

# Avaliando a diversidade genética da bactéria *Vibrio coralliilyticus* através de uma rede particionada

Disciplina de Redes Complexas - PESC - COPPE - UFRJ

*Vinicius W. Salazar, Prof. Daniel R. Figueiredo*

*Outubro de 2019*

## Resumo

## Introdução

Estudos de genômica comparativa têm aplicações em diversas áreas das ciências da vida, como epidemiologia [1], taxonomia [2] e biotecnologia [3], entre outras. Uma estratégia comum empregada nesses estudos é a análise do repertório de genes de uma espécie. Esse conjunto que compreende todos os genes de todas os indivíduos da espécie define o **pangenoma** dessa espécie [4]. O pangenoma de um determinado grupo vai incluir genes que ocorrem em alta frequência, ou seja, são comuns a todos os indivíduos (o genoma “core”, ou persistente), em média frequência (genoma “shell”, ou intermediário) e em baixa frequência (genoma “cloud”, ou acessório), sendo geralmente únicos de um indivíduo ou cepa. Dessa forma, o pangenoma pode ser representado por um diagrama de Venn (Figura 1) (retirada de [5]).

No entanto, um problema fundamental para análises de pangenoma é justamente a determinação de qual desses grupos (“core”, “shell” ou “cloud”) cada gene pertence. Em particular, o genoma “shell” e “cloud” são úteis para entender a adaptação de organismos, logo é importante a sua identificação [6]. Se usarmos uma definição estrita dessas partições, como por exemplo: ‘genoma “core” é composto pelas famílias de genes presentes em >99% dos genomas’, topamos com o efeito de que, a medida que são adicionados genomas, o genoma “core” diminui. De fato, atualmente estudos de genômica bacteriana comumente envolvem centenas a milhares de genomas, logo usar definições estritas das partições torna-se uma limitação na análise do pangenoma. Um outra limitação de estudos em larga escala é a representação do pangenoma: uma vez que espécies tem muitos genes homólogos, é mais conveniente demonstrar a informação sobreposta entre esses genes de uma forma mais compacta, invés de simplesmente concatenar todos os genomas. Diante disso, a representação de pangenomas através de grafos torna-se interessante, pois permite que as homologias e variações entre genomas sejam representadas sem redundância [7].

No presente trabalho, demonstraremos uma aplicação de um novo método de representação de pangenomas. O modelo PPanGGOLiN (\*Partitioned PanGenome Graph of Linked Neighbours) [8] foi publicado recentemente e apresenta uma abordagem promissora para a análise e representação de pangenomas. O modelo representa o pangenoma como um grafo, onde cada vértice é uma família de genes homólogos e cada aresta é uma relação de “contiguidade genética” (ou seja, se as sequências dos genes estão adjacentes na sequência do genoma completo). A abordagem do PPanGGOLiN “preenche a lacuna entre a abordagem pangenômica padrão (que usa um conjunto de famílias de genes independentes e isoladas) e um grafo de pangenoma a nível de sequência”. A vantagem de se usar um grafo a nível de genes, invés de sequências, é a de que isso permite uma representação muito mais compacta em disco, pois os clados são tratados pela presença e ausência (P/A) de genes. Embora isso ignore polimorfismos entre alelos e a presença regiões intergênicas, a abordagem de P/A é adequada para genomas bacterianos, onde o repertório dos genes costuma ser muito mais importante do que polimorfismos e as regiões intergênicas são muito pequenas [8]. Além disso, outra inovação do modelo é a definição de partições usando não somente a frequência de famílias de genes, mas uma combinação dessa informação com o grafo de contiguidade genética para fazer a classificação.

