UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE CIÊNCIAS EXATAS E DA NATUREZA
PROGRAMA DE PÓS-GRADUAÇÃO EM MATEMÁTICA

André Victor de Albuquerque Araujo

Aplicação de metodologia de hypergrafos em dados da bolsa de valores

Recife

2024

André Victor de Albuquerque Araujo

Aplicação de metodologia de hypergrafos em dados da bolsa de valores

Dissertação apresentada ao Programa de Pós-graduação em Matemática do Centro de Ciências Exatas e da Natureza da Universidade Federal de Pernambuco, como requisito parcial para obtenção do grau de Mestre em Matemática.

Área de Concentração: Geometria

Orientador: Prof. Dr. Fernando Antonio Nóbrega dos Santos

Recife

2024

.Catalogação de Publicação na Fonte. UFPE - Biblioteca Central

Araujo, Andre Victor de Albuquerque.
Aplicação de metodologia de hypergrafos em dados da bolsa de valores / Andre Victor de Albuquerque Araujo. - Recife, 2024.
109f.: il.

Dissertação (Mestrado) - Universidade Federal de Pernambuco, Centro de Ciências Exatas e da Natureza, Pós-Graduação em Matemática, 2024.
Orientação: Fernando Antonio Nóbrega Santos.
Inclui referências e apêndices.

1. Grafos; 2. Hypergrafos; 3. Teoria da informação; 4. Bolsa de Valores; 5. Análise topológica de dados. I. Santos, Fernando Antonio Nóbrega. II. Título.

UFPE-Biblioteca Central

ANDRÉ VICTOR DE ALBUQUERQUE ARAUJO

Aplicação de metodologia de hypergrafos em dados da bolsa de valores

Dissertação apresentada ao Programa de Pós-graduação do Departamento de Matemática da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestrado em Matemática.

Aprovada em: 12/01/2024

BANCA EXAMINADORA

_____

Prof. Dr. Fernando Antonio Nóbrega Santos (Orientador)
Universidade Federal de Pernambuco

_____

Prof. Dr. Eudes Naziazeno Galvão (Examinador Interno)
Universidade Federal de Pernambuco

_____

Prof. Dr. Jones Oliveira de Albuquerque (Examinador Externo)
Universidade Federal de Pernambuco

Aos meus pais Maria Betânia e André Luiz,

por sempre estarem comigo em todos os momentos.

AGRADECIMENTOS

Agradeço todo o apoio de minha família, principalmente aos meus pais, André Luiz Meireles Araujo e Maria Betânia Francisca de Albuquerque Araujo, por tudo que fizeram em minha vida. Também agradeço a todos os professores do Departamento de Matemática que ajudaram de alguma maneira a minha formação acadêmica, em especial, ao professor orientador Fernando Antônio Nóbrega Santos por todos esses anos de estudo e orientação, me acompanhando desde a graduação até o presente momento do mestrado. Vale lembrar também todos os colegas e amigos que consegui durante esses anos, que em vários momentos concordaram e discordaram de minhas opiniões, fortalecendo ainda mais meu senso crítico.

RESUMO

O objetivo desta dissertação é aplicar a metodologia desenvolvida por Santos et al. no artigo "*Emergence of high-order functional hubs in the human brain*" para estudar as inter-relações entre as empresas participantes do $S\&P500$ (abreviação de *Standard & Poor's 500*), índice composto por quinhentos ativos cotados nas bolsas de NYSE ou NASDAQ e qualificados devido ao seu tamanho de mercado, sua liquidez e sua representação de grupo industrial. Para tanto, foi desenvolvido um *pipeline* de processamento de dados para construir redes de alta ordem (*high-order networks*) a partir de séries temporais e aplicá-los no fechamento diário da bolsa $S\&P500$ para caracterizar a comunicação de alta ordem (high-order communication) entre as 55 empresas selecionadas, bem como a construção de hypergrafos uniformes e a utilização de métricas multivariadas de modo a definir pesos nestes hypergrafos. Foram revisados os conceitos básicos de grafos e hypergrafos, Teoria de Redes e medidas de informação multivariada, com especial ênfase dada à sua relação com a sinergia e a redundância, bem como examinar as diferenças entre algumas dessas medidas. Assim, foram confirmadas a aplicabilidade dessa metodologia e a possibilidade de continuar a investigação de surgimentos de *high-order hubs* na bolsa de valores. Por fim, disponibilizamos o *script Python* que permite ao usuário recalcular todas as medidas de informação e resultados apresentados neste trabalho.

Palavras-chaves: Grafos. Hypergrafos. Teoria da Informação. Bolsa de Valores. Análise Topológica de Dados.

ABSTRACT

The purpose of this dissertation is to apply the methodology developed by Santos et al. in the article "*Emergence of high-order functional hubs in the human brain*" to study the interrelationships among companies participating in the $S\&P500$ (Standard & Poor's 500), an index composed of five hundred assets listed on the NYSE or NASDAQ exchanges and qualified based on their market size, liquidity, and industrial group representation. To achieve this, a data processing pipeline was developed to construct high-order networks from time series and apply them to the daily closing of the stock market to characterize high-order communication among the selected 55 companies. This involves building uniform hypergraphs and utilizing multivariate metrics to define weights in these hypergraphs. Basic concepts of graphs and hypergraphs, network theory, and multivariate information measures were reviewed, with particular emphasis on their relation to synergy and redundancy, as well as examining differences between some of these measures. Thus, the applicability of this methodology and the possibility to further investigate the emergence of high-order hubs in the stock market were confirmed. Finally, we provide the Python script that allows the user to recalculate all the information measures and results presented in this work.

Keywords: Graphs. Hypergraphs. Irformation Theory. Stock Exchange. Data Topological Analysis.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

As redes (*networks*) oferecem uma estrutura universal para codificar informações sobre interações em sistemas complexos, que muitas vezes envolvem três ou mais subsistemas de forma emaranhada. Nas abordagens padrão de redes, as interações de alta ordem são frequentemente aproximadas por meio de interações de pares, em que as interações entre três nós A, B e C dentro de uma rede são inferidas através da existência de um *clique*, ou seja, interações par a par, ligando A e B, B e C, e C e A. Estas aproximações, embora razoáveis, muitas vezes não são capazes de capturar todos os aspectos importantes das inter-relações. Por exemplo, redes de comunicação podem apresentar cenários onde a comunicação entre pares não implica necessariamente comunicação simultânea entre as três instâncias (A, B, C) (BAUDOT et al., 2019).

A inclusão de interações e conectividade de alta ordem no estudo de redes pode ser vista como uma forma mais realista e informativa de modelar sistemas complexos. Entretanto, isto produz uma enorme carga combinatória. Por exemplo, em uma rede com N nós tem-se até $\binom{N}{2}$ possíveis interações entre pares e até $\binom{N}{3}$ possíveis interações de tripletos (conjuntos de três nós). Mais geralmente, para $2 \leq k < N$, até $\binom{N}{k}$ possíveis interações entre k-nós em uma rede de alta ordem. Embora considerar a estrutura de alta ordem de sistemas complexos traga consigo uma série de desafios combinatórios, é, no entanto, essencial para uma modelagem mais realista e para uma melhor compreensão destes tipos de sistemas.

Nesse sentido, tem havido um esforço significativo recentemente na quantificação de interações de alta ordem em múltiplos sistemas complexos, da ecologia ao contágio social, em diferentes escalas temporais e espaciais, e para áreas como sincronização de osciladores acoplados, expressão genética e psicometria – apenas para citar alguns. Embora algumas dessas abordagens se concentrem na análise topológica de redes de alta ordem, outras usam a teoria da informação para inferir interdependência estatística de alta ordem em sistemas complexos, incluindo o cérebro.

Neste trabalho, tentamos unir abordagens topológicas e teoria da informação para construir uma estrutura de conectividade de alta ordem, matematicamente bem fundamentada usando interdependências de alta ordem. Mais especificamente, a estrutura aqui apresentada avança a metodologia sobre como quantificar interações de alta ordem e construir uma representação de hypergrafos das interdependências observadas, construir um *pipeline* de processamento de

dados para redes de alta ordem a partir de séries temporais e aplicá-lo no fechamento diário da bolsa $S\&P500$ para caracterizar a comunicação de alta ordem.

Começamos apresentando conteúdos fundamentais para tal aplicação, facilitando assim a interpretação da metodologia. No capítulo 2 são apresentados os conceitos e propriedades inerentes aos grafos e hypergrafos, entropia de Shannon e medidas de informação multivariadas, com ênfase dada a sua relação com a sinergia e a redundância.

No capítulo 3, apresentamos um resumo do artigo "*Emergence of high-order functional hubs in the human brain*"Santos et al. (2023). Neste artigo, foi desenvolvido um *pipeline* de processamento de sinal multivariado para construir redes de alta ordem (*high-order networks*), a partir de séries temporais e aplicá-lo aos sinais de ressonância magnética funcional em estado de repouso (fMRI) para caracterizar comunicação de alta ordem (*high-order communication*) entre regiões cerebrais, bem como a construção de hypergrafos uniformes e a utilização de métricas multivariadas de modo a definir pesos nestes hypergrafos. Com destaque para a metodologia utilizada e os resultados obtidos.

No capítulo 4, apresentamos uma aplicação da metodologia desenvolvida em (SANTOS et al., 2023), com o objetivo de estudar as inter-relações entre as empresas participantes do $S\&P500$ (abreviação de *Standard & Poor's 500*, índice composto por quinhentos ativos cotados nas bolsas de NYSE ou NASDAQ e qualificados devido ao seu tamanho de mercado, sua liquidez e sua representação de grupo industrial).

Por fim, no capítulo 5 fazemos uma análise qualitativa dos resultados obtidos e comentamos sobre perspectivas de outras aplicações.

2 GRAFOS, HIPERGRAFOS E MEDIDAS DE INFORMAÇÃO

Neste capítulo serão introduzidas definições e resultados básicos que serão necessários ao longo deste e dos próximos capítulos, o que nos permitirá compreender o contexto no qual o trabalho aqui proposto será desenvolvido e de nos familiarizarmos com os conceitos e as notações que serão utilizados. Para o estudo de tais conceitos, indicamos ao leitor os livros *"Graph Theory"* (DISTEL, 2005), "*Hypergraphs: combinatorics of finite sets*"(BERGE, 1989) e *"Matrix Analysis"* (HORN.; JOHNSON, 1985).

2.1 GRAFOS

Definição 2.1.1. *Um grafo $G = (V, E)$ é um par de conjuntos, tal que $V = V(G)$ é não vazio e $E = E(G) \subseteq [V]^2$, onde $[V]^2$ denota os subconjuntos com 2 elementos de V. Para evitar ambiguidades de notação, assumimos que $V \cap E = \emptyset$. Os elementos de V são chamados de vértices (ou nós, pontos), os elementos de E são as arestas (ou linhas). A maneira usual de representar um grafo é desenhar um ponto para cada vértice e unindo dois desses pontos por uma linha se esses dois vértices correspondentes formarem uma aresta.*

Apesar de podermos representar um grafo através de uma figura, esta serve apenas para

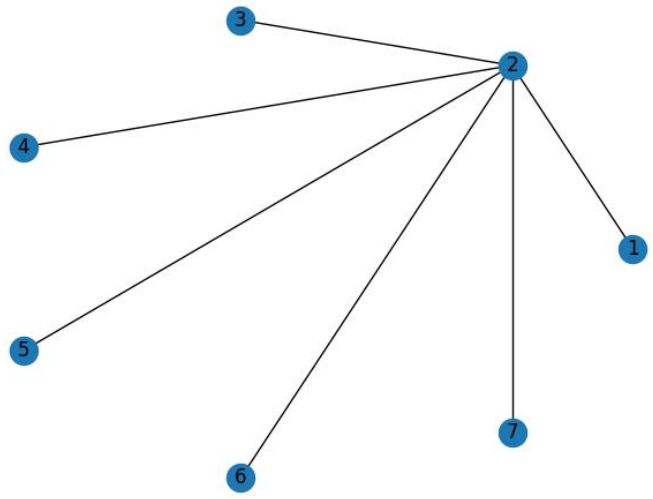


Figura 1 – Grafo $G = (V, E)$, com $V = \{1, 2, 3, 4, 5, 6, 7\}$ e $E = \{\{1, 2\}, \{2, 3\}, \{2, 4\}, \{2, 5\}, \{2, 6\}, \{2, 7\}\}$

orientar nas construções e demonstrações das propriedades deste, visto o que importa é a informação de quais pares de vértices estão conectados por arestas e quais não.

Note ainda que, da forma como foi definido, um grafo $G = (V, E)$ possui no máximo uma

aresta conectando um par de vértices. Na literatura, encontramos uma generalização desta definição de modo que um grafo pode conter mais de uma aresta associadas aos mesmos vértices terminais, bem como arestas associadas a um único vértice (BONDY; MURTY, 1982). Quando esse tipo de situação não acontece, o grafo é dito um *grafo simples*.
Neste trabalho, de acordo com Distel (2005), todo grafo G é grafo simples, ou seja, grafos sem *arestas em paralelo* (arestas que tem os mesmos vértices inicial e final) e sem *laços* (arestas ligando um vértice a si mesmo).
Deste modo, em um grafo (simples) toda aresta $e \in E$ pode ser unicamente representada, a menos de ordem, por um par $e = (u, v)$, com $u, v \in V$ dois vértices distintos, ou ainda, podemos escrever $e = uv$ para representar a aresta ligando os vértices u e v.

Definição 2.1.2. *Um subgrafo H de um grafo $G = (V, E)$ é um grafo formado por subconjuntos dos vértices e das arestas de G, preservando as relações de pertinência de G. Assim, $H = (\tilde{V}, \tilde{E})$, onde $\tilde{V} \subseteq V$ e $\tilde{E} \subseteq E$, de modo que se $e \in \tilde{E}$, com $e = uv$, então $u, v \in \tilde{V}$.*

Definição 2.1.3. *Seja $G = (V, E)$ um grafo. Dizemos que dois vértices $u, v \in V$ são adjacentes se forem unidos por uma arestas $e \in E$, ou seja, existe uma aresta $e \in E$ tal que $e = uv$.*

Algumas situações estão naturalmente associadas a *grafos orientados* (ou *digrafos*), isto é, grafos que possuem arestas com uma direção associada aos dois vértices adjacentes: um vértice será a origem e o outro a extremidade final desta aresta. Por exemplo, o *Instagram* é uma rede social que pode ser pensada como um grafo orientado, tomando os usuários como os vértices e as arestas orientadas conectando usuários a seus seguidores, visto que um usuário pode seguir outro usuário, sem que este necessariamente seja um seguidor do primeiro.

Para grafos não orientados deixamos como referência (BONDY; MURTY, 1982), na qual encontramos a seguinte definição:

Definição 2.1.4. *Um grafo orientado D é uma trinca $D = (V, E, \psi)$ consistindo de um conjunto não vazio $V = V(D)$ de vértices, um conjunto $E = E(D)$ de setas (arestas orientadas), disjunto de V, e uma função de incidência $\psi = \psi_D$ que associa a cada seta de D um par ordenado (não necessariamente distintos) de vértices de D. Se e é uma seta e u e v são vértices tais que $\psi(e) = (u, v)$, então dizemos que e conecta u a v e escrevemos $e = uv$; u é*

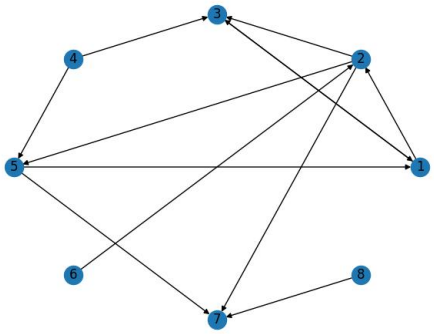


Figura 2 – Grafo orientado

dito o vértice inicial de e, e v é dito seu vértice terminal.
Por conveniência,um grafo orientado também é chamado de digrafo.

Outro conceito importante para um grafo $G = (V, E)$ é o de vizinhança $N(v)$ de um vértice $v \in V$.

Definição 2.1.5. *Seja $G = (V, E)$ um grafo e $v \in V$ um vértice. A vizinhança de v em G é o subconjunto de vértices $N(v) = \left\{u \in V \,\middle|\, \exists\ e \in E \text{ tal que } e = uv\right\}$.* Ou seja, $N(v)$ é o subconjunto de todos os vértices de G que são adjacentes ao vértice v.

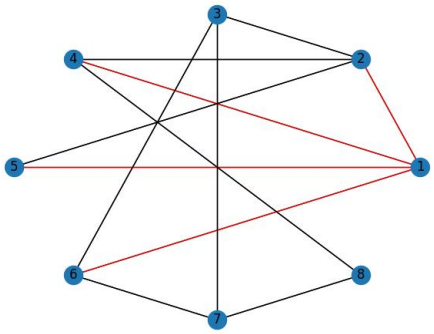


Figura 3 – Vizinhança de um vértice: $N(1) = \{2, 3, 5, 6\}$

Definição 2.1.6. *Seja $G = (V, E)$ grafo. O grau $d(v)$ de um vértice v é a cardinalidade da vizinhança $N(v)$ de v, ou seja, $d(v) = \left|N(v)\right|$, para cada $v \in V$.*

Definição 2.1.7. *Seja $G = (V, E)$ um grafo e sejam $u, v \in V$ dois vértices. Um caminho C, ligando o vértice u ao vértice v, é um subgrafo de G cujos vértices formam uma sequência*

$(v_1, ..., v_n)$, *começando em* $v_1 = u$ *e terminando em* $v_n = v$, *e as arestas são dadas por* $e_i = (v_i, v_{i+1})$, *para* $i = 1, ..., n-1$. *Escrevemos ainda* $C = v_1...v_n$ *para representar o caminho e dizemos que tem tamanho* $n-1$. *Quando* $u = v$, *isto é, o caminho começa e termina em* u, *dizemos que* C *é um ciclo.*

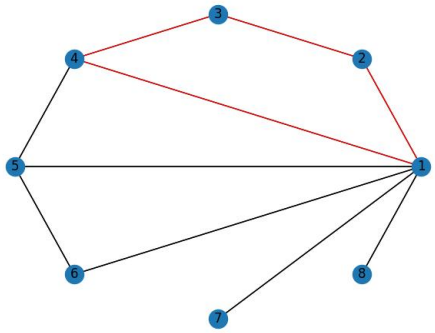


Figura 4 – Caminho e ciclo em um grafo

Definição 2.1.8. *Um grafo ponderado é um grafo* $G = (V, E)$ *munido de uma função peso* $w : E \to N$, *que associa a cada aresta* e *de* G *um valor* $w(e) \in N$ *pertencente ao conjunto numérico* N *(por exemplo,* $\mathbb{N}$, $\mathbb{Z}$ *ou* $\mathbb{R}$*).*

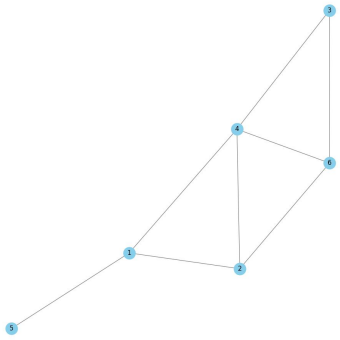
As informações ou dados representados pelos vértices têm suas relações (ligações) expressas a partir das arestas que constituem o grafo, por exemplo, com a utilização de uma função peso. Podemos construir uma matriz mensurando algum parâmetro que permita estimar a influência de um vértice (informação/dado) sobre outro vértice (informação/dado), em um par de vértices (cor)relacionados. A essa matriz chamamos matriz de adjacência, conforme definido a seguir.

Definição 2.1.9. *Seja* $G = (V, E)$ *um grafo (não necessariamente simples), com* $V = \{v_1, v_2, ..., v_n\}$. *A matriz de adjacência* $A(G) = (a_{ij})$ *do grafo* G *é a matriz quadrada de ordem* $n = |G|$, *cuja entrada* a_{ij} *é dada pelo número de arestas que ligam o vértice* v_i *ao vértice* v_j *(*0 *ou* 1 *para grafos simples). Para um grafo ponderado simples* (G, w), *o valor da entrada* a_{ij} *será dado pelo peso da aresta que liga o vértice* v_i *ao vértice* v_j, *ou seja,* $a_{ij} = w(v_i v_j)$.

Modelos envolvendo probabilidades de conexão entre neurônios no córtex somatossensorial de um animal são exemplos de aplicação de matriz de adjacência (MARKRAM et al., 2015). Em

tais modelos, cada neurônio ocupa um vértice de certo grafo ponderado; arestas têm pesos maiores quanto maior a probabilidade de conexão entre dois vértices v_i e v_j quaisquer.

Exemplo 2.1.10. Matriz de adjacência do grafo $G = (V, E)$, com $V = \{1, 2, 3, 4, 5, 6\}$ e $E = \{\{1, 2\}, \{1, 4\}, \{1, 5\}, \{2, 4\}, \{2, 6\}, \{3, 4\}, \{3, 6\}, \{4, 6\}\}$, apresentada Figura 5.



	1	2	3	4	5	6
1	0.0	1.0	0.0	1.0	1.0	0.0
2	1.0	0.0	0.0	1.0	0.0	1.0
3	0.0	0.0	0.0	1.0	0.0	1.0
4	1.0	1.0	1.0	0.0	0.0	1.0
5	1.0	0.0	0.0	0.0	0.0	0.0
6	0.0	1.0	1.0	1.0	0.0	0.0

Figura 5 – Grafo e sua matriz de adjacência

Um outro aspecto que pode ser analisado no estudo de grafos é se todos os vértices estão conectados entre si por arestas.

Definição 2.1.11 *Um grafo é dito completo se, dados dois vértices quaisquer, existe uma aresta que os une. Denotamos por K_n para o grafo completo com n vértices.*

Na abordagem deste trabalho, buscamos obter propriedades de um grafo que produzam informações acerca dos dados analisados. Para tanto, um dos comportamentos a serem observados é se e como ocorrem *agrupamentos* (*clusterings*) de vértices no grafo, observando o quão conectados eles estão entre si. Para isso, faz-se necessário o conceito de *clique* (*click*) de um grafo G.

Definição 2.1.12. *Um clique $(click)$ de um grafo G é um subgrafo completo. Mais precisamente, um r-clique é um clique formado por r vértices, isto é, um K_r subgrafo de G.*

Por outro lado, a definição do que seria um *clique* em um grafo orientado não é tão simples, existindo para isso mais de uma escolha possível. Para maiores detalhes, deixamos como referência SEIDMAN e FOSTER (1978).

Definição 2.1.13. *Uma componente (conexa) de um grafo G é um subgrafo $H = (U, F)$, em que $U \subset V(G)$ é subconjunto de vértices de G para o qual existe, para qualquer par de vértices de U, pelo menos um caminho em G conectando estes vértices; $F \subset E(G)$ contém todas as arestas de G que ligam vértices de U; e maximal com essas propriedades, isto é, de modo que nenhum outro vértice pode ser adicionado ao subconjunto U preservando essa propriedade.*

É importante notar que a matriz de adjacência de um grafo com mais de uma componente pode ser escrita na forma de uma matriz diagonal formada por blocos. Para isso, enumeramos sequencialmente todos os vértices da primeira componente e, somente então, passamos para a segunda componente. Fazemos isso até a última componente.

2.2 HYPERGRAFOS

Nesta seção, apresentamos uma estrutura importante para nossas aplicações: *hypergrafos*. Como referência, indicamos o livro "*Hypergraphs: combinatorics of finite sets*"(BERGE, 1989). Um *hypergrafo* é uma generalização de um grafo sem vértices isolados. Enquanto em um grafo as arestas representam conjuntos com um ou dois vértices, em um hypergrafo, conjuntos quaisquer de vértices podem ser agrupados em uma única "entidade" chamada de *hyperaresta*.

Definição 2.2.1. *Seja $V = \{v_1, v_2, \dots, v_n\}$ um conjunto finito e não vazio, e seja $E = \{E_1, E_2, \dots, E_m\}$, onde cada $E_i \subset V$ é um subconjunto, tal que:*
1. $E_i \neq \emptyset$, *para* $i = 1, \dots, m$; *e*
2. $\bigcup_{i=1}^{m} E_i = V$.
O par $H = (V, E)$ é chamado hypergrafo de ordem $|H| = n$. Os elementos $v_1, v_2, \dots, v_n$ são chamados vértices de H e os subconjuntos $E_1, E_2, \dots, E_m$ de V são chamados de hyperarestas.

Essa estrutura permite representar relações mais complexas entre os vértices do que um grafo, sendo de grande utilidade em diversas aplicações, como modelagens e problemas nos quais as conexões entre elementos não podem ser expressas simplesmente por arestas.

Exemplo 2.2.2. A Figura 6 a seguir ilustra um exemplo de hypergrafo $H = (V, E)$, com

$V = \{1, 2, 3, 5, 6\}$ e $E = \{\{1, 3, 5\}, \{2, 5, 6\}\}$.

Note que para dado i, com $|E_i| > 2$, temos que a representação da hyperaresta E_i é dada por uma curva fechada envolvendo todos os vértices de E_i. Será uma aresta, caso $|E_i| = 2$, ou um laço (*loop*), caso $|E_i| = 1$, como em um grafo.

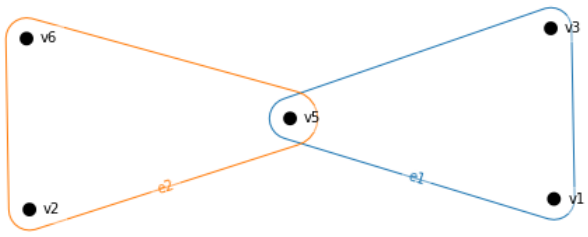


Figura 6 – Representação de um hypergrafo

O hypergrafo é dito simples se $E_i \subseteq E_j \Rightarrow E_i = E_j$, para todo par de hyperarestas. Por exemplo, se um hypergrafo H é simples, com $|E_i| = 2$ para todo i, então H é um grafo simples sem vértices isolados.

Por fim, caso $|E_i| = k$ para todo i, o hypergrafo é dito k-uniforme, ou simplesmente, uniforme.

Em hypergrafos dizemos que dois *vértices* são *adjacentes* se existe uma hyperaresta E_i contendo ambos. Duas *hyperarestas* serão ditas *adjacentes* se a interseção delas for não vazia.

Definição 2.2.3. *A matriz de incidência de um hypergrafo $H = (V, E)$, com $V = \{v_1, v_2, \ldots, v_n\}$ e $E = \{E_1, E_2, \ldots, E_m\}$, é a $m \times n$ matriz (a_{ij}), com número de linhas $m = |E|$ e o número de colunas $n = |V|$. As linhas representam as hyperarestas de H e as colunas representam os vértices de H, de modo que:*

$$a_{ij} = \begin{cases} 1, & \text{se} \quad v_j \in E_i \\ 0, & \text{se} \quad v_j \notin E_i \end{cases}$$

Dado um hypergrafo $H = (V, E)$, podemos construir o hypergrafo dual, denotado por H^*. Para isso, as hyperarestas do hypergrafo original tornam-se os vértices do hypergrafo dual.

