MINISTÉRIO DA SAÚDE
FUNDAÇÃO OSWALDO CRUZ
INSTITUTO OSWALDO CRUZ

Mestrado em Programa de Pós-Graduação em Biologia Computacional e Sistemas

GENOMA DE *RHODNIUS PROLIXUS*: PREDIÇÃO GÊNICA, CONCILIAÇÃO COM VERSÕES ANTERIORES E DISPONIBILIZAÇÃO EM NAVEGADOR WEB

NICOLAS DA MATTA FREIRE ARAUJO

Rio de Janeiro
Março de 2022

Ministério da Saúde

FIOCRUZ
Fundação Oswaldo Cruz

INSTITUTO OSWALDO CRUZ
Programa de Pós-Graduação em Biologia Computacional e Sistemas

NICOLAS DA MATTA FREIRE ARAUJO

Genoma de *Rhodnius prolixus*: predição gênica, conciliação com versões anteriores e disponibilização em navegador web.

Dissertação apresentada ao Instituto Oswaldo Cruz como parte dos requisitos para obtenção do título de Mestre em Biologia Computacional e Sistemas

Orientador: Prof. Dr. Rafael Dias Mesquita

RIO DE JANEIRO
Março de 2022

da Matta Freire Araujo, Nicolas.

Genoma de *Rhodnius prolixus*: predição gênica, conciliação com versões anteriores e disponibilização em navegador web. / Nicolas da Matta Freire Araujo. - Rio de Janeiro, 2022.
117 f.

Dissertação (Mestrado) – Instituto Oswaldo Cruz, Pós-Graduação em Biologia Computacional e Sistemas, 2022.

Orientador: Rafael Dias Mesquita.

Bibliografia: f. 66-80

1. predição gênica. 2. rhodnius prolixus. 3. doença de chagas. 4. bioinformática. I. Título.

Elaborado pelo Sistema de Geração Automática de Ficha Catalográfica da Biblioteca de Manguinhos/Icict/Fiocruz com os dados fornecidos pelo(a) autor(a), sob a responsabilidade de Igor Falce Dias de Lima - CRB-7/6930.

Ministério da Saúde

FIOCRUZ
Fundação Oswaldo Cruz

INSTITUTO OSWALDO CRUZ

Programa de Pós-Graduação em Biologia Computacional e Sistemas

NICOLAS DA MATTA FREIRE ARAUJO

GENOMA DE *RHODNIUS PROLIXUS*: PREDIÇÃO GÊNICA, CONCILIAÇÃO COM VERSÕES ANTERIORES E DISPONIBILIZAÇÃO EM NAVEGADOR WEB

Orientador: Prof. Dr. Rafael Dias Mesquita

Aprovada em: 29/03/2022

EXAMINADORES:

Prof. Dr. Marcos Paulo Catanho de Souza
Prof. Dr. Fabio Passetti
Prof. Dr. Pedro Lagerblad de Oliveira
Prof. Dr. Antonio Basilio de Miranda
Prof. Dr. David Majerowicz

Rio de Janeiro, 29 de março de 2022

AGRADECIMENTOS

Agradeço à minha avó Edina, que apesar de não estar mais comigo, sempre me incentivou e se orgulhou de todas as minhas conquistas.

Agradeço à minha mãe Claudia pelo apoio incondicional e por ter feito todo o possível para que eu pudesse me dedicar aos meus estudos.

Agradeço ao meu padrinho Alexandre, que sempre me aconselhou e me estimulou a ver o mundo de outras formas.

Agradeço aos meus tios Marcello e Emília, primos e madrinha, por todo o apoio e momentos de descontração.

Agradeço à minha namorada Amanda, por me dar suporte emocional e motivação principalmente nos momentos em que duvidei das minhas capacidades.

Agradeço aos meus amigos Tiago e Guilherme, por estarem comigo e me divertirem nos momentos mais estressantes.

Agradeço aos meus colegas de laboratório, por terem me acolhido, me auxiliado quando precisei e pelos papos divertidos durante o almoço.

Agradeço ao professor Rafael, por toda orientação, paciência, explicações, conversas e ajuda que foram essenciais para que eu pudesse desenvolver meu projeto.

Agradeço ao Instituto Oswaldo Cruz e ao seu corpo docente, por todo conhecimento oferecido que foi essencial para que eu evoluísse como profissional.

Agradeço à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo auxílio financeiro que contribuiu para o desenvolvimento da minha dissertação.

Ministério da Saúde

FIOCRUZ
Fundação Oswaldo Cruz

INSTITUTO OSWALDO CRUZ

Genoma de *Rhodnius prolixus*: predição gênica, conciliação com versões anteriores e disponibilização em navegador web

RESUMO

DISSERTAÇÃO DE MESTRADO EM BIOLOGIA COMPUTACIONAL E SISTEMAS

Nicolas da Matta Freire Araujo

A doença de Chagas é uma doença tropical negligenciada que somente possui tratamento paliativo, logo, deve ser contida por medidas de urbanização e controle vetorial. O protozoário *Trypanosoma cruzi*, causador da doença, é transmitido por triatomíneos dos gêneros *Triatoma*, *Panstrongylus* e *Rhodnius*. O *Rhodnius prolixus* é um importante vetor na América Latina, sendo o primeiro triatomíneo a ter seu genoma sequenciado e analisado por um grupo internacional. Entretanto, a versão de montagem do seu genoma mais atual (Hi-C) não possui predição gênica, além de que existem genes preditos exclusivamente em versões anteriores. A técnica Hi-C permite utilizar o mapeamento físico da cromatina para guiar a montagem gerando um genoma de maior qualidade. Logo, se faz necessário uma nova predição gênica na montagem Hi-C juntamente com a conciliação com as predições anteriores, além da disponibilização desses dados em um navegador, para visualização e exploração pela comunidade científica. Portanto, foi feita a predição gênica (P13) da versão mais atual de montagem do genoma utilizando o software AUGUSTUS, que acabou identificando 15.181 transcritos codificadores de proteínas e, alcançou uma completude de 92,7%, sendo a maior dentre as predições de *R. prolixus* consideradas. Em seguida, os genes antigos passaram por filtros para a remoção de sequências com bases indefinidas, redundância e quimeras, totalizando 13.840 genes codificadores de proteína e 1.505 não-codificadores de proteínas. Os não-codificadores foram alinhados contra o genoma Hi-C utilizando tanto o programa Sim4 quanto o Exonerate, destes, apenas 345 genes alinharam em regiões sem predição gênica. Já os codificadores foram utilizados para enriquecer a predição P13 através de um *script* desenvolvido para fazer a conciliação tanto de genes preditos como de genes de transcriptoma. A conciliação com genes de transcriptoma e com genes preditos antigos resultou na predição P15 com 17.500 proteínas com completude de 93,2% sendo a predição de maior qualidade para *R. prolixus*. A P15 juntamente com as montagens de genoma, as predições antigas e dados de RNAseq foram disponibilizados no navegador de genomas JBrowse, hospedado em um servidor do Laboratório de Bioinformática do Instituto de Química da UFRJ. Dessa maneira, a disponibilização de todos esses dados navegáveis, poderá fomentar os estudos biológicos e de controle vetorial com o inseto. Além de possibilitar estudos comparativos com espécies de triatomíneos que venham a ocupar seu nicho biológico. Por fim, o *script* aqui desenvolvido também pode ser usado para a conciliação de genes de outras espécies.

Ministério da Saúde

FIOCRUZ
Fundação Oswaldo Cruz

INSTITUTO OSWALDO CRUZ

***Rhodnius prolixus* genome: gene prediction, conciliation with previous versions and availability in the web browser**

ABSTRACT

MASTER DISSERTATION IN COMPUTATIONAL BIOLOGY AND SYSTEMS

Nicolas da Matta Freire Araujo

Chagas disease is a neglected tropical disease that only has palliative treatment, so it must be contained by urbanization and vector control measures. The protozoan *Trypanosoma cruzi*, which causes the disease, is transmitted by triatomines of the genera *Triatoma*, *Panstrongylus* and *Rhodnius*. *Rhodnius prolixus* is an important vector in Latin America, being the first triatomine to have its genome sequenced and analyzed by an international group. However, the most current assembly version of its genome (Hi-C) has no genetic prediction, and there are genes predicted exclusively in previous versions. The Hi-C technique allows the use of physical chromatin mapping to guide assembly, generating a higher quality genome. Therefore, a new genetic prediction in the Hi-C assembly is necessary along with the conciliation with the previous predictions, in addition to the availability of these data in a browser, for visualization and exploration by the scientific community. Therefore, the gene prediction (P13) of the most current version of genome assembly was performed using the AUGUSTUS software, which ended up identifying 15,181 protein-coding transcripts and reached a completeness of 92.7%, being the highest among the predictions of *R. prolixus* considered. Then, the old genes passed through filters to remove sequences with undefined bases, redundancy and chimeras, totaling 13,840 protein-coding genes and 1,505 non-protein-coding genes. Non-coding were aligned against the Hi-C genome using both the Sim4 and Exonerate programs, of which only 345 genes were aligned in regions without gene prediction. The coding were used to enrich the P13 prediction through a script developed to reconcile both predicted genes and transcriptome genes. Reconciliation with transcriptome genes and old predicted genes resulted in P15 prediction with 17,500 proteins with 93.2% completeness being the highest quality prediction for *R. prolixus*. P15 along with genome assemblies, old predictions and RNAseq data were made available in the JBrowse genome browser, hosted on a server at the Bioinformatics Laboratory of the UFRJ Chemistry Institute. In this way, the availability of all these navigable data will be able to promote biological studies and vector control with the insect. In addition to enabling comparative studies with triatomine species that come to occupy their biological niche. Finally, the script developed here can also be used to reconcile genes from other species.

ÍNDICE

ÍNDICE DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS E ABREVIATURAS

aa	aminoácido
BAM	formato binário de mapa de alinhamento de sequência
BLAST	ferramenta básica de busca de alinhamento local
CDS	sequência codificador
CDC	Centro de Controle e Prevenção de Doenças
cDNA	ácido desoxirribonucleico complementar
DFAM	banco de dados de famílias de elementos repetitivos do DNA
DNA	ácido desoxirribonucleico
EMBL	formato do Laboratório Europeu de Biologia Molecular
EST	marcador de sequência genética
GFF	formato de características gerais
Hi-C	técnica de sequenciamento para capturar a conformação da cromatina
HSSP	estrutura secundária derivada de homologia de proteínas
INCT	Instituto Nacional de Ciência e Tecnologia
l-mer	pedaço de uma sequência de tamanho l
miRNA	micro ácido ribonucleico
mRNA	ácido ribonucleico mensageiro
NCBI	Centro Nacional de Informações sobre Biotecnologia
nt	nucleotídeo
PCR	reação em cadeia da polimerase
PFAM	banco de dados de famílias de proteínas
piRNA	RNA que interage com Piwi
ptn	proteína
RNA	ácido ribonucleico
rRNA	ácido ribonucleico ribossomal
SAM	formato de mapa de alinhamento de sequência
siRNA	pequeno ácido ribonucleico de interferência
SRA	banco de dados de arquivo de leitura de sequência
tRNA	ácido ribonucleico transportador
UCSC	Universidade da Califórnia Santa Cruz
UTR	região não traduzida

1 INTRODUÇÃO

1.1 Doença de Chagas

A doença de Chagas (DC), descoberta por Carlos Chagas em 1909, é uma infecção sistêmica de evolução crônica, que pertence ao grupo de doenças tropicais negligenciadas (DTN) e é causada pelo protozoário flagelado *Trypanosoma cruzi* (1). A transmissão da DC se dá principalmente pela via vetorial, onde os insetos hematófagos da subfamília Triatominae, conhecidos popularmente como barbeiros ou chupões, são os responsáveis. O parasita possui um ciclo biológico complexo heteroxeno que envolve um hospedeiro vertebrado (macaco, gambá, cão, gato, homem, entre outros) e um hospedeiro invertebrado (triatomíneos) (2).

O ciclo biológico da DC (Figura 1.1) se inicia com o repasto sanguíneo (picada) do triatomíneo infectado no hospedeiro vertebrado, que ocorre geralmente à noite. Durante ou após o repasto, o triatomíneo defeca próximo ao local da picada e são nas fezes que estão presentes as tripomastigotas metacíclicas, forma infectante para mamíferos. Normalmente, a picada pode causar uma leve coceira ou ardência, assim, o hospedeiro se coça e possibilita a penetração das tripomastigotas no organismo, ou então, a picada ocorre em uma mucosa que serve como via de entrada (2). As tripomastigotas metacíclicas penetram nas células no lugar da picada e, dentro das células, elas se transformam em amastigotas. As amastigotas intracelulares se multiplicam por fissão binária e então se transformam em tripomastigotas, que entram na corrente sanguínea devido à ruptura da célula hospedeira. Enquanto essa tripomastigota é a forma infectante para o triatomíneo, quando esse faz o repasto sanguíneo no mamífero infectado, as tripomastigotas penetram no inseto e se diferenciam em epimastigotas. No intestino médio, as epimastigotas se multiplicam por fissão binária e, quando migram pro intestino posterior, elas se diferenciam em tripomastigotas metacíclicas, fechando o ciclo (3).

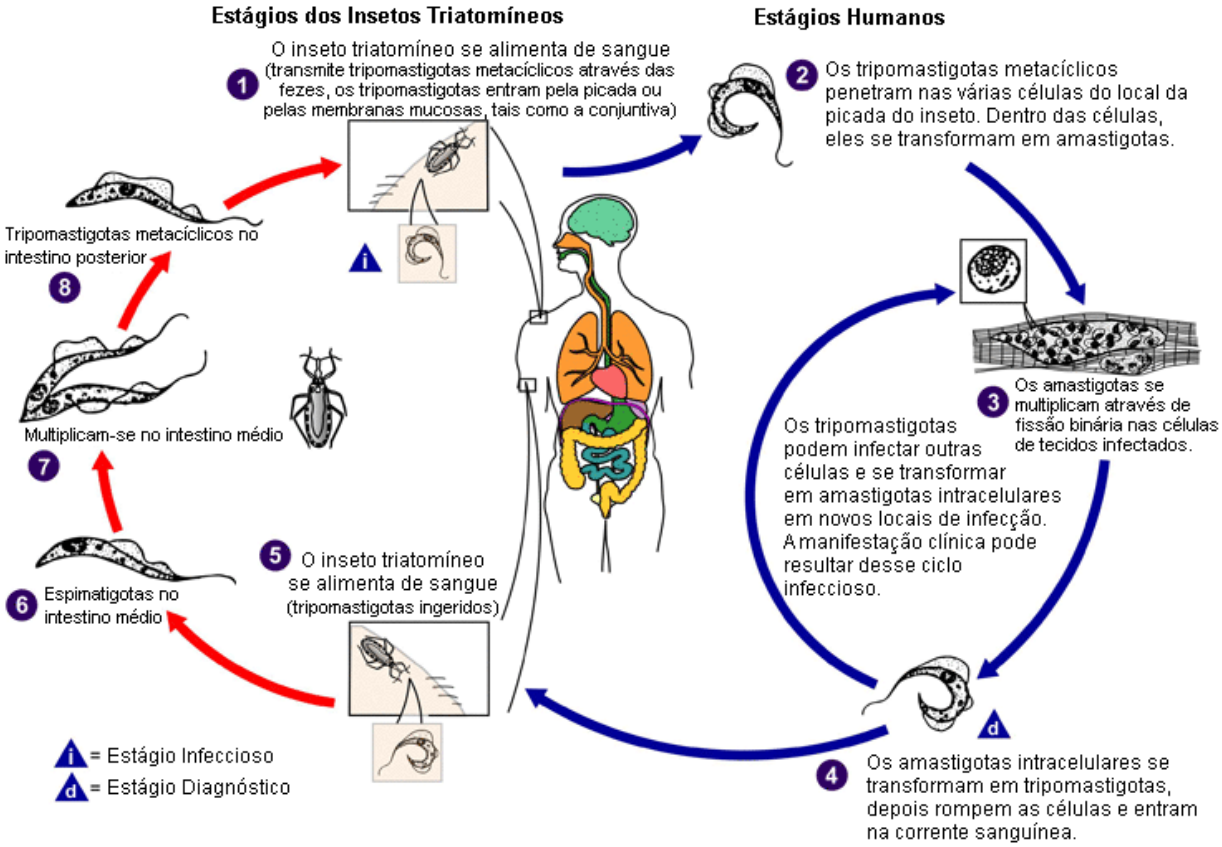

Figura 1.1 - Ciclo biológico do *Trypanosoma cruzi*. Imagem original obtida no site do CDC (https://www.cdc.gov/) e traduzida pelo site BMJ (https://bestpractice.bmj.com/).

A infecção aguda da DC dura de 4-8 semanas e é assintomática na maioria dos casos. Após esse período, os pacientes não tratados permanecerão cronicamente infectados (3). A infecção crônica também é assintomática na maioria dos indivíduos, mas alguns deles podem desenvolver miocardiopatia chagásica ao longo de anos ou décadas. A DC também pode afetar o sistema gastrointestinal, principalmente esôfago e cólon, resultando em danos nos neurônios intramurais que progridem até o quadro de megaesôfago e/ou megacólon (4).

A transmissão da DC pode, ainda, se dar por transfusão sanguínea, alimentar, e por via oral, além da transmissão vetorial. A transfusão sanguínea já foi um dos mecanismos de transmissão mais frequentes. Ao decorrer dos anos, uma triagem sorológica foi estabelecida no Brasil e em vários outros países como forma de controle da transmissão. Portanto, houve uma significativa diminuição da prevalência de sorologia positiva para DC em doadores de sangue, no Brasil por exemplo, na década de cinquenta a prevalência era de 8,3% e em 2005 o número já tinha caído para 0,21% (5–7).

Já a transmissão oral em humanos foi reconhecida como causa de pequenos surtos esporádicos, principalmente na região Amazônica (8–12). Curiosamente, os indivíduos infectados oralmente apresentam sinais e sintomas da infecção por *T. cruzi* com mais frequência do que os indivíduos infectados de forma vetorial, e em alguns casos, a taxa de mortalidade chega a ser de 29% (9,10,13). Além desses três mecanismos de transmissão também existem a transmissão congênita ou placentária, acidentes de laboratório, manuseio de animais infectados, ingestão de carne mal cozida de animais infectados, transplante de órgãos infectados e transmissão sexual (5).

Quanto à sua extensão, a doença de Chagas está presente endemicamente em 21 países da América Latina (Figura 1.2), e estima-se que aproximadamente 10 milhões de pessoas estão infectadas por *T. cruzi* no planeta. Ao redor do mundo, a taxa de mortalidade da DC ultrapassa 10.000 mortes por ano e o custo dos cuidados médicos de todos os pacientes crônicos por ano é, em média, de 267 milhões de dólares (1). A imigração de pessoas latino-americanas chagásicas para países não endêmicos tem contribuído para a geração de novos casos autóctones de DC, uma vez que esses países não possuem procedimentos para lidar com essa doença. Estima-se que 14 milhões de indivíduos contaminados provenientes de regiões endêmicas imigraram para a América do Norte, Europa, Japão e Austrália, onde viabilizaram a transmissão de Chagas pelos mecanismos transfusional, congênito e de transplante de órgãos (14).

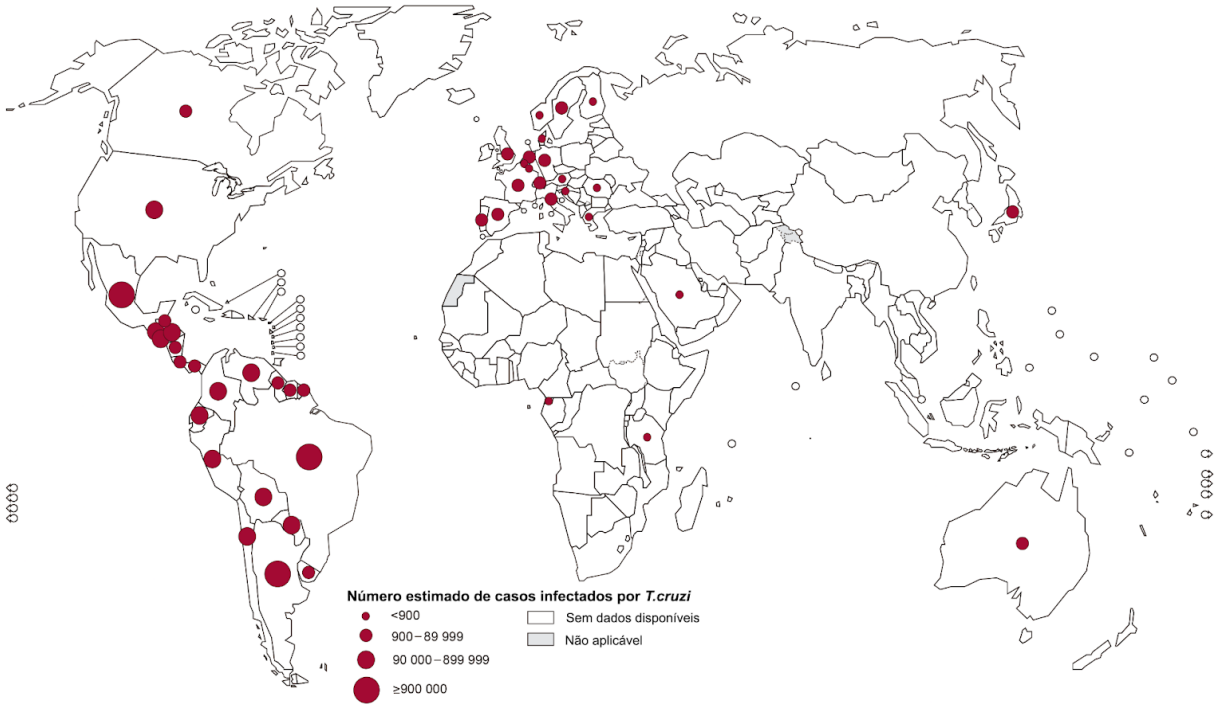

Figura 1.2 - Distribuição dos casos da infecção por *T.cruzi* ao redor do mundo em 2018, segundo a OMS. Imagem adaptada obtida no site da OMS (https://www.who.int/).

O tratamento da DC é feito principalmente através do benzonidazol, e quando necessário, o nifurtimox pode ser utilizado, ambas as drogas são eficientes para o tratamento da infecção aguda. Entretanto, para pacientes crônicos, a taxa de sucesso desses medicamentos cai drasticamente. Como consequência, muito do foco a respeito da DC está na prevenção da transmissão ao invés da busca de novos fármacos para o tratamento desses pacientes crônicos. Logo, ainda são necessárias medidas para diagnosticar rapidamente a infecção aguda e, principalmente, para controlar o vetor (15).

1.2 Triatomíneos

Atualmente são descritas e reconhecidas 150 espécies de triatomíneos adaptadas aos mais diversos biomas, sendo 69 espécies identificadas no Brasil. Os triatomíneos estão agrupados na Ordem Hemiptera, Subordem Heteroptera, Família Reduviidae, onde estão classificados em cinco tribos (Alberprosenini, Bolboderini, Cavernicolini, Rhodniini e Triatomini) e 15 gêneros. Essas espécies estão majoritariamente distribuídas nas Américas (Figura 1.3), desde os Estados Unidos até a Argentina, mas também existem espécies presentes na Ásia e África. Das

cinco tribos de triatomíneos, Triatomini e Rhodniini incluem 88% das espécies conhecidas, e dessas duas tribos, 86,5% das espécies pertencem aos gêneros *Triatoma*, *Panstrongylus* e *Rhodnius* (16). Esses três gêneros são mais bem estudados por serem relevantes na transmissão da DC para humanos.

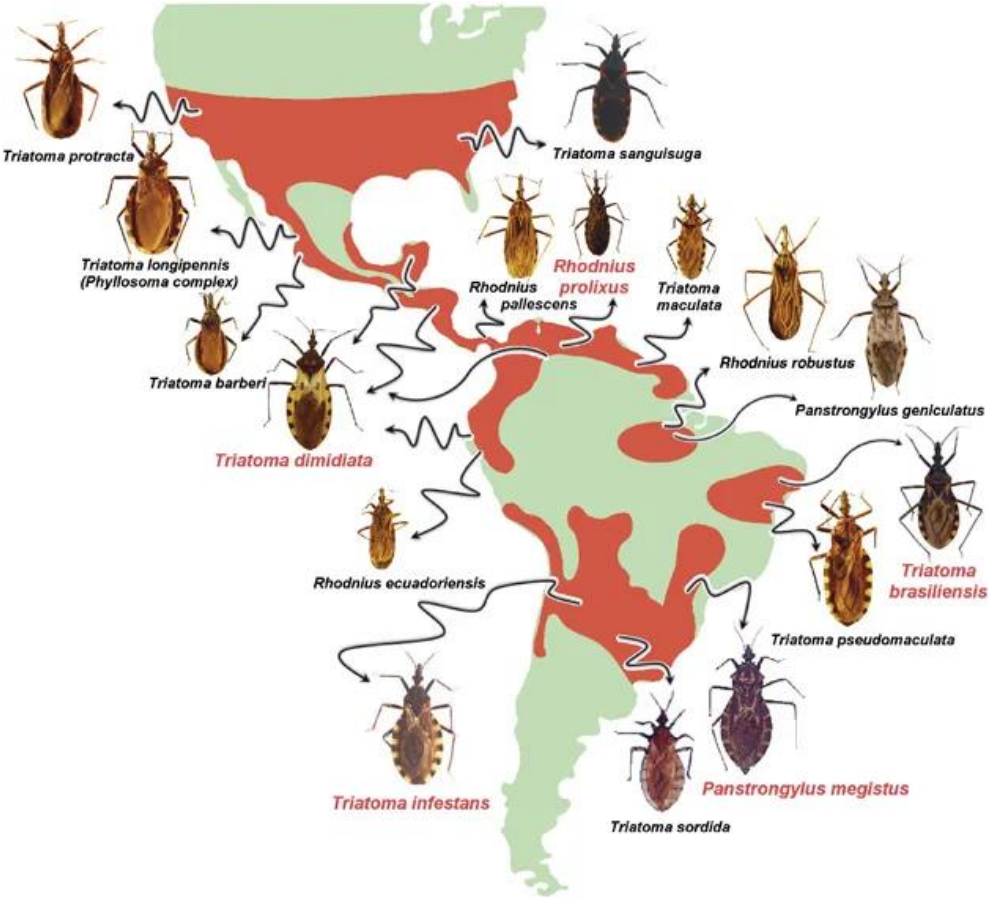

Figura 1.3 - Distribuição das espécies de triatomíneos de maior relevância epidemiológica, 2011. As espécies consideradas como vetores mais importantes estão marcadas em vermelho. Adaptado de GOURBIÈRE, S. et al (2012) (17).

As espécies do gênero *Rhodnius* tem preferência por colonizar palmeiras, já as espécies dos gêneros *Triatoma* e *Panstrongylus* vivem preferencialmente em associação com hospedeiros terrestres. O desmatamento pressionou os triatomíneos a se adaptarem aos ambientes ocupados pelo homem. Como resultado, diversas espécies são encontradas em habitações humanas ou estruturas peridomiciliares, como galinheiros e chiqueiros (18). As espécies desses três gêneros têm uma grande capacidade de domiciliação, o que contribui para que o *Triatoma infestans* (Klug, 1834) e o *Rhodnius prolixus* (Stål, 1859) se tornassem os principais vetores da DC (19).

O *T. infestans* foi erradicado em vários países da América Latina devido às políticas de controle vetorial (5). Por volta do ano 2012, alguns países da América Central endêmicos para DC, transmitida por *R. prolixus*, foram certificados como livres da infecção vetorial por esse inseto. Por outro lado, outros triatomíneos podem ocupar o seu lugar nesse nicho (20). Ainda assim, o *R. prolixus* é um importante vetor na Venezuela e Argentina mesmo que medidas de controle vetorial tenham sido adotadas. Além disso, é difícil dizer como as mudanças climáticas podem influenciar a distribuição geográfica de *Rhodnius*, visto a sua capacidade adaptativa aos ambientes domiciliares (21).

1.3 *Rhodnius prolixus*

O gênero *Rhodnius* possui 20 espécies e é dividido em três grupos: os grupos *prolixus* e *pictipes*, distribuídos ao leste da Cordilheira dos Andes, e o grupo *pallescens*, a oeste. Essas espécies apresentam pouca variação morfológica, sendo difíceis de identificar com base somente na morfologia. Entretanto, a falta de variação morfológica não reflete uma baixa diversidade genética dentro do grupo (22). O gênero tem ocorrência natural desde a América Central até o norte da Argentina, todavia a Amazônia é uma região que concentra uma alta riqueza de espécies. *R. prolixus* é um dos vetores epidemiologicamente mais relevantes da DC na América Latina, especialmente na Venezuela, Colômbia e América Central, devido a sua grande capacidade de domiciliação, como já citado (23).

O inseto *R. prolixus* se desenvolve através de hemimetabolismo, ou seja, ao longo do seu crescimento ele avança pelos seguintes estágios: i) ovo, estágio que precede o nascimento do inseto; ii) ninfa, etapa pós-eclosão onde o inseto se assemelha bastante a fase adulta, porém sistemas como reprodutor e de voo ainda estão amadurecendo; e iii) adulto, ponto onde o organismo está plenamente desenvolvido (24). O tempo médio para a eclosão dos ovos é de 18 dias, após esse evento a ninfa passa por cinco estágios até se tornar um inseto adulto e essa maturação leva cerca de quatro meses (25,26).

