

Relatório de atividades do trabalho de formatura - Julho

Henrique S Requejo

24/07/2020

Conteúdo

1	Resumo	1
2	O que é o R Markdown e como foi meu workflow para gerar esse documento	2
2.1	Sobre o R Markdown e onde começar	2
2.2	Workflow	2
3	Relembrando sobre a modularidade multicamada e entendendo a variável G	4
3.1	Modularidade multicamada	4
3.2	Variável G	5
3.3	Variável G_{norm}	7
4	Visão geral da rede Morcegos-plantas	7
5	Resultados da rede Morcegos-plantas	8
5.1	Distribuição de G_{norm}	8
5.2	Variação de \bar{G} por ω	9
5.3	Seleção das espécies com maior G_{norm}	12
6	Visão geral da rede Formigas-planta	14
7	Resultados da rede Formigas-planta	15
7.1	Distribuição de G_{norm}	15
7.2	Variação de \bar{G} por ω	16
7.3	Seleção das espécies com maior G_{norm}	19
8	Discussão	21
9	Próximas etapas	21
9.1	Agosto	21
9.2	Setembro	21

1 Resumo

Os principais objetivos deste mês eram entender como o R Markdown funciona, avaliar a rede de formigas-plantas, entender como o parâmetro de resolução γ da modularidade multica-mada afeta as curvas de decaimento de \overline{G} por ω e tentar visualizar o motivo de os valores de G estarem aumentando com ω para alguns nós. O último objetivo não foi cumprido ainda, pois fo-quei em aprender R Markdown, o que foi muito bom, pois esta é uma ferramenta que gostei muito e acredito que irei escrever meu TF inteiro nela. Este documento foi todo feito no R Markdown usando as redes morcego-planta e a rede de formigas-plantas . Acredito que este documento e seus códigos possam servir como exemplo e ajudar outras pessoas a entender o funcionamento do R Markdown e como gerar um relatório automatizado das suas análises em PDF, pois aqui estão con-tidos formulas matemáticas no estilo Latex, figuras simples, figuras múltiplas, tabelas e citações. As curvas de decaimento de \overline{G} por ω foram extraídas para diferentes valores de γ , gerando agora uma família de curvas para cada espécie da rede. Isso faz com que a forma de calcular o valor de G_{norm} seja levemente alterada para considerar as variações de G_{norm} em relação a γ . Apesar de algumas variações, como esperado, a maioria das espécies que haviam sido selecionadas como boas conectoras entre camadas (permanecem em mais de um módulo mesmo aumentando a força do acoplamento entre camadas) com γ constante também foram selecionadas quando variamos γ . Neste documento também estão algumas formas diferentes de visualizar as famílias de curvas geradas, pois não tenho certeza qual a melhor forma de apresentar a família de curvas.

2 O que é o R Markdown e como foi meu workflow para gerar esse documento

2.1 Sobre o R Markdown e onde começar

R Markdown é um formato de arquivo para gerar documentos dinâmicos com o R. Um documento R Markdown é escrito em Markdown e contém pedaços de códigos do R (Grolemund, 2014), ou seja, é um editor de texto em que você pode rodar pedaços de códigos R e usar suas saídas para produzir figuras, tabelas ou mesmo mostrar variáveis dentro do próprio texto, muito prático! O R Markdown suporta vários formatos estáticos de saída, como HTML, MS Word, ODT, PDF, entre outros. Para o caso de PDF, o R Markdown utiliza o motor Latex para gerar os documentos. Isso é muito bom, pois podemos utilizar quase todas (talvez todas?) as funcionalidades e praticidades da escrita em Latex. Utilizei sempre o R Studio para realizar as compilações. O [site oficial do R Markdown](https://rmarkdown.rstudio.com/)¹ e o livro R Markdown: The Definitive Guide (Xie et al., 2019), que é gratuito para [ler online](https://bookdown.org/yihui/rmarkdown/)² são muito bons para aprender e foram as fontes que mais consultei durante a elaboração deste relatório.

Um pequeno problema que tive foi que não consegui compilar o documento RMarkdown utilizando diretamente um diretório sincronizado na nuvem (google drive). Ocorre um problema quando o R Studio tenta acessar um arquivo que está sendo sincronizado, impedindo o documento de ser gerado. Tive que trabalhar em um diretório local.

2.2 Workflow

Fui aprendendo com tentativa e erro a melhor forma de gerar o documento. Primeiro, comecei como sugerido na maioria das minhas buscas, usando o código diretamente dentro do documento R Markdown. Isso funcionou perfeitamente, e a grande vantagem de utilizar essa arquitetura de código diretamente dentro do R Markdown é que o documento fica 100% reproduzível, basta fazer o “knit” (como é conhecida a compilação do documento R Markdown) do documento e os resultados vão ser idênticos em todas as compilações. A grande desvantagem disso é que se o código levar muito tempo para ser executado, o documento também levará muito tempo para ser gerado, pois teremos que rodar todo o código R novamente para gerar o documento. Isso é um problema quando o código demora para rodar e precisamos ver o resultado da edição do documento rapidamente (ex: verificar como ficou disposta uma figura ou se o código Latex que usamos está correto).

¹<https://rmarkdown.rstudio.com/>

²<https://bookdown.org/yihui/rmarkdown/>

Para contornar esse problema, usei uma arquitetura para apenas acessar as variáveis globais da sessão R. As variáveis globais usadas no documento são geradas pelo script fonte, dessa forma podemos separar a parte “pesada” do código, e depois acessá-las pelo documento R Markdown apenas para gerar gráficos, tabela e texto. A vantagem dessa arquitetura é que podemos rodar o código “pesado” apenas uma vez para obtermos os dados necessários para construção do documento, depois editar o documento com boa agilidade de compilação, já que “o grosso” do processamento já está feito. Encontrei duas desvantagens nessa arquitetura, a menor delas é que não podemos compilar o documento via o botão “knit”, apenas por linha de comando, o que na verdade as vezes é mais prático e, até mesmo, preferível. A segunda, e maior, desvantagem é que as variáveis globais são sobrescritas toda vez que rodamos o script fonte, isso impede que rodemos o script duas vezes para duas entradas diferentes. Por exemplo, no meu caso tenho um script que faz a análise e gera gráficos da curva de decaimento \overline{G} em relação a ω e γ para uma dada rede multicamada. Se precisar gerar um documento com a análise de duas ou mais redes, terei que rodar o código novamente toda vez que precisar editar o documento, já que as variáveis globais serão sobrescritas para a última rede de entrada utilizada.

Finalmente, mudei para uma arquitetura onde salvamos as variáveis em um arquivo tipo .RData e depois carregamos este arquivo no documento R Markdown. Esta foi a melhor arquitetura que utilizei neste documento, pois podemos rodar o R Markdown pelo botão “Knit” ou linha de comando, precisamos rodar as análises apenas uma vez para cada entrada, salvando os dados de saída em seus respectivos arquivos .RData e depois carregando os mesmos no documento R Markdown, e, principalmente, o documento é gerado rapidamente pois o carregamento do arquivo .RData é rápido, possibilitando uma edição mais ágil, o que faz muita diferença.

Portanto, o workflow utilizado para gerar esse documento no R Markdown foi:

1. Rodar o script R para gerar as análises necessárias para a primeira rede, no caso morcego-plantas.
2. Salvar as variáveis que pretendo usar no documento R Markdown em uma arquivo .RData (ex: bats-plants.RData). Também é possível salvar a imagem inteira da sessão se optar por praticidade.
3. Rodar novamente o script R para gerar as análises necessárias para a segunda rede, no caso formigas-plantas.
4. Salvar as variáveis que pretendo usar no documento R Markdown em uma arquivo .RData (ex: ants-plants.RData).

5. Proceder com a edição normal do documento R Markdown, lembrando de adicionar uma linha de comando para carregar o arquivo .RData (ex: `load("bats.RData")`) antes das respectivas sessões em que os dados serão usados.

Para este documento gerado em PDF e com referências bibliográficas, foram precisos dois arquivos de texto, um chamado `preamble.tex` para definir pacotes e preferências do Latex, e um arquivo `references.bib`, com as referências no estilo `biblatex`. Estes arquivos devem estar referenciados no YAML (cabeçalho) do documento R markdown.

Algo que senti bastante falta é que o RStudio não faz o “autocomplete” para comandos e códigos Latex. Para escrever intensamente em Latex, sugiro usar um editor Latex para uma maior praticidade e depois colar o texto no documento R Markdown. Ele irá rodar normalmente caso os pacotes e configurações presentes no `preamble.tex` sejam os mesmos que foram usados no editor Latex.