Definição 2.2.4. *Para cada hypergrafo $H = (V, E)$, com $V = \{v_1, v_2, \ldots, v_n\}$ e $E = \{E_1, E_2, \ldots, E_m\}$, definimos o hypergrafo dual $H^* = (E^*, V^*)$, cujo conjunto de vértices $E^* = \{e_1, e_2, \ldots, e_m\}$ (que representam respectivamente as hyperarestas $E_1, E_2, \ldots, E_m$) e hy-*

perarestas $V^* = \{V_1, V_2, ..., V_n\}$ *(que representam respectivamente os vértices* $v_1, v_2, \dots, v_n$*), são os conjuntos dados, para cada* j*, por:*

$$V_j = \{ e_i \in E^* \mid v_j \in E_i \text{ em } H\}$$

Claramente cada $V_j \neq \emptyset$ e $\bigcup_{j=1}^{n} V_j = E^*$, visto que $\bigcup_{i=1}^{m} E_i = V$, e, portanto, H^* é um hypergrafo. A matriz de incidência de H^* é a transposta da matriz de incidência do hypergrafo H. Consequentemente, $(H^*)^* = H$. Além disso, se dois vértices v_r e v_s em H são adjacentes, então as hyperarestas correspondentes V_r e V_s em H^* são adjacentes, bem como se duas hyperarestas E_i e E_j em H são adjacentes, então correspondem a vértices e_i e e_j em H^* que também são adjacentes.

2.3 CENTRALIDADE DE AUTOVETOR

A centralidade de autovetor é uma medida de centralidade em grafos baseada na ideia de que a importância de um vértice deve ser diretamente proporcional à soma das importâncias dos vértices aos quais ele está conectado. Isso torna-se de grande utilidade para detectar a importância de um nó, destacando os nós que têm conexões com outros nós que, por sua vez, também são considerados importantes.

Assim, queremos atribuir pontuações a cada um dos vértices de modo que conexões com vértices de pontuação alta contribuam mais para a pontuação de vértice em questão do que conexões iguais a vértices de pontuação baixa.

Para isso, lembramos que dada um grafo G (ou hypergrafo), sua matriz de adjacência A é uma matriz quadrada simétrica com entradas números reais e, portanto, diagonalizável. A medida centralidade de autovetor, proposta por BONACICH (1987), é baseada no conceito de autovalores e autovetores da matriz de adjacência de um grafo G.

Deste modo, considerando $V(G) = \{v_1, ..., v_n\}$, temos que $A = (a_{ij})$, com $a_{ij} = 1$ quando os vértices associados v_i e v_j forem vizinhos, ou $a_{ij} = 0$ caso caso contrário, queremos definir a pontuação de centralidade x_i de um vértice v_i, de modo que:

$$x_i = x(v_i) = \frac{1}{\lambda} \sum_{v \in N(v_i)} x(v) \quad \Rightarrow \quad x_i = \frac{1}{\lambda} \sum_{j=1}^{n} a_{ij}\, x_j \quad \Rightarrow \quad AX = \lambda\, X$$

para $N(v_i)$ a vizinhança do vértice v_i, $X = (x_1, ..., x_n)^t$ o vetor de pontuações e $\lambda > 0$ uma constante.

Estamos interessados em resolver $AX = \lambda X$. Em geral, haverá muitos autovalores diferentes λ's para os quais existe uma solução (autovetor) diferente de zero. Em álgebra linear, o teorema de Perron-Frobenius, provado por Oskar Perron (1907) e Ferdinand Georg Frobenius (1912), afirma que uma matriz real quadrada A com entradas positivas tem um único maior autovalor $\lambda(A)$ e que o autovetor associado tem entradas estritamente positivas. Para matrizes não-negativas (por exemplo, a matriz de adjacência), é necessário considerar uma extensão do Teorema de Perron-Frobenius para o caso em que nem todas entradas da matriz são estritamente positivas.

Teorema 2.3.1. (*Teorema de Perron-Frobenius*) *Seja $A = \left(a_{ij}\right)$ uma matriz real quadrada de ordem n, não nula, não negativa, simétrica e irredutível e sejam $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n$ seus autovalores. Então:*

(i) $\lambda_1 > 0$ e existe um autovetor associado com todas as coordenadas positivas;

(ii) λ_1 é estritamente maior que λ_2;

(iii) $|\lambda_i| \leq \lambda_1$ para $1 \leq i \leq n$.

Demonstração:

(i) Inicialmente observamos que A é uma matriz não negativa e, portanto, $tr(A) = \sum_{i=1}^{n} \lambda_i \geq 0$. Como $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n$, temos que $\lambda_1 \geq 0$. Além disso, $\lambda_1 = 0$ implicaria $\lambda_2 = ... = \lambda_n = 0$ e, consequentemente, a matriz A seria nula. Logo, $\lambda_1 > 0$.

Aplicando o teorema de Rayleigh (HORN.; JOHNSON, 1985), obtemos que

$$\lambda_1 = \max_{||z||=1} z^t A z$$

Seja $y = (y_1, y_2, ..., y_n)^t$ um autovetor unitário associado a λ_1. Então, temos que:

$$\lambda_1 = y^t A y = \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} y_i y_j \leq \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} |y_i|\,|y_j| \leq \lambda_1 = \max_{||z||=1} z^t A z$$

Portanto, $x = (|y_1|, |y_2|, ..., |y_n|)^t$ também é um autovetor unitário associado a λ_1, com todas as coordenadas não negativas. Trocamos y por x.

Supondo, por absurdo, que y possua coordenadas nulas e seja σ uma permutação do conjunto $\{1, 2, ..., n\}$ tal que $|y_{\sigma(i)}| > 0$ para $i \leq m$ e $|y_{\sigma(i)}| = 0$ para $i > m$. Escrevendo a matriz de adjacência segundo a permutação σ obtemos:

$$A_\sigma = \begin{pmatrix} B_{m\times m} & C_{m\times(n-m)} \\ \\ D_{(n-m)\times m} & E_{(n-m)\times(n-m)} \end{pmatrix} \text{ e } y_\sigma = \begin{pmatrix} {y'}_{m\times 1} \\ \\ 0_{(n-m)\times 1} \end{pmatrix}$$

Como as matrizes A e A_σ são semelhantes, temos que possuem os mesmos autovalores, bem como $A_\sigma y_\sigma = \lambda_1 y_\sigma$.

Ou seja,

$$\begin{pmatrix} B & C \\ \\ D & E \end{pmatrix} \begin{pmatrix} {y'}_{m\times 1} \\ \\ 0 \end{pmatrix} = \lambda_1 \begin{pmatrix} {y'}_{m\times 1} \\ \\ 0 \end{pmatrix}$$

Assim, como D é não negativa e y' é positivo, teremos que $D = 0$ e, portanto, A seria uma matriz redutível, contradizendo a hipótese.

Logo, o autovetor y possui todas as coordenadas positivas, como requerido.

(ii) Supondo, por absurdo, que $\lambda_1 = \lambda_2$. Como A é simétrica e, portanto, diagonalizável, podemos escolher dois autovetores ortonormais associados a λ_1 e concluir que

$$x' = \begin{pmatrix} x_1 + |x_1| \\ x_2 + |x_2| \\ \vdots \\ x_n + |x_n| \end{pmatrix} \quad \text{e} \quad y' = \begin{pmatrix} y_1 + |y_1| \\ y_2 + |y_2| \\ \vdots \\ y_n + |y_n| \end{pmatrix}$$

também são autovetores associados a λ_1, onde x_i e y_i são as coordenadas de x e y (se x' for nulo, então trocamos x por $-x$ e recalculamos x'; o mesmo vale para y').

Agora, vemos que $x_i + |x_i| \geq 0$ para todo $1 \leq i \leq n$ e, portanto, $x_i + |x_i| > 0$, pois caso contrário A seria redutível. Da mesma forma, temos que $y_i + |y_i| > 0$, para todo $1 \leq i \leq n$.

Logo, temos que $x_i > 0$ e $y_i > 0$, para todo $1 \leq i \leq n$, e $x^t \cdot y = \sum_{i=1}^{n} x_i y_i > 0$, o que seria um absurdo, pois, por hipótese, x e y são vetores ortogonais.

Portanto, λ_1 é estritamente maior que λ_2.

(iii) Basta notar que, para cada autovalor λ_i, tomamos x um autovetor unitário associado a λ_i, e obtemos que:

$$|\lambda_i| = |x^t A x| = \left| \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j \right| \leq \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} |x_i| \, |x_j| \leq \lambda_1 = \max_{||z||=1} z^t A z$$

Desse modo, $|\lambda_i| \leq \lambda_1$ para $1 \leq i \leq n$. □

Como para um grafo G conexo, sua matriz de adjacência A é irredutível, o Teorema de Perron-Frobenius garante que matrizes de adjacência associadas a grafos conexos possuem o maior

autovalor positivo e, associado a este, um único autovetor unitário positivo. A prova aqui apresentada é encontrada em (FREITAS, 2010).
Assim, a i-ésima entrada do autovetor fornece a pontuação de centralidade relativa do i-ésimo vértice do grafo.

Definição 2.3.2. *Seja G um grafo conexo, com $V(G) = \{v_1, v_2, ..., v_n\}$. A centralidade de autovetor (ou pontuação de centralidade) x_i do vértice v_i é a i-ésima coordenada do autovetor unitário não negativo $x = (x_1, x_2, ..., x_n)^t$ associado ao maior autovalor λ_1 de G, ou seja,*

$$x_i = \frac{1}{\lambda_1} \sum_{j=1}^{n} a_{ij} x_j \,,$$

onde a_{ij} são as entradas da matriz de adjacência de G. O autovalor λ_1 é chamado o índice do grafo G.

Por fim, o *método das potências* é um algoritmo que pode ser usado para encontrar o autovalor dominante λ_1 e seu autovetor associado. O método consiste em determinar o autovalor de maior valor absoluto de uma matriz e seu correspondente autovetor de maneira aproximada (FREITAS, 2010).

Teorema 2.3.3. (*Método das Potências*) *Seja $A \neq 0$ uma matriz real de ordem n e sejam $\lambda_1, \lambda_2, ..., \lambda_n$ seus autovalores e $x_1, x_2, ..., x_n$ seus autovetores correspondentes. Suponha que os autovetores são linearmente independentes, e que:*

$$|\lambda_1| > |\lambda_2| \geq ... \geq |\lambda_n| \,.$$

Seja a sequência y_k definida por:

$$y_{k+1} = Ay_k, \;\; k = 0, 1, 2, ...$$

onde y_0 é um vetor arbitrário não nulo, cuja combinação:

$$y_0 = \sum_{i=1}^{n} c_i x_i$$

tem coeficiente $c_1 \neq 0$.

Então,

$$\lim_{k \to +\infty} \frac{\left(y_{k+1}\right)_r}{\left(y_k\right)_r} = \lambda_1$$

onde o índice r indica a r-ésima coordenada dos vetores y_{k+1} e y_k.
Além disso, $\frac{1}{\lambda_1^k} y_k$ tende a um autovetor associado a λ_1, quando $k \to +\infty$.

Demonstração:

Por hipótese,

$$y_0 = \sum_{i=1}^{n} c_i x_i = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n \,,$$

com $c_1 \neq 0$.

Como $Ax_i = \lambda_i x_i$ e $y_1 = Ay_0$, temos que

$$y_1 = \sum_{i=1}^{n} c_i A x_i = \sum_{i=1}^{n} c_i \lambda_i x_i \quad \Rightarrow \quad y_1 = \lambda_1 \left[c_1 x_1 + c_2 \frac{\lambda_2}{\lambda_1} x_2 + \cdots + c_n \frac{\lambda_n}{\lambda_1} x_n \right]$$

Do mesmo modo, $y_2 = Ay_1 = A^2 y_0$ e, portanto,

$$y_2 = \sum_{i=1}^{n} c_i A^2 x_i = \sum_{i=1}^{n} c_i \lambda_i^2 x_i \quad \Rightarrow \quad y_2 = \lambda_1^2 \left[c_1 x_1 + c_2 \left(\frac{\lambda_2}{\lambda_1} \right)^2 x_2 + \cdots + c_n \left(\frac{\lambda_n}{\lambda_1} \right)^2 x_n \right]$$

Assim, por indução, para todo $k \geq 1$, temos $y_k = Ay_{k-1} = A^k y_0$ e

$$y_k = \lambda_1^k \left[c_1 x_1 + c_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k x_2 + \cdots + c_n \left(\frac{\lambda_n}{\lambda_1} \right)^k x_n \right] .$$

Como, por hipótese $|\lambda_1| > |\lambda_2| \geq \ldots \geq |\lambda_n|$, temos que $\left| \frac{\lambda_i}{\lambda_1} \right| < 1$, para $2 \leq i \leq n$, e, consequentemente, $\left(\frac{\lambda_i}{\lambda_1} \right)^k \to 0$ quando $k \to +\infty$.
Assim, para cada r-ésima coordenada de x_1 não nula, temos que:

$$\lim_{k\to+\infty} \frac{\left(y_{k+1}\right)_r}{\left(y_k\right)_r} = \lim_{k\to+\infty} \frac{\left(A^{k+1} y_0\right)_r}{\left(A^k y_0\right)_r} = \lim_{k\to+\infty} \frac{\lambda_1^{k+1} \left[c_1 x_1 + \sum_{i+2}^{n} c_i \left(\frac{\lambda_i}{\lambda_1} \right)^{k+1} x_i \right]_r}{\lambda_1^k \left[c_1 x_1 + \sum_{i=2}^{n} c_i \left(\frac{\lambda_i}{\lambda_1} \right)^{k} x_i \right]_r} =$$

$$= \lambda_1 \lim_{k\to+\infty} \frac{\left[c_1 x_1 + \sum_{i+2}^{n} c_i \left(\frac{\lambda_i}{\lambda_1} \right)^{k+1} x_i \right]_r}{\left[c_1 x_1 + \sum_{i=2}^{n} c_i \left(\frac{\lambda_i}{\lambda_1} \right)^{k} x_i \right]_r} = \lambda_1 \frac{\left[c_1 x_1 + \sum_{i+2}^{n} c_i \lim_{k\to+\infty} \left(\frac{\lambda_i}{\lambda_1} \right)^{k+1} x_i \right]_r}{\left[c_1 x_1 + \sum_{i=2}^{n} c_i \lim_{k\to+\infty} \left(\frac{\lambda_i}{\lambda_1} \right)^{k} x_i \right]_r} = \lambda_1 \,.$$

Por fim,

$$\lim_{k\to+\infty} \frac{1}{\lambda_1^k} y_k = \lim_{k\to+\infty} \frac{\lambda_1^k \left[c_1 x_1 + \sum_{i=2}^{n} c_i \left(\frac{\lambda_i}{\lambda_1} \right)^{k} x_i \right]}{\lambda_1^k} = c_1 x_1 + \sum_{i=2}^{n} c_i \lim_{k\to+\infty} \left(\frac{\lambda_i}{\lambda_1} \right)^{k} x_i = c_1 x_1 \,.$$

Logo, como $c_1 \neq 0$, quando $k \to +\infty$, temos que $\frac{1}{\lambda_1^k} y_k$ se aproxima de um múltiplo não nulo do autovetor x_1 (ou seja, de um autovetor associado a λ_1).

2.4 TEORIA DA INFORMAÇÃO

Nesta seção abordaremos uma das principais ferramentas utilizada neste trabalho, *a entropia de Shannon*. Uma ferramenta de teoria da informação de grande utilidade e áreas de aplicabilidade, como, por exemplo, em neurociência, na compressão dos dados, na codificação, em sistemas dinâmicos e na codificação genética.

A aplicabilidade da teoria da informação se deve, em grande parte, ao fato de que ela se baseia apenas na distribuição de probabilidade associada a uma ou mais variáveis.

De modo geral, podemos dizer que a teoria da informação utiliza as distribuições de probabilidade associadas aos valores das variáveis para verificar se estes estão ou não correlacionados e, a depender da situação, a forma como a correlação se estabelece, podendo ser aplicada em sistemas lineares ou não.

2.4.1 Medidas da informação

Aplicações da teoria da informação em problemas que envolvem uma e duas variáveis são bem entendidas e bastante implementadas. Contudo, muitos sistemas possuem interações entre três ou mais variáveis. Um exemplo deste tipo de abordagem na Neurociência pode ser observado em (QUIROGA; PANZERI, 2009). Várias medidas de informação foram introduzidas para analisar essas interações multivariadas (MCGILL, 1954; WATANABE, 1960). Frequentemente essas medidas eram introduzidas para medir as chamadas "*sinergia*" e "*redundância*" (confira 2.4.2). Tais medidas de informações multivariadas foram aplicadas em sistemas físicos (MATSUDA, 2000), sistemas biológicos (ANASTASSIOU, 2007) e neurociência. Contudo, essas medidas de informação multivariadas podem diferir de forma significativa e, algumas vezes, de forma sutil.

Neste trabalho, aplicaremos duas métricas de informação multivariada, a *Informação de Interação* (II) (confira 2.5.2) e a *Correlação Total* (TC) (confira 2.5.3), e assim tentar mostrar suas diferentes formas de uso.

Inicialmente, vamos apresentar o conceito de *sinergia e redundância*, bem como essas medidas de informação multivariadas podem ser aplicadas na análise de dados. Além de explicar a teoria por trás dessas métricas, observando em específico duas interações: aquelas que existem dentro de um grupo de variáveis e aquelas que existem entre um grupo de variáveis e outra variável de destino.

Para Timme et al. (2014), é importante ressaltar que, como a teoria da informação utiliza distribuições de probabilidade, experimentos de codificação envolvendo n variáveis e experimentos de rede envolvendo a atividade espontânea de n variáveis são equivalentes em termos da estrutura da distribuição de probabilidade e, portanto, indistinguíveis em termos da aplicação da teoria da informação. Claramente, a escolha da medida de informação e a forma como ela é aplicada ao sistema impactará diretamente nas conclusões que podem ser tiradas da análise. No entanto, em termos da estrutura das distribuições de probabilidade necessárias, esses dois tipos de experimentos são equivalentes. Como resultado, eles podem ser tratados igualmente com as medidas de informação multivariadas que iremos discutir.

Os dados brutos são coletados de alguma fonte, por exemplo, uma variável controlada experimentalmente, valores de ações, genes ou alguma outra fonte. Em seguida, os dados são processados (detecção de pico) e convertidos em distribuições de probabilidade conjuntas. Dependendo do tipo de dado que está sendo analisado, esse processo geralmente envolve a conversão dos dados em estados discretos, bem como das unidades temporais.

Uma vez obtidas essas distribuições de probabilidade conjunta, elas são passadas pela medida de informação escolhida para obter um resultado final.

Por fim, as medidas de informação que discutiremos podem ser definidas diretamente em termos de entropia e informação mútua, como veremos a seguir.

2.4.2 Sinergia e Redundância

Um tópico crucial relacionado com as medidas de informação multivariadas é a distinção entre sinergia e redundância. Muitas das medidas de informação propostas pretendem medir sinergia ou redundância, embora os significados precisos destas não tenham sido acordados (BRENNER et al., 2000; WILLIAMS; BEER, 2010). Para um tratamento recente da sinergia neste contexto, indicamos Griffith e Koch (2014).

Para começar a entender a sinergia, podemos usar um sistema simples. Suponha que duas variáveis X_1 e X_2 fornecem uma informação sobre uma outra variável Y. Ou seja, se conhecemos os estados de X_1 e X_2, então sabemos algo sobre o estado de Y. Assim, dizemos que a parte dessa informação que não é fornecida por conhecer X_1 sozinho e X_2 sozinho é fornecida sinergicamente por X_1 e X_2. A *sinergia* é a informação de bônus recebida por conhecer X_1 e X_2 juntos, em vez de separadamente.

Podemos adotar uma abordagem inicial semelhante à redundância. Novamente, suponha que

X_1 e X_2 forneçam alguma informação sobre Y. Dizemos que a parte comum da informação que X_1 fornece sozinha, bem como a informação que X_2 fornece sozinha, é fornecida em redundância por X_1 e X_2. A *redundância* é a informação recebida de X_1 e X_2.
Para Timme et al. (2014), essas definições imprecisas podem parecer bastante claras, mas, ao tentar medir essas quantidades, medidas distintas podem produzir resultados diferentes. Com base no fato de que o objetivo geral não foi claramente definido, não se pode dizer que uma dessas medidas seja a "correta". Cada medida tem seu próprio uso e suas limitações.

2.5 MEDIDAS DE INFORMAÇÃO MULTIVARIADA

Agora vamos estudar algumas dentre as várias medidas teóricas de informação multivariada que foram introduzidas anteriormente. Uma observação importante é o fato de que os nomes e as notações usados na literatura não têm sido consistentes. Tentaremos esclarecer a discussão tanto quanto possível, listando nomes alternativos quando for apropriado. Iremos nos referir a uma medida de informação por seu nome original até onde sabemos, com a respectiva fonte.

2.5.1 Entropia de Shannon

Grandezas teóricas da informação envolvendo uma e duas variáveis são bem definidas e seus resultados são bem compreendidos. Com respeito à distribuição de probabilidade $p(x)$ de uma variável X, a medida canônica é a entropia de Shannon $H(x)$ (COVER; THOMAS, 2006). A entropia de Shannon é dada por:

$$H(x) = -\sum_{x \in X} p(x)\ \log(p(x)) \tag{2.1}$$

e mede a quantidade de incerteza que está presente na distribuição de probabilidade.
Note que, da forma que é definida, quando a distribuição de probabilidade estiver concentrada perto de um valor, a entropia será baixa, enquanto que no caso de uma distribuição de probabilidade uniforme, a entropia será maximizada.
Ao examinar a relação entre duas variáveis X e Y, a *informação mútua* $I(X, Y)$ mede a quantidade de informação fornecida sobre uma das variáveis ao conhecer o valor da outra

(COVER; THOMAS, 2006). A informação mútua é dada por:

$$\begin{aligned} I(X;Y) = &\ H(X) + H(Y) - H(X,Y) \\ = &\ H(X) - H(X|Y) \\ = &\ H(Y) - H(Y|X) \end{aligned} \tag{2.2}$$

nas quais a entropia conjunta $H(X,Y)$ é dada por:

$$H(X,Y) = -\sum_{x\in X}\sum_{y\in Y} p(x,y)\log\Big(p(x,y)\Big) \tag{2.3}$$

e a entropia condicional $H(X|Y)$ é dada por:

$$\begin{aligned} H(X|Y) = &\ \sum_{y\in Y} p(y)\, H(X|y) \\ = &\ \sum_{y\in Y} p(y) \sum_{x\in X} p(x|y)\log\left(\frac{1}{p(x|y)}\right) \\ = &\ \sum_{y\in Y}\sum_{x\in X} p(x,y)\log\left(\frac{1}{p(x|y)}\right) \end{aligned} \tag{2.4}$$

De fato, as igualdades acimas são consequências da seguinte proposição:

Proposição 2.5.1.1. (*Regra da Cadeia*)

$$H(X,Y) = H(X) + H(Y|X) \tag{2.5}$$

Demonstração:

$$\begin{aligned} H(X,Y) = &\ -\sum_{x\in X}\sum_{y\in Y} p(x,y)\log\Big(p(x,y)\Big) \\ = &\ -\sum_{x\in X}\sum_{y\in Y} p(x,y)\log\Big(p(x)p(y|x)\Big) \\ = &\ -\sum_{x\in X}\sum_{y\in Y} p(x,y)\log\Big(p(x)\Big) - \sum_{x\in X}\sum_{y\in Y} p(x,y)\log\Big(p(y|x)\Big) \\ = &\ -\sum_{x\in X}\left(\sum_{y\in Y} p(x,y)\right)\log\Big(p(x)\Big) - \sum_{x\in X}\sum_{y\in Y} p(x,y)\log\Big(p(y|x)\Big) \\ = &\ -\sum_{x\in X} p(x)\log\Big(p(x)\Big) - \sum_{x\in X} p(x)\sum_{y\in Y} p(y|x)\log\Big(p(y|x)\Big) \\ = &\ H(X) + H(Y|X) \end{aligned}$$

□

A informação mútua pode ser usada como uma medida das interações entre mais de duas variáveis, agrupando as variáveis em conjuntos e tratando cada conjunto como uma única variável com valor de vetor.

Dessa forma, a informação mútua pode ser usada para medir as interações entre um grupo de variáveis e uma variável alvo. Por exemplo, a informação mútua pode ser calculada entre Y e o conjunto $S = \left\{X_1, X_2\right\}$ da seguinte forma:

$$I(Y;S) = \sum_{\substack{y\in Y \\ x_1\in X_1 \\ x_2\in X_2}} p(y,x_1,x_2)\,\log\left(\frac{p(y,x_1,x_2)}{p(y)p(x_1,x_2)}\right) \tag{2.6}$$

No entanto, quando a informação mútua é considerada como na equação anterior, não é possível separar as contribuições de cada variável X_i individualmente no conjunto S. Ainda assim, variando o número de variáveis em S, a informação mútua na equação anterior pode ser usada para medir o ganho ou perda de informação sobre Y por essas variáveis em S.

Nesse contexto, (BETTENCOURT; GINTAUTAS; HAM, 2008) usaram a informação mútua entre uma variável (no caso deles, a atividade de um neurônio) e muitas outras variáveis consideradas em conjunto (no caso deles, as atividades de um grupo de outros neurônios) para examinar a relação entre a quantidade de informação que o grupo de neurônios forneceu sobre um único neurônio e quantidade de informação para um número de neurônios considerados no grupo.

A informação mútua pode ser condicionada a uma terceira variável para produzir a informação mútua condicional (COVER; THOMAS, 2006) e é dada por:

$$\begin{aligned} I(X;Y|Z) &= \sum_{z\in Z} p(z) \sum_{\substack{x\in X \\ y\in Y}} p(x,y|z)\,\log\left(\frac{p(x,y|z)}{p(x|z)p(y|z)}\right) \\ &= \sum_{\substack{x\in X \\ y\in Y \\ z\in Z}} p(x,y,z)\,\log\left(\frac{p(z)p(x,y,z)}{p(x,z)p(y,z)}\right) \end{aligned} \tag{2.7}$$

De fato, temos que

$$\begin{aligned} I(X;Y|Z) &= \sum_{z\in Z} p(z) \sum_{\substack{x\in X \\ y\in Y}} p(x,y|z)\,\log\left(\frac{p(x,y|z)}{p(x|z)p(y|z)}\right) \\ &= \sum_{\substack{x\in X \\ y\in Y \\ z\in Z}} p(z)p(x,y|z)\,\log\left(\frac{\left(p(z)\right)^2 p(x,y|z)}{p(z)p(x|z)p(z)p(y|z)}\right) \\ &= \sum_{\substack{x\in X \\ y\in Y \\ z\in Z}} p(x,y,z)\,\log\left(\frac{p(z)p(x,y,z)}{p(x,z)p(y,z)}\right) \end{aligned}$$

A informação mútua condicional mede a quantidade de informação que uma variável fornece sobre uma segunda variável quando uma terceira variável é conhecida. Observe que é possível que a informação mútua condicional $I(X;Y|Z)$ seja maior ou menor que a informação mútua $I(X;Y)$.

Vejamos um exemplo da aplicação da entropia de Shannon em um conjunto de dados:

Exemplo 2.5.1.2. Considere uma sequência de DNA do cromossomo Y, na qual temos combinações de nucleotídios A, C, T e G.

Vamos aplicar a entropia de Shannon em um pequeno segmento de cromossomo Y, para o qual queremos quantificar sua complexidade.

