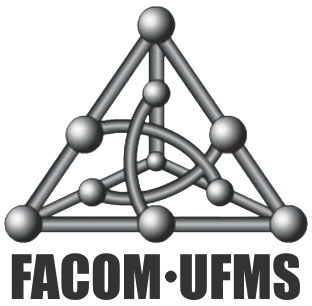
Classificação de sequências metagenômicas

Dissertação de Mestrado

Habib Asseiss Neto

Orientação: Prof. Dr. Nalvo Franco de Almeida Jr.

Área de concentração: Biologia Computacional



Faculdade de Computação
Universidade Federal de Mato Grosso do Sul

Campo Grande, Setembro de 2012

Agradecimentos

À minha família, por estar sempre presente, me apoiando em todas as minhas decisões.

Agradeço ao meu orientador, Professor Nalvo, pela dedicação e pela paciência durante o desenvolvimento deste trabalho e pela preocupação em oferecer oportunidades para minha permanência em Campo Grande. Agradeço também ao Professor Dr. João Carlos Setubal, pelas valiosas dicas em sua "aula" depois do meu exame de qualificação.

Aos meus amigos de Guaraçaí, com os quais contei nas necessárias horas de descanso e aos amigos de Campo Grande, com os quais compartilhei incontáveis horas de estudo e momentos de raiva e de alegria.

Agradeço ainda aos professores da banca examinadora Luciana Montera Cheung, Maria Emília Walter Telles e Said Sadique Adi por suas sugestões e críticas, que contribuíram para a melhoria deste trabalho.

Por fim, agradeço à Capes e à Fundect pelo apoio financeiro durante meu mestrado.

Resumo

Este trabalho discute alguns métodos para o problema de classificação de sequências metagenômicas, e apresenta um método alternativo baseado em comparação de sequências, montagem de fragmentos e filogenia, cuja principal característica é a de agrupar apenas sequências que compartilham similaridade com proteínas conhecidas. O método proposto foi testado com um conjunto de dados metagenômicos simulados e apresentou resultados satisfatórios, quando comparado com outros. Além de se mostrar útil como método direto de classificação, pode também ser utilizado como auxiliar, corrigindo classificações erradas feitas por outras metodologias.

Palavras-chave: bioinformática; metagenômica; classificação.

Abstract

This work presents a discussion of some methods to the problem of classifying metagenomic sequences, and presents an alternative method based on sequence comparison, fragment assembly and phylogeny, whose the main feature is the grouping of only sequences that share similarity to known proteins. The method has been tested on a simulated set of metagenomic sequences and has shown satisfying results when compared to others. Besides proven useful as a direct method of classification, it can also be used as auxiliary one, correcting erroneous classifications made by other methodologies.

Sumário

Capítulo 1

Introdução

Este trabalho está inserido na área de Bioinformática, mais precisamente em técnicas para o estudo de organismos presentes em comunidades microbianas, através do sequenciamento de material genético obtido em amostras dessas comunidades, que costumam ser compostas, dentre outros micro-organismos, de bactérias, arqueias e vírus.

Novas tecnologias de sequenciamento e a drástica redução de seus custos têm permitido a obtenção de informações biológicas diretamente de comunidades microbianas em seus habitats naturais. Agora, ao invés de observar espécies individualmente, é possível estudar dezenas de milhares de espécies, todas juntas. Dados de sequências obtidos diretamente do sequenciamento ambiental são chamados de **metagenoma**, e o estudo desses dados é chamado de **metagenômica** [24].

A metagenômica é uma derivação da genômica microbiana original, com a diferença principal de que na primeira não precisamos obter culturas puras para realizar o sequenciamento. Ela inclui uma grande variedade de técnicas e abordagens, podendo ajudar a estabelecer hipóteses a respeito de interações entre membros de comunidades e de interações com seus ambientes. Assim, a metagenômica está sendo considerada a tecnologia base para a compreensão da evolução de organismos e ecossistemas microbianos [24]. Em última instância, a metagenômica é capaz de oferecer maneiras de melhorar as estratégias de análise de um dado ecossistema e aumentar a compreensão sobre como as diversas espécies de uma comunidade microbiana se relacionam.

Muitos projetos metagenômicos são conhecidos, como o que examinou comunidades microbianas da água do oceano, realizado a partir da expedição *Sorcerer II Global Ocean Sampling* (GOS) [51]. Outros estudos metagenômicos importantes envolveram a análise do ecossistema microbiano do rúmen bovino [5] e o microbioma do intestino humano [44], em ambos os casos revelando um ecossistema complexo, composto principalmente de bactérias não culturadas.

Basicamente um projeto de metagenômica possui as seguintes etapas. O processo inicia com a amostragem e coleção de dados, seguidos pela extração de DNA, sequenciamento, processamento das sequências, também chamados de *reads*, e montagem. Os reads ou contigs participam do processo de predição de genes e anotação e, por fim, um passo de classificação é aplicado [24].

A classificação consiste na atribuição de uma espécie, ou de um grupo taxonômico de mais alta ordem, a cada read da amostra, na tentativa de caracterizar o ecossistema em termos de sua composição e consequentemente em termos das interações entre os organismos presentes.

Vários métodos de classificação são conhecidos na literatura, e são divididos em dois grupos, de acordo com a abordagem adotada: métodos baseados em similaridade de sequência e métodos baseados em composição de sequência. Na primeira categoria, a classificação é feita com base na similaridade de sequências e no uso de programas de comparação, como o Blast [2], para classificar um read de acordo com *hits* encontrado em bases de dados de proteínas ou domínios. Na segunda categoria, os métodos contam com o fato de que diferentes genomas podem possuir diferentes assinaturas em sua composição, tais como conteúdo GC, frequência de oligonucleotídeos, a presença ou a ausência de genes marcadores específicos, tais como o RNA ribossomal 16S. A montagem de reads em contigs para posterior comparação também costuma ser utilizada [24].

No entanto, na metagenômica, a montagem completa não é possível, pois além dos problemas da genômica clássica, como orientação desconhecida, regiões repetidas e falta de cobertura, a amostra é geralmente incompleta e praticamente todas as espécies são amostradas parcialmente. Assim, é difícil mapear reads isolados a suas espécies de origem. Além de fragmentadas e incompletas, as sequências obtidas de uma amostra ambiental formam um volume bem maior do que o obtido de um único organismo [50]. Assim, o problema abordado neste texto é o de classificar sequências

metagenômicas de acordo com suas espécies.

O objetivo deste trabalho foi o de discutir e comparar alguns métodos de classificação de sequências metagenômicas, além de desenvolver uma metodologia alternativa para o problema. O método aqui proposto utiliza abordagens comuns, como comparação de sequências, montagem e filogenia, mas tem como ideia principal a tentativa de evitar que a montagem descrita no parágrafo anterior seja feita para todos os reads simultaneamente, fazendo com que a tarefa seja feita apenas para reads que compartilham similaridade com uma mesma proteína.

Além disso, o resultado da nossa classificação não se resume a uma única espécie, e sim a um grupo de espécies filogeneticamente próximas. Isso é vantajoso uma vez que, independentemente da abordagem aplicada, os métodos tendem a atribuir a cada sequência uma única espécie cujos genes já estão anotados e depositados em bancos de dados de sequências biológicas. Essa atribuição a uma única espécie pode não ser confiável, pois em se tratando da descoberta de novas espécies em nichos ecológicos ainda não explorados, elas certamente não terão seus genomas sequenciados e, muito menos, anotados.

Nosso método foi testado usando dados metagenômicos simulados e apresentou resultados satisfatórios, quando comparados com outros métodos. Resultados preliminares deste trabalho foram publicados em [1].

Este texto está estruturado como segue. No Capítulo 2 uma breve introdução à metagenômica e ao problema de classificação de sequências é apresentada. No Capítulo 3 apresentamos algumas das principais abordagens e também métodos para o problema de classificação de sequências. Nosso método é apresentado no Capítulo 4. Em seguida, no Capítulo 5 apresentamos nossos experimentos e resultados obtidos. Finalmente, no Capítulo 6 concluímos este trabalho.

Capítulo 2

Metagenômica e o problema da classificação de sequências

Diferentemente do que acontece na genômica microbiana clássica, onde precisamos obter culturas puras para realizar o sequenciamento das espécies, na metagenômica esse sequenciamento é feito diretamente a partir da amostra colhida, fazendo uso de uma grande variedade de técnicas e abordagens, podendo assim ajudar a estabelecer hipóteses a respeito de interações entre membros de comunidades e de interações com seus ambientes.

As ferramentas convencionais de estudo de genomas da microbiologia baseiam-se no isolamento de uma espécie em culturas individuais. Os micro-organismos são culivados em laboratório por meio de clonagem a fim de obter o sequenciamento do genoma de sua espécie. No entanto, apenas uma pequena parte dos organismos microbianos presentes na natureza podem ser culturados, o que significa que os dados genômicos existentes são fortemente influenciados pela clonagem e não representam a realidade dos genomas das espécies microbianas.

Além disso, são raros os micróbios que vivem em comunidades de espécies simples. As espécies interagem entre si e com seus habitats. Portanto, uma cultura de clones falha em representar a verdadeira situação de sua espécie na natureza, incluindo principalmente as interações entre organismos [50]. Estudos que não envolvam a cultura em laboratório de organismos microbianos permitem o descobrimento de novos genes e enzimas, o que não era possível com a genômica clássica.

Novas tecnologias de sequenciamento e a drástica redução de seu custo têm ajudado a Bioinformática a superar essas limitações. Tem-se, atualmente, a habilidade de se obter informações genômicas diretamente de comunidades microbianas em seus habitats naturais. Agora, em vez de observar espécies individualmente, é possível estudar dezenas de milhares de espécies, todas juntas. Dados de sequências obtidos diretamente do sequenciamento ambiental são chamados de metagenoma, e o estudo desses dados é chamado de metagenômica.

Na genômica clássica envolvendo um único organismo, praticamente todo o genoma da espécie é sequenciado. A espécie de origem das sequências obtidas é selecionada previamente para sua cultura em laboratório e o processo de montagem oferece a localização de seus genes. Como consequência, é alcançada uma visão quase completa de todos os elementos no organismo sequenciado. Pode não ser trivial reconhecer todos os elementos e alguns erros podem existir, mas é possível realizar a anotação daquelas porções do genoma que foram decifradas com sucesso.

Por outro lado, as sequências obtidas a partir de estudos metagenômicos são fragmentadas. Cada fragmento, ou read, é sequenciado a partir de uma espécie diferente. Como existem diversas espécies na amostra, muitas vezes é impossível determinar a verdadeira espécie de origem do read. O comprimento de cada read pode variar de 20 bp até 700 bp, dependendo do método de sequenciamento usado [50]. Como consequência, a reconstrução de um genoma completo geralmente não é possível. Além de ser fragmentado e incompleto, o volume dos dados de sequenciamento obtidos pelo sequenciamento ambiental é muito maior que o volume obtido na genômica de um único organismo.

Por esses motivos, a Bioinformática exige o desenvolvimento de novos algoritmos para as análises de dados metagenômicos. Essas análises fazem parte de um conjunto de etapas comumente executadas em um projeto metagenômico. Neste capítulo discutimos os principais passos dessas etapas, salientando as diferenças em relação às análises da genômica clássica.

Alguns projetos metagenômicos foram executados com o intuito de examinar o conteúdo genômico de diferentes comunidades de organismos. Um estudo realizado a partir da expedição *Sorcerer II Global Ocean Sampling* (GOS) utilizou sequenciamento *shotgun* para examinar comunidades microbianas em amostras da água do oceano coletadas na expedição. As análises realizadas permitiram predizer mais de

6 milhões de proteínas, muitas das quais não tinham similaridade com proteínas de espécies conhecidas e, assim, representaram novas famílias [51]. Outro estudo metagenômico importante envolveu a análise do ecossistema microbiano do rúmen bovino [5]. O projeto revelou um microbioma denso e complexo utilizando a tecnologia de sequenciamento *pyrosequencing* e mostrou que seu conteúdo genético pode ser útil na produção de biocombustíveis. Outro importante projeto metagenômico estudou o microbioma do intestino humano [44], revelando um ecossistema complexo, composto principalmente de bactérias não culturadas. O sequenciamento utilizado também foi o *pyrosequencing* e o estudo mostrou a existência de espécies microbianas dominantes, oferecendo revelações sobre a cadeia funcional do intestino humano.

Consideramos que um projeto metagenômico típico segue o fluxo de trabalho proposto pelo JGI (*Joint Genome Institute*), seguindo o que foi descrito por Kunin et al. [24]. O processo inicia-se com a amostragem e coleção de dados, seguidos pela extração de DNA, sequenciamento, processamento de reads e montagem. Os reads ou contigs participam do processo de predição de genes e anotação e, por fim, um passo de classificação é aplicado.

Espera-se que alguns detalhes do fluxo de trabalho descrito sejam variáveis em diferentes instalações e alguns aspectos sejam difíceis de reproduzir em pequenos laboratórios de pesquisa. O rápido avanço das tecnologias de sequenciamento pode mudar frequentemente a suíte de ferramentas recomendadas para o estudo metagenômico. No entanto, as considerações e problemas de um projeto metagenômico genérico continuarão úteis mesmo que as ferramentas atuais se tornem obsoletas.

Na seção seguinte são detalhados os passos de um estudo metagenômico genérico, divididos em três etapas principais: amostragem e geração de dados, processamento de sequência e análise dos dados.

2.1 Etapas da metagenômica

Amostragem e geração de dados

Amostragem

O primeiro passo do estudo metagenômico é a obtenção das amostras ambientais. A extração e a purificação dessas amostras ainda são o principal gargalo da metagenômica, que é agravado pelo fato de que não existe uma solução única para essa tarefa.

É desejável que as amostras obtidas envolvam organismos que representem a população como um todo. A composição da comunidade de onde provêm essas amostras tem influência no tipo de análise que pode ser feita posteriormente. As comunidades microbianas são compostas, dentre outros micro-organismos, de bactérias, arqueias e vírus.

A partir da amostra obtida, um processo de filtragem deve ser aplicado para a continuação do estudo metagenômico. A filtragem tem como objetivo eliminar porções da amostra que não se deseja estudar. Se a amostra contiver, por exemplo, exemplares eucariontes, é de interesse eliminá-los do estudo devido ao tamanho dos seus genomas. Portanto, é importante a seleção de uma comunidade que não contenha eucariontes, ou aquela onde seu DNA possa ser excluído e, apesar de novas tecnologias de sequenciamento permitirem o sequenciamento de comunidades contendo eucariontes, os dados tendem a apresentar inúmeros desafios nas etapas subsequentes.

Ainda como parte da etapa de amostragem, temos a tarefa de analisar a complexidade da comunidade em questão. A complexidade, do ponto de vista biológico, da comunidade é uma função do número de espécies da comunidade e suas relativas abundâncias. Uma comunidade com muitas espécies em quantidades semelhantes é mais complexa que uma comunidade com poucas espécies em quantidades desiguais. Como consequência, dados de sequências de uma comunidade menos complexa levam à geração de contigs maiores na fase de montagem.

Uma variável chave que pode afetar o tipo de análise que pode ser aplicado a um

conjunto de dados metagenômicos é a presença ou a ausência de populações dominantes, independente do número total de espécies presentes. Uma população de espécies dominantes possui uma maior representação na amostra, o que resulta em uma maior probabilidade de geração de contigs na montagem. Assim, o sequenciamento de uma comunidade com uma espécie dominante provavelmente produzirá uma porção significativa do genoma dessa espécie e, em alguns casos, genomas quase completos [24].

Sequenciamento

Até recentemente, genomas procariotos eram tipicamente sequenciados usando o método de sequenciamento *shotgun* Sanger [37, 50]. O primeiro passo do sequenciamento Sanger é a quebra do conteúdo do DNA de clones em fragmentos aleatórios (por isso o nome *shotgun*). Os fragmentos são clonados utilizando vetores para produzir material genético suficiente para o sequenciamento. A repetição desse processo garante que todas as partes do genoma sejam contempladas. A seguir, um software de montagem é aplicado para a transformação dos fragmentos em um genoma completo.

Na metagenômica o sequenciamento *shotgun* Sanger pode ser aplicado de maneira semelhante à genômica de culturas de clones. A diferença é que o material genético não é proveniente de um único organismo, mas de uma comunidade microbiana. A amostra pode prover apenas um esboço parcial dos organismos no ambiente, já que o material genômico da espécie mais abundante predomina no conjunto.

