UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

JOÃO FRANCISCO VALIATI

Redes Neurais Aplicadas ao Reconhecimento de Regiões Promotoras na Família *Mycoplasmataceae*

Tese apresentada, como requisito parcial para a obtenção do grau de Doutor em Ciência da Computação

Prof. Dr. Paulo Martins Engel
Orientador

Porto Alegre, julho de 2006.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Valiati, João Francisco

Redes Neurais Aplicadas ao Reconhecimento de Regiões Promotoras na Família Mycoplasmataceae / João Francisco Valiati – Porto Alegre : PPGC da UFRGS, 2006.

150f.: il.

Tese (doutorado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR-RS, 2006. Orientador: Paulo Martins Engel.

1. Bioinformática. 2. Descoberta de conhecimento. 3. Inteligência artificial. 4. Reconhecimento de promotores. 5. Redes neurais artificiais. I. Engel, Paulo Martins. II. Título.

Não consigo conceber um deus que recompense e puna suas criaturas, nem que tenha uma vontade do tipo que experimentamos em nós mesmos. Não consigo, nem quero conceber um indivíduo que sobreviva à sua morte física; que as almas fracas, por medo ou egoísmo absurdo, alimentem esses pensamentos. Eu me satisfaço com o mistério da eternidade da vida e com um vislumbre da maravilhosa estrutura do mundo real, junto com o esforço diligente de compreender uma parte, por menor que seja, da Razão que se manifesta na natureza.

Carl Sagan em Bilhões e Bilhões, citando Einstein, quando o perguntam como é possível enfrentar a morte sem a certeza de uma vida posterior.

AGRADECIMENTOS

MUITO OBRIGADO, a todos que pouco ou muito contribuíram para a realização deste trabalho, em especial:

- Ao meu orientador Prof. Paulo Engel, pela dedicação, acompanhamento, contribuições e opiniões para o desenvolvimento deste trabalho e principalmente pela confiança que demonstrou em relação a minha capacidade;
- À minha família: Pai, Mãe, Clau, Eliane e Iéia, obrigado por todo o apoio imensurável desejado no decorrer deste trabalho, em especial a Eliane por todo nosso debate e desabafo na condição de doutorandos;
- À Pati por todo amor, compreensão e paciência. Por ser ouvinte das questões fundamentais do trabalho e mesmo sem muito entendê-lo me dava forças para seguir adiante. Agradeço à família Pedruzzi por me acolher;
- Aos sempre Amigos (as): Alessandra, André, Cristiano, Daniele, Edicarsia, Estevan, Filipe, Juliane, Juliano, Marcelo, Márcia, Neibal, Renato e Valesca;
- Ao Prof. Sérgio Ceroni e ao bolsista Marcos Carvalho do Centro de Biotecnologia pelas sugestões e dicas para investigação e conduta deste trabalho;
- Ao Dr. Sírio, pelas lições quando era um bolsista de iniciação cientifica;
- Ao colega Sandro Camargo por rotinas usadas nesse trabalho e por nossas discussões para aprimoramento de nossas pesquisas;
- A nossa turminha: Sandro, Michèle e Zé, que tentamos iniciar o laboratório de bioinformática. Aos nossos almoços, cafés e jantas sempre animados;
- As colegas Lú, Vânia e Juliana por nossas discussões das perspectivas de desenvolvimento e conclusão de nossas teses;
- Aos colegas que freqüentam a sala 203 e pelos debates para aperfeiçoar nossas pesquisas;
- À UFRGS, ao Instituto de Informática e ao PPGC por me acolher e pelos meios fornecidos para realização deste curso de Doutorado;
- Ao corpo docente, funcionários e ao pessoal da biblioteca do Instituto de Informática pelo excelente profissionalismo demonstrado;
- À Capes pela concessão da bolsa de estudos, pois sem ela a realização deste curso seria difícil;
- Enfim a todos que sempre torceram por mim.

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS

A	Adenina
ART	Adaptive Resonance Theory
BD	Banco de Dados
BNN	Bayesian Neural Network
C	Citosina
CC	Coeficiente de Correlação
DNA	Ácido Desoxirribonucléico
EM	Expectation Maximization
G	Guanina
GDX	Gradiente Descendente com Termo de Momento
GST	Generalized Suffix Tree
HMM	Hidden Markov Models
HPAM	Hibrid Promoter Analysis Methodology
IGR	Região Intergênica
LMS	Least Mean-Square
LNCC	Laboratório Nacional de Computação Cientifica
LTM	Long Term Memory
MAP	Maximum a posteriori
MCT	Ministério da Ciência e Tecnologia
MCP	McCulloch e Pitts
MDD	Maximal Dependence Decomposition
MHz	Megahertz
MLP	Multilayer Perceptron
MLR	Machine Learning Repository
MOSS	Multi-objetive Scatter Search
mv	milivolts
NCBI	National Center of Biotechnology Information

NNPP	Neural Network Promoter Prediction
ns	Nanosegundos
pb	Pares de Base
PIGS	Projeto Investigação de Genomas Sul
PWM	Position Weight Matrix
RNs	Redes Neurais Artificiais
RNA	Ácido Ribonucléico
RPROP	Resilient Backpropagation
SCG	Scaled Conjugated Gradient
SN	Sensibilidade
SNNS	Stuttgard Neural Network Simulator
SP	Especificidade
T	Timina
TDNN	Temporal Delay Neural Network
TLS	Translation Start Site
TSS	Transcription Start Site
UFRGS	Universidade Federal do Rio Grande do Sul
WAM	Weight Array Model
WTA	Winner Take All
WWW	World Wide Web

LISTA DE FIGURAS

LISTA DE TABELAS

RESUMO

Este trabalho apresenta o estudo, investigação e realização de experimentos práticos, empregados na resolução do problema de reconhecimento de regiões promotoras em organismos da família *Mycoplasmataceae*. A partir disso, é proposta uma metodologia para a solução deste problema baseada nas Redes Neurais Artificiais.

Os promotores são considerados trechos de uma seqüência de DNA que antecedem um gene, podem ser tratados como marcadores de uma seqüência de letras que sinalizam a uma determinada enzima um ponto de ligação. A posição onde se situa o promotor antecede o ponto de início do processo de transcrição, onde uma seqüência de DNA é transformada em um RNA mensageiro e, este potencialmente, em uma proteína.

As Redes Neurais Artificiais representam modelos computacionais, inspirados no funcionamento de neurônios biológicos, empregadas com sucesso como classificadores de padrões. O funcionamento básico das Redes Neurais está ligado ao ajuste de parâmetros que descrevem um modelo representacional.

Uma revisão bibliográfica de trabalhos relacionados, que empregam a metodologia de Redes Neurais ao problema proposto, demonstrou a sua viabilidade. Entretanto, os dados relativos à família *Mycoplasmataceae* apresentam determinadas particularidades de difícil compreensão e caracterização, num espaço restrito de amostras comprovadas.

Desta forma, esta tese relata vários experimentos desenvolvidos, que buscam estratégias para explorar o conteúdo de seqüências de DNA, relativas à presença de promotores. O texto apresenta a discussão de seis experimentos e a contribuição de cada um para consolidação de um framework que agrega soluções robustas consideradas adequadas à solução do problema em questão.

Palavras-Chave: Bioinformática, Descoberta de Conhecimento, Inteligência Artificial, Reconhecimento de Promotores, Redes Neurais Artificiais.

Application of Artificial Neural Networks to Promoter Recognition Regions in *Mycolasmataceae* Family

ABSTRACT

This proposal reports the study, investigation and practical experiments applied for recognition of promoter regions in DNA sequences of the *Mycoplasmataceae* organisms family. The proposed methodology for solution of this problem is based on Artificial Neural Networks.

The promoters are considered specific small sub-sequences of a DNA sequence that precede a gene; they can be considered as landmarks of a sequence of letters, that signalize a binding point to a specific enzyme. The place where the promoter is situated precedes the start point for transcription process, at this point the DNA sequence is converted into the RNA messenger and this in a protein.

The Artificial Neural Networks represent computational models inspired on the functioning of biological neurons and, with the development of research field, they are sucefully applied as pattern classifiers. The Neural Network basic functionality is to adjust its parameters that specify a representational model.

A review of related works in the literature, that apply the Neural Network methodology for the proposed problem, revealed its successful applicability. However the information about the *Mycoplasmataceae* family shows particularities, which are hard to understand and characterize, especially because there is a restrict set of verified samples.

This thesis reports several experiments, which represent an otimized strategy to explore the content of DNA sequences, related with promoter presence. The text discusses six experiments and the contribution of each one in the development of robust methodologies, able to solve this problem.

Keywords: Artificial Intelligence, Artificial Neural Networks, Bioinformatics, Knowledge Discovery, Promoter Recognition.

1 INTRODUÇÃO

Desde a elucidação da dupla hélice de DNA por Watson e Crick em 1953, uma série de avanços na área de Biologia Molecular tem permitido não somente o fácil acesso às seqüências de diversos genomas dos mais diferentes organismos, mas também o desenvolvimento de várias metodologias que permitem a descoberta e exploração das funcionalidades relacionadas ao DNA e proteínas.

No entanto, o grande volume de informações produzidas diariamente, fruto das políticas de seqüenciamento[1] dos mais variados organismos, está agregado a uma complexidade intrínseca, de natureza interdisciplinar, que une ciências como: química, física, matemática, computação, entre outras, para produzir uma sinergia capaz de oferecer alternativas viáveis de entendimento e solução de problemas relacionados aos mecanismos genéticos envolvidos.