3 Relembrando sobre a modularidade multicamada e entendendo a variável G

3.1 Modularidade multicamada

Os dois parâmetros que são variados nesse relatório são o parâmetro de acoplamento ω e o parâmetro de resolução γ . Estes parâmetros fazem parte da generalização multicamada da modularidade (Mucha et al., 2010):

$$Q^M = \frac{1}{\mu} \sum_{i,j,\alpha,\beta} \left[\left(A_{i\alpha,j\alpha} - \gamma_\alpha \frac{k_{i\alpha} k_{j\alpha}}{2m_\alpha} \right) \delta_{\alpha,\beta} + \omega A_{i\alpha,j\beta} \delta_{ij} \right] \delta(c_{i\alpha}, c_{j\beta}) \quad (1)$$

Onde δ é o delta de Kronecker, que retorna o valor 1 caso $c_{i\alpha} = c_{j\beta}$ e zero caso contrário; a variável m representa a soma de grau de cada nó na camada α ; $A_{i\alpha,j\alpha}$ é a matriz de adjacência na camada α ; $A_{i\alpha,j\beta}$ é a matriz de adjacência entre camadas; $k_{i\alpha}$ representa o grau do nó i na camada α e $\mu = \sum_{i,j,\alpha} A_{i\alpha,j\alpha} + \omega \sum_{i,\alpha,\beta} A_{i\alpha,j\beta}$. Note que se utilizarmos $\omega = 0$ e $\gamma = 1$, a equação de modularidade generalizada (eq. 1) é proporcional a média da modularidade monocamada (Newman, 2004; Newman & Girvan, 2004) de cada camada. Isto também está descrito em Bianconi (2018), pg. 147-148:

$$Q^M = \frac{1}{\mu} \sum_{\alpha} \frac{1}{2m_{\alpha}} \sum_{ij} \left(A_{i\alpha, j\alpha} - \gamma_{\alpha} \frac{k_{i\alpha} k_{j\alpha}}{2m_{\alpha}} \right) \delta(c_{i\alpha}, c_{j\alpha}) \quad (2)$$

O parâmetro ω está contido no intervalo (0,1), onde zero representa o desacoplamento total das camadas, onde cada camada é tratada como uma rede individual (desacoplamento total) e a modularidade final é a média da modularidade de cada camada (eq. 2), e 1 representa o acoplamento máximo (funciona como uma rede monocamada agregada de todas as camadas).

A consequência para os nós da rede é que com $\omega = 0$ temos que todos os nós pertencem a diferentes módulos em diferentes camadas, já que as camadas estão sendo analisadas individualmente. Ao aumentarmos o parâmetro ω , os nós tendem a se tornar parte de apenas um módulo em camadas diferentes, sobrando cada vez menos nós que pertencem a diferentes módulos em diferentes camadas. Finalmente, se atingirmos o valor de $\omega = 1$, nenhum nó estará presente em mais de um módulo, exatamente como ocorre ao utilizarmos o algoritmo Louvain em redes monocamada.

O parâmetro de resolução foi introduzido por Reichardt & Bornholdt (2006) para avaliar redes monocamada e foi propagado para a modularidade generalizada multicamada (Mucha et al., 2010), funcionando da mesma forma. De uma forma geral, se $\gamma_2 > \gamma_1$, os módulos encontrados com γ_2 possuem menos nós (módulos menores) e são mais numerosos (maior quantidade de módulos). Os módulos encontrados com γ_2 podem ser submódulos dos obtidos usando γ_1 , mas nem sempre é o caso (Reichardt & Bornholdt, 2006).

Preciso explorar um pouco mais esse parâmetro de resolução γ para entendê-lo melhor. Entender bem o Reichardt & Bornholdt (2006) será fundamental.

3.2 Variável G

A variável G representa o número de módulos que um determinado nó pertence. Como o algoritmo Louvain não permite que um mesmo nó participe de mais de um módulo dentro de uma mesma camada, G é uma variável discreta inteira que possui o mínimo de 1 e o máximo igual ao número de camadas da rede estudada. G possui valores bem definidos nos extremos, se $\omega = 0 \Rightarrow G = 1$ e se $\omega = 1 \Rightarrow G =$ número total de camadas da rede. Entre estes valores de ω o comportamento de G é variado, mas tende a cair conforme a força de acoplamento ω aumenta. O comportamento do decaimento de G em relação a ω é diferente para cada nó da rede, alguns nós resistem e permanecem em mais de um módulo ao aumentarmos ω , já outros rapidamente passam a pertencer a apenas um módulo ao aumentarmos ω . A figura 1 ilustra o comportamento de diferentes nós ao aumentarmos ω . A figura 2 mostra um exemplo de curva de decaimento de G em relação a ω .

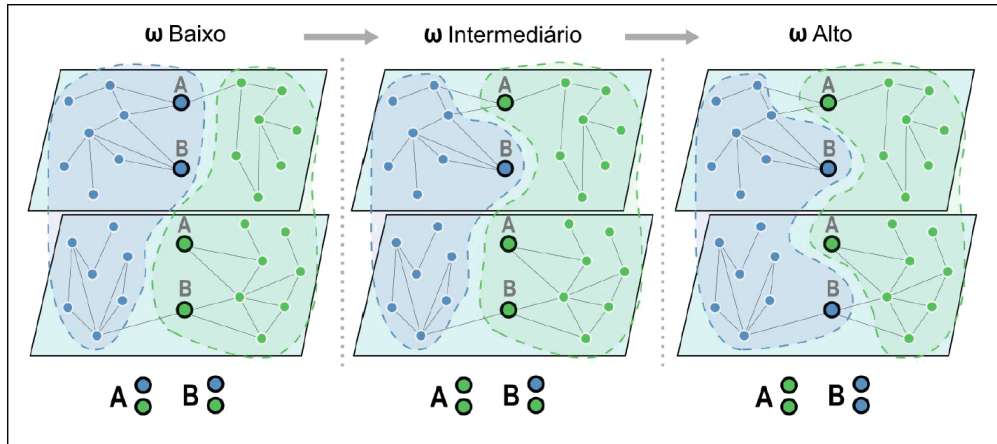


Figura 1: Esquema ilustrativo do que ocorre com alguns nós ao aumentarmos o parâmetro ω . Para valores de ω baixos, vemos que os nós A e B estão em módulos diferentes em camadas diferentes. Ao aumentarmos a força de acoplamento ω , o nó A passa a pertencer ao módulo verde tanto na camada superior como na camada inferior, já o nó B resiste e permanece no módulo azul na camada superior e no módulo verde na camada inferior. Se aumentarmos ainda mais a força de acoplamento ω , em algum momento o nó B acaba cedendo e passa a pertencer a apenas um módulo nas duas camadas.

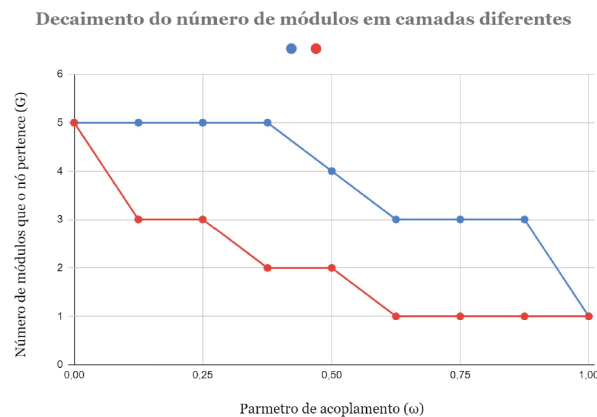


Figura 2: Exemplo hipotético de decaimentos de G (módulos que o nó pertence em diferentes camadas) em relação ao parâmetro ω (constante de acoplamento) esperados. O nó representado pela cor vermelha possui um decaimento rápido de G , portanto, representa uma espécie pouco conectiva do ponto de vista da rede multicamada como um todo. O contrário ocorre com a espécie representada pelo nó azul.

Como o algoritmo Louvain não é determinístico (pode retornar diferentes resultados a cada compilação) e a variável G é inteira e discreta, para termos uma maior confiabilidade precisamos repetir o processo várias vezes para cada valor de ω e extrair uma média de G para cada nó. Assim o valor médio de G é dado por:

$$\overline{G}_\omega = \frac{1}{I} \sum_{i=1}^I G_i \quad (3)$$

Onde I se refere ao número de iterações.

Repetindo isso para diferentes valores de ω , teremos uma curva de decaimento \overline{G} por ω .

Podemos agora repetir esse processo para diferentes valores de γ , assim vamos obter diferentes curvas \overline{G} por ω para cada valor de γ .

3.3 Variável G_{norm}

Para facilitar a classificação dos nós de acordo com sua curva \overline{G} por ω por γ , toda a família de curvas será resumida em uma única variável para cada nó, normalizada pela média. Chamaremos essa variável de G_{norm} , definida na equação 4.

$$G_{norm} = \frac{\sum_{j=1}^{P_2} \sum_{i=1}^{P_1} \overline{G}_{i,j}}{\frac{1}{N} \sum_{k=1}^N \sum_{j=1}^{P_2} \sum_{i=1}^{P_1} \overline{G}_{i,j,k}} \quad (4)$$

Onde P_1, P_2 são o número de partições de ω e γ ; N é o número de nós na rede; i, j são os índices de \overline{G} dentro de ω, γ ; e k é o índice do nó na rede (número de identificação do nó).

4 Visão geral da rede Morcegos-plantas

A rede Morcegos-plantas (Mello et al., 2019) possui 2 camadas (Frugivoria e Nectarivoria), 512 nós e 1248 conexões. A tabela 1 mostra o resumo das propriedades da rede. A figura 3 apresenta uma visão geral da rede.