Esses animais possuem hábitos noturnos, permanecendo escondidos no seu abrigo ao longo do dia e ao anoitecer saem em busca de alimento. Tanto machos como fêmeas se alimentam de sangue, podendo ingerir quantidades suficientes para causar distensão abdominal (25,27). Apesar da fêmea conseguir produzir uma

pequena quantidade de ovos com os recursos estocados antes da muda para adulto, a hematofagia é de suma importância para a oviposição. Através da ingestão de sangue um grande número de ovos viáveis é produzido, o que vai resultar na produção de mais de 30 ovos nas três semanas seguintes à alimentação (28,29).

Além de se infectar com *T. cruzi*, *R. prolixus*, também pode abrigar o *Trypanosoma rangeli* (Tejera, 1920) ou, ainda, ambos. O *T. rangeli* não é capaz de causar doença aos mamíferos, mas a sua infecção promove uma resposta imune com geração de anticorpos que causa uma reação cruzada com antígenos de *T. cruzi* e pode levar a um falso diagnóstico de DC (30,31).

Além da sua importância como vetor da DC, o *R. prolixus* pode ser considerado como um “clássico modelo” para a pesquisa científica. Através dos estudos de Vincent B. Wigglesworth na década de 1930, muito se sabe a respeito da fisiologia de insetos. Como consequência, várias dúvidas a respeito do ciclo de vida, biologia básica e desenvolvimento de insetos hematófagos da Família Reduviidae foram esclarecidas. Além disso, o sequenciamento do genoma de *R. prolixus* em 2015 constituiu um grande avanço na área de entomologia, pois permitiu estudos comparativos com outras espécies de insetos (32).

1.4 Genoma de *R. prolixus*

Como já mencionado, *R. prolixus* ainda é um importante vetor e um modelo de estudos comparativos em entomologia. Antes da chegada da era do sequenciamento, algumas características do genoma desse inseto já haviam sido desvendadas. Na década de 50, já se tinha informações sobre o número de cromossomos, sendo 20 autossômicos e dois sexuais (XY) (33). Nos anos seguintes, foi estimado que o tamanho do genoma de *R. prolixus* era de 600 Mbp com um valor C (quantidade de DNA dentro de um núcleo haplóide) de 0.69 pg, ambos os resultados foram obtidos através de citometria de fluxo (34). Posteriormente, utilizando técnicas de hibridização, observou-se que *R. prolixus* apresentava regiões repetitivas dispersas por toda a sua cromatina, incluindo no cromossomo Y. Este acontecimento embasou os estudos que demonstravam que esse inseto possuía diferentes famílias de elementos transponíveis se comparado aos outros triatomíneos (35).

Então, no final dos anos 2000, diversos pesquisadores se reuniram para tornar possível o sequenciamento do genoma desse inseto (17,36,37). O primeiro sequenciamento do inseto data do ano de 2007 e foi feito através do equipamento ABI 3730, porém essa versão de montagem (1.0.1) não foi muito explorada e por isso não foi considerada neste trabalho (GCA_000181055.1).

Já o segundo sequenciamento iniciou-se em 2010, utilizando a mesma tecnologia anterior juntamente com o pirosequenciamento (Roche 454) e, consequentemente, resultou na montagem 3.0.1 (GCA_000181055.2) e um artigo associado. Naquela época, esta montagem revelou inúmeras expansões em grupos de genes relacionados com mecanismos de defesa e desenvolvimento do inseto, por exemplo. Em contrapartida, também foram observadas algumas reduções em genes envolvidos na excreção, propondo-se uma relação dessa característica aos seus hábitos hematofágicos (38).

Também foram identificadas diversas vias de sinalização do sistema imune, como Toll e Jak/STAT, além daquelas relacionadas à interação com *T. cruzi*. Aparentemente, a infecção pelo protozoário não ativa o sistema imune, sugerindo que o parasita desenvolveu mecanismos de evasão e/ou tolerância. Essa descoberta deixou mais evidente a ótima adaptação do *T. cruzi* ao seu hospedeiro invertebrado. Também foram identificados diversos genes relacionados à sinalização das vias MAPK e TOR, e ao desenvolvimento de ovos e dos padrões embrionários (38). A respeito da resistência a inseticidas, foram encontradas famílias de genes como citocromo P450 e glutationa s-transferase. Alguns genes, com expansões específicas da espécie, estão potencialmente relacionados com resistência, devido ao papel de detoxificação que esses genes exibem em outras espécies de insetos (39).

O terceiro sequenciamento foi desenvolvido em 2015 com as tecnologias de Roche 454, método de Sanger e Illumina e, originou a montagem 3.0.3 (GCA_000181055.3). Apesar dessa montagem ter sido mais explorada que a 1.0.1, ela não possui um artigo próprio, o que dificulta as comparações ao nível biológico com a versão 3.0.1. Por fim, os *contigs* da montagem 3.0.3 foram usados como base para a montagem utilizando a técnica Hi-C afim de gerar a versão 3.0.3 Hi-C, que é a mais atual e está disponível no site DNAzoo (https://bit.ly/3sqNRHn). De maneira resumida, a técnica Hi-C cria ligações covalentes cruzadas entre as proteínas do DNA através da fixação das células com formaldeído. Em seguida, o DNA é

fragmentado e os fragmentos têm um resíduo de biotina incorporado na sua região 5’ gerando junções quiméricas entre sequências próximas no espaço genômico. As junções de biotina são purificadas para que seja realizado o sequenciamento e consequentemente a montagem (40).

Para fazer um histórico das montagens genômicas de *R. prolixus,* iniciamos com a 3.0.1 que contém 702 MB (Tabela 1.1), com o maior *scaffold* de 12.301MB e 86% do genoma em *scaffolds* maiores que 50KB. A versão 3.0.3 apresentou uma pequena melhora nos dados quando comparada à versão 1, em relação ao maior *scaffold* (9,1%) e o valor de N50 (28,4%). Já a montagem Hi-C possui um tamanho menor que as duas anteriores, porém fornece uma grande melhoria nos dados. A versão Hi-C apresenta um valor de *scaffold* N50 47 vezes maior do que a versão 3.0.3, mostrando que a técnica de Hi-C aprimorou a cobertura do genoma e a montagem em si.

Tabela 1.1 - Comparação entre as versões de montagem de *R. prolixus*.

	Versão 3.0.1	**Versão 3.0.3**	**Versão Hi-C**
Tamanho total do genoma	702.645 MB	706.824 MB	671.560 MB
Maior scaffold	12.301 MB	13.426 MB	69.096 MB
Scaffold N50	847.873 KB	1.089 MB	47.233 MB
% do genoma em scaffolds > 50 KB	86	89	92

1.5 Anotação automática de genomas

1.5.1 Anotação de genoma propriamente dita

Após a montagem do genoma é importante identificar regiões funcionais como genes codificadores de proteínas, elementos de repetição, região promotora, regiões regulatórias, genes de RNAs não-codificadores, este processo é conhecido como anotação genômica. Os genes codificadores de proteínas são os responsáveis por dar origem as principais moléculas responsáveis pelo metabolismo da célula. Os elementos de repetição compõem grande parte do genoma de eucariotos e, apesar de não terem suas funções bem definidas, é evidente a sua importância para diversidade genética devido a capacidade em gerar mutações. As regiões promotoras e regulatórias controlam a expressão gênica, definindo quando aquele

gene será ou não transcrito. Os RNAs não-codificadores também têm papel importante na regulação da expressão gênica seja promovendo (tRNA e rRNA) ou inibindo essa expressão (miRNA e siRNA) (41,42).

Quanto a anotação de proteínas, desde a década de 1980 já existiam métodos para tentar predize-las a partir de sequências genômicas de DNA. Entretanto, apenas no começo dos anos 1990 que os primeiros programas para identificar proteínas em eucariotos surgiram. Muitos desses *softwares* utilizavam da similaridade entre o genoma e sequências biológicas conhecidas, obtida de ferramentas de alinhamento como o BLAST (43), para identificar as regiões codificadores. Apesar dessa metodologia ter sido amplamente utilizada, indicar com precisão os limites da região com similaridade nem sempre era possível. Além disso, éxons muito pequenos eram facilmente perdidos e o posicionamento dos íntrons poderia ser errôneo (44,45).

Ainda nos anos 1990, novas abordagens foram desenvolvidas para melhorar a predição de genes codificadores. O conteúdo intrínseco das sequências passou a ser mais considerado, exemplos destes são: composição de nucleotídeo, composição de códon, hexâmero (sequência de 6 nucleotídeos) e etc (45,46). Este último exemplo foi a base para programas como o SORFIND (47) e o Genview2 (48). O modelo de Markov (MM) foi introduzido na predição de proteínas com o GeneMark (49), método conhecido como predição *ab initio*, onde nesse modelo a probabilidade de um nucleotídeo em uma dada posição é definida por *k* nucleotídeos anteriores. Na aplicação deste modelo, é necessário um treinamento com um conjunto de sequências para então obter as probabilidades que são usadas na predição em si. Novos *softwares* surgiram com o objetivo de minimizar o problema da demanda de grande número de sequências codificadores para gerar MMs confiáveis. Dentre eles estavam o Glimmer (50) que usava modelos de Markov interpolados e o GeneMark.hmm (51) que adotou modelos com base no conteúdo de G+C no genoma.

Os MMs foram de suma importância para o desenvolvimento dos programas baseados em modelos ocultos de Markov (HMM) que se mantêm relevantes até os dias atuais. De maneira resumida, as transições entre submodelos que correspondem a componentes de um gene (UTR, éxon, íntron) são modeladas de forma oculta, determinando a probabilidade de gerar nucleotídeos observáveis e assim sendo capaz de prever o componente gênico. Entretanto, o HMM não

consegue determinar com precisão o comprimento de éxons e íntrons uma vez que estes são limitados pela emenda de mRNA (*splicing*). Assim esse modelo é generalizado (GHMM) para ser possível a estimar com acurácia o comprimento desses elementos (52). As análises feitas pelos programas AUGUSTUS (53), GENEID (54) e SNAP (55) usam esse modelo. Com o aumento da quantidade de dados de sequenciamento disponíveis principalmente de RNAseq, os programas de predição passaram a integrar esses dados à metodologia *ab initio*. O uso de evidências (sequências de mRNA e de EST) para aumentar a acurácia e identificar genes de maneira mais confiável aumentou drasticamente. (56–58).

Os últimos 30 anos de desenvolvimento de *softwares* e metodologias para predizer proteínas não revelou a melhor técnica ou protocolo a ser usado. É inegável que os programas que se baseiam nos modelos de Markov ainda são os mais usados nos dias de hoje. Entretanto, não existe um consenso para eleger o melhor entre eles, há aqueles que performam melhor na predição correta de éxons e proteínas, mas falham em predizer proteínas curtas ou tem seu desempenho variável dependendo da espécie, por exemplo (59,60). Recentemente, o aprendizado de máquina tem sido utilizado em diversas áreas da bioinformática, logo é questão de tempo para que esse método seja aplicado na predição de proteínas (61). De forma geral, toda a técnica aplicada na anotação de genomas tem suas limitações. As abordagens baseadas em similaridade são limitadas a disponibilidade de dados de transcriptoma ou de genomas de espécies próximas, já os MMs dependem da qualidade do genoma avaliado, caso a mesma seja baixa, a anotação pode conter genes fragmentados, quiméricos ou mesmo éxons ausentes. Para metodologias que envolvam aprendizado de máquina, um conjunto de alta qualidade, não redundante e balanceado é necessário para um treinamento adequado e criação do modelo de predição. Portanto, o programa/protocolo mais apropriado vai variar dependendo da qualidade dos dados disponíveis, da espécie estudada e etc (62,63).

1.5.2 As predições gênicas de R. prolixus

No caso de *R. prolixus*, as versões 3.0.1 e 3.0.3 da montagem apresentam diferentes predições gênicas (Tabela 1.2), porém a versão Hi-C não possui nenhuma, assim como ainda não foi explorada usando dados de RNAseq. As predições 1.0 até a 1.2 foram criadas com o *pipeline* do Ensembl pela equipe do

VectorBase, seguida de alguma curagem manual da comunidade. Para a predição 1.3 (associada à publicação do *paper* do genoma de *R. prolixus*) foi utilizado o GeneId (38). As predições 3.2 e 3.3 não foram feitas de forma tradicional, ao invés disso, apenas houve a tentativa de posicionar os genes já identificados na predição 3.1. Nelas, alguns genes desapareceram, poucos ou nenhuns genes novos surgiram e todos os outros são exatamente iguais, ou seja, não houve uma melhora considerável ao decorrer das predições, além de que genes antigos e relevantes podem ter sido perdidos. No entanto, a predição 3.4 feita através da anotação da comunidade, conseguiu melhorar minimamente esses problemas, uma vez que alguns novos genes surgiram.

Tabela 1.2 - Predições gênicas de *R. prolixus*.

	N° de transcritos preditos	N° de peptídeos preditos	Montagem do genoma
Predição 1.0	16.184	16.134	3.0.1
Predição 1.1	17.155	15.441	3.0.1
Predição 1.2	17.256	15.441	3.0.1
Predição 1.3	17.262	15.456	3.0.1
Predição 3.1	16.857	15.078	3.0.3
Predição 3.2	15.755	15.078	3.0.3
Predição 3.3	15.752	15.075	3.0.3
Predição 3.4	15.783	15.106	3.0.3

A tabela foi baseada na contagem de transcritos/peptídeos nos arquivos do Vectorbase.

Simultaneamente, apenas a predição mais recente (3.4) está disponível no navegador de genomas do site do VectorBase, tornando inviável navegar pelos transcritos que são exclusivos de versões mais antigas.

1.6 Anotação através de navegação genômica

O rápido avanço nas tecnologias de sequenciamento gerou um grande acervo de dados genômicos, o que criou uma demanda para o desenvolvimento de novas

técnicas para explorar e analisar esses dados. No caso de sequências genômicas, a visualização é indispensável para corrigir erros derivados de um processo automatizado, permitindo uma curadoria mais precisa e palpável. Devido a isso, os navegadores de genoma foram desenvolvidos para permitir que o usuário explorasse as sequências e pudesse encontrar uma região específica do DNA rapidamente e de maneira intuitiva (64,65).

Os navegadores mostram dados e anotações biológicas provenientes de diversas fontes numa interface gráfica, onde esses dados incluem: expressão gênica, comparações entre espécies, variação genotípica, genes mapeados ou preditos, entre outros. Assim, informações oriundas de diferentes fontes podem ser reunidas e integradas num único navegador, possibilitando uma análise mais completa e eficiente dos dados. Geralmente, cada tipo de dado é mostrado na forma de *tracks*, por exemplo, uma *track* para o genoma de um organismo e outra para o conjunto de genes preditos, de maneira que as duas fiquem alinhadas corretamente, permitindo a análise das posições dos genes em relação ao genoma. Além disso, muitos navegadores são customizáveis, como o Ensembl (66), deixando o usuário escolher qual conteúdo ele gostaria de ver ou mesmo inserindo comentários e marcações nas *tracks* de interesse. Existem dois tipos principais de navegadores de genoma, o primeiro é focado em múltiplas espécies o qual integra a sequências e anotações de várias espécies, permitindo a comparação entre elas. O outro tipo é focado em uma espécie específica onde um único organismo pode ter diversas anotações (64,65).

Um dos mais famosos navegadores de genoma é o UCSC *genome browser* (67), que foi desenvolvido com o objetivo de garantir um acesso fácil e rápido aos crescentes volumes de dados genômicos que eram gerados na época. Além das funções clássicas, esse navegador disponibilizava outras ferramentas como BLAT, PCR *in silico*, um navegador de proteoma, e ele ainda continua sendo atualizado e ganhando novas funções (68). Um outro navegador de genomas conhecido é o Jbrowse (69), ele foi criado pela necessidade de uma navegação mais fluida pelos dados genômicos, uma vez que a maioria dos navegadores naquele período utilizavam o protocolo CGI (interface de gateway comum). Esse protocolo não era muito otimizado, o que acabava por gerar lentidão para os usuários e um gasto computacional excessivo para os servidores. Com base nisso, o Jbrowse foi desenvolvido em HTML e JavaScript para oferecer uma interface mais rápida e

compacta, além de proporcionar o uso de *plugins* que adicionam e/ou aprimoram suas funções.

Em 2019, a fusão do VectorBase e do EuPathDB adotou o Jbrowse com navegador oficial para compor o novo site VEuPathDB. Nesse site existem dados dos mais diversos organismos, mas infelizmente, referindo-se a insetos vetores, apenas a predição gênica mais recente fica disponível para navegação (70). O Flybase (71), banco de dados de *Drosophila melanogaster*, também utiliza o mesmo navegador do VEuPathDB juntamente com a recente plataforma InsectBase (72). Em resumo, um navegador de genomas permite que pesquisadores de todo o mundo possam acessar um conjunto de dados integrados, derivados de diversas fontes, sem precisar baixar nem instalar nada. Consequentemente, isso fomenta uma interação entre pesquisadores e o compartilhamento de conhecimento, o que pode gerar novos estudos e parcerias (65).

É evidente que os avanços científicos proporcionados pelo sequenciamento e estudo de genomas ampliaram nossa visão sobre a biologia. Quando se trata do *R. prolixus*, os conhecimentos gerados por esses avanços suportam uma grande comunidade que trabalha com vetores, especialmente o INCT de entomologia molecular (http://www.inctem.bioqmed.ufrj.br/), do qual o grupo de pesquisa do doutor Rafael Dias Mesquita participa. Entretanto, a forma como esses dados genômicos estão organizados e disponibilizados atualmente, sem predição gênica e somente para download, pode prejudicar o desenvolvimento de futuros estudos, incluindo sobre novas formas de combater esse vetor.

1.7 Justificativa

Os dados relacionados a um genoma tendem a ser mais utilizados pela comunidade científica quando há a conveniência de encontrá-los com predição gênica em navegador de genomas. Além disso, a disponibilidade dos genes de diferentes predições no navegador também contribui para a velocidade de pesquisas.

O genoma de *R. prolixus* consiste em um importante recurso para o desenvolvimento do estudo do vetor, incluindo estudos de ciência básica e aplicada, na temática da doença de chagas, como novas formas de combater a transmissão vetorial. Dentre as várias versões de montagem do genoma de *R. prolixus,* a mais

atual e que oferece melhor cobertura, não possui predição gênica, muito menos está disponível em um navegador de genomas, e obviamente não se integra com dados de RNA-seq.

A disponibilização dos dados genômicos totais, incluindo não só a montagem atual, mas todas as anteriores possibilitariam novos e mais profundos estudos com o vetor *R. prolixus*.

2 OBJETIVOS

2.1 Objetivo Geral

Melhorar a qualidade e disponibilidade dos dados genômicos do vetor *Rhodnius prolixus*.

2.2 Objetivos Específicos

- Fazer o mapeamento na montagem Hi-C (atual) de dados de RNA-seq;
- Realizar a predição gênica da versão Hi-C do genoma;
- Conciliar e organizar um conjunto total de genes, contendo os genes preditos na montagem Hi-C mais genes não redundantes de predições anteriores;
- Disponibilizar os dados genômicos em um *website* no navegador de genomas.

3 MATERIAL E MÉTODOS

Um fluxograma da metodologia (Figura 3.1) foi desenvolvido para facilitar a compreensão de todas as etapas realizadas desde o mascaramento do genoma até a disponibilização dos dados no navegador.

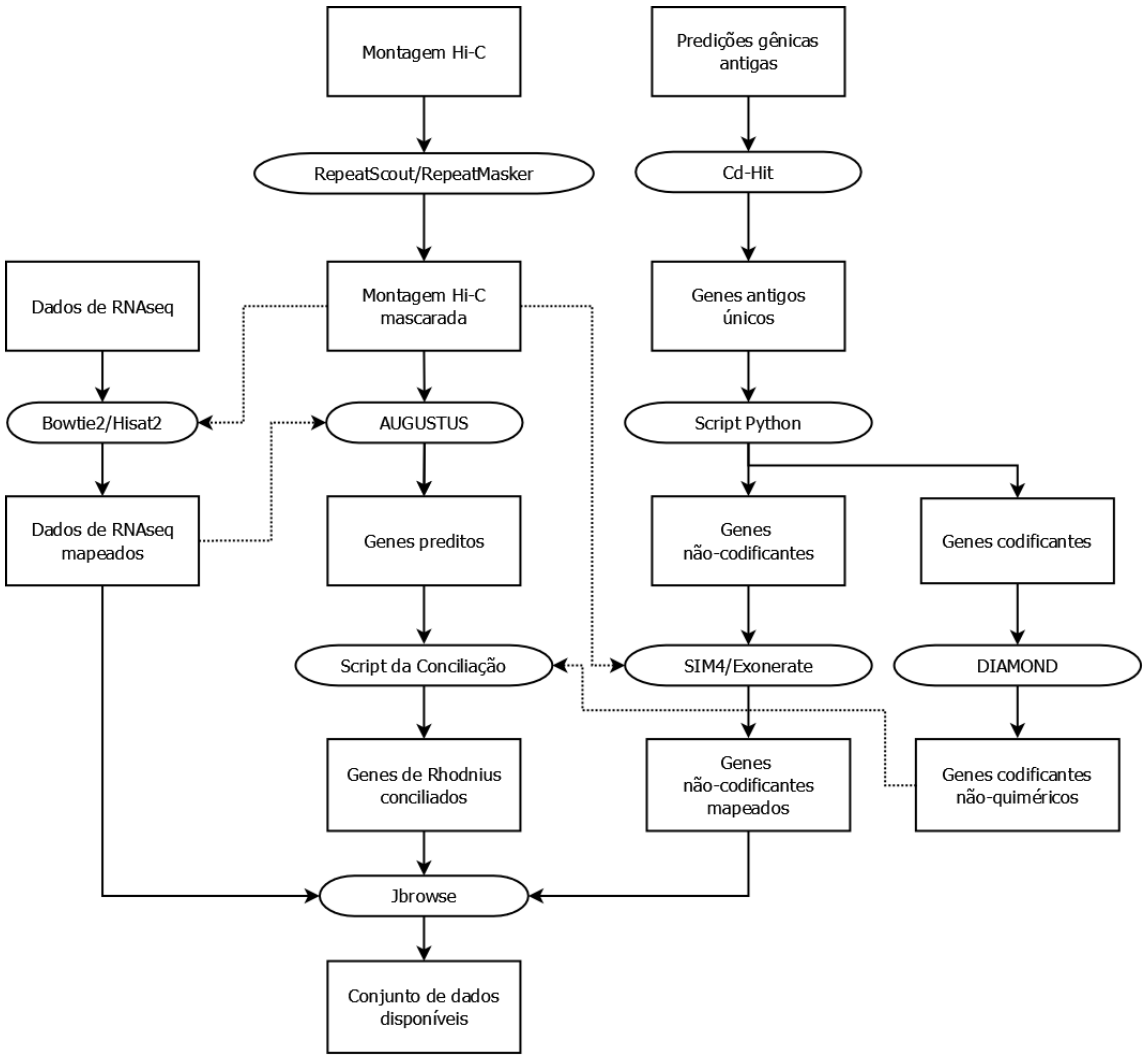

Figura 3.1 - Fluxograma da metodologia. Os retângulos são os dados de entrada e saída, as setas contínuas são o fluxo principal de cada etapa, as formas arredondadas são o *software*/processamento utilizado e as setas pontilhadas mostram a comunicação entre fluxos verticais diferentes.

3.1 Obtenção dos dados

Para ajudar na identificação de genes e predição de variantes de emenda, os dados de RNAseq foram obtidos a partir do banco de dados SRA (NCBI) para serem utilizados no programa de predição gênica na montagem Hi-C do genoma de *R. prolixus*. O filtro utilizado na busca dos dados de RNAseq foi "*Rhodnius*

prolixus"[Organism] AND ("biomol RNA"[Properties] AND "platform illumina"[Properties]), estes dados estão listados na tabela 3.1. Além disso, foram usados dados de RNAseq de ninfas de *R. prolixus* infectadas e não infectadas com *T. cruzi*, em um curso temporal de 12 dias, disponibilizados por colaboradores (Tabela 3.2). As proteínas dos insetos *Acyrthosiphon pisum*, *Aedes aegypti*, *Cimex lectularius*, *Glossina morsitans* e *Pediculus humanus* foram usadas para gerar grupos ortólogos foram obtidas do Ensembl Metazoa (https://metazoa.ensembl.org/index.html).

Tabela 3.1 - Dados de RNAseq obtidos do SRA.

Acession number	**Número de *reads***
ERX1387156	62.251.564
ERX1387157	40.744.142
ERX1387158	85.585.248
SRX1011769	124.995.224
SRX1011778	125.617.370
SRX1011796	103.016.074
SRX6380682	72.590.830
SRX6380683	93.269.222

Tabela 3.2 - Dados de RNAseq de colaboradores.

Nº de Replicatas Biológicas	**Condição**	**Dia**
3	Jejum	0
12	Sangue	1-4-8-12
12	Epimastigota 10^3	1-4-8-12
12	Epimastigota 10^7	1-4-8-12
12	Tripomastigota 10^3	1-4-8-12
12	Tripomastigota 10^7	1-4-8-12

Os dados de RNAseq foram unidos em arquivos únicos para facilitar as análises. Caso a biblioteca fosse *paired*, todas as bibliotecas frente (*forward*) comporiam um único arquivo frente, e o mesmo foi feito para as bibliotecas reversas (*reverse*). A versão de montagem utilizada como base foi a Hi-C do site DNAzoo, onde o genoma foi montado através da tecnologia de Hi-C

(https://www.dnazoo.org/assemblies/Rhodnius_prolixus). As versões de predição gênica foram obtidas a partir do VectorBase (https://legacy.vectorbase.org/organisms/rhodnius-prolixus). Também foram usados dados de transcritos íntegros previamente identificados em um transcriptoma de *R. prolixus* (73), neste trabalho chamados de genes “ouro”.

3.2 Distribuição dos transcritos ao longo das predições de *R. prolixus*

O *software* Cd-hit v4.6 (74) foi usado para agrupar os transcritos já preditos, considerando 95% de identidade, permitindo investigar a presença de transcritos exclusivos para alguma predição e que acabaram por desaparecer ao longo do tempo. Igualmente, esta análise permitiu verificar se houve ou não o surgimento de novas proteínas únicas da predição Hi-C.

3.3 Mascaramento do genoma

3.3.1 Mascaramento do genoma propriamente dito

O software RepeatScout v1.0.5 (75) foi utilizado no modo *default* para a identificação de elementos repetitivos no genoma de *Rhodnius prolixus* na montagem Hi-C (3.0.3). A primeira etapa do mascaramento consistiu em fazer uma tabela de frequência de *l-mers*, e com base nessa tabela, uma biblioteca contendo todos os tipos de repetições foi gerada com ajuda do programa.

A segunda etapa iniciou na filtragem da biblioteca pelo tamanho, selecionando sequências maiores que 50 pares de base. Após o filtro, o RepeatMasker v4.1.0 (http://www.repeatmasker.org/) foi utilizado no modo *default* para realizar a etapa de produção da biblioteca de repetições frequentes. O resultado considerou somente elementos repetitivos com mais de dez repetições. Finalmente, com a biblioteca de repetições foi possível realizar o mascaramento definitivo do genoma usando o RepeatMasker novamente, o parâmetro *xsmall* foi utilizado para que o mascaramento realizado seja do tipo *soft*, ou seja, os nucleotídeos foram convertidos de maiúsculo para minúsculo.

3.3.2 Classificação dos elementos repetitivos

Após o mascaramento, as sequências repetitivas consenso identificadas foram classificadas com o software Hmmsearch v3.1b2 (http://hmmer.org/) utilizando o banco de dados DFAM v3.3 (76). Em seguida, o resultado do Hmmsearch foi processado por um *script* em Python, programado para coletar o melhor *hit* de cada repetição e em seguida buscar a descrição da entrada do banco (*subject*) correspondente em um arquivo EMBL do DFAM. Assim, foi possível obter uma classificação mais completa dos *repeats* e, para as entradas com descrição vazia ou inespecífica, foram buscados manualmente no site do DFAM (https://dfam.org/home) os seus respectivos IDs para que fosse possível classificar o elemento repetitivo.

3.4 Alinhamento dos dados de RNAseq

Os dados de RNAseq de colaboradores já haviam passado por um controle de qualidade. Os adaptadores Illumina foram removidos através do programa cutadapt v1.16 (77) e a limpeza de qualidade de bases usou o programa trimmomatic v0.36 (78). Em seguida, o controle de qualidade foi feito com o programa FastQC (https://www.bioinformatics.babraham.ac.uk). Já para os dados obtidos do SRA, optou-se por não fazer a etapa de limpeza. Devido ao grande volume de dados de RNAseq obtido 238 GB, a perda das leituras com problemas não impediria a identificação das regiões expressas. O crucial nesse caso era que esses dados alinhassem e gerassem evidências para a predição gênica.