## Resultados

Os 15 genomas utilizados para o estudo (Tabela S1) têm um tamanho média de  $5.67 \pm 0.18$  megabases (média aritmética e desvio padrão) e um número de genes médio de  $5269.34 \pm 208.64$ . Depois de serem anotados com Prokka e processados pelo PPanGGOLiN, obtivemos um grafo do pangenoma com uma componente conexa gigante (GCC) contendo  $\sim 94.24\%$  dos vértices e  $\sim 95.48\%$  das arestas. Nesta rede, o grau médio é  $\langle k \rangle \approx 2.43$  e o grau máximo  $k_{max} = 38$ . Algumas das principais métricas da rede:

- Num. vértices GCC/total: 11613/12323
- Núm. arestas GCC/total: 14304/14981
- Grau médio: 2.43
- Grau máximo: 38
- Diâmetro da GCC: 286
- Média dos caminhos mínimos na GCC: 56.68
- Média de clustering: 0.05

## Discussão

### Métodos

**Conjunto de dados.** Para esse experimento, separamos todos 15 genomas publicamente disponíveis (no mês de Dezembro de 2019) (Tabela S1) da bactéria *Vibrio corallilyticus* (Ben-Haim 2003) [9], uma bactéria que causa doença no coral biogênico *Pocillopora damicornis*. Esse organismo foi escolhido por três motivos: 1) tem relevância científica: sabe-se muito pouco sobre sua relação com o coral hospedeiro [10]; 2) o grupo de genomas tem consistência evolutiva, sendo monofilético e podendo ser agrupado a nível de espécie e 3) o conjunto de 15 genomas é grande o suficiente para obter-se resultados consistentes [8], porém pequeno o suficiente para que seja facilmente rodado em um notebook convencional.

Os genomas foram baixados do banco de dados RefSeq [11] através do script `ncbi-genome-download` (<https://github.com/kblin/ncbi-genome-download>), usando como query o tax ID 190893, corresponde ao organismo no banco de dados NCBI Taxonomy [12]. Esse pacote também permite o download de uma tabela de metadados dos genomas baixados, que foi usada como base para a Tabela S1.

Para ter uma compreensão melhor de cada família de genes que seria identificada no modelo, os arquivos de genoma foram pré-processados com Prokka [13]. Ao realizar a predição de cada gene e sua comparação com um banco de dados de referência, esse passo de anotação permite a descrição do conteúdo de cada gene (por exemplo, que proteína que ele codifica) através do cabeçalho da sequência. Com isso, o arquivo .gff gerado pelo Prokka foi utilizado como input do PPanGGOLiN, invés de os arquivos baixados diretamente do NCBI, que possuíam a sequência de genoma completo mas não a descrição do conteúdo de cada sequência de gene.

**Overview do método PPanGGOLiN.** Esse modelo constrói pangenomas através de um modelo gráfico e um método estatístico para classificar as famílias de genes em três classes: “core”, “cloud” e uma ou mais partições “shell”. Inicialmente, é construído um grafo onde cada vértice é uma família de genes e cada aresta é um relacionamento de contiguidade genética, ou seja, duas famílias são ligadas no grafo se contêm genes que são vizinhos nos genomas. Para identificar partições nesse grafo, é estabelecido um modelo estatístico que considera que genes persistentes compartilham organizações genômicas ao longo dos genomas, e que genes transferidos horizontalmente (principalmente da “shell” e “cloud”) se inserem preferencialmente em algumas regiões cromossômicas. Logo, PPanGGOLiN assume que duas famílias que são vizinhos consistentes no grafo são mais prováveis de pertencerem a mesma partição. Isso é obtido através de um “hidden Markov Random Field” (MRF) cuja rede é dada pelo grafo do pangenoma. Paralelo a isso, o pangenoma é representado como uma matriz P/A onde as linhas correspondem a famílias de genes e as colunas correspondem a genomas. Valores são iguais a 1 se existem pelo menos um membro da família e 0 caso contrário. Essa matriz P/A é modelada por um modelo multivariado misto de Bernoulli (Bernoulli Mixture Model, BMM), cujo os parâmetros são estimados através de um algoritmo de Expectativa-Maximização (EM) que leva em consideração as restrições impostas pelo MRF. Cada família é então associada a uma partição de acordo com

o BMM. Isso resulta no grafo particionado formado por vértices que são classificados como “core”, “shell” ou “cloud”. A força das restrições do MRF aumenta de acordo com um parâmetro  $\beta$  e depende no peso das arestas do grafo inicial do pangenoma. Uma representação gráfica do método é ilustrada na Figura 2 (Ambas essa seção do relatório quanto a figura são adaptadas da referência original, [8]).