O processo de seqüenciamento de um organismo é somente a ponta do *iceberg*, é apenas o meio que permite a obtenção de diversas seqüências fragmentadas pela técnica empregada e que necessitam ser conectadas para se obter o DNA inteiro de um organismo. Uma vez consolidada essa informação é que iniciam os problemas para desvendar as questões relativas aos diversos mecanismos envolvidos na regulação dos processos do DNA, na identificação de possíveis genes pela anotação, na identificação de proteínas e suas respectivas famílias, assim como a estrutura conformacional que cada proteína pode assumir, dependendo do papel desempenhado no organismo, ou mesmo identificar as vias metabólicas que permitem a interligação dessas funcionalidades. Durante e após todas essas questões serem levantadas, se busca um elo de ligação com os demais organismos que compartilham as características comuns, por pertencerem à mesma família, ou devido ao meio em que vivem.

Consulta atual (maio de 2006) ao site do NCBI (*National Center of Biotechnology Information*), um dos portais mundiais centralizador de todos os projetos genoma correntes, revelou que atualmente existem 341 genomas procarióticos e 20 genomas eucarióticos completamente seqüenciados e mais 580 genomas procarióticos e 205 genomas eucarióticos que estão sendo seqüenciamento, tudo isto não se contabilizando plantas, fungos e protozoários. No panorama nacional, identificamos 10 genomas procarióticos já seqüenciados.

Para dar uma idéia da quantidade de informação, citamos como exemplo o genoma da bactéria *Escherichia coli* com $5x10^6$ nucleotídeos e o genoma humano com $3x10^9$ nucleotídeos. Nessa imensa quantidade de informação, alguns genomas estão bem caracterizados, como o da *E. coli* utilizado como organismo padrão para a compreensão

[1] Processo de obtenção da informação presente no DNA

de outros organismos. No entanto, muito ainda se desconhece, já se percebe que de uma forma geral a vida apresenta uma tendência natural de se desviar do padrão devido às condições de vida exigidas para cada organismo, seja pelo seu hábitat ou sua necessidade de sobrevivência.

Dadas as inúmeras funcionalidades dos organismos e a estrondosa quantidade de dados já produzidos e que estão sendo obtidos, é clara a necessidade por novos métodos de investigação. Em parte, a compreensão do conteúdo das seqüências moleculares requer a criação de ferramentas novas e específicas, como o emprego da Inteligência Artificial a técnicas sofisticadas de reconhecimento de padrões.

Desta forma, esta tese relata a investigação de um mecanismo de regulação do DNA, chamado de reconhecimento de promotores, relativo a organismos da família *Mycoplasmataceae*. Para realizar esta investigação foram utilizados Bancos de Dados Biológicos, técnicas para transformação de seqüências biológicas em amostras discretizadas e o emprego das Redes Neurais Artificiais (RNs), subárea da Inteligência Artificial, para a criação de modelos computacionais. A associação destas estratégias possibilitou a construção de um framework capaz de contribuir no processo de identificação de promotores no DNA.

1.1 Breve Histórico Motivador

Essa seção pretende fornecer ao leitor uma breve apresentação dos motivos que conduziram o autor à realização dessa pesquisa.

Ao longo do curso de mestrado trabalhei com as RNs aplicadas ao problema de identificação de comandos de voz. Depois de concluída essa etapa continuei investigando a aplicação dessas redes a outros problemas voltados a tarefas de Descoberta do Conhecimento e Mineração de Dados. Ao mesmo tempo, freqüentava disciplinas, na qualidade de aluno especial no Instituto de Informática da UFRGS, realizando cadeiras introdutórias de Bioinformática, o que despertou interesse pela nova área desafiadora.

A implantação de um laboratório de Bioinformática, associação do Centro de Biotecnologia com o Instituto de Informática da UFRGS, abriu as portas para a possibilidade de desenvolvimento de pesquisa na área. O laboratório se propunha a coordenar ações relacionadas ao processamento e análise das informações derivadas do seqüenciamento de genomas pela Rede PIGS (Programa de Investigação de Genomas Sul), inserida no Projeto Rede Sul de Análise de Genomas e Aplicações. Este laboratório seria responsável inicialmente pelo seqüenciamento e análise do genoma do organismo *Mycoplasma hyopneumoniae*, um agente infeccioso importante da suinocultura. O projeto tinha como intuito identificar seqüências e proteínas importantes para o desenvolvimento de metodologias mais rápidas e precisas de diagnóstico, a fim de definir proteínas potenciais para o desenvolvimento de vacinas.

Isto propiciou contato com especialistas do Centro de Biotecnologia e despertou para a necessidade de ferramentas computacionais que auxiliassem na resolução de problemas pertinentes. Assim, se optou em um primeiro momento pelo estudo e investigação de promotores no organismo alvo do projeto.

Por um lado o problema se mostrou interessante e desafiador. Qualquer ferramenta de eficiência comprovada, que forneça bons indicativos a especialistas da área sobre um determinado problema é vista como uma forma de se ganhar tempo e

economizar recursos, uma vez que o custo operacional dessa pesquisa é bastante elevado, tanto pelos materiais empregados para obtenção de reações, quanto pelo alto nível requisitado pelos recursos humanos especializados.

Uma revisão bibliográfica da aplicação das RNs ao problema em questão demonstrou a viabilidade de seu uso a diversos problemas sugeridos pela Bioinformática, relacionados ao reconhecimento de padrões, entre eles o reconhecimento de promotores.

1.2 Objetivos do Trabalho

O principal objetivo deste trabalho é o desenvolvimento de uma metodologia e aplicação computacional, capaz de prover suporte ao reconhecimento de promotores em organismos da família *Mycoplasmataceae*. Esta metodologia servirá de base para a proposta de um framework para reconhecimento de promotores.

Para tal, são empregadas técnicas para aquisição de seqüências, identificação e seleção de características relevantes, produção de conjuntos de amostras, emprego de RNs e realização de testes estatísticos, que apontem a capacidade da metodologia empregada no reconhecimento das regiões de interesse. Todo esse esforço tem a intenção de fornecer maiores subsídios ao pouco que se conhece atualmente sobre este problema e contribuir no desenvolvimento de novas metodologias computacionais destinadas à solução de problemas biotecnológicos.

1.3 Definições dos Capítulos

Esta tese está organizada em cinco capítulos, além da introdução, que visam fornecer ao leitor: uma idéia do problema tratado, da principal metodologia computacional empregada e dos experimentos realizados com seus respectivos resultados obtidos.

Assim, o capítulo 2 apresenta os conceitos biológicos básicos necessários para uma maior compreensão do problema abordado, expondo os elementos envolvidos e os mecanismos de funcionamento da célula e do DNA, até definir as regiões promotoras em organismos procarióticos e a dificuldade de identificá-las.

O capítulo 3 aborda os princípios das Redes Neurais Artificiais, apresentando: origem, definições, forma de funcionamento e principais arquiteturas, reservando uma seção de revisão do estado da arte onde são relatados trabalhos atuais e seu uso ao problema em questão sobre outros organismos procarióticos.

O capítulo 4 relata em ordem cronológica seis experimentos investigativos com a aplicação de modelos neurais ao reconhecimento de promotores. Nele são descritos: os dados utilizados, o pré-processamento empregado, os experimentos realizados, os resultados e uma discussão do que se obteve de solução para cada experimento.

O capítulo 5 consolida em um nível abstrato todas as abordagens exloradas, em um framework com detalhamento das abordagens desenvolvidas.

No capítulo 6 são apresentadas as conclusões desta tese, sendo expostas as contribuições encontradas com sugestão de metodologias para solução do problema, uma discussão das limitações e viabilidade do trabalho desenvolvido e expostas sugestões de trabalhos futuros.

2 CONCEITOS BIOLÓGICOS

Este capítulo aborda a definição de conceitos biológicos básicos necessários para uma maior compreensão do tema abordado. Inicialmente é apresentada uma diferenciação dos organismos procarióticos e eucarióticos, seguindo para o detalhamento dos principais elementos que constituem uma célula, exposição da informação que constitui genes e genomas, descrição dos processos envolvidos na transformação do DNA em proteína, e finalmente é abordado o problema de reconhecimento de regiões promotoras em bactérias, com foco nos organismos *Escherichia coli* e *Mycoplasmas*.

2.1 Organismos Procarióticos e Eucarióticos

Em geral, os organismos apresentam a célula como unidade estrutural e seguindo esta premissa foram divididos em dois grupos que fazem referência a uma característica fundamental da célula: a ausência ou a presença de núcleo celular organizado por uma membrana.

Os procarióticos representam o grupo de organismos unicelulares, cuja organização não apresenta membrana nuclear. O conteúdo nuclear está em contato direto com o citoplasma da célula não estando individualizado, por uma membrana. Os indivíduos que compõem este grupo são considerados uma das formas mais primitivas da matéria viva por aparentarem massas de proteínas homogêneas e amorfas e na natureza pertencem ao reino Monera que compreende as arqueas, bactérias e cianofíceas (algas) (AMABIS e MARTHO, 1995).

Pelo fato de os procariotos apresentarem uma organização bastante simples, acreditava-se que suas atividades metabólicas eram reduzidas. No entanto, é comprovado que as bactérias possuem uma capacidade de síntese bastante elevada, podendo dobrar de massa em 20 minutos e se dividir, em seguida, em duas novas bactérias filhas capazes de crescer e se dividir no mesmo ritmo (ALBERTS et al., 1996).

O outro grupo é formado pelos eucariotos, que ao contrário dos anteriores se caracterizam pela presença de um núcleo bem definido na composição da organização celular, apresentando uma membrana envolvendo os componentes nucleares. Aos eucarióticos pertencem os organismos unicelulares e multicelulares, englobando tanto animais como vegetais que estão distribuídos nos reinos: Protista (protozoários, algas e fungos), Plantae (briófitas e traqueófitas) e Animallia (de poríferos até cordados, onde se encontra o homem) (GOWDAK e MATTOS, 1995).