O programa Bowtie2 v2.2.6 (79) foi utilizado para mapear as leituras (*reads*) contra a versão de montagem Hi-C mascarada do genoma, foi aplicado o modo de alinhamento *end-to-end* que busca por alinhamentos envolvendo todos os nucleotídeos. Para o cálculo de *mismatch*, a seguinte função é utilizada: MN + floor((MX-MN) * (Q/40)), na qual MX (penalidade máxima de *mismatch*) é igual a 6, MN (penalidade mínima de *mismatch*) igual a 2, e Q corresponde a qualidade da base alinhada. A função linear para *score* mínimo de alinhamento é f(x) = 0 + -0.6 * x, sendo x o comprimento da leitura (*read*). Porém, ela foi ajustada para f(x) = -2 + -0.25 * x, onde uma sequência com 100 pares de base obteria -27 de *score*. Logo, considerando a leitura de 100 bases de comprimento, com qualidade média de base igual a Q30, que tenha 5% de erro, obtemos um score de -25 (número de erros: 5 x penalidade de *mismatch*: -5). Dessa forma, apenas são aceitos alinhamentos com até 5% de erro. Por fim, o parâmetro k foi definido para 3, o que retorna o máximo de

3 alinhamentos distintos para uma mesma leitura, evitando a perda de fluidez quando esses dados são visualizados no navegador de genomas. Os arquivos SAM gerados foram convertidos em BAM utilizando o programa Samtools v1.10 (80) e então foram disponibilizados no *browser*.

O *software* Hisat2 v2.1.0 (81) foi utilizado para mapear as leituras contra a versão de montagem Hi-C mascarada do genoma no modo *default*, de forma a aproveitar o máximo possível das leituras. Esse mapeamento foi usado na predição gênica para gerar evidências de íntrons que pudessem melhorar a acurácia da mesma. O hisat2 foi escolhido por ser eficaz em identificar sítios de emenda (*splicing*), e também porque já foi utilizado em um trabalho anterior do nosso grupo com a mesma finalidade.

3.5 Predição gênica

3.5.1 Predição propriamente dita

Na predição gênica foi utilizado o software AUGUSTUS v3.3.3 (53) que já possui treinamento para predição em *R. prolixus*. Primeiramente, foi feita uma predição básica usando o próprio treinamento do AUGUSTUS (Figura 3.2). Os mesmos parâmetros definidos aqui foram utilizados nas predições subsequentes, logo, dentre os principais estavam a predição de genes completos *(--genemodel=complete*) e redução da predição de éxons em regiões repetitivas mascaradas em letras minúsculas (*--softmasking=1*).

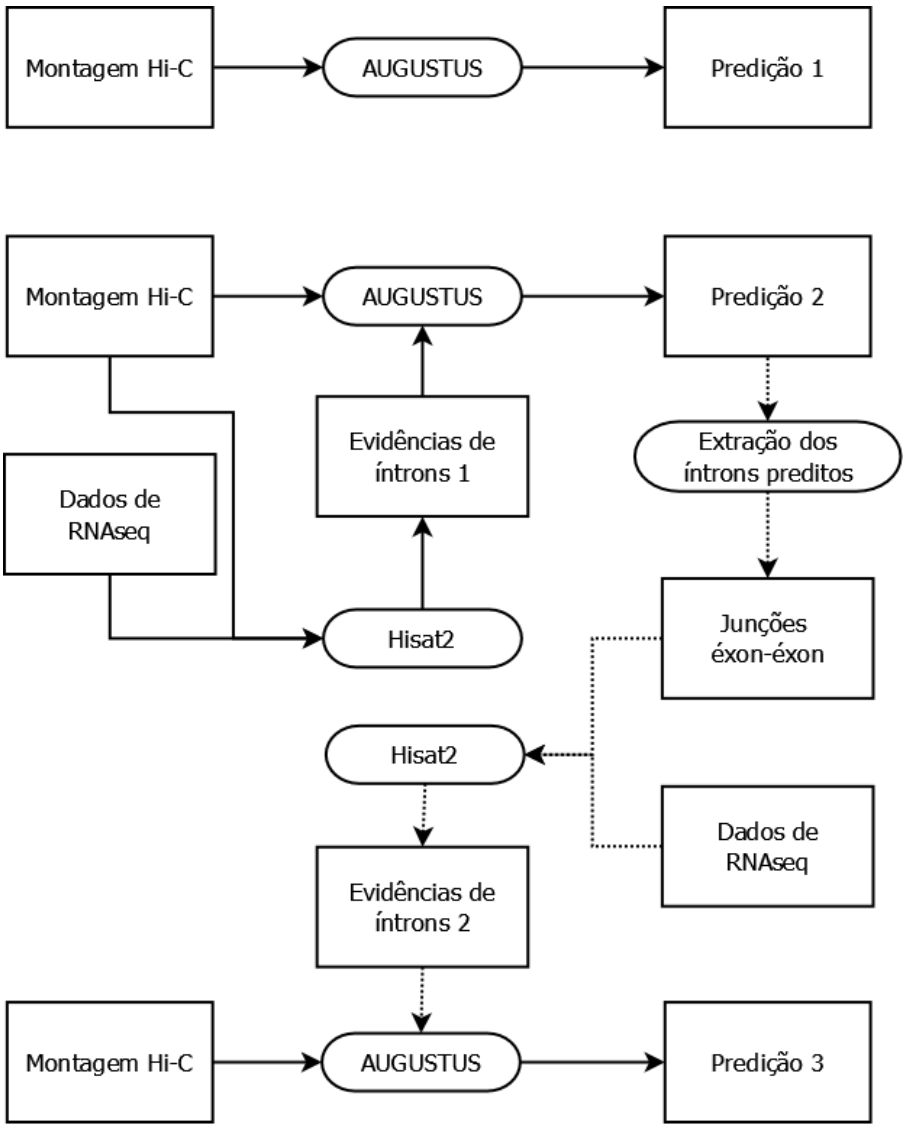

Figura 3.2 - Fluxograma das etapas de predição gênica. Os retângulos indicam as entradas e saídas, as setas contínuas representam o fluxo principal, as formas arrendondadas são o *software*/processamento e as setas pontilhadas indicam comunicação entre etapas diferentes.

Uma segunda predição foi feita se baseando no tópico "Incorporating RNAseq data into AUGUSTUS predictions with BLAT (including iterative mapping)" da página do AUGUSTUS (http://augustus.gobics.de/binaries/readme.rnaseq.html), mas, ao invés do BLAT, usamos o hisat2. Resumidamente, este tópico utiliza evidências de íntrons, obtidas pelo alinhamento de dados de RNAseq, para gerar uma predição parcial que, juntamente com novas evidências intrônicas, originará a predição definitiva. Segundo o criador do protocolo, os íntrons são capazes de produzir bons genes preditos, mesmo sem o modelo de região não traduzida (UTR), tornando-os adequados à maioria das anotações genômicas. Devido a isso, é importante que

haja a execução das duas predições (parcial e definitiva), principalmente a definitiva, para garantir que leituras consigam alinhar corretamente.

O procedimento do primeiro bloco consistiu em alinhar as leituras contra o genoma, onde os dados de RNAseq *paired* foram também alinhados como *single* para tentar aumentar o aproveitamento das leituras. Os arquivos BAM gerados foram ordenados e filtrados, mantendo apenas alinhamentos únicos, e então utilizados para gerar o arquivo de evidências de posições de íntrons em formato GFF. Com as evidências de íntrons geradas, elas foram utilizadas pelo AUGUSTUS para realizar a predição gênica intermediária. O segundo bloco utiliza os genes da predição intermediária para originar sequências de junção éxon-éxon e selecionar as leituras que as confirmam. Este passo é relevante para que apenas os íntrons de genes possivelmente codificadores e com evidências de sequenciamento de transcriptoma fossem utilizados como evidência para a predição, porém as evidências de éxons em si não foram utilizadas. Portanto, as junções éxon-éxon foram alinhadas pela mesma metodologia contra os mesmos dados de RNAseq anteriores, e mais uma vez novas evidências foram geradas para enfim perfazer a predição final.

3.5.2 Avaliação da predição gênica

O software BUSCO v4.1.4 (82) foi utilizado para avaliar a completude das predições gênicas geradas e as já disponíveis para *R. prolixus* de modo a permitir uma comparação de qualidade entre elas. O modo de avaliação utilizado foi o de proteínas e a linhagem escolhida para avaliar a completude foi a ordem *Hemiptera*.

As ausências potenciais de genes também foram buscadas com base nos dados de RNAseq, onde observou-se quantas regiões do genoma tiveram cobertura das leituras (*reads*) mas sem predição gênica. Esta metodologia foi desenvolvida neste trabalho, assim como os *scripts* em Python usados. A metodologia foi feita da seguinte maneira: Os dados de RNAseq foram alinhados contra o arquivo fasta das regiões codificadores dos genes preditos (CDS) utilizando o Hisat2 e seguindo o mesmo pós-processamento dos BAMs do protocolo do AUGUSTUS. O BAM final dessa etapa foi processado pelo samtools depth para calcular a cobertura média. Os dados de RNAseq também foram alinhados com o genoma buscando regiões sem predição gênica, maiores que 500 nucleotídeos e com cobertura igual ou superior à média das CDS.

3.5.3 Comparação das classes de proteínas entre as predições de R. prolixus e anotação automática

A anotação das classes das proteínas preditas foi feita pelo PANTHER 13.0 (83) com o objetivo de verificar se havia alguma diferença estatística entre as predições. De maneira resumida, utilizou-se o *script* pantherScore2.1.pl para comparação das proteínas preditas com o banco de dados, o resultado foi tratado por uma rotina Python para remover as duplicatas do mesmo gene contra diferentes entradas do banco. O resultado foi analisado pelo site do PANTHER (http://www.pantherdb.org) através da opção "*PANTHER Generic Mapping e Statistical Overrepresentation test*". Finalmente, as classes proteicas foram obtidas para a predição 1.3 e 3.4 resultando em uma tabela para cada. Todas as predições foram confrontadas com a predição Hi-C (P13) para que fosse possível verificar entre elas as diferenças nas classes proteicas. Além disso, o Interproscan v5.52-86.0 (84) foi utilizado para fazer a anotação das proteínas preditas pelo AUGUSTUS para a disponibilização no navegador, os bancos de dados usados nessa análise foram SUPERFAMILY (85), CDD (86), Pfam (87) e PANTHER.

3.5.4 Análise de expansão e contração de famílias gênicas na predição Hi-C

As novas proteínas preditas foram avaliadas utilizando o programa Orthofinder v2.5.4 (88) que gera uma série de resultados incluindo grupos ortólogos e árvores filogenéticas. Cinco espécies de insetos foram selecionadas para compor o conjunto de proteínas a ser processado pelo *software*, onde para todos os casos, apenas o maior transcrito de cada gene foi analisado. Dentre essas espécies estavam, *Acyrthosiphon pisum* e *Cimex lectularius*, ambos pertencem à ordem *Hemiptera*, sendo um herbívoro e outro hematófago, respectivamente. *Aedes aegypti* e *Pediculus humanus* foram escolhidos por serem insetos hematófagos e, geralmente, vetores de doenças. A última espécie foi *Glossina morsitans* que também é hematófaga e, semelhante a *R. prolixus*, é capaz de transmitir um protozoário do gênero *Trypanosoma*. Os grupos ortólogos obtidos foram submetidos ao Interproscan para que fosse possível identificar qual família proteica estava representada em cada grupo (considerando o IPR mais frequente). Esse resultado foi submetido a um *script* Python que unia os grupos pela anotação e em seguida fazia a extração das expansões e contrações. As expansões de famílias gênicas para *Rhodnius* foram consideradas apenas quando este inseto era responsável por

pelo menos 51% das proteínas presentes naquele grupo. Já as contrações gênicas foram obtidas quando *R. prolixus* representava no máximo 10% do grupo sendo que pelo menos 4 das outras 5 espécies deveriam conter alguma proteína agrupada. Qualquer grupo anotado como elemento transponível ou com uma diferença sutil entre o número de proteínas foi removido da análise. Por fim, os resultados mais relevantes numérica e biologicamente foram curados manualmente. As ausências em P13 foram recuperadas do resultado do BUSCO.

3.6 Mapeamento dos genes das predições antigas na montagem 3.0.3 Hi-C

3.6.1 Remoção de redundância

Para cada predição gênica anterior foi feita uma renomeação, onde cada CDS recebeu um número que permitiu identificar a predição de origem. Posteriormente, um *script* em Python foi utilizado para remover os genes que continham alguma base não identificada (N) na sua sequência. A remoção da redundância de todas as CDS, de todas as predições gênicas, foi feita com o programa Cd-hit v4.6, em 99% de identidade.

3.6.2 Remoção de genes potencialmente quiméricos e fragmentados

Genes quiméricos e fragmentados que podiam estar no conjunto de genes proveniente da etapa do Cd-hit foram removidos baseados em análise de similaridade a nível proteico (DIAMOND v0.9.22) (89) contra o banco de dados UniRef90 (90). Apenas os alinhamentos que cobriram pelo menos 80% tanto do gene predito como da proteína do banco de dados foram aceitos. Um *script* em Python foi utilizado para filtrar o resultado proveniente do DIAMOND, a lógica da rotina consistia em utilizar a fórmula de Rost (Figura 3.3) para determinar se um alinhamento entre duas proteínas é confiável no que diz respeito ao grau de identidade *versus* comprimento e potencial de serem homólogas (91). Dessa forma, há uma melhor separação entre os alinhamentos de proteínas similares e os alinhamentos de proteínas que não possuem relação entre si, mantendo apenas resultados com maior qualidade por conterem similaridade com sequências de um banco de dados como o uniref90.

Figura 3.3 - Fórmula extraída do artigo de Rost para verificar a significância do alinhamento entre proteínas. p^l = valor de corte de identidade, n = distância em porcentagem da curva original HSSP, L = número de resíduos alinhados.

A fórmula de Rost representa, de maneira simplificada, um gráfico exponencial negativo onde o eixo y é o percentual de resíduos idênticos, "$p^l(n)$", e o eixo x é o número de resíduos alinhados, "L". O valor de "n" permite alterar o platô mínimo da curva, para x em torno de 300, isto é, modificando o percentual mínimo aceitável de resíduos idênticos (91). Essa fórmula permite uma melhor separação entre verdadeiros positivos (proteínas similares) e falso positivos (proteínas não-similares) e consequentemente aumentando a acurácia. Também foi observado que acima de um valor de corte de 30% de identidade, 90% dos alinhamentos detectados eram entre proteínas realmente similares. Com base nisso, o "n" utilizado no *script* foi 10, assim o valor de corte mínimo girou em torno dos 30% (alinhamentos em torno de 300 aa) para elevar a probabilidade de aceitar apenas alinhamentos entre homólogos. Os genes não-codificadores de proteínas foram removidos antes desta etapa para prosseguir para o mapeamento.

3.6.3 Mapeamento dos genes não-codificadores

O mapeamento dos genes não-codificadores, ou seja, aqueles não redundantes e sem "Ns", foi feito contra a montagem Hi-C do genoma utilizando-se primeiro um *script* em *Perl* desenvolvido em nosso laboratório que usa o SIM4 v1.0 (92) para mapear os genes. Esse programa faz um BLAST dos genes contra o genoma, de forma a identificar os *scaffolds* que possuam algum *hit*. Em seguida, esses *scaffolds* passam pelo SIM4, onde é feita a tentativa de mapear os genes previamente alinhados. Por fim, os mapeamentos são classificados em: 1) perfeito único, alinhamento único do códon de início ao códon de término e com pelo menos 99% de identidade; 2) repetitivo, múltiplos alinhamentos perfeitos; 3) fragmentado, alinhamentos com pelo menos 99% de identidade que não contemplam todo o gene. Os alinhamentos completos (perfeitos únicos) foram aceitos e os arquivos GFF gerados. Os genes que não geraram resultados satisfatórios passaram por uma segunda rodada de mapeamento com o Exonerate v2.2.0 (93) e a mesma lógica do procedimento anterior.

3.7 Conciliação da predição gênica atual com os genes codificadores antigos

Um *script* em Python foi desenvolvido para realizar duas tarefas distintas: 1) Conciliar CDS “ouro”, oriundas de um transcriptoma com uma predição gênica (modo predição contra transcriptoma) e 2) Conciliar uma predição gênica anterior com uma predição gênica atual (modo predição contra predição). Ambos os modos trabalham de maneira semelhante, entretanto no primeiro modo os dados de transcriptoma tem primazia sobre a predição, já no segundo, a predição mais recente tem prioridade sobre as antigas. A entrada de ambos os modos consiste em arquivos fasta de proteínas (as CDS são convertidas para peptídeos) e arquivos GFF no formato GFF3. Primeiramente, um alinhamento é feito utilizando o DIAMOND para obter as identidades e coberturas entre os genes *query* e os genes *subjects* que são filtrados pela fórmula de Rost igualmente à etapa dos genes antigos. Apenas os alinhamentos específicos (sem *gaps* e *mismatches*) são selecionados para a etapa de classificação onde estes são identificados. Feita a classificação, começa uma etapa de tratamento, usando o Exonerate, para selecionar a melhor estrutura gênica dentre a dupla alinhada. Finalmente, é feita uma conciliação dos genes *query* com os genes *subject* resultando em um arquivo GFF3 e um arquivo fasta.

3.7.1 Modo predição x transcriptoma

Nesse modo os alinhamentos são classificados em: i) idêntico, onde o alinhamento tem 100% de identidade e cobertura para ambos (gene predito e CDS idênticos); ii) similar, 100% de identidade, sem *gaps* e com cobertura de no mínimo 80% para ambos (gene predito e CDS similares); iii) duplicação 1xn, que contempla alinhamentos onde existem mais de um CDS *subject* para o mesmo gene predito *query* (parálogos ou duplicações), nela é considerada 100% de identidade, sem *gaps* e com cobertura de no mínimo 80% para cada alinhamento; iv) duplicação nx1, que representa o inverso, mais de um gene predito *query* para o mesmo CDS *subject* (parálogos ou duplicações), nela é considerada 100% de identidade, sem *gaps* e com cobertura de no mínimo 80% para cada alinhamento; v) predito quimérico, contém os alinhamentos onde mais de um CDS *subject* alcançam pelo menos uma cobertura de 80% cada, e estão contidos em um gene predito *query* (quimérico), em todos os alinhamentos é considerado 100% de identidade e

ausência de *gaps*; e vi) predito fragmentado, que possui *queries* (genes preditos) alinhando no mesmo CDS *subject* e isso resulta em um alinhamento “completo”. A mesma lógica de completude (>= 80% de cobertura) é aplicada individualmente para os *queries* e para o *subject* (como a soma dos alinhamentos), em todos os casos é considerada identidade de 100% e sem *gaps*. Com os alinhamentos diagnosticados, apenas as classes “similar”, “predito quimérico” e “predito fragmentado” passam pela etapa de tratamento.

O tratamento e conciliação na classe “similar” consiste em alinhar o CDS oriundo do transcriptoma contra o genoma na região do gene predito (incluindo as regiões intergênicas anterior e posterior) com o programa Exonerate. Se o resultado do Exonerate identificar um gene com 100% de cobertura e 100% de identidade em relação ao CDS haverá uma troca, removendo o predito gênico do AUGUSTUS e inserindo o do Exonerate. Se o Exonerate predisser um gene na região com cobertura e identidade, em relação ao CDS, entre 99-80%, a anotação do gene anteriormente predito pelo AUGUSTUS será modificada no arquivo GFF3, adicionando-se o rótulo “similar to” e o nome do CDS em questão.

A classe “predito quimérico” é tratada e conciliada da seguinte maneira: 1) alinhar os CDS do transcriptoma contra o genoma na região do gene predito (incluindo as regiões intergênicas) de maneira semelhante a classe “similar”. Se o resultado do Exonerate predisser um gene, com pelo menos 80% de cobertura e identidade, para cada uma de todas as CDS que alinharam contra um determinado gene quimérico, esses genes avançam para a etapa de verificação de qualidade; 2) verificar se há *overlap* (sobreposição) entre os preditos gênicos, caso não haja, os dois preditos entram no resultado final enquanto o gene quimérico do AUGUSTUS é removido. Se o um dos preditos do Exonerate está completamente contido dentro do outro, apenas o maior é mantido, assim este se manterá no GFF3 final enquanto o gene quimérico do AUGUSTUS é deletado. Por fim, se os genes oriundos do Exonerate apresentarem um *overlap* entre si, algumas validações são feitas para decidir qual deles prevalecerá no resultado em detrimento do gene do AUGUSTUS. Primeiro, a estrutura gênica (presença de códon de iniciação e terminação) é avaliada, em seguida identifica-se qual predito possui o maior *score* normalizado proveniente do Exonerate. Caso haja um empate nos dois critérios, o predito que aparece primeiro, considerando a posição no genoma, é mantido. Entretanto, caso

ambos possuam uma estrutura gênica ruim, o gene quimérico do AUGUSTUS é preservado.

A classe “predito fragmentado” tem seu tratamento e conciliação feitos da seguinte forma: é verificado se os genes fragmentados do AUGUSTUS estão no mesmo *scaffold*/*contig* e se estes encontram-se adjacentes no genoma. Se sim, o alinhamento da CDS do transcriptoma é feito contra a região do genoma que contempla o início do primeiro fragmento do AUGUSTUS até o término do último fragmento (incluindo as regiões intergênicas). Caso o Exonerate identifique um gene com pelo menos 80% de cobertura e identidade, este gene vai ser colocado no lugar dos fragmentos preditos pelo AUGUSTUS.

O resultado final desse modo é um arquivo GFF3 baseado no próprio GFF3 da predição, porém atualizado com os genes provenientes do transcriptoma. A partir desse arquivo, é gerado um arquivo fasta com as sequências codificadores. Para exemplificar melhor as mudanças feitas por esse *script*, um gene predito quimérico e um fragmentado foram analisados antes e depois do tratamento. O site do Interpro (https://www.ebi.ac.uk/interpro/) foi utilizado para observar famílias proteicas e domínios conversados e o BLAST foi usado contra o RefSeq (94) para verificar cobertura, identidade e espécie alinhada.

3.7.2 Modo predição x predição

Nesta outra rotina, a interpretação dos resultados do DIAMOND funciona de maneira semelhante. A classe “idêntico” contém os alinhamentos com 100% de cobertura e identidade para ambos os genes, sem gaps. Já os alinhamentos com cobertura de no mínimo 80% para ambos, mas ainda com 100% de identidade e sem gaps, são classificados como “similar”. O grupo “relacionado” agrupa os alinhamentos que não são tão bons quanto os anteriores. Portanto, para aproveitar esses dados, além do uso da fórmula de Rost, foi calculado um valor de corte exigente de *bitscore*, almejando um valor maior que o valor médio do grupo “similar”. Sendo assim, o valor de corte foi definido como a média de *bitscores* dos alinhamentos contidos em “idêntico” e “similar”. O *bitscore* foi escolhido, pois diferente do *E-value*, essa métrica não leva em consideração o tamanho do banco de dados. Logo, essa mensuração atenta em calcular a similaridade independentemente do tamanho da sequência comparada e do tamanho do banco, sendo assim quanto maior o *bitscore* maior a similaridade entre as proteínas

analisadas. Caso o alinhamento esteja dentro do valor de corte, ele entra na classe “relacionado”. “Ausente” é a última classe que contém os alinhamentos com *bitscore* menor do que o ponto de corte, assim como genes de predições anteriores que não tiveram resultados quando comparados com a predição atual.

A etapa de tratamento terá as seguintes ações para os genes classificados: a) Os “idênticos” têm o nome, originalmente dado pelo AUGUSTUS, substituído pelo nome já existente no gene da predição gênica anterior, além disso uma descrição (“*new version*”) é adicionada. Para a classe “similar” e “relacionado” os genes da predição atual terão incluída uma descrição (“*similar to*” ou “related to” mais o nome do gene da predição anterior) mostrando que são similares (ou relacionados) a determinados genes já preditos anteriormente.

A etapa de tratamento também consiste na utilização do Exonerate para buscar genes no genoma, porém aqui somente os genes de predições anteriores que caíram no grupo “ausente” são tratados. Diferentemente do modo anterior, é dado ao programa Exonerate o genoma completo para a realização da predição dos genes ausentes. Os alinhamentos que mostrarem mais de 80% de identidade e cobertura entre o resultado do Exonerate com o gene oriundo da predição anterior são aceitos, mas somente se este resultado não tiver sobreposição com nenhum gene já predito pelo AUGUSTUS.

Adicionalmente, genes das predições anteriores que foram classificados no grupo “ausente”, e que não apresentaram resultados de Exonerate satisfatórios contra o genoma atual (versão Hi-C) foram organizados para compor uma complementação do genoma. Os *scaffolds* de versões antigas de montagem do genoma foram mascarados com Ns para deixar somente os genes classificados como ausentes, seus GFFs editados para refletir as mesmas mudanças, e foram agrupados em um conjunto de dados acessório, que será fusionado com o genoma atual (versão Hi-C) e sua predição, feita por este trabalho. Logo, trechos das montagens anteriores do genoma são adicionadas a montagem atual, com seus respectivos GFFs.

3.8 Disponibilização do conjunto total de genes

O navegador de genomas JBrowse (69), foi usado na sua versão de desenvolvedor 1.16.8 pois esta permite a instalação de plugins. O navegador está

instalado em um servidor Intel Core i3-2100, 12Gb RAM, disco de 2Tb, Linux Ubuntu 18.04.5 LTS com Apache 2.4.29, no Laboratório de Bioinformática, Instituto de Química da Universidade Federal do Rio de Janeiro (http://www.bioinfo.iq.ufrj.br/genomes). Todas as versões de montagem de genoma, juntamente com todas as predições de *R. prolixus* foram incluídas. A predição gênica da versão Hi-C está disponível, assim como o conjunto genômico completo de *Rhodnius*, que possui o genoma Hi-C junto com *scaffolds* de genes ausentes na Hi-C. Nosso navegador disponibiliza as versões antigas e a versão mais nova, indisponíveis no repositório oficial do Vectorbase (https://vectorbase.org/vectorbase/app/jbrowse?data=/a/service/jbrowse/tracks/default).

4 RESULTADOS

4.1 Distribuição dos transcritos ao longo das predições de *R. prolixus*

Para comparar as duas montagens anteriores do genoma de *R. prolixus,* seus transcritos compartilhados e exclusivos foram identificados. Foi feita uma clusterização (Figura 4.1) onde a maioria dos *clusters* (86%, 14.520) continha transcritos de ambas as predições e apenas 12% (2.070) eram exclusivos das predições 1.x. Além disso, ao longo das predições 3.x, apenas 359 (2%) transcritos foram descobertos.

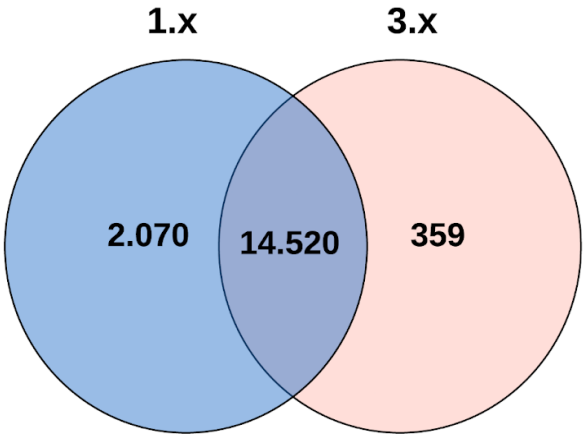

Figura 4.1 - Diagrama de Venn de grupos gerados pelo Cd-hit. Inicialmente foram removidos parálogos e duplicações (95% de identidade) dentro de uma predição, além de todos os transcritos com bases indefinidas (Ns). As predições 1.0 a 1.3 foram agrupadas em "1.x" e o mesmo foi feito com as predições da versão "3.x" (3.1 a 3.4) para comparação em 95% de identidade

4.2 Classificação dos elementos repetitivos

O RepeatScout permitiu a identificação das sequências repetitivas consenso presentes na montagem Hi-C. O arquivo fasta gerado continha 4.485 sequências que foram classificadas com o Hmmsearch e o DFAM (Tabela 4.1).

Tabela 4.1 - Classificação das sequências repetitivas consenso da montagem Hi-C de *R. prolixus* pelo Hmmsearch e o DFAM. (continua)

Classe	n	%
DNA transposon	1498	33,4

LINEs	1208	26,9

Tabela 4.1 - Classificação das sequências repetitivas consenso da montagem Hi-C de *R. prolixus* pelo Hmmsearch e o DFAM. (conclusão)

LTRs	369	8,2
SINEs	125	2,8
Outros	73	1,6
Retroposon	8	0,2
Sem classificação	1027	22,9
Desconhecidos	177	4,0
Total	4485	100

LTR: *Long terminal repeat*/ LINE: *Long interspersed nuclear element*/ SINE: *Short interspersed nuclear element*. A classe “Outros” engloba repetições simples, satélites e alguns tipos de RNAs. A classe denominada "Sem classificação" refere-se a repetições com entrada no banco de dados, mas sem classificação definida. A classe denominada "Desconhecidos" faz referência a repetições que não tiveram correspondência no banco de dados.

Entre os classificados e conhecidos, DNA transposon foi a mais encontrada e em segundo lugar ficou LINE, seguida de LTR, SINE, Outros e, por fim, Retroposon. Também foi possível identificar famílias específicas dentro das classes citadas acima, sendo cada uma delas organizada com suas respectivas famílias em tabelas separadas (Tabela 4.2-4.5).

