro atributos: Renda per Capita, Despesas per Capita, Valores de Bens e Escolaridade, os quais são utilizados pela DAE para uma análise padronizada e evitando usar os demais atributos por conta da baixa diversidade de valores, o que não agregaria no contexto de diferenciação dos grupos. Dentre esses atributos, Renda per capita, Despesas per Capita e Valores de Bens são quantitativos, já Escolaridade é um valor qualitativo. Como forma de evitar diversidade de valores, optamos por usar os atributos categóricos disponíveis no dataset. Esses atributos foram categorizados com valores de 1 a 10, onde os indivíduos que têm um desses atributos como categoria 1 estão em um intervalo de valor que os caracteriza como mais necessitados em relação a esse atributo, aqueles de categoria de 2 até 10, têm menor necessidade gradativamente.

Da mesma forma, o atributo de Escolaridade também foi categorizado, porém trata-se de um atributo qualitativo, portanto sua categorização segue a seguinte ordem: 1 = Escola pública, 2 = Filantrópico, 3 = particular > 50%, 4 = particular < 50%, 5 = particular. De forma análoga ao desenvolvido para os demais atributos, aqueles com escolaridade de categoria 1 têm maior prioridade que os demais de categoria 2 ou maior.

Por fim, a categorização dos dados é resumida em:

Tabela 1: Dicionário de dados

NOME	TIPO	VALORES	CATEGORIAS
Renda per Capita	Quantitativo	0 - 292	De 1 a 10
Despesas per Capita	Quantitativo	0 - 1659	De 1 a 10
Valores de bem	Quantitativo	0 - 700000	De 1 a 10
Escolaridade	Qualitativo	Pública, Filantrópica, Particular > 50%, Particular < 50%, Particula	De 1 a 5, respectivamente

Após a seleção dos dados que serão utilizados, é importante que eles passem pelo processo de normalização para um valor correspondente no intervalo de $[0, 1]$, facilitando as próximas etapas de detecção de comunidades. Dessa forma, cada atributo foi normalizado utilizando o método min-max. Para obter o atributo normalizado Y , divide-se a subtração do valor

da categoria do atributo C pelo valor mínimo da categoria C Min, pela subtração da quantidade de categorias do atributo A pelo valor mínimo da categoria C Min. Como para todas as categorias o valor mínimo é 1, C Min é equivalente a 1. Portanto o método usado é dado pela seguinte equação:

$$Y_{ia} = \frac{C_{ia} - 1}{A_{ia} - 1} \quad (3)$$

3.3 Método de Modelagem

A modelagem do problema utilizando grafos também segue um processo. Uma Matriz de Similaridade é uma matriz que descreve a similaridade de dois objetos com relação aos atributos deles e são amplamente utilizadas para modelagem de grafos. Para o desenvolvimento deste artigo foi utilizado o método matemático de Distância Euclidiana, dado pela seguinte fórmula:

$$d(P, Q) = \sqrt{(w_2 - w_1)^2 + (x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (4)$$

A Distância Euclidiana d entre os vértices P e Q é dada pela raiz quadrada do somatório da diferença de um determinado atributo entre os vértices, no caso deste artigo são quatro atributos identificados por w, x, y, z , respectivamente representando os atributos Renda per Capita, Despesas per Capita, Valores de Bens e Escolaridade, após tratamento. Essa distância foi utilizada como medida de similaridade para a matriz de um grafo não direcionado, portanto será usada como peso para as arestas do grafo modelado e será fundamental para os próximos passos deste artigo.

Após o tratamento dos dados, foi criada a matriz de similaridade no formato 961 x 961, servindo como base para a primeira modelagem do problema utilizando grafos. A partir da matriz de similaridade criada em Excel e lida na variável "df" pela biblioteca pandas, foi criado o grafo não direcionado e ponderado g, dado pela seguinte função da biblioteca iGraph:

```
1 g=ig.Graph.Weighted_Adjacency(df.values.tolist(), mode="UNDIRECTED")
```

Como resultado foi obtido o seguinte grafo após distribuição dos vértices e arestas, com tempo de execução de 366 segundos:

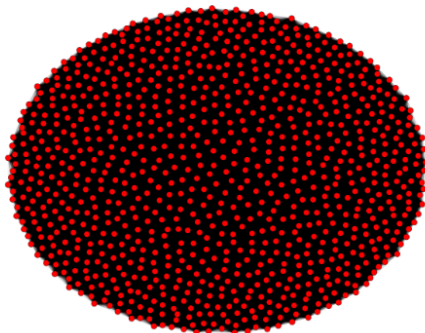


Figura 1: Grafo gerado a partir da matriz de similaridade

3.4 Método de solução

Com base nos resultados da modelagem do problema, a próxima etapa é caracterizada pela aplicação dos algoritmos de detecção de comunidades. Segundo SCHAEFFER (2007), um cluster em um grafo recebe o nome de comunidade, se tratando de uma ferramenta eficiente para problemas que necessitam de agrupamento de dados que detêm certa semelhança. Prosseguindo com os resultados obtidos nos processos anteriores deste trabalho e das informações encontradas no trabalho de SCHAEFFER (2007), uma solução disponível é a Clusterização Hierárquica.

A Clusterização Hierárquica é usada em problemas onde há sobreposição de comunidades, ou seja, vértices do grafo podem fazer parte de mais de uma comunidade. Portanto, aplicando hierarquização às comunidades, cada nível hierárquico pode conter grupos de níveis inferiores a ele, permitindo a interseccionalidade desses vértices.

Também dentro da definição de clusterização hierárquica, existem formas diferentes de abordar o processo de detecção das comunidades, por aglomeração e por divisão. Os métodos aglomerativos se iniciam com cada vértice representando uma comunidade, subsequentemente os vértices que possuem uma maior semelhança são unidos e formam um único cluster, sendo repetido esse processo até que se encontre uma condição de parada e as comunidades são definidas.

De forma semelhante, os métodos divisivos se iniciam com uma grande comunidade formada por todos os vértices e aqueles que apresentam maior semelhança são divididos em grupos menores, até que a condição de parada seja encontrada.

Neste trabalho, foi usado o algoritmo FastGreedy, um algoritmo hierárquico aglomerativo que utiliza a medida de modularidade como condição de parada. Cada união de comunidades no Fast Greedy é decidida com base na união que resultará na maior adição à modularidade do grafo. Desta forma, quando não é possível aumentar a modularidade, o algoritmo é encerrado e as comunidades detectadas são definidas. Um exemplo do funcionamento do algoritmo pode ser representado pelo dendrograma a seguir:

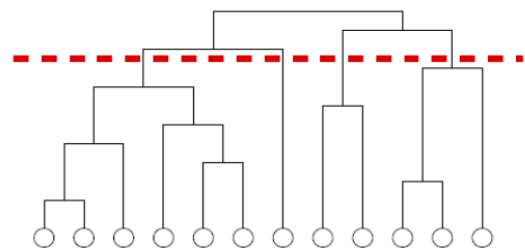


Figura 2: Exemplo de dendrograma

4 Resultados

Nesta seção serão expostos os resultados obtidos a partir da aplicação do algoritmo Fast Greedy na modelagem produzida na seção 3.3. Observa-se que todos os experimentos foram realizados em um computador dotado de um processador Intel(R) Core(TM) i5-10300H CPU @ 2.50GHz, 8GB de memória RAM, sistema operacional Windows 11 Home 64bits (10.0, Compila-

ção 22621). Os modelos e experimentos foram implementados em Python 3.10.13, com as bibliotecas: igraph 0.11.2, matplotlib 3.7.2 e pandas 2.0.3.

Com o objetivo de simplificar os relacionamentos dos indivíduos dentro das comunidades, foi realizado um corte nas arestas do grafo em duas magnitudes, primeiramente considerando apenas aquelas que representassem uma similaridade no mínimo 60% e posteriormente com 70% de similaridade mínima, ambas em relação a aresta de maior peso.