2.2 A Célula

A célula é a unidade funcional que compõe os organismos vivos sejam eles procarióticos ou eucarióticos (HUNTER, 1993). A célula é a menor porção de um organismo, capaz de viver como unidade dependente de um todo ou livremente sob certas condições. É o local onde ocorrem todas as reações metabólicas de um organismo, como: transporte de substâncias, produção e transformação de energia para dar suporte à sua sobrevivência. Por exemplo, as células das folhas com pigmento verde transformam energia solar em energia química ou as células animais e vegetais são capazes de transformar energia de um composto energético (alimento) em energia mecânica para a manutenção de suas atividades vitais.

Uma única célula executa todas as funções vitais nos organismos unicelulares. Nos organismos pluricelulares, tais funções são realizadas por células especializadas reunidas em tecidos, que em conjunto compõem os órgãos, que por sua vez associados a outros órgãos, para a realização de uma mesma tarefa, constituem os aparelhos ou sistemas. O conjunto de aparelhos ou sistemas forma o indivíduo e permite a sua sobrevivência. Alguns tecidos são bastante familiares: ossos, músculos, nervos; outros tecidos fazem parte do aparelho digestório, respiratório, urinário e reprodutivo. A pele e o sangue são tipos de tecidos distintos, feitos por células altamente especializadas. Tecidos endócrinos compreendem uma rede de glândulas produtoras de hormônios que exercem um controle global sobre vários aspectos do corpo do ser humano como um todo (GOWDAK e MATTOS, 1995).

Somente nos organismos vertebrados (que possuem espinha dorsal) é estimado que existam mais de 200 tipos diferentes de células especializadas (ALBERTS et al., 1996). Destas, algumas são grandes outras pequenas, por exemplo, uma única célula nervosa do homem liga seu pé à corda espinhal; ao mesmo tempo em que uma gota de seu sangue possui mais de 10 mil células. Algumas se dividem rapidamente, outras não se dividem totalmente: as células da medula óssea se dividem todas em poucas horas, enquanto que as células nervosas podem viver 100 anos sem nunca se dividir. Embora toda essa diversidade, todas as células nos organismos multicelulares têm exatamente o mesmo código genético, ou seja, uma célula da pele possui o mesmo código genético de um linfócito ou hepatócito. A diferença entre elas está na expressão gênica, isto é, o produto de um gene que será produzido ou não, em determinada quantidade, para suprir as necessidades do organismo.

De uma forma geral, todas as células independentemente do organismo a que pertençam contêm citoplasma e material genético, envoltos em uma membrana e são dotadas de mecanismos básicos para permitir a tradução das mensagens genéticas no principal tipo de molécula biológica, a proteína. A fig. 2.1 apresenta alguns dos elementos básicos que constituem a célula, em que se destacam:

- **Membrana Celular**: é a fronteira da célula com o espaço extracelular que delimita o recinto celular. Quimicamente, é formada por uma bicamada de fosfolipídios e proteínas, onde os fosfolipídios são lipídios (gorduras) associados a um grupo fosfato. A extremidade com o grupo fosfato é hidrofílica (que possui afinidade pela água) e as caudas do fosfolipídio são hidrofóbicas (que repele a água). A membrana das células possui duas camadas destas moléculas, com um lado hidrofóbico interno e um lado hidrofílico externo (MINATTI, 2003). Assim, a água e outros materiais são contidos pela membrana, sendo o contato com o

meio externo permitido somente por poros ou canais especiais.

Muitas das ações de permeabilidade seletiva, reconhecimento do limite celular e interações intracelular/intercelular acontecem na membrana. Nos organismos unicelulares, a membrana tem habilidade para englobar partículas que sirvam de alimento para a célula ou até desempenhar um papel locomotor para o organismo. Em bactérias e arqueas, a membrana tem um papel fundamental na produção de energia pela manutenção da grande diferença de acidez entre as partes interna e externa da célula. Nos organismos multicelulares, a membrana contém todos os sinais envolvidos no mecanismo de tradução e adesão de moléculas, entre outros.

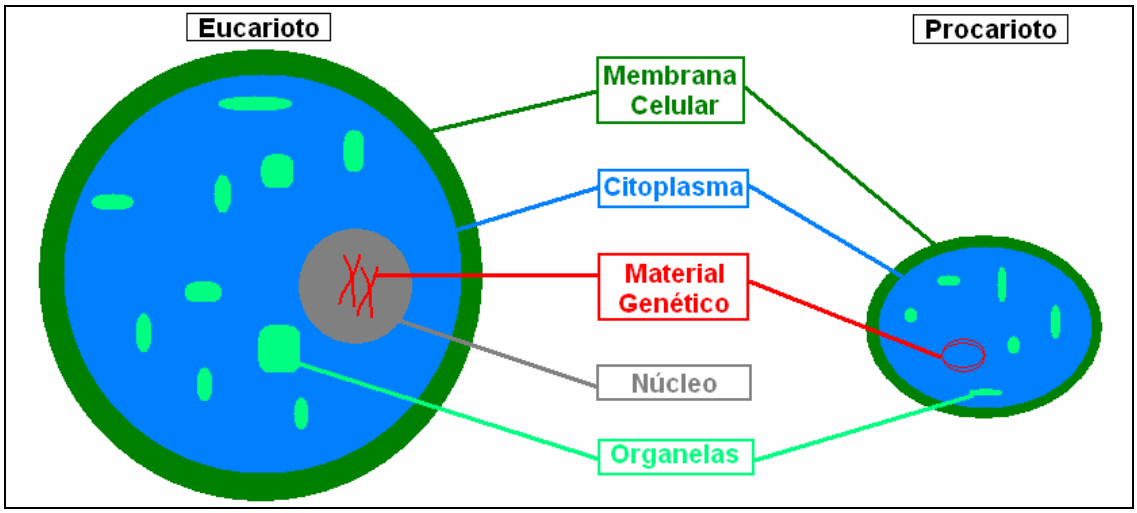


Figura 2.1: Principais partes da célula

- **Citoplasma**: é a porção da célula composta por uma forma de gel que agrega diferentes substâncias e estruturas do interior da célula. Nas bactérias, o citoplasma contém todo o material biológico, enquanto que nos organismos eucarióticos ele se encontra presente entre o núcleo e a membrana celular (COOPER e HAUSMAN, 2004).
- **Núcleo**: é a principal característica das células eucarióticas, contendo o material genético da célula sob forma de cromatina (substância que tem grande afinidade por certos corantes). A cromatina contém longas seqüências de DNA sob uma variedade de conformações e cercada por proteínas nucleares. O núcleo está separado do restante da célula por uma membrana nuclear, sendo considerado a parte mais visível da maioria das células e pode ser identificado por uso do microscópio (LODISH et al., 2000).
- **Material Genético**: é o código para todos os outros constituintes da célula. Esta informação está geralmente armazenada em longas fitas de DNA, sob forma linear nos eucariotos e circular nos procarióticos. Já na maioria dos vírus o material genético está armazenado no RNA. O material genético contém o plano para a geração de todas as proteínas que a célula pode produzir (BONATO, 2005).
- **Organelas:** são os demais elementos contidos no citoplasma responsáveis por funções específicas para a manutenção da célula.

Alguns exemplos são**:** ribossomos (agrupamento de proteínas que participam da tradução da informação genética em proteínas), retículo endoplasmático (síntese de lipídios e proteínas da membrana), complexo de Golgi (síntese e transporte de moléculas) e mitocôndria (especializada no metabolismo oxidativo) (ZAHA et al., 2003).

2.2.1 Componentes Inorgânicos da Célula

Os componentes inorgânicos da célula são água e sais minerais.

A água é o componente que existe em maior quantidade na célula constituindo cerca de 70% do peso de qualquer célula. Seu papel dentro da célula é fundamental uma vez que é o solvente de íons minerais e outras substâncias, isto permite que as enzimas participem de processos metabólicos no interior da célula, servindo também como meio de transporte para a eliminação e aquisição de substâncias pelas células (ALBERTS et al., 1996).

Os sais minerais constituem pequenas moléculas como açúcares e íons inorgânicos e estão presentes nos seres vivos sob duas formas: 1) dissociados em íons, em concentrações e funções variadas, por exemplo, K+ e Na+ que participam dos fenômenos de condução nervosa pelos neurônios; 2) ou, na forma cristalina de carbonatos ou fosfatos, na constituição de esqueletos (GOWDAK e MATTOS, 1995).

2.2.2 Componentes Orgânicos da Célula

As substâncias que transportam e regulam as reações químicas são referidas como biomoléculas. As biomoléculas incluem as proteínas, os carboidratos e os lipídios.

O material genético especifica como, quando e o quanto de proteínas é necessário para a manutenção celular. Tais proteínas terão a tarefa de controlar o fluxo de energia e materiais na célula, participando da transformação de carboidratos, lipídios e outras moléculas, cumprindo todas as funções essenciais para a célula. O próprio material genético é conhecido por ser uma macromolécula particular, o DNA.