Tabela 4.2 - Famílias de DNA transposons encontradas nas repetições consenso da montagem Hi-C. (continua)

Família	**n**	**%**
hAT	671	44,9
Tc1-Mariner	383	25,6
PIF-Harbinger	120	8,0
Helitron	66	4,4
PiggyBac	35	2,3
CACTA	32	2,1
Kolobok	26	1,7
Crypton	14	0,9
P	13	0,9
IS3EU	12	0,8
Sola	9	0,6
Merlin	8	0,5

Tabela 4.2 - Famílias de DNA transposons encontradas nas repetições consenso da montagem Hi-C. (conclusão)

Maverick	6	0,4
Zisupton	6	0,4
Zator	3	0,2
Dada	2	0,1
Ginger	2	0,1
Academ	1	0,1
Novosib	1	0,1
Inespecífico	75	5,0

Dentre todas as classes de repetições, os DNA transposons apresentaram a maior diversidade. A família hAT foi a mais observada, correspondendo a 44,8% do total, seguida pelo Tc1-Mariner com 25,6% e do PIF-Harbinger com 8%. Os 21,6% restantes equivalem a famílias como Helitron, PIggyBac, CACTA, Kolobok e Crypton. Para os LTRs, apenas 3 famílias foram identificadas, sendo Gypsy a com maior representatividade (13,8%), sucedido por Ty1-Copia (3,3%) e Bel-Pao (3%). Entre todas as classes de *repeats*, os LTRs apresentam o maior número agrupado em Inespecífico.

Tabela 4.3 - Famílias de LTRs encontradas nos repeats consenso da montagem Hi-C.

Família	**n**	**%**
Gypsy	51	13,8
Ty1-Copia	12	3,3
Bel-Pao	11	3,0
Inespecífico	295	79,9

Já para os LINEs um total de 8 famílias foram identificadas, das quais 46,1% corresponde a RTE, 24,4% para CR1 e 15,3% para R1. As famílias remanescentes foram L1, Penelope, R4, Proto2 e R2, sobrando apenas 3 repetições sem família definida.

Tabela 4.4 - Famílias de LINEs encontradas nos repeats consenso da montagem Hi-C.

Família	n	%
RTE	557	46,1
CR1	295	24,4
R1	185	15,3
L1	99	8,2
Penelope	59	4,9
R4	6	0,5
Proto2	2	0,2
R2	2	0,2
Inespecífico	3	0,2

Finalmente na classe SINE foi observada uma prevalência das famílias tRNA (47,2%), MIR (27,2%) e Alu (9,6%). As dez ocorrências restantes pertencem às famílias ID, U, 5S e B2.

As outras famílias de elementos repetitivos que não foram mostradas em tabelas correspondem a L2-derived e L1-dep para retroposon. No grupo Outros foram reunidos repetições simples, satélites e alguns tipos de RNAs.

Tabela 4.5 - Famílias de SINEs encontradas nos repeats consenso da montagem Hi-C.

Família	n	%
tRNA	59	47,2
MIR	34	27,2
Alu	12	9,6
ID	3	2,4
U	3	2,4
5S	2	1,6
B2	2	1,6
Inespecífico	10	8,0

4.3 Alinhamento dos dados de RNAseq

Os dados de RNAseq (Tabela 4.6) foram alinhados separadamente com a versão Hi-C do genoma usando o programa bowtie2 e o hisat2. Praticamente todos os dados obtiveram boas taxas de alinhamento no bowtie2, tanto no modo *single* como *paired*, exceto dos dados de gônadas (47,4%). Apesar das taxas serem menores que as do hisat2, isto é compreensível, uma vez que o bowtie2 foi usado no modo *end-to-end* juntamente com um filtro de até 5% de erro. Esses parâmetros acabaram reduzindo o número de leituras alinhadas, porém garante que regiões verdadeiramente codificadores possam ser visualizadas no navegador de genoma. Já para o hisat2, as taxas de alinhamento foram ótimas (73,5-96%) em ambos os modos. No caso do hisat2, um alinhamento local era o suficiente para gerar evidências para o AUGUSTUS, sendo assim, não houve a necessidade de se ter aumentada a restrição na função de *score*.

Tabela 4.6 - Dados de RNAseq usados no alinhamento com o genoma pelo programa bowtie2 e hisat2.

Origem	***Acession number***	**Taxa de alinhamento *single* Bowtie2 (%±DP)**	**Taxa de alinhamento *paired* Bowtie2 (%±DP)**	**Taxa de alinhamento *single* Hisat2 (%±DP)**	**Taxa de alinhamento *paired* Hisat2 (%±DP)**
Corpo todo em diferentes condições de alimentação e infecção por *T. cruzi*	RNAseq de colaboradores	73,4±1,8	-	86,7±1,8	-
Órgãos quimiossensoriais	ERX1387156, ERX1387157, ERX1387158	73,3±8,4	72,7±8,4	73,5±4,4	75,5±4,2
Antenas	SRX1011769, SRX1011778, SRX1011796	80,8±4,5	80,8±4,5	91,4±2	96±0,4
Gônadas	SRX6380682, SRX6380683	47,4±1,1	47,4±1,1	76,9±0,6	77,1±0,4

Após gerar os arquivos BAM, eles foram movidos para o servidor do navegador de genomas, onde podem ser visualizados, como exemplificado na Figura 4.2.

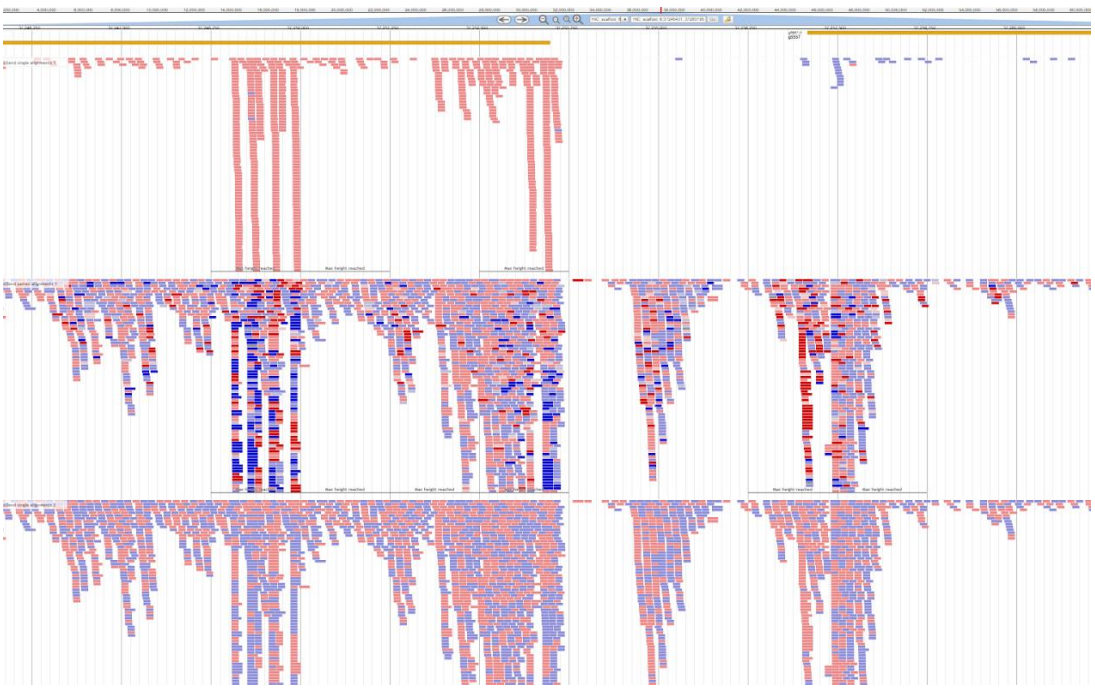

Figura 4.2 - Imagem do navegador de genomas com os genes preditos e os dados de RNAseq alinhados. De cima para baixo, é possível observar a track dos genes preditos (linhas amarelas), do RNA-seq de colaboradores (ninfas), do SRA alinhamento single e do SRA alinhamento paired.

É possível observar regiões bastante populadas de RNAseq e com algum gene predito (extremidades da figura), como também regiões com alinhamentos, mas sem genes (centro da figura). As trilhas (*tracks*) de dados do SRA contêm alinhamentos em maior quantidade, o que já era esperado, pois estes possuem mais bibliotecas que os dados de colaboradores.

4.4 Predição gênica

4.4.1 Predições gênicas preliminares

Diversas predições foram feitas para o genoma de *R. prolixus* com o objetivo de se obter a melhor, ou seja, maior completude e número de transcritos próximos aos já preditos para o inseto. Um resumo de todas as predições se encontra na Tabela 4.7. Primeiramente, foi realizado um treinamento *ab initio* no AUGUSTUS com genes já descritos em *R. prolixus* (P4). Em seguida, foi utilizado o treinamento já disponível no AUGUSTUS para este inseto (P7). O treinamento P4 mostrou 35.093 genes e 37.685 transcritos. Já o treinamento interno do próprio AUGUSTUS

gerou a predição P7 com 23.697 genes e 25.876 transcritos, uma redução de 32,5% e 31,3% no número de genes e transcritos se comparada à predição P4 sem impacto considerável da qualidade avaliada pelo BUSCO, onde houve uma pequena perda de 3% de completude (de 95,5 para 92,5%). O treinamento interno foi escolhido como o melhor por exibir uma quantidade menor de genes e transcritos sem grande perda de completude, apesar deste número ainda ser maior do que a média observada nas predições anteriores (Tabela 1.2).

Tabela 4.7 - Histórico de todas as predições preliminares realizadas para a montagem Hi-C de *R. prolixus*.

Predições	Condição	Genes	Transcritos	Completude (%)	BUSCO
P4	Treinamento *ab initio* com OE e RNAseq validado	35093	37685	95,5	v4.1.4
P7	Treinamento interno com RNAseq validado	23697	25876	92,5	v4.1.4
P10	Treinamento interno com RNAseq validado, protocolo otimizado	23982	26222	92,7	v4.1.4

A tabela mostra apenas a predição final para cada tentativa de predição preliminar.

A melhoria do protocolo de predição, utilizando tanto do modo *single* como *paired* no alinhamento dos dados de RNA-seq, levou a uma nova rodada de predição que apresentou um ligeiro aumento do número de genes e transcritos com um sutil aumento (0,2) na completude (P10).

4.4.2 Predição gênica propriamente dita

Após a escolha do treinamento e do protocolo de mapeamento dos dados de RNA-seq a predição propriamente dita foi realizada com a adição do parâmetro *--softmasking=1* na linha de comando do AUGUSTUS, como já mencionado nos métodos. E assim, a predição P13 apresentou 13.129 genes e 15.181 transcritos (Figura 4.3).

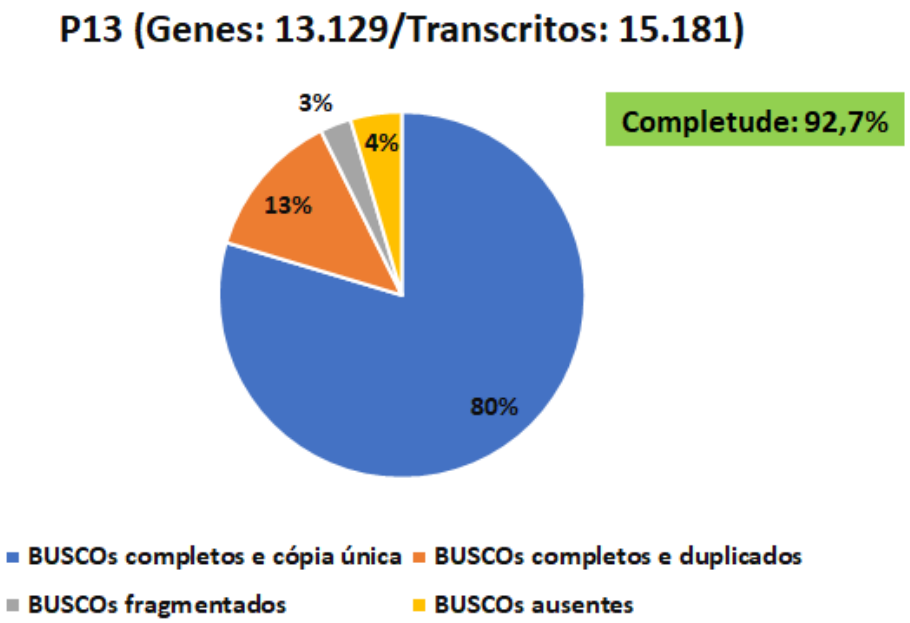

Figura 4.3 - Resultado da avaliação do BUSCO na predição final (P13). BUSCOs completos e cópia única: 1999. BUSCOs completos e duplicados: 328. BUSCOs fragmentados: 69. BUSCOs ausentes: 114.

A predição P13 foi considerada a predição final e apresentou uma redução de 45,3% no número de genes e 42,1% no de transcritos, quando comparada à P10, sem perda de completude (92,7%), sugerindo que apenas genes/transcritos espúrios, provavelmente provenientes de regiões repetitivas, foram descartados e ressaltando a importância do mascaramento do genoma previamente a predição gênica. Essa redução deixou a quantidade de genes/transcritos próxima do que já foi observado nas predições anteriores de *R. prolixus* (Tabela 1.2). Ademais, dentre os 15.181 transcritos, 87,9% deles são suportados por evidências de íntrons, ou seja, existem dados de RNAseq mapeados na região onde esses transcritos foram preditos. Isso indica que grande parte dos transcritos identificados são de alta confiabilidade, pois além de serem preditos pelas probabilidades do modelo treinado do AUGUSTUS, eles também são suportados por dados de transcriptoma.

4.4.3 Comparação com outras predições de R. prolixus

Os resultados do BUSCO para todas as predições de *R. prolixus* estão presentes na tabela abaixo (Tabela 4.8). Como o AUGUSTUS só faz a predição de proteínas, somente os fastas de peptídeos das predições anteriores foram utilizados na análise. A predição 1.0 possui o maior número de proteínas (16.134), enquanto a predição P13 apresenta a maior completude (92,7%), além do menor número de

genes fragmentados (69) e de genes ausentes (114) juntamente com outras predições (1.1, 1.2, 1.3).

Tabela 4.8 - Resumo dos resultados do BUSCO para as predições de *Rhodnius prolixus* e os seus respectivos totais de proteínas.

Predição	BUSCOs completos	BUSCOs fragmentados	BUSCOs ausentes	Completude (%)	Proteínas
Predição 1.0	2069	87	354	82,4	16134
Predição 1.1	2298	98	114	91,6	15441
Predição 1.2	2298	98	114	91,6	15441
Predição 1.3	2298	98	114	91,6	15456
Predição 3.1	2292	98	120	91,3	15078
Predição 3.2	2292	98	120	91,3	15078
Predição 3.3	2292	98	120	91,3	15075
Predição 3.4	2305	93	112	91,8	15106
Predição P13 (AUGUSTUS)	**2327**	**69**	**114**	**92,7**	**15181**

Além de comparar as predições ao nível de completude e número de proteínas, a análise da distribuição dos transcritos 95% idênticos pelas predições foi refeita incluindo P13 (Figura 4.4).

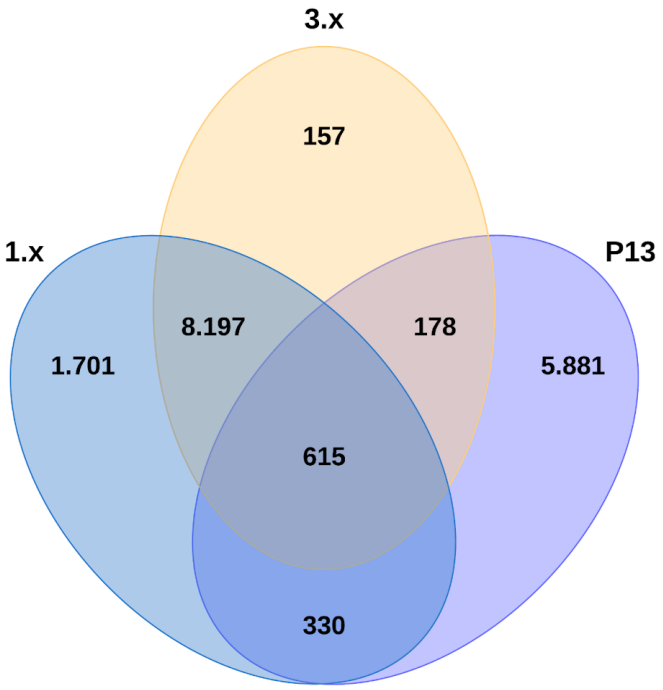

Figura 4.4 - Diagrama de Venn de grupos gerados pelo Cd-hit. Inicialmente, foram removidos parálogos e duplicações (95% de identidade) dentro de uma predição, além de todos os transcritos com bases indefinidas (Ns). As predições 1.0 a 1.3 foram agrupadas em 1.x e o mesmo foi feito com as predições da versão 3.x (3.1 a 3.4) para comparação com P13 em 95% de identidade.

Quanto à exclusividade, a predição P13 possui a maior representatividade (34,47%), seguida das predições 1.x (9,97%) e 3.x (0,92%). As predições 1.x e 3.x apresentam uma maior interação entre si, compartilhando 8.197 grupos, já a P13 interage pouco com as predições 1.x (1,93%) e um pouco menos com as 3.x (1,04%). Por último, apenas 615 grupos (3,61%) são partilhados entre todas as predições.

Após a predição dos genes, foi feita a comparação das predições antigas (1.3 e 3.4) contra a P13, de forma a obter as classes proteicas mais representadas em cada caso e verificar se havia alguma diferença relevante. Dentre as 15.456 proteínas da versão 1.3, cerca de 85,4% delas tiveram algum *hit* contra o banco de dados do PANTHER13. Já para a predição 3.4, das 15.106 proteínas, 86,1% possuíam alguma correspondência no banco de dados. Por último, 86,6% das 15.181 proteínas preditas para P13 encontraram similaridades no PANTHER. Através dessas proteínas com *hits*, obteve-se as classes proteicas limitadas à comparação com P13, porém não havia diferença estática que poderia indicar uma expansão/contração de alguma classe na predição P13 em relação as outras.

4.4.4 Avaliação do BAM da predição

Uma vez que os genes foram preditos, comparou-se P13 com o arquivo BAM dos RNAseq alinhados, para que fosse possível extrair alguma informação a respeito de regiões de alinhamento sem genes em P13. Primeiramente, calculou-se que para os genes preditos havia uma cobertura média dos dados de RNAseq de 958 vezes. Com isso em mente, o BAM do alinhamento dos dados de RNAseq foi analisado buscando intervalos com cobertura igual ou superior a 950 vezes, resultando em 88.824 intervalos. Quando essas regiões genômicas são filtradas por um tamanho de pelo menos 500 nucleotídeos, obtêm-se 1.886 regiões. Finalmente, quando as posições desses intervalos foram comparadas com o GFF da predição, foram diagnosticadas 603 prováveis regiões genômicas sem predição.

4.4.5 Expansões e contrações de famílias gênicas em P13

O OrthoFinder retornou os grupos ortólogos das proteínas dos seguintes insetos: *A. aegypti*, *A. pisum*, *C. lectularius*, *G. morsitans, P. humanus* e *R. prolixus*. Em seguida, o Interproscan classificou esses grupos e os mais relevantes foram curados manualmente para confirmar tal anotação. Abaixo encontra-se a tabela 4.9 com as 4 expansões mais relevantes em *Rhodnius*.

Tabela 4.9 - Algumas expansões de famílias gênicas relevantes em P13.

Descrição	Aaeg	Apis	Clec	Gmor	Phum	Rpro
Triabina/Procalina	0	0	1	0	0	38
OG0000850	0	0	0	0	0	13
OG0002608	0	0	1	0	0	7
OG0004020	0	0	0	0	0	7
OG0004021	0	0	0	0	0	7
OG0009395	0	0	0	0	0	4
Nitroforina	0	0	0	0	0	28
OG0000239	0	0	0	0	0	28
Alérgeno de ácaro, grupo 7	3	2	7	2	1	20
OG0001664	2	2	1	1	1	2
OG0002624	0	0	0	0	0	8
OG0006858	0	0	0	0	0	6
OG0008252	0	0	4	0	0	1
OG0009552	1	0	0	1	0	1
OG0012069	0	0	1	0	0	1
OG0012155	0	0	1	0	0	1
Canal de cloreto	3	3	4	2	2	14
OG0001695	1	2	3	1	1	1
OG0002623	0	0	0	0	0	8
OG0002873	2	1	1	1	1	1
OG0009397	0	0	0	0	0	4

Em comparação aos outros insetos, *R. prolixus* apresentou um grande número de proteínas associadas aos seus hábitos hematofágicos (66 proteínas e 6 grupos ortólogos no total), como a triabina/procalina e nitroforina. Além disso, ele possui uma grande quantidade de alérgeno de ácaro, 20 proteínas ao longo de 7 grupos, e canal de cloreto, 14 proteínas divididas em 4 grupos. As contrações de famílias gênicas foram avaliadas de maneira semelhante (Tabela 4.10).

Tabela 4.10 - Algumas contrações de famílias gênicas relevantes em P13.

Descrição	Aaeg	Apis	Clec	Gmor	Phum	Rpro
Domínio Tudor	10	44	8	10	7	4
OG0000336	0	22	0	0	0	0
OG0001321	1	6	1	0	2	0
OG0002328	1	1	1	3	1	1
OG0002471	1	1	1	3	1	1
OG0007839	1	0	1	1	1	1
OG0007959	0	4	0	0	1	0
OG0008581	3	0	0	1	0	0
OG0008712	1	0	1	1	1	0
OG0008837	1	1	0	1	0	1
OG0008985	0	4	0	0	0	0
OG0009264	0	1	3	0	0	0
OG0009696	1	2	0	0	0	0
OG0011328	0	2	0	0	0	0
Domínio *Paired*	8	8	12	9	8	4
OG0000183	6	5	9	6	5	2
OG0000939	2	3	2	2	2	1
OG0009303	0	0	1	1	1	1

Ao contrário das expansões, as contrações encontradas em *R. prolixus* não apresentaram valores tão distantes dos outros insetos. Apesar disso, foi possível observar que nas famílias do domínio Tudor e do domínio *Paired*, o triatomíneo alcançava aproximadamente metade da mediana de proteínas de cada família.

4.5 Conciliação da predição P13 com transcritos completos confiáveis (Transcriptomas)

A primeira conciliação foi feita usando os genes da P13 contra os 2.990 transcritos oriundos de transcriptoma (73), o que originou a predição P14. Objetivando identificar e corrigir genes quiméricos e fragmentados que poderiam ter sido preditos pelo AUGUSTUS. Nosso script em *Python* resolveu: 10 genes

quimércos, que geraram 20 genes e dois genes quiméricos onde só foi possível resolver uma das partes da quimera, gerando 2 genes não quiméricos (Figura 4.5).

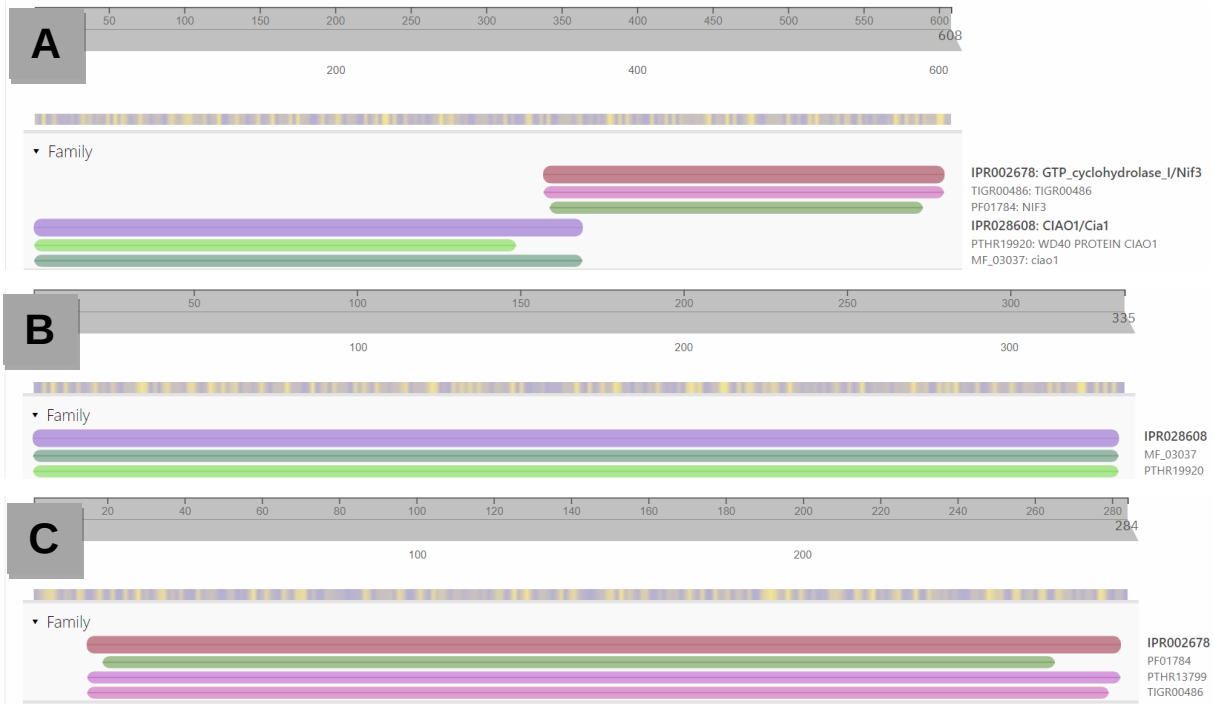

Figura 4.5 - Visualização no Interpro de um gene quimérico antes e após o tratamento do *script* da conciliação. O painel A corresponde ao gene predito quimérico RPHIC01198 com duas famílias identificadas GTP ciclohidrolase (IPR002678) e CIAO1/Cia1 (IPR028608). O painel B refere-se ao transcrito Rp10073 que possui apenas IPR028608. O painel C mostra o transcrito RP10071 com o IPR002678.

O predito quimérico RPHIC01198 teve duas famílias caracterizadas (GTP ciclohidrolase e CIAO1/Cia1), ademais é possível observar uma sobreposição entre as regiões destas famílias. Quando o *script* da conciliação é aplicado, os dois transcritos resultantes (Rp10073 e RP10071) abrangem cada uma das regiões inicialmente observadas, indicando que não houve uma perda funcional. Além disso, tanto a sequência quimérica como a dos transcritos foram avaliadas contra o Refseq (Tabela 4.11).

Tabela 4.11 - Comparação no BLAST de um gene predito quimérico antes e após o tratamento do *script* da conciliação.

Transcrito	*Acession*	Identidade	*Gaps*	Cobertura do *subject*	Organismo	Banco de dados
RPHIC01198	XP_034938017.1	51%	10%	43,4%	*Chelonus insularis*	Refseq
Rp10073	XP_014262244.1	60%	2%	90%	*Cimex lectularius*	Refseq
RP10071	XP_014262243.1	75%	0%	99,1%	*Cimex lectularius*	Refseq

Após o tratamento do gene quimérico (Rp10073 e RP10071), obteve-se *hits* de melhor qualidade, com identidade e cobertura superiores ao observado para o quimérico predito (RPHIC01198). A partir desse resultado é possível inferir que o gene predito é quimérico, uma vez que se nota uma maior similaridade de Rp10073 e de RP10071 em relação aos dados presentes no banco. Também foram identificados 4 genes fragmentados que viraram 2 genes (Figura 4.6).

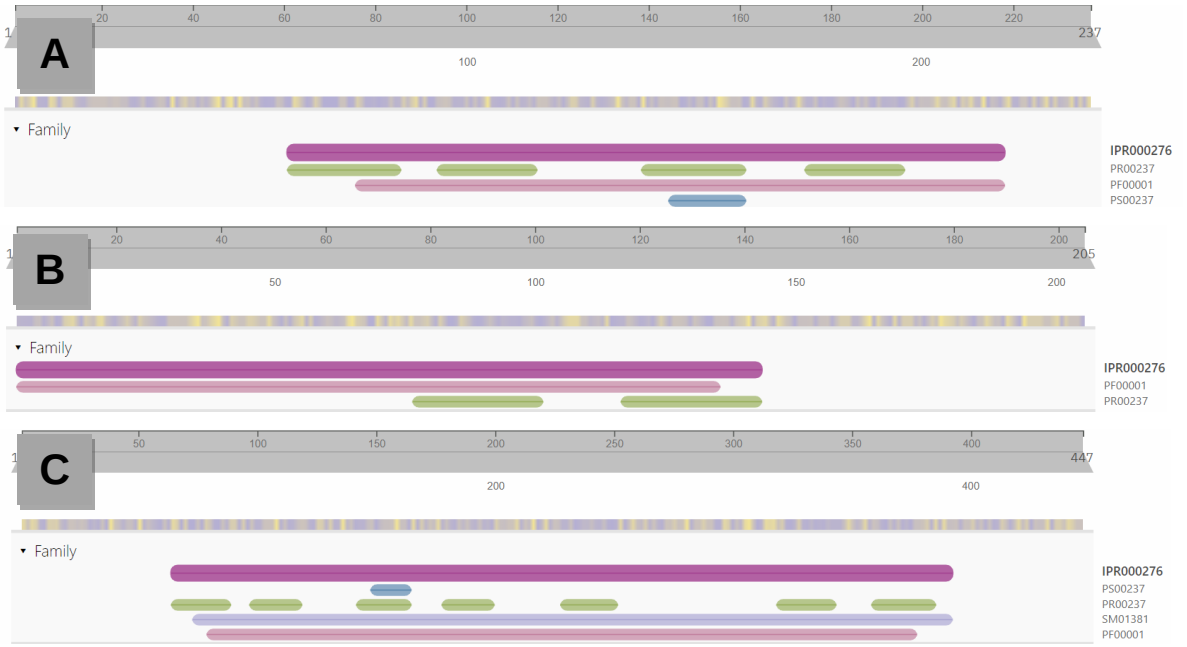

Figura 4.6 - Visualização no Interpro de um gene fragmentado antes e após o tratamento do *script* da conciliação. O painel A mostra o predito fragmentado RPHIC02791 agrupado na família de receptor acoplado a proteína G semelhante a rhodopsina (IPR000276). O painel B corresponde ao gene predito fragmentado RPHIC02790 com a mesma família (IPR000276). O painel C refere-se ao transcrito MF377526.1_cds_ATI14906.1_1 incluído após o tratamento que possui características iguais aos transcritos anteriores.