No entanto, tecnologias de sequenciamento de nova geração ganharam espaço rapidamente e estão substituindo o método Sanger para sequenciamento de pequenos genomas e de metagenomas. Essas tecnologias produzem uma quantidade de reads muito maior se comparadas ao Sanger, mas o tamanho médio dos reads é bem menor. O método de *pyrosequencing*, utilizado pelo Roche 454 [26] produz reads de 200 a 600 bp [14, 31]. Outras tecnologias de sequenciamento de nova geração, como o ABI SOLiD e o Illumina GAII, produzem reads ainda menores (em torno de 25 bp a 100 bp), que limitam o estudo metagenômico em geral [50]. Estimativas sugerem que reads de 100 bp estão próximos do tamanho mínimo para se obter informações úteis, o que torna o sequenciamento 454 muito indicado para a metagenômica [14].

Ambos os métodos, Sanger e 454, oferecem precisão de sequenciamento diferentes, que é alta no início de cada read e diminui conforme seu comprimento. Três tipos de erros podem ocorrer no sequenciamento:

1. erros de substituição, o que significa que um nucleotídeo errado foi lido;
2. erros de remoção, onde um ou mais nucleotídeos são omitidos;
3. erros de inserção, onde um ou mais nucleotídeos são erroneamente adicionados à sequência durante o processo de leitura.

Outras características que apontam o 454 como a tecnologia mais interessante do ponto de vista da metagenômica são a velocidade e o baixo custo do sequenciamento. O método *pyrosequencing* realizado no 454 ainda é assunto de estudos e pesquisas e espera-se, num futuro próximo, um aumento significativo no tamanho dos reads sequenciados.

Processamento de sequência

Montagem

Ao sequenciar um único genoma completamente, os reads são montados em sequências contíguas mais longas, denominadas *contigs*, a fim de se obter um genoma completo. Em dados genômicos, análises dos contigs permitem encontrar informações importantes do genoma da espécie, como *open reading frame* (ORF), ou a parte de um gene que codifica proteína.

Mais especificamente, a montagem é o processo de combinar reads em porções contíguas de DNA (contigs), baseando-se na similaridade de sequência entre os reads. A sequência consenso para um contig é então determinada com base no nucleotídeo de melhor qualidade para um dado read em cada posição, além da frequência em que os nucleotídeos ocorrem naquela posição. O número de reads sobre cada base consenso é chamado de cobertura.

Na metagenômica, a montagem completa não é possível, pois além dos problemas da genômica clássica, como orientação desconhecida, regiões repetidas e falta de

cobertura, a amostra é geralmente incompleta e praticamente todas as espécies são amostradas parcialmente. Assim, é difícil mapear reads isolados a suas espécies de origem. Além de fragmentada e incompleta, a amostra obtida na metagenômica tem um volume muito maior do que na genômica clássica [50].

Alguns métodos de montagem, como Phrap e Arachne [3], que foram desenvolvidos para a montagem de genomas isolados do sequenciamento Sanger, são capazes de oferecer bons resultados mesmo na montagem de sequências metagenômicas, desde que sequenciadas também pelo Sanger. No entanto, para reads menores, essas técnicas não são adequadas. Alguns métodos são capazes de montar reads muito pequenos, como EULER, VELVET [52, 50]. O montador CAP3 [17] também é indicado e foi usado com sucesso em dados metagenômicos microbianos de sequenciamento *pyrosequencing* do intestino humano [44].

Predição de genes

Os genes são as unidades funcionais básicas do genoma, que podem constituir unidades funcionais maiores, como unidades de transcrição (trecho de DNA transcrito em uma molécula de RNA que codifica um gene). Devido às características de incompletude e fragmentação dos dados metagenômicos, o processo de identificação de genes é um grande desafio.

Denomina-se predição de genes o processo de identificar porções de uma sequência de DNA que codificam genes. Esse processo é um dos passos mais importantes para o entendimento do genoma de uma espécie que foi sequenciado. Diversos métodos de predição de genes foram desenvolvidos para a identificação de genes codificadores em genomas microbianos completos, entre eles o Glimmer [35].

Esses métodos necessitam de uma fase de treinamento sobre alguma base de dados que inclui o genoma alvo ou um treinamento sobre o genoma de espécies relacionadas. Tais métodos convencionais podem, em princípio, ser aplicados a dados metagenômicos, dado que reads de um sequenciamento podem ser montados em contigs maiores para que forneçam informações suficientes para treinamento.

A predição de genes usando contigs metagenômicos pode ser melhorada pela combinação de contigs e reads em conjuntos filogenéticos separados. No entanto, a

montagem de reads metagenômicos é problemática. Foi demonstrado sobre metagenomas artificiais que a qualidade da montagem depende fortemente da cobertura de cada espécie nesse metagenoma [28]. Também foi mostrado que contigs pequenos correm o risco de sofrer quimerismo, isto é, um read de uma espécie A junta-se com o read de uma espécie B na montagem, o que limita o uso de contigs para análises posteriores.

Existem duas abordagens principais para predição de genes. A predição "baseada em evidência" utiliza buscas para identificar genes semelhantes àqueles observados anteriormente. Exemplos desta abordagem são simples comparações contra bancos de dados de proteínas, por exemplo, usando Blast [2]. A segunda abordagem, *ab initio*, baseia-se em características intrínsecas à sequência de DNA para identificar regiões codificantes e não-codificantes. Logo, esta abordagem não está limitada à identificação apenas de genes que possuem homólogos em bancos de dados.

Uma parte dos metagenomas podm ainda manter-se em reads isolados, não classificados, mesmo depois de aplicadas as etapas do estudo metagenômico. Em alguns casos, a montagem do metagenoma falha completamente. Por essa razão, a habilidade de predizer genes em reads individuais é essencial para explorar e entender completamente um metagenoma.

Anotação funcional

A predição de genes normalmente é seguida pelo processo de anotação funcional. Denomina-se anotação o processo de, a partir de reads produzidos na etapa de sequenciamento, realizar a interpretação necessária para extrair os significados biológicos da amostra. Em outras palavras, deseja-se anexar informações biológicas a sequências metagenômicas. Essas informações identificam características chave dos genomas, em particular, genes e seus produtos [43].

Esse processo busca entender o potencial funcional da comunidade microbiana de onde se estuda o metagenoma, além de mostrar do que os organismos são capazes como uma comunidade. A anotação funcional de dados metagenômicos é semelhante à anotação genômica e baseia-se em comparações dos genes preditos com sequências já existentes e anotadas previamente.

Essa tarefa é desafiadora quando aplicada a dados genômicos regulares, e é ainda mais complicada quando se trata da metagenômica, onde as informações são parciais e grande parte dos dados não possui homólogos em bases de dados conhecidas.

Análise dos dados

Populações dominantes

Uma das análises que podem ser feitas em dados metagenômicos é uma reavaliação da estimativa de composição da comunidade. Esse é um passo importante para a interpretação dos dados, já que podem ocorrer desvios nas estimativas de composição realizadas inicialmente.

A análise de população dominante permite conhecer melhor o ambiente estudado. Se uma amostra possui baixa complexidade, é possível que partes do genoma da população dominante tenham cobertura suficiente para permitir uma reconstrução compreensiva de algum genoma. Se mais de uma população dominante for sequenciada, a ação combinada dessas populações pode se tornar aparente.

Classificação

Ainda como parte da interpretação do ecossistema em estudo, outra análise importante é o processo de associação entre sequências e espécies, denominado **classificação**. Existem diversas abordagens utilizadas para o problema, como a comparação de sequências com domínios conhecidos ou através da frequência de oligonucleotídeos. Abordaremos com mais detalhes a tarefa de classificação na próxima seção deste capítulo.

2.2 Classificação de sequências

O fluxo de execução apresentado na seção anterior produz uma coleção de reads, contigs e genes. Associar esses dados com os organismos dos quais eles foram origi-

nados é fortemente recomendado para a interpretação da amostra, o que permite a obtenção de mais detalhes da diversidade biológica do ecossistema estudado.

O processo de classificação é definido por Brady e Salzberg [4] como a atribuição de um rótulo específico, por exemplo, um grupo filogenético, aos membros do conjunto de dados de entrada. Os mesmos autores definem *binning* como o agrupamento de reads do conjunto de dados em subgrupos distintos, que podem permanecer não rotulados. Neste caso, tanto o processo de *binning* quanto a classificação têm o objetivo comum de classificar as sequências. No entanto, *binning* não necessariamente identifica a espécie ou família que uma dada sequência pertence, podendo agrupar sequências em grupos taxonômicos diferentes.

Os autores do JGI, no entanto, definem tanto *binning* quanto classificação como o processo de associação entre sequências e espécies, ou grupos taxonômicos de mais alta ordem, sem distinção [24]. Consideramos, portanto, esta definição, onde ambos os termos se referem à atribuição de sequências a grupos relacionados filogeneticamente. A resolução dos grupos pode variar de níveis altos, como filo, até níveis mais baixos, como a espécie ou a cepa de um micro-organismo, dependendo de fatores como a estrutura da comunidade e a qualidade do sequenciamento [7].

A tarefa de classificação de sequências é complicada, pois o sequenciamento na metagenômica produz reads pequenos, atingindo uma média de 450 bp na tecnologia 454. Isso significa que as sequências obtidas têm pouca informação para que possam ser diferenciadas umas das outras e, assim, associadas a grupos distintos. Diferentes métodos de classificação estão disponíveis, divididos em duas categorias distintas: métodos baseados em similaridade de sequência e métodos baseados em composição de sequência. Detalhes desses métodos e das diferentes abordagens para o problema da classificação são apresentados no Capítulo 3.

Capítulo 3

Métodos conhecidos de classificação de sequências

Diversos métodos de classificação de sequências estão disponíveis e são adequados para classificar sequências oriundas de projetos metagenômicos. Os métodos são divididos em duas categorias: métodos baseados em similaridade de sequência e métodos baseados em composição de sequência. Na primeira categoria, a classificação é feita com base na similaridade de sequências e no uso de programas de comparação, como o Blast, para classificar um read de acordo com *hits* encontrado em bases de dados de proteínas ou domínios. Na segunda categoria, os métodos contam com o fato de que diferentes genomas podem possuir diferentes assinaturas em sua composição, tais como conteúdo GC, frequência de oligonucleotídeos, a presença ou a ausência de genes marcadores específicos, tais como o RNA ribossomal 16S.

Por definição, dado um read de um organismo cujo genoma ainda não foi sequenciado, nenhum método de classificação pode predizer corretamente o rótulo de espécie, pois esse rótulo não existe [4]. Para classificação filogenética de mais alta ordem, no entanto, é possível que já tenham sido encontrados anteriormente gêneros, famílias ou clados relacionados. Os métodos aqui descritos se baseiam nessa característica e buscam realizar a classificação em taxonomias de mais alta ordem sempre que não for possível predizer corretamente a espécie.

Algumas abordagens conhecidas para o problema de classificação são:

- Abordagens baseadas na composição das sequências
  - Usar a composição das sequências, comparando frequências de oligonucleotídeos, para caracterizar os reads filogeneticamente. Esta abordagem utiliza genes específicos para a identificação de fragmentos. Esses genes, chamados de genes marcadores, são altamente conservados e funcionam como âncoras para identificar os organismos de origem dos reads de entrada.
- Abordagens baseadas na comparação de sequências
  - Comparar os reads de uma amostra com domínios conservados e com famílias de proteínas usando bases de dados conhecidas, que incluem anotações e alinhamentos múltiplos de sequências. Os reads podem, então, ser classificados em taxonomias com base na construção de árvores filogenéticas para as famílias encontradas. Nessa situação, domínios e famílias de proteínas são considerados marcadores filogenéticos e são usados para identificar os organismos dos reads de entrada.
  - Comparar reads com proteínas ou genes conhecidos, provenientes de bases de dados conhecidas, usando programas de comparação por similaridade de sequência, como o Blast. Com os reads da entrada, são realizadas comparações diretas com bases de dados de sequências conhecidas. A atribuição de sequências às suas respectivas espécies é baseada no melhor resultado obtido pela comparação.

A seguir explicitamos ferramentas e métodos estudados que utilizam as abordagens citadas, analisando alguns de seus resultados.

3.1 Phymm

O Phymm é um método para classificação baseado tanto em composição de sequências quanto em similaridade de sequências e é capaz de gerar bons resultados, mesmo quando utilizados fragmentos muito pequenos, comumente presentes em dados metagenômicos [4].

Modelos de Markov Interpolados (*Interpolated Markov Models* - IMMs) são usados para caracterizar oligonucleotídeos de tamanhos variáveis. Modelos de Markov são uma ferramenta conhecida para a análise de sequências biológicas, e o modelo predominante para análises microbianas é a cadeia de Markov de ordem fixa. O modelo foi usado com sucesso para busca de genes de bactérias no sistema Glimmer [35], mas o Phymm é a primeira implementação conhecida de IMMs para o problema de classificação filogenética de um modo mais geral [4]. O objetivo dos experimentos do Phymm é determinar quão boa é a classificação generalizada de grandes amostras metagenômicas que possivelmente incluem organismos fortemente relacionados, sendo testado tanto em dados metagenômicos reais quanto em simulados.

Modelos de Markov

Um modelo probabilístico de sequências de DNA pode ser descrito como uma sequência de variáveis aleatórias X_1, X_2, ..., onde X_i corresponde à posição i da sequência. Cada variável aleatória X_i assume um valor do conjunto $\{a, c, t, g\}$ de bases. A probabilidade do valor que a variável X_i assume dependerá do contexto local, isto é, da base imediatamente anterior à base da posição i. Normalmente denota-se $\{a, c, t, g\}$ como o conjunto de possíveis *estados* que uma variável pode assumir. Em outras palavras, a variável X_i está no estado a se $X_i = a$.

Uma cadeia de Markov de grau 1, também chamada de cadeia de Markov de primeira ordem, é uma sequência de variáveis aleatórias onde a probabilidade de X_i assumir um valor em particular depende apenas do valor da variável precedente X_{i-1}. Uma cadeia de Markov de grau k é uma generalização dessa definição, onde a probabilidade de X_i depende somente das k bases precedentes. Uma matriz de transição, A, é uma matriz que guarda em suas células as probabilidades de transição definidas para um espaço de estados. Para sequências de DNA, uma cadeia de Markov de primeira ordem é especificada inteiramente por uma matriz de 16 probabilidades: $p(a|a)$, $p(a|c)$, ..., $p(t|t)$.

Um modelo de 5^a ordem, por exemplo, usa as cinco bases anteriores para predizer a próxima base. No entanto, a aprendizagem de tais modelos pode ser difícil quando não há dados de treinamento suficientes para estimar com exatidão a probabilidade de ocorrência de uma base dadas todas as possíveis combinações de cinco bases

precedentes.

Um modelo de Markov interpolado, também chamado de modelo de Markov de ordem variável, supera esse problema ao utilizar um número variável de estados para computar a probabilidade do próximo estado. Mais precisamente, um IMM usa uma combinação linear de todas as probabilidades baseadas nas $0, 1, 2, \ldots, k$ bases precedentes, onde k é parâmetro do algoritmo. Assim, um IMM usa um contexto maior para fazer uma predição sempre que possível, tirando proveito da maior precisão oferecida por um modelo de Markov de mais alta ordem. Onde as estatísticas de oligômeros são insuficientes para produzir boas estimativas, o IMM pode utilizar sequências menores para suas predições.

No Phymm, os IMMs são usados para classificar sequências baseando-se nos padrões de DNA distintos a um grupo filogenético, tanto para espécies ou gêneros, quanto para grupos de mais alta ordem. Durante a fase de treinamento, o algoritmo IMM constrói distribuições de probabilidade representando padrões de nucleotídeos observados que caracterizam cada espécie. Na fase de classificação, cada IMM é usado como método de pontuação: o algoritmo examina os nucleotídeos em uma determinada sequência e calcula uma pontuação correspondente à probabilidade de que essa sequência foi gerada da mesma distribuição daquela usada para treinar o IMM.

Em outras palavras, um IMM, quando usado para pontuar uma sequência de DNA, computa a pontuação correspondente à probabilidade de que o IMM tenha gerado essa sequência, o que pode ser usado para estimar a probabilidade de a sequência pertencer à classe de sequências na qual o IMM foi treinado. O Phymm possui um grande grupo de IMMs treinados com cromossomos e plasmídeos de organismos coletados da base de dados RefSeq do US National Center for Biology Information (NCBI) [33], chamada de biblioteca de referência.