Para isso, podemos fatiar esse segmento de sequência em janelas (sub-segmentos sequenciais) de tamanho, digamos 250, e andando de 50 em 50 até chegarmos no último nucleotídio, e calculamos a frequência desses elementos em cada janela, ganhando assim, para cada janela, $P(A)$, $P(C)$, $P(T)$ e $P(G)$.

Em cada janela, podemos calcular a entropia $H(Y)$, onde aqui Y denota a sequência do sub-segmento do cromossomo Y.

Obtemos, portanto, para cada janela, a seguinte expressão:

$$\begin{aligned} H(Y) = & - \sum_{y \in \Big\{A,C,T,G\Big\}} p(y)\, \log\Big(P(y)\Big) \\ = & -P(A)\, \log\Big(P(A)\Big) - P(C)\, \log\Big(P(C)\Big) \\ & -P(T)\, \log\Big(P(T)\Big) - P(G)\, \log\Big(P(G)\Big) \end{aligned} \tag{2.8}$$

Note que para cada janela iremos calcular a entropia de Shannon, ao final do processo podemos plotar tais números em um gráfico, e observar se existem trechos distintos ou chamativos. Com essa métrica seríamos capazes de saber se em uma sequência de DNA qual região do cromossomo é mais ou menos surpreendente, no sentido de ter maior ou menor conteúdo de informação.

2.5.2 Informação da Interação (II)

A primeira tentativa de quantificar a relação entre três variáveis em uma distribuição de probabilidade conjunta foi a *informação de interação* II, introduzida em (MCGILL, 1954). Ele tenta estender o conceito de informação mútua como a informação obtida sobre uma variável

conhecendo a outra. A informação de interação é dado por:

$$\begin{aligned} II(X;Y;Z) = &\ I(X;Y|Z) - I(X;Y) \\ = &\ I(X;Z|Y) - I(X;Z) \\ = &\ I(Z;Y|X) - I(Z;Y) \end{aligned} \tag{2.9}$$

De fato, temos as igualdades como consequência das seguintes proposições:

Proposição 2.5.2.1.

$$I(X;Y|Z) - I(X;Z|Y) = I(X;Y) - I(X;Z) \tag{2.10}$$

Demonstração:

$$\begin{aligned} I(X;Y|Z) - I(X;Z|Y) = &\ \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\left(\log\left(\frac{p(z)p(x,y,z)}{p(x,z)p(y,z)}\right) - \log\left(\frac{p(y)p(x,y,z)}{p(x,y)p(y,z)}\right)\right) \\ = &\ \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\,\log\left(\frac{p(z)p(x,y)}{p(y)p(x,z)}\right) \\ = &\ \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\left(\log\left(\frac{p(x,y)}{p(x)p(y)}\right) - \log\left(\frac{p(x,z)}{p(x)p(z)}\right)\right) \\ = &\ \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\,\log\left(\frac{p(x,y)}{p(x)p(y)}\right) - \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\,\log\left(\frac{p(x,z)}{p(x)p(z)}\right) \\ = &\ \sum_{\substack{x\in X\\ y\in Y}}\left(\sum_{z\in Z} p(x,y,z)\right)\log\left(\frac{p(x,y)}{p(x)p(y)}\right) - \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\,\log\left(\frac{p(x,z)}{p(x)p(z)}\right) \\ = &\ \sum_{\substack{x\in X\\ y\in Y}} p(x,y)\log\left(\frac{p(x,y)}{p(x)p(y)}\right) - \sum_{\substack{x\in X\\ z\in Z}}\left(\sum_{y\in Y} p(x,y,z)\right)\log\left(\frac{p(x,z)}{p(x)p(z)}\right) \\ = &\ \sum_{\substack{x\in X\\ y\in Y}} p(x,y)\,\log\left(\frac{p(x,y)}{p(x)p(y)}\right) - \sum_{\substack{x\in X\\ z\in Z}} p(x,z)\,\log\left(\frac{p(x,z)}{p(x)p(z)}\right) \\ = &\ I(X;Y) - I(X;Z) \end{aligned}$$

□

Proposição 2.5.2.2.

$$I(X;Y|Z) - I(Z;Y|X) = I(X;Y) - I(Z;Y) \tag{2.11}$$

Demonstração:

$$
\begin{aligned}
I(X;Y|Z) - I(Z;Y|X) = & \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\left(\log\left(\frac{p(z)p(x,y,z)}{p(x,z)p(y,z)}\right) - \log\left(\frac{p(x)p(x,y,z)}{p(x,y)p(x,z)}\right)\right)\\
= & \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\,\log\left(\frac{p(z)p(x,y)}{p(x)p(y,z)}\right)\\
= & \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\left(\log\left(\frac{p(x,y)}{p(x)p(y)}\right) - \log\left(\frac{p(y,z)}{p(y)p(z)}\right)\right)\\
= & \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\,\log\left(\frac{p(x,y)}{p(x)p(y)}\right) - \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\,\log\left(\frac{p(y,z)}{p(y)p(z)}\right)\\
= & \sum_{\substack{x\in X\\ y\in Y}}\left(\sum_{z\in Z} p(x,y,z)\right)\log\left(\frac{p(x,y)}{p(x)p(y)}\right) - \sum_{\substack{x\in X\\ y\in Y\\ z\in Z}} p(x,y,z)\,\log\left(\frac{p(y,z)}{p(y)p(z)}\right)\\
= & \sum_{\substack{x\in X\\ y\in Y}} p(x,y)\log\left(\frac{p(x,y)}{p(x)p(y)}\right) - \sum_{\substack{y\in Y\\ z\in Z}}\left(\sum_{x\in X} p(x,y,z)\right)\log\left(\frac{p(x,z)}{p(x)p(z)}\right)\\
= & \sum_{\substack{x\in X\\ y\in Y}} p(x,y)\,\log\left(\frac{p(x,y)}{p(x)p(y)}\right) - \sum_{\substack{y\in Y\\ z\in Z}} p(y,z)\,\log\left(\frac{p(x,z)}{p(x)p(z)}\right)\\
= & \; I(X;Y) - I(Z;Y)
\end{aligned}
$$

□

Usando o fato de que a informação mútua condicional pode ser maior ou menor que a informação mútua para o mesmo conjunto de variáveis, a informação de interação pode ser positiva ou negativa. Da informação de interação, McGill disse "Nós vemos que $II(X;Y;Z)$ é o ganho (ou perda) na informação da transmitida entre quaisquer duas das variáveis, devido ao conhecimento adicional da terceira variável" (MCGILL, 1954).

A informação de interação também podem ser escritas como:

$$
II(X;Y;Z) = \; I(X,Y;Z) - I(X;Z) - I(Y;Z) \tag{2.12}
$$

A informação de interação tem sido amplamente utilizada na literatura e referida como a sinergia (GAT; TISHBY, 1999; BRENNER et al., 2000; SCHNEIDMAN et al., 2003; ANASTASSIOU, 2007) na forma dada pela equação 2.12 ou como o índice de redundância-sinergia (CHECHIK et al., 2001). Os termos "sinergia" e "redundância" são utilizados por interpretarmos um resultado

de informação de interação positivo como implicado por uma interação sinérgica entre as variáveis e um resultado de informação de interação negativo como implicado por uma interação redundante entre as variáveis.

Assim, assumimos esta interpretação da informação de interação e que a informação de interação mede corretamente interações multivariadas, então sinergia e redundância serão consideradas qualidades mutuamente exclusivas das interações entre variáveis.

Observe que as informações de interação não diferenciam teoricamente entre suas três variáveis de entrada. A equação 2.12 está estruturada de forma que aparentemente as variáveis X e Y estão sendo relacionadas a Z, mas, de acordo com a equação 2.9, podemos ver que permutando as variáveis o resultado será o mesmo.

Desta forma, a informação de interação mede as interações entre um grupo de variáveis, ao contrário das interações entre um grupo de variáveis e uma variável alvo.

Por fim, podemos reescrever a equação 2.12 da informação de interação em termos da entropia de Shannon e das entropias conjuntas das variáveis:

$$\begin{aligned} II(X;Y;Z) = \quad & -H(X) - H(Y) - H(Z) \\ & +H(X,Y) + H(X,Z) + H(Y,Z) \\ & -H(X,Y,Z) \end{aligned} \tag{2.13}$$

De fato,

$$\begin{aligned} & II(X;Y;Z) = I(X,Y;Z) - I(X;Z) - I(Y;Z) \\ & = \Big(H(X,Y) + H(Z) - H(X,Y,Z)\Big) - \Big(H(X) + H(Z) - H(X,Z)\Big) \\ & -\Big(H(Y) + H(Z) - H(Y,Z)\Big) \\ & = -H(X) - H(Y) - H(Z) + H(X,Y) + H(X,Z) + H(Y,Z) - H(X,Y,Z) \end{aligned}$$

Isto nos leva a uma generalização da informação de interação para n variáveis (JAKULIN; BRATKO, 2004): se $S = \{X_1, X_2, ...X_n\}$, então a informação de interação é

$$II(S) = -\sum_{T \subseteq S} (-1)^{|S|-|T|} H(T) \tag{2.14}$$

Nessa equação, T é um subconjunto de S e $|T|$ denota o tamanho do subconjunto de S. Nela, é evidente que a informação de interação trata todas as variáveis de entrada igualmente e mede as interações entre todas as variáveis para qualquer número de variáveis de entrada.

Uma medida semelhante à informação de interação foi introduzida por A. J. Bell e é chamada de co-informação CI (BELL, 2003). É dada pela seguinte expansão:

$$CI(S) = -\sum_{T \subseteq S} (-1)^{|T|} H(T) = -(-1)^{|S|} \sum_{T \subseteq S} (-1)^{|S|-|T|} H(T) = (-1)^{|S|} II(S) \tag{2.15}$$

Claramente, a co-informação é igual à informação de interação, quando S contém um número par de variáveis, e é igual a menos a informação de interação, quando S contém um número ímpar de variáveis. Então, para o caso de três variáveis, a co-informação se torna:

$$CI(X;Y;Z) = I(X;Y) - I(X;Y|Z) = I(X;Z) + I(Y;Z) - I(X;Y;Z) \tag{2.16}$$

Como a co-informação está diretamente relacionada à informação de interação para sistemas com qualquer número de variáveis, não apresentamos resultados para ela. A co-informação também é conhecida como informação mútua generalizada (MATSUDA, 2000).

2.5.3 Correlação Total (TC)

A informação de interação encontra sua base conceitual ao estender a ideia de informação mútua como a informação obtida sobre uma variável quando a outra variável é conhecida. Alternativamente, poderíamos estender a ideia da informação mútua como a divergência de Kullback-Leibler entre a distribuição conjunta e o modelo independente. Se fizermos isso, chegaremos à correlação total TC, introduzida em (WATANABE, 1960), dada por:

$$TC(S) = \sum_{\vec{x} \in S} p(\vec{x}) \log \left(\frac{p(\vec{x})}{p(x_1)p(x_2)...p(x_n)} \right) \tag{2.17}$$

Na última equação, $\vec{x}$ é um vetor contendo estados individuais das variáveis X.
Assim como na informação de interação, a correlação total também trata todas as variáveis de entrada igualmente, portanto, ela mede as interações entre um grupo de variáveis.
A correlação total também pode ser escrita em termos de entropias como:

$$TC(S) = \left(\sum_{X_i \in S} H(X_i) \right) - H(S) \tag{2.18}$$

Nesta forma, a correlação total também é conhecida como a multi-informação (SCHNEIDMAN et al., 2003).
Usando a equação 2.2 (a equação que relaciona $I(X;Y)$ e as entropias de Shannon), podemos reescrever a correlação total em termos de informações mútuas:

$$TC(S) \ = \ \sum_{i=2}^{n} I(X_1, ..., X_{i-1}; X_i) \tag{2.19}$$

De fato,

$$\sum_{i=2}^{n} I(X_1, \dots, X_{i-1}; X_i) = I(X_1; X_2) + I(X_1, X_2; X_3) + \cdots + I(X_1, \dots, X_{n-1}; X_n)$$
$$= \Big(H(X_1) + H(X_2) - H(X_1, X_2)\Big) + \Big(H(X_1, X_2) + H(X_3) - H(X_1, X_2, X_3)\Big) + \cdots +$$
$$+ \Big(H(X_1, X_2, ..., X_{n-1}) + H(X_n) - H(X_1, X_2, ..., X_n)\Big)$$
$$= H(X_1) + H(X_2) + \cdots + H(X_n) - H(S) = TC(S)$$

2.5.4 Correlação Total Dual

Depois que a correlação total foi introduzida, uma medida com uma estrutura semelhante, chamada de correlação total dual DTC, foi introduzida por T. Han (HAN, 1975; HAN, 1978). Para $S = \Big\{X_1, X_2, \dots, X_n\Big\}$, a correlação total dual é dada por:

$$DTC(S) = \left(\sum_{X_i \in S} H(S \setminus X_i)\right) - (n-1)H(S) \tag{2.20}$$

Nesta equação, $S \setminus X_i$ é o subconjunto de S para o qual X_i foi removido e $n = |S|$ é o número de variáveis em S. Assim como a correlação total, a correlação total dual também mede as interações dentro de um grupo de variáveis e trata todas as variáveis de entrada igualmente. De acordo com Abdallah e Plumbley (2010), podemos reescrever a equação 2.20 da correlação total dual como:

$$DTC(S) = H(S) - \sum_{X_i \in S} H(X_i | S \setminus X_i) \tag{2.21}$$

A correlação total dual calcula a quantidade de entropia presente em S além da soma das entropias de cada variável X_i condicionada a todas as outras variáveis $S \setminus X_i$.

A correlação total dual também é chamada de excesso de entropia (OLBRICH et al., 2008) e de informação de ligação (ABDALLAH; PLUMBLEY, 2010).

Usando as equações 2.2, 2.20 e 2.21, a correlação total dual também pode ser relacionada com a correlação total pela expressão:

$$DTC(S) = \left(\sum_{X_i \in S} I(S \setminus X_i; X_i)\right) - TC(S) \tag{2.22}$$

3 HIGH-ORDER HUBS NO CÉREBRO HUMANO

Neste capítulo apresentamos um resumo do artigo de Santos et al. (2023), com destaque para a metodologia utilizada e os resultados obtidos.

Nesse trabalho, foi desenvolvido um *pipeline* de processamento de sinal multivariado para construir redes de alta ordem (*high-order networks*) a partir de séries temporais e aplicá-lo aos sinais de ressonância magnética funcional em estado de repouso (rs-fMRI) para caracterizar comunicação de alta ordem (*high-order communication*) entre regiões cerebrais, bem como a construção de *hypergrafos uniformes* e a utilização de métricas multivariadas de modo a definir pesos nestes hypergrafos.

3.1 INTRODUÇÃO

As redes (networks) oferecem uma estrutura universal para codificar informações sobre interações inerentes a um sistema complexo, que muitas vezes envolvem três ou mais subsistemas emaranhados. Entendemos que um sistema é dito complexo quando suas propriedades não são uma consequência natural de seus elementos constituintes vistos isoladamente. Sistemas complexos são sistemas que são compostos de várias partes que interagem com a habilidade de gerar novas qualidades no comportamento coletivo. As propriedades emergentes deste tipo de sistema decorrem de relações não-lineares entre as partes. Costuma-se dizer que num sistema complexo o todo é mais que a soma das partes.

Em uma abordagem padrão de redes, informações de interações de alta ordem (*high-order interactions*) são frequentemente aproximadas por meio de interações de pares. Tais aproximações, embora razoáveis, podem não ser capazes de capturar todos os aspectos importantes. Por exemplo, a comunicação em redes podem exibir cenários nos quais a comunicação entre pares de instâncias (A, B), (A, C) e (B, C) não implicam necessariamente a comunicação simultânea entre as três instâncias (A, B, C) (BAUDOT et al., 2019).

A inclusão de conexões e interações de alta ordem no estudo de redes pode ser vista como uma maneira mais realista e informativa de modelar sistemas complexos. Mais especificamente, a estrutura apresentada no artigo supracitado avança na metodologia sobre como quantificar interações de alta ordem e construir uma representação hypergráfica das interdependências em dados de série temporal.

3.2 REDES CEREBRAIS DE ALTA ORDEM

Em (SANTOS et al., 2023) é desenvolvida e aplicada uma metodologia para o cérebro humano, a fim de construir representações de alta ordem de funções de atividade cerebral a partir de séries temporais registradas em 92 regiões cerebrais, usando imagens de ressonância magnética funcional em estado de repouso (rs-fMRI). Por série temporal, entendemos como um conjunto de observações feitas sobre uma ou mais variáveis, em sequência ao longo do tempo.

Em nível funcional, as interações da rede cerebral são normalmente representadas através de medidas de similaridade entre séries temporais de atividade em duas regiões do cérebro. Existe uma infinidade de maneiras de definir essas interações entre pares, como coeficiente de correlação de Pearson, covariância e informação mútua. Algumas dessas métricas são feitas sob medida para neurociência (PEREDA; QUIROGA; BHATTACHARYA, 2005; MELLEMA; MONTILLO, 2022), outras para sistemas complexos em geral (CLIFF et al., 2022).

Para Santos et al. (2023), construir uma rede de alta ordem a partir de séries temporais requer processamento multivariado de sinais e a identificação sistemática das hyperarestas mais relevantes. A partir de uma regra de conectividade de alta ordem adequada, pode-se definir *hubs* em uma rede, o que contorna a questão combinatória da complexidade, enumerando apenas as mais importantes arestas de alta ordem sem conhecimento prévio de quais são as hyperarestas mais importantes.

A Figura 7 esboça a metodologia de processamento de sinal multivariado para construir redes de alta ordem. O método em (SANTOS et al., 2023) se baseia em heurísticas usadas no passado para definir redes como hypergrafos uniformes (COOPER; DUTLE, 2012). Em interações entre pares, definem-se arestas através de qualquer simulação confiável de pares ou métrica de similaridade entre dois nós em uma rede. As métricas de similaridade multivariadas são usadas para definir pesos de ordem superior nas hyperarestas de um hypergrafo uniforme. É usado ainda o fato de que um hypergrafo uniforme pode ser representado como uma matriz de adjacência de alta ordem, responsável pela conexão entre hyperarestas (ESTRADA; ROSS, 2018). Consequentemente, vários algoritmos já usados para matrizes de adjacência de pares como autovetor de centralidade ou modularidade poderiam ser herdados no contexto de alta ordem (ESTRADA; ROSS, 2018; SERRANO; HERNáNDEZ-SERRANO; GóMEZ, 2020), acelerando a ponte metodológica entre redes de pares e redes de alta ordem.

Em (SANTOS et al., 2023) é constatado o desempenho da metodologia, aplica em dados de

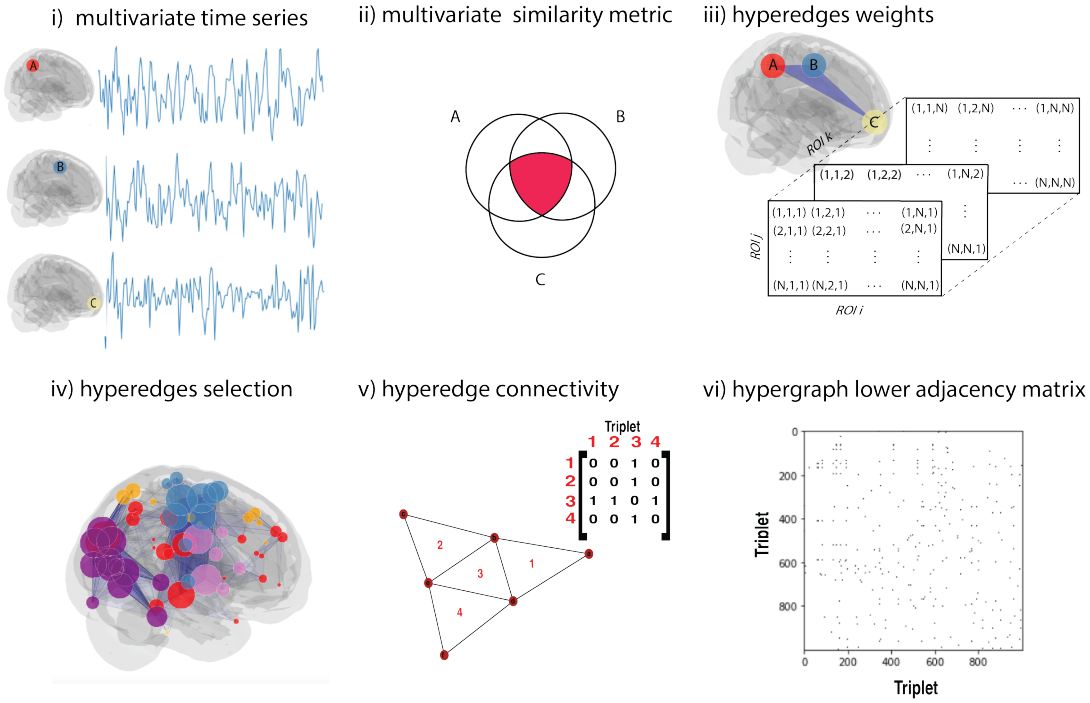


Figura 7 – Construção heurística de um hypergrafo uniforme: i) Começamos com séries temporais multivariadas como entradas, que neste trabalho são sinais BOLD de fMRI em estado de repouso. ii) Em analogia com o caso de estudo de pares, definimos pesos de conectividade de ordem superior por meio de estimativas de dependências estatísticas multivariadas. Dependências estatísticas de alta ordem podem ser quantificadas por meio, por exemplo, de informações de interação multivariada ou correlação total. iii) Uma vez os pesos das hyperarestas estão definidos, iv) podemos explorar diferentes maneiras de selecionar as hyperarestas mais importantes. v) Nós podemos também explorar regras de conectividade de alta ordem para vi) representar o hypergrafo como uma matriz de adjacência. Consequentemente, cada medida de similaridade estatística de alta ordem poderia potencialmente definir um hypergrafo uniforme a partir de séries temporais.
Fonte: Figura 2 do artigo (SANTOS et al., 2023)

neuroimagem do Projeto Humano Conectoma (HCP) (ESSEN et al., 2013), com foco no cérebro de 100 indivíduos em condições de estado de repouso. Para isso, usaram séries temporais correspondentes ao período de repouso para indicar atividade cerebral BOLD de 92 regiões do AAL Atlas (TZOURIO-MAZOYER et al., 2002), que produz até 125.580 interações de 3 pontos. A atividade cerebral é representada por meio de uma matriz de adjacência de ordem superior, e a complexidade combinatória de tal hypergrafo exige restringir o estudo às hyperarestas mais importantes. Isso leva a investigar o surgimento de *hubs* de alta ordem em redes cerebrais através de centralidade de autovetor do hypergrafo.

As análises de Santos et al. (2023) revelam que esses *hubs* de alta ordem estão longe do aleatório: cada métrica de similaridade multivariada é compatível com um sistema diferente no cérebro. Por exemplo, descobrem que alguns *hubs* fornecem tripletos de segregação e integração que são centrados nos sistemas somatossensorial e visual. Essa descoberta sugere que *hubs* de alta ordem podem ser considerados propriedades emergentes de alta ordem das redes

cerebrais funcionais. Além disso, a conectividade funcional de um *hub* de alta ordem centrado no sistema somatossensorial se correlaciona com a velocidade de marcha dos indivíduos. Essas descobertas fornecem evidências da ampla aplicação, viabilidade e relevância da metodologia, sugerindo uma rota promissora para representação hypergráfica de atividade cerebral, abrindo caminhos interessantes para pesquisas futuras em neurociência de rede de alta ordem e outros sistemas complexos.

3.3 METODOLOGIA UTILIZADA

Nesta seção apresentamos a metodologia descrita no artigo (SANTOS et al., 2023) para representar a estrutura estatística de séries temporais como hypergrafos, seguindo heurísticas análogas aos desenvolvimentos iniciais da conectividade de pares. A metodologia de Santos et al. (2023) para séries temporais multivariadas gerais é implementada em cinco etapas, conforme descrito a seguir (ver Figura 7):

1. Usamos um determinado conjunto de N séries temporais como entrada. Como primeiro passo, atribuímos um nó por série temporal, que será a base para a construção de um hypergrafo. Para séries temporais correspondentes à fMRI em estado de repouso, cada nó corresponde a uma região cerebral diferente;

2. Escolhemos uma ordem $k \in \{3, ..., N\}$, e calcule os $\binom{N}{k}$ termos de ordem k associados a uma métrica de interdependência de ordem superior entre todos os grupos possíveis de k nós diferentes. Tal como no caso da conectividade aos pares, diferentes medidas de similaridade multivariadas podem ser consideradas, e a escolha ótima é geralmente dependente do domínio e do problema. Entre as várias métricas de informação multivariada disponíveis na literatura (TIMME et al., 2014), foram utilizadas duas medidas: *Informação de Interação* (II) (veja seção 2.5.2), que quantifica dependências estatísticas de tupla única, e *Correlação Total* (TC) (confira seção 2.5.3), que quantifica as dependências estatísticas totais acumuladas em todos os subconjuntos de k-uplas.

3. Uma vez calculadas todas as interdependências, procedemos à seleção de hyperarestas. Isso é feito porque o número de interações de ordem k entre os N nós cresce com $\mathcal{O}(N^k)$, e de outra forma teríamos um hypergrafo ponderado completo. Existem diversas maneiras de

acessar a importância das arestas em uma rede, e essas estratégias também se traduzem em conectividade de alto nível. Aqui, por simplicidade, limitamos as interdependências de ordem superior e mantemos as hyperarestas mais fortes no hypergrafo.

4. Seguimos um procedimento para codificar o hypergrafo k-uniforme resultante como uma matriz de hyperadjacência. Dentre as múltiplas alternativas (ESTRADA; ROSS, 2018), a utilizada no trabalho herda a representação matricial de adjacência inferior de complexos simpliciais em hypergrafos uniformes, que também foi recentemente adaptada para desenvolver centralidades vetoriais em hypergrafos (KOVALENKO et al., 2022). Simplificando, duas hyperarestas de dimensão $k+1$ são conectadas se compartilharem uma hyperarestas de dimensão k (ESTRADA; ROSS, 2018; SERRANO; HERNáNDEZ-SERRANO; GóMEZ, 2020; SERRANO; GóMEZ, 2020). Por exemplo, dois triângulos são conectados se tiverem uma aresta em comum.

5. Como etapa final, após a especificação do processamento multivariado de sinais e da matriz de hyperadjacência, muitos recursos topológicos da ciência de redes podem ser aproveitados. Neste trabalho, foi usado uma extensão da centralidade de autovetor, introduzida em (SERRANO; GóMEZ, 2020), para investigar *hubs* de alta ordem no cérebro humano. *Hubs* de alta ordem foram explorados recentemente na ciência de redes em diversos ambientes (SERRANO; HERNáNDEZ-SERRANO; GóMEZ, 2020; KOVALENKO et al., 2021). Neste trabalho, nos basearemos no teorema de Perron-Frobenius (MACCLUER, 2006) para explorar centros de alta ordem em hypergrafos uniformes. Assim, investigamos os espectros da matriz de hyperadjacência e identificamos os hubs como tripletos com maior centralidade de autovetores. Representar o hypergrafo como uma matriz de hyperadjacência nos permite, por sua vez, calcular também outras características, como modularidade e centralidade de intermediação, que podem ser exploradas em diferentes contextos.