Tabela 1: Propriedades da rede Morcegos-plantas

Propriedade	Valor
Número de Camadas	2
Tipo de conexões	Frugivoria e Nectarivoria
Número de nós	512
Número de conexões	1248

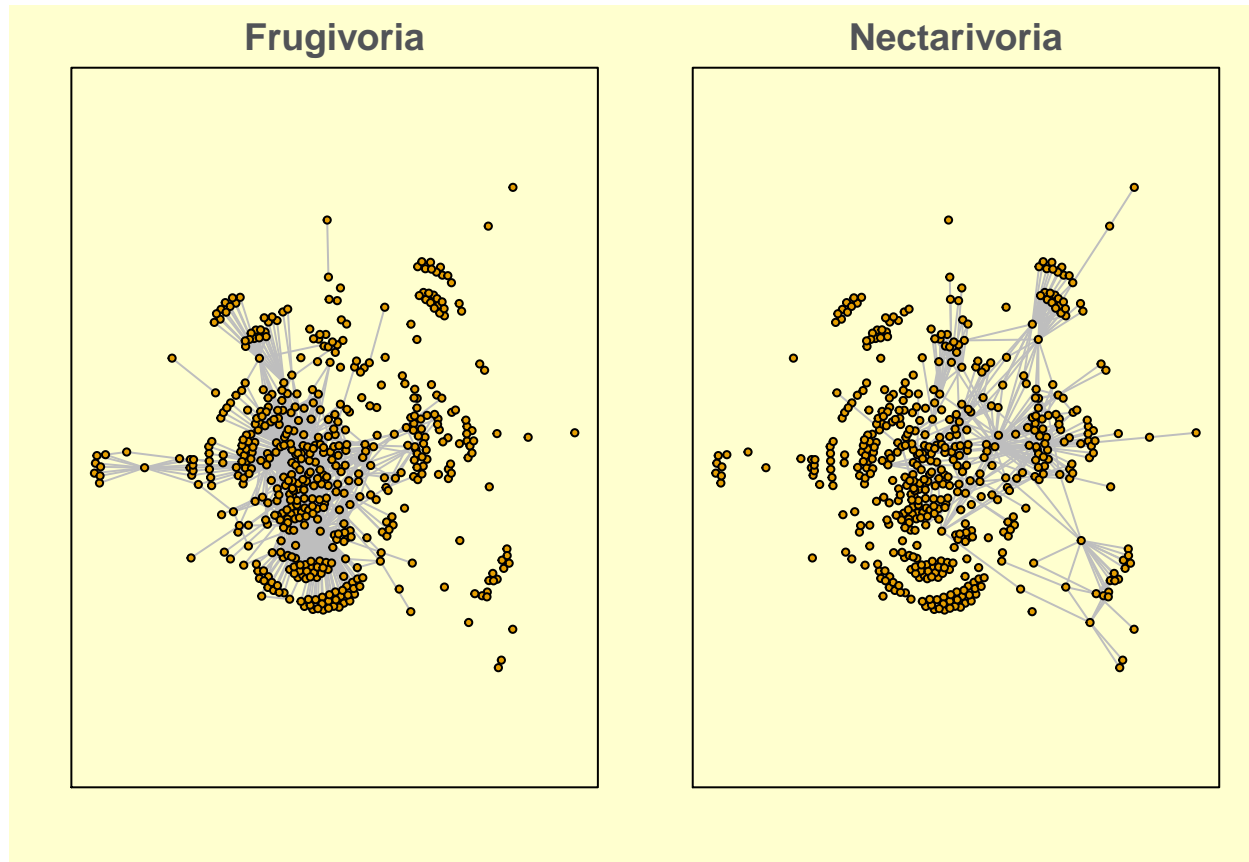


Figura 3: Visão geral da rede Morcegos-plantas .

5 Resultados da rede Morcegos-plantas

5.1 Distribuição de G_{norm}

A variável G foi calculada para 10 partições de ω , ou seja, o tamanho do passo dado dentro de ω foi de 0.1. O processo foi repetido para 16 partições de γ , com γ começando em 0.25, com passos de 0.25 até um γ máximo de 4. O cálculo de \overline{G} foi feito usando 50 iterações. A tabela 2 resume os parâmetros de execução do código e a figura 4 mostra a distribuição dos valores de G_{norm} médio obtidos.

Tabela 2: Parâmetros de execucao

Parâmetro	Valor
Iterações	50
Partições de omega	10

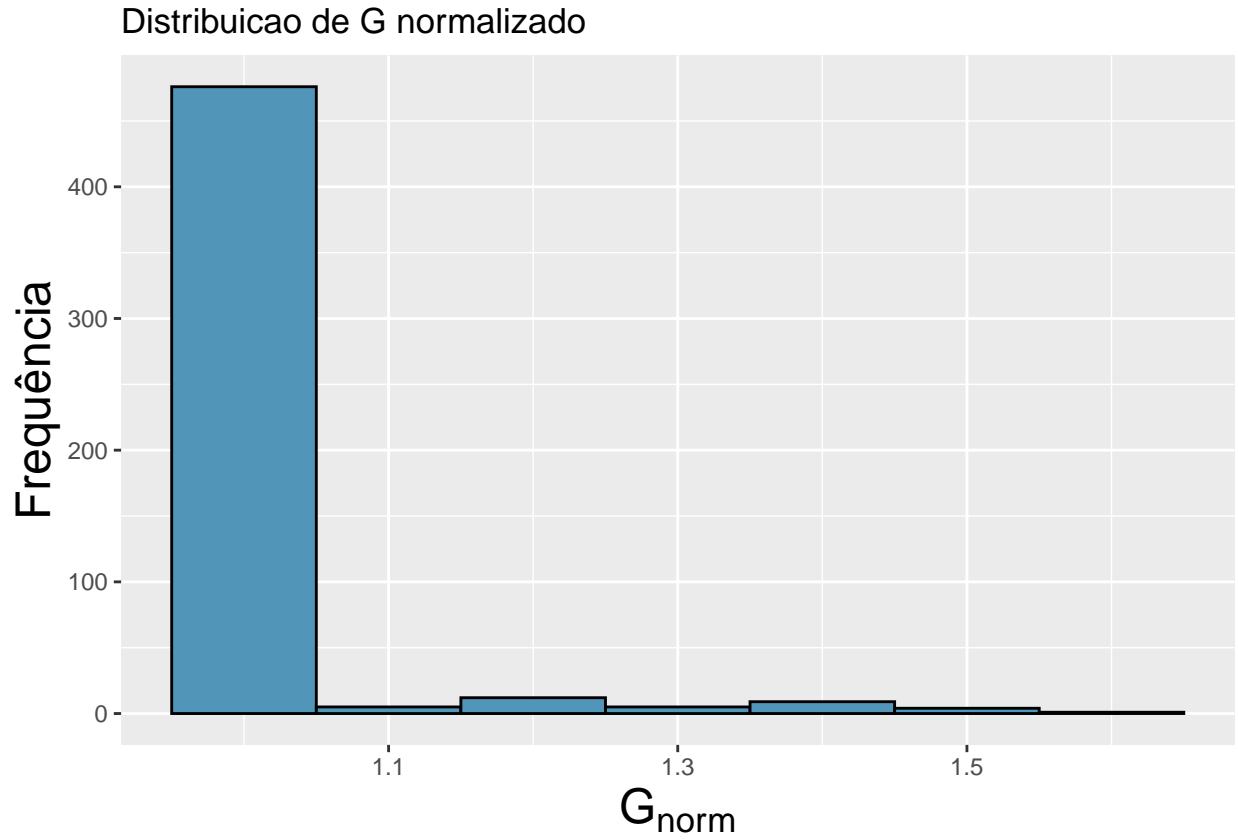


Figura 4: Distribuição de G_{norm} médio da rede Morcegos-plantas .

5.2 Variação de \bar{G} por ω

Como temos dados em 3 dimensões $(\bar{G}, \omega, \gamma)$ temos algumas formas diferentes para apresentar os valores de \bar{G} em relação a ω e γ , não sei dizer se devemos usar uma delas, as três ou alguma outra. A figura 5 mostra curvas de decaimento de \bar{G} por ω para diferentes nós com diferentes valores de G_{norm} e para diferentes valores de γ . A figura 6 mostra a superfície 3D formada por \bar{G} em relação a ω e γ . A figura 7 mostra a mesma superfície da figura 6 mas no formato de mapa de calor.