Os genes preditos fragmentados (RPHIC02790 e RPHIC02791) tiveram sua família (receptor acoplado a proteína G semelhante a rhodopsina) e domínios identificados, assim foi possível observar que substituição pelo transcrito MF377526.1_cds_ATI14906.1_1 não ocasionou nenhuma perda funcional, uma vez que todos os domínios previamente detectados foram mantidos juntamente com a família. Os fragmentos como o transcrito ainda foram avaliados contra o Refseq (Tabela 4.12).

Tabela 4.12 - Comparação no BLAST de genes preditos fragmentados antes e após o tratamento do *script* da conciliação.

Transcrito	*Acession*	Identidade	*Gaps*	Cobertura do	Organismo	Banco de

				subject		dados
RPHIC02790	XP_014254146.1	92%	0,98%	48%	*Cimex lectularius*	Refseq
RPHIC02791	XP_014254146.1	86%	0%	41%	*Cimex lectularius*	Refseq
MF377526.1 _cds_ATI14906.1_1	XP_014254146.1	88%	0,5%	93,8%	*Cimex lectularius*	Refseq

Semelhante ao predito quimérico, o tratamento dos genes preditos fragmentados melhorou a qualidade do *hit* contra o Refseq, já que houve o aumento no percentual de cobertura do *subject* pelo *query*. Este fato reforça a classificação dada pelo *script* que os genes RPHIC02790 e RPHIC02791 são realmente fragmentados. Um resumo dos genes quiméricos e fragmentados substituídos se encontra na tabela 4.13.

Tabela 4.13 - Substituições dos genes preditos quiméricos e fragmentados pelos transcritos.

Transcritos	Genes preditos	% de cobertura do transcrito no gene predito
RP27800, RP20609	RPHIC05963	88, 88
Rp10073, RP10071	RPHIC01198	95, 100
Rp43708	RPHIC08968	87
RP65162, RP71370	RPHIC10731	99, 100
Rp22729, Rp8271	RPHIC01555	96, 100
AY340272.1, Rp97986	RPHIC11164	89, 84
RP21920, RP4090	RPHIC11290	94, 100
Rp10453, KX572140.1	RPHIC07746	85, 91
Rp121414, Rp11206	RPHIC02608	100, 98
RP21418, RP11814	RPHIC05818	97, 100
RP12264	RPHIC08300	85
RP93034, Rp26561	RPHIC01639	100, 100
RP42200	RPHIC06057, RPHIC06058	55, 45
MF377526.1_cds_ATI14906.1_1	RPHIC02790, RPHIC02791	46, 49

As 12 primeiras linhas representam os genes quiméricos e as duas últimas os fragmentados.

Dentre os genes classificados como “similares” (mais de 80% de identidade e cobertura contra os transcritos), houve casos onde um transcrito foi completamente alinhado (100% cobertura e identidade) no genoma pelo Exonerate, na mesma região do gene considerado seu “par”. Neste caso nosso script removeu o predito gênico criado pelo AUGUSTUS, substituindo-o pelo resultado do Exonerate. Foram 505 substituições como esta, melhorando a integridade dos genes da predição. Outros 136 genes ganharam uma descrição indicando similaridade pois possuíam pelo menos 80% de identidade e cobertura em relação com o transcrito comparado com ele. Também houve 603 genes que tiveram seu nome trocado pelo nome do transcrito já que eles possuíam 100% de identidade e cobertura quando comparados entre si. Finalmente, uma análise usando o BUSCO foi realizada para estimar as

diferenças entre a predição P13 e a conciliada com os genes oriundos do transcriptoma (73). A completude se manteve em 92,7%, apesar de neste universo o número de genes completos cair em 2 e o número de fragmentados e ausentes ter aumentado em 1 cada.

4.6 Conciliação da predição P14 com genes antigos

4.6.1 Preparação dos CDS dos genes antigos

Os arquivos fasta das predições anteriores foram reunidos em um só arquivo, totalizando 132.004 transcritos. Esse fasta foi submetido a um *script* que removia transcritos com Ns em sua sequência, o que reduziu o número deles para 131.814. Em seguida foi realizada a remoção de redundâncias com o Cd-Hit, resultando em uma diminuição de 84% na quantidade de transcritos (19.008). Após essa etapa, dois conjuntos de genes foram criados: 1.505 genes não-codificadores de proteínas e 17.503 codificadores de proteínas. Os genes codificadores foram filtrados para remover os possivelmente quiméricos utilizando um *script* desenvolvido em Python baseado em resultados de similaridade (DIAMOND) contra o banco de dados Uniref90. Essa filtragem ocasionou a eliminação de 3.663 transcritos e os 13.840 que sobraram foram encaminhados para a conciliação, junto com os 1.505 genes não-codificadores (Tabela 4.14).

Tabela 4.14 - Resumo do processamento de transcritos já preditos para *R. prolixus*.

Predição	Transcritos
Predição 1.0	16.184
Predição 1.1	17.155
Predição 1.2	17.256
Predição 1.3	17.262
Predição 3.1	16.857
Predição 3.2	15.755
Predição 3.3	15.752
Predição 3.4	15.783
Predição total	132.004
Total sem Ns	131.814

Total não redundante	19.008
Total codificador	17.503
Total não quimérico	13.840
Total não-codificador	1.505
Total alinhado	414

4.6.2 Conciliação dos genes

Inicialmente os transcritos não-codificadores foram alinhados contra a montagem Hi-C inicialmente utilizando o SIM4 e apenas 13 transcritos alinharam perfeitamente, os restantes foram alinhados com o Exonerate e dos 1.492 transcritos, 401 alinharam. Dos 414 genes não-codificadores, 345 mapearam em regiões que não havia genes e foram adicionados na predição gênica após a conciliação.

Os genes antigos codificadores de proteínas foram organizados nas seguintes classes “idênticos”, “similares”, “relacionados” e “ausentes”. Nesse processo 621 genes idênticos tiveram seus nomes já conhecidos usados para nomear os genes presentes na predição atual. Além disso, 3.275 descrições de genes antigos foram adicionadas aos genes novos nas classes “similar” e “relacionado”, de forma a indicar uma relação com um gene já conhecido. Por fim, 2.315 genes antigos identificados como “ausentes” foram adicionados para complementar a predição P14, gerando a predição P15. Como esperado, na análise do BUSCO a completude aumentou de 92,7% para 93,2% como consequência do maior número de genes completos, também houve redução de genes fragmentados e ausentes (Tabela 4.15). Considerando os genes não-codificadores que alinharam na montagem Hi-C, o total de genes ficou em 17.845. Além disso, também foi verificado a presença de códons de iniciação e terminação nos transcritos de P15 em relação as predições antigas (Tabela 4.16). Cerca de 92,1% dos transcritos de P15 apresentavam tanto o códon de iniciação como de terminação, já para as outras predições, o valor máximo alcançado foi de 40,28%. Quanto aos transcritos que continham um códon de terminação precoce, P15 exibia um percentual de apenas 0,42% enquanto para as outras predições esse valor era por volta de 30%.

Tabela 4.15 - Comparação entre a predição P13 e as predições conciliadas.

Predição	BUSCOs completos	BUSCOs fragmentados	BUSCOs ausentes	Completude (%)	Proteínas
Predição P13	2327	69	114	92,7	15.181
Predição P14	2325	70	115	92,7	15.185
Predição P15	2341	69	100	93,2	17.500

Tabela 4.16 - Resumo da presença de códons de iniciação e terminação nas predições de *R. prolixus*.

Predição	Genes íntegros	Somente códon de iniciação	Somente códon de terminação	Nenhum dos códons	Fragmentos
Predição 1.0	5.826 (36,11%)	1.758 (10,9%)	1.738 (10,77%)	1.947 (12,07%)	4.865 (30,15%)
Predição 1.1	6.170 (39,96%)	1.536 (9,95%)	1.509 (9,77%)	1.651 (10,69%)	4.575 (29,63%)
Predição 1.2	6.170 (39,96%)	1.536 (9,95%)	1.509 (9,77%)	1.651 (10,69%)	4.575 (29,63%)
Predição 1.3	6.197 (40,09%)	1.535 (9,93%)	1.509 (9,77%)	1.645 (10,64%)	4.570 (29,57%)
Predição 3.1	6.071 (40,26%)	1.479 (9,81%)	1.452 (9,63%)	1.589 (10,54%)	4.487 (29,76%)
Predição 3.2	6.071 (40,26%)	1.479 (9,81%)	1.452 (9,63%)	1.589 (10,54%)	4.487 (29,76%)
Predição 3.3	6.071 (40,28%)	1.479 (9,81%)	1.452 (9,63%)	1.589 (10,54%)	4.484 (29,74%)
Predição 3.4	5.861 (38,79%)	1.420 (9,4%)	1.392 (9,20%)	1.518 (10,04%)	4.915 (32,57%)
Predição P15	**16.296 (93,12%)**	**657 (3,75%)**	**182 (1,04%)**	**291 (1,67%)**	**74 (0,42%)**

Genes íntegros apresentam tanto códon de iniciação como de terminação. Fragmentos apresentam um ou mais códons de terminação no meio da sequência.

Para ilustrar as modificações realizadas 3 genes foram escolhidos, dois listados na lista de genes do BUSCO e outro não listado e estão mostrados abaixo (Tabela 4.17), com seu comprimento, anotação de função e versão do genoma de origem.

Tabela 4.17 - Exemplo de genes antigos adicionados em P15.

Gene	Comprimento	Anotação	BUSCO	Predição
RPTMP07790	1024 aa	DNA polimerase mitocondrial,	Sim	1.0

		subunidade gamma-1		
RPTMP04087	214 aa	Proteína de ligação ao fator liberador de corticotropina	Sim	1.0
RPRC002158	402 aa	Receptor olfativo de inseto	Não	1.1

Os genes antigos adicionados, tanto os que contribuíram para a completude como os que não, ajudaram a enriquecer P15 de alguma forma. Por exemplo, mesmo RPRC002158 que não esteja incluído no BUSCO, ele possui uma função relevante para o contexto biológico do inseto. Dos 2.315 genes antigos incluídos, apenas 15 deles contribuíram para aumentar a completude de P15 no BUSCO (Tabela 4.18).

Tabela 4.18 - Genes antigos responsáveis pelo aumento da completude em P15.

Gene	Anotação	Predição
RPTMP07790-RA	DNA polimerase mitocondrial, subunidade gamma-1	1.0
RPRC001163-RA	Polirribonucleotídeo nucleotidiltransferase 1, mitocondrial	1.1
RPTMP14330-RA	Proteína não caracterizada LOC106672026	1.0
RPTMP03446-RA	Suposta permease da superfamília do facilitador principal	1.0
RPTMP04087-RA	Proteína de ligação ao fator liberador de corticotropina	1.0
RPTMP09432-RA	CCAAT/proteína de ligação ao intensificador	1.0
RPRC000797-RA	Bursicona	1.1
RPTMP11504-RA	Subunidade de RNA polimerase dirigida por DNA	1.0
RPTMP14931-RA	Subunidade de gama-secretase Aph-1	1.0
RPTMP09686-RA	Proteína ACYPI009643	1.0
RPRC001572-RA	Proteína que interage com a subunidade beta-1 da ATPase transportadora de sódio/potássio	1.1
RPTMP14128-RA	Proteína ACYPI001799	1.0
RPRC002006-RA	Proteína não caracterizada homóloga KIAA1143	1.1
RPTMP07035-RA	Proteína ribossomal semelhante a L14	1.0
RPTMP15104-RA	Suposta hidrolase tipo haloácido desalogenase	1.0

4.7 Jbrowse

Em vista de tornar possível a exploração dos dados de *R. prolixus*, todas as versões de montagem do genoma (V1, V3, Hi-C) e mais a versão completa[1] estão

[1] A versão completa é a versão Hi-C complementada com *scaffolds* contendo somente os genes antigos ausentes na montagem Hi-C.

disponíveis em (http://www.bioinfo.iq.ufrj.br/genomes). Para cada versão genômica todas as predições foram adicionadas ao navegador, desde as antigas até a P15 conciliada.

Toda a organização se baseia nas montagens de genoma, partindo da versão 1.0 até a completa com os *scaffolds* de genes antigos não mapeados nela (Figura 4.7). A versão 1 contém as predições 1.0 até a 1.3, já a versão 3 contém as predições 3.1 até 3.4, enquanto a versão Hi-C contém P13-15, por fim a completa contém também P14-15. É importante destacar que a versão de predição gênica P15, quando acessada pela montagem genômica Hi-C somente mostra os genes presentes nesta montagem, sem os genes antigos ausentes que tiveram seus scaffolds adicionados na versão completa. Estes genes somente estão presentes na versão da montagem completa, na predição P15.

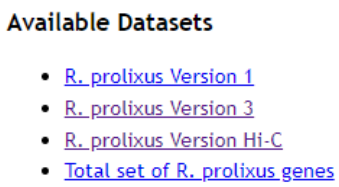

Figura 4.7 - Menu inicial do navegador de genomas.

Acessando um dos *links* é possível visualizar as *tracks* disponíveis do lado esquerdo enquanto a navegação em si se encontra a direita (Figura 4.8). Além de navegar, o Jbrowse permite a pesquisa de nomes de *scaffolds*, genes ou mesmo descrições/anotações (Figura 4.9).

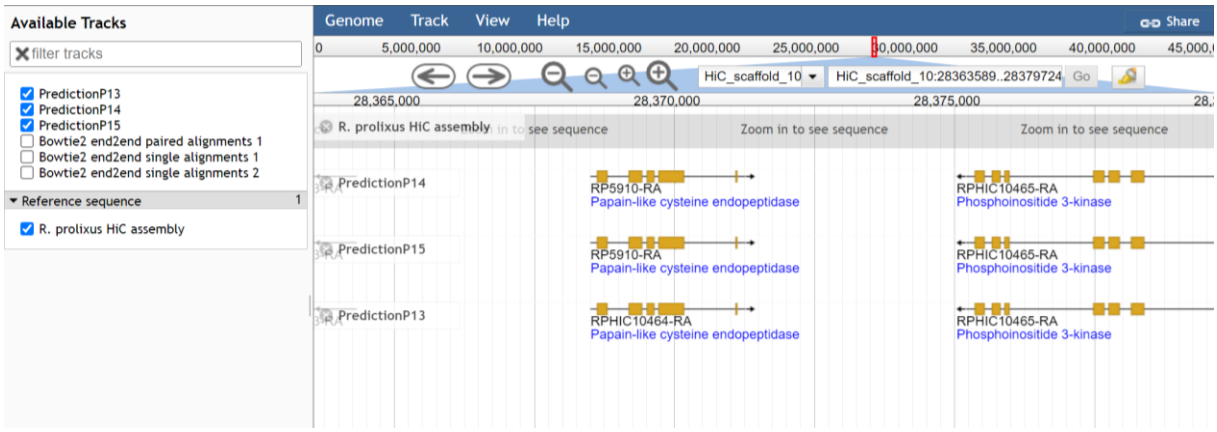

Figura 4.8 - Menu de navegação do Jbrowse.

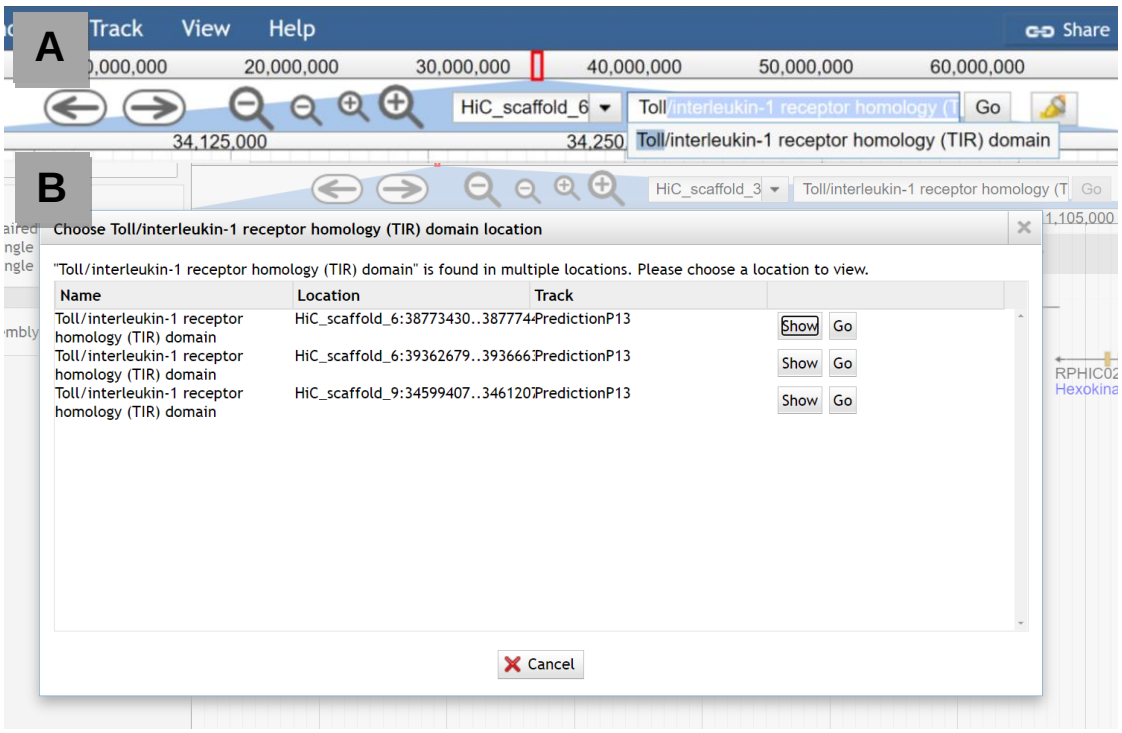

Figura 4.9 - Resultado de uma pesquisa por anotação no Jbrowse. O painel A mostra um termo sendo inserido na barra de pesquisa. Já o painel B mostra o resultado do termo pesquisado.

Além das funções nativas, um *plugin* foi instalado para melhorar a experiência do usuário no momento de explorar a sequência de algum gene/transcrito. Através dele, o usuário pode mostrar ou mascarar diversas regiões gênicas e obter, íntrons, éxons, CDS, regiões anteriores (*upstream*) e posteriores (*downstream*) ao gene. É importante destacar que esta predição não mapeou regiões 3’ e 5’ não traduzidas oriundas de transcriptomas, então estas opções talvez tenham efeito somente em genes antigos importados para esta predição. A janela permite que a sequência em formato fasta seja copiada (Figura 4.10).

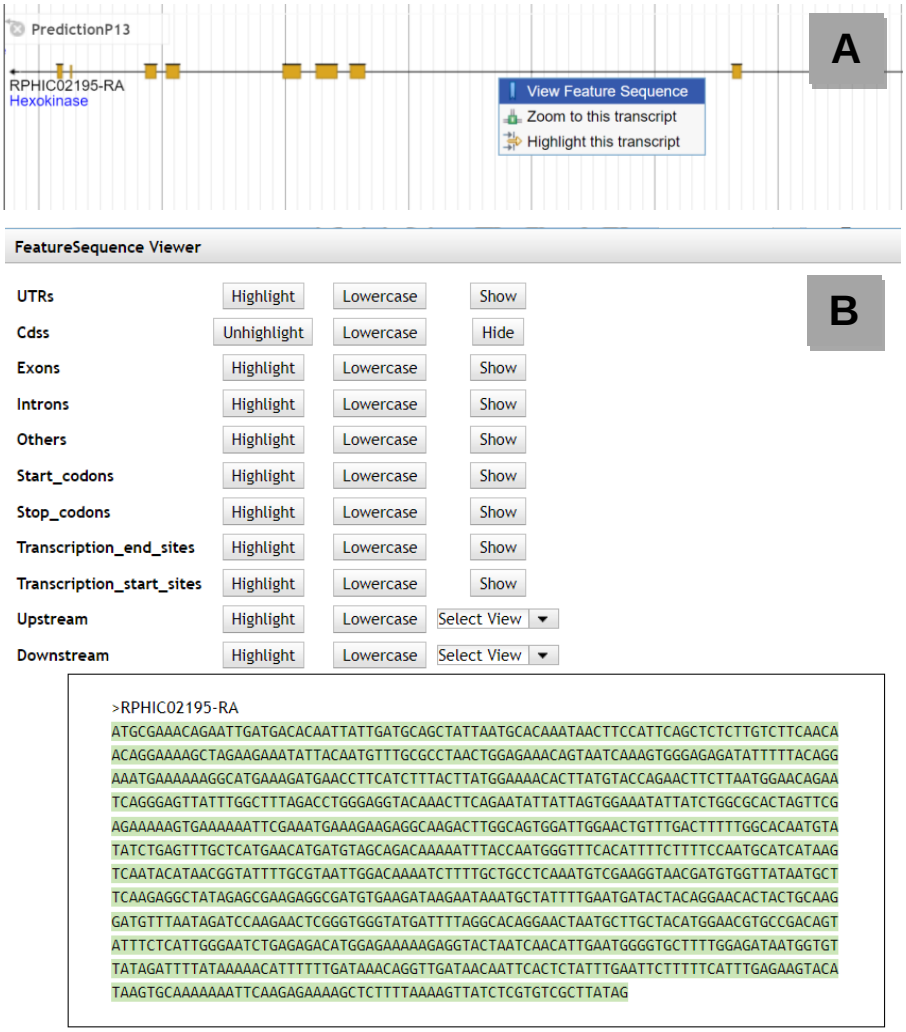

Figura 4.10 - Janela do *plugin FeatureSequence Viewer*. O painel A mostra a opção utilizada para acessar o *plugin*. Já o painel B mostra as possibilidades de exploração do transcrito RPHIC02195-RA, nesta imagem apenas as CDSs estão sendo mostradas.

Por fim, os dados de RNAseq foram reorganizados para dar um suporte biológico aos pesquisadores informando de qual parte do corpo do inseto os dados vieram, sendo possível observar em qual tecido aquele gene/éxon é mais expresso (Figura 4.11).

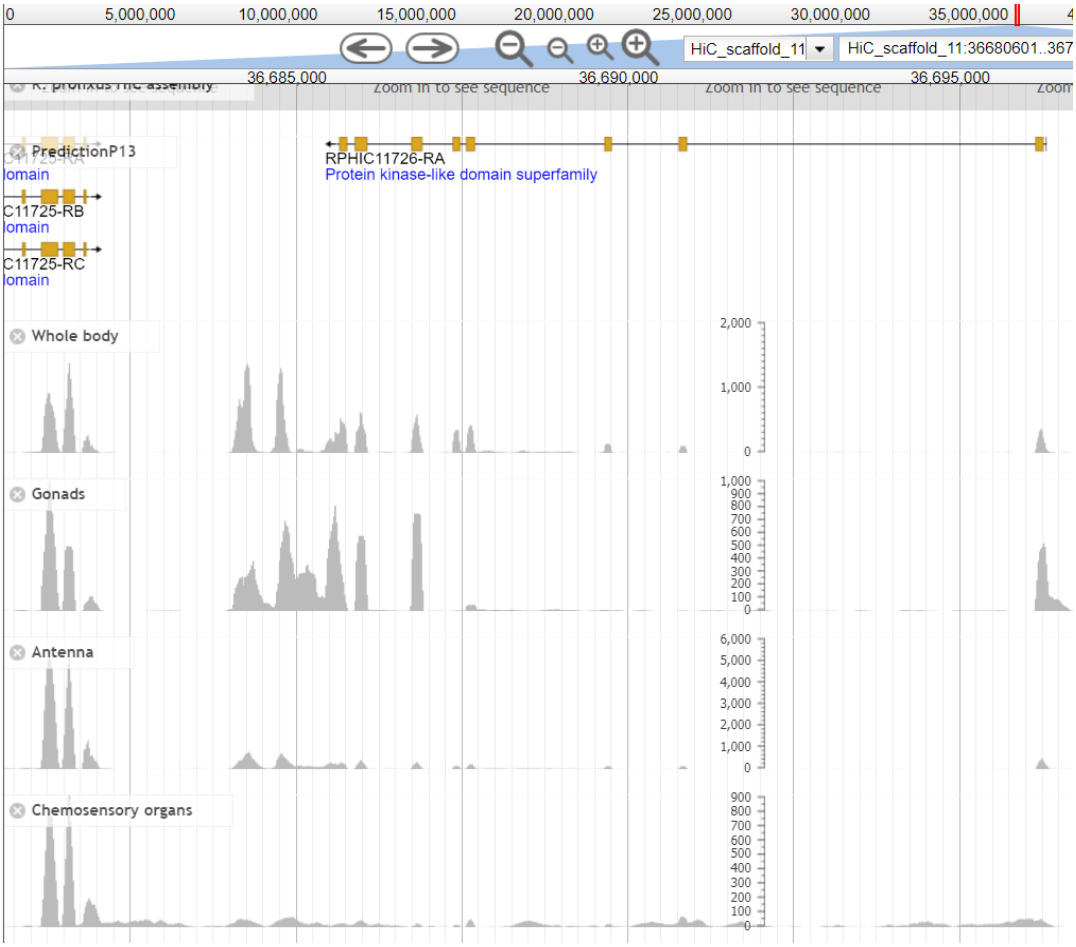

Figura 4.11 - Organização dos dados de RNAseq no Jbrowse.

5 DISCUSSÃO

5.1 Elementos repetitivos

A fim de observar os elementos repetitivos presentes na versão Hi-C do genoma de *R. prolixus* foi feita a classificação das repetições identificadas. Mesmo que não seja bem definida a função desses elementos, já foi observado que eles são responsáveis pela diversidade gênica, mutações e dano ao DNA (95,96). Um total de 4.485 elementos repetitivos, compreendendo 38 famílias, foram encontrados neste trabalho, possuindo uma quantidade maior de famílias que o encontrado no artigo do genoma de *Rhodnius prolixus* (38).

Na montagem 3.0.1, para a classe DNA transposon, a família Tc1-Mariner foi altamente prevalente (38,97,98). Na versão 3.0.3, o Tc1-Mariner continuava como a família de DNA transposon que aparecia em maior quantidade (99). Já na montagem Hi-C, a prevalência de hAT é maior do que a de Tc1-Mariner, o que contraria a literatura de forma geral.

A classe LINE não foi bem definida na versão 3.0.1, porém é possível encontrar as principais famílias encontradas, como Jockey, LoA e CR1-like (38). Na montagem 3.0.3, Jockey ainda é a família predominante, seguida de I e LoA (99). Em contrapartida, na versão Hi-C, Jockey está presente (incluída em R1 juntamente com LoA e I) porém não é a família preponderante. CR1-like assume o lugar de Jockey, acompanhada de R1 e L1.

Na classe LTR, as três famílias já tinham sido identificadas (Gypsy, Ty-Copia e Bel-Pao) na versão 3.0.1 do genoma, com Bel-Pao em maior quantidade quando comparado a Gypsy (38). Já para a versão 3.0.3 (99), foi encontrado uma prevalência maior de Gypsy em relação a Bel-Pao, corroborando com nossos resultados, além da presença das três famílias aqui relatadas.

Os SINEs não estão muito bem descritos em *R. prolixus*, porém foi identificada a família de tRNA para esta classe (100). No presente trabalho, foram identificadas 7 famílias, sendo a mais relevante a tRNA, precedida por MIR e Alu.

Biologicamente, a variação no número de repetições está relacionada a um evento conhecido como *burst of transposition* (explosão de transposição). Esse fenômeno é responsável por uma multiplicação ou aumento súbito de um ou mais

elementos repetitivos. Como consequência há uma drástica reconstrução genômica que normalmente é associada com a formação de novos grupos filogenéticos (101). Os motivos para essa explosão acontecer geralmente são associados a mudanças no ambiente externo, poliploidia, hibridização interespecífica, ou mesmo a domesticação (102–107).