A vantagem que a utilização de IMMs possui em relação a outros métodos é que eles usam informações de diversos oligonucleotídeos de diferentes tamanhos e combinam os resultados. Em um dado genoma, por exemplo, ao invés de ter que escolher entre 5 ou 6 bases para classificação no nível de classe ou filo, um IMM pode fazer ambos. Nesse caso, alguns 5-mers podem ocorrer tão raramente que não devem ser usados para predição. O IMM irá, então, retroceder para uma cadeia de Markov de menor ordem. Por outro lado, certos 8-mers podem ocorrer muito frequentemente, e para esses casos, o IMM pode usar esse contexto maior e fazer uma melhor

predição. Além disso, o IMM pode combinar a evidência da cadeia de oitava ordem com a da cadeia de quinta ordem. Com essas características, o IMM possui todas as informações disponíveis para uma cadeia de quinta ordem, somadas a possíveis informações adicionais [36].

PhymmBL

Uma variação do método original, denominada PhymmBL, pode ser vista como um híbrido entre Phymm e Blast. O PhymmBL realiza uma combinação das saídas de ambos os métodos, selecionando o melhor resultado. O híbrido PhymmBL produz melhorias significativas quando comparado aos métodos Phymm e Blast isoladamente [4]. Os resultados indicam que essas duas abordagens são, de certa forma, complementares e que o PhymmBL se aproveita de informações de ambas. O Phymm tem como uma de suas características a geração de uma pontuação para todos os reads da entrada. O Blast, por outro lado, pode, ocasionalmente, não retornar *hit* como saída.

A combinação dos resultados do Phymm e do Blast se dá da seguinte maneira: primeiro obtém-se o melhor E-value de cada read do resultado da execução isolada do Blast. Depois, para os resultados da execução isolada do Phymm, obtém-se a melhor pontuação calculadoa com IMM para cada read. Como o Blast pode não ter a pontuação para um determinado read, se isso acontecer, usa-se o valor da melhor pontuação do Phymm, que certamente existe. Caso a pontuação do Blast exista, comparam-se as pontuações e considera-se o melhor valor para gerar a saída.

Experimentos

A avaliação da classificação foi testada em dois grupos de experimentos isolados com dados reais e sintéticos. Os processos envolveram o Phymm e o Blast isoladamente, além da versão híbrida PhymmBL.

Os experimentos demonstraram que, isoladamente, o Blast é o melhor método para classificar reads metagenômicos. O Blast se apresenta falho quando a sequência comparada é completamente diferente de qualquer dado existente no banco de dados,

que é geralmente o caso que ocorre com amostras de projetos metagenômicos. Foi previamente observado que abordagens com genes marcadores, como o RNA ribossomal 16S, não aparentam melhorar a precisão da classificação em comparação com o Blast isolado [4, 25]. Por outro lado, o método baseado em IMM provê um claro reforço ao Blast com o método híbrido, PhymmBL, que superou cada um dos dois métodos isolados. Nota-se que o Phymm contribuiu com um número substancial de atribuições corretas em que o Blast errou, e vice versa.

Dados metagenômicos sintéticos

A classificação foi testada com três grupos de experimentos, nos quais foram atribuídos conjuntos de reads metagenômicos sintéticos a rótulos taxonômicos usando Phymm, Blast e PhymmBL. Foram repetidos todos os três grupos de experimentos várias vezes, com cada iteração configurada para excluir comparações de cada read a clados de níveis cada vez mais genéricos, esperando que esses dados levassem à descoberta de novas espécies.

No experimento do Phymm, cada read foi pontuado com cada IMM da biblioteca de referência e, a seguir, classificado usando os rótulos de clado pertencentes ao organismo cujo IMM gerou a melhor pontuação para aquele read. No experimento do Blast, cada read foi submetido como uma consulta Blastn contra uma base de dados construída com os mesmos genomas usados nos IMMs e os rótulos foram atribuídos usando rótulos conhecidos do melhor *hit*. Finalmente, o PhymmBL foi usado para pontuar reads usando ambos os métodos isolados, atribuindo o melhor resultado com uma combinação da pontuação dos dois métodos.

Os testes foram focados inicialmente no experimento do Phymm em que as comparações do nível de espécie foram mascaradas. O experimento foi repetido para reads de 100, 200, 400, 800 e 1000 bp. A classificação do nível de gênero foi a tarefa mais difícil, com 53 possíveis gêneros disponíveis como rótulos. Seguindo na árvore filogenética, os conjuntos de dados continham 48 famílias de bactérias e arqueias, 34 ordens, 21 classes e 14 filos. Como esperado, a precisão do Phymm aumentou conforme o tamanho dos reads, de 32,8% para gênero com reads de 100 bp, até 89,9% para filo com reads de 1000 bp. Para reads com 400 bp, tamanho próximo do oferecido pelo Roche 454, a precisão no nível de gênero foi de 60,3%.

Tabela 3.1: Comparação do desempenho dos resultados das execuções do Phymm, Blast e PhymmBL, para reads de 1000 bp.

Grupo	**Phymm**	**Blast**	**PhymmBL**
Gênero	71,1%	73,8%	78,4%
Família	77,5%	79,2%	84,8%
Ordem	80,6%	80,8%	86,9%
Classe	85,4%	84,1%	90,6%
Filo	89,8%	88,0%	93,8%

Em muitos casos, os resultados do Blast superaram os do Phymm quando executados isoladamente. Para reads de 800 bp e 1000 bp, no entanto, os resultados do Phymm foram melhores nos níveis de classe e filo. Já o híbrido PhymmBL produziu resultados com melhorias significativas. Para todos os tamanhos de reads e níveis de clado, o método híbrido superou ambos Phymm e Blast. Como visto na Tabela 3.1, o híbrido mostra melhoria sobre o Blast em todos os níveis taxonômicos para o conjunto com reads de tamanho 1000 bp.

Dados metagenômicos reais

O Phymm também foi testado em dados reais, especificamente no que foi chamado de metagenoma de drenagem ácida de mina (*Acid Mine Drainage* - AMD) [47], que é substancialmente bem caracterizado. O metagenoma contém três populações dominantes, *Ferroplasma acidarmanus* e *Leptospirillum sp.* grupos II e III.

Dois experimentos foram executados para a classificação dos dados da AMD: o primeiro usando a biblioteca RefSeq inalterada como referência para os três grupos, e o segundo, no qual foram adicionados esboços de genomas para os três grupos junto aos dados da RefSeq para criar uma biblioteca de referência aumentada. Essa modificação permitiu uma avaliação do desempenho da classificação do Phymm, Blast e PhymmBL na presença de dados de referência mais e menos específicos.

Foi examinado quão bem, usando apenas a biblioteca RefSeq sem a adição dos esboços dos genomas, o Phymm, o Blast e o PhymmBL puderam caracterizar as três populações dominantes. O PhymmBL previu corretamente o filo para *F. acidarmanus*, como *Euryarchaeota*, em 61,0% das vezes. No entanto, aproximadamente

80% dos reads de *Leptospirillum sp.* foram atribuídos para *Proteobacteria*, sendo que o filo correto seria *Nitrospirae*.

Quando foram analisados os grupos de reads com a biblioteca de referência aumentada com os esboços de genomas, a predição no nível de espécie do PhymmBL teve uma taxa de acerto de 98% para cada grupo. Esses resultados dão uma indicação da robustez do PhymmBL na presença de dados com erros de sequenciamento e mutações naturais.

3.2 Carma

CARMA [22] é um algoritmo que realiza predição da origem taxonômica de fragmentos de DNA, indicado para execução com reads provenientes de amostras ambientais, originados pelo sequenciamento 454. O algoritmo busca por domínios Pfam [12] conservados e famílias de proteínas em reads sem a necessidade do passo de montagem. A Pfam é uma base de dados de famílias de proteínas que incluem suas anotações e alinhamentos múltiplos de sequências gerados com modelos ocultos de Markov. Segundo os autores, o método oferece uma alta precisão para classificação de uma ampla variedade de grupos taxonômicos e fragmentos de genes de tamanhos tão pequenos quanto 27 aminoácidos foram classificados corretamente no nível de gênero. Um estudo comparativo foi realizado em três amostras aquáticas microbianas, revelando profundas diferenças em suas composições taxonômicas.

Nesse estudo, trechos dos reads sequenciados que contêm fragmentos que potencialmente codificam proteínas, ou fragmentos de genes, são chamados *Environmental Gene Tags*, ou EGTs.

O método é dividido em dois componentes principais: o primeiro é usado para a detecção de domínios Pfam e famílias de proteínas, considerados marcadores filogenéticos para identificar organismos fontes de reads de DNA. O segundo componente constrói uma árvore filogenética para cada família Pfam detectada, chamada de *árvore da família*. Cada árvore consiste de todos os EGTs associados a sua respectiva família, além de todos os membros da família cuja origem taxonômica é conhecida.

Profile Hidden Markov Models, ou pHMMs, são usados no primeiro componente do método. Esses modelos são muito precisos para a detecção de sequências conservadas e de sinais funcionais, mesmo em reads pequenos, tornando esta técnica adequada para a análise de reads de 454 que não passaram pelo processo de montagem.

EGTs são identificados usando modelos ocultos de Markov (pHMMs) a partir da base de dados Pfam, onde cada família é representada por alinhamentos múltiplos, pHMMs e um arquivo de anotação. Uma busca por similaridade de cada read da amostra é realizada contra a base Pfam usando Blastx. Apenas reads cujos *hits* obtiveram E-value ≤ 10 são considerados. Esse passo de pré-processamento reduz o custo computacional necessário na busca utilizando pHMMs.

Após o passo de pré-processamento com Blast, os reads são rastreados em busca de domínios Pfam conservados e famílias de proteínas usando pHMMs da base Pfam. Cada read é traduzido de acordo com o melhor *hit* do Blastx. As sequências traduzidas são alinhadas às famílias associadas usando seus pHMMs.

No segundo componente do método, uma árvore filogenética é construída para cada família Pfam detectada. EGTs são classificados em grupos taxonômicos de alta ordem baseado em suas relações filogenéticas com membros de famílias já conhecidas. O alinhamento múltiplo de membros de grupos taxonômicos conhecidos e EGTs correspondentes de cada família Pfam são usados para calcular uma distância par a par de todas as combinações desses dados. Uma árvore filogenética não-enraizada é construída com base nas distâncias calculadas. EGTs são classificados dependendo de suas relações filogenéticas com membros de grupos taxonômicos conhecidos. Se um EGT g está localizado em um grupo com membros de taxonomias conhecidas e possuem um táxon t em comum, então g é classificado como t. Caso contrário, g é classificado como *táxon desconhecido*.

Experimentos

O algoritmo foi testado extensivamente em conjuntos de dados sintéticos. Rótulos de genes ambientais (EGTs) muito pequenos, de até 27 aminoácidos, puderam ser classificados. O conjunto de dados de teste envolveu 77 genomas completos obtidos do GenBank, dos quais as origens taxonômicas foram obtidas da base de dados de taxonomia do NCBI. Um metagenoma sintético foi construído fragmentando

os genomas usando uma versão prévia do MetaSim [34], configurado para gerar fragmentos aleatórios que variam de 80 a 120 bp.

No primeiro experimento, o algoritmo completo, isto é, a detecção de EGTs seguida pela classificação filogenética, foi avaliado no metagenoma sintético compreendendo os 77 genomas completos obtidos. Erros artificiais foram introduzidos para simular os erros de sequenciamento produzidos pelo 454. O metagenoma sintético representa uma comunidade microbiana complexa, com fragmentos de sequência de bactérias e arqueias, em 10 filos, 11 classes, 29 ordens e 62 gêneros.

Um EGT foi encontrado em aproximadamente 15% dos 2,7 milhões de fragmentos analisados [22]. Em face do tamanho dos fragmentos, uma classificação com alta precisão foi conseguida. Na média, a origem taxonômica foi predita corretamente para 84% (super-reino) a 61% (ordem) de EGTs identificados. Apesar de a proporção de EGTs classificados corretamente diminuir do super-reino para ordem, a proporção de EGTs classificados erroneamente (taxa de falso negativo) é aproximadamente 7% para todos os níveis taxonômicos. Reciprocamente, a proporção de EGTs que não podem ser atribuídos a nenhum grupo taxonômico (taxa de desconhecidos) aumenta de 10% para super-reino até 31% para ordem.

Para todos os níveis taxonômicos, predições confiáveis foram obtidas. A taxa de falso positivo, isto é, a probabilidade de que um EGT tenha sido classificado erroneamente em um certo grupo taxonômico, depende da representação daquele grupo na base de dados de famílias de proteínas Pfam. A média das taxas de falso positivo variou de 2,5% para super-reino a 0,1% para ordem. A maior taxa de falso positivo foi 4,7% (para *Proteobacteria*).

No segundo experimento, a classificação de EGTs pequenos foi testada extensivamente para um amplo grupo de categorias taxonômicas, incluindo fragmentos de DNA de arqueias, bactérias e vírus. No geral, EGTs puderam ser classificados corretamente no nível de gênero. Para classes bem representadas (todos os super-reinos, 20 filos, 27 classes, 59 ordens e 69 gêneros), entre 97% (super-reino) e 68% (gênero) de táxons preditos estavam corretos. Entre 7% (super-reino) 44% (gênero) dos EGTs não puderem ser atribuídos a nenhum grupo taxonômico e foram, então, classificados como *táxon desconhecido*.

A precisão do método depende de quão bem é representada uma classe taxonômica

na base de dados Pfam. Os táxons de EGTs de classes fracamente representadas frequentemente não podem ser inferidas a partir da árvore filogenética e, nesse caso, os EGTs devem ser classificados como *táxon desconhecido*.

De acordo com os autores, os resultados dos experimentos mostraram claramente que fragmentos pequenos de domínios Pfam e famílias de proteínas são marcadores filogenéticos bem adequados para inferir as afiliações taxonômicas de fragmentos de DNA provenientes de amostras ambientais. Em comparação com métodos que dependem de apenas alguns genes marcadores, como 16S rRNA, o uso das famílias da Pfam oferece uma visão geral da composição taxonômica das amostras microbianas ambientais.

3.3 Megan

Uma outra abordagem para a classificação de sequências de metagenoma é a comparação direta com proteínas ou genes conhecidos. Essa estratégia é seguida pelo MEGAN (*Metagenome Analyser*) e pode ser aplicada em reads obtidos de qualquer projeto de metagenoma, independentemente da tecnologia de sequenciamento usada [18]. De modo geral, o programa recebe um resultado de uma comparação Blast como entrada e produz uma classificação taxonômica dos reads como saída.

MEGAN baseia sua classificação taxonômica na taxonomia do NCBI, que é uma classificação estruturada hierarquicamente de todas as espécies presentes no NCBI. Uma análise taxonômica é realizada aplicando cada read a um nó da taxonomia do NCBI, baseando-se no conteúdo de genes. Para cada read que corresponde à sequência de algum gene, o programa nomeia o read como sendo o nó do ancestral comum mais recente existente na árvore que possui aquele gene. Esse algoritmo é chamado *Lowest Common Ancestor* (LCA). Dada a simplicidade do algoritmo e a facilidade de uso do programa, o MEGAN é amplamente utilizado para classificação taxonômica, mesmo em conjuntos de dados muito grandes [19].

O fluxo de execução do MEGAN é realizado em quatro passos. Primeiramente, os reads são coletados de uma amostra usando qualquer protocolo de sequenciamento. A seguir, no segundo passo, uma comparação de sequência de todos os reads contra uma ou mais bases de dados é realizada usando alguma ferramenta de comparação

por similaridade. Um exemplo comum é a execução do Blastx contra a base *nr*.

No terceiro passo, o método processa os resultados da comparação para associar todos os *hits* a sequências conhecidas e atribuir um táxon para cada sequência, de acordo com a taxonomia do NCBI. A partir da árvore do NCBI, o algoritmo associa, para cada read r do conjunto de entrada, o mais recente ancestral comum (LCA) do conjunto de táxons dos *hits* de r. Como resultado, sequências específicas de uma espécie são associadas a táxons próximos das folhas, enquanto sequências amplamente conservadas são associadas a táxons de mais alta ordem, mais próximos da raiz da árvore.

No quarto e último passo, o usuário interage com o programa, podendo executar o algoritmo *Lowest Common Ancestor* (LCA), analisar os dados, inspecionar as atribuições de reads e produzir relatórios dos resultados em diferentes níveis da taxonomia do NCBI.

Experimentos

Experimentos do MEGAN envolveram a análise de amostras do projeto do Mar de Sargaços [48]. De quatro regiões de amostragem, 1,66 milhões de reads com tamanho médio de 800 bp foram determinados por sequenciamento Sanger em quatro conjuntos. Esses conjuntos de reads foram separados em dois grupos, o primeiro envolvendo 10.000 reads do conjunto 1 e o segundo envolvendo 10.000 reads selecionados aleatoriamente dos conjuntos 2, 3 e 4. Em ambos os conjuntos, comparações Blastx foram executadas contra a base *nr* usando parâmetros padrão. Para o conjunto 1, 1% dos reads não obtiveram *hits* ou permaneceram não rotulados. Para os conjuntos 2, 3 e 4, $< 3\%$ dos reads não tiveram *hits* ou permaneceram não rotulados.