2.2.2.1 Proteínas

As proteínas exercem papéis cruciais em todos os processos biológicos e fazem parte dos componentes celulares, sendo moléculas muito mais complexas que os carboidratos ou lipídios. Algumas proteínas, na forma de enzimas, funcionam como catalisadores para acelerar certas reações químicas; também atuam como transportadoras e armazenadoras de muitas moléculas e iontes. As proteínas podem habilitar alguns tipos celulares e organismos com capacidade de contraírem-se, mudarem de forma ou se deslocarem no meio ambiente; conferindo assim, movimento coordenado a estas células. Outro papel importante das proteínas é fornecer proteção ou resistência a estruturas biológicas, constituindo filamentos e formando o citoesqueleto celular. Apesar da grande diversidade nas tarefas, as proteínas são constituídas basicamente por aminoácidos que compartilham os elementos carbono (C), hidrogênio (H), oxigênio (O) e nitrogênio (N). Para uma maior compreensão algumas definições são apresentadas:

a)Aminoácidos: são compostos orgânicos de fórmula geral apresentada na fig. 2.2.

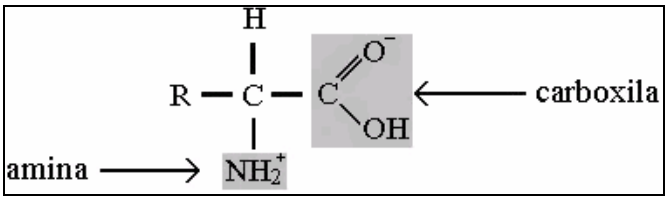


Figura 2.2: Fórmula geral dos aminoácidos

Os grupos carboxila e amina, são constantes nos 20 aminoácidos existentes na natureza, estando ligados a um átomo de carbono (C), que por sua vez está ligado a um átomo de hidrogênio (H). O radical R é que vai apresentar uma composição química variável determinando o tipo do aminoácido. Um polímero de aminoácidos é que dá origem a uma proteína.

Na natureza, somente os vegetais produzem os 20 aminoácidos diferentes, de onde são sintetizadas as proteínas. Os animais produzem alguns deles e os demais são retirados de alimentos de origem vegetal. Aos aminoácidos sintetizados pelo animal dá-se o nome de naturais e os obtidos pela alimentação são chamados de essenciais. Na ausência dos essenciais, a célula animal não consegue sintetizar determinadas proteínas, o que compromete o desenvolvimento do organismo vivo (GOWDAK e MATTOS, 1995).

b)Ligação peptídica: é o nome dado quando ocorre a ligação entre uma carboxila de uma molécula de aminoácido com o grupo amina de outra molécula de aminoácido, ocorrendo liberação de uma molécula da água a cada ligação peptídica, fig. 2.3, constituindo uma ligação N-terminal para C-terminal devido aos elementos presentes em suas pontas.

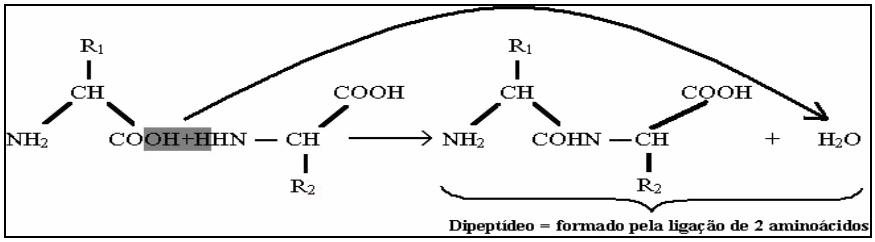


Figura 2.3: Ligação peptídica

A uma ligação contendo um número *n* de aminoácidos dá-se o nome de polipeptídeo. Desta forma, todas as proteínas são polipeptídeos. Muitas das proteínas existentes apresentam um número médio de 300 aminoácidos em sua composição, no entanto pode haver proteínas com 100 ou 5000 aminoácidos.

c)Especificidade: trata da ordenação da seqüência de aminoácidos ao longo da molécula. Duas proteínas diferentes podem resultar, por hidrólise[2], nos mesmos aminoácidos em mesmas proporções. No entanto, o que vai diferenciar estas duas proteínas é a ordem dos aminoácidos nelas inseridos, uma vez que a conformação adquirida por cada proteína vai variar em decorrência desse posicionamento, implicando a função desempenhada pela proteína (COOPER e HAUSMAN, 2004).

d)Enzimas: são um tipo especial de proteínas que agem como

[2] Quebra de uma molécula pela água.

biocatalizadoras, ou seja, aceleram ou retardam o metabolismo do organismo.

2.2.2.2 Carboidratos

Os carboidratos, também conhecidos como açúcares, são compostos orgânicos de origem vegetal, formados por carbono (C), hidrogênio (H) e oxigênio (O) que servem de combustível para os organismos. Por exemplo, a quebra da molécula de glicose (glicólise) em duas moléculas de piruvato, duas moléculas de NADH e duas moléculas energéticas (ATP), representa uma importante fonte de energia.

Estes açúcares apresentam grande importância biológica e estão divididos em monossacarídeos, dissacarídeos e polissacarídeos. Os monossacarídeos mais importantes são as pentoses: ribose e desoxirribose e as hexoses: glicose (molécula básica para obtenção de energia), frutose e galactose (açúcar do leite materno). Dentre os dissacarídeos destacam-se: a maltose, a sacarose (açúcar comum) e a lactose (açúcar do leite de vaca). E os principais polissacarídeos são: o amido (reserva energética vegetal), o glicogênio (reserva energética animal) e a celulose (GOWDAK e MATTOS, 1995).

2.2.2.3 Lipídios

Os lipídios representam um conjunto de substâncias químicas caracterizadas pela alta solubilidade em solventes orgânicos e baixa solubilidade em água. Juntamente com proteínas, ácidos nucléicos e carboidratos são componentes essenciais das estruturas biológicas, estando presentes em todos os tecidos, principalmente nas membranas e nas células de gordura. Ao contrário das demais moléculas, os lipídios não são polímeros, isto é, não são repetições de unidades básicas.

Apesar de a estrutura química ser simples as funções realizadas pelos lipídios são complexas, atuando em etapas cruciais do metabolismo e na definição das estruturas celulares. Dentre algumas habilidades dos lipídios estão formar filmes sobre a superfície da água ou a capacidade de formar “microenvelopes” que envolvem moléculas orgânicas e as entregam no “endereço biológico” correto (MINATTI, 2003).

2.2.2.4 Ácidos Nucléicos

Toda a informação genética de qualquer criatura viva está armazenada no DNA, ácido desoxirribonucléico e no RNA, ácido ribonucléico. Os componentes básicos para formação de uma cadeia de DNA ou RNA são quatro unidades simples de ácidos nucléicos, conhecidos como nucleotídeos ou pares de base.

No DNA e RNA os nucleotídeos se dividem em dois grupos: os púricos, adenina (A) e guanina (G) e os pirimídicos, timina (T) e citosina (C). No RNA a base pirimídica timina (T) é substituída por uracila (U) (ALBERTS, 2003). Cada nucleotídeo é composto por três partes: uma base (púrica ou pirimídica), um açúcar (desoxirribose no DNA ou ribose no RNA) e um ou mais grupos fosfato.

Representativamente, na composição das seqüências de nucleotídeos estão sempre abreviados em suas primeiras letras, como por exemplo, CTTGACAT. A ligação de um nucleotídeo com o próximo na seqüência é direcional, uma fita de DNA apresenta uma extremidade chamada 5' e uma cauda chamada 3', isto devido à posição do átomo de carbono onde ocorre a ligação entre uma base com a próxima de forma direcional, ou seja, uma base se liga à outra pelas posições 5 e 3 de carbono de seu açúcar. A razão de se designar nucleotídeos como pares de base é devido ao fato de o

DNA consistir de duas fitas complementares ligadas entre si, sendo que a fita complementar apresenta uma direção oposta (3' - 5'). Outra característica do DNA é ter um formato de dupla hélice, fig. 2.4, onde as fitas de polipeptídios são mantidas unidas por pontes de hidrogênio: as adeninas ligam-se exclusivamente com as timinas (A=T) por duas pontes de hidrogênio, enquanto que as guaninas ligam-se exclusivamente com as citosinas (G≡C) por três pontes de hidrogênio (SETUBAL e MEIDANIS, 1997).

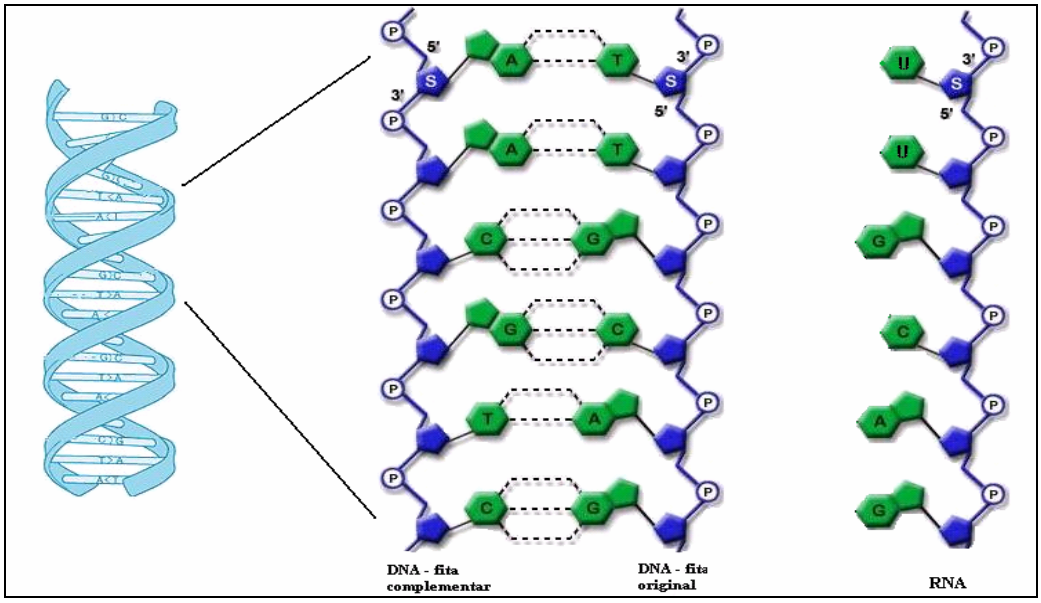


Figura 2.4: Molécula de DNA e RNA

2.3 Genes e Genoma

Toda a informação genética de um organismo pode ser armazenada em uma ou mais moléculas distintas de DNA ou RNA, cada uma dessas moléculas é chamada de cromossomo, o qual está inserido no núcleo dos organismos eucarióticos e presente no citoplasma dos procarióticos (FELSENFELD e GROUDINE, 2003).