As diferenças aqui relatadas podem ser decorrentes da metodologia, uma vez que o foco desta dissertação não é classificação de elementos repetitivos, o que confere a esses dados um caráter expositivo. Outro fator a ser considerado é que a montagem aqui utilizada foi feita através da técnica de Hi-C. Esta técnica gera leituras maiores para fazer a montagem e, graças a isso, permite uma melhor identificação de regiões repetitivas no genoma, visto que a montagem dessas regiões é um desafio (108). A utilização da técnica de Hi-C na montagem de genomas conhecidos, pode resultar em uma variação no número das regiões repetitivas já identificadas. Por exemplo, leituras menores provenientes de elementos repetitivos, têm uma menor probabilidade de possuírem uma ponta ancorada a uma região não repetitiva, o que leva a propagação do elemento repetitivo ao longo da montagem. Com uma leitura maior, e uma montagem mais eficiente, esse problema é minimizado e a extensão de regiões repetitivas seria praticamente resolvida, consequentemente, alteraria a quantidade de repetições identificadas. Sendo assim, são necessários novos estudos sobre o mobiloma, conjunto de elementos repetitivos, de *R. prolixus* na montagem Hi-C.

5.2 Predição gênica com AUGUSTUS

Já é amplamente conhecido que as proteínas desempenham um papel essencial para o metabolismo da célula e manutenção da homeostase. Portanto, é de suma importância identificar as proteínas de um organismo, seja experimentalmente ou computacionalmente (109). Uma vez que o *Rhodnius prolixus* é um dos principais vetores da doença de Chagas, o conhecimento acerca do seu metabolismo e proteínas envolvidas permitirá avanços no estudo de sua biologia e até a busca por novas medidas de controle vetorial (27,39,110). Em relação à quantidade de transcritos (codificadores de proteína), a predição do AUGUSTUS (P13) se mostrou semelhante às predições mais recentes de *R. prolixus*. Em termos de completude, a predição P13 é a melhor que as anteriores,

alcançando 92,7%. Esse dado conseguiu superar a predição 1.3 de Mesquita *et al* 2015 em quase todos os parâmetros de qualidade do BUSCO, se igualando somente no número de BUSCOs ausentes (38). A predição da montagem Hi-C também alcançou uma completude maior que as predições feitas na versão 3.0.3 do genoma (91,8%), e dessa vez foi inferior apenas no número de BUSCOs ausentes para a predição 3.4.

A qualidade da predição se deve ao AUGUSTUS e um dos pontos decisivos para a sua escolha foi possuir um modelo treinado para predizer genes em *R. prolixus* (53). Dentre os programas de predição disponíveis, ele é um dos que possui o maior número de modelos treinados, cerca de 109, possibilitando a identificação de genes para diversas espécies (53). Quando comparado a outros *softwares* de predição, como o Genscan (111), GeneID (54), GlimmerHMM (112), Snap (55), o AUGUSTUS se mostrou mais eficaz na predição correta de éxons e proteínas (60). Além disso, como vários desses programas, o AUGUSTUS permite a utilização de dados de RNAseq para identificar variantes de *splicing* (emenda). Apesar da infinidade de *softwares* disponíveis para anotação de genomas, o AUGUSTUS continua sendo utilizado e alcançando bons resultados, seja na sua versão *standalone* ou integrado em alguma *pipeline* (113,114).

Apesar da excelente predição obtida, houveram regiões do genoma sem predição que foram identificadas como possíveis éxons/genes devido a quantidade de dados de RNAseq alinhado. Os pseudogenes são considerados cópias defeituosas de genes que acabaram por perder sua função original ao longo do processo evolutivo (115). Então, essas regiões encontradas podem ser pseudogenes que acabam sendo transcritos no genoma do inseto. Em humanos já foi observado que esses elementos gênico tinham uma capacidade de regular supressores tumorais (116). Em *Helicoverpa zea* foi sugerido que alguns dos seus pseudogenes poderia ter uma função de resistência a inseticidas (117), e para *R. prolixus*, foram identificados 677 elementos divididos entre pseudogenes e RNAs na predição 3.3. Outra possibilidade é que tais regiões podem flanquear porções do genoma que ainda não foram completamente identificadas (regiões de Ns), o que levaria ao alinhamento de leituras, porém sem a predição de genes, uma vez que a proteína não preencheria os requisitos necessários para ser predita pelo AUGUSTUS. Logo, é necessária uma investigação posterior dessas regiões para tentar identificar o elemento gênico ali presente.

Após a predição, uma etapa de suma importância foi a anotação gênica que permite a caracterização e identificação dos genes preditos, seja ela feita de maneira manual ou automatizada (118,119). Neste trabalho, o Interproscan (84) foi utilizado para fazer anotação das proteínas preditas pelo AUGUSTUS, das 15.181 proteínas cerca de 12.072 (80%) tiveram sua função caracterizada. Além disso, as anotações das proteínas foram incluídas no GFF da predição, possibilitando o usuário do navegador pesquisar proteínas pela função biológica/molecular, ou mesmo navegar pelo genoma e ter uma noção do que é codificado pelos genes preditos.

O PANTHER (83) foi utilizado para verificar se havia alguma classe proteica aumentada ou diminuída na predição P13 em relação as predições 1.3 e 3.4. Através do enriquecimento funcional, foi possível observar que não havia nenhum viés em P13, ou seja, favorecimento ou não de alguma família gênica. Esse dado mostra que mesmo com as 5.881 proteínas exclusivas de P13 em comparação com as predições 1 (1.0 até 1.3) e as predições 3 (3.1 até 3.4), não há nenhuma divergência significante ao nível funcional/biológico.

Através da análise do Orthofinder foi possível obter um pouco de informação a respeito de expansões e contrações de famílias gênicas na predição P13 de *R. prolixus*. Dentre as expansões, os grupos de triabina/procalina e de nitroforinas foram os mais relevantes, ambos relacionados com hematofagia. Essas proteínas são secretadas na saliva do inseto com o objetivo de interferir no equilíbrio hemostático do hospedeiro facilitando o repasto sanguíneo. De maneira geral, os efeitos mais conhecidos são a vasodilatação mediante transporte e liberação de óxido nítrico, inibição da resposta inflamatória através da ligação com a histamina liberada pelas células do sistema imune e também interferência na coagulação sanguínea via inibição do fator X (120–122). As nitroforinas fazem parte da família das lipocalinas, já as triabinas/procalinas compõe uma família própria que está intimamente relacionada com as lipocalinas, sendo que ambos os grupos pertencem a superfamília das calicinas. Em *R. prolixus*, já foram observados 5 tipos de nitroforinas (NP1-4, NP7), onde elas representavam cerca de 50% do total de proteínas da saliva (123). Enquanto a triabina/procalina foi encontrada em *R. prolixus* como também em outros membros da subfamília Triatominae, porém essa proteína não está tão bem descrita no gênero *Rhodnius* como a nitroforina (124). Apesar de não ser observado neste trabalho, também já foi descrito em *C.*

lectularius a presença de nitroforina, porém com uma diversidade de tipos mais limitada se comparado a *Rhodnius* (125,126). Já os outros insetos hematófagos avaliados não apresentaram valores nos grupos de hematofagia aqui destacados, entretanto eles possuem outras proteínas que desempenham funções semelhantes como a apirase (127–130).

A família gênica de alérgeno de ácaro foi encontrado em todos os insetos avaliados, porém *R. prolixus* apresentou uma maior quantidade de proteínas, especialmente nos grupos OG0002624 e OG0006858. Essa família foi caracterizada primeiramente com a proteína Derp 7 (*Dermatophagoides pteronyssinus)* e em seguida o homólogo Derf 7 (*Dermatophagoides farinae*) foi descoberto. Ambas as proteínas são conhecidas por desencadear quadros de alergia, dermatite e bronquite em humanos (131). Além disso, alérgenos de ácaro possuem o domínio START que é comumente encontrado em proteínas de ligação ao hormônio juvenil e em proteínas *takeout*. Em *Amblyomma americanum* foram identificadas duas moléculas semelhantes a Derp 7 e Derf 7 que potencialmente agiriam como proteínas de ligação quimiossensorial (132–134). Considerando as proteínas de ligação ao hormônio juvenil e *takeout*, esse grupo ortólogo poderia estar envolvido com o desenvolvimento e reprodução ou então com regulação do ciclo circadiano, da alimentação e do exercício do inseto (135,136). Por outro lado, esse grupo poderia ter relação com a adaptabilidade ao ambiente, busca por alimento e parceiro ou mesmo fuga de predadores, através dos órgãos quimiossensoriais (137). Esta última possibilidade é particularmente interessante pois *R. prolixus* possui uma grande capacidade de se adaptar às habitações humanas e consequentemente necessita de um repertório gênico diverso para reconhecer os sinais ao seu redor e ser capaz de responder apropriadamente (19).

A última família analisada dentre as expansões foi a de canais de cloreto onde todos os insetos tiveram proteínas agrupadas, com destaque para *R. prolixus* nos grupos OG0002623 e OG0009397. Esses canais iônicos são responsáveis pelo balanço eletrolítico e osmótico e, estão presentes em praticamente todos os organismos. Nos insetos, tais canais se encontram em grande quantidade nos túbulos de Malpighi que realizam a excreção de sais e produtos nitrogenados como também reabsorção de moléculas (138). *R. prolixus* é conhecido pela sua capacidade de excretar rapidamente os fluidos obtidos na alimentação com sangue, seja para evitar predação ou um desequilíbrio homeostático devido a ingestão de

grande quantidade de fluido (cerca de 10 vezes o seu próprio peso). O inseto utiliza-se de canais iônicos para eliminar grande parte do líquido e concentrar a parte nutritiva (células sanguíneas) para a digestão. A excreção é feita principalmente na porção distal os túbulos de Malpighi, onde são secretados K^+, Cl^-, Na^+ e consequentemente água, já na sua porção proximal os íons K^+ e Cl^- são reabsorvidos (139–141). Como já mencionado, essa aptidão de *Rhodnius* em excretar fluidos e íons se destaca perante os outros insetos hematófagos, corroborando com a expansão aqui observada.

Já a respeito das contrações de famílias gênicas, para *Rhodnius prolixus* nenhuma família alcançou uma diferença tão perceptível no número de proteínas como ocorreu nas expansões. Mesmo assim, no grupo de domínio Tudor, o triatomíneo alcançou um número inferior ao dos outros insetos que obtiveram valores relativamente próximos. O domínio Tudor foi inicialmente detectado em *Drosophila melanogaster* em um trabalho que pesquisava fatores que regulam o desenvolvimento embrionário e fertilidade, e subsequentemente foi encontrado conservado evolutivamente em diversas espécies (142–145). As proteínas Tudor estão envolvidas em processos como metilação do DNA, metabolismo de RNA e reparo e detecção de dano ao DNA. Essas proteínas também tem a capacidade de inibir a replicação de elementos transponíveis em células da linhagem germinativa através de piRNA mantendo a estabilidade genômica. Além disso, em muitos organismos o nocaute de Tudor levou a problemas na oogênese e espermatogênese levando a diminuição na fertilidade (146–148). Nas montagens iniciais do genoma de *R. prolixus* foram identificadas moléculas com o domínio Tudor, como também outras proteínas (Vas, Maelstrom) que compõem a via de piRNA que é responsável por desempenhar as funções acima citadas. Em estudos posteriores de transcriptoma de ovário foram identificados alguns genes ortólogos aos de *D. melanogaster* da via de piRNA incluindo aqueles com o domínio Tudor. Na predição 1.2 de *Rhodnius* foram identificados 13 proteínas com o domínio Tudor em um trabalho que comparava o proteoma não redundante de 18 insetos (149–151). Apesar de apenas 4 proteínas Tudor terem sidos anotadas na montagem Hi-C, a literatura não sugere que *R. prolixus* tenha se diferenciado dos outros insetos ao ponto de fazer uma contração nesses genes. Portanto, ainda é necessário um estudo mais aprofundado a respeito desse domínio uma vez que as proteínas que o contém são excelentes alvos para o controle vetorial.

A contração da família de proteínas com domínio *paired* foi semelhante ao caso anterior, com apenas 4 proteínas sendo encontradas em *R. prolixus*. O domínio *Paired* foi descrito inicialmente em *D. melanogaster* com o nome de *paired box* dentro da proteína *paired* (prd). Posteriormente outras proteínas com o mesmo domínio foram descobertas como *gooseberry* proximal e distal (ambas em *Drosophila*) e também a proteína Pax, sendo que esta última também está presente em mamíferos. As proteínas com o domínio *paired* agem como reguladoras da transcrição do DNA sendo essenciais para o desenvolvimento embrionário e organogênese (152,153). A proteína Pax ainda não está bem elucidada em *R. prolixus*, mesmo que em predições anteriores (predição 1.2) 8 proteínas tenham sido detectadas (151). Um dos possíveis motivos da diminuição aqui relatada deve-se a montagem de genoma utilizada, entretanto, quando comparado aos outros insetos, essa diferença pode estar associada com a evolução da família Pax. Em organismos mais simples observou-se a presença de genes Pax, em esponjas do mesmo gênero, que não possuíam relações homólogas. Estes eventos de duplicação/deleção assimétricos dificultou o desenho do caminho evolutivo dessa família (154). Além disso, já foi sugerido que o domínio *paired* possui uma relação de descendência da transponsase de Tc1-Mariner, que é muito comum em *Rhodnius*, após um processo de domesticação desse elemento (155). Curiosamente, na montagem Hi-C foram detectadas menos repetições dessa família em relação a literatura, como também menos proteínas com o domínio *paired* em comparação a predição 1.2. Portanto, novos estudos são necessários para caracterizar as proteínas Pax em *R. prolixus* como também se a alta prevalência de Tc1-Mariner neste inseto pode ter alguma influência sobre essas proteínas.

5.3 Conciliação das predições

A era do sequenciamento aumentou consideravelmente o volume de dados gerados pela literatura. Esse crescimento permitiu a expansão dos nossos conhecimentos sobre genes e proteínas, como também das relações de homologia entre as sequências de DNA e aminoácidos (156). A anotação funcional dessas sequências é de suma importância para caracterizar as funções biológicas dos genes. Entretanto, no início essa anotação não era padronizada, o que ocasionava erros, como uma associação de nome errônea ou até mesmo uma predição de

função mais específica do que o suporte experimental oferecia na época (157,158). Atualmente, já existem bancos de dados curados e procedimentos padrões visando minimizar tais problemas (159,160).

A literatura não apresenta claramente métodos para conciliar predições ou mesmo compará-las a fim de gerar a mais completa possível. O Ensembl utiliza uma *pipeline* para realizar a anotação de genomas, onde sequências conhecidas de cDNA e de proteínas da mesma espécie são usadas para gerar modelos gênicos para as etapas posteriores (161). Contudo, somente são utilizados os dados provenientes de bancos como SWISS-PROT (162), EMBL (163) e RefSeq (164), então provavelmente genes preditos anteriormente, sem comprovação experimental, não devem ser incluídos. Por outro lado, também existem ferramentas como o CONTRAST (165) que visam complementar uma predição gênica se baseando em um genoma bem anotado. Apesar deste *software* tentar complementar a predição, ele faz uma predição *de novo*, sendo treinado com o genoma anotado, ao invés de se aproveitar de outras sequências já preditas. O GeneValidator (166) por sua vez identifica genes preditos errôneos através da comparação com sequências de grandes bancos de dados. Entretanto, esse *software* não ajuda no processo de conciliação em si, apenas executa um diagnóstico para posterior curadoria manual.

Neste trabalho, a identidade de sequência foi utilizada para auxiliar na conciliação das predições existentes de *R. prolixus*. Através do Cd-hit, observou-se que muitos transcritos das primeiras predições (1.0 a 1.3) não estavam mais presentes nas posteriores. E quando essa análise foi expandida para a predição P13, encontrou-se que muitos dos transcritos preditos na montagem Hi-C eram exclusivos da mesma. Isso pode ser resultado das técnicas utilizadas para realizar as montagens de genoma, o que o ocasionou a perda de determinadas regiões. Esses acontecimentos só deixaram mais evidente a necessidade de uma conciliação, com identificação dos genes antigos ausentes na predição P13. Dessa forma, é possível obter uma predição mais completa com mais genes disponíveis para a comunidade. Uma vez que novos estudos podem ser capazes de melhorar a sua anotação, como já foi feito por Coelho *et al* 2021 (99).

Para resolver o problema descrito acima, um *script* com o objetivo de conciliar todas as predições de *R. prolixus* foi desenvolvido. Através dele, tanto genes obtidos de transcriptoma como genes preditos antigos foram incorporados a predição P13. Quanto aos dados de transcriptoma, o tratamento o qual os genes preditos

quiméricos e fragmentados foram submetidos se mostrou eficaz na maioria dos casos. Como exemplificado para os genes RPHIC01198 (quimérico), RPHIC02790 e RPHIC02791 (fragmentados), o processamento do *script* não impactou na perda de domínios funcionais como também aumentou a identidade e cobertura com proteínas de organismos próximos a *Rhodnius* como *C. lectularius*. Entretanto, o modo de predição x transcriptoma ainda precisa ser aperfeiçoado pois apesar de não haver perda direta na completude (92,7%), BUSCOs considerados completos foram perdidos na transição de P13 para P14. Já o modo predição x predição gerou a predição mais completa desse inseto atualmente (93,2%), através da adição de 2.315 genes antigos sendo que 15 deles contribuíram diretamente para aumentar a completude no BUSCO. Ainda foi possível observar que grande parte dos transcritos contidos em P15 possuem códon de iniciação e terminação além de não terem códon de término prematuro. Esse fato evidencia tanto a qualidade da predição do AUGUSTUS como também a eficácia dos filtros aplicados tanto na seleção dos genes antigos de *Rhodnius* como também no *script* da conciliação. Obviamente, ainda é necessária uma forma de otimizar a adição de genes que possam contribuir para a completude em detrimento de genes que pouco agregam ao conjunto. Entretanto, esse *script* é um primeiro passo para ajudar na integração de dados gênicos sejam eles provenientes de testes *in silico* ou *in vivo*.

5.4 Navegador de genomas

Os navegadores de genoma são uma conhecida plataforma para colaboração entre pesquisadores, seja para compartilhar dados ou mesmo trocar conhecimento (64). Com o passar do tempo eles foram evoluindo e permitindo a adição de dados cada vez maiores além de agregar novas funções. Alguns navegadores mais recentes permitem até análises filogenômicas com visualização da divergência genética cromossomo por cromossomo (167). Esses *softwares* se mostraram importantes na pandemia de 2020 quando o genoma do SARS-CoV-2 foi montado pela primeira vez e compartilhado entre a comunidade científica. Em seguida, novas montagens foram surgindo juntamente com as variantes do vírus permitindo que cientistas ao redor do mundo comparassem esses genomas identificando as mutações envolvidas (168).

O VectorBase, atualmente integrante do consórcio VEuPathDB, é uma das principais plataformas que disponibilizam dados de insetos vetores sendo um deles *R. prolixus* (70). Entretanto, apenas a predição mais recente fica disponível para visualização no navegador enquanto as mais antigas ficam disponíveis apenas para *download*. Por sua vez isso não contribui para estudos de comparação entre as predições que poderiam resultar em atualizações de genes antigos como melhora da estrutura gênica ou anotação, por exemplo.

Com o objetivo de mitigar esses problemas, o Jbrowse (69) foi instalado e carregado com todas as predições do vetor até o momento. Apesar de existir outras opções disponíveis como o Gbrowser (169) e o UCSC browser (68), esse navegador foi o de escolha pois: 1) é de código aberto permitindo que o usuário crie seus próprios *plugins* como também possui uma comunidade ativa que faz a manutenção do código; 2) possui uma navegação fluída e robusta; 3) já havia sido utilizado por membros do nosso laboratório. Além disso, o Jbrowse foi o navegador escolhido por grandes bancos de dados como o VEuPathDB, InsectBase e Flybase, tendo este último migrado do Gbrowse para o Jbrowse devido a uma performance mais rápida, melhores opções de busca e de navegação (70–72).

Dessa maneira, com todas as predições disponíveis, toda a comunidade de entomologia pode desfrutar desses dados e também das predições P13, P14 e P15. Além disso, dados de RNAseq estão disponibilizados na montagem Hi-C para que regiões sem predição possam ser exploradas a fim de descobrir novos genes. Por fim, grande parte dos genes está anotada para facilitar a busca dentro do Jbrowse como também permitir que pesquisadores recuperem esses dados e os utilizem em experimentos para comprovar a sua existência e função biológica/molecular.

6 CONCLUSÕES

- O alinhamento dos dados de RNAseq mostraram que nem todas as regiões gênicas conseguiram ser preditas pelo AUGUSTUS, indicando que ainda existem novos genes a serem preditos;
- A qualidade da montagem Hi-C permitiu uma excelente predição gênica (P13) que alcançou uma completude superior a predição 1.3 de Mesquita *et al* 2015 (38) e a predição 3.4 do VectorBase;
- A conciliação se mostrou eficaz em combinar dados de transcriptoma e as predições antigas com P13, a fim de gerar um conjunto de genes mais completo (P15), baseado na montagem Hi-C, mas ao mesmo tempo contendo trechos de *scaffolds* de outras montagens;
- O Jbrowse foi capaz de disponibilizar todos os dados genômicos do inseto como também os alinhamentos de RNAseq de maneira eficiente;
- As diferenças sutis encontradas nas classes e famílias de elementos repetitivos sugerem que o mobiloma de *R. prolixus* está praticamente elucidado principalmente considerando que a montagem Hi-C aumenta a contiguidade dos *scaffolds*;
- As expansões de famílias gênicas em P13 se relacionaram principalmente com os hábitos hematofágicos do inseto e estão de acordo com o que já foi observado na literatura;
- As contrações de famílias gênicas não são tão evidentes como as expansões, porém ainda foi possível observar genes associados ao desenvolvimento e reprodução do inseto, que precisam ser melhor estudados.

7 REFERÊNCIAS BIBLIOGRÁFICAS

1. WHO. First WHO report on neglected tropical diseases: working to overcome the global impact of neglected tropical diseases. World Heal Organ. 2010;1–184.
2. Costa M, Tavares V, Aquino MV, Moreira D. Doença De Chagas: Uma Revisão Bibliográfica. Rev Eletrônica da Fac Ceres. 2013;2(1).
3. Pérez-Molina JA, Molina I. Chagas disease. Lancet. 2018;391(10115):82–94.
4. Bern C. Chagas' Disease. Longo DL, editor. N Engl J Med [Internet]. 2015 Jul 30;373(5):456–66. Available from: http://www.nejm.org/doi/10.1056/NEJMra1410150
5. Coura JR. The main sceneries of chagas disease transmission. The vectors, blood and oral transmissions - A comprehensive review. Mem Inst Oswaldo Cruz. 2015;110(3):277–82.
6. Moraes-Souza H, Ferreira-Silva MM. O controle da transmissão transfusional. Rev Soc Bras Med Trop [Internet]. 2011;44(suppl 2):64–7. Available from: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0037-86822011000800010&lng=pt&tlng=pt
7. SCHMUNIS GA. Trypanosoma cruzi , the etiologic agent of Chagas ' disease : status in the blood supply in endemic and nonendemic countries. Transfusion [Internet]. 1991;31(6):547–57. Available from: https://notifylibrary.org/sites/default/files/T.cruzi%2C Schmunis%2C 1991.pdf
8. Shikanai-Yasuda MA, Carvalho NB. Oral transmission of chagas disease. Clin Infect Dis. 2012;54(6):845–52.
9. Shikanai-Yasuda MA, Marcondes CB, Guedes LA, Siqueira GS, Barone AA, Dias JC, et al. Possible oral transmission of acute Chagas' disease in Brazil. Vol. 33, Revista do Instituto de Medicina Tropical de São Paulo. 1991. p. 351–7.
10. Dias JP, Bastos C, Araújo E, Mascarenhas AV, Netto EM, Grassi F, et al. Acute Chagas disease outbreak associated with oral transmission. Rev Soc Bras Med Trop. 2008;41(3):296–300.
11. Bastos CJC, Aras R, Mota G, Reis F, Dias JP, de Jesus RS, et al. Clinical outcomes of thirteen patients with acute chagas disease acquired through oral

transmission from two urban outbreaks in Northeastern Brazil. PLoS Negl Trop Dis. 2010;4(6):16–7.

12. Barroso Ferreira RT, Branquinho MR, Cardarelli-Leite P. Transmissão oral da doença de Chagas pelo consumo de açaí: um desafio para a Vigilância Sanitária. Vigilância Sanitária em Debate. 2014;2(4):4–11.
13. Pinto AY das N, Valente SA da S, Valente V da C. Emerging acute Chagas disease in Amazonian Brazil: case reports with serious cardiac involvement. Braz J Infect Dis. 2004;8(6):454–60.
14. Schmunis GA. Epidemiology of Chagas disease in non-endemic countries: The role of international migration. Mem Inst Oswaldo Cruz. 2007;102(SUPPL. 1):75–85.
15. Jannin J, Villa L. An overview of Chagas disease treatment. Mem Inst Oswaldo Cruz. 2007;102(SUPPL. 1):95–7.
16. Gorla D, Noireau F. Geographic distribution of Triatominae vectors in America [Internet]. Second Edi. American Trypanosomiasis Chagas Disease: One Hundred Years of Research: Second Edition. Elsevier Inc.; 2017. 197–221 p. Available from: http://dx.doi.org/10.1016/B978-0-12-801029-7.00009-5
17. Gourbière S, Dorn P, Tripet F, Dumonteil E. Genetics and evolution of triatomines: From phylogeny to vector control. Heredity (Edinb) [Internet]. 2012;108(3):190–202. Available from: www.nature.com/hdy
18. Ministério da Saúde. Guia de Vigilânica Epidemiológica [Internet]. Guia de vigilância epidemiológica. 2005. 17–34 p. Available from: www.saude.gov.br/svs%0Ahttp://bvsms.saude.gov.br/bvs/publicacoes/guia_vigilancia_epidemiologica_7ed.pdf
19. Costa J, Lorenzo M. Biology, diversity and strategies for the monitoring and control of triatomines - Chagas disease vectors. Mem Inst Oswaldo Cruz. 2009;104(SUPPL. 1):46–51.
20. Hashimoto K, Schofield CJ. Elimination of Rhodnius prolixus in Central America. Parasit Vectors [Internet]. 2012;5(1):45. Available from: http://www.parasitesandvectors.com/content/5/1/45
21. Medone P, Ceccarelli S, Parham PE, Figuera A, Rabinovich JE. The impact of climate change on the geographical distribution of two vectors of chagas disease: Implications for the force of infection. Philos Trans R Soc B Biol Sci. 2015;370(1665):1–12.

22. Justi SA, Galvão C. The Evolutionary Origin of Diversity in Chagas Disease Vectors. Trends Parasitol. 2017;33(1):42–52.
23. Pavan MG. Especiação em triatomíneos uma abordagem filogenética, biogeográfica e comportamental dos vetores de Chagas Rhodnius prolixus e R. robustus s.l. (Hemiptera: Reduviidae). 2013;261. Available from: https://www.arca.fiocruz.br/handle/icict/13176
24. GRIMALDI D, Engel MS. Evolution of the Insects. Cambridge University Press; 2005.
25. Buxton PA. The biology of a blood-sucking bug, Rhodnius prolixus. Trans Entomol Soc London. 1930;78:227–36.
26. Nunes-da-Fonseca R, Berni M, Tobias-Santos V, Pane A, Araujo HM. Rhodnius prolixus: From classical physiology to modern developmental biology. Genesis. 2017;55(5):1–11.
27. Dorn PL, Noireau FC, Krafsur ES, Lanzaro GC, Cornel AJ. Genetics of major insect vectors. Genetics and Evolution of Infectious Diseases. 2011. 411–472 p.
28. Noriega FG. Autogeny in three species of Triatominae: Rhodnius prolixus, Triatoma rubrovaria, and Triatoma infestans (Hemiptera: Reduviidae). J Med Entomol. 1992;29(2):273–7.
29. Chiang RG, Chiang JA. Reproductive physiology in the blood feeding insect, Rhodnius prolixus, from copulation to the control of egg production. J Insect Physiol [Internet]. 2017;97:27–37. Available from: http://dx.doi.org/10.1016/j.jinsphys.2016.06.001
30. Guhl F, De Sanchez N, Jaramillo CA. Longitudinal studies of the immune response of Colombian patients infected with Trypanosoma cruzi and T. rangeli. Parasitology. 1988;96(3):449–60.
31. De Moraes MH, Guarneri AA, Girardi FP, Rodrigues JB, Eger I, Tyler KM, et al. Different serological cross-reactivity of Trypanosoma rangeli forms in Trypanosoma cruzi-infected patients sera. Parasites and Vectors. 2008;1(1):1–10.
32. Lazzari CR. Celebrating the sequencing of the Rhodnius prolixus genome: A tribute to the memory of Vincent B. Wigglesworth. J Insect Physiol [Internet]. 2017;97:1–2. Available from: http://dx.doi.org/10.1016/j.jinsphys.2017.02.005
33. Panzera F, Pérez R, Hornos S, Panzera Y, Delgado V, Nicolini P.

Chromosome Numbers in the Triatominae a Review. Mem Inst Oswaldo Cruz, Rio Janeiro. 1996;91(February):515–8.