A tecnologia de sequenciamento Roche GS20 foi usada no conjunto de dados de mamute de Poinar et al. [32], obtendo 302.692 reads de tamanho médio de 95 bp. Como outros exemplares de mamute já estudados mostraram conter grande quantidade de sequências ambientais, além do DNA do próprio animal, o estudo foi considerado um projeto de metagenômica. Para determinar a distribuição de sequências ambientais na amostra, uma comparação Blastx foi executada contra toda a base de sequências *nr*. De 302.692 reads da entrada, 52.179 resultaram em um ou mais alinhamentos (17,2%). Esses resultados foram, então, usados como entrada para o MEGAN e

o algoritmo LCA para computar as atribuições de reads a táxons, obtendo, assim, uma estimativa do conteúdo taxonômico da amostra.

Para avaliar a identificação de espécies em reads tão pequenos quanto 35 bp, originários de sequenciamentos de nova geração para metagenômica, um conjunto de reads de um genoma já conhecido foi processado como se contivesse dados metagnômicos e, então, foi analisada a precisão das atribuições.

Para esse experimento, foram selecionadas sequências de genoma de dois organismos, *E. coli K12* e *B. bacteriovorus HD100*. Para cada genoma, intervalos de tamanhos de sequências de 35 bp, 100 bp, 200 bp e 800 bp foram usadas, já que esses tamanhos correspondem ao tamanho médio gerado pelas tecnologias de sequenciamento mais recentes. Reads simulados de sequenciamento *shotgun* foram comparados à base de dados *nr* usando Blastx e então processados com o MEGAN, retendo apenas *hits* que estão dentro de 20% do valor do melhor *hit* para um read, além de descartar todas as atribuições isoladas (reads que tiveram um só *hit*). A porcentagem de reads classificados para *E. coli* é mostrada na Tabela 3.2. Os resultads mostram que reads pequenos, em geral, podem ser usados em análises metagenômicas, embora com o custo de oferecer uma baixa predição.

Tabela 3.2: Resultados para a simulação de *E. coli* no MEGAN.

	35 bp	**100 bp**	**200 bp**	**800 bp**
Enterobacteriaceae	22%	64%	73%	85%
Gammaproteobacteria	24%	77%	86%	94%
Proteobacteria	25%	83%	89%	96%

Segundo os autores, para análises metagenômicas mais detalhadas, é de particular interesse resolver a árvore taxonômica no nível de espécie. A análise de dados metagenômicos produzirá um conjunto significativo de sequências que não podem ser atribuídas a alguma espécie conhecida. Reads de tamanho pequeno resultam em baixa predição, o que reduz a eficiência das novas tecnologias de sequenciamento. Atribuições de reads menores que 50 bp sofrerão com resultados de baixa confiança, enquanto reads de tamanho a partir de 100 bp podem ser atribuídos com um nível de confiança razoável.

3.4 RAIphy

O RAIphy é um método de classificação baseado em composição de sequências que modela uma unidade taxonômica com um conjunto de parâmetros chamados de Índice de Abundância Relativa (*Relative Abundance Index* - RAI) [30]. De acordo com esse modelo, um índice é calculado para cada k-mer baseado nas estatísticas de excesso ou falta de abundância obtidas de uma unidade taxonômica. Essas estatísticas são obtidas com referência a uma sequência de modelos de Markov. Um dado fragmento aleatório de genoma recebe uma pontuação de associação relativa a um táxon somando-se os valores de índices no modelo RAI para o táxon, para cada k-mer encontrado no fragmento. O fragmento é atribuído ao táxon que resulta na maior pontuação.

Existem três componentes do algoritmo RAIphy: o Índice de Abundância Relativa (RAI) e seu modelo para fragmentos genômicos e unidades taxonômicas; a métrica de classificação; e o algoritmo iterativo para refinar os modelos usados no processo de classificação. A seguir esses três componentes são detalhados.

Índice de Abundância Relativa

O Índice de Abundância Relativa (RAI) é uma medida da abundância relativa de oligonucleotídeos em fragmentos genômicos. De acordo com a disponibilidade de sequências genômicas para uma unidade taxonômica, uma pontuação é atribuída a cada possível k-mer. Essas pontuações são mais altas para k-mers representados em excesso e mais baixas para k-mers sub-representados. As representações de cada k-mer são medidas que usam a probabilidade de um k-mer ser observado e a frequência de fato observada.

Para um k-mer representado em excesso, a frequência observada é maior do que a frequência esperada e a pontuação RAI é positiva. A pontuação é negativa para k-mers sub-representados. Usando contagens de frequências relativas como estimativas para probabilidades de k-mers em uma sequência, calcula-se as probabilidades de k-mers esperadas.

As probabilidades usadas para obter os valores RAI para k-mers são estimadas

usando frequência relativa de ocorrência daquele k-mer na unidade taxonômica. Assim, para cada unidade taxonômica para a qual são conhecidas sequências genômicas, computa-se um perfil RAI correspondente. Esse perfil serve como um modelo para aquela unidade taxonômica.

Métrica de Classificação

O segundo componente é a métrica de classificação. Para atribuir um fragmento de genoma de uma fonte desconhecida a uma unidade taxonômica, primeiro computam-se as frequências relativas de ocorrência de cada k-mer do fragmento. Para cada unidade taxonômica candidata, obtém-se uma pontuação de associação. Para um k-mer que ocorre frequentemente, a frequência de ocorrência será alta e o valor RAI do k-mer para a unidade taxonômica será positivo. Quanto mais frequentemente o k-mer ocorre, maiores serão os valores tanto do RAI quanto da frequência da ocorrência. Para k-mers que ocorrem menos frequentemente que o esperado, a frequência de ocorrência será baixa e o valor RAI do k-mer para a unidade taxonômica será negativo.

Refinamento dos Modelos

O terceiro componente do método é responsável pelo refinamento dos modelos de genomas. O componente usa heurísticas que presumem que fragmentos genômicos de genomas desconhecidos de fato existem na entrada analisada. O procedimento de refinamento consiste na repetição de duas fases. Na primeira, perfis RAI são estimados a partir de genomas de organismos conhecidos. Cada fragmento é classificado realizando sua atribuição a genomas que retornaram a pontuação RAI máxima. Na segunda fase, as frequências de oligonucleotídeos e, subsequentemente, os perfis RAI para cada classe são recalculados.

As duas fases são repetidas iterativamente até que um critério de parada seja atingido. A cada refinamento, os fragmentos são representados com perfis RAI melhorados e, assim, a pontuação média de associação tende a aumentar. Quando o aumento da mudança na pontuação média de associação se torna pequeno o suficiente, o processo de refinamento é parado. No algoritmo, o critério de parada é atingido quando a melhoria na pontuação é menor que 1% do atingido na iteração

anterior.

Experimentos

Para conduzir os experimentos de forma controlada, os autores usaram dados metagenômicos sintéticos, que foram criados usando genomas disponíveis na base de dados RefSeq do NCBI [33]. Foi construído um banco de dados que armazena perfis RAI para 1.146 genomas disponíveis. Esses dados serviram como entrada para uma execução inicial do RAIphy. Para classificação filogenética e rotulação, foram coletadas informações taxonômicas da base de dados de taxonomias do NCBI. Os dados coletados compreendem 609 espécies, 318 gêneros, 158 famílias, 88 ordens, 41 classes e 26 filos.

Uma vez que o RAIphy foi desenvolvido como um algoritmo iterativo, que reavalia seus modelos de acordo com as mudanças na pontuação de associação, os parâmetros permaneceram constantes para todo o espectro de comprimentos de fragmentos.

Um dos conjuntos de experimentos realizados consistiu em testar a precisão do método para reads no intervalo de 100 a 1.000 bp. Os experimentos foram divididos em faixas, de acordo com o tamanho dos reads. As primeiras faixas representam os reads de menor tamanho, que correspondem a fragmentos de sequência gerados a partir de um sequenciamento de metagenômica [30].

O desempenho da classificação para fragmentos genômicos de maior tamanho, variando de 800 bp até 50 Kbp, também foi avaliado. Essas faixas também são significativas, pois elas representam os tamanhos obtidos nos contigs gerados por processos de montagem.

Na classificação taxonômica, a geração de um baixo número de predições altamente confiáveis é preferida à predição da maioria dos fragmentos em que as rotulações são pouco confiáveis. Quando isso acontece, fragmentos genômicos com pontuações confiáveis podem ser classificados, enquanto fragmentos duvidosos são rotulados como “desconhecidos”.

O RAIphy também foi testado usando um conjunto de dados reais (não simulados). O experimento foi realizado em um subconjunto do metagenoma *Acid Mine Drainage*

(AMD) [47]. A amostra AMD, como já foi dito, consistiu de uma comunidade de baixa diversidade, denominada pelas três populações microbianas: *Ferroplasma acidarmanus* e *Leptospirillum sp.* grupos II e III. Como esses organismos existem de forma abundante na comunidade, foi possível realizar a montagem de esboços de seus genomas (*draft genomes*). Assim, foi possível determinar de forma precisa quais reads pertencem a cada organismo, utilizando alinhamento de sequências.

É importante lembrar que o controle com dados originados de sequenciamentos reais é muito limitado, pois os rótulos verdadeiros de reads não são completamente conhecidos ou a rotulação tem baixa qualidade.

Os autores conduziram um experimento comparando o RAIphy com o Phymm, o MEGAN e o CARMA. Esse experimento permitiu observar a precisão da classificação do método para um subconjunto de um metagenoma real. A atribuição de taxonomia do RAIphy superou os métodos Phymm e MEGAN. O PhymmBL, no entanto, se saiu melhor, a um custo de tempo de execução substancialmente maior [30]. O tempo de execução do RAIphy escala linearmente com o tamanho médio dos fragmentos do metagenoma da amostra. Os experimentos foram executados em quatro horas para classificar o metagenoma AMD. Com o mesmo conjunto de dados, tanto o CARMA quanto o MEGAN necessitaram mais que 24 horas. O PhymmBL ainda levou 464 horas para processar o mesmo conjunto de dados [30].

Capítulo 4

BINGRO

Neste capítulo descrevemos o método de classificação de sequências metagenômicas proposto neste trabalho. O método, denomindado *BINGRO*, de **BIN**ning by **GRO**uping, tem como entrada um conjunto de reads, como saída um conjunto de espécies para cada read e se baseia principalmente em similaridade de sequências, montagem de fragmentos e filogenia. Dentro dessa abordagem, o método possui três grandes fases:

- agrupamento de reads de acordo com as proteínas obtidas como *hits* na base *nr*, usando Blastx;
- montagem de cada grupo de reads obtidos na fase anterior; e
- construção de uma árvore filogenética de proteínas *nr*, obtidas a partir de Blastx dos contigs da fase anterior.

A principal característica do BINGRO é que ele realiza a montagem apenas de reads similares. Para isso, uma fase de agrupamento é realizada sobre os reads da entrada de acordo com as proteínas obtidas de uma execução do Blastx. A montagem é, então, realizada para os reads de cada agrupamento obtido. Os contigs gerados pelas montagens são usados para a obtenção de proteínas de espécies semelhantes através de uma nova execução do Blastx. A partir dos melhores *hits* dessa nova execução, uma árvore filogenética é construída com o intuito de selecionar um pequeno grupo das espécies mais próximas, que constituirão o resultado da classificação do método para os reads que foram arupadas. O fluxo de execução pode ser visto na Figura 4.1.

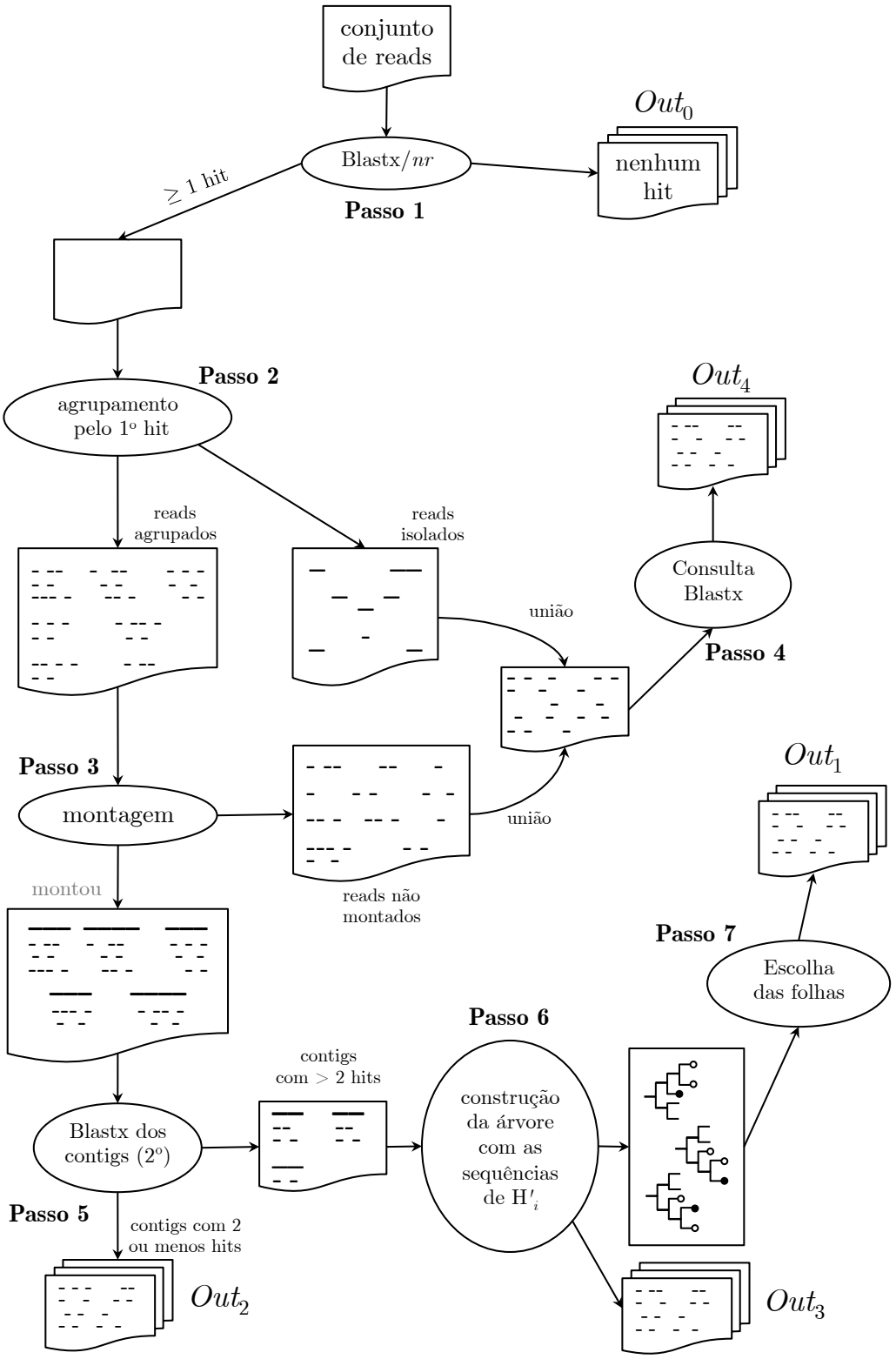


Figura 4.1: Fluxograma representando a execução do BINGRO, detalhando todos os passos e saídas.

Ao invés de oferecer como saída uma única espécie para cada read, como é feito no Phymm [4], BINGRO gera como saída um pequeno conjunto de nós da árvore filogenética, representando as espécies mais próximas do contig montado. A ideia por trás dessa abordagem é que podemos esperar contigs mais promissores ao montar apenas reads que tiveram inicialmente a mesma proteína como *hit*, em contraste com a montagem de todos os reads da amostra. Além disso, ao oferecer como saída um conjunto de espécies próximas, BINGRO tenta garantir que, se não é possível atribuir ao read uma espécie ainda não sequenciada, a espécie correta é filogeneticamente próxima àquelas representadas na árvore.

BINGRO recebe como entrada um conjunto R de reads em um arquivo no formato Fasta, e gera como saída 4 conjuntos, denominados Out_1, Out_2, Out_3 e Out_4. Esses conjuntos são compostos de reads da entrada e da determinação das espécies de cada read. Cada conjunto de saída é preenchido em determinado passo do método. Um quinto conjunto, denominado Out_0, é também gerado e contém reads cujas espécies não podem ser determinadas.