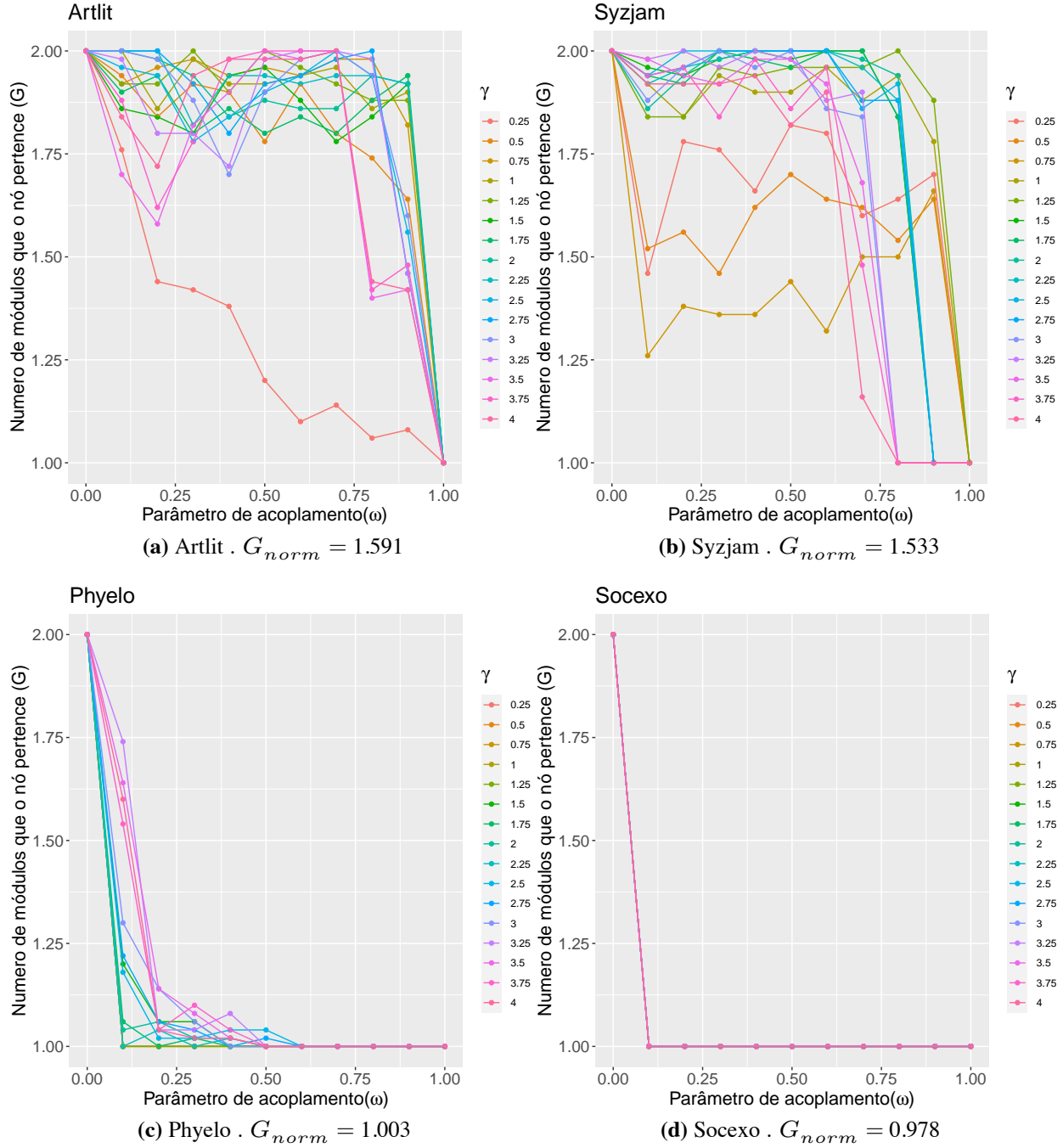


Figura 5: Exemplos de curvas do decaimento de \bar{G} em relação a ω e γ para diferentes valores de γ da rede Morcegos-plantas . (a) Curvas de \bar{G} da espécie com maior valor de G_{norm} da rede. (b) Segundo maior valor de G_{norm} . (c) Valor de G_{norm} mais proximo da média geral da rede. (d) Curvas de \bar{G} referente a uma espécie com valor de G_{norm} abaixo da média da rede.

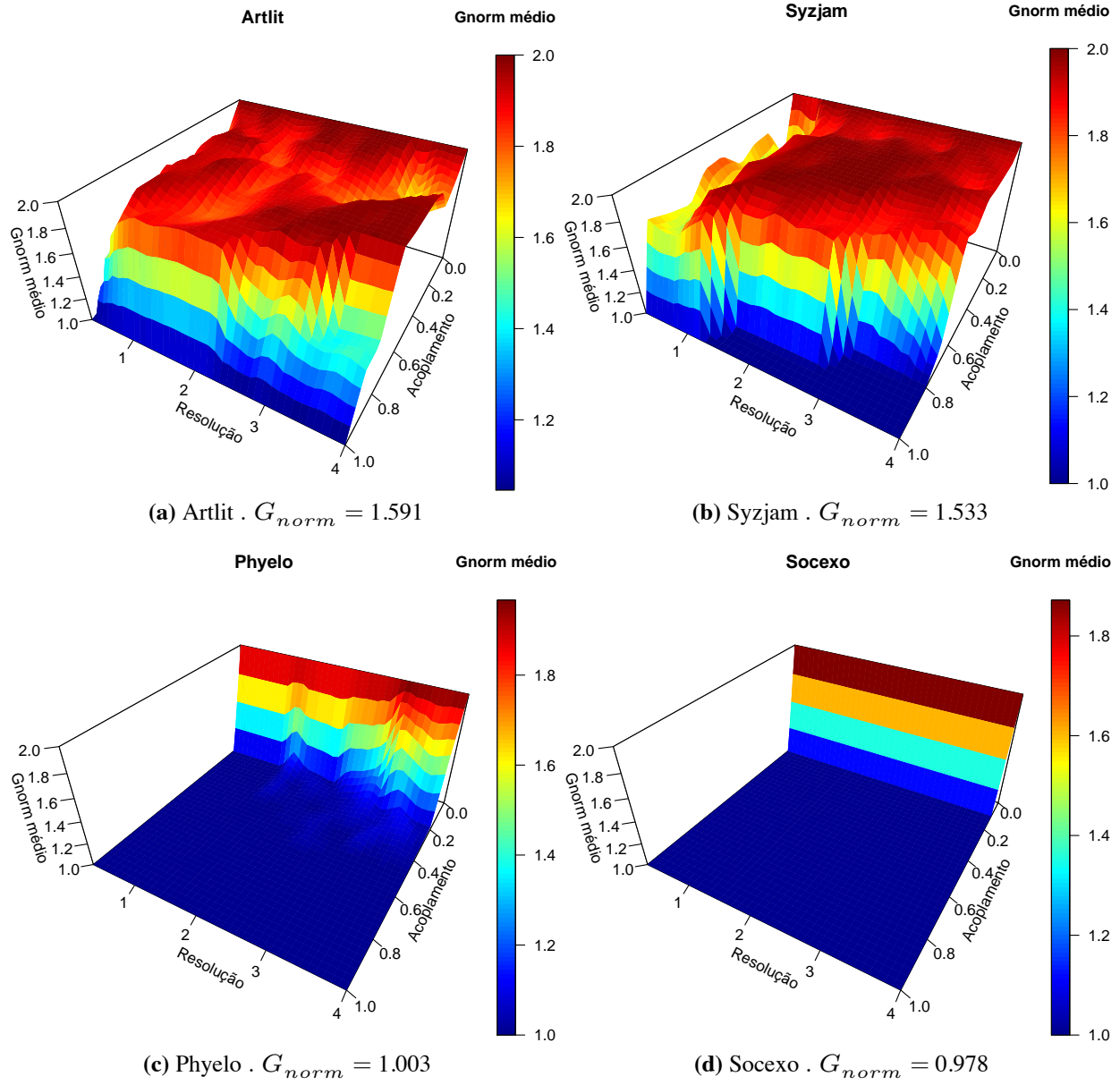


Figura 6: Exemplos de superfícies do decaimento de \bar{G} em relação a ω e γ para diferentes valores de γ da rede Morcegos-plantas . (a) Superfície de \bar{G} da especie com maior valor de G_{norm} da rede. (b) Segundo maior valor de G_{norm} . (c) Valor de G_{norm} mais proximo da média geral da rede. (d) Superfície de \bar{G} referente a uma espécie com valor de G_{norm} abaixo da média da rede.