**Principais equações do modelo.** PPanGGOLiN visa classificar padrões de P/A de famílias de genes em  $K$  partições ( $K \in \mathbb{N}; K \geq 3$ ). A entrada consiste em uma matriz binária  $X$  onde  $X_{i,j}$  é 1 se uma família  $i$  está presente no genoma  $j$  é 0 caso contrário, onde  $1 \leq i \leq F$  para cada uma das  $F$  famílias e  $1 \leq j \leq N$  para cada um dos genomas  $N$ . Uma primeira abordagem leva em conta o modelo misto de Bernoulli (BMM) estimado através do algoritmo de Expectativa-Maximização. O número de partições  $K$  pode ser maior que 3, devido a possível presença de padrões antagonistas de P/A entre diferentes linhagens de uma espécie. Logo, duas partições vão corresponder ao genoma “core” e “cloud” e  $K - 2$  partições vão corresponder ao genoma “shell”. No BMM, a matriz com os vetores  $X_i = (X_{i,j})_{1 \leq j \leq N}$  descrevendo P/A das famílias, que são assumidas independentes e distribuídas de forma idêntica com um modelo misto dado por:

$$P(X_i = (x_{i,j})_{1 \leq j \leq N}) = \sum_{k=1}^K \pi_k \prod_{j=1}^N \epsilon^{|x_{i,j} - \mu_{kj}|} (1 - \epsilon_{kj})^{1 - |x_{i,j} - \mu_{kj}|}$$

aonde  $\pi = (\pi_1, \dots, \pi_k, \dots, \pi_K)$  representa as proporções de mistura satisfazendo  $\pi_k \in ]0, 1[; (\sum_{k=1}^K \pi_k = 1)$  e  $\pi_k$  é a proporção desconhecida de famílias de genes pertencendo a  $k$ -ésima partição.  $\mu_k = (\mu_{kj})_{1 \leq j \leq N} \in [0; 1]^N$  são os vetores centrais de P/A da  $k$ -ésima partição representando o estado binário mais provável e  $\epsilon_k = (\epsilon_{kj})_{1 \leq j \leq N} \in [0, \frac{1}{2}]^N$  são os vetores de dispersão de  $\mu_k$ . Os parâmetros desse modelo e as partições correspondentes são determinados pelo algoritmo EM.

Para selecionar o  $K$  (número de partições) ótimo, denominado  $\hat{K}$ , o EM faz múltiplas partições aumentando  $K$ . Após os primeiros 10 passos, o índice  $ICL$  (*Integrated Completed Likelihood*) é calculado para cada  $K$ . O  $ICL$  corresponde ao *Bayesian Information Criterion* ( $BIC$ ) penalizado pela entropia média estimada, e é calculado como:

$$ICL(K) = BIC(K) - \sum_{k=1}^K \sum_{i=1}^F p(z_i | X, \hat{\theta}, k) \log(p(z_i | X, \hat{\theta}, k)); \forall p(z_i | X, \hat{\theta}, k) > 0$$

e

$$BIC(K) = \log \mathbb{P}_K(X | \hat{\theta}) - \frac{1}{2\dim(K)} \log F$$

aonde  $\log \mathbb{P}_K(X | \theta)$  é a *log-likelihood* dos dados em um BMM multivariado com  $K$  partições e  $\theta = (\{\pi_k\}_{1 \leq k \leq K}, \{\mu_{kj}\}_{1 \leq k \leq K, 1 \leq j \leq N}, \{\epsilon_{kj}\}_{1 \leq k \leq K, 1 \leq j \leq N})$ , onde  $\hat{\theta}$  é o estimador de máxima verossimilhança (aproximado através do algoritmo EM) e  $\dim(K)$  é a dimensão do espaço de parâmetros para esse modelo. Estimar o  $ICL$  nos permite selecionar o melhor número de partições como  $\hat{K} = \text{argmin}((1 - \delta_{ICL})(ICL(K)))$  aonde  $\delta_{ICL}$  é uma margem suficientemente pequena para evitar obter um  $K$  muito alto que não traria um ganho significativo em relação a um  $K$  menor.