Tendo como guia as três fases descritas anteriormente, BINGRO é composto por 6 passos, que são descritos detalhadamente nas seções a seguir. A descrição desses 6 passos pode ser acompanhada através da Figura 4.1. A entrada do método é composta por:

- o conjunto R de reads;
- κ, o número máximo de *hits* a ser considerado pela execução Blastx; e
- δ, a distância máxima entre folhas da árvore, a partir da qual espécies são consideradas próximas.

4.1 Agrupamento de reads

Dado o conjunto de entrada R, nossa primeira tarefa é a de realizar a comparação de todos os reads contra a base de sequências de proteínas não redundantes *nr*, utilizando Blastx, a fim de obter as primeiras informações sobre os dados presentes na amostra.

Usamos a saída do Blastx para gerar agrupamentos de maneira que cada read $r_i \in R$ seja agrupado de acordo com seu primeiro *hit*. Chamamos esse agrupamento de G_i. Como a consulta Blastx é feita a uma base de proteínas, denominamos o primeiro *hit* como a proteína p_i do grupo, e chamamos de s_i a sua espécie.

Em outras palavras, criamos um grupo G_i contendo os reads da entrada que tiveram p_i como primeiro *hit*. Dessa forma, reads com equivalência a uma mesma proteína, de acordo com o Blastx, são adicionados a um mesmo agrupamento.

Essa fase é obtida pela execução dos passos 1 e 2, assim descritos:

Passo 1. Execute Blastx de todos os reads de R contra a base *nr*. Adicione todos os reads sem *hits* ao conjunto Out_0 e descarte-os do restante do método.

Passo 2. Crie um grupo G_i de todos os reads contendo p_i como primeiro *hit*. Seja s_i sua espécie.

Reads sem *hits* e grupos com apenas um read

Se, para um determinado read da entrada, Blastx não retornou *hit* algum, esse read será descartado do restante do fluxo de execução. Todos os reads com essa característica são adicionados ao conjunto de saída Out_0 e não é possível continuar o método para eles. O conjunto Out_0, portanto, não oferece classificação para os reads que armazena. Caso a consulta tenha retornado pelo menos um *hit*, teremos um grupo G_i e o método pode continuar no Passo 2 (veja a Figura 4.1).

Quando do agrupamento, pode acontecer de um read não ter sido agrupado. Isso acontece quando apenas esse read teve uma determinada proteína como *hit*. Reads desse tipo, chamados de *reads isolados*, são separados e posteriormente juntados aos reads que não foram montados em contigs, para posterior execução do Passo 4.

4.2 Montagem

Estratégias de sequenciamento usadas em projetos metagenômicos produzem uma grande quantidade de reads pequenos. A montagem refere-se ao alinhamento e à fusão de reads em regiões que se sobrepõem a fim de se obter uma sequência maior, denominada contig. Para o processo de montagem, contamos com o uso do software CAP3 [17].

Resumidamente, o algoritmo de montagem do CAP3 consiste de três fases principais. Na primeira fase, regiões $5'$ e $3'$ de baixa qualidade de cada read são identificadas e removidas e sobreposições entre os reads são calculadas. Sobreposições falsas são identificadas e removidas. Na segunda fase, os reads são unidos para formar contigs em ordem decrescente de pontuações de sobreposição e correções são aplicadas aos contigs. Na terceira fase, um alinhamento múltiplo de sequências é construído e uma sequência consenso é computada para o contig, juntamente com um valor de qualidade de cada base.

Seguindo o fluxo de execução, realizamos a montagem dos reads de cada grupo não isolado (formado por pelo menos dois reads), obtendo um certo número de contigs como resultado. O resultado ideal da montagem seria uma saída de apenas um grande contig, mas esse não é sempre o caso. Assim, selecionamos o maior contig gerado, denominando-o C_i e adicionando-o a uma lista denominada "sequências para Blast de contigs". Quando a montagem for realizada para todos os grupos, a lista será usada como entrada para uma nova execução do Blastx, que constitui o Passo 5.

Os dados da consulta da nova execução do Blastx serão os contigs obtidos a partir das montagens, presentes na lista de "sequências para Blast de contigs". A ideia por trás dessa abordagem é que podemos obter contigs mais promissores ao montar os reads de uma mesma proteína, em vez de simplesmente realizar a montagem de todos os reads da entrada.

A parte principal desta fase consiste na execução do Passo 3.

Passo 3. Para cada grupo G_i obtido pelo Passo 2 e com pelo menos dois reads, faça a montagem dos reads de G_i. Seja C_i o maior contig gerado. C_i fará parte da entrada para o Passo 5.

Contigs vazios e reads não montados

As exceções para o Passo 3 são os casos em que um read não participa de uma montagem, ou a determinação de contigs vazios, que acontecem devido a má qualidade dos reads envolvidos ou até mesmo pela quantidade insuficiente de reads. Em ambos os casos, chamamos esses reads de "reads não montados" e os adicionamos à entrada do Passo 4.

Lembre-se de que nós já havíamos adicionado os chamados reads isolados à entrada do Passo 4 (esses reads eram a exceção do Passo 2). Dessa forma, o Passo 4 pode ser agora executado, contendo como entrada reads isolados (exceção do Passo 2) e reads não montados (exceção do Passo 3). O tratamento para esses reads é o mesmo: é feita, para cada um deles, uma consulta na saída do Blastx já executado no Passo 1.

A saída do Passo 4 é o que chamamos de Out_4. Veja a Figura 4.1.

Passo 4. Para cada read isolado ou não montado, selecione os $k \leq \kappa$ primeiros *hits*, obtidos no Blastx do Passo 1, e retorne como saída as respectivas espécies desses *hits*.

4.3 Filogenia

O próximo passo do método é criar uma árvore filogenética T_i para cada grupo G_i de reads, com a intenção de encontrar espécies que sejam próximas da proteína p_i, e associar a essas espécies cada read pertencente a G_i.

Para tanto, precisamos primeiro escolher quem serão as folhas dessa árvore. Essa escolha é feita através do Passo 5, onde uma segunda execução do Blastx é feita, só que agora tomando como entrada o contig C_i obtido no Passo 3. Dessa forma, as folhas da árvore, quando for possível construí-la, serão os $k \leq \kappa$ *hits* obtidos por esse Blastx dos contigs, além da proteína p_i.

Passo 5. Execute Blastx contra *nr* de cada contig C_i montado no Passo 3, obtendo como saída os $k \leq \kappa$ primeiros *hits* e gerando assim o conjunto $H_i = \{h_1, h_2, \ldots, h_k\}$.

O principal critério para avançarmos rumo à construção de T_i é o tamanho do conjunto H_i. Seja $H'_i = H_i \cup \{p_i\}$. Avançaremos para a construção de T_i somente se $|H'_i| \geq 4$. O número quatro aqui escolhido refere-se ao fato de que não faz sentido a construção de uma árvore filogenética para 3 ou menos objetos.

Dessa forma, todos os reads de um grupo G_i tal que $|H'_i| < 4$ farão parte da exceção do Passo 5 e serão resolvidos sem a construção de T_i. A resposta do método para cada tal read será dada pelas espécies correspondentes às proteínas pertencentes a H'_i. A saída dessa exceção formará o conjunto Out_2 (veja a Figura 4.1).

Escolhidas as folhas de cada árvore T_i, vamos à sua construção. O algoritmo de construção da árvore toma como entrada um alinhamento múltiplo das sequências.

Alinhamento múltiplo

BINGRO usa o software Muscle [9] para a construção de um alinhamento múltiplo das sequências de H'_i. Os passos fundamentais do Muscle são a construção de uma árvore guia e o alinhamento par a par, usado para gerar um alinhamento progressivo, que é ao final refinado.

Uma maneira alternativa de obter o alinhamento é a realizada pelo CLUSTALW [45]: cada par de sequências da entrada é alinhado e uma identidade par a par é computada. As identidades são convertidas em uma medida de distância e, finalmente, a

matriz de distâncias é convertida para uma árvore guia usando o método de agrupamento (*cluster*) *Neighbor-Joining*.

O Muscle, porém, usa um método consideravelmente mais rápido, a um custo de obter um resultado mais aproximado, para computar distâncias: o método conta o número de subsequências pequenas (chamadas de k-mer) que duas sequências tem em comum, sem construir um alinhamento. Essa abordagem é executada muito mais rapidamente que o método do CLUSTALW, mas as árvores serão geralmente menos precisas. Este primeiro passo é chamado de agrupamento de k-mer ("*k-mer clustering*").

O segundo passo é usar a árvore guia para construir o que é conhecido como alinhamento progressivo. Em cada nó da árvore binária, um alinhamento par a par é construído, progressivamente das folhas até a raiz. A partir desses alinhamentos, pode-se computar, para cada par de sequências, suas identidades par a par. O resultado é uma matriz de distâncias, que será usada para estimar uma nova árvore guia. Compara-se a nova árvore com a árvore antiga e os subgrupos são realinhados, quando necessário, para produzir um alinhamento múltiplo progressivo da nova árvore. Esse procedimento é repetido até que a árvore estabilize ou até que um número máximo de iterações seja atingido. Esse processo é conhecido como "refinamento de árvore", embora ele também tenda a melhorar o próprio alinhamento.

O próximo passo é projetado para melhorar o alinhamento múltiplo. O conjunto de sequências é dividido em dois subconjuntos e um perfil é construído para cada um baseado no alinhamento múltiplo atual. Esses dois perfis são então realinhados entre si usando o mesmo algoritmo par a par do passo anterior. Se isso melhorar a qualidade do alinhamento, então o novo alinhamento é mantido e, em caso contrário, é descartado.

Qualidade do alinhamento

O uso de alinhamentos múltiplos na análise filogenética, principalmente com sequências pouco conservadas, necessita da eliminação de posições fracamente alinhadas e regiões divergentes, já que, para criar uma árvore de boa qualidade, as sequências não devem ser nem muito similares, a ponto de serem desprovidas de informações relevantes, nem muito divergentes, a ponto de muitas posições serem saturadas por

múltiplas substituições [13].

O software Gblocks [6] faz esse trabalho, selecionando blocos conservados de um alinhamento múltiplo de acordo com um conjunto de características das posições do alinhamento. Primeiro, o grau de conservação de cada posição do alinhamento é calculado, sendo definido de acordo com uma dentre três classificações: não-conservado, conservado e altamente conservado. Todos os segmentos contíguos não-conservados maiores que um certo parâmetro são rejeitados, pois nessas regiões os alinhamentos são normalmente ambíguos. Nos blocos restantes, suas extremidades são removidas até que os blocos estejam cercados por posições altamente conservadas. Dessa maneira, os blocos selecionados são ancorados por posições que podem ser alinhadas com alta confiança.

Em seguida, o método retira do alinhamento segmentos com excesso de *gaps*. Posições não-conservadas adjacentes a esses segmentos também são removidas, até que uma posição conservada seja alcançada, pois regiões adjacentes a um *gap* são difíceis de alinhar.

Executamos o Gblocks com o intuito de tornar os dados mais apropriados para a reconstrução filogenética. O software toma como entrada o alinhamento das proteínas realizado pelo Muscle.

Construção da árvore

Para a construção da árvore, BINGRO utiliza o software RAxML [42], tendo como entrada o alinhamento gerado pelo Muscle e filtrado pelo Gblocks. RAxML é descrito a seguir.

Várias técnicas podem ser usadas para computar os relacionamentos evolutivos da filogenia, sendo a que apresenta maior precisão a da máxima probabilidade (*maximum likelihood*) [10]. Infelizmente, o número de possibilidades de topologias da árvore cresce exponencialmente a medida que o número de objetos aumenta, além do custo computacional da própria função probabilidade (*likelihood*) ser alto. Assim, a introdução de heurísticas para reduzir o espaço de busca em termos de topologias de árvores potenciais torna-se inevitável.

Máxima probabilidade (*Maximum Likelihood*) é um método para inferência de filogenia. O método avalia uma hipótese sobre a história evolucionária em termos da probabilidade de o modelo proposto e uma história hipotética darem origem ao conjunto de dados observado. A suposição é que uma história com uma alta probabilidade de atingir o estado observado é preferida sobre uma história com uma menor probabilidade. O método busca por árvores com a maior probabilidade (*likelihood*).

RAxML constrói uma árvore, conhecida como *árvore de parcimônia*, com o utilitário `dnapars` do pacote PHYLIP [11], e a utiliza como árvore inicial. Parcimônia é um método estatístico comumente usado para estimar filogenias onde a árvore filogenética preferida é aquela que requer a menor mudança evolutiva para explicar os dados observados. A árvore de parcimônia está relacionada com a *maximum likelihood* sobre modelos evolucionários simples, uma vez que espera-se obter uma árvore inicial com um valor de probabilidade relativamente bom comparado a árvores iniciais aleatórias ou que utilizam *neighbor-joining* [46].

O utilitário `dnapars` utiliza adição gradativa (*stepwise addition*) [10] para a construção da árvore e é relativamente rápido. O algoritmo de adição gradativa permite a construção de árvores iniciais distintas ao utilizar uma ordem aleatória de sequências de entrada. Assim, o RAxML pode ser executado várias vezes com diferentes árvores iniciais e mesmo assim gerar árvores finais distintas. O conjunto de árvores finais pode ser usado para construir uma árvore consenso e aumentar a confiabilidade do resultado final, já que o RAxML explora o espaço de busca com diferentes pontos de partida.

Na execução do RAxML, é selecionado o modelo de substituição de aminoácidos baseado na matriz PAM (*Point Accepted Mutation*) de Dayhoff [8].

Finalmente, o Passo 6 é responsável pela construção da árvore filogenética das sequências de H'_i.

Passo 6. Usando RAxML, construa a árvore filogenética T_i para as sequências de H'_i.

A saída do RAxML é a árvore filogenética gerada no formato Newick, um padrão

para representar árvores em formato facilmente lido por computador, que faz uso da correspondência entre parênteses aninhados e vírgulas [21]. A Figura 4.2, gerada pela ferramenta `nw_display`, representa graficamente um exemplo de saída do Passo 6.

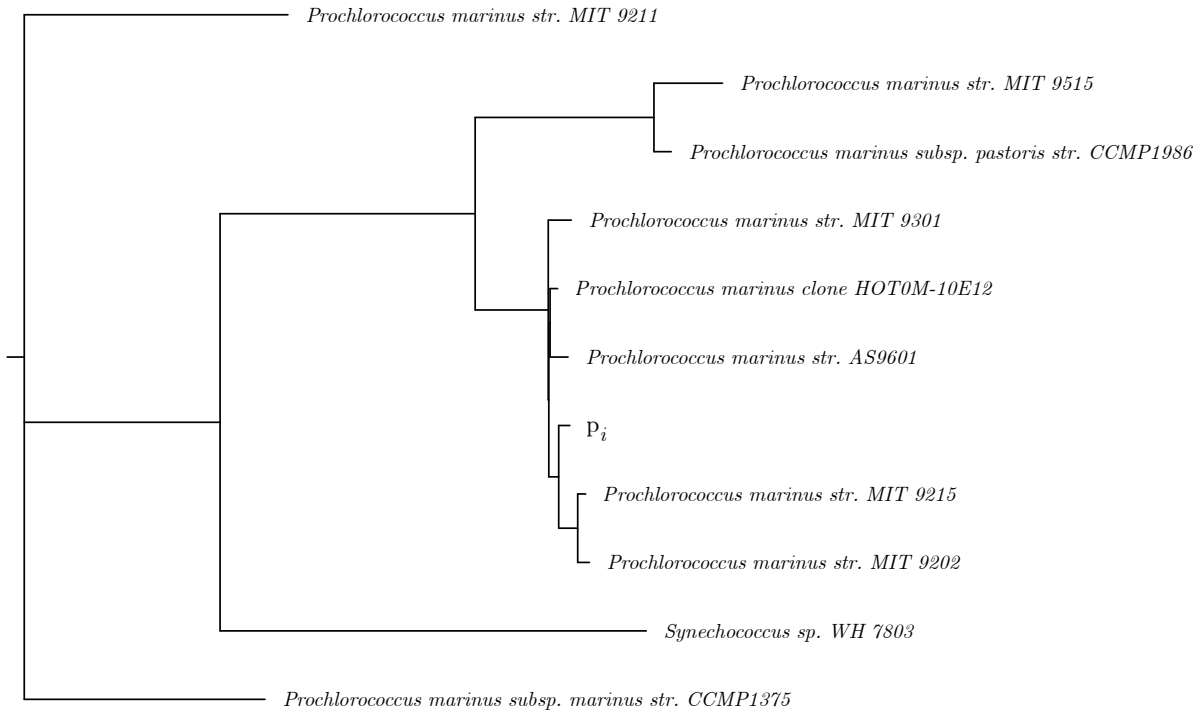


Figura 4.2: Árvore filogenética representada graficamente gerada a partir da árvore no formato Newick pelo software Newick Utilities. As distâncias entre os nós foram omitidas.