34. PÉREZ R, RAMSEY J, O'CONNOR JE, SALAZAR-SCHETTINO PM, FERRANDIS I, BARGUES MD, et al. GENOME SIZE DETERMINATION IN CHAGAS DISEASE TRANSMITTING BUGS (HEMIPTERA-TRIATOMINAE) BY FLOW CYTOMETRY. Oliveira PL, editor. Am J Trop Med Hyg [Internet]. 2007 Mar 1;76(3):516–21. Available from: https://d1wqtxts1xzle7.cloudfront.net/46082661/516-with-cover-page-v2.pdf?Expires=1654546506&Signature=TbEXaYS9PkzVS~1novlLsqR-VrKATt5Dlo16olZsSlQvwSVn0i5qcZYMWq-XAJkoY3clt06xaha8321iD27RNLOfD380otyGTLy9W-l-166dU3cxeymT7A8~3zvO3MCaNjmFrXo0jcFzKm1bJpfZuHD0
35. Pita S, Panzera F, Sánchez A, Panzera Y, Palomeque T, Lorite P. Distribution and evolution of repeated sequences in genomes of triatominae (Hemiptera-Reduviidae) inferred from genomic in situ hybridization. PLoS One. 2014;9(12):1–17.
36. Huebner E, Gondim K, Urmenyi T, Lowenberger C, Bisch P, Steel C, et al. The Case for Sequencing the Genome of the Blood-Feeding Hemipteran Insect,. Genome. 2005;(June 2014):1–14.
37. Megy K, Hammond M, Lawson D, Bruggner R V., Birney E, Collins FH. Genomic resources for invertebrate vectors of human pathogens, and the role of VectorBase. Infect Genet Evol. 2009;9(3):308–13.
38. Mesquita RD, Vionette-Amaral RJ, Lowenberger C, Rivera-Pomar R, Monteiro FA, Minx P, et al. Genome of Rhodnius prolixus, an insect vector of Chagas disease, reveals unique adaptations to hematophagy and parasite infection. Proc Natl Acad Sci U S A. 2015;112(48):14936–41.
39. Schama R, Pedrini N, Juárez MP, Nelson DR, Torres AQ, Valle D, et al. Rhodnius prolixus supergene families of enzymes potentially associated with insecticide resistance. Insect Biochem Mol Biol. 2016;69:91–104.
40. van Berkum NL, Lieberman-Aiden E, Williams L, Imakaev M, Gnirke A, Mirny LA, et al. Hi-C: A method to study the three-dimensional architecture of genomes. J Vis Exp. 2010;(39):1–7.
41. Elliott TA, Gregory TR. What's in a genome? The C-value enigma and the evolution of eukaryotic genome content. Philos Trans R Soc B Biol Sci.

2015;370(1678).
42. Gibney ER, Nolan CM. Epigenetics and gene expression. Heredity (Edinb). 2010;105(1):4–13.
43. McGinnis S, Madden TL. BLAST: At the core of a powerful and diverse set of sequence analysis tools. Nucleic Acids Res. 2004;32(WEB SERVER ISS.):20–5.
44. Burset M, Guigó R. Evaluation of gene structure prediction programs. Genomics. 1996;34(3):353–67.
45. Mathé C, Sagot MF, Schiex T, Rouzé P. Current methods of gene prediction, their strengths and weaknesses. Nucleic Acids Res. 2002;30(19):4103–17.
46. Fickett JW, Tung C shung. Assessment of protein coding measures. Nucleic Acids Res. 1992;20(24):6441–50.
47. Hutchinson GB, Hayden MR. The prediction of exons through an analysis of spliceable open reading frames. Nucleic Acids Res. 1992;20(13):3453–62.
48. Milanesi L, Kolchanov NA, Rogozin IB, Ischenko I V., Kel AE, Orlov YL, et al. Genviewer: a Computing Tool for Protein-Coding Regions Prediction in Nucleotide Sequences. 1993;573–87.
49. Borodovsky M, McIninch J. GENMARK: Parallel gene recognition for both DNA strands. Comput Chem. 1993;17(2):123–33.
50. Salzberg SL, Deicher AL, Kasif S, White O. Microbial gene identification using interpolated Markov models. Nucleic Acids Res. 1998;26(2):544–8.
51. Lukashin A V., Borodovsky M. GeneMark.hmm: New solutions for gene finding. Nucleic Acids Res. 1998;26(4):1107–15.
52. Guigó R, Agarwal P, Abril JF, Burset M, Fickett JW. An assessment of gene prediction accuracy in large DNA sequences. Genome Res. 2000;10(10):1631–42.
53. Stanke M, Waack S. Gene prediction with a hidden Markov model and a new intron submodel. Bioinformatics. 2003;19(SUPPL. 2):215–25.
54. Guigó R, Knudsen S, Drake N, Smith T. Prediction of gene structure. J Mol Biol. 1992;226(1):141–57.
55. Korf I. Gene finding in novel genomes. BMC Bioinformatics [Internet]. 2004;5(1):1–9. Available from: https://rdcu.be/cGCHB
56. Costa EB, Paulan SC. Processo de sequenciamento e Montagem de genomas. An da Esc Reg Informática da Soc Bras Comput (SBC)-Regional

Mato Grosso. 2016;7:1–8.

57. Kremer FS, Pinto S. Capítulo 5. Anotação de genomas. UFPEL [Internet]. 2016; Available from: http://labbioinfo.ufpel.edu.br/aulas_2016/
58. Wang Z, Chen Y, Li Y. A brief review of computational gene prediction methods. Genomics, proteomics Bioinforma / Beijing Genomics Inst [Internet]. 2004;2(4):216–21. Available from: http://dx.doi.org/10.1016/S1672-0229(04)02028-5
59. Huang Y, Chen SY, Deng F. Well-characterized sequence features of eukaryote genomes and implications for ab initio gene prediction. Comput Struct Biotechnol J [Internet]. 2016;14:298–303. Available from: http://dx.doi.org/10.1016/j.csbj.2016.07.002
60. Scalzitti N, Jeannin-Girardon A, Collet P, Poch O, Thompson JD. A benchmark study of ab initio gene prediction methods in diverse eukaryotic organisms. BMC Genomics. 2020;21(1):1–20.
61. Yip KY, Cheng C, Gerstein M. Machine learning and genome annotation: A match meant to be? Genome Biol. 2013;14(5).
62. Salzberg SL. Next-generation genome annotation: We still struggle to get it right. Genome Biol. 2019;20(1):19–21.
63. Danchin A, Ouzounis C, Tokuyasu T, Zucker JD. No wisdom in the crowd: genome annotation in the era of big data – current status and future prospects. Microb Biotechnol. 2018;11(4):588–605.
64. Nielsen CB, Cantor M, Dubchak I, Gordon D, Wang T. Visualizing genomes: techniques and challenges. Nat Methods [Internet]. 2010 Mar 25;7(S3):S5–15. Available from: http://www.nature.com/articles/nmeth.1422
65. Wang J, Kong L, Gao G, Luo J. A brief introduction to web-based genome browsers. Brief Bioinform. 2013;14(2):131–43.
66. Hubbard T, Barker D, Birney E, Cameron G, Chen Y, Clark L, et al. The Ensembl genome database project. Nucleic Acids Res. 2002;30(1):38–41.
67. Karolchik D, Baertsch R, Diekhans M, Furey TS, Hinrichs A, Lu YT, et al. The UCSC Genome Browser Database. Nucleic Acids Res. 2003;31(1):51–4.
68. Navarro Gonzalez J, Zweig AS, Speir ML, Schmelter D, Rosenbloom KR, Raney BJ, et al. The UCSC genome browser database: 2021 update. Nucleic Acids Res. 2021;49(D1):D1046–57.
69. Skinner ME, Uzilov A V., Stein LD, Mungall CJ, Holmes IH. JBrowse: A next-

generation genome browser. Genome Res. 2009;19(9):1630–8.

70. Amos B, Aurrecoechea C, Barba M, Barreto A, Basenko EY, Bażant W, et al. VEuPathDB: the eukaryotic pathogen, vector and host bioinformatics resource center. Nucleic Acids Res. 2022;50(D1):D898–911.
71. Larkin A, Marygold SJ, Antonazzo G, Attrill H, dos Santos G, Garapati P V., et al. FlyBase: Updates to the Drosophila melanogaster knowledge base. Nucleic Acids Res. 2021;49(D1):D899–907.
72. Mei Y, Jing D, Tang S, Chen X, Chen H, Duanmu H, et al. InsectBase 2.0: a comprehensive gene resource for insects. Nucleic Acids Res. 2022;50(D1):D1040–5.
73. Ribeiro JMC, Genta FA, Sorgine MHF, Logullo R, Mesquita RD, Paiva-Silva GO, et al. An Insight into the Transcriptome of the Digestive Tract of the Bloodsucking Bug, Rhodnius prolixus. PLoS Negl Trop Dis. 2014;8(1):27.
74. Li W, Godzik A. Cd-hit: A fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics. 2006;22(13):1658–9.
75. Price AL, Jones NC, Pevzner PA. De novo identification of repeat families in large genomes. Bioinformatics. 2005;21(SUPPL. 1):351–8.
76. Storer J, Hubley R, Rosen J, Wheeler TJ, Smit AF. The Dfam community resource of transposable element families, sequence models, and genome annotations. Mob DNA. 2021;12(1):1–14.
77. Martin M. Cutadapt removes adapter sequences from high-throughput sequencing reads. EMBnet.journal [Internet]. 2011 May 2;17(1):10. Available from: http://journal.embnet.org/index.php/embnetjournal/article/view/200
78. Bolger AM, Lohse M, Usadel B. Trimmomatic: A flexible trimmer for Illumina sequence data. Bioinformatics. 2014;30(15):2114–20.
79. Langmead B, Salzberg SL. Fast gapped-read alignment with Bowtie 2. Nat Methods. 2012;9(4):357–9.
80. Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, et al. Twelve years of SAMtools and BCFtools. Gigascience. 2021;10(2):1–4.
81. Kim D, Langmead B, Salzberg1 SL. HISAT: a fast spliced aligner with low memory requirements Daehwan HHS Public Access. Nat Methods. 2015;12(4):357–60.
82. Simão FA, Waterhouse RM, Ioannidis P, Kriventseva E V., Zdobnov EM. BUSCO: Assessing genome assembly and annotation completeness with

single-copy orthologs. Bioinformatics. 2015;31(19):3210–2.

83. Mi H, Ebert D, Muruganujan A, Mills C, Albou LP, Mushayamaha T, et al. PANTHER version 16: A revised family classification, tree-based classification tool, enhancer regions and extensive API. Nucleic Acids Res. 2021;49(D1):D394–403.
84. Jones P, Binns D, Chang HY, Fraser M, Li W, McAnulla C, et al. InterProScan 5: Genome-scale protein function classification. Bioinformatics. 2014;30(9):1236–40.
85. Gough J, Chothia C. SUPERFAMILY: HMMs representing all proteins of known structure. SCOP sequence searches, alignments and genome assignments. Nucleic Acids Res. 2002;30(1):268–72.
86. Marchler-Bauer A, Zheng C, Chitsaz F, Derbyshire MK, Geer LY, Geer RC, et al. CDD: Conserved domains and protein three-dimensional structure. Nucleic Acids Res. 2013;41(D1):348–52.
87. Mistry J, Chuguransky S, Williams L, Qureshi M, Salazar GA, Sonnhammer ELL, et al. Pfam: The protein families database in 2021. Nucleic Acids Res. 2021;49(D1):D412–9.
88. Emms DM, Kelly S. OrthoFinder: Phylogenetic orthology inference for comparative genomics. bioRxiv. 2018;1–14.
89. Buchfink B, Xie C, Huson DH. Fast and sensitive protein alignment using DIAMOND. Nat Methods. 2014;12(1):59–60.
90. Suzek BE, Wang Y, Huang H, McGarvey PB, Wu CH. UniRef clusters: A comprehensive and scalable alternative for improving sequence similarity searches. Bioinformatics. 2015;31(6):926–32.
91. Rost B. Twilight zone of protein sequence alignments. Protein Eng. 1999;12(2):85–94.
92. Florea L, Hartzell G, Zhang Z, Rubin GM, Miller W. A computer program for aligning a cDNA sequence with a genomic DNA sequence. Genome Res. 1998;8(9):967–74.
93. Slater GSC, Birney E. Automated generation of heuristics for biological sequence comparison. BMC Bioinformatics. 2005;6:1–11.
94. O'Leary NA, Wright MW, Brister JR, Ciufo S, Haddad D, McVeigh R, et al. Reference sequence (RefSeq) database at NCBI: Current status, taxonomic expansion, and functional annotation. Nucleic Acids Res. 2016;44(D1):D733–

45.

95. Chénais B, Caruso A, Hiard S, Casse N. The impact of transposable elements on eukaryotic genomes: From genome size increase to genetic adaptation to stressful environments. Gene [Internet]. 2012;509(1):7–15. Available from: http://dx.doi.org/10.1016/j.gene.2012.07.042
96. Miller WJ, Capy P. Mobile genetic elements as Natural Tools for Genome Evolution. Mob Genet Elements. 2004;260:1–20.
97. Fernández-Medina RD, Granzotto A, Ribeiro JM, Carareto CMA. Transposition burst of mariner-like elements in the sequenced genome of Rhodnius prolixus. Insect Biochem Mol Biol. 2016;69:14–24.
98. Castro MRJ, Goubert C, Carareto CMA, Monteiro FA, Vieira C. Homology-free detection of transposable elements unveils their dynamics in three ecologically distinct rhodnius species. Genes (Basel). 2020;11(2):1–14.
99. Coelho VL, de Brito TF, de Abreu Brito IA, Cardoso MA, Berni MA, Araujo HMM, et al. Analysis of ovarian transcriptomes reveals thousands of novel genes in the insect vector Rhodnius prolixus. Sci Rep [Internet]. 2021;11(1):1–17. Available from: https://doi.org/10.1038/s41598-021-81387-1
100. Castro MRJ. Dinâmica evolutiva de elementos de transposição nas espécies irmãs Rhodnius robustus e R . prolixus. 2019;29–33.
101. Belyayev A. Bursts of transposable elements as an evolutionary driving force. J Evol Biol. 2014;27(12):2573–84.
102. Belyayev A, Kalendar R, Brodsky L, Nevo E, Schulman AH, Raskina O. Transposable elements in a marginal plant population: Temporal fluctuations provide new insights into genome evolution of wild diploid wheat. Mob DNA. 2010;1(1):1–16.
103. de Boer JG, Yazawa R, Davidson WS, Koop BF. Bursts and horizontal evolution of DNA transposons in the speciation of pseudotetraploid salmonids. BMC Genomics. 2007;8:1–10.
104. Kraitshtein Z, Yaakov B, Khasdan V, Kashkush K. Genetic and epigenetic dynamics of a retrotransposon after allopolyploidization of wheat. Genetics. 2010;186(3):801–12.
105. Kenan-Eichler M, Leshkowitz D, Tal L, Noor E, Melamed-Bessudo C, Feldman M, et al. Wheat hybridization and polyploidization results in deregulation of small RNAs. Genetics. 2011;188(2):263–72.

106. Ungerer MC, Strakosh SC, Stimpson KM. Proliferation of Ty3/gypsy-like retrotransposons in hybrid sunflower taxa inferred from phylogenetic data. BMC Biol. 2009;7:1–13.
107. Naito K, Cho E, Yang G, Campbell MA, Yano K, Okumoto Y, et al. Dramatic amplification of a rice transposable element during recent domestication. Proc Natl Acad Sci U S A. 2006;103(47):17620–5.
108. Treangen TJ, Salzberg SL. Repetitive DNA and next-generation sequencing: computational challenges and solutions. Nat Rev Genet [Internet]. 2012 Jan 29;13(1):36–46. Available from: http://www.nature.com/articles/nrg3117
109. Doerks T, Bairoch A, Bork P. Protein annotation: Detective work for function prediction. Trends Genet. 1998;14(6):248–50.
110. Ouali R, Vieira LR, Salmon D, Bousbata S. Early post-prandial regulation of protein expression in the midgut of chagas disease vector rhodnius prolixus highlights new potential targets for vector control strategy. Microorganisms. 2021;9(4).
111. Burge C, Karlin S. Prediction of complete gene structures in human genomic DNA11Edited by F. E. Cohen. J Mol Biol [Internet]. 1997;268(1):78–94. Available from: http://www.sciencedirect.com/science/article/pii/S0022283697909517
112. Salzberg SL, Pertea M, Delcher AL, Gardner MJ, Tettelin H. Interpolated Markov models for eukaryotic gene finding. Genomics. 1999;59(1):24–31.
113. Bondar EI, Feranchuk SI, Birukhov V V, Kuzmin DA, Sharov V V. Annotation of Siberian larch genome draft assembly. 2021;18699.
114. Venkatraman M, Fleischer RC, Tsuchiya MTN. Comparative Analysis of Annotation Pipelines Using the First Japanese White-Eye (Zosterops japonicus) Genome. Genome Biol Evol. 2021;13(5):1–5.
115. Mighell AJ, Smith NR, Robinson PA, Markham AF. Vertebrate pseudogenes. FEBS Lett. 2000;468(2–3):109–14.
116. Pink RC, Wicks K, Caley DP, Punch EK, Jacobs L, Carter DRF. Pseudogenes: Pseudo-functional or key regulators in health and diseasě. Rna. 2011;17(5):792–8.
117. Lawrie RD, Mitchell RD, Deguenon JM, Ponnusamy L, Reisig D, Pozo-Valdivia A Del, et al. Characterization of long non-coding rnas in the bollworm, helicoverpa zea, and their possible role in cry1ac-resistance. Insects.

2022;13(1).

118. Ashurst JL, Collins JE. Gene Annotation: Prediction and Testing. Annu Rev Genomics Hum Genet. 2003;4:69–88.
119. Mudge JM, Harrow J. The state of play in higher eukaryote gene annotation. Nat Rev Genet [Internet]. 2016;17(12):758–72. Available from: http://dx.doi.org/10.1038/nrg.2016.119
120. Andersen JF, Champagne DE, Weichsel A, Ribeiro JMC, Balfour CA, Dress V, et al. Nitric oxide binding and crystallization of recombinant nitrophorin I, a nitric oxide transport protein from the blood-sucking bug Rhodnius prolixus. Biochemistry. 1997;36(15):4423–8.
121. Montfort WR, Weichsel A, Andersen JF. Nitrophorins and related antihemostatic lipocalins from Rhodnius prolixus and other blood-sucking arthropods. Biochim Biophys Acta - Protein Struct Mol Enzymol. 2000;1482(1–2):110–8.
122. Andersen JF, Gudderra NP, Francischetti IMB, Ribeiro JMC. The role of salivary lipocalins in blood feeding by Rhodnius prolixus. Arch Insect Biochem Physiol. 2005;58(2):97–105.
123. Hernández-Vargas MJ, Santibáñez-López CE, Corzo G. An insight into the triabin protein family of American hematophagous reduviids: Functional, structural and phylogenetic analysis. Toxins (Basel). 2016;8(2).
124. Santiago PB, Charneau S, Mandacaru SC, Bentes KL da S, Bastos IMD, de Sousa MV, et al. Proteomic Mapping of Multifunctional Complexes Within Triatomine Saliva. Front Cell Infect Microbiol. 2020;10(September):1–14.
125. Walker FA. Nitric oxide interaction with insect nitrophorins and thoughts on the electron configuration of the {FeNO}6 complex. J Inorg Biochem. 2005;99(1):216–36.
126. Knipp M, He C. Nitrophorins: Nitrite disproportionation reaction and other novel functionalities of insect heme-based nitric oxide transport proteins. IUBMB Life. 2011;63(5):304–12.
127. Mumcuoglu KY, Galun R, Kaminchik Y, Panet A, Levanon A. Antihemostatic activity in salivary glands of the human body louse, Pediculus humanus humanus (Anoplura: Pediculidae). J Insect Physiol. 1996;42(11–12):1083–7.
128. Champagne DE. Antihemostatic molecules from saliva of blood-feeding arthropods. Pathophysiol Haemost Thromb. 2006;34(4–5):221–7.

129. Arcà B, Ribeiro JM. Saliva of hematophagous insects: a multifaceted toolkit. Curr Opin Insect Sci. 2018;29:102–9.
130. Cohen E, Moussian B. Extracellular Composite Matrices in Arthropods [Internet]. Cohen E, Moussian B, editors. Extracellular Composite Matrices in Arthropods. Cham: Springer International Publishing; 2016. 1–712 p. Available from: http://link.springer.com/10.1007/978-3-319-40740-1
131. Tan KW, Jobichen C, Ong TC, Gao YF, Tiong YS, Wong KN, et al. Crystal Structure of Der f 7, a Dust Mite Allergen from Dermatophagoides farinae. PLoS One. 2012;7(9):1–8.
132. Saito K, Su ZH, Emi A, Mita K, Takeda M, Fujiwara Y. Cloning and expression analysis of takeout/JHBP family genes of silkworm, Bombyx mori. Insect Mol Biol. 2006;15(3):245–51.
133. Fujikawa K, Seno K, Ozaki M. A novel takeout-like protein expressed in the taste and olfactory organs of the blowfly, Phormia regina. FEBS J. 2006;273(18):4311–21.
134. Renthal R, Manghnani L, Bernal S, Qu Y, Griffith WP, Lohmeyer K, et al. The chemosensory appendage proteome of Amblyomma americanum (Acari: Ixodidae) reveals putative odorant-binding and other chemoreception-related proteins. Insect Sci. 2017;24(5):730–42.
135. Li K, Jia QQ, Li S. Juvenile hormone signaling – a mini review. Insect Sci. 2019;26(4):600–6.
136. Zhang H, Yu H, Zhao X, Liu X, Feng X, Huang X. Investigations of Takeout proteins' ligand binding and release mechanism using molecular dynamics simulation. J Biomol Struct Dyn. 2017;35(7):1464–73.
137. Sánchez-Gracia A, Vieira FG, Rozas J. Molecular evolution of the major chemosensory gene families in insects. Heredity (Edinb). 2009;103(3):208–16.
138. Beyenbach KW. Transport mechanisms of diuresis in Malpighian tubules of insects. J Exp Biol. 2003;206(21):3845–56.
139. O'Donnell MJ, Maddrell SH. Secretion by the Malpighian tubules of Rhodnius prolixus stal: electrical events. J Exp Biol [Internet]. 1984 May 1;110(1):275–90. Available from: https://journals.biologists.com/jeb/article/110/1/275/4193/Secretion-by-the-Malpighian-tubules-of-Rhodnius
140. Ianowski JP, O'Donnell MJ. Electrochemical gradients for Na+, K+, Cl- and H+

across the apical membrane in Malpighian (renal) tubule cells of Rhodnius prolixus. J Exp Biol. 2006;209(10):1964–75.

141. Martini S V., Nascimento SB, Morales MM. Rhodnius prolixus Malpighian tubules and control of diuresis by neurohormones. An Acad Bras Cienc. 2007;79(1):87–95.
142. Boswell R. tudor, a gene required for assembly of the germ plasm in Drosophila melanogaster. Cell [Internet]. 1985 Nov;43(1):97–104. Available from: https://linkinghub.elsevier.com/retrieve/pii/0092867485900157
143. Jin J, Xie X, Chen C, Park JG, Stark C, James DA, et al. Eukaryotic protein domains as functional units of cellular evolution. Sci Signal. 2009;2(98):1–18.
144. Li J, Xue X, Ruan J, Wu M, Zhu Z, Zang J. Cloning, purification, crystallization and preliminary crystallographic analysis of the tandem tudor domain of Sgf29 from Saccharomyces cerevisiae. Acta Crystallogr Sect F Struct Biol Cryst Commun. 2010;66(8):902–4.
145. Brasil JN, Cabral LM, Eloy NB, Primo LMF, Barroso-Neto IL, Grangeiro LPP, et al. AIP1 is a novel Agenet/Tudor domain protein from Arabidopsis that interacts with regulators of DNA replication, transcription and chromatin remodeling. BMC Plant Biol [Internet]. 2015;15(1):1–21. Available from: http://dx.doi.org/10.1186/s12870-015-0641-z
146. Tanaka T, Hosokawa M, Vagin V V., Reuter M, Hayashi E, Mochizuki AL, et al. Tudor domain containing 7 (Tdrd7) is essential for dynamic ribonucleoprotein (RNP) remodeling of chromatoid bodies during spermatogenesis. Proc Natl Acad Sci U S A. 2011;108(26):10579–84.
147. Siomi MC, Mannen T, Siomi H. How does the royal family of tudor rule the PIWI-interacting RNA pathway? Genes Dev. 2010;24(7):636–46.
148. Kim M, Ki BS, Hong K, Park SP, Ko JJ, Choi Y. Tudor domain containing protein TDRD12 expresses at the acrosome of spermatids in mouse testis. Asian-Australasian J Anim Sci. 2016;29(7):944–51.
149. Lu H ling, Tanguy S, Rispe C, Gauthier JP, Walsh T, Gordon K, et al. Expansion of genes encoding piRNA-associated argonaute proteins in the pea aphid: Diversification of expression profiles in different plastic morphs. PLoS One. 2011;6(12).
150. Vidal NM, Grazziotin AL, Iyer LM, Aravind L, Venancio TM. Transcription factors, chromatin proteins and the diversification of Hemiptera. Insect Biochem

Mol Biol [Internet]. 2016 Feb;69(3):1–13. Available from: https://linkinghub.elsevier.com/retrieve/pii/S0965174815300199

151. Brito T, Julio A, Berni M, de Castro Poncio L, Bernardes ES, Araujo H, et al. Transcriptomic and functional analyses of the piRNA pathway in the Chagas disease vector Rhodnius prolixus. PLoS Negl Trop Dis. 2018;12(10):1–20.
152. Bopp D, Burri M, Baumgartner S, Frigerio G, Noll M. Conservation of a large protein domain in the segmentation gene paired and in functionally related genes of Drosophila. Cell. 1986;47(6):1033–40.
153. Baumgartner S, Bopp D, Burri M, Noll M. Structure of two genes at the gooseberry locus related to the paired gene and their spatial expression during Drosophila embryogenesis. Genes Dev. 1987;1(10):1247–67.
154. Paixão-Côrtes VR, Salzano FM, Bortolini MC. Origins and evolvability of the PAX family. Semin Cell Dev Biol [Internet]. 2015;44:64–74. Available from: http://dx.doi.org/10.1016/j.semcdb.2015.08.014
155. Breitling R, Gerber JK. Origin of the paired domain. Dev Genes Evol. 2000;210(12):644–50.
156. Rubin GM, Yandell MD, Wortman JR, Gabor Miklos GL, Nelson CR, Hariharan IK, et al. Comparative genomics of the eukaryotes. Science (80-). 2000;287(5461):2204–15.
157. Schnoes AM, Brown SD, Dodevski I, Babbitt PC. Annotation error in public databases: Misannotation of molecular function in enzyme superfamilies. PLoS Comput Biol. 2009;5(12).
158. Dall'Olio GM, Bertranpetit J, Laayouni H. The annotation and the usage of scientific databases could be improved with public issue tracker software. Database (Oxford). 2010;2010:1–6.
159. Poux S, Magrane M, Arighi CN, Bridge A, O'Donovan C, Laiho K. Expert curation in UniProtKB: A case study on dealing with conflicting and erroneous data. Database. 2014;2014:1–9.
160. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, et al. Gene Ontology: tool for the unification of biology. Nat Genet [Internet]. 2000 May;25(1):25–9. Available from: http://www.nature.com/articles/ng0500_25
161. Curwen V, Eyras E, Andrews TD, Clarke L, Mongin E, Searle SMJ, et al. The Ensembl automatic gene annotation system. Genome Res. 2004;14(5):942–50.

162. Boeckmann B, Bairoch A, Apweiler R, Blatter MC, Estreicher A, Gasteiger E, et al. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. Nucleic Acids Res. 2003;31(1):365–70.
163. Stoesser G, Moseley MA, Sleep J, McGowran M, Garcia-Pastor M, Sterk P. The EMBL nucleotide sequence database. Nucleic Acids Res. 1998;26(1):8–15.
164. Pruitt KD, Katz KS, Sicotte H, Maglott DR. Introducing RefSeq and LocusLink: Curated human genome resources at the NCBI. Trends Genet. 2000;16(1):44–7.
165. Flicek P. Gene prediction: Compare and CONTRAST. Genome Biol. 2007;8(12):10–2.
166. Dragan MA, Moghul I, Priyam A, Bustos C, Wurm Y. GeneValidator: Identify problems with protein-coding gene predictions. Bioinformatics. 2016;32(10):1559–61.
167. Harris AJ, Foley NM, Williams TL, Murphy WJ. Tree House Explorer : A Novel Genome Browser for Phylogenomics. 2022;
168. Gültekin V, Allmer J. Novel perspectives for SARS-CoV-2 genome browsing. J Integr Bioinform. 2021;18(1):19–26.
169. Stein LD, Mungall C, Shu S, Caudy M, Mangone M, Day A, et al. The generic genome browser: A building block for a model organism system database. Genome Res. 2002;12(10):1599–610.