Para o grafo do pangenoma, tomamos a definição formal de um grafo  $G = (V, E)$  tendo um conjunto de vértices  $V = \{(v_i)_{1 \leq i \leq F}\}$  onde  $F$  é o número de famílias de genes no pangenoma associada com um conjunto de vértices  $E = \{e_{i \sim i'}\} = \{(v_i, v_{i'})\}, v_i \in V, v_{i'} \in V$  onde o par de vértices  $(v_i, v_{i'})$  são famílias de genes que têm seus genes  $(v_i, v_{i'})$  adjacentes no genoma  $j$  e cada aresta  $\{e_{i \sim i'}\}$  tem um peso  $w_{i \sim i'}$  proporcional ao número de adjacências desses genes em  $N$  genomas.

Do grafo previamente descrito, a informação de vizinhança das famílias de genes é usada para melhorar os resultados da fragmentação. A abordagem EM descrita acima é estendida ao combinar a matriz  $X$  com o grafo de pangenoma  $G$ . Isso depende do modelo hidden Markov Random Field (MRF) cuja estrutura do grafo é dada por  $G$ . Essa abordagem é denominada NEM (*Neighboring Expectation-Maximization*), e tende a suavizar a partição da matriz P/A agrupando famílias de genes que tem uma maioria ponderada de

vizinhos pertencendo a mesma partição. A variável previamente introduzida  $\{Z_i\}_{1 \leq i \leq F}$  é uma variável latente indicando a qual partição cada família de gene pertence. Essas variáveis aleatórias agora são distribuídas de acordo com um MRF, seguindo a distribuição de Gibbs:

$$\mathbb{P}(\{Z_i\}_{1 \leq i \leq F}) = W_\beta^{-1} \exp\left(\sum_{i=1}^F \sum_{k=1}^K \pi_k 1_{Z_i=k} + \beta \frac{F}{\sum_{i \sim i'} w_{i \sim i'}} \sum_{i \sim i'} w_{i \sim i'} 1_{Z_i=Z_{i'}}\right)$$

onde  $1_A$  é a função indicadora do evento  $A$ , e o segundo somatório trata de cada par  $(i \sim i')$  de cada família de genes vizinha. O parâmetro  $\beta$  corresponde ao coeficiente de regularidade espacial e a função que ele multiplica é o termo de correção que garante que a suavização espacial seja balanceada para o número  $F$  de famílias. O termo  $W_\beta$  é uma constante de normalização, que pode não ser computada, dada o número grande de configurações possíveis. O grau de dependência entre os elementos é dado por  $\beta$ . Agora, os vetores  $(X_i)_{1 \leq i \leq F}$  não são mais independentes. No entanto, condicional aos grupos latentes  $(Z_i)_{1 \leq i \leq F}$ , são independentes e seguem a seguinte distribuição multivariada de Bernoulli:

$$\mathbb{P}(\{X_i\}_{1 \leq i \leq F} \mid \{Z_i\}_{1 \leq i \leq F}) = \prod_{i=1}^F \prod_{j=1}^N \epsilon_{Z_{i,j}}^{|x_{ij}-\mu_{Z_{i,j}}|} (1 - \epsilon_{Z_{i,j}})^{1-|x_{ij}-\mu_{Z_{i,j}}|}$$

Como o número ótimo de partições  $\hat{K}$  não é determinado pelo NEM, é necessário rodar o EM na matriz P/A previamente.

Essa seção foi adaptada de [8] e algumas equações foram omitidas para simplificar.

**Processamento *ad hoc*.** Quaisquer outras etapas de processamento após execução do PPanGGOLiN, como a geração da Tabela 1 ou da Figura 3, foram executadas no ambiente IPython em notebooks Jupyter [14]. Para a geração da distribuição de graus, o grafo resultando do PPanGGOLiN foi importado com o pacote NetworkX [15] e as figuras foram geradas com Matplotlib [16]. A biblioteca Pandas [17] auxiliou no processamento. As figuras da rede foram geradas com Gephi [18]. Para todos esses pacotes, foi usada a distribuição estável mais recente desde Novembro de 2019.