Critério para a escolha das espécies

Dada a árvore T_i gerada com o RAxML, precisamos escolher quais as espécies representativas dos reads do grupo G_i. Para tanto, precisamos usar as distâncias entre os nós em T_i. Particularmente, estamos interessados nas distâncias em que p_i se encontra em relação aos demais nós de T_i. Para obter essas distâncias, uma análise da árvore é feita com o pacote de programas Newick Utilities [21]. Computamos, com a ferramenta `nw_distance`, uma matriz das distâncias entre todos os nós de T_i.

A análise de T_i tem o intuito de obter informações de proximidade entre as espécies do grupo. Selecionamos, então, o conjunto S_i das folhas da árvore cuja distância

para p_i é menor ou igual a um certo δ (entrada para o método). O método termina adicionando ao conjunto de saída Out_1 todos os reads de G_i e o conjunto de espécies S_i como a classificação desses reads. Finalmente o Passo 7, passo final de nosso método, executa essa tarefa.

Passo 7. Determine quais as folhas de T_i estão à distância não maior que δ de p_i. Seja S_i o conjunto de espécies distintas representadas pelas proteínas presentes nessas folhas. Atribua as espécies de S_i a todos os reads de G_i.

Para o cálculo de δ, BINGRO usa do seguinte raciocínio. Seja f uma folha de T_i e seja m_f a média das distâncias entre f e todas as outras folhas de T_i. Uma folha g está *próxima* de f se a distância entre f e g for menor ou igual a m_f. Assim, a distância usada para reportar as espécies mais próximas na árvore é $\delta = m_f$.

Exceções na construção da árvore

Há ainda exceções que ocorrem na geração da árvore. Mesmo com 4 ou mais elementos, sua construção pode não ser bem sucedida e o resultado é uma árvore vazia. Nesse caso, para cada grupo que não obteve a geração da árvore, seus reads são adicionados ao conjunto de saída Out_3, juntamente com sua classificação, isto é, exatamente as espécies correspondentes às proteínas pertencentes a H'_i.

Capítulo 5

Experimentos

A avaliação de qualquer método relacionado à metagenômica é complicada devido à falta de confiabilidade da anotação de reads em bases de dados conhecidas [16].

Com o objetivo de testar o BINGRO, fizemos vários experimentos levando em consideração sua utilização como método individual, executando-o direta e isoladamente com conjuntos de reads de entrada e produzindo uma classificação desses reads no nível taxonômico de espécie como saída, e considerando-o como método auxiliar de algum programa de classificação já conhecido, procurando classificar corretamente reads que esse outro programa tennha classificado incorretamente.

5.1 Dados simulados

Ambos os métodos de sequenciamento descritos no Capítulo 1 produzem erros. O sequenciamento Sanger produz taxas de erros que variam de 0,001% para mais de 1% e dependem fortemente do software que é usado para processamento posterior dos reads. Para o sequenciamento 454 foram reportadas taxas de erro a partir 0,49% em reads de tamanho entre 100 e 200 bp.

Assim, a ideia para testar o BINGRO é a de utilizar conjuntos de reads que apresentam taxas de erros similares às apresentadas pelo sequenciamento 454, uma vez que esse tipo de sequenciamento é mais vantajoso para projetos de metagenômica [16].

Para tanto, utilizamos conjuntos de reads disponibilizados por Hoff [16], que por sua vez, utilizou o programa de simulação de reads MetaSim [34], que toma como entrada um conjunto de genomas e gera reads simulados com diferentes taxas de erros, definidas pelo usuário.

Os conjuntos de reads disponibilizados por Hoff foram, como já dissemos, gerados pelo MetaSim. Esses conjuntos foram gerados com o objetivo de abranger uma ampla gama de filos. Assim, um conjunto de microorganismos procariotos foi selecionado de acordo com esse critério de abrangência para a geração. A Tabela 5.1 mostra a lista de genomas selecionados por Hoff, e também utilizados para os nossos testes.

Tabela 5.1: Espécies cujos genomas foram usados para gerar os reads simulados, os filos a que pertencem e suas contagens na amostra.

Espécie	**Filo**	**Contagem**
Acholeplasma laidlawii PG-8A	Termicutes	4,25%
Buchnera aphidicola str. APS	Proteobacteria (β)	1,81%
Burkholderia pseudomallei K96243	Proteobacteria (γ)	20,64%
Chlorobium tepidum TLS	Bacteriodetes	6,06%
Corynebacterium jeikeium K411	Actinobacteria	7,02%
Desulfurococcus kamchatkensis 1221n	Crenarcheota	3,94%
Dictyoglomus thermophilum H-6-12	Dictyoglomi	5,62%
Exiguobacterium sibiricum 255-15	Firmicutes	8,56%
Herpetosiphon aurantiacus ATCC 23779	Chloroflexi	r,00%
Hydrogenobaculum sp. Y04AAS1	Aquificae	4,30%
Natronomonas pharaonis DSM 2160	Euryarchaeota	7,24%
Nitrosopumilus maritimus SCM1	Crenarcheota	4,56%
Prochlorococcus marinus str. MIT 9312	Cyanobacteria	4,83%
Wolbachia endosymbiont str. TRS	Proteobacteria (α)	3,07%

O uso dos conjuntos de reads simulados oferece vantagens como a disponibilidade de informações importantes para nossas avaliações. As espécies que originaram os reads, por exemplo, são usadas para avaliar nossa taxa de acerto. Além disso, a qualidade da anotação tende a ser maior para genomas completos e, por isso, usamos reads simulados originados de genomas anotados para nossos experimentos.

Os quatro conjuntos de reads simulados disponibilizados por Hoff e utilizados em nossos experimentos apresentam tamanho médio de 450 bp e taxas de erro variando

de 0 a 2,8%, que são características aproximadas de reads oriundos de sequenciamento 454. Eles foram denominados S_1, S_2, S_3 e S_4. Cada conjunto possui 98.974 reads e diferentes taxas de erros, de acordo com o que está detalhado na Tabela 5.2.

Tabela 5.2: Quatro conjuntos de reads (com 98.974 reads cada) simulados e suas taxas de erro.

Conjunto	**Taxa de erro**
S_1	0,00%
S_2	0,22%
S_3	0,49%
S_4	2,80%

5.2 Execução do método

Executamos o BINGRO com o intuito de obter a classificação dos reads de entrada no nível taxonômico de espécie. Uma execução de nosso método foi feita para cada conjunto de entrada S_1, S_2, S_3 e S_4. Nesta seção descrevemos em detalhes o que ocorre com os dados de entrada no fluxo de execução mostrado na Figura 4.1.

Verificamos que a execução principal do BINGRO, quando tomado como classificador de sequências metagenômicas, oferece um bom resultado, sendo capaz de classificar corretamente espécies de 50,74% a 55,27% dos reads de entrada.

As execuções usaram como base para consultas Blast a base de sequências de proteínas não redundantes *nr* baixada do NCBI em 9 de fevereiro de 2012.

Blastx inicial

A execução do Blastx realizada no passo inicial do algoritmo retorna *hits* para 95.765 reads do conjunto de entrada S_1. Ou seja, 3.209 reads, cerca de 3% da entrada, não puderam ser verificados pelo BINGRO, já que não possuem *hits* no Blast. Esses reads compõem o conjunto de saída Out_0, isto é, os elementos pertencentes ao conjunto de saída no Passo 1.

O número de *hits* diminui conforme a taxa de erros aumenta nos conjuntos restantes, chegando a 83.425 no conjunto S_4, que possui maior taxa de erros. Esse é um comportamento esperado, pois a diminuição da qualidade da sequência influencia o resultado da comparação Blast. No nosso pior caso, portanto, temos uma perda de 15% dos reads de entrada porque eles não retornam *hits* no Blast. As contagens das perdas devido à falta de *hits* podem ser vistas na Tabela 5.3.

Agrupamento

O agrupamento realizado no Passo 2 do método prepara o conjunto de reads para posterior montagem. Os agrupamentos que possuem apenas um read isolado e aqueles que agruparam mas não obtiveram contig farão parte da saída do Passo 4, no conjunto de saída Out_4. Para a entrada S_1, 20.423 reads, isto é, 20,63% da entrada são adicionados ao conjunto Out_4, dos quais 17.279 estão corretos. Em outras palavras, 84,60% dos reads do conjunto Out_4 foram classificados corretamente.

Para as execuções restantes os resultados são semelhantes, como pode ser visto na Figura 5.1. No entanto, uma variação mais expressiva pode ser vista na execução da entrada S_4. Devido à maior taxa de erro na geração dos reads desta entrada, seu conjunto de saída Out_0 armazena muitos elementos, o que ocorre com menos intensidade nos outros casos. Por este motivo, as outras saídas de S_4 são prejudicadas com falta de elementos e sua execução possui uma baixa taxa de acerto.

Montagem

Quando a montagem do Passo 3 é bem sucedida, isto é, quando existe um contig não vazio para o agrupamento, uma entrada para uma nova execução do Blastx é construída. Essa execução, realizada no Passo 5, é chamada *Blastx dos contigs* e, eventualmente, uma árvore será gerada com seus *hits*. No entanto, caso um agrupamento tenha 3 *hits* ou menos, BINGRO finaliza adicionando os reads desse agrupamento ao conjunto de saída Out_2, gerando como saída as espécies dos *hits*.

Essas exceções não se beneficiam de filogenia, mesmo porque não existe elementos suficientes para a construção das árvores. No entanto, esses casos tiram proveito da montagem, que oferece um contig para consultas Blastx montado a partir de reads

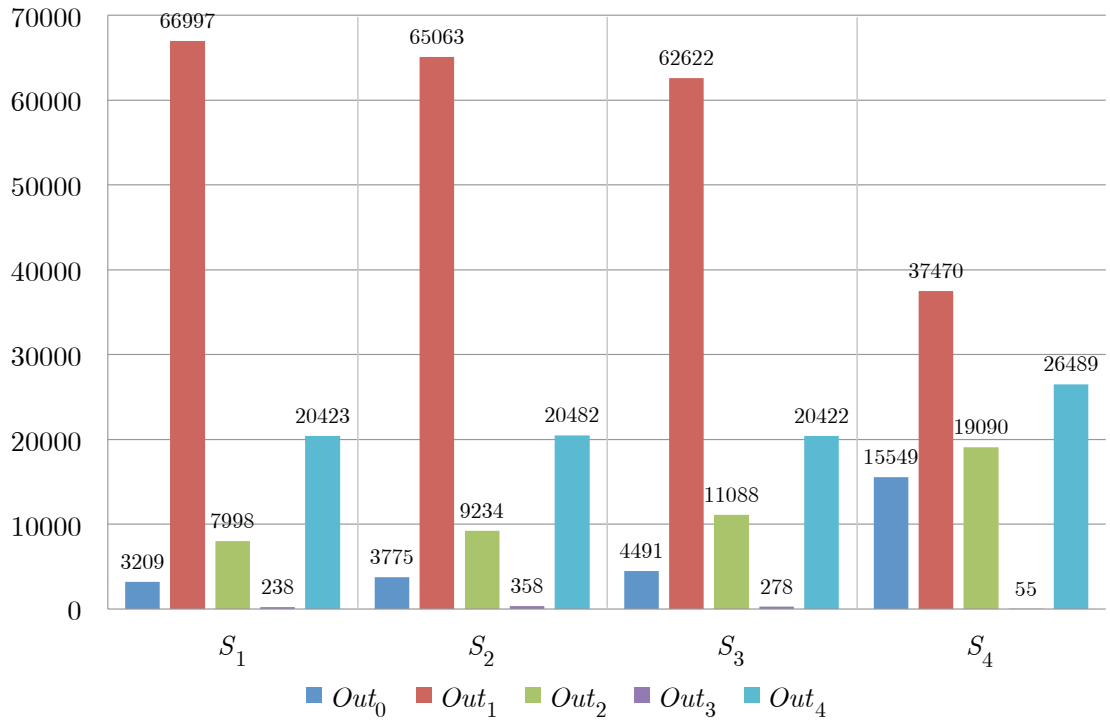


Figura 5.1: As contagens dos elementos dos conjuntos de saída para cada caso da entrada.

que compartilham informações (têm como primeiro *hit* do Blastx inicial a mesma proteína). Para o primeiro conjunto de entrada, S_1, o tamanho do conjunto Out_2 é de 7.998 reads, dos quais 7.924, ou 99,07% estão corretos.

Essa taxa de acerto para elementos do conjunto Out_2 se mantém proporcionalmente estável nos conjuntos de entrada restantes. A execução de S_2 possui 9.234 elementos em Out_2, dos quais 9.154 estão corretos. Em S_3, 11.088 elementos estão presentes em Out_2, onde 11.026 são classificados corretamente. Por fim, 19.090 elementos pertencem à saída Out_2 de S_4, sendo 18.981 corretos. Uma visão geral é oferecida pelas Figuras 5.1 e 5.2.

Filogenia

A construção da árvore é o passo final do método e é onde a maioria dos reads da entrada é adicionada no fluxo de execução.

Após a construção da árvore no Passo 6 e a seleção das folhas do Passo 7, temos a saída final do método no conjunto Out_1. Para a execução da entrada S_1, o conjunto de saída armazena 66.997 reads classificados, 24.810 ou 37,03% deles com classificação correta. Para a entrada S_2, o conjunto tem 65.063 reads, com acerto de 39,10% deles. A entrada S_3 conta com 62.622 elementos no conjunto de saída, com 38,95% dos reads corretos. Por fim, a execução da entrada S_4 produziu a saída Out_1 com 37.470 elementos, com 38,88% de acerto.

Os reads dos agrupamentos onde a construção não é bem sucedida são adicionados ao conjunto de saída Out_3. Essa saída é uma exceção que ocorre ocasionalmente e o número seu de elementos é muito pequeno para todas as execuções, variando de 55 para S_4 até 358 para S_2. A porcentagem de acerto dos elementos desse grupo é alta: 90,33%, 91,89%, 89,56% e 96,36% para as execuções de S_1, S_2, S_3 e S_4, respectivamente.

Tabela 5.3: Taxa de erro da simulação, contagem de reads sem *hits* e taxa de acerto do BINGRO para cada um dos conjuntos de entrada.

Conjunto	**Taxa de erro**	**Reads sem *hits***	**Classificação correta**
S_1	0,00%	3.209 (3,24%)	50.228 (50,7%)
S_2	0,22%	3.775 (3,81%)	52.150 (52,7%)
S_3	0,49%	4.491 (4,53%)	52.662 (53,2%)
S_4	2,80%	15.549 (15,71%)	54.706 (55,3%)

Comparação com outros métodos

Os experimentos envolveram também a execução dos softwares RAIphy, Phymm e PhymmBL, utilizando os mesmos conjuntos de dados de entrada, e seus resultados foram avaliados e comparados com os resultados do nosso método.