Observe que a complexidade combinatória de hyperarestas de ordem k em uma rede de N nós é $\mathcal{O}(N^k)$. A abordagem em (SANTOS et al., 2023) consistiu em contornar esse gargalo computacional com a utilização de análises de rede na matriz de hyperadjacência. Este procedimento é ilustrado em uma rede simples de triângulos na Figura 7(v). Esta metodologia permite inferir interações realistas de k-corpos a partir de séries temporais e representá-las como um hypergrafo, o que é crucial para o desenvolvimento teórico da ciência de redes.

Como prova de conceito, foi aplicada a metodologia descrita acima aos dados de fMRI de 100 indivíduos não aparentados (adultos jovens) do Projeto Conectoma Humano (ESSEN et al., 2013) e o grupo de relatórios calculou a média dos resultados de interdependência de ordem superior. Para ilustrar a flexibilidade da estrutura, a metodologia foi aplicada utilizando duas métricas de informação multivariada, Informação de Interacção (II) e Correlação Total (TC), que correspondem a generalizações alternativas de coeficientes de correlação multivariados. Para ambas as métricas, foram selecionados os 1000 tripletos mais fortes no hypergrafo, o que permitiu construir uma representação da matriz de adjacência do hypergrafo.

3.4 HIGH-ORDER HUBS NO CÉREBRO HUMANO

As hyperarestas definidas através da Informação de Interação (II) e Correlação Total (TC) são ilustradas na Figura 8 através da projeção dos tripletos correspondentes (iv) e da matriz de adjacência (vi) usando métodos de visualização de dados para complexos simpliciais desenvolvidos em (SANTOS et al., 2019; CENTENO et al., 2022).

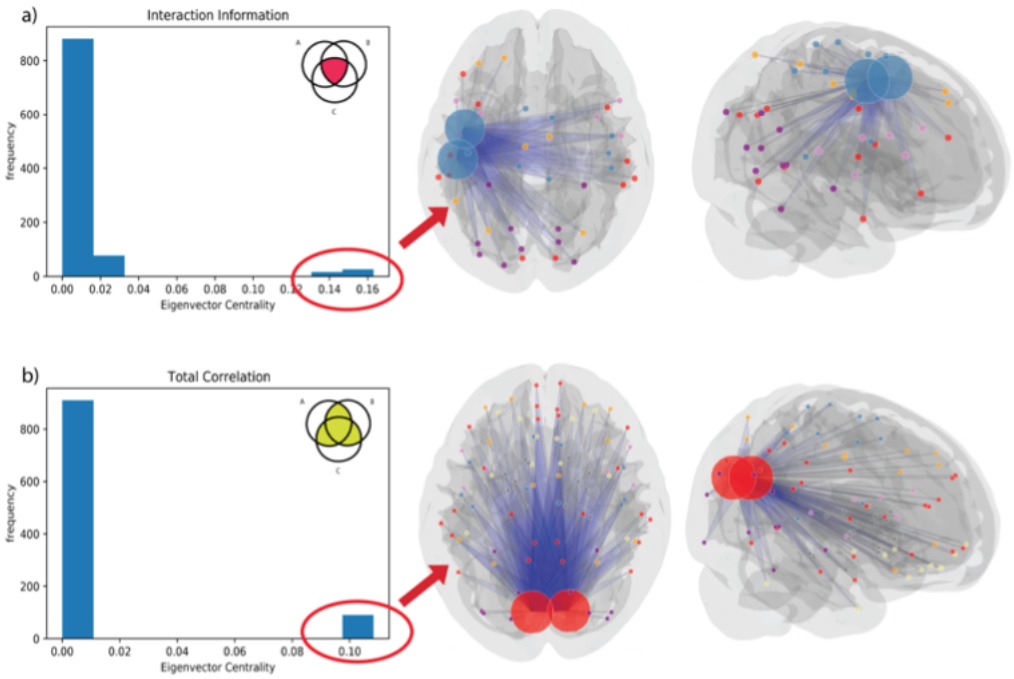


Figura 8 – Emergência de *high-order hubs* em redes cerebrais funcionais: Ao calcular o EC de alta ordem baseado nos 1000 tripletos mais fortes, tanto para informação de interação (a) quanto para correlação total (b) , apenas uma pequena fração desses tripletos têm centralidade diferente de zero. A projeção de tripletos com EC mais elevado revela o surgimento de um *hub* central de alta ordem no sistema sensório-motor para II (a) e sistema visual (b).
Fonte: Figura 3 do artigo (SANTOS et al., 2023)

Santos et al. (2023) descobrem que os tripletos mais fortes são consistentes com a organização da rede do cérebro em sub-redes. A Figura 8(a) mostra a distribuição da Centralidade do Autovetor (EC) dos 1000 tripletos com o maior II. A partir deste histograma, fica claro que apenas uma pequena fração tem uma EC longe de zero (apenas 42 tripletos estão destacados no círculo vermelho). Dada a complexidade combinatória das interações de alta ordem, a interpretação de todas as interações de alta ordem nas redes cerebrais se tornará mais evidente quando forem apenas projetados os tripletos com EC mais alta no cérebro (Figura 8(a)).

3.4.1 Informação da Interação e o sistema motor primário no cérebro

Para II, foi observado que todos os tripletos centrais com o EC mais alto compartilham dois nós e um link em comum, cobrindo todo o cérebro, produzindo um padrão identificado como *high-order hubs*.

Os dois nós compartilhados entre os 42 tripletos com alto EC são os giros pré e pós-centrais. Dado o papel dessas duas áreas no funcionamento do cérebro, esses *hubs* emergentes de alta ordem foram associados ao sistema sensório-motor primário. Uma característica interessante sobre os giros pré e pós-centrais é que eles são "topograficamente organizados", ou seja, áreas adjacentes da superfície motora (ou receptiva) do corpo humano são mapeadas em áreas vizinhas do giro pré e pós-central. O giro pré (e pós) central corresponde ao córtex motor primário (e somatossensorial), e o tamanho relativo da representação cortical reflete a densidade dos nervos motores (e somatossensoriais) em cada parte do corpo (SANTOS et al., 2023).

A faixa somatossensorial (giro pós-central) contém um mapa somatotópico invertido do lado oposto do corpo que quase reflete o da faixa motora (giro pré-central). No entanto, muito recentemente, este mapa foi revisitado, e regiões integrativas da faixa motora também foram relatadas, relacionando regiões interefetoras com regiões pré-frontais, insulares e subcorticais da rede cíngulo-opercular (CON), críticas para a ação executiva e controle fisiológico, excitação e processamento de erros e dor (GORDON et al., 2022). Dado que o banco de dados do HCP consiste principalmente de indivíduos destros, o fato de os centros de ordem superior estarem no hemisfério esquerdo do cérebro é consistente com a bem conhecida assimetria na conectividade funcional das áreas sensório-motoras devido à lateralidade (AMUNTS et al., 1996; TEJAVIBULYA et al., 2022). Além disso, devido à necessidade de integração sensório-motora para executar um movimento, a atividade nessas duas áreas é altamente acoplada. Por exemplo, quando uma

pessoa move a mão direita, o aumento do fluxo sanguíneo é registrado no giro pré-central esquerdo e a ativação simultânea é observada na área correspondente do giro pós-central (HAVEL et al., 2006; KOCAK et al., 2009).

Santos et al. (2023) justificam os resultados obtidos com a compreensão atual do giro pré e pós-central, uma interpretação plausível é que, dado que os giros pré e pós-central estão altamente acoplados no cérebro, qualquer interação relevante entre essas duas áreas e outra terceira área no cérebro seria pelo menos uma interação de três pontos entre sensório-motor e outras regiões corticais.

3.4.2 Correlação total e o sistema visual no cérebro

Para a segunda métrica de informação multivariada, Correlação Total (TC), foi utilizado um procedimento semelhante ao II, ou seja, foi calculado a TC de todos os tripletos possíveis no atlas AAL e escolhemos os 1000 tripletos com o maior TC para construir um hypergrafo uniforme. Novamente, apenas uma pequena fração dos tripletos tem uma EC não próxima de zero. Quando projetados esses tripletos com EC elevada no cérebro, descobrimos que todos os tripletos envolvem duas áreas específicas do cérebro. Neste caso, os *hubs* de ordem superior estão associados à área visual de Broadman 17, nomeadamente o cuneus esquerdo e direito, como ilustrado na Figura 8(b). Cada cuneus (dos hemisférios esquerdo e direito) recebe informações visuais via tálamo pulvinar da retina superior contralateral representando o campo visual inferior, bem como via tálamo geniculado. O cuneus também é conhecido por seu papel extra-retiniano e é modulado, por exemplo, por atenção, memória de trabalho e expectativa de recompensa (VANNI et al., 2001; DOÑAMAYOR; SCHOENFELD; MÜNTE, 2012; BLUHM et al., 2011). Dada a divisão contralateral dos mapas visuais do cuneus e seu papel no cérebro, e semelhante ao caso sensório-motor, sob uma perspectiva de alta ordem, a integração ou modulação da informação visual está provavelmente promovendo pelo menos interações de três pontos definido pelo TC. É importante notar que os resultados obtidos estão alinhados com o trabalho recente de Luppi et al. (2022), onde os autores relataram que os núcleos redundantes são mais prevalentes nos sistemas visual e motor.

Deste modo, as hyperarestas de EC mais elevado na representação do hypergrafo do cérebro são compatíveis com o conhecimento atual em neuroanatomia funcional. Pelo menos numa perspectiva qualitativa, o padrão emergente de centros redundantes de alta ordem definidos

via II e TC são compatíveis com a segregação local e os princípios de integração global.

4 HIGH-ORDER HUBS NA BOLSA DE VALORES $\mathbf{S\&P500}$

Neste capítulo apresentamos uma aplicação da metodologia desenvolvida no artigo "*Emergence of high-order functional hubs in the human brain*"(SANTOS et al., 2023), com o objetivo de estudar as inter-relações entre as empresas participantes do $S\&P500$ (abreviação de *Standard & Poor's 500*), índice composto por quinhentos ativos cotados nas bolsas de NYSE ou NASDAQ e qualificados devido ao seu tamanho de mercado, sua liquidez e sua representação de grupo industrial.

Para isso, a fim de comparação, foram realizadas seleções de faixas temporais em momentos específicos: antes, durante e um pouco depois de uma fase de grande instabilidade dessa bolsa de valores (período crítico da epidemia da COVID-19), entre abril de 2019 e outubro de 2021.

Os dados estudados foram obtidos a partir da plataforma *Yahoo Finance*, na qual foi possível ter acesso de forma gratuita às planilhas de 55 empresas selecionadas pelo fato da co-existência nos períodos de tempo selecionados. Os dados obtidos da $S\&P500$ foram selecionados na aba "*Adj Close*", que se refere ao valor de fechamento diário de cada empresa.

Dessa forma, foi possível estruturar uma tabela com o valor diário de fechamento de 55 empresas selecionadas da $S\&P500$.

4.1 OBTENÇÃO DOS DADOS

O *Yahoo Finance* é uma plataforma online que oferece informações financeiras abrangentes, notícias, análises e ferramentas para investidores, abrangendo uma variedade de ativos como ações, títulos, commodities, moedas e índices. Os usuários podem acessar cotações em tempo real, gráficos interativos, notícias de mercado e análises especializadas, além de recursos para acompanhar e gerenciar portfólios de investimentos.

O $S\&P500$, ou *Standard & Poor's* 500, é um índice ponderado pelo valor de mercado que monitora o desempenho de 500 das maiores empresas negociadas nos EUA. Considerado um indicador representativo do mercado de ações dos EUA, o $S\&P500$ é bastante usado como referência por inúmeros investidores e profissionais financeiros. Empresas são selecionadas com base em critérios como capitalização de mercado, liquidez e representação setorial, conforme determinado pelo comitê da $S\&P$ *Dow Jones Indices*.

A coluna "*Adj Close*" (Ajuste de Fechamento) em tabelas financeiras, como no *Yahoo Finance*,

representa o preço de fechamento de uma ação, ajustado para eventos corporativos, como dividendos e divisões de ações. Este ajuste é importantíssimo para uma avaliação precisa do desempenho do investimento ao longo do tempo, evitando distorções no histórico de preços. O cálculo leva em consideração esses eventos, oferecendo uma visão mais realista do retorno total do investimento, especialmente relevante para análises a longo prazo. Ao consultar essa coluna, os investidores podem obter uma representação mais precisa do desempenho histórico da ação, considerando não apenas a valorização do preço, mas também os efeitos de eventos corporativos ao longo do tempo.

Dessa forma, foi possível estruturar uma tabela com valores diários de fechamento de algumas empresas pré-selecionadas com a utilização da biblioteca *yfinance* no *Python* (confira Figura 9). Inicialmente, foram selecionadas as 55 empresas com base em critérios como capitalização de mercado, liquidez e representação setorial, bem como a necessidade de coexistência nos intervalos de tempo escolhidos (pré, pós e durante a fase de grande instabilidade). A partir da coluna "Adj Close", produzimos as séries temporais destas 55 empresas alocadas dentro do período selecionado. Nesta tabela (ver Figura 9), as linhas são indexadas pelas datas selecionadas sequencialmente (dia a dia) e as colunas, pelos *tickets* (códigos) das empresas (confira Apêndice B).

```
In [4]: data_total.head()
```

Out[4]:

	Adj Close										...	Volume		
	AAPL	ABBV	ABT	ACN	ADBE	AMGN	AMT	AMZN	AVGO	AXP	...	NKE	NVDA	PEP
Date														
2019-04-01	46.085369	64.740814	73.590393	164.389893	272.170013	165.859161	172.889908	90.709503	262.596863	104.427200	...	6737400	48382400	5025700
2019-04-02	46.755295	66.576126	73.553436	163.504166	271.350006	166.429993	175.233887	90.698997	260.243805	103.772758	...	4433800	44092000	3242600
2019-04-03	47.075802	66.584137	73.442574	165.201050	271.500000	166.853806	174.488464	91.035004	261.231384	103.361427	...	4080900	78350400	4534400
2019-04-04	47.157730	66.367737	72.629623	165.173065	267.890015	166.343491	173.374847	90.943001	259.342194	103.314514	...	3660700	45737600	3022500
2019-04-05	47.473423	66.880646	72.980682	166.096085	267.450012	169.007339	175.808655	91.863998	261.300049	104.102592	...	7367400	48174400	3971500

5 rows × 330 columns

Figura 9 – Tabela dos dados coletados com a utilização da biblioteca *yfinance* no *Python*

4.2 TRATAMENTO DE DADOS

Em seguida, os dados foram processados e estimados em distribuições de probabilidade conjuntas. Para isso, diante do tipo de dado que está sendo analisado, esse processo precisará

passar por uma conversão para estados discretos.

Assim, calculamos as taxas de variação percentuais diária referente ao dia anterior (confira Figura 10), ou seja, para cada empresa foi calculado o quanto (em porcentagem) a empresa cresceu/decresceu de um dia para o outro.

```
In [17]: DF.head()
```

Out[17]:

Date	AAPL	ABBV	ABT	ACN	ADBE	AMGN	AMT	AMZN	AVGO	AXP	...	NKE	NVDA	PEP	PFE
2019-04-02	1.453662	2.834861	-0.050220	-0.538796	-0.301285	0.344166	1.355764	-0.011582	-0.896072	-0.626697	...	-1.009030	0.395017	-0.262279	0.1634(
2019-04-03	0.685499	0.012033	-0.150724	1.037823	0.055277	0.254649	-0.425387	0.370463	0.379482	-0.396377	...	0.118497	3.071012	-0.098601	-0.1631
2019-04-04	0.174035	-0.325003	-1.106919	-0.016940	-1.329645	-0.305846	-0.638218	-0.101063	-0.723187	-0.045387	...	0.958940	-0.190870	-0.074049	-0.2334(
2019-04-05	0.669440	0.772829	0.483355	0.558820	-0.164247	1.601415	1.403783	1.012720	0.754931	0.762795	...	0.140709	1.428894	0.123504	0.5849(
2019-04-08	1.573596	0.635168	-0.607628	0.426622	0.508501	-0.271224	0.235007	0.684705	0.065754	-0.234314	...	-0.784533	0.439922	0.312409	0.3489:

5 rows × 55 columns

Figura 10 – Tabela dos dados percentuais

Dessa forma, minimizando eventuais discrepâncias, assim como, foi possível comparar empresas que têm patrimônios gigantescos com empresas de menor porte.

A fim de calcular a entropia de Shannon, bem como a *Mutual Information*, a *Interaction Information* e a *Total Correlation*, medidas de informação dependentes diretamente da entropia de Shannon, precisamos discretizar nossos dados.

Discretizar dados é o processo de converter variáveis contínuas em categorias ou intervalos, muito utilizado em análises estatísticas quando deseja-se simplificar a complexidade dos dados ou quando determinadas análises ou algoritmos requerem variáveis discretas. Esse processo de discretização deve ser realizado com cuidado, pois introduz uma perda de informação.

Dessa forma, para cada empresa, os valores percentuais foram distribuídos em n sub-intervalo (no nosso caso, discretizamos com $n = 20$ sub-intervalos) e, naturalmente, associá-los a números inteiros de 0 a $n - 1$ (confira Figura 11).

Após a etapa de discretização dos dados, foi determinado o tamanho da janela de tempo. Esse tamanho refere-se ao intervalo de observações agrupadas para análise. A definição do tamanho de janelas de tempo é uma consideração delicada ao lidar com a análise de uma série temporal, depende do contexto da análise e dos padrões nos dados.

No nosso caso, escolhemos 20 dias contínuos.

Agora, passemos ao cálculo da entropia de Shannon de uma empresa X para cada uma das 55 selecionadas:

```
In [18]: DF_bin.head()
```

Out[18]:

Date	AAPL	ABBV	ABT	ACN	ADBE	AMGN	AMT	AMZN	AVGO	AXP	...	NKE	NVDA	PEP	PFE	PG	TMO	UNH	V	VZ	WMT
2019-04-02	15	19	8	5	7	12	16	9	5	6	...	4	10	6	11	9	6	6	11	2	3
2019-04-03	12	9	8	15	8	11	6	11	10	7	...	10	17	7	8	8	12	13	12	15	11
2019-04-04	10	7	3	8	3	7	5	8	5	9	...	14	8	8	8	6	5	13	5	12	16
2019-04-05	12	14	12	12	7	17	16	14	12	13	...	10	13	10	13	10	11	13	9	11	15
2019-04-08	15	13	5	11	11	8	11	13	9	8	...	4	10	12	12	17	12	9	9	10	13

5 rows × 55 columns

Figura 11 – Tabela dos dados discretizados para 20 sub-intervalos

4.3 CÁLCULO DA ENTROPIA DE SHANNON

Para a empresa X, começamos pela janela 1 e procedemos para o cálculo da entropia dessa janela. Para cada valor presente, obtém-se a frequência de ocorrência, ou seja, quantas vezes cada valor aparece na janela, e calcula-se a probabilidade de ocorrência, dividindo a frequência de ocorrência desse valor pelo número total de valores presentes nesta janela.

Por fim, aplica-se a fórmula da entropia de Shannon (confira 2.4) na janela. Tal valor é salvo e prosseguimos o algoritmo para janela de tempo seguinte até chegarmos à última janela. Os dados das entropias de Shannon são salvos por janela de tempo e na forma vetorial, cujas colunas (entradas) são indexadas pelos *tickets* (códigos) das empresas.

Criamos *rolling windows* (janelas deslizantes) de tamanho 20, referentes a entropia de Shannon para a empresa X. Com a utilização das janelas deslizantes, foi possível construir um gráfico e acompanhar o comportamento da entropia de Shannon da empresa X durante todo o período observado (confira Figura 12).

Uma análise preliminar da entropia de Shannon das 55 empresas produz resultados interessantes, mas não muito conclusivos. Claramente, podemos ver na Figura 12 que entre março e abril de 2020 ocorreu uma grande pertubação na bolsa de valores $S\&P500$, justamente no momento mais crítico da pandemia da COVID-19. Isso fica mais evidente na Figura 13, que apresenta as janelas deslizantes médias (*medium rolling windows*) das entropias de Shannon das 55 empresas observadas.

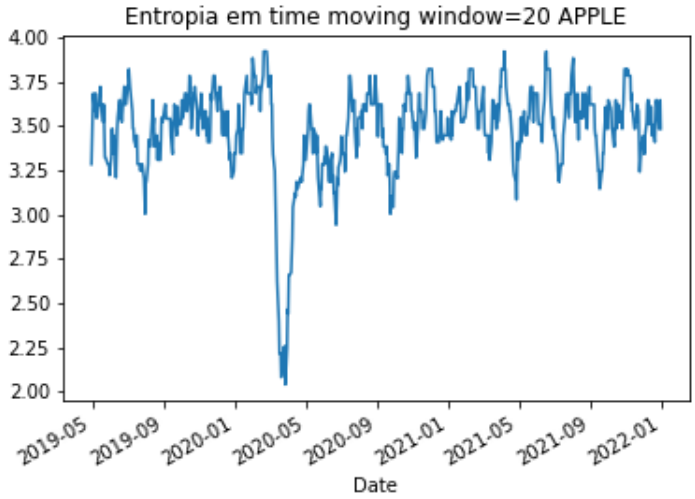


Figura 12 – Entropia da Apple em uma janela de tempo móvel de tamanho 20, note uma queda acentuada entre Janeiro e Maio de 2020, onde aconteceram *Lockdowns* da COVID-19.

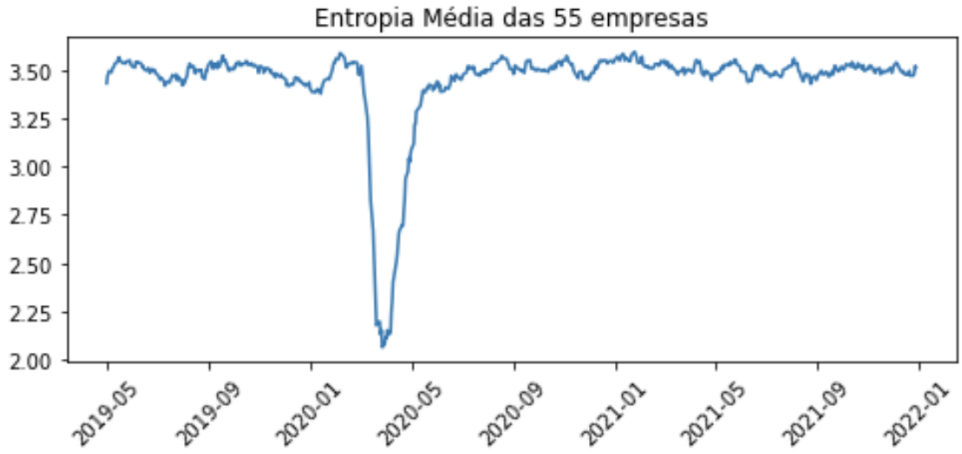


Figura 13 – Entropia média em um time movie window 20

4.4 CÁLCULO DA *MUTUAL INFORMATION*

Quando consideramos os dados como várias variáveis independentes, a entropia de Shannon pode ser aplicada a cada variável separadamente para avaliar a incerteza associada a cada uma delas. Quando estamos pensando em variáveis estatisticamente dependentes, a Entropia de Shannon também pode ser usada para avaliar a dependência entre as variáveis, com a utilização de uma outra medida de informação.

Passamos agora para a *mutual information*, onde podemos estudar relações entre duas variáveis, ou seja, o quanto uma delas interfere na outra.

O processo do estudo da *mutual information* é feito utilizando os dados das empresas já tratados, ou seja, convertidos em taxas de variação e discretizados, e refazemos o processo de

janela de tempo móvel.

Como vimos em 2.5.1, ao examinar a relação entre duas variáveis, a *mutual information* mede a quantidade de informação fornecida sobre uma das variáveis ao conhecer o valor da outra. Assim, para cada janela de tempo analisada, foi criada uma matriz indexada nas linhas e colunas pelos *tickets* das empresas, e cada entrada da linha i e coluna j, representa a *mutual information* da i-ésima empresa com a j-ésima. Note que a simetria da fórmula 2.2 produzirá como resultado uma matriz simétrica. Esse processo é feito até chegarmos na última janela de tempo. Os dados da *mutual information* são salvos por janela de tempo e na forma matricial (ou tabela) cujas linhas e colunas são indexadas pelos tickets das empresas.

Criamos mapas de calor em cada janela de tempo. Para efeito de comparação, a construção desses mapas de calor passa por um processo de normalização. Pegamos o maior e o menor valores de todas as entradas das matrizes obtidas e associamos os valores intermediários a uma faixa de valores no padrão de cores RGB. Podemos construir agora, para cada janela de tempo, um mapa de calor que representa a intensidade da quantidade de informação que uma das variáveis fornece ao conhecer a outra. As figuras 14, 15 e 16 ilustram os mapas de calor da *mutual information* médias nos períodos pré, durante e pós instabilidade provocada pela COVID-19. Note que, a partir da paleta de cores, podemos observar que no período pré-crítico existia uma dependência moderada entre as empresas; durante a fase de instabilidade da pandemia, essa dependência média cai para valores críticos; e, no pós-instabilidade, sobe para valores mais elevados em relação ao período pré-crítico.

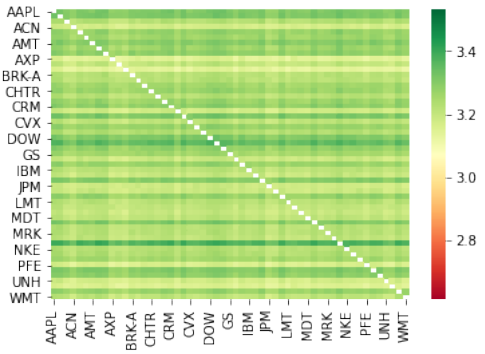


Figura 14 – Mapa de calor com a *Mutual Information* média

No lugar de um gráfico, produzimos um vídeo (ou GIF), formado por um conjunto de imagens sequenciais, começando na primeira janela de tempo até a última, e foi possível acompanhar as inter-dependências entre duas empresas quaisquer com o passar do tempo.

Entretanto, permanece a dúvida: será que essas relações existem em ordem superior? Isto é,

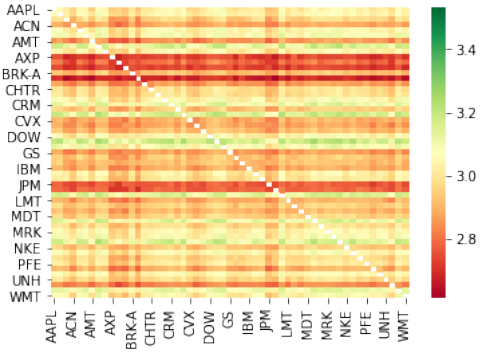


Figura 15 – Mapa de calor com a *Mutual Information* média

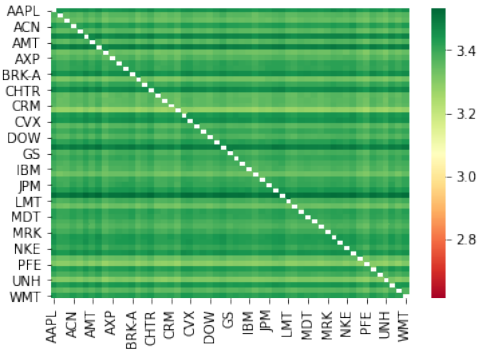


Figura 16 – Mapa de calor com a *Mutual Information* média

será que são consequência de inter-relações de 3 ou mais empresas?