## Referências

- [1] J. L. Gardy and N. J. Loman, “Towards a genomics-informed, real-time, global pathogen surveillance system,” *Nat. Rev. Genet.*, vol. 19, no. 1, pp. 9–20, 2018.
- [2] C. C. Thompson, L. Chimetto, R. A. Edwards, J. Swings, E. Stackebrandt, and F. L. Thompson, “Microbial genomic taxonomy,” *BMC Genomics*, vol. 14, no. 1, p. 913, 2013.
- [3] Z. Sun et al., “Expanding the biotechnology potential of lactobacilli through comparative genomics of 213 strains and associated genera,” *Nat. Commun.*, vol. 6, p. 8322, 2015.
- [4] H. Tettelin et al., “Genome analysis of multiple pathogenic isolates of *Streptococcus agalactiae*: Implications for the microbial ‘pan-genome,’” *Proc. Natl. Acad. Sci.*, vol. 102, no. 45, p. 16530, 2005.
- [5] C. G. P. McCarthy and D. A. Fitzpatrick, “Pan-genome analyses of model fungal species,” *Microb. Genomics*, vol. 5, no. 2, 2019.
- [6] O. Lukjancenko, T. M. Wassenaar, and D. W. Ussery, “Comparison of 61 Sequenced *Escherichia coli* Genomes,” *Microbial Ecology*. 2010.
- [7] T. Marschall et al., “Computational pan-genomics: Status, promises and challenges,” *Brief. Bioinform.*, 2018.
- [8] G. Gautreau et al., “PPanGGOLiN: Depicting microbial diversity via a Partitioned Pangenome Graph,” *bioRxiv*, p. 836239, 2019.

- [9] Y. Ben-Haim et al., “*Vibrio coralliilyticus* sp. nov., a temperature-dependent pathogen of the coral *Pocillopora damicornis*,” *Int. J. Syst. Evol. Microbiol.*, vol. 53, no. 1, pp. 309–315, 2003.
- [10] J. Vidal-Dupiol, O. Ladrière, A. L. Meistertzheim, L. Fouré, M. Adjero, and G. Mitta, “Physiological responses of the scleractinian coral *Pocillopora damicornis* to bacterial stress from *Vibrio coralliilyticus*,” *J. Exp. Biol.*, vol. 214, no. 9, pp. 1533–1545, 2011.
- [11] N. A. O’Leary et al., “Reference sequence (RefSeq) database at NCBI: Current status, taxonomic expansion, and functional annotation,” *Nucleic Acids Res.*, 2016.
- [12] S. Federhen, “The NCBI Taxonomy database,” *Nucleic Acids Res.*, 2012.
- [13] T. Seemann, “Prokka: rapid prokaryotic genome annotation,” *Bioinformatics*, vol. 30, no. 14, pp. 2068–2069, 2014.
- [14] M. Ragan-Kelley et al., “The Jupyter/IPython architecture: a unified view of computational research, from interactive exploration to communication and publication,” in *AGU Fall Meeting Abstracts*, 2014.
- [15] A. Hagberg, P. Swart, and D. S Chult, “Exploring network structure, dynamics, and function using NetworkX,” 2008.
- [16] J. D. Hunter, “Matplotlib: A 2D graphics environment,” *Comput. Sci. Eng.*, 2007.
- [17] W. McKinney, “pandas: a foundational Python library for data analysis and statistics,” *Python High Perform. Sci. Comput.*, vol. 14, 2011.
- [18] M. Bastian, S. Heymann, and M. Jacomy, “Gephi: an open source software for exploring and manipulating networks,” in *Third international AAAI conference on weblogs and social media*, 2009.

**Tabela 1:** nome dos organismos, número de acesso, tamanho (em megabases), conteúdo GC, número de genes e número de proteínas.

name_and_strain	assembly_accession	Size..Mb.	GC.	Genes	Proteins
<i>Vibrio coralliilyticus</i> _ATCC_BAA-450	GCF_000176135.1	5.68063	45.7000	5250	5035
<i>Vibrio coralliilyticus</i> _P1	GCF_000195475.1	5.51326	45.7000	5207	4960
<i>Vibrio coralliilyticus</i> _OCN008	GCF_000461895.1	5.53490	45.7000	5250	4466
<i>Vibrio coralliilyticus</i> _OCN014	GCF_000763535.1	5.73279	45.8007	5292	4988
<i>Vibrio coralliilyticus</i> _RE98	GCF_000772065.1	6.03782	45.5020	5742	5533
<i>Vibrio coralliilyticus</i> _S2052	GCF_000967465.1	5.43392	45.7000	5001	4889
<i>Vibrio coralliilyticus</i> _S2043	GCF_000967485.1	5.43504	45.7000	5000	4890
<i>Vibrio coralliilyticus</i> _RE22A	GCF_001297935.1	5.68477	45.7510	5203	5034
<i>Vibrio coralliilyticus</i> _58	GCF_001693615.1	5.49001	45.5782	5042	4753
<i>Vibrio coralliilyticus</i> _SNUTY-1	GCF_002073995.1	5.84268	45.6306	5525	4316
<i>Vibrio coralliilyticus</i> _080116A	GCF_002286405.1	5.63628	45.7000	5224	5047
<i>Vibrio coralliilyticus</i> _RE87	GCF_002286655.1	5.58929	45.8000	5149	5014
<i>Vibrio coralliilyticus</i> _AIC-7	GCF_002287625.1	5.95294	45.3000	5582	5435
<i>Vibrio coralliilyticus</i> _NA0301	GCF_002742585.1	5.68896	45.7000	5286	5111
<i>Vibrio coralliilyticus</i> _RE22B	GCF_003391375.1	5.78497	45.8078	5287	5064