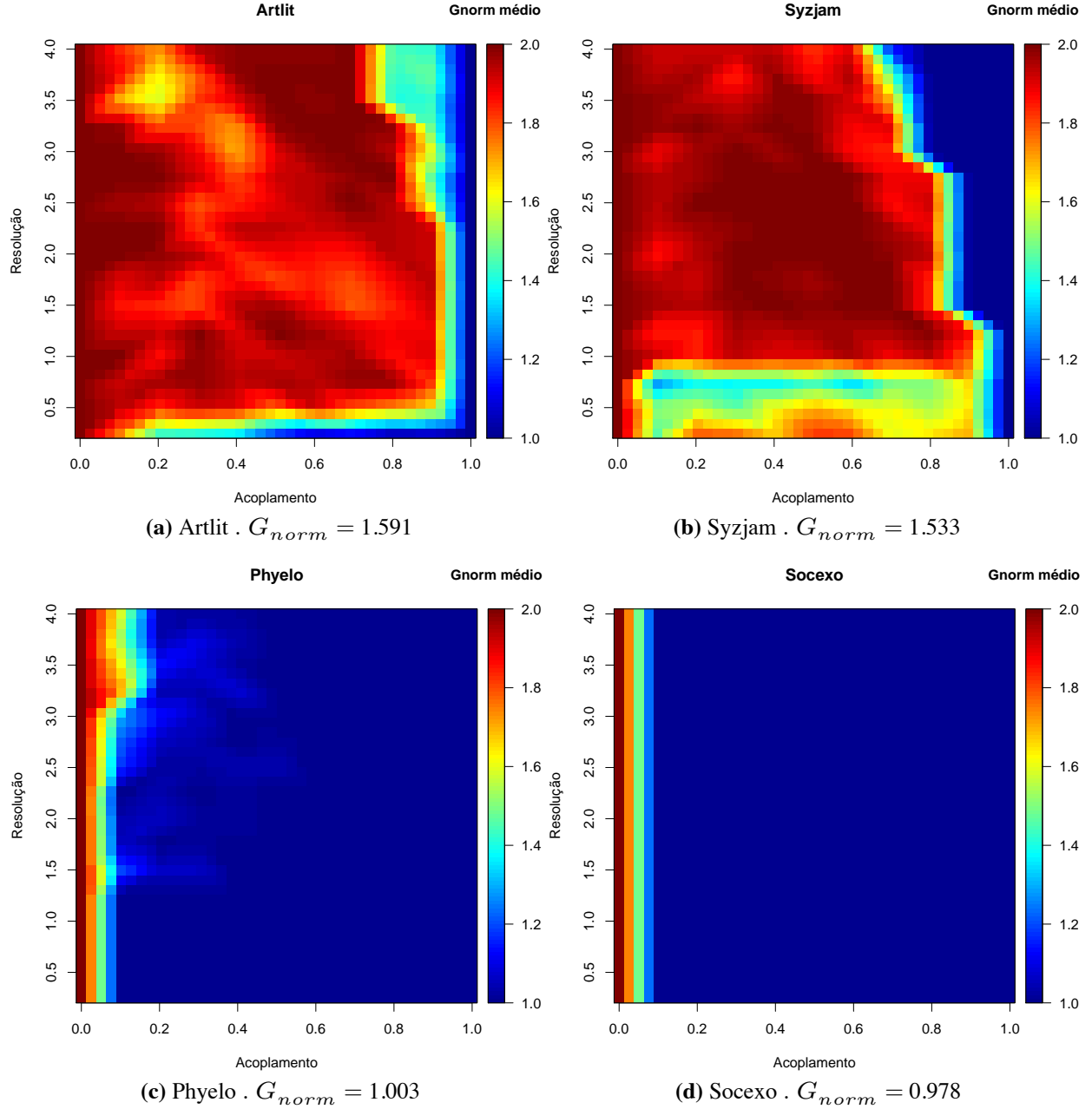


Figura 7: Exemplos de mapas de calor do decaimento de \bar{G} em relação a ω e γ para diferentes valores de γ da rede Morcegos-plantas . (a) Mapa de calor de \bar{G} da especie com maior valor de G_{norm} da rede. (b) Segundo maior valor de G_{norm} . (c) Valor de G_{norm} mais proximo da média geral da rede. (d) Mapa de calor de \bar{G} referente a uma espécie com valor de G_{norm} abaixo da média da rede.

5.3 Seleção das espécies com maior G_{norm} .

A figura 8 e a tabela 3 mostram as espécies com valor de G_{norm} acima de 1.35, ou seja, aquelas com decaimento de G mais lento da rede Morcegos-plantas.

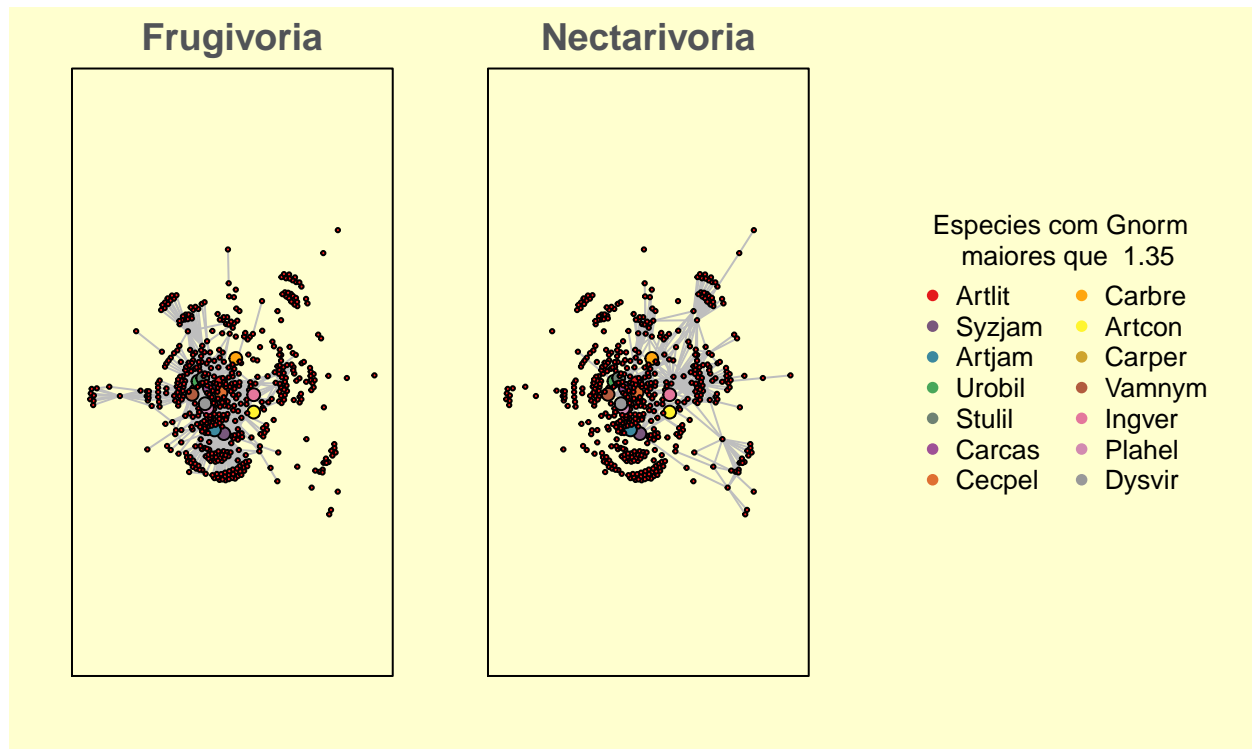


Figura 8: Espécies com G_{norm} maiores que 1.35 em destaque de tamanho e cor.

Tabela 3: Espécies com valores de G_{norm} maiores que 1.35

Espécie	G_{norm}
Artlit	1.591
Syzjam	1.533
Artjam	1.491
Urobil	1.491
Stulil	1.472
Carcas	1.400
Cecpel	1.390
Carbre	1.388
Artcon	1.381
Carper	1.363
Vamnym	1.362
Ingver	1.361
Plahel	1.357
Dysvir	1.351

6 Visão geral da rede Formigas-planta

A rede Formigas-planta (Costa et al., 2016) possui 5 camadas (Flor , Fruta , Visita , EFN e Tropho), 108 nós e 370 conexões. A tabela 4 mostra o resumo das propriedades da rede. A figura 9 apresenta uma visão geral da rede.

Tabela 4: Propriedades da rede Formigas-planta

Propriedade	Valor
Número de Camadas	5
Tipo de conexões	Flor , Fruta , Visita , EFN e Tropho
Número de nós	108
Número de conexões	370