4.5 CÁLCULO DA *INTERACTION INFORMATION* E *TOTAL CORRELATION*

Passemos agora ao estudo das inter-relações entre 3 empresas ao mesmo tempo ao longo das janelas de tempo. Este estudo pode ser generalizado para k empresas, com $3 \leq k < n$. Enfim, neste trabalho nos restringimos ao estudo de inter-relações entre 3 empresas. Aqui tratamos agrupamentos formados por 3 empresas como tripletos. E como obter tripletos e suas inter-relações?

Foi apresentado anteriormente duas medidas de informação: a *Interaction Information* e a *Total Correlation*, que foram utilizadas para medir as inter-relações sobre o conjunto desses tripletos. O processo de cálculo foi parecido com o realizado no caso das relações par a par (feita com *mutual information*).

Então, partindo dos dados discretizados, para cada janela de tempo criamos uma lista (tabela), onde nela é computada para cada tripleto de empresas a medida de informação requerida. É importante notar que não é necessário computar repetições, visto que $II(X, Y, Z) = II(X, Z, Y) = II(Y, X, Z)$, pela fórmula 2.9, bem como pela fórmula 2.19 da *Total Correlation*.

Uma vez calculadas todas as interdependências, procedemos à seleção de hyperarestas mais fortes. Isso é feito porque o número de interações tem ordem $\binom{55}{3}$ entre os mais de 26 mil tripletos, e, além do que, teríamos um hypergrafo ponderado completo. Então, por simplicidade, limitamos as interdependências de ordem superior e escolhemos as hyperarestas mais fortes no hypergrafo, com base nas informações adquiridas através da *Interaction Information* ou da *Total Correlation*.

Com base nos resultados da Figura 13, obtidos a partir da entropia de Shannon média, escolhemos três períodos temporais para estudo: pré, durante e pós-instabilidade (mesmos períodos utilizados nos cálculos da *Mutual Information*). Tomamos as *Interaction Information* e *Total Correlation* médias dos tripletos nesses períodos e escolhemos os 100 maiores tripletos em cada um deles.

Neste trabalho, foi usado uma extensão da centralidade do autovetor (ver seção 2.3), introduzida em (SERRANO; HERNáNDEZ-SERRANO; GóMEZ, 2020), para investigar o surgimento de *hubs* de alta ordem no estudo da bolsa de valores $S\&P500$.

Para isso, representamos o hypergrafo por uma matriz de hyperadjacência, na qual cada tripleto é considerado vizinho a outro se, e somente se, possuem um *link* (aresta) em comum, e investigamos o espectro dessa matriz, identificando os *hubs* como tripletos com maior cen-

tralidade de autovetor.

Com essa matriz geramos um novo grafo, no qual os vértices representam os tripletos e as arestas correspondem aos pares de tripletos que são vizinhos.

As figuras 17 e 18 abaixo foram geradas com a utilização da biblioteca *NetworkX* no *Python* e representam os grafos construídos com tripletos em cada período: pré, durante e pós-instabilidade.

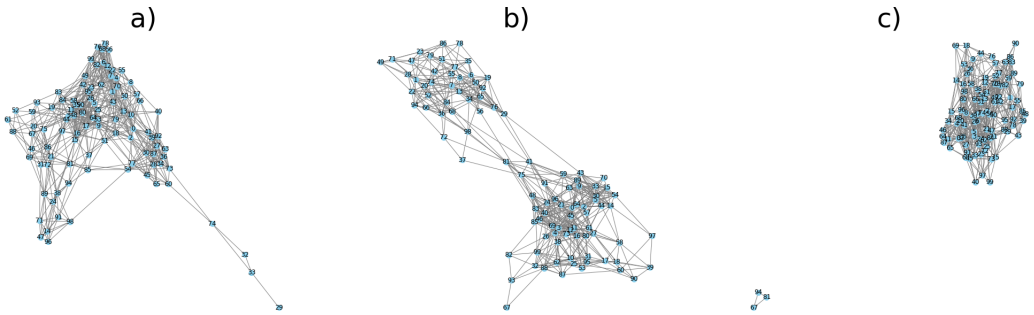


Figura 17 – Grafos produzidos com a utilização da biblioteca *NetworkX* do *Python* dos 100 tripletos mais relevantes com a *Interaction Information*: a) pré-instabilidade; b) durante a instabilidade; e c) pós-instabilidade

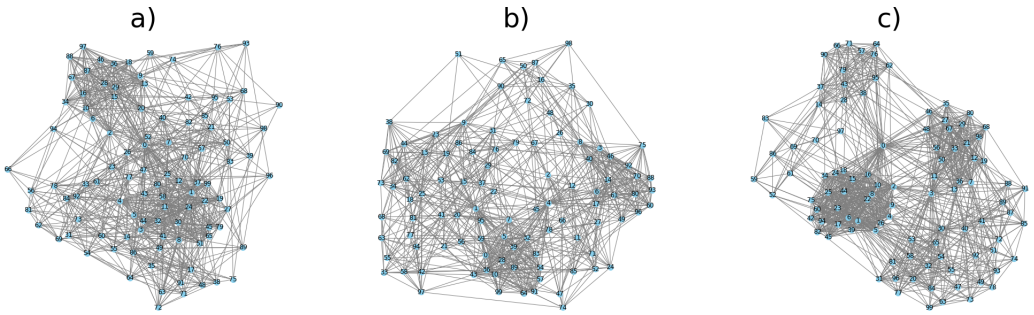


Figura 18 – Grafos produzidos com a utilização da biblioteca *NetworkX* do *Python* dos 100 tripletos mais relevantes com a *Total Correlation*: a) pré-instabilidade; b) durante a instabilidade; e c) pós-instabilidade

Feito isso, calculamos o EC (*eigenvector centrality*) do grafo, ou seja, uma medida que computa a importância de cada vértice do grafo gerado. A ideia por trás do autovalor de centralidade é que a importância de um vértice é diretamente proporcional às importâncias dos vértices aos quais está conectado (ver seção 2.3). Em outras palavras, estamos computando a importância de cada tripleto para o sistema.

4.6 RESULTADOS OBTIDOS

Apresentamos os resultados obtidos com a *Interaction Information (II)* e *Total Correlation (TC)*. A interpretação da II está relacionada à quantidade de informação sobre uma variável X que será ganha ou perdida ao levar em consideração uma variável Z em adição a outra variável Y. Se $II(X;Y;Z)$ for próximo de zero, então X e Y são independentes dado Z, indicando que Z explica completamente a relação entre X e Y. Quanto mais distante de zero for $II(X;Y;Z)$, haverá uma interação significativa entre X, Y e Z.

Para *Total Correlation*, sua interpretação está relacionada ao grau de incerteza associado com as três variáveis simultaneamente. Se há dependência entre as variáveis, a entropia conjunta será menor, indicando uma redução na incerteza quando uma dessas variáveis for conhecida. Uma alta TC indica que as variáveis estão fortemente relacionadas e que o conhecimento de uma variável reduzirá substancialmente a incerteza sobre as outras duas.

Em cada período selecionado, foram calculadas as médias da *Interaction Information* e *Total Correlation* para todas as janelas compreendidas neste período.

Para efeito de resultados, trabalharemos com os 100 tripletos mais sinérgicos. Ou seja, em cada período é aplicado o método z-*score* à medida de informação multivariada calculada e são selecionados os valores mais positivos. O método z-*score*, ou pontuação padrão, consiste numa maneira de redistribuir os dados em termos de sua relação com a média e o desvio padrão deles. Obter um z-*score* é simplesmente mapear os dados em uma distribuição cuja média é definida como 0 e cujo desvio padrão é definido como 1.

Calculamos os EC's (*eigenvector centrality*) do grafo e construímos o hypergrafo dos 20 tripletos de maior EC.

4.6.1 Período Pré-Instabilidade

Como explicado anteriormente, foi selecionado o período entre $02/04/2019$ e $21/04/2020$, e foram calculadas as médias da *Interaction Information* e da *Total Correlation* para todas as janelas compreendidas neste período.

Foram selecionados os 100 tripletos sinergicamente mais representativos, calculados os EC's (*eigenvector centrality*) do grafo e construímos o hypergrafo dos 20 tripletos de maior EC.

4.6.1.1 *Resultados II na pré-instabilidade*

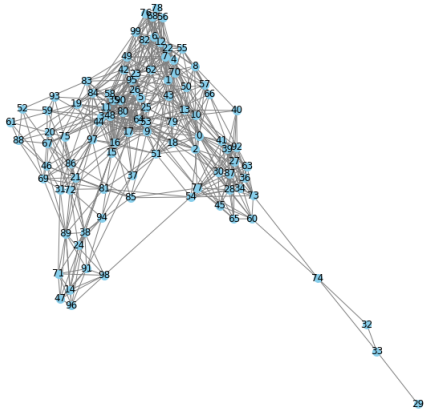


Figura 19 – Grafo produzido com a utilização da biblioteca *NetworkX* do *Python* dos 100 tripletos mais relevantes com a *Interaction Information* média no período pré-instabilidade

Calculamos os EC's (*eigenvector centrality*) do grafo, obtendo como resultado a tabela 1.

Tabela 1 – Tripletos x *Eigenvalues centrality* para *Interaction Information* média da pré-instabilidade

Tripleto	***Eigenvalue centrality***
23 – [ADBE, NFLX, PG]	0.24828918281476972
03 – [NFLX, PG, TMO]	0.235193558587225
09 – [AAPL, NFLX, PG]	0.2329155141591195
26 – [AMZN, NFLX, PG]	0.2093964519343948
05 – [CRM, NFLX, PG]	0.2058040274794558
25 – [AVGO, NFLX, PG]	0.20515768567781395
17 – [ABBV, NFLX, PG]	0.20123614419412175
16 – [JNJ, NFLX, PG]	0.19538691293427332
53 – [GOOGL, NFLX, PG]	0.19169787870455418
58 – [NFLX, PEP, PG]	0.1886709244474578
15 – [AMT, NFLX, PG]	0.18625880192124103
44 – [AMGN, NFLX, PG]	0.18400641996432868
11 – [ABT, NFLX, PG]	0.18294075720861863
48 – [LLY, NFLX, PG]	0.18197645665688025
35 – [CVS, NFLX, PG]	0.18107609873343916
90 – [COST, NFLX, PG]	0.17656789758084002
...	

Fonte: o autor

Criamos um histograma com 30 subdivisões para os autovalores de centralidade e representado na figura 20.

Notamos que, no período pré-instabilidade, a maioria dos tripletos apresenta um EC abaixo de 0.15. Dois grupos se destacam: um deles com EC entre 0.17 e 0.21, e um outro grupo

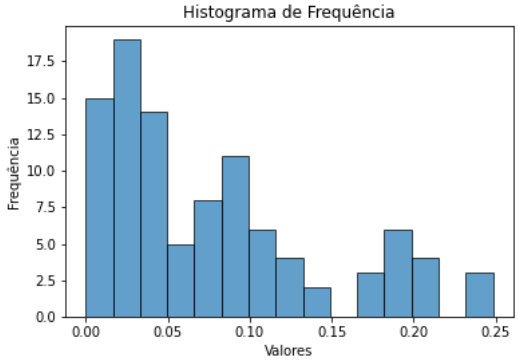


Figura 20 – Histograma dos EC's para *Interaction Information* média da pré-instabilidade

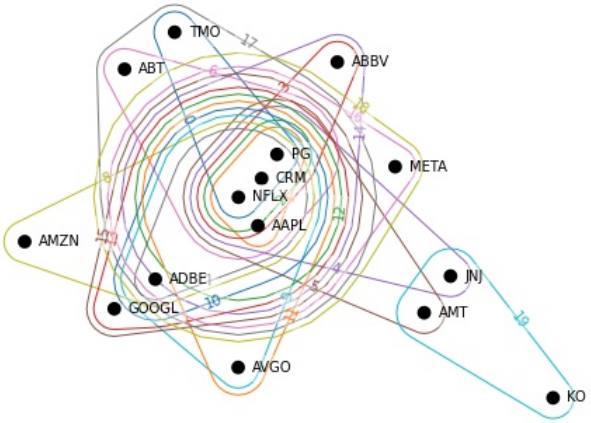


Figura 21 – Hubs no Hypergrafo com os 20 tripletos com maior centralidade de autovetor para sinergia na *Interaction Information* média da pré-instabilidade

menor, com apenas 3 tripletos, possuem um EC acima de 0.23 e formado pelos tripletos: [ADBE, NFLX, PG], [NFLX, PG, TMO] e [AAPL, NFLX, PG]. Note ainda que as empresas NFLX e PG formam um link (aresta) comum a este grupo, bem como ao grupo intermediário de centralidades (conf. tabela 1 e figura 21).

Para os 100 tripletos mais representativos por redundância, foram refeitas todas as construções acima descritas e obtido o hypergrafo descrito na figura 22.

Agora, a maioria dos tripletos apresenta um EC abaixo de 0.10. Um grande grupo se destaca com EC entre 0.19 e 0.20, e 2 tripletos possuem EC de 0.22 e 0.24 formados por: [AXP, BA, BAC] e [BA, BAC, PFE]. Note ainda que as empresas BA e BAC formam um link (aresta) comum a este grupo, bem como ao grupo intermediário de autovalores.

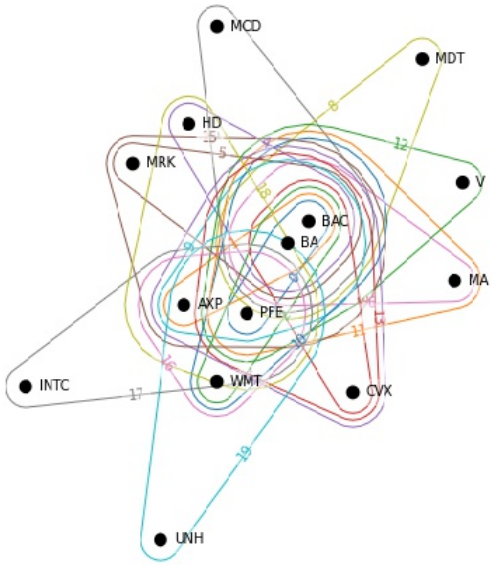


Figura 22 – Hubs no Hypergrafo com os 20 tripletos com maior centralidade de autovetor para redundância na *Interaction Information* média da pré-instabilidade

4.6.1.2 Resultados TC na pré-instabilidade

Como explicado anteriormente, foi calculada a média da *Total Correlation* para todas as janelas compreendidas neste período. Foram selecionados os 100 tripletos mais representativos, produzida a matriz de incidência desses tripletos e construído o grafo associado a essa matriz (confira Figura 23).

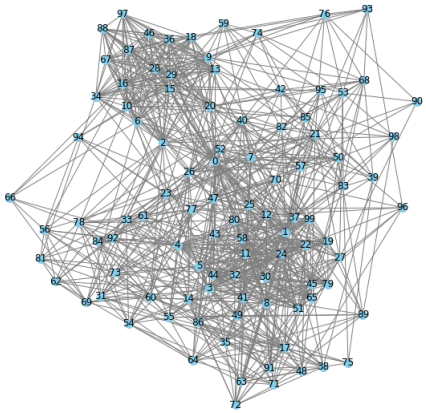


Figura 23 – Grafo produzido com a utilização da biblioteca *NetworkX* do *Python* dos 100 tripletos mais relevantes com a *Total Correlation* média da pré-instabilidade da pandemia da COVID-19

Calculamos os EC's (*eigenvector centrality*) do grafo, obtendo como resultado a tabela 2

Tabela 2 – Tripletos x *Eigenvalues centrality* para *Total Correlation* média da pré-instabilidade

Tripleto	***Eigenvalue centrality***
00 – [AXP, BAC, PFE]	0.27936235298499545
08 – [BAC, PFE, WMT]	0.18828235037436156
03 – [BAC, PFE, V]	0.1879454391077968
04 – [BAC, PFE, UNH]	0.18113477311376705
12 – [BA, BAC, PFE]	0.17228382877905551
11 – [BAC, HD, PFE]	0.16809866701843199
05 – [BAC, MCD, PFE]	0.16760016187396456
43 – [BAC, CSCO, PFE]	0.16722177785119965
30 – [BAC, KO, PFE]	0.16330236128246617
44 – [BAC, JPM, PFE]	0.15976553363009624
24 – [BAC, CVX, PFE]	0.15916516849096127
19 – [BAC, INTC, PFE]	0.15874726697594244
80 – [BAC, LMT, PFE]	0.1586784109533024
32 – [BAC, MDT, PFE]	0.15801403928582253
22 – [ACN, BAC, PFE]	0.15765364519289365
58 – [BAC, NVDA, PFE]	0.15410316754061495
99 – [BAC, MRK, PFE]	0.15410316754061495
51 – [BAC, MSFT, PFE]	0.1490973028132884
65 – [BAC, MA, PFE]	0.1490973028132884
79 – [BAC, NKE, PFE]	0.1490973028132884
...	

Fonte: o autor

Criamos um histograma com 30 subdivisões para os autovalores de centralidade e representado na figura 24.

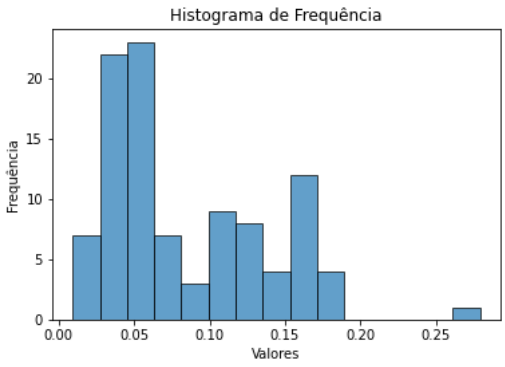


Figura 24 – Histograma dos EC's para *Total Correlation* média da pré-instabilidade da pandemia

Observamos que a maioria dos tripletos para TC apresentam um EC abaixo de 0.1. Um único tripleto se destaca, possui um EC acima de 0.27 e formado pelas empresas AXP, BAC e PFE. Note ainda que as empresas NFLX e PG formam um link (aresta) comum a este grupo, bem como ao grupo intermediário de autovalores (conf. tabela 2).

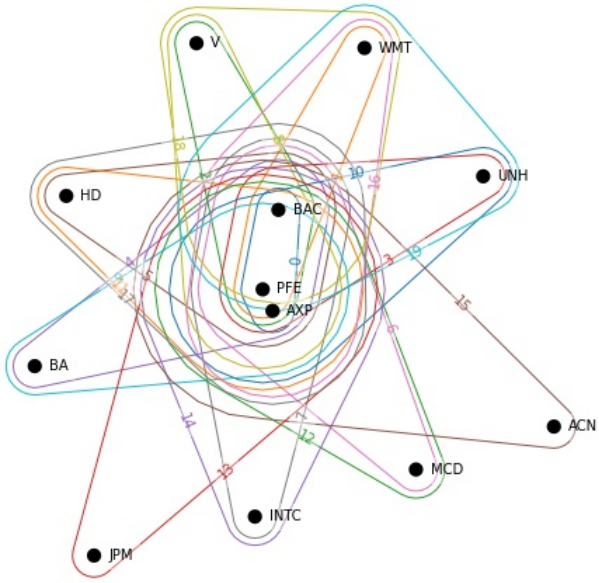


Figura 25 – Hubs no Hypergrafo com os 20 tripletos com maior centralidade de autovetor para *Total Correlation* média na pré-instabilidade

4.6.2 Período de instabilidade da pandemia

Da mesma forma que no período pré-instabilidade, foi selecionado o período entre $04/02/2020$ e $07/08/2020$, e calculadas as médias da *Interaction Information* e da *Total Correlation* para todas as janelas compreendidas neste período.

4.6.2.1 Resultados II durante a instabilidade

Novamente, foram selecionados os 100 tripletos mais representativos, produzida a matriz de incidência desses tripletos e construído o grafo associado a essa matriz (confira figura 26) e obtidos os resultados apresentados na tabela 3 e na figura 27.
Observamos que, durante a fase crítica da pandemia da COVID-19, mais de 40% dos tripletos apresenta um EC muito próximo de zero. Um único tripleto, formado pelas empresas ADBE, GOOGL e MSFT, se destaca isolado com com EC de 0.307 (conf. tabela 3).

4.6.2.2 Resultados TC durante a instabilidade

Da mesma forma, foi calculada a média da *Total Correlation* para todas as janelas compreendidas neste período, obtendo os seguintes resultados apresentados nas figuras 30, 31, 32,

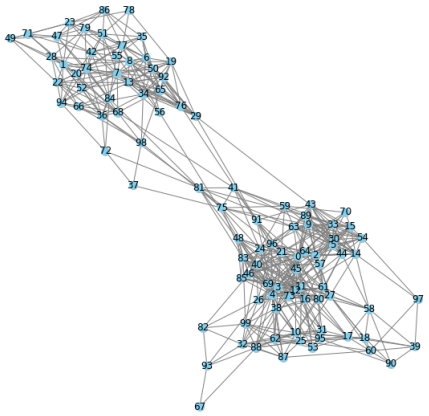


Figura 26 – Grafo produzido com a utilização da biblioteca *NetworkX* do *Python* dos 100 tripletos mais relevantes com a *Interaction Information* média durante a instabilidade da pandemia

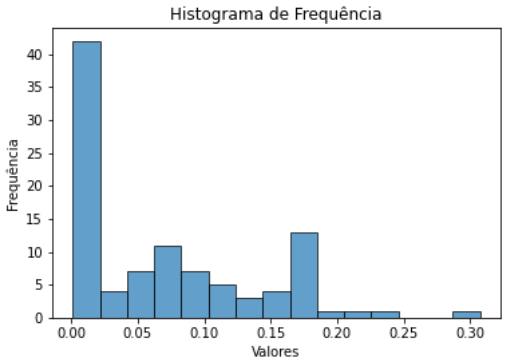


Figura 27 – Histograma dos EC's para *Interaction Information* média na instabilidade da pandemia

?? e tabela 4.

4.6.3 Período após a instabilidade da pandemia

4.6.3.1 Resultados II após a instabilidade

Foi selecionado o período entre $04/01/2021$ e $17/12/2021$, e calculada a média da *Interaction Information* para todas as janelas neste período, obtendo os resultados das figuras 33, 34, 35 e 36, bem como da tabela 6.

Tabela 3 – Tripletos x *Eigenvalues centrality* para *Interaction Information* média na instabilidade da pandemia

Tripleto	*Eigenvalue centrality*
00 – [ADBE GOOGL MSFT]	0.30745091082965
04 – [CRM GOOGL MSFT]	0.23864745271723173
02 – [ADBE CRM MSFT]	0.22437178897602755
44 – [ADBE META MSFT]	0.1864806736592571
12 – [GOOGL META MSFT]	0.18478029880481323
09 – [ADBE INTC MSFT]	0.1843460966067093
63 – [ADBE MSFT TMO]	0.1828379303446331
33 – [ADBE COST MSFT]	0.18137475492143273
14 – [ADBE AMZN MSFT]	0.18133253335517757
11 – [AMZN GOOGL MSFT]	0.17335432053936342
05 – [AAPL ADBE MSFT]	0.17283759895451334
30 – [ADBE AVGO MSFT]	0.17236059527816386
89 – [ADBE AMGN MSFT]	0.16997825353297658
83 – [GOOGL MSFT TMO]	0.16892731721702955
24 – [GOOGL INTC MSFT]	0.1657937543281542
03 – [AAPL GOOGL MSFT]	0.16572994287524928
15 – [ADBE ENPH MSFT]	0.16569366599453472
21 – [COST GOOGL MSFT]	0.16282241264287764
43 – [ADBE LLY MSFT]	0.16216660059856852
69 – [AVGO GOOGL MSFT]	0.16200098215823824
54 – [ADBE MSFT NFLX]	0.15833372184255395
...	

Fonte: o autor

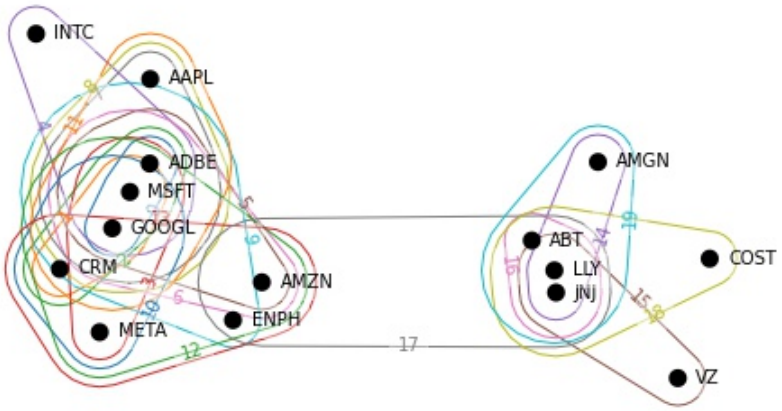


Figura 28 – Hubs no Hypergrafo com os 20 tripletos com maior centralidade de autovetor para sinergia na *Interaction Information* média durante a instabilidade da pandemia da COVID-19

4.6.3.2 Resultados TC após a instabilidade

Foi selecionado o período entre $04/01/2021$ e $17/12/2021$, e calculada a média da *Total Correlation* para todas as janelas compreendidas neste período, obtendo os resultados apresentados nas figuras 37, 38, 39, 40 e tabela 6.

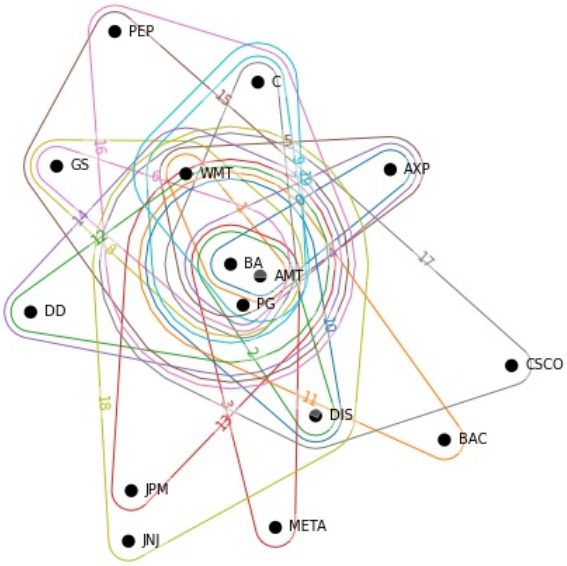


Figura 29 – Hubs no Hypergrafo com os 20 tripletos com maior centralidade de autovetor para redundância na *Interaction Information* média durante a instabilidade da pandemia da COVID-19

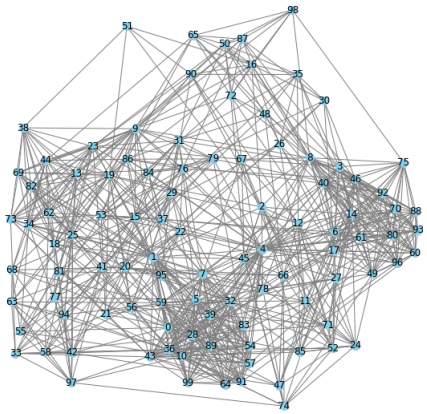


Figura 30 – Grafo produzido com a utilização da biblioteca *NetworkX* do *Python* dos 100 tripletos mais relevantes com a *Total Correlation* média durante a instabilidade da pandemia da COVID-19

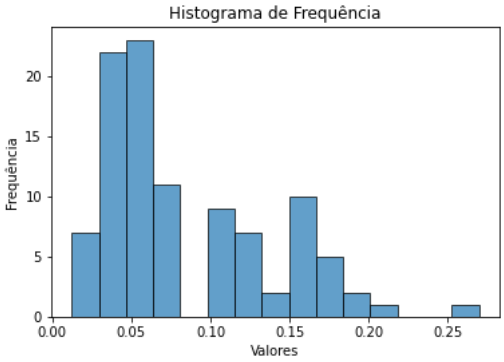


Figura 31 – Histograma dos EC's para *Total Correlation* média na instabilidade da pandemia

Tabela 4 – Tripletos x *Eigenvalues centrality* para *Total Correlation* média na instabilidade

Tripleto	*Eigenvalue centrality*
04 – [AXP, BA, C]	0.2695692578698861
01 – [BA, C, KO]	0.2088361470340198
32 – [BA, BAC, C]	0.1932026279787982
36 – [BA, C, JPM]	0.1867322775318642
07 – [BA, C, PG]	0.17722245344614262
00 – [AMT, BA, C]	0.17578669500232955
10 – [BA, C, V]	0.17421731921758518
05 – [BA, C, LMT]	0.17144504387827278
28 – [ABT, BA, C]	0.16890497506758187
59 – [BA, C, CAT]	0.16646362163703216
43 – [BA, C, WMT]	0.1629117125452749
89 – [BA, C, IBM]	0.15963294866866046
39 – [BA, C, CSCO]	0.15963294866866043
83 – [BA, C, HD]	0.15939599791821016
54 – [BA, C, PEP]	0.15842090085905877
57 – [BA, C, JNJ]	0.1569894244221844
64 – [BA, C, MMM]	0.1569894244221844
91 – [ACN, BA, C]	0.1569894244221844
99 – [BA, C, PFE]	0.1539095887659493
14 – [AXP, BA, BAC]	0.1388920470545299
...	