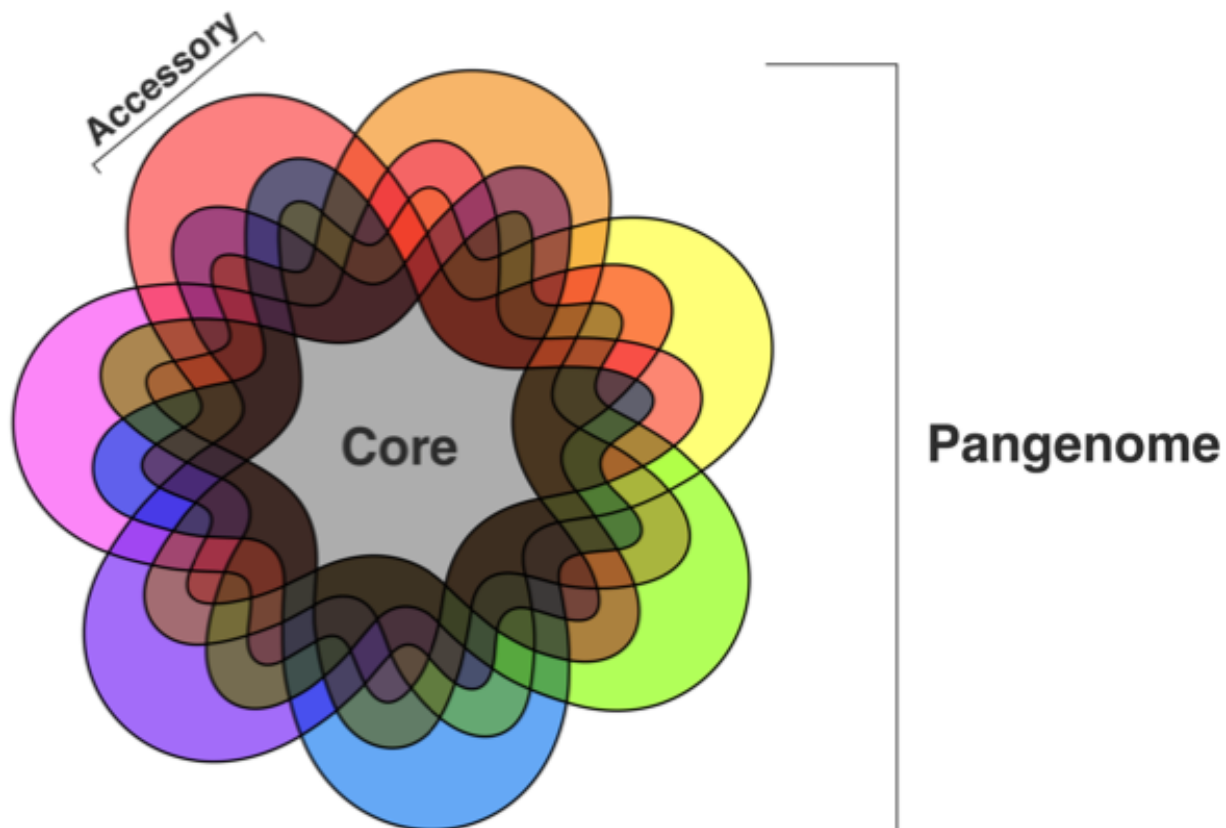


Figure 1: Entendendo o pangenoma como um diagrama de Venn: para um determinado grupo, seu pangenoma compreende os genes comuns a grande maioria dos indivíduos (genoma core), genes únicos de cada indivíduo (genoma acessório), e genes em frequências intermediárias.