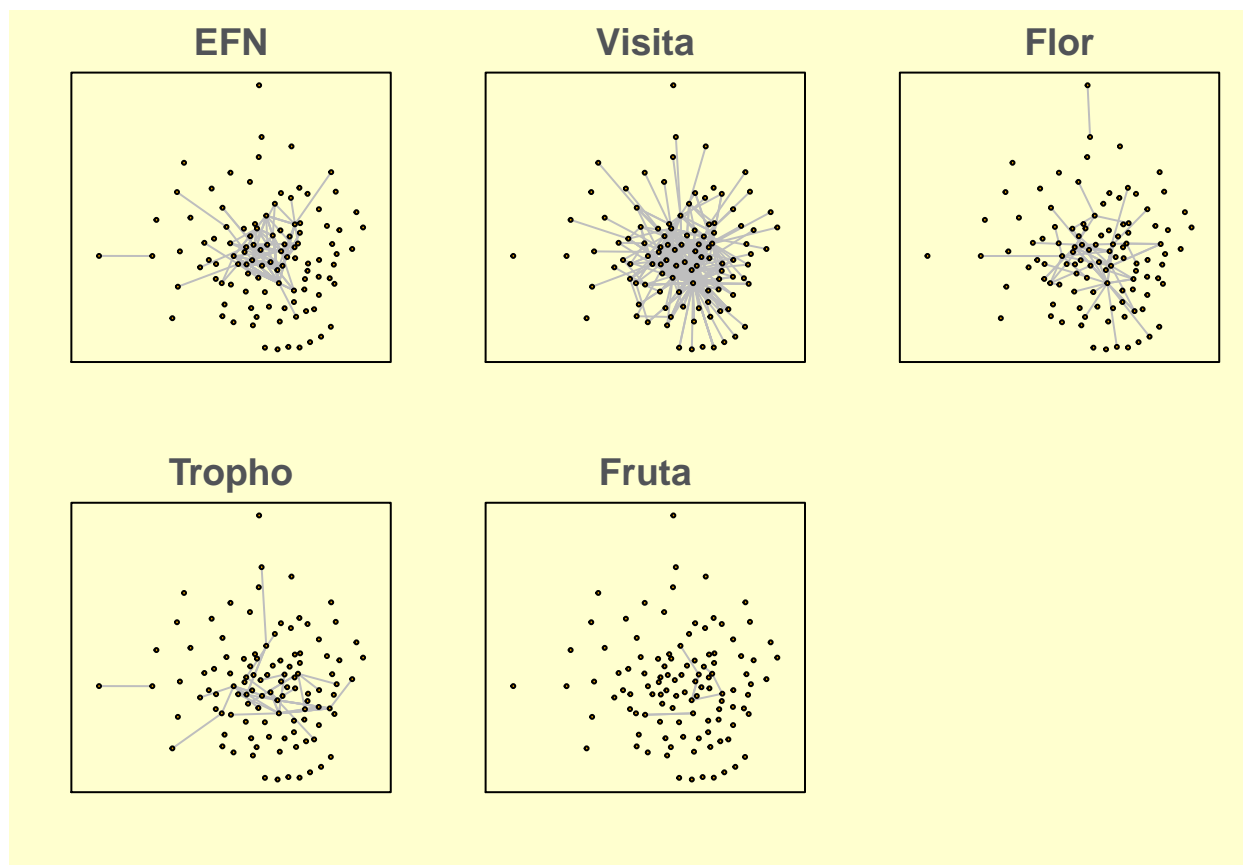


Figura 9: Visão geral da rede Formigas-planta .

7 Resultados da rede Formigas-planta

7.1 Distribuição de G_{norm}

A variável G foi calculada para 10 partições de ω , ou seja, o tamanho do passo dado dentro de ω foi de 0.1. O processo foi repetido para 16 partições de γ , com γ começando em 0.25, com passos de -0.25 até um γ máximo de 4. O cálculo de \bar{G} foi feito usando 100 iterações. A tabela 5 resume os parâmetros de execução do código e a figura 10 mostra a distribuição dos valores de G_{norm} médio obtidos.

Tabela 5: Parâmetros de execucao

Parâmetro	Valor
Iterações	100
Partições de omega	10

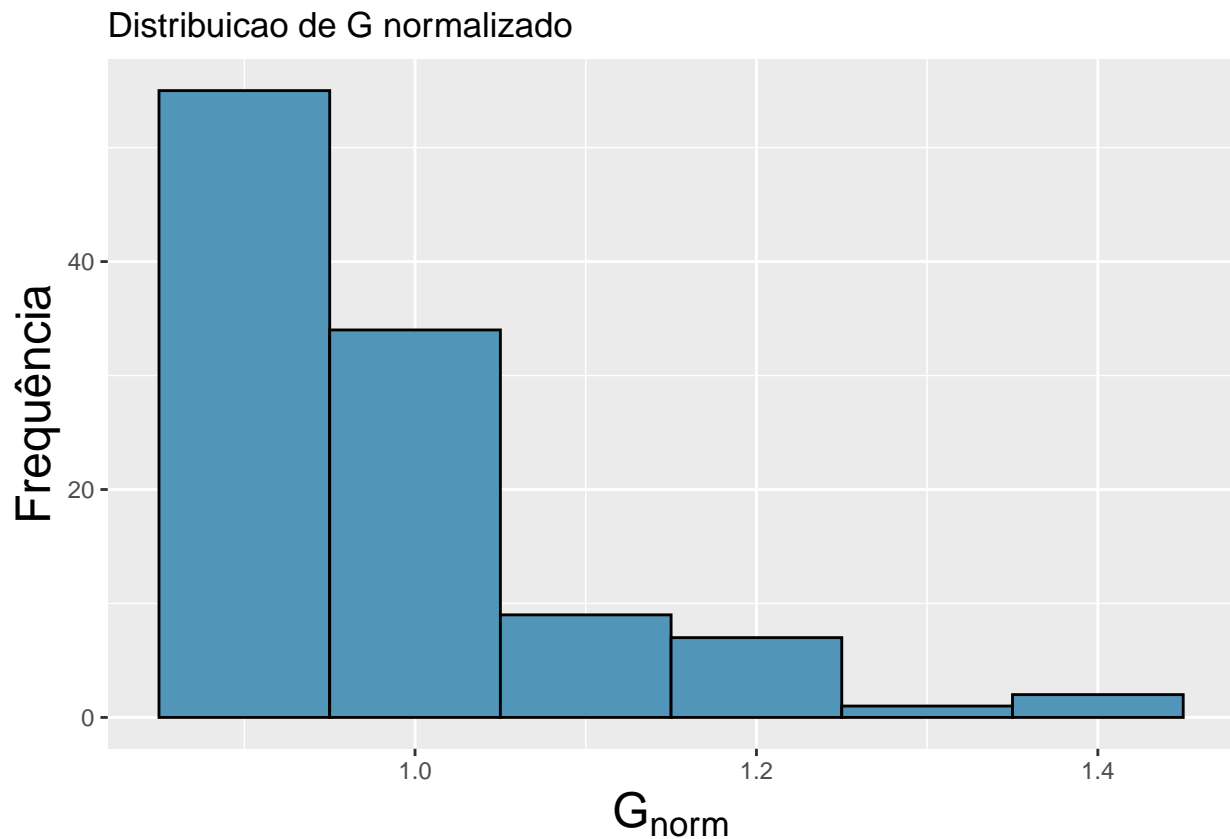


Figura 10: Distribuição de G_{norm} médio da rede Formigas-planta .

7.2 Variação de \overline{G} por ω

Como temos dados em 3 dimensões $(\overline{G}, \omega, \gamma)$ temos algumas formas diferentes para apresentar os valores de \overline{G} em relação a ω e γ , não sei dizer se devemos usar uma delas, as três ou alguma outra. A figura 11 mostra curvas de decaimento de \overline{G} por ω para diferentes nós com diferentes valores de G_{norm} e para diferentes valores de γ . A figura 12 mostra a superfície 3D formada por \overline{G} em relação a ω e γ . A figura 13 mostra a mesma superfície da figura 12 mas no formato de mapa de calor.