A partir da definição da abordagem para a utilização do algoritmo fast greedy, o seguinte código foi usado:

```
1 # Obtem o maior peso
2 max_weight = max(g.es['weight'])
3 print(max_weight)
4
5 # Cria um subgrafo com arestas que possui
   60% ou mais do maior peso
6 sub_g=g.subgraph_edges(g.es.select(
   weight_ge=max_weight*0.60),
   delete_vertices=False)
```

O trecho de código se inicia pela definição da variável `max_weight`, em seguida é gerado o sub grafo definido por `sub_g`, o qual representa o corte das arestas que não representem no mínimo 60% de similaridade em relação a aresta de maior peso (similaridade), sendo isso repetido para o corte no valor de 70%. Portanto, essa ação é feita com o objetivo de retirar as relações entre vértices que representam uma similaridade muito pequena.

Dessa forma, o próximo passo é a aplicação do algoritmo fast greedy, o qual possui uma função própria na biblioteca iGraph, representado no código a seguir:

```
1 # Cria um dicionário para armazenar as
   comunidades e seus vértices
2 communities = {}
3 for vertex, community in enumerate(
   membership):
4     if community not in communities:
5         communities[community] = []
6         communities[community].append(vertex)
7
8
9 # Remove as comunidades que possuem apenas
   um vértice
10 for community, vertices in list(communities
   .items()):
11     if len(vertices) == 1:
12         del communities[community]
13     membership = [m if m != community
   else None for m in membership]
```

O código inicia criando o dendrograma da aplicação do fast greedy, em seguida ele é transformado em partições do grafo e as comunidades são associadas por meio de uma lista que utiliza os índices como identificador do vértice e seus valores como a comunidade que pertencem. Por fim, é criado um dicionário que associa os vértices às comunidades, além de remover comunidades que tenham apenas um vértice, finalizando o processo de detecção de comunidades.

Para analisar o grafo das comunidades, foi impressa cada comunidade com sua respectiva cor, quantidade de vértices, quantidade de arestas e a lista de vértices pertencentes à comunidade, assim como para os vértices sem comunidade, ou

seja, sem arestas. Esse processo é dado pelo trecho de código a seguir:

```
1 # Imprime os resultados
2 for community, vertices in
   vertices_mapped_to_original.items():
3     # Encontra as arestas que possui ambos
       os vértices na comunidade atual
4     edges = [e for e in sub_g.es if e.
       source in vertices and e.target in
       vertices]
5     # Calcula a quantidade de arestas
6     num_edges = len(edges)
7     color = colors[community % len(colors)]
8     print(f"Comunidade {community} (cor: {
       color}, quantidade de vértices: {
       len(vertices)}, quantidade de
       arestas: {num_edges}): {vertices}")
9
10 # Encontra os vértices que não possui
    comunidade
11 vertices_sem_comunidade = [v for v in
   vertex_labels_original if v not in sum(
   vertices_mapped_to_original.values(),
   [])]
12
13 # Imprime os vértices que não possui
    comunidade
14 print(f"Vértices sem comunidade, quantidade
   de vértices: {len(
   vertices_sem_comunidade)}: {
   vertices_sem_comunidade}")
```

O primeiro experimento realizado utilizou o mínimo de 60% de similaridade entre indivíduos, inicialmente o grafo possuía 961 vértices e 46062 arestas e após a aplicação do algoritmo Fast Greedy, durando em média 68 segundos, foram encontradas 3 comunidades representadas por cores. A primeira comunidade denominada Vermelha apresenta 507 vértices e 13383 arestas, a segunda comunidade denominada Verde possui 378 vértices e 6384 arestas, por fim a comunidade denominada Azul possui 72 e 289 arestas. Foi também criado um grupo de vértices desconexos na cor Branca, com 4 indivíduos.