O software que apresenta melhores resultados é o PhymmBL, que é uma combinação

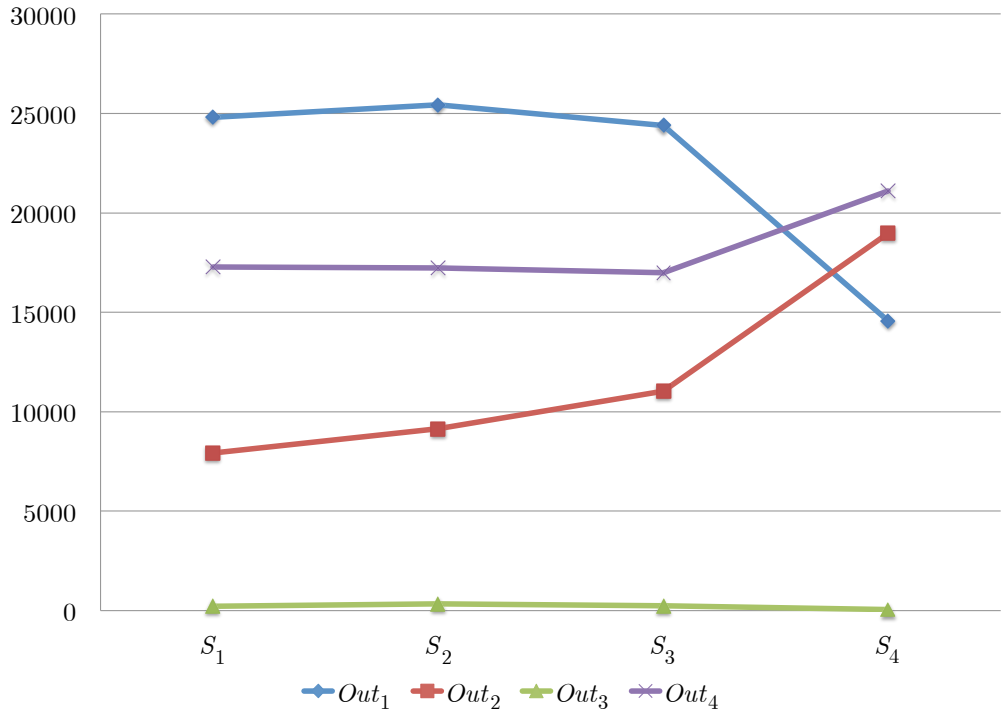


Figura 5.2: Contagem dos acertos do BINGRO em cada saída para todos os conjuntos de entrada.

do Phymm com o Blast. Para o conjunto de entrada S_1, a classificação de espécie foi correta para 94.314, ou 95,29% do reads. Apesar da boa taxa de acerto, o programa tem um tempo de execução consideravelmente alto. Em uma CPU Intel Xeon de quatro núcleos a 2,83 GHz, com 8 GB de memória, a execução para a entrada S_1 teve duração de 528 horas.

O RAIphy apresenta resultados que se assemelham aos de nosso método, mantendo taxas de acerto que vão de 57,4% para o conjunto de entrada S_1, até 45,5% para a entrada com a maior taxa de erro. O tempo de execução do RAIphy é expressivamente menor. No conjunto de entrada S_1, utilizando a mesma máquina da execução do PhymmBL, o programa finalizou as classificações dos conjuntos de entrada em uma média de 24 minutos cada.

Em relação ao tempo de execução, nosso método se posiciona entre o RAIphy e o PhymmBL. A execução do conjunto de entrada S_1 tomou 86 horas, contabilizando o tempo gasto para as duas execuções Blastx.

Uma comparação dos resultados das execuções do nosso método com RAIphy, Phymm e PhymmBL, para os mesmos 4 conjuntos de entrada, pode ser vista na Tabela 5.4.

Tabela 5.4: Comparação entre nosso método, RAIphy, Phymm e PhymmBL.

Conjunto	**Nosso método**	**RAIphy**	**Phymm**	**PhymmBL**
S_1	50.228 (50,7%)	56.848 (57,4%)	87.483 (88,4%)	94.314 (95,29%)
S_2	52.150 (52,7%)	52.676 (53,2%)	87.095 (87,9%)	94.345 (95,32%)
S_3	52.662 (53,2%)	52.224 (52,8%)	86.791 (87,7%)	94.270 (95,24%)
S_4	54.706 (55,3%)	44.997 (45,5%)	80.608 (81,4%)	94.657 (95,63%)

Os experimentos também envolveram execuções do Blastx dos contigs, que foram tomadas como método isolado e seus resultados foram analisados considerando como classificação de cada read o primeiro *hit* obtido. Essas execuções possuem a mesma limitação de perdas devido ao Blastx inicial (perdas devido a falta de *hits*). A Tabela 5.5 mostra os acertos das execuções isoladas de Blastx e suas perdas devido a falta de *hits*.

Além disso, execuções adicionais do Blastx foram consideradas após os passos de agrupamento e montagem do nosso método. A saída de todos os dados de entrada foi dada apenas pelo Passo 4 do método. Isto é, os reads da entrada passam pelo processo de agrupamento e a montagem é aplicada seguindo as mesmas regras do método. A segunda execução do Blastx é realizada com os contigs montados e os resultados são considerados pelos seus 3 primeiros *hits*. Se um read pertence a um agrupamento isolado nesse caso, a montagem não é realizada e a proteína p_i é respondida. Essas execuções obtêm um ótimo resultado, como pode ser visto na Tabela 5.5.

É importante salientar que a coluna "Acertos do Blastx" da Tabela 5.5 considera um acerto quando a espécie correta aparece em um dos 3 primeiros *hits* do Blastx. Esse critério é usado aqui para podermos comparar com a saída de nosso método. Os acertos considerando apenas o primeiro *hit*, sem os passos de agrupamento e montagem do nosso método, são mostrados na coluna "Acertos do Blastx no Passo 4" da mesma tabela. Lembramos, no entanto, que os conjuntos de teste são formados por genomas de espécies já conhecidas na base *nr*.

Tabela 5.5: Acertos para execuções do Blastx tomadas como método de classificação. Os acertos da coluna "Acertos do Blastx isolado" são considerados quando a espécie correta é o primeiro *hit* de Blastx. A coluna "Acertos do Blastx no Passo 4" considera acerto quando a espécie correta aparece em pelo menos um dos 3 primeiros *hits* de Blastx.

Entrada	Acertos do Blastx isolado	Acertos do Blastx no Passo 4	Reads perdidos
S_1	88.565 (89,48%)	91.808 (92,75%)	3.209
S_2	87.923 (88,83%)	91.121 (92,06%)	3.775
S_3	87.188 (88,09%)	90.240 (91,17%)	4.491
S_4	75.240 (76,01%)	77.191 (77,99%)	15.549

5.3 Execução como método de teste

Uma análise importante dos nossos resultados leva em consideração o conjunto de reads cuja classificação do Phymm é incorreta. Dentre todos os reads que o Phymm classifica como uma espécie diferente da espécie verdadeira, contabilizamos os reads que nosso método classificou corretamente. Desse modo, podemos oferecer o BINGRO como um guia, que serve como método de teste para softwares como o Phymm, quando se tem os resultados verdadeiros para cada read.

A análise da execução do nosso método como método auxiliar é feita em duas partes, primeiro utilizando como comparação o Phymm, e, em seguida, o PhymmBL. Cada read com classificação incorreta desses métodos foram analisados dentro do nosso fluxo de execução e, portanto, sabemos em quais conjuntos de saída, $Out_1, \ldots, Out_4$, os reads são adicionados.

Phymm

Para cada read que o Phymm classifica incorretamente, verificamos e contabilizamos onde, nas saídas do nosso método, se encontra esse read. Essas contagens nos permitem dizer que nosso método pode agir como um método auxiliar para o Phymm, já que, para o conjunto S_1, cerca de 80% dos reads estão corretos de acordo com nossa definição de acerto, isto é, a espécie correta pertence ao pequeno conjunto de espécie dado como resposta pelo método. Chamamos esse valor de *taxa de correção*. Valores

semelhantes são obtidos para os outros conjuntos de entrada, sendo a entrada S_4 a que apresenta menor taxa de correção, com 54,37%.

Tabela 5.6: Erros do Phymm que são classificados corretamente no nosso método. A última linha da tabela mostra o número de acertos do nosso método e o percentual, quando comparado ao número total de erros do Phymm.

	S_1		S_2		S_3		S_4	
Erros	11.491		11.879		12.183		18.366	
Out_1	5.486	(47,74%)	5.656	(47,61%)	5.508	(45,21%)	3.708	(20,18%)
Out_2	501	(4,35%)	565	(4,75%)	700	(5,74%)	2.163	(11,77%)
Out_3	30	(0,26%)	32	(0,27%)	27	(0,22%)	11	(0,05%)
Out_4	2.453	(21,34%)	2.543	(21,40%)	2.493	(20,46%)	4.105	(22,35%)
Soma	8.470	(73,70%)	8.796	(74,04%)	8.728	(71,64%)	9.987	(54,37%)

PhymmBL

Para o primeiro conjunto de entrada, S_1, o PhymmBL obteve 4.660 classificações incorretas. Desses elementos, 3.743, ou 80,20%, foram classificados corretamente pelo nosso método. No pior caso, que é o conjunto de entrada S_4, o PhymmBL não teve mudança significativa no desempenho, errando 4.317 vezes. Uma taxa de correção de 56,78% é obtida pelo nosso método. Os valores para os outros conjuntos de entrada, bem como o conjunto de saída em que os erros se encaixam podem ser vistos na Tabela 5.7.

Como tanto o nosso método quanto o PhymmBL utilizam Blast para resolver o problema de classificação, a classificação correta de grande parte dos erros do PhymmBL se deve ao fato de que a nossa resposta para os conjuntos de saída $Out_1, \ldots, Out_4$, é, na verdade, um conjunto de espécies próximas às folhas das árvores, ou os primeiros *hits* do Blastx dos contigs, dependendo do caso de saída. No entanto, como veremos a seguir, o a resposta como um conjunto de espécies não é exclusivamente o fator determinante para a obtenção dessas taxas de correção.

Tabela 5.7: Erros do PhymmBL que são classificados corretamente no nosso método. A última linha da tabela mostra o número de acertos do nosso método e o percentual, quando comparado ao número total de erros do PhymmBL.

	S_1		S_2		S_3		S_4	
Erros	4.660		4.629		4.704		4.317	
Out_1	2.812	(60,34%)	2.780	(60,05%)	2.745	(58,35%)	1.385	(32,08%)
Out_2	17	(0,36%)	20	(0,43%)	28	(0,59%)	60	(1,38%)
Out_3	12	(0,25%)	6	(0,12%)	12	(0,25%)	2	(0,04%)
Out_4	902	(19,25%)	884	(19,09%)	838	(17,81%)	1.005	(23,28%)
Soma	3.743	(80,20%)	3.690	(79,69%)	3.623	(77,00%)	2.452	(56,78%)

Correções

A taxa de correção que nosso método consegue atingir dentro dos erros do Phymm e do PhymmBL pode ser explicada, em parte, pelo conjunto de espécies que é atribuído a cada read da nossa classificação. Para garantir que esse fato isolado não seja o único responsável pela nossa taxa de correção, contabilizamos, dentre as classificações incorretas dos próprios Phymm e PhymmBL, quantas classificações estariam corretas se analisássemos o nosso pequeno conjunto de espécies de saída.

Em outras palavras, analisamos os reads classificados incorretamente pelos softwares Phymm. Para cada read, se a classificação dada como incorreta no Phymm pertence ao conjunto de saída do nosso método (que é um conjunto de espécies), contabilizamos para esse read um *acerto fictício*. O acerto fictício é um valor que indica se o nosso método auxilia o Phymm apenas por que responde como saída um conjunto de espécies próximas. Pelos valores obtidos dessas contagens, vemos que isso não é verdade.

Na execução da entrada S_1 com o Phymm, dos 11.491 reads classificados incorretamente, 4.379 obtiveram acertos fictícios. Isto é, 4.379 reads estariam corretos se a resposta do Phymm fosse exatamente nosso conjunto de espécies da saída. Como nosso método classifica 8.470 desses reads corretamente, vemos que o auxílio que nosso método oferece não é dependente exclusivamente do fato de respondermos como saída um conjunto de espécies.

As execuções dos outros conjuntos de entrada oferecem acertos fictícios semelhantes

a da entrada S_1. Para o PhymmBL, que já se aproveita de resultados do Blast, os acertos fictícios foram menores, mas ainda proporcionais ao número de erros. Na execução de S_1, que produziu 4.660 reads classificados incorretamente, 2.409 foram acertos fictícios, isto é, teriam sido classificados corretamente se oferecesse como saída o nosso conjunto de saída. Como a soma dos nossos acertos é 3.743, concluímos que nosso método auxilia o PhymmBL. As contagens de acertos fictícios e classificações incorretas do Phymm, bem como o número de acertos do nosso método, podem ser vistos na Tabela 5.8. Os mesmos valores para PhymmBL podem ser vistos na Tabela 5.9.

Tabela 5.8: Contagens de acertos fictícios para cada conjunto de entrada nas execuções do Phymm.

Entrada	Erros do Phymm	Acertos fictícios	Acertos do nosso método
S_1	11.491	4.379	8.470
S_2	11.879	4.450	8.796
S_3	12.183	4.362	8.728
S_4	18.366	2.902	9.987

Tabela 5.9: Contagens de acertos fictícios para cada conjunto de entrada nas execuções do PhymmBL.

Entrada	Erros do PhymmBL	Acertos fictícios	Acertos do nosso método
S_1	4.660	2.409	3.743
S_2	4.629	2.427	3.690
S_3	4.704	2.410	3.623
S_4	4.317	1.716	2.452

5.4 Comparação com outras ferramentas

Os experimentos realizados neste estudo e descritos neste capítulo nos permitiram observar que o nosso método possui condições de ser entendido como um método classificador de sequências metagenômicas isolado, já que, com os mesmos dados de entrada, foi possível produzir classificações com taxas de acerto que se encontram entre os resultados do RAIphy e do Phymm.

Observamos que, quando temos a maior taxa de erro no conjunto de entrada S_4, a porcentagem geral de acertos do nosso método é maior, conforme mostra a Tabela 5.3. A razão pela qual isso acontece é que, como mostra a Figura 5.1, o número de elementos no conjunto de saída Out_1 para a entrada S_4 diminuem consideravelmente, fazendo com que os dados se distribuam nas saídas restantes. Os acertos são baixos na saída Out_1 (38,88%), mas bem maiores nas saídas restantes, com 99,42%, 96,36% e 84,60% de acerto nas saídas Out_2, Out_3 e Out_4, respectivamente. Portanto, mesmo com uma taxa de erro alta, podemos ter uma quantidade considerável de acertos.

A execução do nosso método como método auxiliar também apresentou resultados favoráveis. Foi possível "corrigir" grande parte dos erros obtidos pelos softwares Phymm e PhymmBL. Obtemos uma taxa de correção que vai de 54,37% a 80,20% para esses softwares.

Confirmando os resultados de Brady e Salzberg [4], nossos experimentos mostram que o Blast é, isoladamente, o melhor método de classificação de sequências para rótulos de espécie. No entanto, existe a necessidade de as espécies já serem conhecidas no banco de dados de consulta. Sabemos que nenhum método de classificação pode classificar corretamente o rótulo de espécie para um read de um organismo cujo genoma ainda não foi sequenciado [4].

Como nosso método não depende exclusivamente dos resultados Blast, contando também com agrupamento, montagem e filogenia, podemos ter melhores resultados quando utilizados dados metagenômicos reais. Além disso, o fato de respondermos um conjunto de espécie para cada classificação pode dar pistas sobre a espécie verdadeira, caso ela ainda não tenha sido sequenciada.

5.5 Parâmetros

Nossos experimentos foram realizados executando o método com os parâmetros descritos a seguir.

Ambas as execuções Blastx foram feitas com valor de E-value 10^{-5} sobre a base *nr*. O parâmetro `num_descriptions` foi definido como 20 e a execução é definida para

usar 8 threads através da opção `num_threads`. Por fim, um limite de retorno dos *hits* é definido como 20 usando a opção `num_alignments`.

Na seção de montagem do algoritmo, o programa de montagem CAP3 foi executado com seus parâmetros padrão.

Na seção de filogenia, selecionamos um valor máximo de $\kappa = 10$ primeiros *hits* para construir o conjunto H_i. Além disso, um limite máximo de 4 folhas é considerado ao construir o conjunto de saída.

Capítulo 6

Conclusão

Novas tecnologias de sequenciamento a baixo custo têm permitido a obtenção de informações biológicas diretamente de comunidades microbianas em seus habitats naturais. Dados de sequências obtidos diretamente do sequenciamento ambiental são chamados de metagenoma, e o estudo desses dados é chamado de metagenômica.

A metagenômica é uma derivação da genômica microbiana original, com a diferença principal de que na primeira não precisamos obter culturas puras para realizar o sequenciamento. Ela pode ajudar a estabelecer hipóteses a respeito de interações entre membros de comunidades e de interações com seus ambientes. Assim, a metagenômica está sendo considerada a tecnologia base para a compreensão da evolução de organismos e ecossistemas microbianos.

Basicamente um projeto de metagenômica possui as seguintes etapas. O processo inicia com a amostragem e coleção de dados, seguidos pela extração de DNA, sequenciamento, processamento das sequências, também chamados de *reads*, e montagem. Os reads ou contigs participam do processo de predição de genes e anotação e, por fim, um passo de classificação é aplicado.

A classificação consiste na atribuição de uma espécie, ou de um grupo taxonômico de mais alta ordem, a cada read da amostra, na tentativa de caracterizar o ecossistema em termos de sua composição e consequentemente em termos das interações entre os organismos presentes.