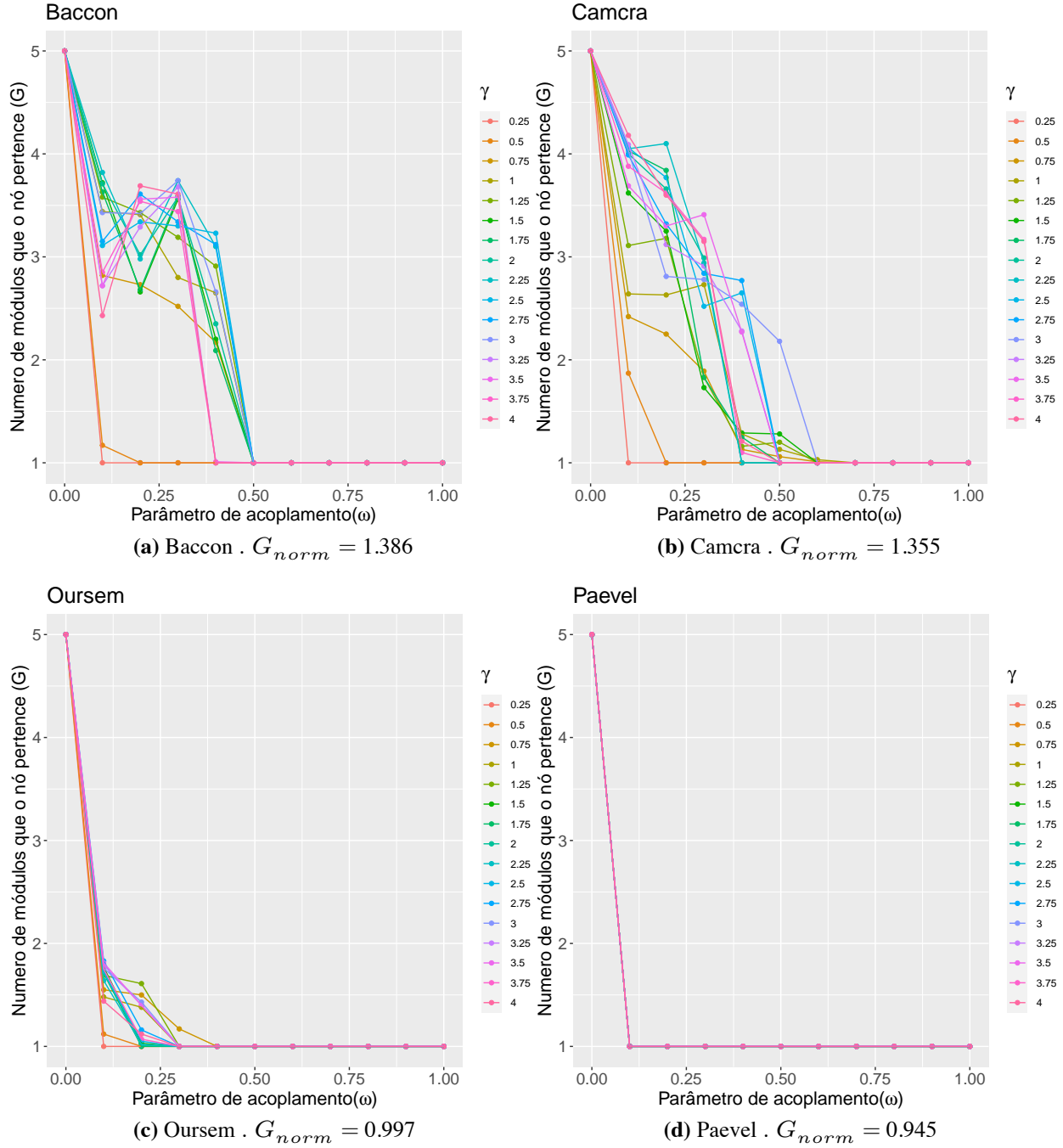


Figura 11: Exemplos de curvas do decaimento de \bar{G} em relação a ω e γ para diferentes valores de γ da rede Formigas-planta . (a) Curvas de \bar{G} da espécie com maior valor de G_{norm} da rede. (b) Segundo maior valor de G_{norm} . (c) Valor de G_{norm} mais proximo da média geral da rede. (d) Curvas de \bar{G} referente a uma espécie com valor de G_{norm} abaixo da média da rede.

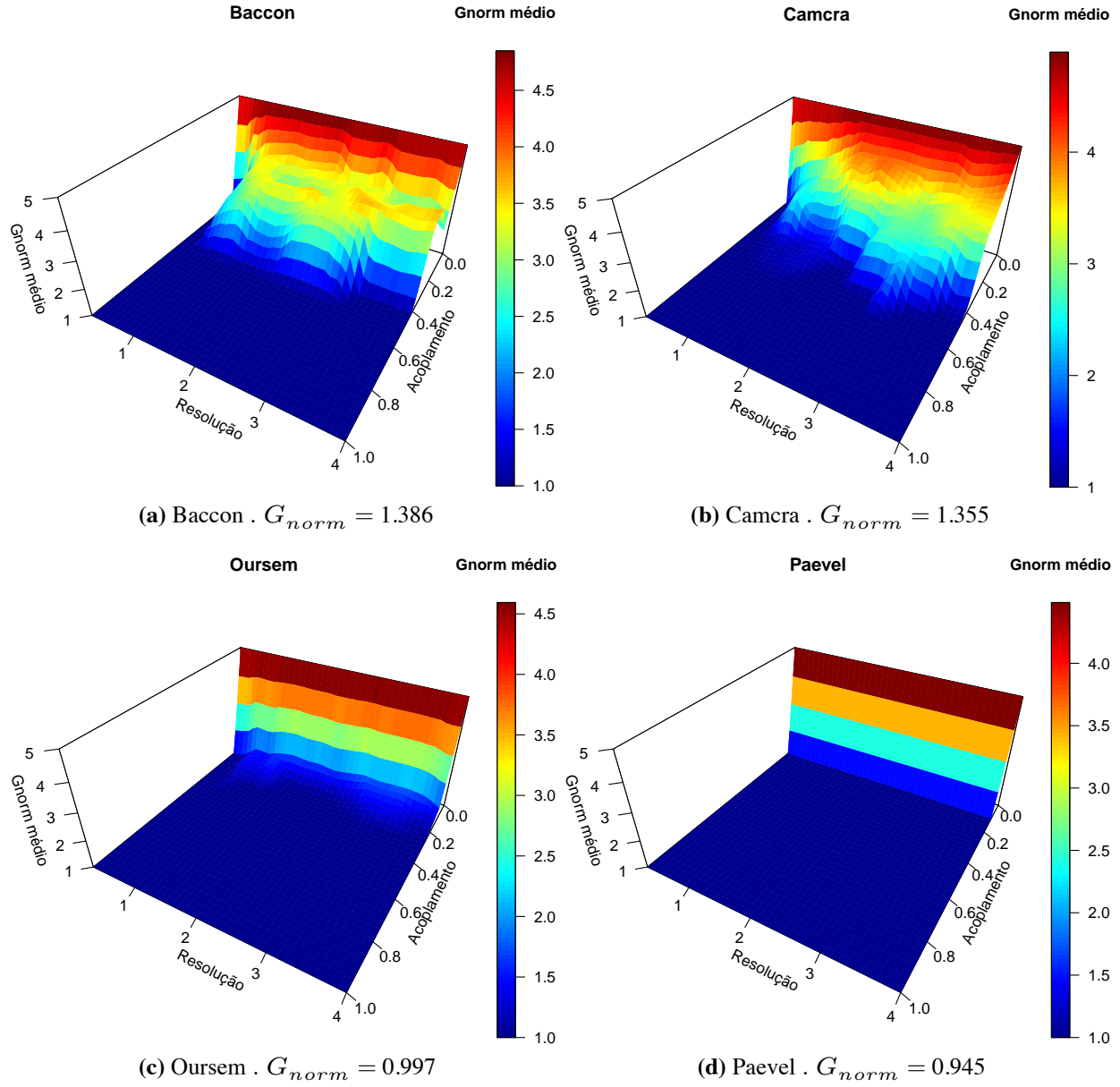


Figura 12: Exemplos de superfícies do decaimento de \bar{G} em relação a ω e γ para diferentes valores de γ da rede Formigas-planta . (a) Superfície de \bar{G} da especie com maior valor de G_{norm} da rede. (b) Segundo maior valor de G_{norm} . (c) Valor de G_{norm} mais proximo da média geral da rede. (d) Superfície de \bar{G} referente a uma espécie com valor de G_{norm} abaixo da média da rede.