Fonte: o autor

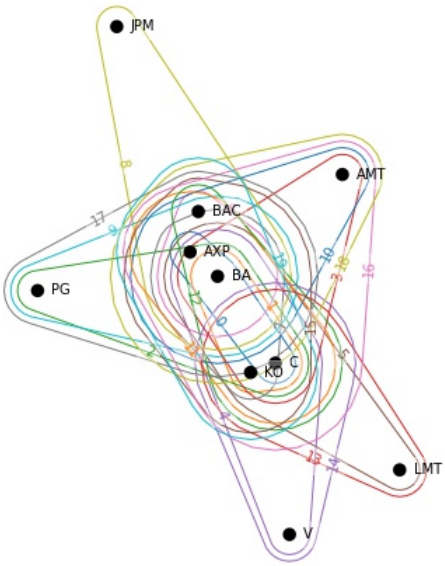


Figura 32 – Hubs no Hypergrafo com os 20 tripletos com maior centralidade de autovetor para *Total Correlation* média na instabilidade da pandemia da COVID-19

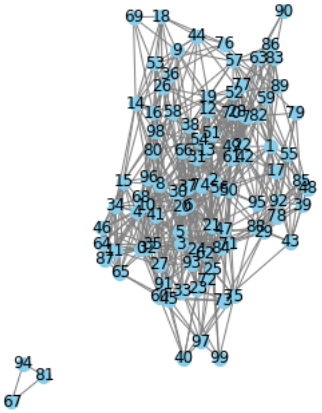


Figura 33 – Grafo produzido com a utilização da biblioteca *NetworkX* do *Python* dos 100 tripletos mais relevantes com a *Interaction Information* média pouco depois a pandemia da COVID-19

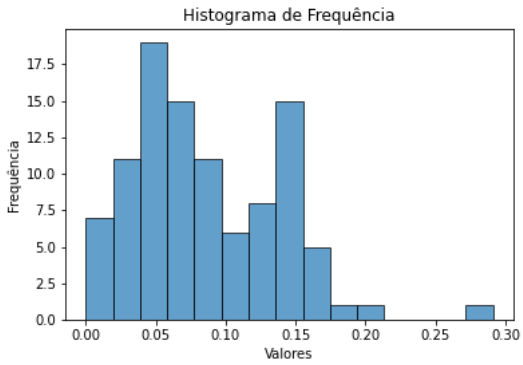


Figura 34 – Histograma dos EC's para *Interaction Information* média pouco depois a pandemia da COVID-19

Tabela 5 – Tripletos x *Eigenvalues centrality* para *Interaction Information* média pós-pandemia

Tripleto	***Eigenvalue centrality***
02 – [AMZN GOOGL LLY]	0.29088312618306456
05 – [AAPL AMZN GOOGL]	0.20549599896274814
20 – [AAPL GOOGL LLY]	0.17471149490035298
03 – [AMZN GOOGL MSFT]	0.17021140168330906
37 – [AMZN GOOGL NVDA]	0.16760992525227822
06 – [ACN AMZN GOOGL]	0.1620655308490949
71 – [AMZN CHTR GOOGL]	0.16081548473416657
54 – [AAPL AMZN LLY]	0.15874298430350361
25 – [AMZN GOOGL KO]	0.15283547795675903
47 – [AMGN AMZN GOOGL]	0.15131192544711933
07 – [AMGN AMZN LLY]	0.15069466296268777
42 – [AMZN CHTR LLY]	0.14971183339238325
19 – [AMZN LLY NVDA]	0.1492814552345015
38 – [GOOGL LLY NVDA]	0.14240532760682797
23 – [AMZN BRK-A GOOGL]	0.14173792144767966
50 – [CHTR GOOGL LLY]	0.1402844926793168
93 – [AMZN GOOGL NFLX]	0.13994011265631767
30 – [GOOGL LLY MSFT]	0.13893064825804854
00 – [AAPL GOOGL MSFT]	0.13890646972814388
62 – [AMZN CVX GOOGL]	0.13809723836426568
49 – [AMGN GOOGL LLY]	0.13722789179999256
...	

Fonte: o autor

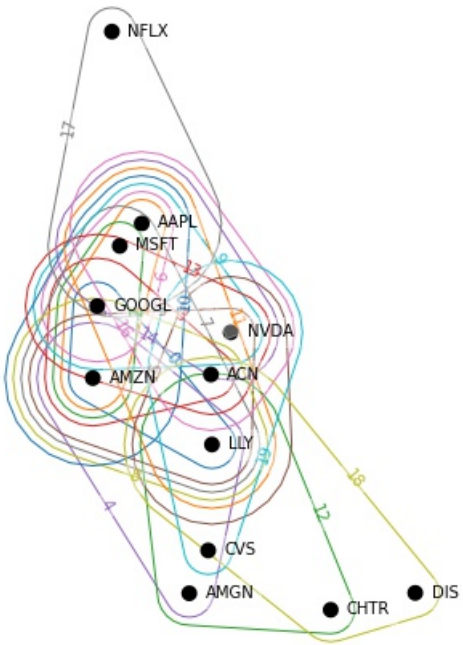


Figura 35 – Hubs no Hypergrafo com os 20 tripletos com maior centralidade de autovetor para sinergia na *Interaction Information* média na pós-pandemia da COVID-19

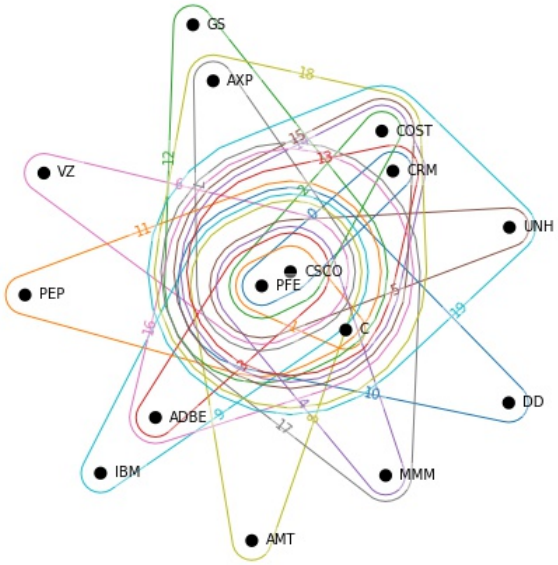


Figura 36 – Hubs no Hypergrafo com os 20 tripletos com maior centralidade de autovetor para redundância na *Interaction Information* média na pós-pandemia da COVID-19

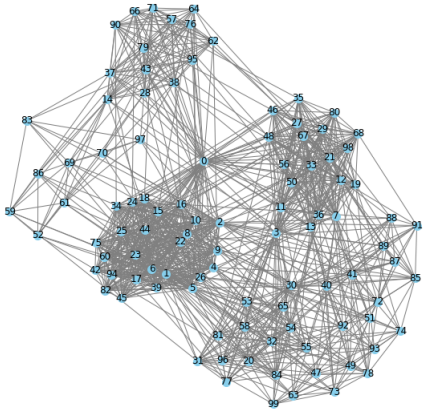


Figura 37 – Grafo produzido com a utilização da biblioteca *NetworkX* do *Python* dos 100 tripletos mais relevantes com a *Total Correlation* média pouco depois a pandemia da COVID-19

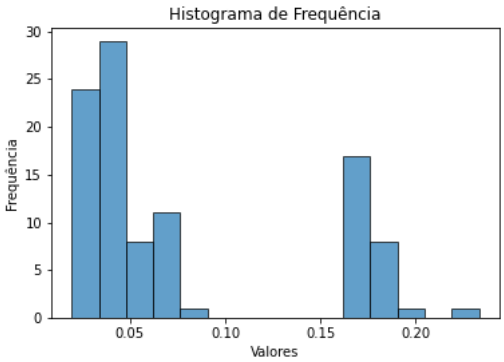


Figura 38 – Histograma dos EC's para *Total Correlation* média pouco depois a pandemia da COVID-19

Tabela 6 – Tripletos x *Eigenvalues centrality* para *Total Correlation* média pós-pandemia

Tripleto	***Eigenvalue centrality***
00 – [CSCO, PFE, UNH]	0.23327170102711306
01 – [C, CSCO, PFE]	0.19609170853544186
02 – [CSCO, INTC, PFE]	0.1871877755290601
04 – [CSCO, MA, PFE]	0.18290221255267652
05 – [AVGO, CSCO, PFE]	0.1828992732334619
06 – [CSCO, PEP, PFE]	0.1789384125539855
26 – [CMCSA, CSCO, PFE]	0.17875744242345887
09 – [CRM, CSCO, PFE]	0.17760870810897733
08 – [ABT, CSCO, PFE]	0.17720707855870885
10 – [COST, CSCO, PFE]	0.17639394746633885
16 – [CSCO, PFE, VZ]	0.1752754440779152
15 – [ADBE, CSCO, PFE]	0.17413531555054748
18 – [AXP, CSCO, PFE]	0.17413531555054748
17 – [AMT, CSCO, PFE]	0.17288022888128943
23 – [CSCO, MMM, PFE]	0.17288022888128943
24 – [CSCO, DD, PFE]	0.17278733090660298
34 – [CSCO, IBM, PFE]	0.17278733090660298
22 – [ABBV, CSCO, PFE]	0.17257844720885718
25 – [CSCO, DOW, PFE]	0.17153899494069763
39 – [CSCO, GS, PFE]	0.17153899494069763
44 – [CSCO, PFE, TMO]	0.17153899494069763
42 – [CSCO, LMT, PFE]	0.16939221867784382
45 – [CSCO, HON, PFE]	0.16939221867784382
60 – [CSCO, META, PFE]	0.16939221867784382
75 – [CSCO, MRK, PFE]	0.16939221867784382
...	

Fonte: o autor

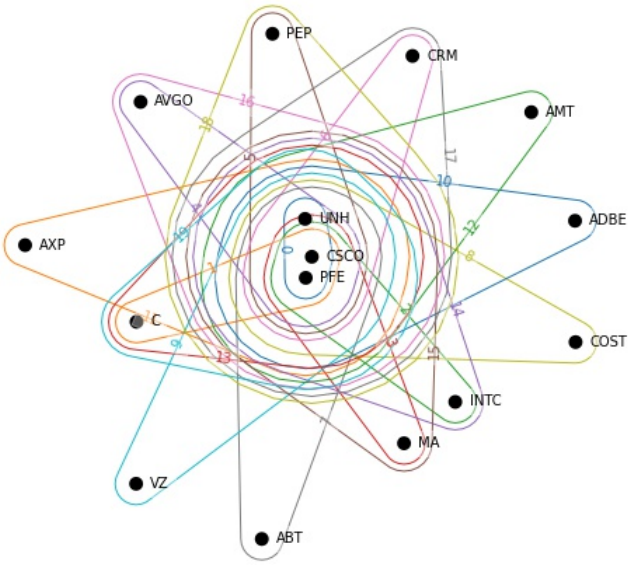


Figura 39 – Hubs no Hypergrafo com os 20 tripletos com maior centralidade de autovetor para sinergia na *Total Correlation* média na pós-pandemia da COVID-19

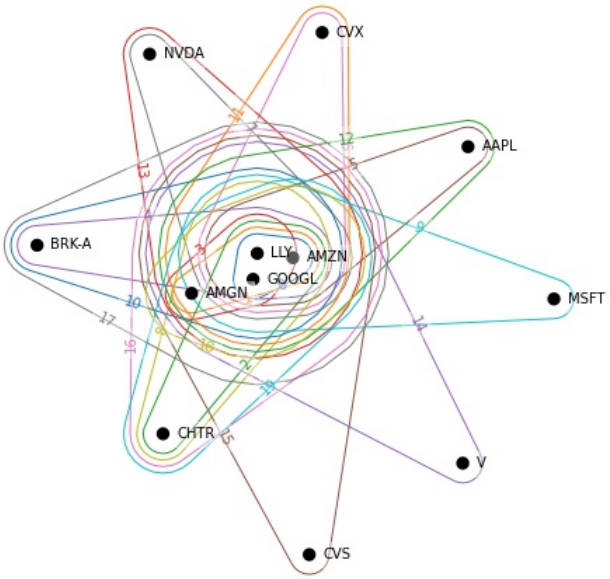


Figura 40 – Hubs no Hypergrafo com os 20 tripletos com maior centralidade de autovetor para redundância na *Total Correlation* média na pós-pandemia da COVID-19

5 CONCLUSÕES E PERSPECTIVAS

Apresentamos uma aplicação da metodologia desenvolvida no artigo "Emergence of high-order functional hubs in the human brain"de Santos et al. (2023), com o objetivo de estudar as inter-relações entre as empresas participantes do $S\&P500$.

Por questão de simplicidade e de limitação computacional disponível, decidimos efetuar os cálculos para os 100 tripletos mais relevantes do mais dos 26 mil obtidos a partir das 55 empresas selecionadas para o estudo.

Como prova de conceito, aplicamos a metodologia descrita acima aos dados da bolsa de valores, utilizando duas métricas de informação multivariada, a *Interaction Information* e a *Total Correlation*, que correspondem a generalizações alternativas da *Mutual Information*.

Escolhendo a discretização dos dados com 20 sub-intervalos e o número de 100 tripletos mais relevantes por limitação computacional, observamos o surgimento de *hubs* de alta ordem com a obtenção dos autovetores de centralidade mais relevantes em cada situação.

Encontramos resultados semelhantes aos obtidos por Santos et al. (2023), bem como pudemos observar que a utilização de métricas de informação multivariadas se mostra uma ferramenta capaz de medir a dependência ou independência entre variáveis que acreditamos serem estatisticamente dependentes.

Identificamos que a utilização da metodologia, bem como sua generalização para hypergrafos k-uniformes com $k \geq 3$, pode ser aplicada em diversas áreas, desde sistemas complexos como o cérebro humano até o mercado financeiro da bolsa de valores.

Por fim, visto que conseguimos avançar nos trabalhos mesmo com algumas limitações computacionais e de obtenção de dados públicos gratuitos da bolsa de valores, uma perspectiva futura seria investigar a aplicabilidade da metodologia para um número maior de empresas correlacionadas, bem como outros grupos empresariais a serem selecionados e com acesso a um banco de dados mais completo.

REFERÊNCIAS

ABDALLAH, S. A.; PLUMBLEY, M. D. *A measure of statistical complexity based on predictive information*. 2010. Disponível em: <https://arxiv.org/abs/1012.1890>.

AMUNTS, K.; SCHLAUG, G.; SCHLEICHER, A.; STEINMETZ, H.; DABRINGHAUS, A.; ROLAND, P. E.; ZILLES, K. Asymmetry in the human motor cortex and handedness. *Neuroimage*, Elsevier, v. 4, n. 3, p. 216–222, 1996.

ANASTASSIOU, D. Molecular systems biology. v. 3, p. 83, 2007.

BAR-YAM, Y. General features of complex systems. *Encyclopedia of Life Support Systems (EOLSS) UNESCO Publishers*, UK: Oxford, 2022.

BARABÁSI, A.-L. Network science. *[S.l.]: Cambridge university press*, 2016.

BARABÁSI, A.-L.; FRANGOS, J. Linked: The new science of networks science of networks. *[S.l.]: Basic Books*, 2014.

BAUDOT, P.; TAPIA, M.; BENNEQUIN, D.; GOAILLARD, J.-M. Topological information data analysis. *Entropy*, v. 21, n. 9, 2019. ISSN 1099-4300. Disponível em: <https://www.mdpi.com/1099-4300/21/9/869>.

BELL, A. International workshop on independent component analysis and blind signal separation. p. 921, 2003.

BERGE, C. *Hypergraphs: Combinatorics of Finite Sets*. [S.l.]: North-Holland, 1989.

BETTENCOURT, L.; GINTAUTAS, V.; HAM, M. Identification of functional information subgraphs in complex networks. *Physical review letters 100 (23)*, v. 4, p. 238701, 2008.

BLUHM, R. L.; CLARK, C. R.; MCFARLANE, A. C.; MOORES, K. A.; SHAW, M. E.; LANIUS, R. A. Default network connectivity during a working memory task. *Human brain mapping*, Wiley Online Library, v. 32, n. 7, p. 1029–1035, 2011.

BONACICH, P. *Power and Centrality: A Family of Measures*. [S.l.]: Social Networks, 1987. v. 29.

BONDY, J. A.; MURTY, U. S. R. *Graph Theory with Applications*. Fifth printing. NORfH-HOLLAND: New York, 1982. Disponível em: <https://www.zib.de/groetschel/teaching/WS1314/BondyMurtyGTWA.pdf>.

BRENNER, N.; STRONG, S.; KOBERLE, R.; BIALEK, W.; STEVENINCK, R. de Ruyter van. Neural computation. v. 12, p. 1531, 2000.

CENTENO, E. G. Z.; MORENI, G.; VRIEND, C.; DOUW, L.; SANTOS, F. A. N. A hands-on tutorial on network and topological neuroscience. *Brain Structure and Function*, v. 227, 2022.

CHECHIK, G.; GLOBERSON, A.; TISHBY, N.; ANDERSON, M.; YOUNG, E.; NELKEN, I. In t.g. dietterich, s. becker, z. ghahramani (eds.), neaural information processing systems. MIT Press, v. 1, n. 4, p. 173, 2001.

CLAUSET, A.; SHALIZI, C. R.; NEWMAN, M. E. J. Power-law distributions in empirical data. *SIAM Review*, Society for Industrial and Applied Mathematics, v. 51, n. 4, p. 661–703, 2009. ISSN 00361445, 10957200. Disponível em: <http://www.jstor.org/stable/25662336>.

CLIFF, O. M.; LIZIER, J. T.; TSUCHIYA, N.; FULCHER, B. D. Unifying pairwise interactions in complex dynamics. 1 2022. Disponível em: <https://arxiv.org/abs/2201.11941v1>.

COOPER, J.; DUTLE, A. Spectra of uniform hypergraphs. *Linear Algebra and its applications*, v. 436, p. 3268, 2012.

COVER, T. M.; THOMAS, J. A. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. USA: Wiley-Interscience, 2006. ISBN 0471241954.

DISTEL, R. *Graph Theory*. Eletronic Edition. Spriger-Verlag: New York, 2005. Disponível em: <https://contacts.ucalgary.ca/info/math/files/info/unitis/courses/PMAT60351/F2006/LEC1/GraphTheoryIII.pdf>.

DOÑAMAYOR, N.; SCHOENFELD, M. A.; MÜNTE, T. F. Magneto-and electroencephalographic manifestations of reward anticipation and delivery. *Neuroimage*, Elsevier, v. 62, n. 1, p. 17–29, 2012.

ERDŐS, P.; RÉNYI, A. On random graphs i. *Publ. Math. Debrecen*, v. 6, p. 290–297, 1959.

ERDŐS, P.; RÉNYI, A. On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, v. 5, n. 1, p. 17–60, 1960. ISSN 0007-4985.

ERDŐS, P.; RÉNYI, A. On the strength of connectedness of a random graph. *Acta Mathematica Academiae Scientiarum Hungarica*, Springer, v. 12, n. 1-2, p. 261–267, 1964. ISSN 0001-5954.

ESSEN, D. C. V.; SMITH, S. M.; BARCH, D. M.; BEHRENS, T. E.; YACOUB, E.; UGURBIL, K.; CONSORTIUM, W.-M. H. et al. The wu-minn human connectome project: an overview. *Neuroimage*, Elsevier, v. 80, p. 62–79, 2013.

ESTRADA, E.; ROSS, G. J. Centralities in simplicial complexes. applications to protein interaction networks. *Journal of Theoretical Biology*, v. 438, p. 46, 2018.

FREITAS, L. Q. *Medidas de Centralidade em Grafos*. [S.l.]: Dissertação de Mestrado do Programa de Pós-graduação em Engenharia de Produção. Rio de Janeiro:UFRJ/COPPE, 2010.

GAT, I.; TISHBY, N. In m.s. kearns, s.a. solla, d.a. cohn (eds.). neural information processing systems. MIT Press, v. 11, p. 111, 1999.

GORDON, E. M.; CHAUVIN, R. J.; VAN, A. N.; RAJESH, A.; NIELSEN, A.; NEWBOLD, D. J.; LYNCH, C. J.; SEIDER, N. A.; KRIMMEL, S. R.; SCHEIDTER, K. M.; AL. et. A mind-body interface alternates with effector-specific regions in motor cortex. *bioRxiv*, 2022.

GRIFFITH, V.; KOCH, C. *Quantifying synergistic mutual information*. 2014.

HAN, T. Information and control. v. 29, p. 337, 1975.

HAN, T. Information and control. v. 36, p. 133, 1978.

HAVEL, P.; BRAUN, B.; RAU, S.; TONN, J.-C.; FESL, G.; BRÜCKMANN, H.; ILMBERGER, J. Reproducibility of activation in four motor paradigms. *Journal of neurology*, Springer, v. 253, n. 4, p. 471–476, 2006.

HORN., A.; JOHNSON, C. *Matrix Analysis*. [S.l.]: Cambridge Uni- versity Press, Cambridge, 1985.

JAKULIN, A.; BRATKO, I. *Quantifying and Visualizing Attribute Interactions*. 2004.

KOCAK, M.; ULMER, J. L.; UGUREL, M. S.; GAGGL, W.; PROST, R. W. Motor homunculus: passive mapping in healthy volunteers by using functional mr imaging—initial results. *Radiology*, Radiological Society of North America, v. 251, n. 2, p. 485–492, 2009.

KOVALENKO, K.; ROMANCE, M.; ALEJA, D.; VASILYEVA, E.; CRIADO, R.; RAIGORODSKII, A. M.; FLORES, J.; PERC, K. A.-B. M.; BOCCALETTI, S. Vector centrality in networks with higher-order interactions. 2021.

KOVALENKO, K.; ROMANCE, M.; VASILYEVA, E.; ALEJA, D.; CRIADO, R.; MUSATOV, D.; RAIGORODSKII, A.; FLORES, J.; SAMOYLENKO, I.; ALFARO-BITTNER, K.; AL. et. Vector centrality in hypergraphs. *Chaos, Solitons & Fractals*, v. 162, p. 112397, 2022.

LUPPI, A. I.; MEDIANO, P. A.; ROSAS, F. E.; HOLLAND, N.; FRYER, T. D.; O'BRIEN, J. T.; ROWE, J. B.; MENON, D. K.; BOR, D.; STAMATAKIS, E. A. A synergistic core for human brain evolution and cognition. *Nature Neuroscience*, Nature Publishing Group, v. 25, n. 6, p. 771–782, 2022.

MACCLUER, C. R. The many proofs and applications of perron's theorem. v. 42, p. 487, 2006. Disponível em: <http://dx.doi.org/10.1137/S0036144599359449>.

MARKRAM, H.; MULLER, E.; RAMASWAMY, S.; REIMANN, M.; ABDELLAH, M.; SANCHEZ, C.; AILAMAKI, A.; ALONSO-NANCLARES, L.; ANTILLE, N.; ARSEVER, S.; KAHOU, G.; BERGER, T.; BILGILI, A.; BUNCIC, N.; CHALIMOURDA, A.; CHINDEMI, G.; COURCOL, J.-D.; DELALONDRE, F.; DELATTRE, V.; DRUCKMANN, S.; DUMUSC, R.; DYNES, J.; EILEMANN, S.; GAL, E.; GEVAERT, M.; GHOBRIL, J.-P.; GIDON, A.; GRAHAM, J.; GUPTA, A.; HAENEL, V.; HAY, E.; HEINIS, T.; HERNANDO, J.; HINES, M.; KANARI, L.; KELLER, D.; KENYON, J.; KHAZEN, G.; KIM, Y.; KING, J.; KISVARDAY, Z.; KUMBHAR, P.; LASSERRE, S.; LE Bé, J.-V.; MAGALHãES, B.; MERCHáN-PéREZ, A.; MEYSTRE, J.; MORRICE, B.; MULLER, J.; MUñOZ-CéSPEDES, A.; MURALIDHAR, S.; MUTHURASA, K.; NACHBAUR, D.; NEWTON, T.; NOLTE, M.; OVCHARENKO, A.; PALACIOS, J.; PASTOR, L.; PERIN, R.; RANJAN, R.; RIACHI, I.; RODRíGUEZ, J.-R.; RIQUELME, J.; RöSSERT, C.; SFYRAKIS, K.; SHI, Y.; SHILLCOCK, J.; SILBERBERG, G.; SILVA, R.; TAUHEED, F.; TELEFONT, M.; TOLEDO-RODRIGUEZ, M.; TRäNKLER, T.; VAN GEIT, W.; DíAZ, J.; WALKER, R.; WANG, Y.; ZANINETTA, S.; DEFELIPE, J.; HILL, S.; SEGEV, I.; SCHüRMANN, F. Reconstruction and simulation of neocortical microcircuitry. *Cell*, v. 163, n. 2, p. 456–492, 2015. ISSN 0092-8674. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0092867415011915>.

MATSUDA, H. Physical review e. v. 62, p. 3096, 2000.

MCGILL, W. J. Multivariate information transmission. *EIRE Trans Info Theory*, v. 4, p. 93–111, 1954.

MELLEMA, C. J.; MONTILLO, A. Reproducible measures of correlative and causal brain connectivity. 1 2022. Disponível em: <https://arxiv.org/abs/2201.13378v1>.

NEWMAN, M. E. J. The structure and function of complex networks. *SIAM review, SIAM*, v. 46, n. 2, p. 167–256, 2003. ISSN 0036-1445.

NEWMAN, M. E. J. Power laws, pareto distributions and zipf's law. *Contemporary physics*, Taylor & Francis, v. 46, n. 5, p. 323–351, 2005. ISSN 0010-7514.