Neste trabalho apresentamos o estudo de alguns métodos para o problema da classificação de sequências metagenômicas, além de um método alternativo, baseado em comparação de sequências, montagem e filogenia. A principal característica do nosso método é o fato de que, na fase de montagem, apenas reads que compartilham alguma similaridade com proteínas conhecidas são agrupados.

Nosso método apresenta como resultado não apenas uma única espécie, mas sim um conjunto de espécies filogeneticamente próximas, a partir de uma árvore filogenética, cujos objetos são a proteína resultante da montagem dos reads agrupados e as proteínas similares a ela.

Outra característica importante do método proposto é o fato de que suas etapas podem ser executadas de forma independente, no sentido de que podemos escolher outros programas ou abordagens para cada uma das etapas. Por exemplo, o Blast poderia ser substituído nas etapas em que foi usado. Montadores diferentes poderiam ser utilizados. A construção da filogenia poderia também ser feita por meio de outro programa.

Os experimentos foram feitos com o uso de dados metagenômicos simulados, contendo reads com taxas controladas de erro, e mostraram que nosso método conseguiu, em alguns casos, superar outros métodos.

Além disso, nosso método mostrou-se útil, não somente aplicável diretamente na classificação de um conjunto de sequências, mas também como método de teste de outros programa, classificando corretamente sequências erroneamente classificadas por eles, obviamente quando se tem esses resultados verdadeiros.

Resultados preliminares deste trabalho foram publicados em [1], em 2011. Algumas sugestões recebidas durante a apresentação desse trabalho no BSB2011 foram incorporadas. Em particular, o tratamento das saídas foi alterado, tornando o fluxograma mais bem organizado.

Trabalhos futuros

O método proposto neste trabalho foi implementado de maneira que cada agrupamento G_i foi armazenado em um diretório contendo diferentes arquivos. Entre

eles, um arquivo Fasta representando os reads de tal agrupamento. Todas as manipulações foram realizadas dentro de cada diretório, incluindo a geração dos contigs e a construção da árvore.

Esta abordagem, apesar de eficaz, consome um tempo considerável de exeução fazendo com que o sistema operacional realize operações de entrada e saída, manipulando um número muito grande de arquivos. Uma opção seria a retirada de algumas dessas tarefas realizadas em arquivos e transferi-las para execuções em memória. Poderíamos, assim, melhorar consideravelmente o tempo das nossas execuções, reduzindo, por exemplo, a execução de 53 horas tomadas para um conjunto de entrada descrita no Capítulo 5.

Algumas etapas do nosso método podem ser avaliadas com diferentes opções. A montagem, por exemplo, realizada com CAP3 [17], sofre de resultados não desejados, como contigs nulos. Apesar de isso ser característica da metagenômica, onde a montagem é intrinsecamente prejudicada, podemos avaliar a eficiência de montadores diferentes. O mesmo vale para o agrupamento inicial dos read, onde algum tipo de filtragem prévia pode ser aplicada.

Referências Bibliográficas

[1] N. Almeida, H. Asseiss, and J.C. Setubal. Using assembly and phylogeny to classify metagenomic sequences. *6th Brazilian Symposium on Bioinformatics*, pages 65–68, August 2011.

[2] S.F. Altschul, T.L. Madden, A.A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D.J. Lipman. Gapped Blast and Psi-Blast: a new generation of protein database search programs. *Nucleic Acid Research*, 25:3389–3402, 1997.

[3] S. Batzoglou, D.B. Jaffe, K. Stanley, J. Butler, S. Gnerre, E. Mauceli, B. Berger, J.P. Mesirov, and E.S. Lander. ARACHNE: a whole-genome shotgun assembler. *Genome research*, 12(1):177–189, January 2002.

[4] A. Brady and S.L. Salzberg. Phymm and PhymmBL: metagenomic phylogenetic classification with interpolated Markov models. *Nature Methods*, 6(9):673–676, August 2009.

[5] J.M. Brulc, D.A. Antonopoulos, M. Berg E. Miller, M.K. Wilson, A.C. Yannarell, E.A. Dinsdale, R.E. Edwards, E.D. Frank, J.B. Emerson, P. Wacklin, P.M. Coutinho, B. Henrissat, K.E. Nelson, and B.A. White. Gene-centric metagenomics of the fiber-adherent bovine rumen microbiome reveals forage specific glycoside hydrolases. *Proceedings of the National Academy of Sciences of the United States of America*, 106(6):1948–1953, February 2009.

[6] J. Castresana. Selection of Conserved Blocks from Multiple Alignments for Their Use in Phylogenetic Analysis. *Molecular Biology and Evolution*, 17(4):540–552, April 2000.

[7] C.K. Chan, A. Hsu, S. Halgamuge, and S.L. Tang. Binning sequences using very sparse labels within a metagenome. *BMC Bioinformatics*, 9(1):215+, April 2008.

[8] M.O. Dayhoff, R.M. Schwartz, and B.C. Orcutt. A model of evolutionary change in proteins. *Atlas of protein sequence and structure*, 5(suppl 3):345–351, 1978.

[9] R.C. Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic acids research*, 32(5):1792–1797, March 2004.

[10] J. Felsenstein. Evolutionary trees from DNA sequences: a maximum likelihood approach. *Journal of Molecular Evolution*, 17(6):368–376, 1981.

[11] J. Felsenstein. PHYLIP - Phylogeny Inference Package (Version 3.2). *Cladistics*, 5:164–166, 1989.

[12] R.D. Finn, J. Mistry, B. Schuster-Bockler, S. Griffiths-Jones, V. Hollich, T. Lassmann, S. Moxon, M. Marshall, A. Khanna, R. Durbin, S.R. Eddy, E.L.L. Sonnhammer, and A. Bateman. Pfam: clans, web tools and services. *Nucleic Acids Research*, Database Issue 34:D247–D251, 2006.

[13] N. Goldman. Phylogenetic information and experimental design in molecular systematics. *Proc Biol Sci*, pages 1779–1786, 1998.

[14] T. Harkins and T. Jarvie. Metagenomics analysis using the genome sequencer flx system. *Nature Methods*, 4(6), June 2007.

[15] M. Hess, A. Sczyrba, R. Egan, T.W. Kim, H. Chokhawala, G. Schroth, S. Luo, D.S. Clark, F. Chen, T. Zhang, R.I. Mackie, L.A. Pennacchio, S.G. Tringe, A. Visel, T. Woyke, Z. Wang, and E.M. Rubin. Metagenomic Discovery of Biomass-Degrading Genes and Genomes from Cow Rumen. *Science*, 331(6016):463–467, January 2011.

[16] K. Hoff. The effect of sequencing errors on metagenomic gene prediction. *BMC Genomics*, 10(1):520+, November 2009.

[17] X. Huang and A. Madan. CAP3: A DNA Sequence Assembly Program. *Genome Research*, 9(9):868–877, September 1999.

[18] D.H. Huson, A.F. Auch, J. Qi, and S.C. Schuster. MEGAN analysis of metagenomic data. *Genome research*, 17(3):377–386, March 2007.

[19] D.H. Huson, S. Mitra, H.J. Ruscheweyh, N. Weber, and S.C. Schuster. Integrative analysis of environmental sequences using MEGAN4. *Genome Research*, 21(9):1552–1560, September 2011.

[20] D.B. Jaffe, J. Butler, S. Gnerre, E. Mauceli, K. Lindblad-Toh, J.P. Mesirov, M.C. Zody, and E.S. Lander. Whole-genome sequence assembly for mammalian genomes: Arachne 2. *Genome research*, 13(1):91–96, January 2003.

[21] T. Junier and E.M. Zdobnov. The Newick utilities: high-throughput phylogenetic tree processing in the UNIX shell. *Bioinformatics*, 26(13):1669–1670, July 2010.

[22] L. Krause, N.N. Diaz, A. Goesmann, S. Kelley, T.W. Nattkemper, F. Rohwer, R.A. Edwards, and J. Stoye. Phylogenetic classification of short environmental DNA fragments. *Nucleic acids research*, 36(7):2230–2239, April 2008.

[23] L. Krause, N.N. Diaz, A. Goesmann, S. Kelley, T.W. Nattkemper, F. Rohwer, R.A. Edwards, and J. Stoye. Phylogenetic classification of short environmental DNA fragments. *Nucleic Acids Research*, 36(7):2230–2239, April 2008.

[24] V. Kunin, A. Copeland, A. Lapidus, K. Mavromatis, and P. Hugenholtz. A Bioinformatician's Guide to Metagenomics. *Microbiol. Mol. Biol. Rev.*, 72(4):557–578, December 2008.

[25] C. Manichanh, C.E. Chapple, L. Frangeul, K. Gloux, R. Guigo, and J. Dore. A comparison of random sequence reads versus 16S rDNA sequences for estimating the biodiversity of a metagenomic library. *Nucl. Acids Res.*, 36(16):5180–5188, September 2008.

[26] M. Margulies, M. Egholm, W. E. Altman, S. Attiya, J.S. Bader, L.A. Bemben, J. Berka, M.S. Braverman, Y.J. Chen, Z. Chen, S. B. Dewell, L. Du, J.M. Fierro, X.V. Gomes, B.C. Godwin, W. He, S. Helgesen, C. H. Ho, G.P. Irzyk, S.C. Jando, M.L.I. Alenquer, T.P. Jarvie, K.B. Jirage, J.B. Kim, J.R. Knight, J.R. Lanza, J.H. Leamon, S.M. Lefkowitz, M. Lei, J. Li, K.L. Lohman, H. Lu, V.B. Makhijani, K.E. McDade, M.P. McKenna, E.W. Myers, E. Nickerson, J.R. Nobile, R. Plant, B.P. Puc, M.T. Ronan, G.T. Roth, G.J. Sarkis, J.F. Simons, J.W. Simpson, M. Srinivasan, K.R. Tartaro, A. Tomasz, K.A. Vogt, G.A. Volkmer, S.H. Wang, Y. Wang, M.P. Weiner, P. Yu, R.F. Begley, and J.M. Rothberg. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 437(7057):376–380, July 2005.

[27] V.M. Markowitz. Microbial genome data resources. *Current Opinion in Biotechnology*, 18(3):267–272, June 2007.

[28] K. Mavromatis, N. Ivanova, K. Barry, H. Shapiro, E. Goltsman, A.C. McHardy, I. Rigoutsos, A. Salamov, F. Korzeniewski, M. Land, A. Lapidus, I. Grigoriev, P. Richardson, P. Hugenholtz, and N.C. Kyrpides. Use of simulated data sets to evaluate the fidelity of metagenomic processing methods. *Nature Methods*, 4(6):495–500, April 2007.

[29] A.C. McHardy, H.G. Martin, A. Tsirigos, P. Hugenholtz, and I. Rigoutsos. Accurate phylogenetic classification of variable-length DNA fragments. *Nature Methods*, 4(1):63–72, December 2006.

[30] O. Nalbantoglu, S. Way, S. Hinrichs, and K. Sayood. RAIphy: Phylogenetic classification of metagenomics samples using iterative refinement of relative abundance index profiles. *BMC Bioinformatics*, 12(1):41+, 2011.

[31] J.F. Petrosino, S. Highlander, R.A. Luna, R.A. Gibbs, and J. Versalovic. Metagenomic pyrosequencing and microbial identification. *Clin Chem*, 55(5):856–866, May 2009.

[32] H.N. Poinar, C. Schwarz, J. Qi, B. Shapiro, R.D.E. MacPhee, B. Buigues, A. Tikhonov, D.H. Huson, L.P. Tomsho, A. Auch, M. Rampp, W. Miller, and S.C. Schuster. Metagenomics to Paleogenomics: Large-Scale Sequencing of Mammoth DNA. *Science*, 311(5759):392–394, January 2006.

[33] K.D. Pruitt, T. Tatusova, and D.R. Maglott. NCBI reference sequences (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins. *Nucleic acids research*, 35(Database issue):D61–D65, January 2007.

[34] D.C. Richter, F. Ott, A.F. Auch, R. Schmid, and D.H. Huson. MetaSim: A Sequencing Simulator for Genomics and Metagenomics. *PLoS ONE*, 3(10):e3373+, October 2008.

[35] S.L. Salzberg, A.L. Delcher, S. Kasif, and O. White. Microbial gene identification using interpolated Markov models. *Nucleic Acids Research*, 26(2):544–548, January 1998.

[36] S.L. Salzberg, M. Pertea, A.L. Delcher, M.J. Gardner, and H. Tettelin. Interpolated Markov models for eukaryotic gene finding. *Genomics*, 59(1):24–31, July 1999.

[37] F. Sanger, S. Nicklen, and A.R. Coulson. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America*, 74(12):5463–5467, December 1977.

[38] P.D. Schloss and J. Handelsman. A statistical toolbox for metagenomics: assessing functional diversity in microbial communities. *BMC bioinformatics*, 9(1):34+, January 2008.

[39] J.D. Selengut, D.H. Haft, T. Davidsen, A. Ganapathy, M. Gwinn-Giglio, W.C. Nelson, A.R. Richter, and O. White. TIGRFAMs and Genome Properties: tools for the assignment of molecular function and biological process in prokaryotic genomes. *Nucleic acids research*, 35(Database issue), January 2007.

[40] R. Seshadri, S.A. Kravitz, L. Smarr, P. Gilna, and M. Frazier. CAMERA: A Community Resource for Metagenomics. *PLoS Biol*, 5(3):e75+, March 2007.

[41] J.C. Setubal and J. Meidanis. *Introduction to Computational Molecular Biology.* PWS Publishing, January 1997.

[42] A. Stamatakis. RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, 22(21):2688–2690, November 2006.

[43] L.D. Stein. Genome annotation: from sequence to biology. *Nature Reviews Genetics*, 2(7):493–503, July 2001.

[44] L. Tasse, J. Bercovici, S. Pizzut-Serin, P. Robe, J. Tap, C. Klopp, B.L. Cantarel, P.M. Coutinho, B. Henrissat, M. Leclerc, J. Doré, P. Monsan, M. Remaud-Simeon, and G. Potocki-Veronese. Functional metagenomics to mine the human gut microbiome for dietary fiber catabolic enzymes. *Genome Research*, 20(11):1605–1612, November 2010.

[45] J.D. Thompson, D.G. Higgins, and T.J. Gibson. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic acids research*, 22(22):4673–4680, November 1994.

[46] C. Tuffley and M. Steel. Links between maximum likelihood and maximum parsimony under a simple model of site substitution. *Bulletin of Mathematical Biology*, 59:581–607, 1997. 10.1007/BF02459467.

[47] G.W. Tyson, J Chapman, P Hugenholtz, E.E. Allen, R.J. Ram, P.M. Richardson, V.V. Solovyev, E.M. Rubin, D.S. Rokhsar, and J.F. Banfield. Community structure and metabolism through reconstruction of microbial genomes from the environment. *Nature*, 428(6978):37–43, March 2004.

[48] J.C. Venter, K. Remington, J.F. Heidelberg, A.L. Halpern, D. Rusch, J. A. Eisen, D. Wu, I. Paulsen, K.E. Nelson, W. Nelson, D.E. Fouts, S. Levy, A.H. Knap, M.W. Lomas, K. Nealson, O. White, J. Peterson, J. Hoffman, R. Parsons, H. Baden-Tillson, C. Pfannkoch, Y.H. Rogers, and H.O. Smith. Environmental Genome Shotgun Sequencing of the Sargasso Sea. *Science*, 304(5667):66–74, April 2004.

[49] F. Warnecke and P. Hugenholtz. Building on basic metagenomics with complementary technologies. *Genome Biology*, 8(12):231+, December 2007.

[50] J.C. Wooley, A. Godzik, and I. Friedberg. A Primer on Metagenomics. *PLoS Comput Biol*, 6(2):e1000667+, February 2010.

[51] S. Yooseph, G. Sutton, D.B. Rusch, A.L. Halpern, S.J. Williamson, K. Remington, J.A. Eisen, K.B. Heidelberg, G. Manning, W. Li, L. Jaroszewski, P. Cieplak, C.S. Miller, H. Li, S.T. Mashiyama, M.P. Joachimiak, C. van Belle, J.-Marc Chandonia, D.A. Soergel, Y. Zhai, K. Natarajan, S. Lee, B.J. Raphael, V. Bafna, R. Friedman, S.E. Brenner, A. Godzik, D. Eisenberg, J.E. Dixon, S.S. Taylor, R.L. Strausberg, M. Frazier, and J.C. Venter. The Sorcerer II Global Ocean Sampling Expedition: Expanding the Universe of Protein Families. *PLoS Biol*, 5(3):e16+, March 2007.

[52] D.R. Zerbino and E. Birney. Velvet: algorithms for de novo short read assembly using de Bruijn graphs. *Genome research*, 18(5):821–829, May 2008.