Os ácidos nucléicos do DNA carregam codificada a informação da estrutura primária[3] de uma proteína. Cada tripla, não sobreposta, de nucleotídeos, chamada de *códon*, corresponde a um aminoácido particular. Como existem quatro nucleotídeos (bases) diferentes é possível formar 64 (4^3) triplas diferentes, o que é mais do que necessário para codificar os 20 aminoácidos existentes (ATTWOOD, 1999). A razão de haver *códons* em excesso é devido ao fato que a maioria dos aminoácidos é codificada por mais de um *códon* e três destes possíveis (64) *códons* indicam a terminação de uma seqüência protéica (HOOD e GALAS, 2003).

O processo de sintetizar proteínas de uma seqüência de *códons* em uma seqüência de aminoácidos parece bastante básico. Entretanto, existe uma variedade de problemas, uma vez que existem três possíveis lugares para iniciar a interpretação de um segmento de DNA. Por exemplo, para a sentença CATGCCTAGGTGT poderia ler-se: CAT-GCC-TAG-GTG, onde o último T é ignorado; ou ATG-CCT-AGG-TGT, onde o primeiro C é ignorado; ou TGC-CTA-GGT, onde as duas primeiras bases (CA) e as duas últimas bases (GT) são deixadas de fora. Cada uma destas interpretações é

[3] Seqüência de aminoácidos que formam uma cadeia polipeptídica.

chamada de um *reading frame*. Uma interpretação de uma seqüência suficientemente longa de *códons* sem a intervenção de *stop códons* (*códons* terminais de uma seqüência protéica) é chamada de *open reading frame*, ou simplesmente *ORF*, e deveria ser traduzida em uma proteína. Alguns organismos codificam proteínas diferentes com sobreposição de *reading frames*, que pelo processo de leitura acarreta a alteração de um caractere e isso resulta numa proteína completamente diferente, mas com uma possível funcionalidade.

Note que não são possíveis apenas três *reading frames* numa seqüência de DNA, é preciso lembrar que o DNA é formado de uma dupla fita e que esta outra fita é um complemento da primeira. Voltando ao exemplo anterior (CATGCCTAGGTGT), a fita complementar é a seqüência ACACCTAGGCATG que é lida invertida e em sentido oposto. E assim, pode ser interpretada de três outras formas, permitindo um total de seis *reading frames* para todas as seqüências de DNA (SETUBAL e MEIDANIS, 1997).

O gene é considerado a porção do DNA que contém a informação necessária para a produção de proteínas, essenciais para a manutenção do organismo. Um exemplo clássico é comparar o DNA a um denso texto, escrito com um alfabeto de quatro letras (bases). Este texto geralmente é bastante extenso, uma vez que se tratando do DNA humano é formado por aproximadamente 3 bilhões de bases (letras ou caracteres), o que permitiria escrever uma vasta enciclopédia com centenas de volumes (DULBECCO, 1997). Dentro deste texto, o gene pode ser tratado como frases soltas nesse emaranhado de informações; identificar os genes significa identificar essas frases. Da mesma forma, que em algumas situações é complicado interpretar um texto, é muito mais difícil compreender a informação de um gene, uma frase, que possui um significado oculto (BALL, 2003). Ao conjunto de todos os genes de um organismo é dada a denominação de genoma.

No entanto, a célula é dotada de mecanismos apropriados, proteínas de reconhecimento, que identificam a localização dos genes dentro do DNA realizando a decodificação de determinadas regiões (palavras-chave) que se encontram próximas aos genes. A maior dificuldade encontrada é que cada gene possuí uma palavra-chave específica para si, tornando sua identificação uma tarefa árdua já que é estimado que existam de 30000-35000 genes no caso do genoma humano (HOOD e GALAS, 2003).

A informação presente no DNA que não codifica genes é dado o nome de *íntron*. Como freqüentemente são encontrados nos organismos eucarióticos (a maioria das bactérias não apresenta *íntrons*), estes são dotados de uma variedade de diferentes sistemas capazes de reconhecer e remover estes *íntrons* (DULBECCO, 1997). *Exon* é chamada a seqüência do DNA que resulta em uma proteína.

O DNA apresenta uma grande quantidade de informação, além das seqüências que codificam proteínas. Todas as células de um organismo têm o mesmo DNA, no entanto cada tipo de célula produz um conjunto diferente de proteínas dependendo de sua localização e função dentro do organismo, ou decorrente do ambiente onde se encontra.

Diante da gama de informações contidas no DNA de um organismo e do mecanismo de controle gerencial de todas as atividades realizadas na célula automaticamente, torna-se mais claro o papel da informática no tratamento da informação proveniente de projetos genoma, que deve ser capaz de prover ferramentas para dar suporte à análise destes dados (GIBAS e JAMBECK, 2001).

2.4 O Fluxo da Informação Genética

O dogma central da biologia molecular é representado na fig. 2.5, nela se observa que o DNA atua como um modelo de auto-replicação e é a base para transcrever o RNA que por sua vez é fonte para tradução de proteínas.

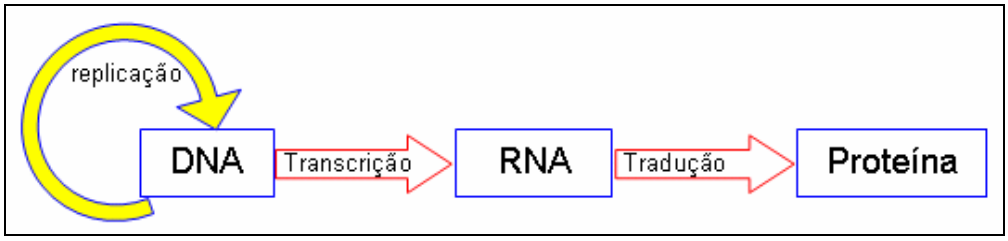


Figura 2.5: Dogma central da biologia molecular

Este princípio agrega as funções do genoma em termos da sua informação constituinte, por meio da informação genética armazenada no DNA que é conservada e transmitida para sua progênie através do processo de replicação; e que também fornece manutenção a um organismo individual através da transcrição e tradução, no nível estrutural, bioquímico e celular (GIBAS e JAMBECK, 2001).

Desta forma, o DNA é uma informação fundamental por permitir a continuidade da existência e juntamente com a atuação da maquinaria celular torna possível a dinâmica da vida.

2.4.1 Replicação do DNA

A célula é a unidade fundamental de um ser vivo, uma única célula dá origem a muitas outras células através de um processo de repetição serial conhecido como divisão celular. No entanto, antes de cada divisão celular é necessária a duplicação das muitas moléculas que a compõem, incluindo o DNA (ALBERTS, 2003). Para o DNA dar continuidade ao material genético presente na sua composição, um mecanismo conhecido como replicação é implementado. Este processo ocorre quando um complexo protéico (DNA-polimerase) se insere entre os dois filamentos do DNA (dupla fita), separa a ligação química entre as bases, desenrola a macromolécula e produz para cada uma dessas fitas uma nova fita, já ligando as bases existentes às novas produzidas, sendo geradas novas duas duplas hélices complementares e idênticas à original, fig. 2.6. Esta duplicação é chamada de semiconservativa porque cada uma das novas fitas do DNA tem uma fita nova e uma fita antiga, originária da molécula mãe (OLBY, 2003).

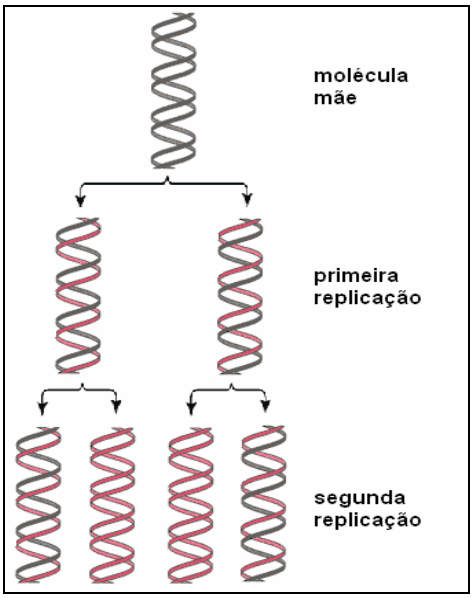


Figura 2.6: Replicação semiconservativa do DNA

2.4.2 Transcrição

O processo de transcrição torna possível a síntese de todos os RNAs que a célula necessita para seus processos, sendo os principais: o rRNA (ribossomal) principal componente dos ribossomos, o tRNA (transportador) que conduz os aminoácidos até os ribossomos para síntese protéica e o mRNA (mensageiro) que contém a informação genética do DNA para os ribossomos (WRANGE, 2005).

Durante a transcrição, a célula se encarrega do controle da expressão gênica, ou seja, por quantos genes transcrever e em que dado momento produzi-los, determinando a decisão de iniciar ou não esse processo.