NEWMAN, M. E. J. Networks. *[S.l.]: Oxford university press*, 2018.

OLBRICH, E.; BERTSCHINGER, N.; AY, N.; JOST, J. *A measure of statistical complexity based on predictive information*. 2008. 407 p.

PEREDA, E.; QUIROGA, R. Q.; BHATTACHARYA, J. Nonlinear multivariate analysis of neurophysiological signals. *Progress in neurobiology*, Elsevier, v. 77, n. 1-2, p. 1–37, 2005.

QUIROGA, R. Q.; PANZERI, S. Extracting information from neuronalpopulations: information theory and decoding approaches. *Nat Rev Neurosci*, v. 10, 2009.

SANTOS, F. A.; TEWARIE, P. K.; BAUDOT, P.; LUCHICCHI, A.; SOUZA, D. B. de; GIRIER, G.; MILAN, A. P.; BROEDERS, T.; CENTENO, E. G.; COFRE, R.; ROSAS, F. E.; CARONE, D.; KENNEDY, J.; STAM, C. J.; HILLEBRAND, A.; DESROCHES, M.; RODRIGUES, S.; SCHOONHEIM, M.; DOUW, L.; QUAX, R. Emergence of high-order functional hubs in the human brain. *bioRxiv*, Cold Spring Harbor Laboratory, 2023. Disponível em: <https://www.biorxiv.org/content/early/2023/02/12/2023.02.10.528083>.

SANTOS, F. A. N.; RAPOSO, E. P.; COUTINHO-FILHO, M. D.; COPELLI, M.; STAM, C. J.; DOUW, L. Topological phase transitions in functional brain networks. *hysical Review*, v. 100, 2019.

SCHNEIDMAN, E.; II, M. B.; SEGEV, R.; BIALEK, W. Nature. v. 440, p. 1007, 2003.

SEIDMAN, S. B.; FOSTER, B. L. A graph-theoretic generalization of the clique concept. *Journal of Mathematical sociology*, Taylor & Francis, v. 6, n. 1, p. 139–154, 1978.

SERRANO, D. H.; GóMEZ, D. S. Centrality measures in simplicial complexes: Applications of topological data analysis to network science. *Applied Mathematics and Computation*, v. 382, p. 125331, 2020.

SERRANO, D. H.; HERNáNDEZ-SERRANO, J.; GóMEZ, D. S. Simplicial degree in complex networks. applications of topological data analysis to network science. *Chaos, Solitons Fractals*, v. 137, p. 109839, 2020.

SOLOMONOFF, R.; RAPOPORT, A. Connectivity of random nets. *The bulletin of mathematical biophysics*, Springer, v. 13, n. 2, p. 107–117, 1951.

STEEN, M. V.; STEEN, M. v. Graph theory and complex networks, an introduction. *SIAM review, SIAM*, v. 144, 2010. Disponível em: <http://www.di.unipi.it/~ricci/book-watermarked.pdf>.

TEJAVIBULYA, L.; PETERSON, H.; GREENE, A.; GAO, S.; ROLISON, M.; NOBLE, S.; SCHEINOST, D. Large-scale differences in functional organization of left-and right-handed individuals using whole-brain, data-driven analysis of connectivity. *NeuroImage*, Elsevier, v. 252, p. 119040, 2022.

TIMME, N.; ALFORD, W.; FLECKER, B.; BEGGS, J. Multivariate information measures: an experimentalist's perspective. *Journal of Computational Neuroscience*, v. 36, p. 119–140, 2014. Disponível em: <http://www.beggslab.com/uploads/1/0/1/7/101719922/29timmeetal2013.pdf>.

TZOURIO-MAZOYER, N.; LANDEAU, B.; PAPATHANASSIOU, D.; CRIVELLO, F.; ETARD, O.; DELCROIX, N.; MAZOYER, B.; JOLIOT, M. Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain. *Neuroimage*, Elsevier, v. 15, n. 1, p. 273–289, 2002.

VANNI, S.; TANSKANEN, T.; SEPPÄ, M.; UUTELA, K.; HARI, R. Coinciding early activation of the human primary visual cortex and anteromedial cuneus. *Proceedings of the National Academy of Sciences*, National Acad Sciences, v. 98, n. 5, p. 2776–2780, 2001.

WATANABE, S. Information theoretical analysis of multivariate correlation. *IBM J Res Dev*, v. 4, p. 66–82, 1960.

WATTS, D. J.; STROGATZ, S. H. Collective dynamics of 'small-world' networks. *Nature, Nature Publishing Group*, v. 393, n. 6684, p. 440, 1998.

WILLIAMS, P. L.; BEER, R. D. *Nonnegative Decomposition of Multivariate Information*. 2010.

APÊNDICE A – TEORIA DE REDES

As redes (networks) oferecem uma estrutura universal para codificar informações sobre interações inerentes a um sistema complexo. Constituem uma teoria de enorme utilidade para a descrição deste tipo de sistema. De forma simplificada, estuda-se como as partes de um sistema complexo e suas relações explicam o comportamento coletivo do sistema e a relação deste com o ambiente que o circunda (BAR-YAM, 2022).

Por exemplo, em sistemas complexos como o cérebro humano ou a bolsa de valores, as interações entre as partes não são totalmente conhecidas. Nestes casos, a teoria de grafos torna-se uma excelente candidata para o estudo destes sistemas e, mais recentemente, a topologia algébrica vem contribuindo de forma significativa à análise topológica dos dados de sistemas complexos.

De forma simples, podemos definir uma *rede* (*network*) como uma coleção de objetos interconectados. Deste modo, os conceitos de rede e de grafo são praticamente indistinguíveis. Os objetos são os vértices do grafo que representa a rede e as conexões suas arestas.

Para uma visão geral da gama de aplicações da teoria de redes nos mais variados campos do saber, indicamos alguns trabalhos de Barabási (BARABÁSI, 2016; BARABÁSI; FRANGOS, 2014), de van Steen (STEEN; STEEN, 2010) e de Newman (NEWMAN, 2018; NEWMAN, 2003). Veremos abaixo os três modelos mais comuns de redes complexas: redes aleatórias (*random networks*), redes de mundo pequeno (*small-world networks*) e redes sem escala (*scale-free networks*).

A.1 REDES ALEATÓRIAS (*RANDOM NETWORKS*)

Redes aleatórias (*random networks*) são representadas por grafos nos quais a conexão de dois vértices por uma aresta depende de uma distribuição de probabilidade pré-estabelecida.

Embora o estudo de grafos aleatórios seja comumente associado aos nomes de Paul Erdös e Alfréd Rényi, com seus aclamados artigos nas décadas de 1950 e 1960 (ERDŐS; RÉNYI, 1959; ERDŐS; RÉNYI, 1960; ERDŐS; RÉNYI, 1964), Newman (NEWMAN, 2018) aponta como os pioneiros neste estudo seriam Solomonoff e Rapoport (SOLOMONOFF; RAPOPORT, 1951).

Para definir grafos aleatórios de Erdös e Rényi (*ER random graphs*), há pelo menos duas formas que consideramos equivalentes nesta dissertação. Na primeira, consideramos um grafo não orientado $ER(n, p)$ de ordem n e com a probabilidade de conexão entre dois vértices

distintos quaisquer igual a p; na segunda, dada uma quantidade m de arestas, a escolhemos aleatoriamente arestas incidindo a pares de vértices, distintos ou não.

Vejamos algumas de suas principais características e propriedades:

A distribuição do grau dos vértices em um grafo $ER(n,p)$ é dada por

$$P(k) = \binom{n-1}{k} p^k \, (1-p)^{n-1-k}$$

e o valor médio do grau dos vértices, representado na literatura por c, será dado por

$$\begin{aligned} c &= \sum_{k=1}^{n-1} k \, P(k) = \sum_{k=1}^{n-1} k \binom{n-1}{k} p^k \, (1-p)^{n-1-k} \\ &= p(n-1) \sum_{k=1}^{n-1} \binom{n-2}{k-1} p^{k-1} \, (1-p)^{n-1-k} \\ &= p(n-1) \sum_{r=0}^{n-2} \binom{n-2}{r} p^r \, (1-p)^{n-2-r} \\ \Rightarrow \quad & c = p(n-1) \end{aligned}$$

O resultado acima diz que, em média, o grau do vértice (número de arestas conectadas a um vértice) é igual ao produto da probabilidade de conexão entre dois vértices distintos quaisquer p pelo número $n-1$ dos demais vértices (NEWMAN, 2018).

Deste modo,

$$p = \frac{c}{n-1} \, . \tag{A.1}$$

Estatisticamente, observa-se que para $n \to \infty$ teremos que $p \to 0$ e, utilizando expansão em série de Taylor e a equação A.1, obtemos que

$$\ln\left[(1-p)^{n-1-k}\right] = (n-1-k) \, \ln\left(1 - \frac{c}{n-1}\right) \simeq \frac{(n-1-k)c}{n-1} \simeq -c \, . \tag{A.2}$$

Compondo com a função exponencial, obtemos $(1-p)^{n-1-k} \simeq e^{-c}$.

Além disso, para $n >> 0$, temos que

$$\binom{n-1}{k} = \frac{(n-1)!}{(n-1-k)!k!} \simeq \frac{(n-1)^k}{k!}$$

Portanto, podemos concluir que distribuição de grau para um número de vértices $n \gg 0$ é dada por:

$$P(k) = \frac{(n-1)^k}{k!} \, p^k \, e^{-c} = \frac{(n-1)^k}{k!} \left(\frac{c}{n-1}\right)^k \, e^{-c} = \frac{c^k \, e^{-c}}{k!} \, . \tag{A.3}$$

Observe que a expressão A.3 é exatamente a distribuição de Poisson. Dessa forma, para o número de vértices $n \gg 0$, o grafo de Erdös e Rényi $ER(n,p)$ tem uma distribuição de grau de Poisson e é chamado de grafo aleatório de Poisson.

Por fim, vale ressaltar que grafos aleatórios de Poisson apresentam várias limitações (NEWMAN, 2018). Por exemplo, grafos aleatórios não apresentam correlação entre os graus de vértices adjacentes, comportamento oposto ao de redes reais. Além disso, muitas das redes reais apresentem agrupamentos de vértices em "comunidades", enquanto que grafos aleatórios não apresentam tal comportamento. Mas talvez a maior discordância entre grafos aleatórios e grafos reais esteja em suas distribuições de grau. Isso ocorre porque grande parte das redes reais apresentam distribuição de cauda longa (*right-skewed distribution*), na qual a maioria dos vértices tem baixo grau e poucos vértices, na cauda da distribuição, possuem alto grau (*hubs*). Redes aleatórias, por sua vez, apresentam uma distribuição de grau do tipo Poisson, sem grandes discrepâncias de valor de grau dos vértices.

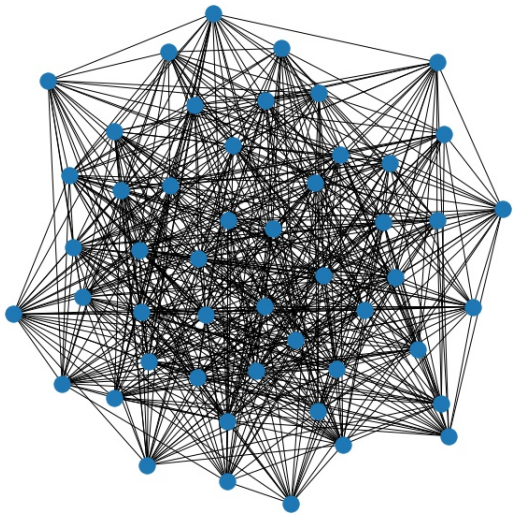


Figura 41 – Grafo $ER(50, 0.5)$ gerado com o pacote *Networkx* implementado na linguagem de programação *Python*

A.2 REDES DE MUNDO PEQUENO (*SMALL-WORLD NETWORKS*)

As redes de mundo pequeno (*small-world networks*) apresentam grande similaridade com os grafos aleatórios ER. Temos de fato um valor médio baixo para os comprimentos dos caminhos mínimos, contudo existe a tendência de se formar pequenos *clusters* na rede.

Watts e Strogatz (WATTS; STROGATZ, 1998) foram os primeiros a criar um algoritmo para este tipo de rede.

Segue a descrição do algoritmo:

Considere um conjunto de n vértices $\{v_1, ... v_n\}$ e um número par k . Para garantir que o grafo

terá poucas arestas, escolha $n \gg k \geq \ln(n)$.

1º passo: Ordene os n vértices em uma circunferência e, para cada vértice, ligue os primeiros $k/2$ vértices vizinhos no sentido horário e depois os $k/2$ vértices vizinhos no sentido anti-horário.

2º passo: Com uma probabilidade p, troque a aresta (u, v) por uma aresta (u, w) , em que w é um vértice diferente de u e (u, w) é uma aresta ainda não existente do grafo.

Vamos nos referir ao grafo aleatório de Watts-Strogatz por $WS(n, p, k)$.

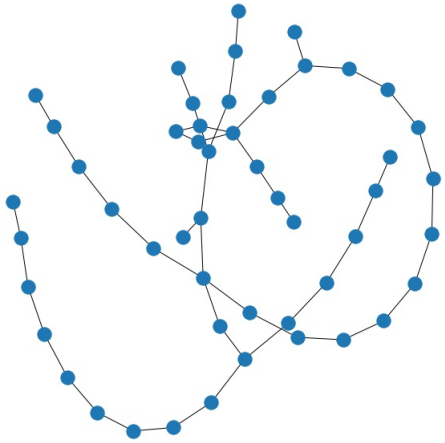


Figura 42 – Grafo de Watts-Strogatz $WS(50, 2, 0.3)$ gerado com o pacote *Networkx* implementado na linguagem de programação *Python*

A seguir apresentamos dois aspectos vantajosos do modelo WS em relação a outros modelos de redes, caso do ER, por exemplo.

A primeira é que redes WS apresentam transitividade compatível com redes reais, propriedade mensurável a partir do coeficiente de agrupamento da rede. Redes reais costumam ter alto coeficiente de agrupamento (NEWMAN, 2018). A segunda vantagem é a possibilidade em observar em redes WS o efeito de mundo pequeno (*small-world effect*). Esse efeito faz com que o comprimento de caminho entre qualquer par de vértices na rede seja pequeno, mesmo apresentando uma grande quantidade de vértices (BARABÁSI; FRANGOS, 2014).

A.3 REDES SEM ESCALA (*SCALE-FREE NETWORKS*)

Quando a distribuição dos graus dos vértices de em uma rede segue uma lei de potência dizemos que a própria rede segue uma lei de potências.

Assim, a probabilidade de um vértice arbitrário ter grau k será dada por

$$P(k) = C\, k^{-\alpha} \tag{A.4}$$

onde C e α são constantes, respectivamente, a constante de normalização e o expoente de escala.

Na literatura, as redes que seguem uma distribuição de lei de potência são também chamadas redes livres de escala (*scale-free networks*). A razão disso é que uma lei de potência é a única distribuição que tem a mesma forma qualquer que seja a escala em que olhemos (NEWMAN, 2005). Observa-se que muitas redes no mundo real são descritas por modelos livres de escala que possuem expoente de escala $2 < \alpha < 3$.

Há algumas maneiras de visualizar e detectar leis de potência associadas a redes livres de escala. A seguir apresentamos algumas (NEWMAN, 2018).

Uma primeira e mais simples forma de analisar leis de potência para redes livres de escala é utilizar um histograma da distribuição do grau com retângulos (*bins*) de largura maior, para que mais amostras sejam abrangidas por cada retângulo. Essa escolha implica em menos ruídos (oscilações) na cauda do histograma, mas reduz o nível de detalhamento da amostra. Uma alternativa para superar esse último problema seria utilizar retângulos de tamanhos diferentes em diferentes regiões do histograma.

Uma segunda forma de explorar leis de potência para redes livres de escala é construir uma função de distribuição cumulativa, definida por

$$\mathcal{P}(k) = \sum_{r=k}^{\infty} P(r)\,. \tag{A.5}$$

Aqui $\mathcal{P}(k)$ representa a fração dos vértices que possuem grau maior ou igual a k, ou ainda, a probabilidade de que um vértice escolhido aleatoriamente tenha grau maior ou igual a k. Uma das vantagens de se construir uma função de distribuição cumulativa $\mathcal{P}(k)$ é que quando a distribuição de grau $P(r)$ segue uma lei de potência com expoente de escala α, $\mathcal{P}(k)$ também segue uma lei de potência. A diferença é que o expoente de escala para $\mathcal{P}(k)$ será igual a $\alpha - 1$. Outra vantagem é que neste caso não é necessário se preocupar com a melhor escolha da largura dos retângulos em um histograma, preocupação associada à construção anterior.

Por outro lado, a escolha pela construção da função de distribuição cumulativa também tem suas desvantagens. A mais séria delas é que pontos consecutivos em um gráfico cumulativo são correlacionados e valores adjacentes não são todos independentes. Consequentemente, por exemplo, não podemos obter o valor do expoente de escala α da distribuição de lei de potência a partir do ajuste da inclinação da porção "reta" da curva. Em vez disso, podemos obter o expoente de escala α diretamente a partir dos dados, via expressão A.6 (NEWMAN,

2018; CLAUSET; SHALIZI; NEWMAN, 2009)

$$\alpha = 1 + N\left[\sum_i \ln\left(\frac{k_i}{k_{min} - \frac{1}{2}}\right)\right]^{-1} \tag{A.6}$$

onde k_{min} é o grau mínimo para o qual a lei de potência vale e N é o número de vértices com grau maior ou igual a k_{min}.

APÊNDICE B – TABELA DE TICKETS × EMPRESAS

Código/Ticket	*Nome da empresa*
1 – [AAPL]	Apple Inc.
2 – [ABBV]	AbbVie
3 – [ABT]	Abbott Laboratories
4 – [ACN]	Accenture
5 – [ADBE]	Adobe Inc.
6 – [AMGN]	Amgen
7 – [AMT]	American Tower Corporation
8 – [AMZN]	Amazon
9 – [AVGO]	Broadcom
10 – [AXP]	American Express
11 – [BA]	Boeing
12 – [BAC]	Bank of America Corp
13 – [BRK-A]	Berkshire Hathaway Inc Class A
14 – [C]	Citigroup Inc
15 – [CAT]	Caterpillar Inc.
16 – [CHTR]	Charter Communications
17 – [CMCSA]	Comcast
18 – [COST]	Costco
19 – [CRM]	Salesforce
20 – [CSCO]	Cisco Systems
21 – [CVS]	CVS Caremark
22 – [CVX]	Chevron
23 – [DD]	DuPont
24 – [DIS]	The Walt Disney Company
25 – [DOW]	Dow Jones Industrial Average
26 – [ENPH]	Enphase Energy Inc.
27 – [META]	Meta

Código/Ticket	*Nome da empresa*
28 – [GOOGL]	Alphabet Inc Class A
29 – [GS]	Goldman Sachs Group Inc.
30 – [HD]	The Home Depot
31 – [HON]	Honeywell
32 – [IBM]	IBM
33 – [INTC]	Intel
34 – [JNJ]	Johnson & Johnson
35 – [JPM]	JPMorgan Chase & Co
36 – [KO]	The Coca-Cola Company
37 – [LLY]	Eli Lilly and Company
38 – [LMT]	Lockheed Martin
39 – [MA]	Mastercard
40 – [MCD]	McDonald's
41 – [MDT]	Medtronic
42 – [MMM]	3M
43 – [MRK]	Merck Sharp and Dohme
44 – [MSFT]	Microsoft
45 – [NFLX]	Netflix
46 – [NKE]	Nike, Inc.
47 – [NVDA]	Nvidia
48 – [PEP]	PepsiCo
49 – [PFE]	Pfizer
50 – [PG]	Procter Gamble
51 – [TMO]	Thermo Fisher Scientific Inc.
52 – [UNH]	UnitedHealth Group
53 – [V]	Visa
54 – [VZ]	Verizon Communications
55 – [WMT]	Walmart

APÊNDICE C – CÓDIGO PYTHON

In [ ]:
```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import seaborn as sns
import yfinance as yf
import networkx as nx
from scipy.stats import zscore
import hypernetx as hnx
```

In [ ]:
```
## Rodados em:
## pandas version: 1.3.5
## numpy version: 1.18.2
## seaborn version: 0.10.0
## yfinance version: 0.2.18
## networkx version: 2.4
##
print('pandas version:',pd.__version__)
print('numpy version:',np.__version__)
print('seaborn version:',sns.__version__)
print('yfinance version:',yf.__version__)
print('networkx version:',nx.__version__)
```

In [ ]:
```
#Função para salva imagem de hypergrafo 3-uniforme ordenado pela ce
ntralidade
def hypergraph(tripl_filt,name_fig):
    tripl_filt=tripl_filt.sort_values(by='centralite', ascending=Fa
lse)
    lista_de_triplas = [list(row) for row in tripl_filt[['Var1','Va
r2','Var3']].values]
    H1=hnx.Hypergraph(lista_de_triplas)
    hnx.draw(H1)
    plt.savefig(name_fig)
```

In [ ]:
```
#Forma geral para gerar os resultados em função da métrica com os t
empos
def centralidade_tripletos_sinerg(triplets):
#Exemplo triplets=interac_avg(1,246)  e name=pos_TC

# Aplicar Z-score à coluna 'compute'
    triplets['compute_zscore'] = zscore(triplets['compute'])

    #triplets['compute_zscore'].hist(bins=30)

    triplets_ordenados = triplets.sort_values(by='compute_zscore',
ascending=False)
    tripl_filt=triplets_ordenados[:100]
    tripl_filt=tripl_filt.reset_index()
    #tripl_filt


#Criando grafo
    G=nx.Graph()
    for i in range(0,len(tripl_filt)-1):
        for j in range(i+1,len(tripl_filt)):
            C1={tripl_filt.at[i,'Var1'],tripl_filt.at[i,'Var2'],tri
pl_filt.at[i,'Var3']}
            C2={tripl_filt.at[j,'Var1'],tripl_filt.at[j,'Var2'],tri
pl_filt.at[j,'Var3']}
            if (len(C1 & C2)==2):
                G.add_edge(i,j)



    eigenvector_centrality = nx.eigenvector_centrality(G)
    sorted_eigenvector_centrality_reverse = dict(sorted(eigenvector
_centrality.items(),
                                                        key=lambda
item: float(item[1]), reverse=True))


    tripl_filt['centralite'] = dict(sorted(eigenvector_centrality.i
tems(), key=lambda item: float(item[0]),
                                           reverse=False)).values()


    return tripl_filt
```

In [ ]:
```
#Obtendo tabela com centralidades para redundancia
def centralidade_tripletos_redun(triplets):
#Exemplo triplets=interac_avg(1,246)  e name=pos_TC

# Aplicar Z-score à coluna 'compute'
    triplets['compute_zscore'] = zscore(triplets['compute'])

    #triplets['compute_zscore'].hist(bins=30)

    triplets_ordenados = triplets.sort_values(by='compute_zscore',
ascending=True)
    tripl_filt=triplets_ordenados[:100]
    tripl_filt=tripl_filt.reset_index()
    #tripl_filt


#Criando grafo
    G=nx.Graph()
    for i in range(0,len(tripl_filt)-1):
        for j in range(i+1,len(tripl_filt)):
            C1={tripl_filt.at[i,'Var1'],tripl_filt.at[i,'Var2'],tri
pl_filt.at[i,'Var3']}
            C2={tripl_filt.at[j,'Var1'],tripl_filt.at[j,'Var2'],tri
pl_filt.at[j,'Var3']}
            if (len(C1 & C2)==2):
                G.add_edge(i,j)

    eigenvector_centrality = nx.eigenvector_centrality(G)
    sorted_eigenvector_centrality_reverse = dict(sorted(eigenvector
_centrality.items(),
                                                        key=lambda
item: float(item[1]), reverse=True))


    tripl_filt['centralite'] = dict(sorted(eigenvector_centrality.i
tems(), key=lambda item: float(item[0]),
                                           reverse=False)).values()


    return tripl_filt
```

BÁSICO

In [ ]:
```
# Calcula a média da mutual information em determinado período
def mutua_avg(start,end):
    dataframe0=pd.read_csv('DADOS/MUTUA_INFO/'+str(start)+'.csv', i
ndex_col=0)
    for i in range(start+1,end+1):
        dataframe=pd.read_csv('DADOS/MUTUA_INFO/'+str(i)+'.csv', in
dex_col=0)
        dataframe0=dataframe0.add(dataframe, fill_value=0)
    dataframe0=dataframe0/((end-start)+1)
    return dataframe0
```

In [ ]:
```
# Calcula a média da interaction information em determinado período
def interac_avg(start,end):
    dataframe0=pd.read_csv('DADOS/II/'+str(start)+'.csv')
    for i in range(start+1,end+1):
        dataframe=pd.read_csv('DADOS/II/'+str(i)+'.csv')
        dataframe0['compute']=dataframe0['compute'].add(dataframe['
compute'], fill_value=0)
        #print(i)
    dataframe0['compute']=dataframe0['compute']/((end-start)+1)
    return dataframe0
```

In [ ]:
```
# Calcula a média da total correlation em determinado período
def correla_avg(start,end):
    dataframe0=pd.read_csv('DADOS/TC/'+str(start)+'.csv')
    for i in range(start+1,end+1):
        dataframe=pd.read_csv('DADOS/TC/'+str(i)+'.csv')
        dataframe0['compute']=dataframe0['compute'].add(dataframe['
compute'], fill_value=0)
        #print(i)
    dataframe0['compute']=dataframe0['compute']/((end-start)+1)
    return dataframe0
```

In [ ]:
```
#Calcula a entropia de uma variável x
def entropy(x):
    unique, counts = np.unique(x, return_counts=True)
    probs = counts / len(x)
    return -np.sum(probs * np.log2(probs))
```

In [ ]:
```
#Calcula a informação mutua entre duas variáveis x e y
def mutual_information(x, y):
    H_x_given_y = conditional_entropy(x, y)
    H_x = entropy(x)
    return H_x - H_x_given_y
```

In [ ]:
```
#Calcula a entropia condicional de x dado y
def conditional_entropy(x, y):
    unique_y, counts_y = np.unique(y, return_counts=True)
    probs_y = counts_y / len(y)
    H_x_given_y = 0
    for i in range(len(unique_y)):
        y_val = unique_y[i]
        p_y = probs_y[i]
        x_given_y = x[y == y_val]
        p_x_given_y = len(x_given_y) / len(y)
        H_x_given_y += p_y * entropy(x_given_y) * p_x_given_y
    return H_x_given_y
```

In [ ]:
```
#Calcula a entropia conjunta de x e y
def joint_entropy_2(x, y):
    combined_xy = np.column_stack((x, y))
    unique, counts = np.unique(combined_xy, axis=0, return_counts=T
rue)
    probs = counts / len(combined_xy)
    return -np.sum(probs * np.log2(probs))
```

In [ ]:
```
#Calcula a entropia conjunta de x, y e z
def joint_entropy_3(x, y, z):
    combined_xyz = np.column_stack((x, y, z))
    unique, counts = np.unique(combined_xyz, axis=0, return_counts=
True)
    probs = counts / len(combined_xyz)
    return -np.sum(probs * np.log2(probs))
```