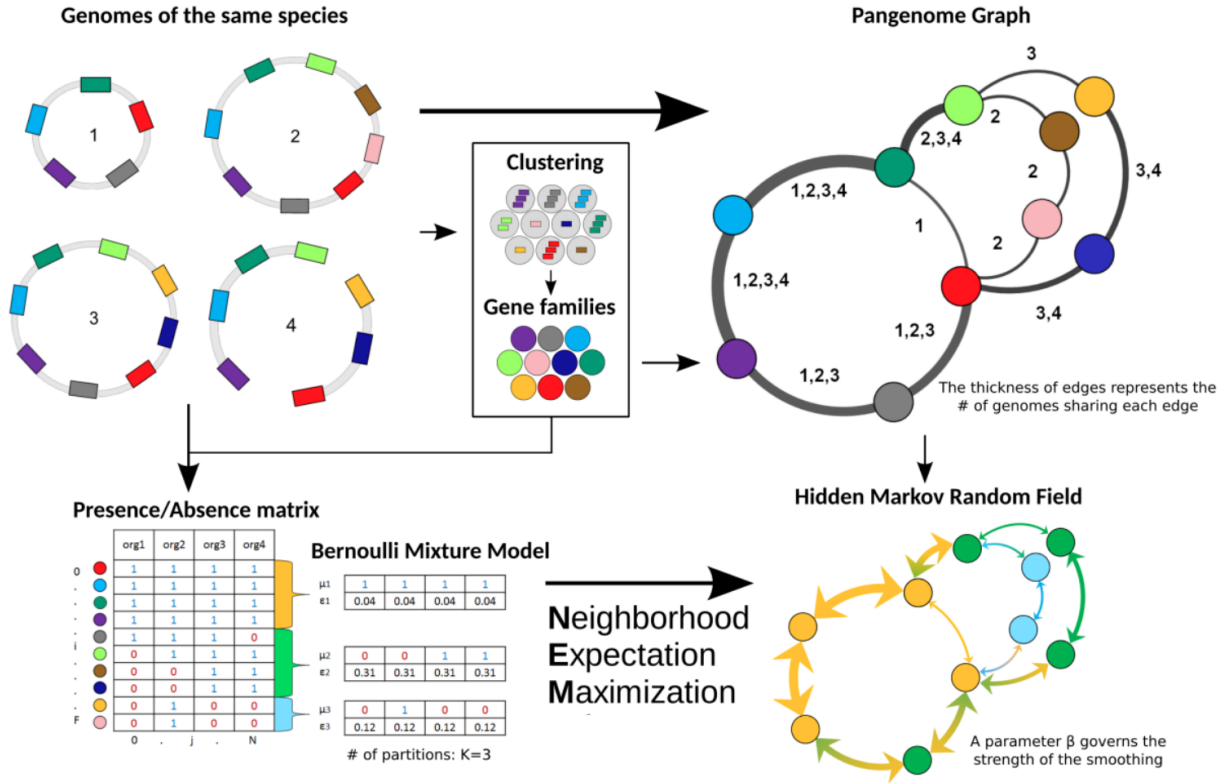


Figure 2: Representação gráfica do modelo PPanGGOLiN em 4 genomas: Esse método requer genomas anotados da mesma espécie. O grafo de pangenoma é construído combinando genes homólogos com sua vizinhança genômica. Paralelamente é construída uma matriz P/A de famílias de genes x genomas. Esse pangenoma é dividido em  $K$  partições (nesse exemplo,  $K = 3$ ) ao estimar-se os melhores parâmetros através do algoritmo EM. Esse método envolve a maximização da verossimilhança de um modelo multivariado BMM que é suavizado pelo espalhamento das partições ao longo do grafo que usa o MRF, penalizando famílias classificadas inadequadamente de acordo com o grafo. Esse processo é repetido até o um trade-off maximizando a verossimilhança geral. O resultado é um grafo particionado do pangenoma, classificando as famílias nas partições. Adaptado de [8]