As enzimas RNA polimerase atuam como elementos essenciais para a realização da transcrição assumindo um papel imprescindível uma vez que: reconhecem e ligam-se a seqüências específicas do DNA, desnaturam a molécula expondo a seqüência de pares de base a ser copiada, sustentam a dupla fita de DNA separada na região de síntese, renaturam a molécula na região imediatamente posterior à da síntese e terminam a síntese do RNA.

Três etapas são identificadas na transcrição: o início, quando ocorre a identificação de regiões específicas no DNA; o alongamento, quando os ribonucleotídeos são sucessivamente incorporados a cadeia; e a terminação, quando com ou sem participação de proteínas específicas, seqüências no DNA são identificadas e a síntese é encerrada.

Com base em regiões específicas do DNA, tem início a transcrição do mRNA quando, uma molécula de RNA polimerase liga-se a uma determinada localização específica da molécula de DNA, estabelecendo a leitura e o direcionamento sempre na orientação 5'-3'. As porções de DNA próximas à região que determina um gene contêm sinais que podem ser reconhecidos pela RNA polimerase e sinalizam o ponto de início da transcrição, estas regiões são chamadas de promotoras. O primeiro nucleotídeo transcrito, denominado sítio de início da transcrição (*Transcription Start Site*-TSS) é designado como +1 na molécula de DNA, os pares de base localizados à montante (*upstream*) do sítio de início recebem sinal negativo e números crescentes, enquanto que

os nucleotídeos à jusante (*downstream*) da posição +1 são marcados com números positivos crescentes, fig. 2.7.

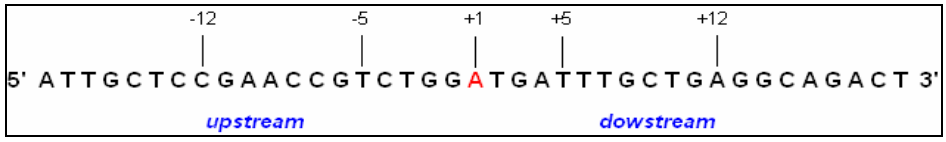


Figura 2.7: Demarcação de uma seqüência de DNA em relação ao sítio de início

Quando acoplada ao promotor, a RNA polimerase, forma um complexo que liga determinadas unidades, conhecidas como fatores sigma (σ), que especificam a afinidade da RNA polimerase à seqüência de DNA. Quando a transcrição é iniciada, no entanto a ligação desses fatores σ entre a cadeia de DNA e a RNA polimerase não permite a movimentação da RNA polimerase ao longo da fita de DNA, implementando um mecanismo de iniciação abortiva.

No momento em que ocorre a liberação por parte dos fatores σ da molécula de RNA polimerase, a dupla fita é desnaturada ocorrendo a separação da mesma o que permite o início da etapa de alongamento do mRNA, onde ocorre a incorporação de sucessivos ribonucleotídeos livres. A cada deslocamento da RNA polimerase ocorre a junção de um novo ribonucleotídeo à cadeia de mRNA que está sendo liberada pelo complexo de transcrição, ocorrendo sempre a desnaturação do par de base à frente desse complexo e a renaturação, ou seja, o pareamento da dupla fita ao final da região de abrangência da RNA polimerase. A fig. 2.8 apresenta a formação de uma cadeia de mRNA; essa figura mostra de forma esquemática para fim didático o funcionamento da transcrição desconsiderando o tamanho proporcional da RNA polimerase em relação à dupla fita, que se estende por aproximadamente 70 nucleotídeos dependendo do organismo.

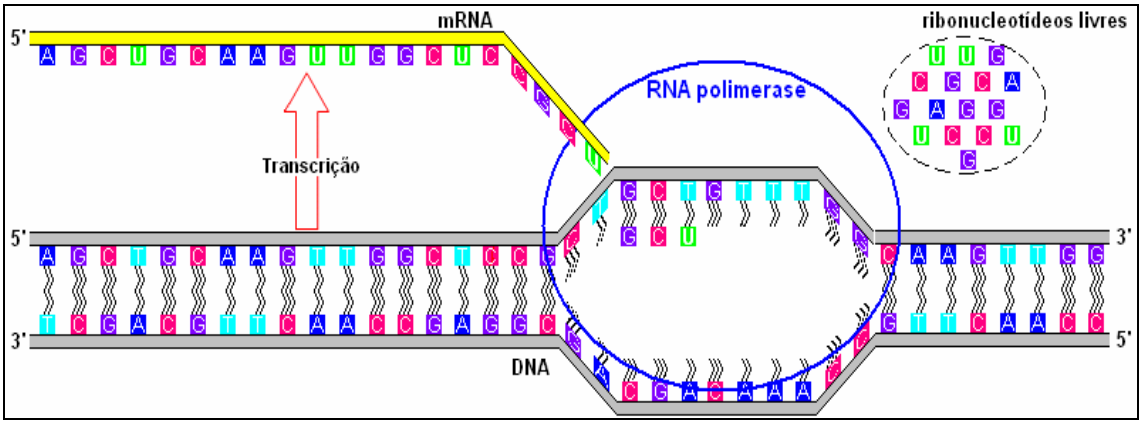


Figura 2.8: Atuação da RNA polimerase para a síntese de RNA

Embora possam ocorrer pausas momentâneas durante o ciclo de transcrição, seu término somente ocorre quando uma sinalização da seqüência de DNA indicar o fim do processo. Na maioria dos casos essa terminação tem a participação da cadeia de mRNA em formação que adquire uma forma especial e favorece o desligamento do complexo (WIESLANDER, 2005).

O processo descrito acima, relata o processo de transcrição nos organismos procariotos. Nos eucariotos, a transcrição segue os mesmos moldes se diferenciando pela presença de um número maior de elementos envolvidos e pelo fato de a transcrição

ocorrer no núcleo celular e posteriormente o mRNA formado ser conduzido ao citoplasma.

2.4.3 Tradução

A última fase do dogma central da biologia molecular para efetivar a transformação do DNA em proteína é a tradução do mRNA em uma seqüência protéica.

Uma vez constituídos os 3 principais RNAs, através da transcrição, é possível dar início e prosseguimento à construção de proteínas. Cada um dos diferentes mRNAs gerados armazena as instruções para compor proteínas específicas e funcionais. No entanto, por si só essa informação é inútil sem a presença dos mecanismos para sua interpretação. É como aqueles antigos cartões perfurados sem uma leitora para interpretar seu conteúdo (ALBERTS et al., 1996).

Desta forma, ocorre a participação dos outros dois RNAs transcritos que servem como ferramentas para a tradução desse código. O rRNA juntamente com outras proteínas participantes compõem os ribossomos, cuja função é gerenciar e organizar a receptividade dos aminoácidos que apresentam correspondência com o mRNA, assim como estabelecer a ligação peptídica (fig. 2.3) entre esses aminoácidos, fig 2.9A.

Após a transcrição, o RNA resultante é chamado de transcrito primário. Os transcritos primários de RNA mensageiro (mRNA), em procariotos, sofrem pouco ou nenhum processamento após sua síntese e, em geral, são traduzidos ainda durante a sua produção. Entretanto, nos eucariotos, o transcrito primário necessita de algumas alterações para adquirir maior estabilidade e caracterizar a molécula de RNA que irá para o citoplasma para ser traduzida. O RNA transportador (tRNA) e o RNA ribossômico (rRNA), ao contrário do mRNA procariótico, são gerados por quebras e outras alterações dos transcritos recém-sintetizados. De uma única cadeia nascente de RNA contendo regiões espaçadoras podem ser produzidas, por exemplo: três tipos de moléculas de rRNA e uma de tRNA, vários tipos de tRNA ou, até mesmo, várias cópias de um mesmo tRNA - sendo isso determinado pela seqüência do transcrito. Entre outras modificações pós-transcricionais possíveis, podemos citar a adição de nucleotídeos aos términos das cadeias de RNA e a alteração de bases e de unidades de ribose dos RNAs. Em procariotos é comum a adição da seqüência terminal CCA à ponta 3' de tRNAs e a metilação de algumas bases do rRNA, já nos eucariotos o que normalmente ocorre é a metilação da hidroxila 2' de uma a cada cem unidades de ribose do rRNA. Nos eucariotos, o processamento do transcrito primário que leva à formação do tRNA maduro inclui a clivagem da seqüência líder ou inicial 5'; o *splicing* ou processamento de *íntrons*; a substituição do terminal 3' UU por CCA e a modificação de várias bases. O processamento dos mRNAs em eucariotos inclui três etapas principais: a adição do *cap* 5´, o *splicing* e a adição da cauda de poliadenilato.

Por outro lado, o tRNA é o aparato portador de duas entidades primordiais da tradução: em uma extremidade do tRNA encontra-se um determinado aminoácido e na extremidade oposta há um *anticódon*, fig.2.9B, ou seja, um grupo de três bases complementares, que constitui um par com o *códon* do mRNA (DULBECCO, 1997).

Um aspecto relevante a ser observado é que o processo de expressão gênica (transcrição-tradução) envolve o RNA não somente como substrato essencial (no caso o mRNA), mas também como constituinte do aparato; os componentes rRNA e tRNA são codificados por genes e são gerados pelo processo de transcrição exatamente como o mRNA, mas com a diferença de que não existe um processo de tradução subseqüente.