In [ ]:
```
#Calcula a informação de interação entre as variáveis x, y e z
def interaction_information(x, y, z):
    H_x = entropy(x)
    H_y = entropy(y)
    H_z = entropy(z)
    H_xy = joint_entropy_2(x, y)
    H_xz = joint_entropy_2(x, z)
    H_yz = joint_entropy_2(y, z)
    H_xyz = joint_entropy_3(x, y, z)

    II_xyz = -H_x - H_y - H_z + H_xy + H_xz + H_yz - H_xyz

    return II_xyz
```

In [ ]:
```
#Calcula a correlação total entre as variáveis x, y e z
def total_correlation(x, y, z):
    H_x = entropy(x)
    H_y = entropy(y)
    H_z = entropy(z)
    H_xyz = joint_entropy_3(x, y, z)

    TC_xyz = H_x + H_y + H_z - H_xyz

    return TC_xyz
```

In [ ]:
```
#Retorna uma tabela com os tripletos calculados usando interaction
information
def  interac_trip(data):
#interaction_information
# Loop for para criar as linhas do DataFrame
    dados_linhas=[]
    II_DF=pd.DataFrame()
    for i in range(0,len(data.columns)-2):
        for j in range(i+1,len(data.columns)-1):
            for k in range(j+1,len(data.columns)):
                linha={'Var1': data.columns[i],'Var2': data.columns
[j], 'Var3': data.columns[k],
                       'compute': interaction_information(data[data
.columns[i]],data[data.columns[j]],
                                                          data[data
.columns[k]])}
                dados_linhas.append(linha)
# Crie um DataFrame a partir da lista de dados de linhas
    II_DF = pd.DataFrame(dados_linhas)
    return II_DF
```

In [ ]:
```
#Retorna uma tabela com os tripletos calculados usando total correl
ation
def  total_corr_trip(data):
#interaction_information
#Loop for para criar as linhas do DataFrame
    dados_linhas=[]
    TC_DF=pd.DataFrame()
    for i in range(0,len(data.columns)-2):
        for j in range(i+1,len(data.columns)-1):
            for k in range(j+1,len(data.columns)):
                linha={'Var1': data.columns[i],'Var2': data.columns
[j], 'Var3': data.columns[k],
                       'compute': total_correlation(data[data.colum
ns[i]],data[data.columns[j]],
                                                    data[data.colum
ns[k]])}
                dados_linhas.append(linha)
#Crie um DataFrame a partir da lista de dados de linhas
    TC_DF = pd.DataFrame(dados_linhas)
    return TC_DF
```

In [ ]:
```
#Retorna uma tabela com os valores médios para os dados pré-selecio
nados
def mean_window(dataframe,start,end,window=20):
#Suponha que você tenha uma lista de DataFrames chamada dataframes
    dataframes = []
#inicializar um DataFrame vazio para armazenar os valores de CORR
    count=start
#calcular o cp para cada par de variáveis
    for i in range(start,len(dataframe)+1):
        if (window+i)<=len(dataframe):
            if count <= end:
                dataframes.append(dataframe[i:i+window])
                count=count+1
#Inicialize o DataFrame médio com o primeiro DataFrame da lista
    media_dataframe = dataframes[0]
#Itere sobre os DataFrames restantes na lista e some-os ao DataFram
e médio
    for df in dataframes[1:]:
        media_dataframe = media_dataframe.add(df, fill_value=0)
#Divida o DataFrame médio pelo número de DataFrames para calcular a
média
    numero_de_dataframes = len(dataframes)
    media_dataframe = media_dataframe / numero_de_dataframes
    return(media_dataframe)
```

In [ ]:
```
#Forma geral para gerar os resultados em função da métrica com os t
empos - sineriga
def estudo_tripletos_sinerg(triplets,name):
#Exemplo triplets=interac_avg(1,246)  e name=pos_TC

#Aplicar Z-score à coluna 'compute'
```

```
    triplets['compute_zscore'] = zscore(triplets['compute'])

    triplets['compute_zscore'].hist(bins=30)

    triplets_ordenados = triplets.sort_values(by='compute_zscore',
ascending=False)
    tripl_filt=triplets_ordenados[:100]
    tripl_filt=tripl_filt.reset_index()
    print(tripl_filt[0:4])


#Criando grafo
    G=nx.Graph()
    for i in range(0,len(tripl_filt)-1):
        for j in range(i+1,len(tripl_filt)):
            C1={tripl_filt.at[i,'Var1'],tripl_filt.at[i,'Var2'],tri
pl_filt.at[i,'Var3']}
            C2={tripl_filt.at[j,'Var1'],tripl_filt.at[j,'Var2'],tri
pl_filt.at[j,'Var3']}
            if (len(C1 & C2)==2):
                G.add_edge(i,j)



#Definindo tamanho e titulo da figura
    plt.figure(figsize=(10, 10))
    plt.title('a)')
#Configuração estética do grafo
    node_color = 'skyblue'
    node_size = 100
    edge_color = 'gray'
#Escolha um algoritmo de posicionamento do grafo
    pos = nx.spring_layout(G)
#Desenhe o grafo com o posicionamento específico
    nx.draw(G, pos=pos, with_labels=True, node_color=node_color, no
de_size=node_size, edge_color=edge_color)
    plt.savefig('grafo_trip_'+name+'.png')
#Exiba o gráfico
    plt.show()



    eigenvector_centrality = nx.eigenvector_centrality(G)
    sorted_eigenvector_centrality_reverse = dict(sorted(eigenvector
_centrality.items(),
                                                        key=lambda
item: float(item[1]), reverse=True))



#Iterar sobre os primeiros n itens
    for key, value in list(sorted_eigenvector_centrality_reverse.it
ems())[:30]:
```

```
        print(f'Centralidade do tripleto: {key}, Valor: {value}')

    print('\n')


    LIST=[]
    for i in list(sorted_eigenvector_centrality_reverse)[:n]:
        LIST.append(tripl_filt.at[i,'Var1'])
        LIST.append(tripl_filt.at[i,'Var2'])
        LIST.append(tripl_filt.at[i,'Var3'])
        print(i,tripl_filt.at[i,'Var1'],tripl_filt.at[i,'Var2'],tri
pl_filt.at[i,'Var3'])
    LIST_ONLY=list(set(LIST))



#Extraia as chaves e valores do dicionário
    categorias = list(sorted_eigenvector_centrality_reverse.keys())
    valores = list(sorted_eigenvector_centrality_reverse.values())

#Crie um histograma
    plt.bar(categorias, valores)

#Adicione rótulos aos eixos
    plt.xlabel('Tripletos')
    plt.ylabel('Valores')

#Exiba o histograma
    plt.show()


#Crie um histograma
    plt.hist(valores, bins=15, edgecolor='black', alpha=0.7)

#Configure rótulos e título
    plt.xlabel('Valores')
    plt.ylabel('Frequência')
    plt.title('Histograma de Frequência')
    plt.savefig('Histo_'+name+'.png')

#Exiba o histograma
    plt.show()
    return
```

In [ ]:
```
#Forma geral para gerar os resultados em função da métrica com os t
empos
def estudo_tripletos_redun(triplets,name):
#Exemplo triplets=interac_avg(1,246)  e name=pos_TC

#Aplicar Z-score à coluna 'compute'
    triplets['compute_zscore'] = zscore(triplets['compute'])
```

```
    triplets['compute_zscore'].hist(bins=30)

    triplets_ordenados = triplets.sort_values(by='compute_zscore',
ascending=True)
    tripl_filt=triplets_ordenados[:100]
    tripl_filt=tripl_filt.reset_index()
    print(tripl_filt[:4])


#Criando grafo
    G=nx.Graph()
    for i in range(0,len(tripl_filt)-1):
        for j in range(i+1,len(tripl_filt)):
            C1={tripl_filt.at[i,'Var1'],tripl_filt.at[i,'Var2'],tri
pl_filt.at[i,'Var3']}
            C2={tripl_filt.at[j,'Var1'],tripl_filt.at[j,'Var2'],tri
pl_filt.at[j,'Var3']}
            if (len(C1 & C2)==2):
                G.add_edge(i,j)



#Definindo tamanho e titulo da figura
    plt.figure(figsize=(10, 10))
    plt.title('a)')
#Configuração estética do grafo
    node_color = 'skyblue'
    node_size = 100
    edge_color = 'gray'
#Escolha um algoritmo de posicionamento do grafo
    pos = nx.spring_layout(G)
#Desenhe o grafo com o posicionamento específico
    nx.draw(G, pos=pos, with_labels=True, node_color=node_color, no
de_size=node_size, edge_color=edge_color)
    plt.savefig('grafo_trip_'+name+'.png')
#Exiba o gráfico
    plt.show()



    eigenvector_centrality = nx.eigenvector_centrality(G)
    sorted_eigenvector_centrality_reverse = dict(sorted(eigenvector
_centrality.items(),
                                                        key=lambda
item: float(item[1]), reverse=True))



#Iterar sobre os primeiros n itens
    for key, value in list(sorted_eigenvector_centrality_reverse.it
ems())[:30]:
        print(key,tripl_filt.at[key,'Var1'],tripl_filt.at[key,'Var2
'],tripl_filt.at[key,'Var3'],value)
```

```
    print('\n')


    LIST=[]
    for i in list(sorted_eigenvector_centrality_reverse)[:n]:
        LIST.append(tripl_filt.at[i,'Var1'])
        LIST.append(tripl_filt.at[i,'Var2'])
        LIST.append(tripl_filt.at[i,'Var3'])
    LIST_ONLY=list(set(LIST))



# Extraia as chaves e valores do dicionário
    categorias = list(sorted_eigenvector_centrality_reverse.keys())
    valores = list(sorted_eigenvector_centrality_reverse.values())

# Crie um histograma
    plt.bar(categorias, valores)

# Adicione rótulos aos eixos
    plt.xlabel('Tripletos')
    plt.ylabel('Valores')

# Exiba o histograma
    plt.show()


# Crie um histograma
    plt.hist(valores, bins=15, edgecolor='black', alpha=0.7)

# Configure rótulos e título
    plt.xlabel('Valores')
    plt.ylabel('Frequência')
    plt.title('Histograma de Frequência')
    plt.savefig('Histo_'+name+'.png')

#Exiba o histograma
    plt.show()
    return
```

In [ ]:
```
#Selecionamos os tickets das 55 empresas da S&P500 no ano de 2019 a
2021.
tickets_select = ['AAPL', 'ABBV', 'ABT', 'ACN', 'ADBE', 'AMGN', 'AM
T', 'AMZN',
                  'AVGO', 'AXP', 'BA', 'BAC', 'BRK-A', 'C', 'CAT',
'CHTR', 'CMCSA',
                  'COST', 'CRM', 'CSCO', 'CVS', 'CVX' ,'DD', 'DIS',
'DOW', 'ENPH', 'META',
                  'GOOGL', 'GS', 'HD', 'HON', 'IBM', 'INTC', 'JNJ',
'JPM', 'KO', 'LLY', 'LMT',
                  'MA', 'MCD', 'MDT', 'MMM', 'MRK', 'MSFT', 'NFLX',
'NKE', 'NVDA', 'PEP', 'PFE',
                  'PG', 'TMO', 'UNH', 'V', 'VZ', 'WMT']
```

In [ ]:
```
#Baixamos os dados e verificamos se há lacunas
data_total=yf.download(tickets_select, '2019-04-01','2021-12-31')
print('Falta de dados:',data_total.isna().sum().sum())
#data_total.to_csv("data_total.csv")
```

In [ ]:
```
#Plot dos dados médios por dia em todo o período baixado
data_total['Adj Close'].mean(1).plot()
```

In [ ]:
```
#Selecionamos a coluna adj_close, tal seleção faz sentido, pela def
inição do adj_close.
DF=(data_total['Adj Close'].pct_change()*100)[1:]

#Vamos discretizar os dados
DF_bin=DF.apply(lambda s: pd.qcut(s, 20, labels=False))

#Verificando se há células vazias
print(DF_bin.isna().sum().sum())

#Imprimos algumas linhas do nosso dado discretizado
DF_bin.head()

#Salva os dados discretizados
#DF_bin.to_csv("data_total_bin.csv")
```

In [ ]:
```
#Defini tamanho da janela de tempo
window=20
#Cria uma tabela vazia
DFFF=pd.DataFrame()
#Cria uma tabela separando-a em janelas de tempo
DFF=DF.rolling(window).mean()

#Importante para o gráfico de comparação para outras medidas
#Copia DFF em DFF1
DFF1=DFF.copy()

#Para cada empresa é criada uma coluna de entropia para cada janela
de tempo
for name in DF_bin.columns:
    DFF[name + '_entropy'] = DF_bin[name].rolling(window).apply(lam
bda x: entropy(x))
    DFFF[name + '_entropy'] = DF_bin[name].rolling(window).apply(la
mbda x: entropy(x))
#DFF.to_csv("DF_WINDOW-50_SHANNON.csv")
```

In [ ]:
```
#Plota a entropia média das 55 empresas no período baixado
plt.figure(figsize=(8, 3))
plt.title('Entropia Média das 55 empresas')
plt.plot(DFFF[19:].mean(1))
plt.xticks(rotation=45)
plt.savefig('Entropia_media.png')
```

In [ ]:
```
#Plota a entropia da apple calculada em janelas de tempo de tamanho
20
plt.title('Entropia em time moving window=20 APPLE')
DFF[19:]['AAPL_entropy'].plot()
plt.savefig('AAPL_entropy_delta=20.png')
```

Mutual Information (MI)

In [ ]:
```
#Retorna uma tabela com a mutual information calculada
def mutual_info_window(data):
# inicializar um DataFrame vazio para armazenar os valores de MI
    mi_df = pd.DataFrame(columns=data.columns, index=data.columns)
# calcular o MI para cada par de variáveis
    for col1 in data.columns:
        for col2 in data.columns:
            if col1 != col2:
                mi = mutual_information(data[col1], data[col2])
                mi_df.at[col1, col2] = mi
    mi_df=mi_df.fillna(0)
    return mi_df
```

In [ ]:
```
#Calcula a mutual information para todas as janelas de tempo (de ta
manho 20) e armazaena em uma pasta
#inicializar um DataFrame vazio para armazenar os valores de MI
mi_df = pd.DataFrame(columns=DF_bin.columns, index=DF_bin.columns)
count=1
# calcular o MI para cada par de variáveis
for i in range(0,len(DF_bin)+1):
    if (window+i)<=len(DF_bin):
        for col1 in DF_bin.columns:
            for col2 in DF_bin.columns:
                if col1 != col2:
                    mi = mutual_information(DF_bin[col1][i:i+window
], DF_bin[col2][i:i+window])
                    mi_df.at[col1, col2] = mi
        #mi_df.to_csv("/content/drive/MyDrive/Financial Tranding/Fi
nancial projec B3/Pasta Sem Título 2/MUTUA_INFO-BINS_20/" + str(cou
nt)+ ".csv") #colab
        mi_df.to_csv("MUTUA_INFO/" + str(count)+ ".csv") #maquina f
ísica
        count=count+1
        mi_df = pd.DataFrame(columns=DF_bin.columns, index=DF_bin.c
olumns)
```

In [ ]:
```
#Valor minimo de todas as mutual information calculadas
MIN=min(list([mutua_avg(1,246).min().min(),mutua_avg(213,322).mi
n().min(),mutua_avg(444,636).min().min()]))
```

In [ ]:
```
#Valor maximo de todas as mutual information calculadas
MAX=max(list([mutua_avg(1,246).max().max(),mutua_avg(213,322).ma
x().max(),mutua_avg(444,636).max().max()]))
```

In [ ]:
```
#Mapa de calor médio das Mutual Information dentre janelas 1 a 246
(PRE INSTABILIDADE)
sns.heatmap(mutua_avg(1,246),cmap='RdYlGn',vmin=MIN,vmax=MAX)
```

In [ ]:
```
#Mapa de calor médio das Mutual Information dentre janelas 213 a 32
2 (DURANTE INSTABILIDADE)
sns.heatmap(mutua_avg(213,322),cmap='RdYlGn',vmin=MIN,vmax=MAX)
```

In [ ]:
```
#Mapa de calor médio das Mutual Information dentre janelas 444 a 63
6 (POS INSTABILIDADE)
sns.heatmap(mutua_avg(444,636),cmap='RdYlGn',vmin=MIN,vmax=MAX)
```

Interaction Information (II)

+

Total Correlation (TC)

In [ ]:
```
#Calcula a interaction information e total correlation armazenando,
cada janela, em determinada pasta
#(para cada métrica)
# Inicialize uma lista vazia para armazenar os dados de cada linha
counte=1
# Loop for para criar as linhas do DataFrame
for e in range(0,len(DF_bin)-window+1):
    dados_linhas_II=[]
    dados_linhas_TC=[]

    II_DF=pd.DataFrame()
    TC_DF=pd.DataFrame()
    for i in range(0,len(DF_bin.columns)-2):
        for j in range(i+1,len(DF_bin.columns)-1):
            for k in range(j+1,len(DF_bin.columns)):
                linha_II={'Var1': DF_bin.columns[i],'Var2': DF_bin.
columns[j], 'Var3': DF_bin.columns[k],
                          'compute': interaction_information(DF_bin
[DF_bin.columns[i]][e:e+window],
                                                             DF_bin
[DF_bin.columns[j]][e:e+window],
                                                             DF_bin
[DF_bin.columns[k]][e:e+window])}
                linha_TC={'Var1': DF_bin.columns[i],'Var2': DF_bin.
columns[j], 'Var3': DF_bin.columns[k],
                          'compute': total_correlation(DF_bin[DF_bi
n.columns[i]][e:e+window],
                                                       DF_bin[DF_bi
n.columns[j]][e:e+window],
                                                       DF_bin[DF_bi
n.columns[k]][e:e+window])}
                dados_linhas_II.append(linha_II)
                dados_linhas_TC.append(linha_TC)
# Crie um DataFrame a partir da lista de dados de linhas
    II_DF = pd.DataFrame(dados_linhas_II)
    TC_DF = pd.DataFrame(dados_linhas_TC)

    #II_DF.to_csv("/content/drive/MyDrive/Financial Tranding/Financ
ial projec B3/Pasta Sem Título 2/II-BINS_20/" + str(counte)+ ".csv"
) #COLAB
    II_DF.to_csv("DADOS/II/" + str(counte)+ ".csv") #PC FISICO
    TC_DF.to_csv("DADOS/TC/" + str(counte)+ ".csv") #PC FISICO

    counte=counte+1
    II_DF=pd.DataFrame()
    TC_DF=pd.DataFrame()
```

Estudo dos TRIPLETOS-II e TC

PRE

In [ ]:
```
#Localiza o período de tempo antes da instabilidade
# Obs: Em 11 de março de 2020, a COVID-19 foi caracterizada pela OM
S como uma pandemia
#Start: Start: 2019-04-02 End: 2019-05-01 | windows:1
#End: Start: 2020-03-23 End: 2020-04-21 | windows:246
DF.mean(1).plot()
plt.axvline(x='2019-04-02', color='red', linestyle='--', label='Dat
a Inicial')
plt.axvline(x='2020-02-07', color='red', linestyle='--', label='Dat
a Final')
plt.xlabel('Data')
plt.ylabel('Média dos valores percentuais')
plt.title('Gráfico com data destacada')
plt.grid(True)
plt.show()
```

In [ ]:
```
#Calcula tripletos, grafo, EC sinergicos pre instabilidade com Inte
raction Information
estudo_tripletos_sinerg(interac_avg(1,246),'PRE-SINERGIA-II')
```

In [ ]:
```
#Calcula tripletos, grafo, EC sinergicos pre instabilidade com Tota
l Correlation
estudo_tripletos_sinerg(correla_avg(1,246),'PRE-SINERGIA-TC')
```

In [ ]:
```
#Calcula tripletos, grafo, EC redundante pre instabilidade com Inte
raction Information
estudo_tripletos_redun(interac_avg(1,246),'PRE-REDUNDANCIA-II')
```

In [ ]:
```
#Calcula tripletos, grafo, EC redundantes pre instabilidade com Tot
al Correlation
estudo_tripletos_redun(correla_avg(1,246),'PRE-REDUNDANCIA_TC')
```

DURANTE

In [ ]:
```
#Localiza o período de tempo durante instabilidade
#Start: Start: 2020-02-04 End: 2020-03-04 | windows:213
#END: Start: 2020-07-10 End: 2020-08-07 | windows:322
DF.mean(1).plot()
plt.axvline(x='2020-02-04', color='red', linestyle='--', label='Dat
a Inicial')
plt.axvline(x='2020-08-07', color='red', linestyle='--', label='Dat
a Final')
plt.xlabel('Data')
plt.ylabel('Média dos valores percentuais')
plt.title('Gráfico com Data Enfatizada')
plt.grid(True)
plt.show()
```

In [ ]:
```
#Calcula tripletos, grafo, EC sinergicos durante instabilidade com
Interaction Information
estudo_tripletos_sinerg(interac_avg(213,322),'DURANTE-SINERGIA-II')
```

In [ ]:
```
#Calcula tripletos, grafo, EC sinergicos durante instabilidade com
Total Correlation
estudo_tripletos_sinerg(correla_avg(213,322),'DURANTE-SINERGIA-TC')
```

In [ ]:
```
#Calcula tripletos, grafo, EC redundantes durante instabilidade com
Interaction Information
estudo_tripletos_redun(interac_avg(213,322),'DURANTE-REDUNDANCIA-I
I')
```

In [ ]:
```
#Calcula tripletos, grafo, EC redundantes durante instabilidade com
Total Correlation
estudo_tripletos_redun(correla_avg(213,322),'DURANTE-REDUNDANCIA-T
C')
```

POS

In [ ]:
```
#Localiza o período de tempo pos instabilidade
#Start: Start: 2021-01-04 End: 2021-02-02 | windows:444
#END: Start: 2021-11-18 End: 2021-12-17 | windows:666
DF.mean(1).plot()
plt.axvline(x='2021-01-04', color='red', linestyle='--', label='Dat
a Inicial')
plt.axvline(x='2021-10-07', color='red', linestyle='--', label='Dat
a Final')
plt.xlabel('Data')
plt.ylabel('Média dos valores percentuais')
plt.title('Gráfico com Data Enfatizada')
plt.grid(True)
plt.show()
```

In [ ]:
```
#Calcula tripletos, grafo, EC sinergicos pos instabilidade com Inte
raction Information
estudo_tripletos_sinerg(interac_avg(444,636),'POS-SINERGIA-II')
```

In [ ]:
```
#Calcula tripletos, grafo, EC sinergicos pos instabilidade com Tota
l Correlation
estudo_tripletos_sinerg(correla_avg(444,636),'POS-SINERGIA-TC')
```

In [ ]:
```
#Calcula tripletos, grafo, EC redundantes pos instabilidade com Int
eraction Information
estudo_tripletos_redun(interac_avg(444,636),'POS-REDUNDANCIA-II')
```

In [ ]:
```
#Calcula tripletos, grafo, EC redundantes pos instabilidade com Tot
al Correlation
estudo_tripletos_redun(correla_avg(444,636),'POS-REDUNDANCIA-TC')
```

Hypergrafos

In [ ]:
```
#Exemplos de hypergrafos
H1 = hnx.Hypergraph({'e1': ['v1', 'v2', 'v3'], 'e2': ['v2', 'v3', '
v4']})
H2 = hnx.Hypergraph({'e1': ['v1', 'v3', 'v5'], 'e2': ['v2', 'v5', '
v6']})
H3 = hnx.Hypergraph({'e1': ['v4', 'v5', 'v6'], 'e2': ['v6', 'v7', '
v8']})
hnx.draw(H1)
```

In [ ]:
```
#
# INTERACTION INFORMATION
#
x_1_II=centralidade_tripletos_sinerg(interac_avg(1,246))
y_1_II=centralidade_tripletos_sinerg(interac_avg(213,322))
z_1_II=centralidade_tripletos_sinerg(interac_avg(444,636))

x_2_II=centralidade_tripletos_redun(interac_avg(1,246))
y_2_II=centralidade_tripletos_redun(interac_avg(213,322))
z_2_II=centralidade_tripletos_redun(interac_avg(444,636))

#
# TOTAL CORRELATION
#
x_1_TC=centralidade_tripletos_sinerg(correla_avg(1,246))
y_1_TC=centralidade_tripletos_sinerg(correla_avg(213,322))
z_1_TC=centralidade_tripletos_sinerg(correla_avg(444,636))

x_2_TC=centralidade_tripletos_redun(correla_avg(1,246))
y_2_TC=centralidade_tripletos_redun(correla_avg(213,322))
z_2_TC=centralidade_tripletos_redun(correla_avg(444,636))
```

As seguintes linhas plotam as 20 hyperarestas mais forte para cada métrica e período

In [ ]:
```
hypergraph(x_1_II[:20],'20-HUB_HYPERGRAFO-PRE-SINER-II.jpg')
```

In [ ]:
```
hypergraph(y_1_II[:20],'20-HUB_HYPERGRAFO-DURANTE-SINER-II.jpg')
```

In [ ]:
```
hypergraph(z_1_II[:20],'20-HUB_HYPERGRAFO-POS-SINER-II.jpg')
```

In [ ]:
```

```

In [ ]:
```
hypergraph(x_2_II[:20],'20-HUB_HYPERGRAFO-PRE-REDUN-II.jpg')
```

In [ ]:
```
hypergraph(y_2_II[:20],'20-HUB_HYPERGRAFO-DURANTE-REDUN-II.jpg')
```

In [ ]:
```
hypergraph(z_2_II[:20],'20-HUB_HYPERGRAFO-POS-REDUN-II.jpg')
```

In [ ]:
```

```

In [ ]:
```
hypergraph(x_1_TC[:20],'20-HUB_HYPERGRAFO-PRE-SINER-TC.jpg')
```

In [ ]:
```
hypergraph(y_1_TC[:20],'20-HUB_HYPERGRAFO-DURANTE-SINER-TC.jpg')
```

In [ ]:
```
hypergraph(z_1_TC[:20],'20-HUB_HYPERGRAFO-POS-SINER-TC.jpg')
```

In [ ]:
```

```

In [ ]:
```
hypergraph(x_2_TC[:20],'20-HUB_HYPERGRAFO-PRE-REDUN-TC.jpg')
```

In [ ]:
```
hypergraph(y_2_TC[:20],'20-HUB_HYPERGRAFO-DURANTE-REDUN-TC.jpg')
```

In [ ]:
```
hypergraph(z_2_TC[:20],'20-HUB_HYPERGRAFO-POS-REDUN-TC.jpg')
```

In [ ]:
```
print('Empresas mais fortes em PRE: ')
for i in LIST_PRE_ONLY:
    print(i,end=' ')
print('\n\nEmpresas mais fortes em CAOS: ')
for i in LIST_CAOS_ONLY:
    print(i,end=' ')
print('\n\nEmpresas mais forte em POS: ')
for i in LIST_POS_ONLY:
    print(i,end=' ')
```

In [ ]:
```
print('Interseções\n')
print('PRE E POS:')
for i in (set(LIST_PRE_ONLY)&set(LIST_POS_ONLY)):
    print(i,end=' ')

print('\n\nPRE E CAOS:')
for i in (set(LIST_PRE_ONLY)&set(LIST_CAOS_ONLY)):
    print(i,end=' ')

print('\n\nCAOS E POS:')
for i in (set(LIST_CAOS_ONLY)&set(LIST_POS_ONLY)):
    print(i,end=' ')

print('\n\nPRE E CAOS E POS:')
for i in (set(LIST_PRE_ONLY)&set(LIST_CAOS_ONLY)&set(LIST_POS_ONLY)
):
    print(i,end=' ')
```