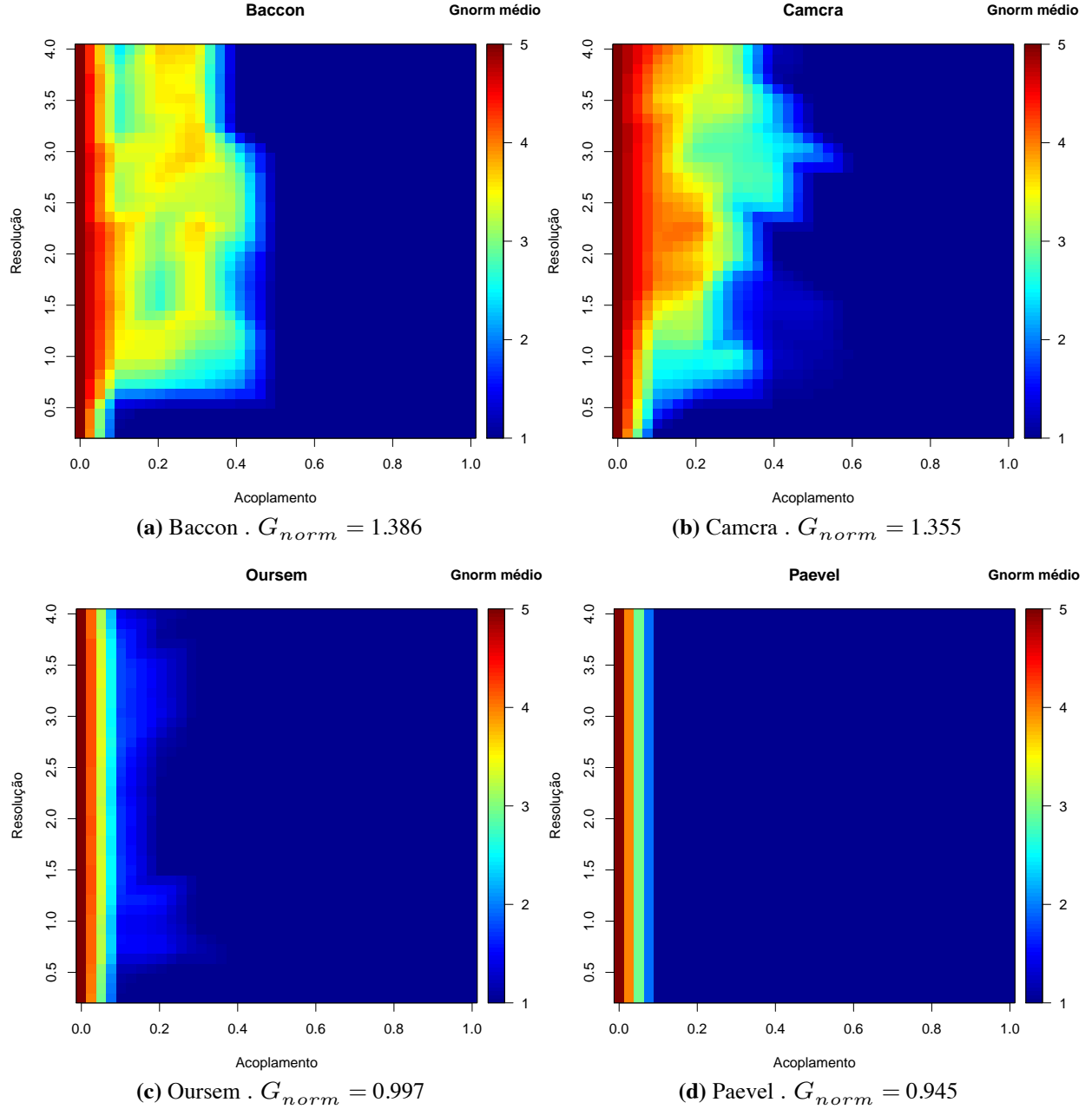


Figura 13: Exemplos de mapas de calor do decaimento de \bar{G} em relação a ω e γ para diferentes valores de γ da rede Formigas-planta . (a) Mapa de calor de \bar{G} da especie com maior valor de G_{norm} da rede. (b) Segundo maior valor de G_{norm} . (c) Valor de G_{norm} mais proximo da média geral da rede. (d) Mapa de calor de \bar{G} referente a uma espécie com valor de G_{norm} abaixo da média da rede.

7.3 Seleção das espécies com maior G_{norm} .

A figura 14 e a tabela 6 mostram as espécies com valor de G_{norm} acima de 1.21, ou seja, aquelas com decaimento de G mais lento da rede Formigas-planta.

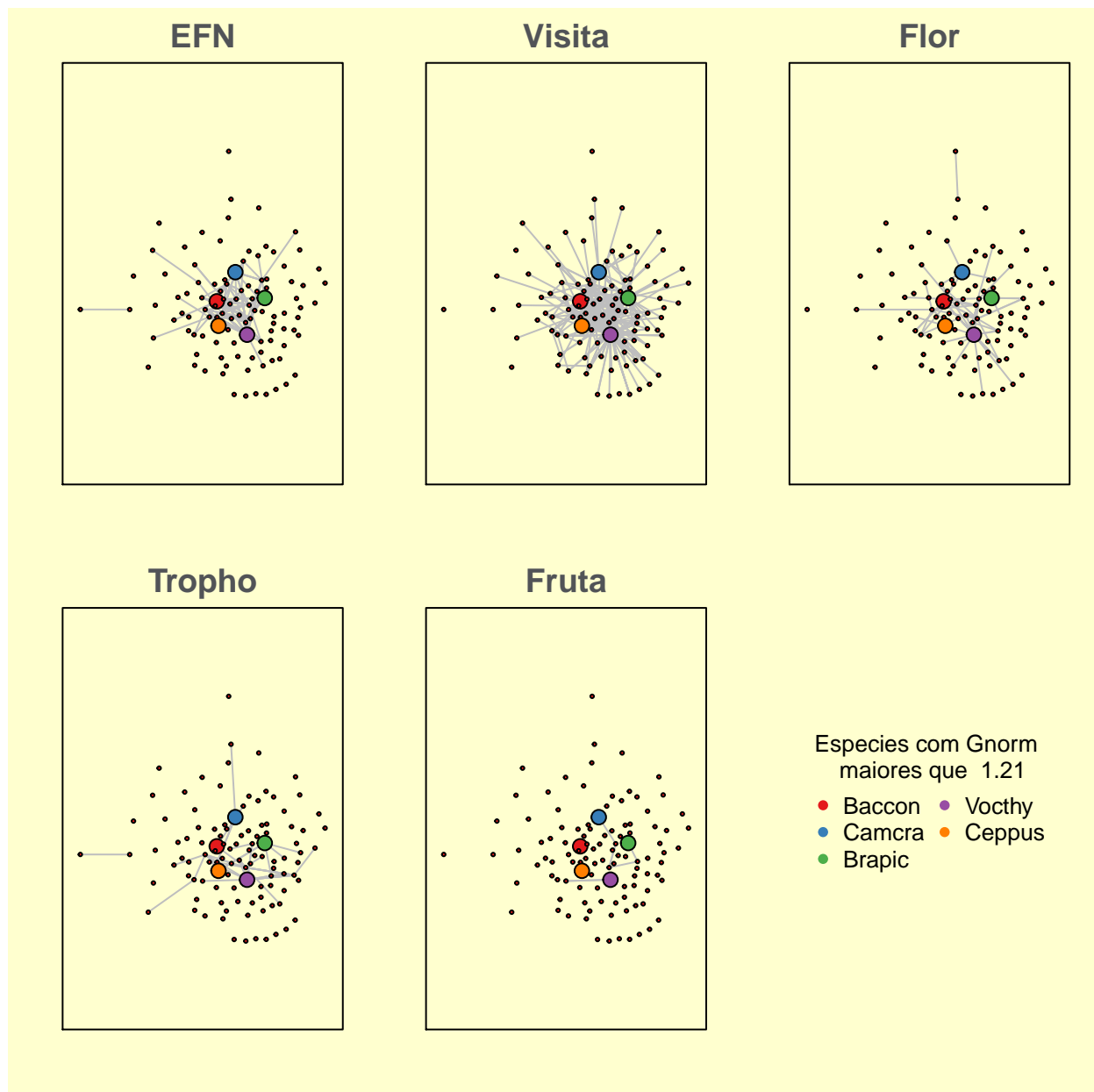


Figura 14: Espécies com G_{norm} maiores que 1.21 em destaque de tamanho e cor.

Tabela 6: Espécies com valores de G_{norm} maiores que 1.21

Espécie	G_{norm}
Baccon	1.386
Camcra	1.355
Brapic	1.341
Vochty	1.235
Ceppus	1.222

8 Discussão

A variação de resolução γ não parece influenciar G de forma tão intensa quanto a variação no acoplamento ω , mas ela existe e com certeza é mais robustos selecionar nós com alto G_{norm} como candidatos a centrais considerando a variação em γ também, pois assim minimizamos a probabilidade de estarmos observando uma exceção.

Devido a essa variação de G_{norm} em relação a γ , podemos tentar identificar espécies que sejam boas conectoras entre camadas dentro de módulos menores (submódulos?) que fazem parte de módulos maiores, sendo importantes para a manutenção deste módulo maior de que fazem parte.

9 Próximas etapas

9.1 Agosto

- Compreender melhor o parâmetro de acoplamento γ com visualização.
- Buscar uma boa forma de visualizar os módulos para rastrear a situação de um nó. Acredito que assim será possível entender melhor como estão posicionados os nós de maior G_{norm} em relação aos módulos ao variarmos ω e γ . Acredito que isso ajudará no insight do que significa na prática essa variável G_{norm} alta que apenas alguns nós possuem.

9.2 Setembro

- Validar ou desvalidar G_{norm} como forma de localizar espécies que são boas conectoras entre camadas usando diversas redes sintéticas (usar simulação do Rafael Pinheiro) para verificar se o comportamento de G_{norm} é consistente, e compará-lo com outras métricas.
- Começar a escrita oficial do TCC.

10 Referências

- Bianconi, G. (2018). *Multilayer networks: Structure and function*. Oxford University Press.
- Costa, F. V., Mello, M. A. R., Bronstein, J. L., Guerra, T. J., Muylaert, R. L., Leite, A. C., & Neves, F. S. (2016). Few ant species play a central role linking different plant resources in a network in rupestrian grasslands. *PLOS ONE*, 11(12), e0167161. <https://doi.org/10.1371/journal.pone.0167161>
- Grolemund, G. (2014). *Introduction to r markdown*. https://rmarkdown.rstudio.com/articles_intro.html
- Mello, M. A. R., Felix, G. M., Pinheiro, R. B. P., Muylaert, R. L., Geiselman, C., Santana, S. E., Tschapka, M., Lotfi, N., Rodrigues, F. A., & Stevens, R. D. (2019). Insights into the assembly rules of a continent-wide multilayer network. *Nature Ecology & Evolution*, 3(11), 1525–1532. <https://doi.org/10.1038/s41559-019-1002-3>
- Mucha, P. J., Richardson, T., Macon, K., Porter, M. A., & Onnela, J.-P. (2010). Community structure in time-dependent, multiscale, and multiplex networks. *Science*, 328(5980), 876–878. <https://doi.org/10.1126/science.1184819>
- Newman, M. E. J. (2004). Analysis of weighted networks. *Physical Review E*, 70(5). <https://doi.org/10.1103/physreve.70.056131>
- Newman, M. E. J., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E*, 69(2). <https://doi.org/10.1103/physreve.69.026113>
- Reichardt, J., & Bornholdt, S. (2006). Statistical mechanics of community detection. *Physical Review E*, 74(1). <https://doi.org/10.1103/physreve.74.016110>
- Xie, Y., Allaire, J. J., & Grolemund, G. (2019). *R markdown the definitive guide*. CRC Press.