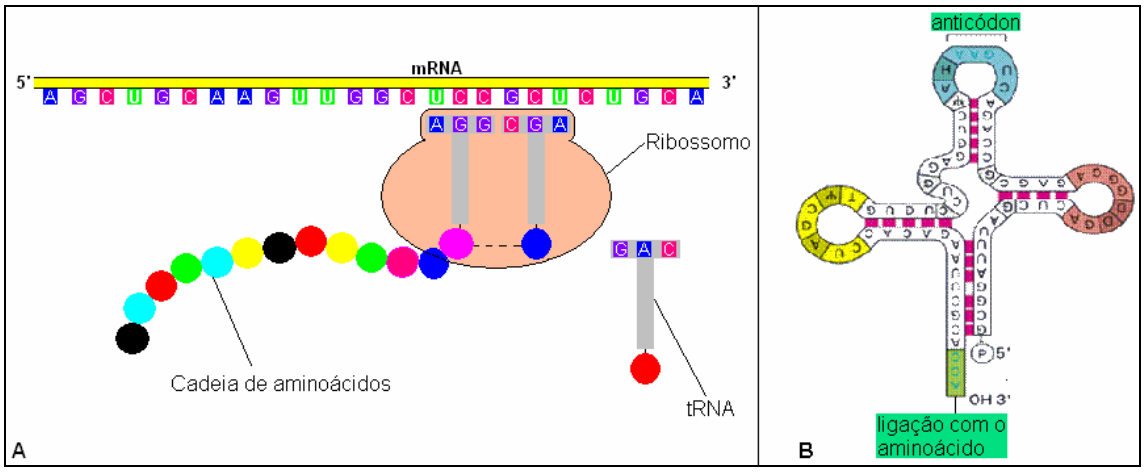


Figura 2.9: (A) Processo de transcrição, (B) Representação ampliada do tRNA

Todos os ribossomos estão presentes em meio celular e são idênticos em sua composição, realizando a síntese de diferentes proteínas associadas a diferentes mRNAs que fornecem as seqüências efetivamente codificadoras. De forma a complementar a informação anterior, cada ribossomo constitui o ambiente que controla o reconhecimento entre o *códon* do mRNA e o *anticódon* do tRNA. A síntese protéica tem origem desde o início de uma região codificadora até o seu final, interpretando o código genético como uma série de trincas adjacentes (NYGARD, 2005). Uma proteína é montada pela adição seqüencial de aminoácidos, na direção que vai da extremidade N-terminal para a C-terminal, à medida que o ribossomo se desloca ao longo do mRNA (LEWIN, 2001).

Um mesmo mRNA pode ter uma série de ribossomos acoplados traduzindo várias cópias da mesma proteína, fig. 2.10. A determinação de término de uma cadeia ocorre quando o ribossomo encontra um *códon* de terminação que não possui nenhum tRNA que se encaixe, sendo a cadeia polipeptídica liberada do complexo.

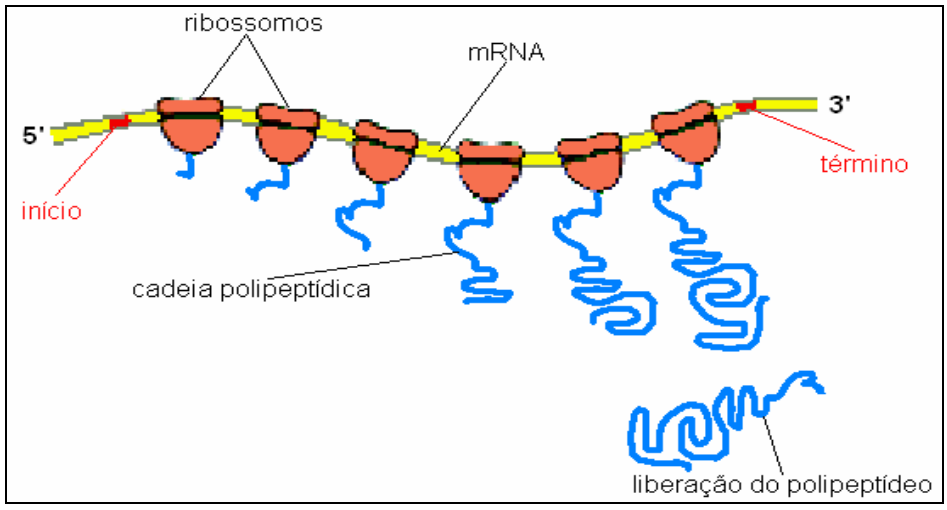


Figura 2.10: Vários ribossomos produzindo proteínas e liberação da cadeia polipeptídica

A nova cadeia fruto da associação dos diferentes RNAs é a seqüência de aminoácidos, os quais apresentam propriedades físico-químicas que ocasionam o dobramento da cadeia à medida que é traduzida gerando uma forma final enovelada que determina a sua funcionalidade.

2.5 Regiões Promotoras

As regiões promotoras desempenham um papel crucial na síntese de proteínas uma vez que são responsáveis por determinar a fronteira entre uma região do DNA que é transcrita em mRNA, do resto da informação que não é transcrita, para posteriormente ser traduzida em proteína.

Inicialmente, os estudos para identificação das regiões promotoras começaram na década de 70 no organismo *Escherichia coli* (por ser a bactéria usada como modelo para a descoberta de diversos mecanismos celulares e genéticos), com a realização de experimentos de mutação que indicavam sua ocorrência sempre na orientação 5'. Estudos posteriores, trataram da comparação da seqüência relativa ao início da transcrição tentando encontrar elementos comuns detectados pela RNA polimerase, ao final se obteve um levantamento mais consistente quando um número considerável de genes da *E. coli* foi comparado. Esta comparação revelou que a região a montante do ponto de início da transcrição apresentava dois conjuntos de seqüências similares na variedade de genes analisados, em comum os dois conjuntos eram compreendidos por 6 nucleotídeos cada e estavam localizados a 10 e 35 pares de base, respectivamente, anteriores ao ponto de início da transcrição e ficaram conhecidos como regiões -10 e -35 em relação ao ponto de início que é +1. As seqüências localizadas nestas posições não eram idênticas em todos os promotores avaliados, entretanto elas foram similares o suficiente para se estabelecer seqüências consenso[4] (ZAHA et al., 2003).

Conforme mostra a fig. 2.11, existem duas seqüências conservadas nos organismos procarióticos, que usam como padrão o organismo *E. coli*, sendo apresentada a região -10 com maior consenso constituída pelos seguintes elementos:

$$T_{80}\ A_{95}\ T_{45}\ A_{60}\ A_{50}\ T_{96},$$

onde os valores subscritos representam o percentual dos nucleotídeos mais freqüentes naquela posição. Neste exemplo se constata um intervalo de 45-96%, com o primeiro e segundo nucleotídeos (TA) e o último nucleotídeo (T) aparecendo com maior consenso. Enquanto que a região consenso para a posição -35 é dada pelo trecho:

$$T_{82}\ T_{84}\ G_{78}\ A_{65}\ C_{54}\ A_{45},$$

estando estas duas regiões separadas por ~17 pares de base (LEWIN, 2001).

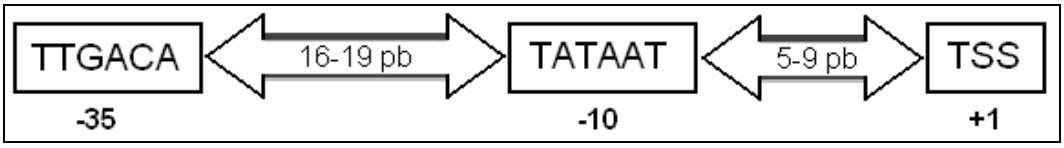


Figura 2.11: Esquema inicial de representação de promotores em procariotos

Pelo fato de a região -10 ser mais conservada a mesma pode receber duas outras denominações: *TATA-box* ou *Pribnow box*. Enquanto para a região -35 não há outra denominação.

Os experimentos de comparação de seqüências somente sugeriam posições muito conservadas que antecipavam os genes, não indicando sua funcionalidade.

[4] Seqüências que são mais comuns nas comparações entre outras seqüências, que apresentam maior freqüência.

Demais experimentos de mutagênese[5] e proteção[6] vieram demonstrar posteriormente tal funcionalidade (ZAHA et al., 2003).

Em estudo realizado em 2004, Browning e Busby relatam que é necessário se conhecer todas as partes e funcionalidades da RNA polimerase, as quais estão intimamente ligadas ao reconhecimento de regiões promotoras (BROWNING e BUSBY, 2004). Uma análise em alta resolução da RNA polimerase revelou em sua formação a presença de duas grandes subunidades β e β' que comportam o sítio ativo[7] e duas outras subunidades α com domínios independentes unidos por uma ligação flexível de ~20 aminoácidos: uma α com domínio amino-terminal (αNTD) grande, com a função de unir as subunidades β e β' e outra α com domínio carbono-terminal (αCTD) menor, que é um módulo de ligação do DNA com certos promotores

Conforme a fig. 2.12, que apresenta a interação da RNA polimerase com a dupla hélice de DNA, antes da RNA polimerase dar início à transcrição por um determinado promotor ela deve interagir com subunidades σ para formar um complexo. As subunidades σ apresentam 3 funções: garantem o reconhecimento de seqüências promotoras específicas, posicionam o complexo de transcrição sobre o promotor alvo e facilitam a separação da dupla fita de DNA no ponto de início da transcrição (TSS). Estas subunidades são proteínas de multi-domínios ligadas umas às outras, sendo os domínios 2, 3 e 4 envolvidos diretamente no reconhecimento de promotores.