Analisando a quantidade de atributos semelhantes, a tabela a seguir mostra aqueles que mais aparecem em cada comunidade e sua porcentagem:

Tabela 2: Categorias dominantes para cada atributo no experimento de 60% de similaridade entre vértices

COMUNIDADE	ESCOLARIDADE	RENDA	DESPESA	VALOR DE BENS
Vermelho	1 - 86,78%	3 - 12,82%	2 - 12,62%	1 - 40,63%
Verde	1 - 87,83%	7 - 13,75%	8 - 16,13%	1 - 58,99%
Azul	1 - 84,72%	1 - 26,38%	3 - 22,22%	1 - 34,72%
Branco	3 - 100%	5 - 50%	7 - 50%, (houve empate)	4 - 50%, 5 - 50%, (houve empate)

É possível observar que dentro das comunidades criadas nesse experimento que nos atributos de Escolaridade e Valor Total de Bens, o grupo mais necessitado, de categoria 1, foi predominante em todas as comunidades, havendo variação nas categorias dos atributos apenas em Renda per capita e Despesas per capita onde os grupos predominantes foram de categoria 1 a 7 e 2 a 8, respectivamente. Levando em consideração a

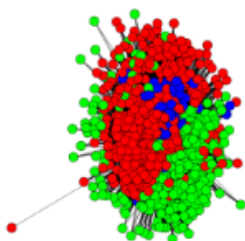


Figura 3: Grafo após a detecção de comunidades do algoritmo fast greedy com 60% de similaridade entre vértices

predominância de categorias mais economicamente vulneráveis, as comunidades Vermelha e Azul têm maior semelhança e contemplam os indivíduos mais vulneráveis.

Além disso, o grupo Branco tem apenas indivíduos com categorias de menor necessidade em relação aos demais grupos, justificando a exclusão desses indivíduos dos demais grupos. Isso leva à conclusão de que as comunidades detectadas não possuem diferenças claras, mostrando que essa abordagem leva à criação de comunidades parecidas e homogêneas.

O segundo experimento levou em consideração apenas as arestas que representassem no mínimo 70% de semelhança, tendo inicialmente 961 vértices e 10861 arestas. Após a aplicação do Fast Greedy, foram detectadas 4 comunidades, também identificadas por cores. A primeira comunidade denominada Vermelha possui 249 vértices e 962 arestas, a segunda comunidade representada por Verde, possui 212 vértices e 273 arestas, a terceira comunidade, da cor Amarela, possui 343 vértices e 1283 arestas, por fim a comunidade Rosa possui 83 vértices e 136 arestas.

Nesse experimento também foi formado um grupo de vértices desconexos da cor Branca totalizando 74 vértices. A tabela a seguir mostra as categorias dominantes para cada atributo:

Tabela 3: Categorias dominantes para cada atributo no experimento de 70% de similaridade entre vértices

COMUNIDADE	ESCOLARIDADE	RENDA	DESPESA	VALOR DE BENS
Vermelho	1 - 86,34%	1 - 20,48%	1 - 12,85%	1 - 45,78%
Verde	1 - 80,18%	9 - 16,50%	10 - 29,71%	1 - 75,47%
Amarelo	1 - 92,71%	5 - 13,99%	3 - 19,82%	1 - 32,65%
Rosa	1 - 97,59%	3 - 38,55%	6 - 27,71%	1 - 79,51%
Branco	1 - 66,21%	4 - 29,72%	7 - 22,97%	4 - 22,97%

Observando os resultados coletados neste experimento, também é possível concluir que os atributos de Escolaridade e Valor Total de bens de categoria 1 foram predominantes em todas as comunidades, porém houve maior variação de predominância nos demais atributos de Renda per capita e Despesas per capita, englobando categorias de 1 a 9 e 1 a 10, respectivamente.

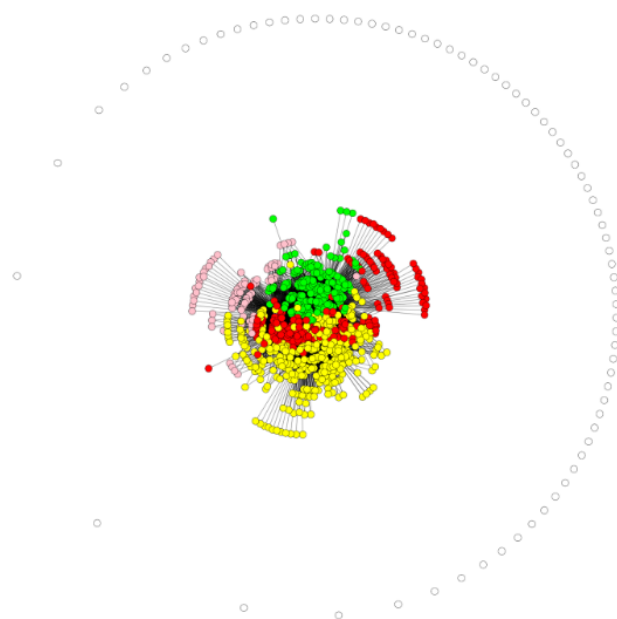


Figura 4: Grafo após a detecção de comunidades do algoritmo fast greedy com 70% de similaridade entre vértices