8 APÊNDICES E/OU ANEXOS

8.1 RepeatScout e RepeatMasker

#1 build frequency table

build_lmer_table -sequence Rhodnius_prolixus-3.0.3_HiC.fasta -freq rprohic.freq

#2 create fasta file containing all kinds of repeats. The output (output_repeats.fas) is a fasta file with headers (>R=1, >R=232 etc.). It contains also trivial simple repeats (CACACA...), tandem repeats

RepeatScout -sequence Rhodnius_prolixus-3.0.3_HiC.fasta -output rprohic.fasta -freq rprohic.freq

#3 filter out short (<50bp) sequences. Remove "anything that is over 50% low-complexity vis a vis TRF or NSEG.". Perl script.

filter-stage-1.prl rprohic.fasta > rprohic.fasta.filter_1

#4 run RepeatMasker on your genome of interest using filtered RepeatScout library

RepeatMasker Rhodnius_prolixus-3.0.3_HiC.fasta -lib rprohic.fasta.filter_1 -xsmall

#5 filtering putative repeats by copy number. By default only sequences occurring > 10 times in the genome are kept

cat rprohic.fasta.filter_1 | filter-stage-2.prl --cat=Rhodnius_prolixus-3.0.3_HiC.fasta.out > rprohic.fasta.filter_2

#6 run RepeatMasker on your genome of interest using filtered RepeatScout library

RepeatMasker Rhodnius_prolixus-3.0.3_HiC.fasta -lib rprohic.fasta.filter_2 -xsmall

8.2 Bowtie2

#1 Cria uma lista de index do genoma referência

bowtie2-build -f -a --seed 12345 Rhodnius_prolixus-3.0.3_HiC.fasta.masked rprohic

#2 Alinhamento end-to-end

bowtie2 --end-to-end --score-min L,-2,-0.25 -k 3 --threads 20 --mm --seed 12345 -x rprohic -U rp_rnaseq.fastq -S rp_rnaseq.sam

#3 Conversão para BAM

samtools view -@ 8 -Sb rp_total_s.sam > rp_total_s.bam

8.3 Hisat2

nohup hisat2 -p 14 -x rprohic -U rp_in.fastq,rp_out_1.fastq,rp_out_2.fastq,rp_out_1_1.fastq,rp_out_2_2.fastq | samtools view -@ 8 -Sb > rp_total_s.bam &

8.4 Augustus

#Predição básica:

augustus --species=rhodnius --gff3=on --introns=on --genemodel=complete --softmasking=1 Rhodnius_prolixus-3.0.3_HiC_masked

#Predição usando RNAseq:

#Filtrando os alinhamentos do Hisat2:

samtools sort -@ 20 -n -m 4G -o rp_s.s.bam rp_s.bam
samtools sort -@ 20 -n -m 4G -o rp_p.s.bam rp_p.bam

```
filterBam --uniq --in rp_s.s.bam --out rp_s.sf.bam
filterBam --uniq --in rp_p.s.bam --out rp_p.sf.bam
samtools sort -@ 20 -n -m 4G -o rp_s.ssf.bam rp_s.sf.bam
samtools sort -@ 20 -n -m 4G -o rp_p.ssf.bam rp_p.sf.bam

#Salvando os headers para usar em etapas posteriores:

samtools view -H rp_s.ssf.bam > headers_s.txt
samtools view -H rp_p.ssf.bam > headers_p.txt

#Gerando hints de introns

bam2hints --intronsonly --in=rp_s.ssf.bam --out=rp_hints_s.gff
bam2hints --intronsonly --in=rp_p.ssf.bam --out=rp_hints_p.gff
cat rp_hints_s.gff rp_hints_p.gff > rp_hints1.gff

#Predição com hints 1

augustus --species=rhodnius --extrinsicCfgFile=/home/src/augustus-3.3.3/config/species/rhodnius/extrinsic.cfg --alternatives-from-evidence=true --hintsfile=rp_hints1.gff --allow_hinted_splicesites=atac --gff3=on --introns=on --genemodel=complete --softmasking=1 Rhodnius_prolixus-3.0.3_HiC_masked > rphic2.gff3

#Extraindo os introns da predição

cat rphic2.gff3 | grep -P "\tintron\t" > rp_introns.gff
cat rp_hints1.gff rp_introns.gff | perl -ne '@array = split(/\t/, $_);print "$array[0]:$array[3]-$array[4]\n";'| sort -u > introns.lst

# Montando as junções exon-exon

intron2exex.pl --introns=rp_introns.lst --seq=Rhodnius_prolixus-3.0.3_HiC_masked --exex=rp_exex.fasta --map=map.psl
```

#Indexando as junções

hisat2-build -f Rhodnius_prolixus-3.0.3_HiC_masked rp_exex

#Alinhamento single:

hisat2 -p 14 -x rp_exex -U rp_in.fastq, rp_out_1.fastq, rp_out_2.fastq, rp_out_1_1.fastq, rp_out_2_2.fastq -S rp_s2.sam

#Alinhamento paired:

hisat2 -p 14 -x rp_exex -1 rp_out_1.fastq,rp_out_1_1.fastq -2 rp_out_2.fastq, rp_out_2_2.fastq -S rp_p2.sam

#Descartando os alinhamentos ruins:

samtools view -@ 20 -S -F 4 rp_s2.sam > rp_s2.F.sam
samtools view -@ 20 -S -F 4 rpo_rna_total.sam > rp_p2.F.sam

#Mapeando os alinhamentos locais ex-ex globalmente no genoma:

samMap.pl rp_s2.F.sam map.psl > rp_s2.global.sam
samMap.pl rp_p2.F.sam map.psl > rp_p2.global.sam

#Criando um arquivo sam com headers:

cat header_s.txt rp_s2.global.sam > rp_s2.global.h.sam
cat header_p.txt rp_p2.global.sam > rp_p2.global.h.sam
samtools view -Sb -@ 20 rp_s2.global.h.sam > rp_s2.global.h.bam
samtools view -Sb -@ 20 rp_p2.global.h.sam > rp_p2.global.h.bam

#Descartando alinhamentos contendo introns do bam original:

bamtools filter -in rp_s.ssf.bam -out rp_s.ssf.noN.bam -script operation_N_filter.txt
bamtools filter -in rp_p.ssf.bam -out rp_p.ssf.noN.bam -script operation_N_filter.txt

#Juntando os dois arquivos:

bamtools merge -in rp_s2.global.h.bam -in rp_s.ssf.noN.bam -out rp_total_s2.bam
bamtools merge -in rp_p2.global.h.bam -in rp_p.ssf.noN.bam -out rp_total_p2.bam

#Filtrando os alinhamentos do Hisat2:

samtools sort -@ 20 -n -m 4G -o rp_total_s2.s.bam rp_total_s2.bam
samtools sort -@ 20 -n -m 4G -o rp_total_p2.s.bam rp_total_p2.bam
filterBam --uniq --in rp_total_s2.s.bam --out rp_total_s2.sf.bam
filterBam --uniq --in rp_total_p2.s.bam --out rp_total_p2.sf.bam
samtools sort -@ 20 -n -m 4G -o rp_total_s2.ssf.bam rp_total_s2.sf.bam
samtools sort -@ 20 -n -m 4G -o rp_total_p2.ssf.bam rp_total_p2.sf.bam

#Gerando hints de introns

bam2hints --intronsonly --in=rp_total_s2.ssf.bam --out=rp_hints_s2.gff
bam2hints --intronsonly --in=rp_total_p2.ssf.bam --out=rp_hints_p2.gff
cat rp_hints_s2.gff rp_hints_p2.gff > rp_hints2.gff

#Predição com hints 2

augustus --species=rhodnius --extrinsicCfgFile=/home/src/augustus-3.3.3/config/species/rhodnius/extrinsic.cfg --alternatives-from-evidence=true --hintsfile=rp_hints2.gff --allow_hinted_splicesites=atac --gff3=on --introns=on --genemodel=complete --softmasking=1 Rhodnius_prolixus-3.0.3_HiC_masked

8.5 Busco

run_busco -i rphic2.aa -o rphic2.busco -l hemiptera_odb10 -m proteins -c 8

8.6 Cd-Hit

cdhit-est -i rpropredictions.fasta -o rpropredictions_99.fasta -c 0.99 -T 12

8.7 Jbrowse

Comandos para adicionar arquivos ao Jbrowse, a pasta onde estão os arquivos deve estar nomeada como data para evitar a criação de novas pastas.

Para adicionar sequências de referências:
../bin/prepare-refseqs.pl --fasta arquivo.fa

Para adicionar GFFs:
../bin/flatfile-to-json.pl --gff arquivo.gff3 --trackType CanvasFeatures --trackLabel nome da track --type gene --nameAttributes "name,id,alias,gene_id"

Para permitir a busca no navegador:
../bin/generate-names.pl

Para ativar o plugin FeatureSequence:

```
  "menuTemplate" : [
   {
   "content" : "function(track,feature){return track.browser.plugins.FeatureSequence.callFxn(track, feature); }",
   "label" : "View Feature Sequence",
   "action" : "contentDialog",
   "iconClass" : "dijitIconBookmark"
    }
    ],
```

Para adicionar BAM no trackList.json:

```
{
"label" : "nome da track",
"urlTemplate" : "arquivo BAM",
```

```
"storeClass" : "JBrowse/Store/SeqFeature/BAM",
"type" : "Alignments2"
}
```

Além de criar um index do BAM com o comando:

```
samtools index rproT103-D1.sorted.bam
```

8.8 *Script* DIAMOND

```
from argparse import ArgumentParser
import subprocess
import pandas as pd
from math import exp
from Bio import SeqIO


def making_db(cores, fasta_subject):
    """
    This function makes a database for DIAMOND alignment.
    :param cores: Number of cores to be used.
    :param fasta_subject: Fasta to be used to make the database.
    """
    db_line = f'diamond makedb --in {fasta_subject} --db {fasta_subject.split(".")[0]} -p
{cores}'
    try:
        subprocess.call(db_line, shell=True)
    except Exception as err:
        print(f'Check if DIAMOND is installed and on the PATH.\n\n{err}')
    return


def running_diamond(cores, database, fasta_query, mode, diamond_out):
    """
```

```
    This function runs the DIAMOND.
    :param cores: Number of cores to be used.
    :param database: Database to be aligned against fasta query.
    :param fasta_query: Fasta query to be aligned against the database.
    :param mode: Blast mode to be used (blastp or blastx).
    :param diamond_out: Diamond output.
    """
    if mode == 'blastx':
        diamond_line = 'diamond {} -p {} --sensitive -d {} -q {} -o {} -f 6 qseqid sseqid pident length mismatch gapopen qstart qend qlen sstart send slen evalue bitscore'.format(
            mode, cores, database, fasta_query, diamond_out)
        try:
            subprocess.call(diamond_line, shell=True)
        except Exception as err:
            print(
                f'Check if DIAMOND is installed and on the PATH.\n\n{err}')
    elif mode == 'blastp':
        diamond_line = 'diamond {} -p {} --sensitive -d {} -q {} -o {} -f 6 qseqid sseqid pident length mismatch gapopen qstart qend qlen sstart send slen evalue bitscore'.format(
            mode, cores, database, fasta_query, diamond_out)
        try:
            subprocess.call(diamond_line, shell=True)
        except Exception as err:
            print(
                f'Check if DIAMOND is installed and on the PATH.\n\n{err}')
    return


def rost(fasta_query, diamond_out):
    """
    This function uses the Rost formula to filter the alignment results.
    (https://doi.org/10.1093/protein/12.2.85)
```

```
:param fasta_query: Fasta query to be aligned against the database.
:param diamond_out: Diamond output.
"""
cutoff = []
fasta_out = fasta_query.split('.')[0] + '_valid.fa'
n = 10  # Position of the line on the graphic. In my results, n=10 generates
# more reliable alignmnets betweentruly homologous sequences.
# This block filter the DIAMOND output using the Rost formula to identify
# only relevant alignments.
with open(diamond_out, 'r') as d_out:
    d_out = d_out.readlines()
    for line in d_out:
        length = int(line.split()[3])  # Alignment length
        pi = round(n + (480 * (length ** (-0.32 * (1 + exp(- length / 1000))
                                    ))), 2)  # Identity cutoff
        cutoff.append(pi)

# Making a dataframe to be easier to manipulate the data and get only the
# valid alignments
df_diamond = pd.read_table(diamond_out, delimiter='\t',
                           names=['Query ID', 'Suject ID',
                                  '% identical matches',
                                  'Alignment length', 'Mismatches',
                                  'Gaps_openings',
                                  'Qstart', 'Qend', 'Qlength', 'Sstart',
                                  'Send',
                                  'Slength', 'Evalue', 'Bitscore'])
df_diamond['Validation'] = cutoff
# Getting only the alignments with a identity value greater than the cutoff
df_curated = df_diamond[
    df_diamond['% identical matches'] >= df_diamond['Validation']]
df_curated.to_csv(r'diamond_filt.tsv', sep='\t', index=None, mode='w')

# Getting the queries with valid alignments
```

```
    with open('diamond_filt.tsv', 'r') as diamond_temp:
        diam_temp = diamond_temp.readlines()
        valid_align = [line.split()[0] for line in diam_temp]

    # Creating a fasta with the aproved sequences
    with open(fasta_query, 'r') as fa_inp:
        for record in SeqIO.parse(fa_inp, 'fasta'):
            if record.id in valid_align:
                with open(fasta_out, 'a') as fa_out:
                    SeqIO.write(record, fa_out, 'fasta')
    return


def diamond_pipe(cores, fasta_query, fasta_subject, mode, database):
    """
    This function runs the DIAMOND software and filters the results.
    :param cores: Nº of cores
    :param database: Database to align the fasta_query
    :param fasta_query: Fasta (query) used in the alignment
    :param fasta_subject: Fasta used to build the database (only used if the
    database is empty)
    :param mode: Diamond mode (blastx or blastp)
    :return:
    """
    diamond_out = 'diamond.out'
    if database is None:
        making_db(cores, fasta_subject)
        database = fasta_subject.split('.')[0] + '.dmnd'
        running_diamond(cores, database, fasta_query, mode, diamond_out)
        rost(fasta_query, diamond_out)
    elif database is None and fasta_subject is None:
        raise 'Both database and fasta subject were not defined, at least one' \
              ' has to be.'
    else:
```

```
        running_diamond(cores, database, fasta_query, mode, diamond_out)
        rost(fasta_query, diamond_out)

    return


# Checking if the script is running directly or being imported
if __name__ == '__main__':
    # Program parameters
    parser = ArgumentParser(
        description='Run DIAMOND and filter the valid alignments.')

    parser.add_argument('-fq', required=True, help='fasta query')
    parser.add_argument('-fs', required=False,
                        help='fasta subject to create database')
    parser.add_argument('-c', required=True, help='cores')
    parser.add_argument('-db', required=False, help='database')
    parser.add_argument('-m', required=True, choices=['blastx', 'blastp'],
                        help='alignment mode')

    # Parsing the arguments
    args = parser.parse_args()
    # Executing the function
    diamond_pipe(args.c, args.fq, args.fs, args.m, args.db)
```

8.9 *Script* para filtrar sequências com bases indefinidas

```
from Bio import SeqIO
from argparse import ArgumentParser


def nctd(seq):
    """
    Function that checks if a sequence of nctd is valid.
```

```
    :param seq: Sequence to be analyzed
    :return: Return if the sequence is valid or not
    """
    valid_nuc = 'ATCG'  # Defining valid nucleotides
    for n in seq.upper():
        if n not in valid_nuc:  # Checking if each letter of seq is contained in the valid nctds
            return False
    return True


def ptn(seq):
    """
    Function that checks if a sequence of ptn is valid.
    :param seq: Sequence to be analyzed
    :return: Return if the sequence is valid or not
    """
    valid_ptn = 'ABCDEFGHIKLMNPQRSTVWYZ'  # Defining valid aminoacids
    seq = seq.rstrip('*')  # Control to avoid consider protein sequences with a '*' at end as invalid
    for a in seq.upper():
        if a not in valid_ptn:  # Checking if each letter of ptn is contained in valid aas
            return False
    return True


parser = ArgumentParser(description='Checks if the characters in a sequence within a fasta are valid and generates a '
                                    'fasta with only valid sequences.')  # Describing the program
parser.add_argument('-fi', required=True, help='Fasta input')
parser.add_argument('-fo', required=True, help='Fasta output')
parser.add_argument('-m', required=True, choices=['n', 'p'], help='Validation mode: nucleotide or protein')
args = parser.parse_args()  # Parsing arguments with the args variable
```

```
c_s_total = 0
c_s_final = 0
# Running the script
try:
    with open(args.fi, 'r') as fasta_in:
        for record in SeqIO.parse(fasta_in, 'fasta'):
            if args.m == 'p':  # Checking the mode
                valid = ptn(record.seq)
                c_s_total += 1  # Counting the total of sequences
                if valid:
                    with open(args.fo, 'a') as fasta_out:
                        SeqIO.write(record, fasta_out, 'fasta')
                        c_s_final += 1  # Counting the total of valid sequences
            elif args.m == 'n':  # Checking the mode
                valid = nctd(record.seq)
                c_s_total += 1  # Counting the total of sequences
                if valid:
                    with open(args.fo, 'a') as fasta_out:
                        SeqIO.write(record, fasta_out, 'fasta')
                        c_s_final += 1  # Counting the total of valid sequences
            else:
                raise Exception("Invalid mode! There's only modes n or p in parameter -m.")
except Exception as err:
    print(f'Probably your fasta file is not valid.\nError: {err}')

if c_s_total == 0:
    raise Exception('Total sequences is equal to 0, then your fasta is not adequately formatted.')
else:
    print(f'Total sequences: {c_s_total}')
    print(f'Total valid sequences: {c_s_final}')
```

8.10 Comandos e *scripts* usados para identificar regiões com alinhamento de RNAseq mas sem predição gênica

```
# Comandos do Samtools
samtools sort rp_final.bam
samtools depth -a --reference Rhodnius_prolixus-3.0.3_HiC_masked -o depth.out rp_final.bam

# Script que calcula a média de hits por nucleotídeo

from argparse import ArgumentParser
import numpy as np


def genes_with_hit(gff3):
    """
    Function that gets only the AUGUSTUS predicted genes with intron hints.
    :param gff3: gff3 file
    :return: Returns a list containing genes with hints
    """
    if gff3 is not None:
        file = open(gff3, 'r')
        genes = {}
        for linha in file:
            # Getting the gene name
            if linha.startswith('HiC_scaffold_') and \
                    linha.split()[2] == 'transcript':
                gene = linha.split()[8].split(';')[0].split('ID=')[1]
            # Getting the % of the gene supported by hints
            elif linha.startswith('# % of transcript'):
                hint = float(linha.split()[9])
                if hint > 0:  # If there's hints, the gene value will be True
                    hint = True
                    genes[gene] = hint
```

```
            else:  # If there's no hints, the gene value will be False
                hint = False
                genes[gene] = hint

        genes_hint = [k for k in genes if genes[k]]  # Getting only the genes
        # with hints
        file.close()
        return genes_hint
    return


def get_hits(file, genes_list=None):
    """
    Read the samtools depth output e calculate the mean of reads per nctd.
    :param file: depth file from samtools depth
    :param genes_list: opcional list containing genes with hint from Augustus
    GFF
    :return: Return the mean of reads per ntcd
    """

    file = open(file, 'r')
    genes = {}
    lengths = []
    hits = []

    for linha in file:
        gene = linha.split()[0]  # Gene name
        length = int(linha.split()[1])  # Gene length
        reads = int(linha.split()[2])  # Nº of reads
        if gene not in genes:  # Checking if the gene is in dict
            genes[gene] = [length, reads]
        else:  # If the gene is in dict, the number of reads will increase to
            # gene
            read = genes[gene][1] + reads
```

```
            genes[gene] = [length, read]
    # If a gff3 file is used
    if genes_list is not None:
        genes = {k: genes[k] for k in genes if k in genes_list}

    for k in genes:  # Putting the sizes and hits in lists
        lengths.append(genes[k][0])
        hits.append(genes[k][1])

    len_mean = np.mean(lengths)  # Mean of genes size
    hit_mean = round(float(np.mean(hits)), 2)  # Mean of genes hits
    hit_per_nt = round(hit_mean / len_mean, 2)  # Mean of hits by nctd

    file.close()
    return genes, hit_mean, hit_per_nt


# Describing the program
parser = ArgumentParser(
    description='Script that returns the mean of hits by nctd of a depth file.')
parser.add_argument('-df', required=True, help='Samtools depth output.')
parser.add_argument('-gff3', required=False, help='GFF3 from AUGUSTUS gene '
                                                  'preditcion.')

args = parser.parse_args()  # Parsing the arguments with the args variable

geneshint_list = genes_with_hit(args.gff3)
depth, mean_hit_gene, mean_hit_nt = get_hits(args.df, geneshint_list)

print(f"""Média de hits por gene: {mean_hit_gene}
Média de hits por nucleotídeo: {mean_hit_nt}""")

# Script para identificar os intervalos sem predição
```

```
import os
from argparse import ArgumentParser


def filter_depth(depthfile, threshold, tempfile):
    """
    Function to extract the nctds with the wanted threshold and write it in a
    temporary file.
    :param depthfile: depth file from samtools depth
    :param threshold: number of hits per nctd
    :param tempfile:  name of the temporary file
    :return: Returns the temporary file with all nctds with 'threshold' hits
    """
    file = open(depthfile, 'r')
    out = open(tempfile, 'w')

    for linha in file:
        x = int(linha.split('\t')[2])
        if x >= int(threshold):
            out.write(linha)

    file.close()
    out.close()

    return


def readgff(gff):
    """
    Function to extract the genes location of a gff3.

    :param gff: GFF3 file
    :return: Returns a dict with all gene locations
    """
```

```
    gff = open(gff, 'r')
    dic = {}

    for linha in gff:
        if not linha.startswith('#') and linha.split()[2] == 'gene':
            scaff = linha.split()[0]
            pos_i = linha.split()[3]
            pos_t = linha.split()[4]
            rang = pos_i + '-' + pos_t
            # Adding the gene on the dict
            if scaff not in dic:
                dic[scaff] = [rang]
            # If the gene is already in the dictionary, it add only its
            # positions in the dict
            else:
                dic[scaff].append(rang)

    gff.close()
    return dic


def get_range(depthfile, length):
    """
    Function to generate the continuous intervals of the nctds.
    :param depthfile: temp file create with filter_depth function
    :param length: dezire size of the interval
    :return: Return a dict with the intervals and the counting of total
    intervals and filtered intervals
    """
    rang = []
    dic = {}
    intervalo = {}
```

```
# Finding the last line of the file
with open(depthfile, 'rb') as f:  # Opening the file in binary mode to be
    # faster
    f.seek(-2, os.SEEK_END)
    while f.read(1) != b'\n':
        f.seek(-2, os.SEEK_CUR)
        # Getting the last line to don't lose the last interval
    last_line = f.readline().decode()

with open(depthfile, 'r') as inp:
    for linha in inp:
        scaff = linha.split()[0]  # Getting the scaffold
        nctd = linha.split()[1]  # Getting the nctd position
        # Making sure that the position found doesn't macth with the last
        # line of the file
        if int(nctd) != int(last_line.split()[1]):
            if len(rang) == 0:
                rang.append(nctd)  # Adding the first position
            elif len(rang) > 0:
                # Ensuring that the next position is adjacent to the
                # previous one
                if int(rang[-1]) + 1 == int(nctd):
                    rang.append(nctd)
                else:
                    # If the positions are not adjacent, the first and last
                    # values in the list are used to determine
                    # the range where there is alignment
                    x = rang[0] + '-' + rang[-1]
                    if scaff not in dic:
                        # If the scaffold key is not in the dictionary, it
                        # is added here with the associated value
                        # in list format
                        dic[scaff] = [x]
                        rang = [nctd]
```

```
                else:
                    # If the scaffold key exists, the range is added to
                    # the corresponding list
                    dic[scaff].append(x)
                    rang = [nctd]
        else:  # Resolving the problem of losing the last interval
            rang.append(nctd)
            x = rang[0] + '-' + rang[-1]
            if scaff not in dic:
                dic[scaff] = [x]
                rang = [nctd]
            else:
                dic[scaff].append(x)
                rang = [nctd]

for k in dic:  # Filtering ranges by size
    for i in range(len(dic[k])):
        if abs(int(dic[k][i].split('-')[0]) - int(dic[k][i].split('-')[1])) >= int(length):
            if k not in intervalo:
                intervalo[k] = [dic[k][i]]
            else:
                intervalo[k].append(dic[k][i])

count_total = 0
count_filtro = 0
# Counting the total ranges and the filtered ranges
for k in dic:
    count_total += len(dic[k])
for k in intervalo:
    count_filtro += len(intervalo[k])

return intervalo, count_total, count_filtro
```

```
def number_range(x1, x2, x3, x4):
    """
    Function that compare the gene positions and the intervals.
    :param x1: interval initial position
    :param x2: interval final position
    :param x3: gene initial position
    :param x4: gene final position
    :return: Return True if there's an overlap and False if there's no overlap
    """
    if x3 in range(x1, x2 + 1) or x4 in range(x1, x2 + 1):
        return True
    elif x1 in range(x3, x4 + 1) or x2 in range(x3, x4 + 1):
        return True
    return False


def compair_pos(gff, df):
    """
    Function to check overlap between intervals and gene positions.
    :param gff: Dict with the gene positions of the GFF3
    :param df: Dict with the intervals of the depth file
    :return: Return the intervals without prediction and the counting of these
    intervals
    """
    f = []
    for k in df:  # Comparing dictionaries and seeing if there is an overlap
        # between genes
        for j in gff:
            if k == j:  # Checking if the scaffold is the same for the two genes
                for e in df[k]:
                    for i in gff[j]:
                        pos_df1 = int(e.split('-')[0])
                        pos_df2 = int(e.split('-')[1])
                        pos_gff1 = int(i.split('-')[0])
```

```
                pos_gff2 = int(i.split('-')[1])
                # Checking overlaps
                x = number_range(pos_df1, pos_df2, pos_gff1, pos_gff2)
                if x:
                    f.append(e)  # Adding intervals with overlap

    # Ensuring that all overlapping ranges are removed
    for k in df:
        df[k] = [e for e in df[k] if e not in f]

    nopred_count = 0
    for k in df:
        nopred_count += len(df[k])
    return df, nopred_count


# Describing the program
parser = ArgumentParser(description='This script takes the output from the '
                                    'samtools depth and returns the nucleotide '
                                    'intervals with the size and number of hits'
                                    ' chosen by the user. A gff3 with the gene '
                                    'prediction, from the same sequence where '
                                    'the samtools depth output came from, can '
                                    'be used to identify regions where there is'
                                    ' RNAseq data alignment but no gene '
                                    'prediction.')
parser.add_argument('-df', required=True, help='Result from samtools depth.')
parser.add_argument('-gff3', required=False, help='GFF3 from AUGUSTUS gene '
                                        'prediction.')
parser.add_argument('-n', required=True, help='Number of hits by nucleotide.')
parser.add_argument('-l', required=True, help='Desired interval size.')
parser.add_argument('-out', required=True, help='Output with the nucleotide '
                                        'intervals without prediction.')
```

```
args = parser.parse_args()  # Parsing the arguments with the args variable

temp_file = args.out.split('.')[0] + '.temp'
filter_depth(args.df, args.n, temp_file)
df, total_ranges, filtered_ranges = get_range(temp_file, args.l)
out = open(args.out, 'w')
out.write(f'#Chosen number of hits per nucleotide: {args.n}\n')
out.write(f'#Chosen interval size: {args.l}\n')
out.write(f'#Number of total intervals with {args.n} hits/nt: {total_ranges}\n')
out.write(f'#Number of intervals with {args.l}nt: {filtered_ranges}\n')

if args.gff3 is None:  # If the gff3 file is not used
    out.write('\n')
    for k in df:
        for e in df[k]:
            out.write(f'{k} \t {e} \n')
else:
    gff3 = readgff(args.gff3)
    ranges_nopred, nopred_ranges = compair_pos(gff3, df)
    out.write(f'#Number of intervals of{args.l}nt without prediction: '
              f'{nopred_ranges}\n\n')
    for k in ranges_nopred:
        for e in ranges_nopred[k]:
            out.write(f'{k} \t {e} \n')

out.close()
os.remove(temp_file)
```

8.11 Famílias gênicas em P13

Tabela 8.1 - Expansões de famílias gênicas na predição P13.

Descrição	Aaeg	Apis	Clec	Gmor	Phum	Rpro
Nitroforina	0	0	0	0	0	28
Alérgeno de ácaro, grupo 7	3	2	7	2	1	20
Canal de cloreto	0	0	0	0	0	12

Domínio de lectina SUEL de ligação a D-galactosídeo/L-ramnose	2	1	2	2	1	10
Domínio PLAT/LH2	0	2	0	1	0	8
Multicobre oxidase, tipo 1	0	0	1	2	0	7
Transglutaminase, C-terminal	0	1	0	0	1	7
Triabina/Procalina	0	0	0	0	0	7
Cadeia pesada de dineína 3, domínio de tampa AAA+	0	0	0	0	0	5
Cinurenina formamidase/semelhante à ciclase	0	2	2	0	0	5
Gama-cristalino	0	0	0	0	0	4
Alfa-2-macroglobulina	0	1	1	0	0	3
Fosfato de etanolamina GPI transferase 1, C-terminal	0	0	1	0	0	3
Domínio de paciffastina	0	0	0	0	0	3

Tabela 8.2 - Contrações de famílias gênicas na predição P13.

Descrição	Aaeg	Apis	Clec	Gmor	Phum	Rpro
Domínio de ligação ao DNA tipo Myb/SANT 5	8	126	14	10	0	1
Motivo BESS	4	45	9	6	0	2
Dedo de zinco tipo FLYWCH	12	309	4	4	10	2
Domínio *paired*	8	8	12	9	8	4
Domínio tudor	10	44	8	10	7	4
Imunoglobulina	11	14	18	10	11	5
Imunoglobulina conjunto E	13	31	11	15	12	7