Neste experimento, a comunidade Vermelha é a que engloba os indivíduos com maior vulnerabilidade econômica, tendo todos os atributos com categoria 1 predominante. Da mesma forma que o experimento anterior, o grupo Branco possui maior divergência dos demais, sendo o único que predomina a categoria 4 no atributo de Valor total de bens.

5 Conclusões

Bolsas de auxílio em instituições federais de ensino superior são ferramentas para amenizar disparidades sociais e econômicas enfrentadas por discentes em situações de vulnerabilidade. Aliado às instabilidades econômicas e sociais crescentes no país, a disponibilidade e a alocação coerentes desses recursos de auxílio se tornam fatores essenciais para que a oportunidade de acesso pleno ao ensino superior seja alcançada.

Após o estudo e observação das condições que a Diretoria de Assuntos Estudantis (DAE) se encontra, devido ao longo tempo da análise manual dos candidatos, fica-se claro a importância do desenvolvimento de uma solução computacional para auxiliar os examinadores na sua tomada de decisão, portanto, desenvolvemos uma solução baseada em grafos que melhora a eficiência do processo de distribuição de bolsas ao realizar o agrupamento dos candidatos. Inicialmente tivemos dificuldade em encontrar métodos de clustering que satisfizesse as condições do nosso problema, por isso foi necessário de um tratamento mais rigoroso dos dados, para que eles pudessem ser utilizados corretamente e coerentemente com os algoritmos de clusterização hierárquicos.

Vale também ressaltar a importância de domínio sobre os conceitos de análise de dados, tanto para o tratamento dos mesmos quanto para compreender os resultados apresentados pelo algoritmo. Tendo os resultados dos dois experimentos, fica evidente que o algoritmo que leva em consideração apenas as arestas com 70% ou mais de similaridade conseguiu identificar comunidades mais diversas em relação às categorias do-

minantes em cada atributo, o que fica evidente pela comunidade Vermelha que possui todos os atributos com categoria predominante 1, ou seja, que representam maior vulnerabilidade econômica. Assim como foi identificada a comunidade Verde, que possui dois atributos com categoria 9 e 10, respectivamente, sendo essa uma comunidade menos vulnerável, assim as comunidades detectadas apresentam maiores diferenças entre si, o que gera uma análise mais concreta e coesa com os objetivos da análise socioeconômica.

O que leva à conclusão de que para essa solução ser plenamente viável para os auxiliares da Diretoria de Assuntos Estudantis (DAE), será necessária a adição de mais etapas automatizadas analisando os resultados do algoritmo e que isso seja disponibilizado aos usuários de forma completa, clara e simples, o que viabiliza a eficiência proposta por este trabalho.

Portanto concluímos que a solução proposta e os resultados coletados são coerentes e promissores com a proposta deste trabalho e tem possibilidade de facilitar o processo rigoroso de análise dos pedidos de auxílio. Entretanto serão necessárias adições de processos e testes para facilitar o trabalho dos auxiliares da DAE e confirmar a coerência dos resultados com o processo de decisão deste órgão.

Referências

- Aaron Clauset, M. E. J. Newman, C. M. (2004). Finding community structure in very large networks. *PHYSICAL REVIEW*.
- Barabási, A.-L. (2013). *Network Science*. Cambridge University Press.
- Carvalho, L. F. (2018). Matrizes de distâncias euclidianas. *TCC-Letícia Figueiredo de Carvalho*, (pp.60).
- Pankaj Chejara, W. W. G. (2017). Comparative analysis of community detection algorithms. *Conference on Information and Communication Technology (CICT)*.
- SCHAEFFER, S. E. (2007). Graph clustering. *Computer science review*.