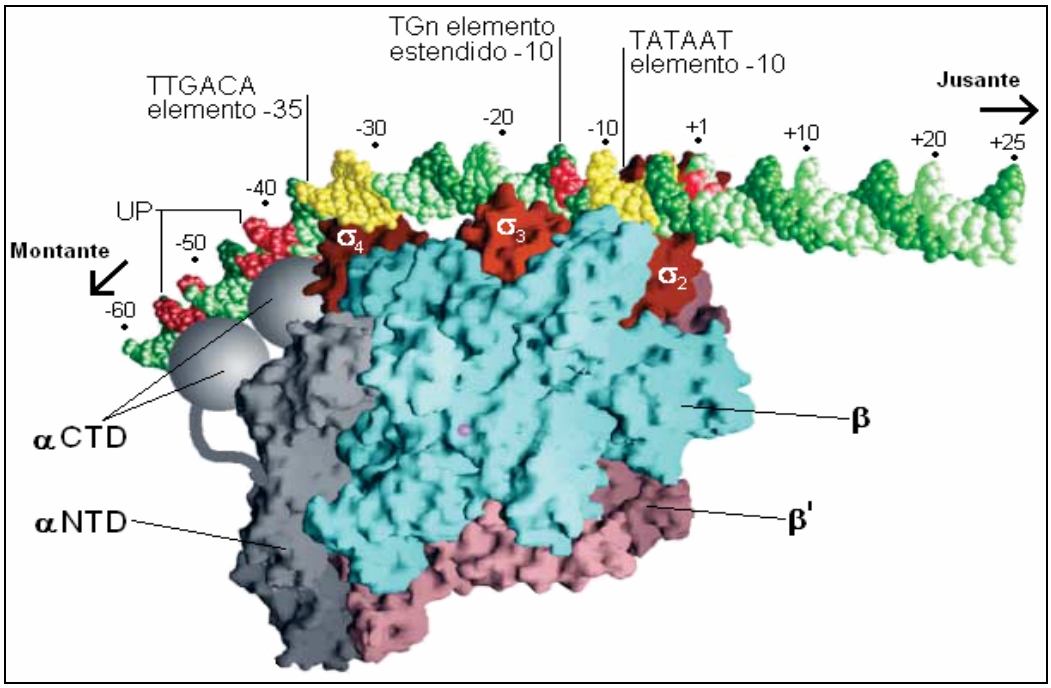


Figura 2.12: Interação da RNA polimerase com a dupla fita de DNA sobre a região promotora (MURAKAMI et al., 2002)

[5] Processo através do qual se originam as mutações; variação hereditária, súbita e espontânea, em um indivíduo geneticamente puro.

[6] Que usam moléculas que protegem os nucleotídeos de uma possível degradação.

[7] A região específica de uma enzima que se liga ao substrato e catalisa uma reação enzimática.

Com base nessa definição mais detalhada da RNA polimerase, quatro diferentes elementos foram identificados como fazendo parte do promotor na seqüência de DNA. Os dois principais elementos ainda são os hexámeros -10 e -35; o hexámero -10 é reconhecido pelo domínio 2 da subunidade σ e o hexámero -35 é reconhecido pelo domínio 4. Os dois elementos adicionais são: o elemento -10 estendido, que é um motivo[8] de 3-4 pares de base localizado imediatamente a montante da região -10 reconhecido pelo domínio 3 da subunidade σ; e o elemento UP, seqüência localizada a ~20 pares de base a montante da região -35 e reconhecida pelo αCTD da subunidade α. Entretanto, a contribuição relativa de cada elemento difere de promotor para promotor e não há garantia que a ocorrência destes quatro elementos é perfeita. Tanto a quantidade de RNAs polimerase quanto a distribuição de fatores sigma são limitados na célula, existindo uma intensa competição entre diferentes promotores pela RNA polimerase.

Na conclusão desse estudo foram determinados 5 mecanismos moleculares distintos que parecem garantir uma prudente distribuição da RNA polimerase entre os promotores concorrentes: a seqüência do DNA promotor, os fatores sigma, pequenos *ligands*[9], fatores de transcrição e a estrutura conformacional que assume o cromossomo bacterial. Cada um desses mecanismos atua no sentido de seguir as políticas adotadas pela célula que facilitam a ativação da transcrição ou que reprimem a atuação da RNA polimerase.

Basicamente, a função do promotor é mostrar exatamente para a RNA polimerase onde é o TSS. Caso este mecanismo não funcione corretamente pode ocorrer a síntese de proteínas erradas, comprometendo o funcionamento do organismo (FRIEDBERG, 2003). O maior problema na identificação dos promotores é que as regiões de inicialização são bastante complexas, devido aos vários fatores que concorrem para a região ser promotora.

No caso dos organismos eucarióticos, antes da RNA polimerase se conectar à região promotora é necessário que algumas proteínas denominadas fatores de transcrição se liguem a sub-regiões específicas dos promotores (sítios de ligação dos fatores de transcrição). Os promotores eucarióticos podem conter diferentes sub-regiões, entre elas: *TATA-box*, *CAAT-box*, *GC-box* (que são consideradas regiões de maior consenso), juntamente com outros sítios de ligação diferentes, o que resulta num grande número de combinações onde estas sub-regiões podem aparecer dentro da cadeia de DNA. O principal problema com as sub-regiões nos promotores eucarióticos é a grande variabilidade entre as mesmas, podendo aparecer em diferentes combinações. Suas localizações em relação ao TSS variam para diferentes promotores e nem todas essas sub-regiões específicas precisam existir para um determinado promotor (WERNER, 2002).

Indiferente ao tipo de organismo, o reconhecimento de regiões promotoras é realizado de forma automática pelos mecanismos apropriados presentes no interior da(s) célula(s). Por outro lado, a identificação dessas regiões por cadeias de caracteres e a aplicação de técnicas computacionais exigem bastante conhecimento da estrutura biológica dos diferentes organismos e ferramentas computacionais eficientes.

---

[8] Uma região conservada dentro de uma seqüência protéica.

[9] Qualquer átomo ou molécula anexada a um átomo central (BRITANICA.COM, 2005)

2.5.1 Promotores nos *Mycoplasmas*

2.5.1.1 Mycoplasmas

Os *Mycoplasmas* reúnem uma família de organismos que se caracterizam pelo tamanho mínimo da massa e ausência de parede celular, possuindo apenas uma membrana flexível, o que torna difícil sua identificação. Uma de suas características é que podem viver tanto dentro de células, sem matar a célula hospedeira, à semelhança do que fazem alguns vírus e bactérias, como também podem viver e crescer fora das células, nos fluídos corporais, coisa de que os vírus não são capazes. São responsáveis por doenças como a artrite reumatóide, inflamações alérgicas, pneumonia atípica e outras doenças, atuando como parasitas de humanos, mamíferos, peixes, répteis, artrópodes e plantas (RAZIN et al., 1998).

Existem mais de 100 espécies do gênero *Mycoplasma* reconhecidas, todas elas são membros da classe denominada *Molicutes*, bactérias caracterizadas pela ausência de parede celular, apresentam genomas de tamanho pequeno (0,6-1,35 Mega de pares de base) em relação a maioria dos genomas procarióticos e baixo conteúdo GC (18-40%) em seus genomas (HUTCHISON e MONTAGUE, 2002).

Segundo Maniloff (2002), várias questões relativas às vantagens de um genoma reduzido, ausência de parede celular e código genético alternado permanecem obscuras.

2.5.1.2 Promotores

Além dos fatores que tornam o reconhecimento de promotores na *E. coli* uma tarefa difícil, seu reconhecimento nos *Mycoplasmas*, que é foco deste trabalho, é um problema de complexa resolução. Alguns experimentos realizados com seqüências promotoras do organismo *Mycoplasma pneumoniae* revelaram que existem várias possíveis tendências para a região -10: TA(AGT)AAT, TAA(GT)AT, TACTAT e TATTAA; e um fraco consenso na região -35 apresentando uma curta seqüência TTGA, relativamente conservada (WEINER et al., 2000). Estudos anteriores Waldo et al. (1999), demonstraram que a pobre definição da região -35 ocorre devido à insuficiente quantidade de dados experimentais para identificar alguma conservação.

Uma característica que dificulta a identificação das regiões promotoras em grande parte dos organismos procarióticos é a presença de múltiplos fatores sigma. Entretanto o *M. pneumoniae*, com somente um fator sigma, apresenta uma surpreendente variabilidade (BROWNING e BUSBY, 2004). Isto pode refletir um processo mais complexo de início da transcrição, do que se esperava, diante da simplicidade da estrutura desse organismo. Os pesquisadores identificaram que características infreqüentes em outras espécies bacteriais parecem ser comuns no *M. pneumoniae* (WEINER et al., 2000).

Conforme estudo realizado por Ussery e Hallin (2004), em 152 genomas procarióticos, existe uma maior concentração de conteúdo AT nos 200 pb *upstream* do TSS do que nos 200 pb *downstream*. Apesar de ser encontrado um conteúdo AT de ~80% em regiões intergênicas e de ~70% em regiões codificantes do *M. hyopneumoniae* (VASCONCELOS et al., 2005), ainda não existe um consenso definido para as regões promotoras que são ricas em AT.

Em estudo promissor realizado por Eskin et al. (2002) e seu grupo, para identificar seqüências consenso em regiões intergênicas de 20 genomas de bactérias, incluindo os genomas de *E. coli*, *M. pneumoniae* e *M. genitalium*, foi concluído que é

muito difícil se derivar um bom consenso a partir dos pouco promotores mapeados. Em particular, os sinais ricos em AT nos *Mycoplasmas*, não representam nenhum dos promotores identificados no estudo experimental de Weiner et al. (2000).

3 REDES NEURAIS

Este capítulo aborda a computação neural, a qual surgiu inspirada no funcionamento de neurônios biológicos, descrevendo sua aplicabilidade à solução de problemas relacionados ao tema desta proposta. Um breve histórico é apresentado relatando o início das pesquisas na área, estendendo-se à consolidação das Redes Neurais (RNs) como uma metodologia adequada para o reconhecimento de promotores. No decorrer do capítulo são tratadas definições conceituais de: neurônios, funcionamento das redes, funções de ativação, formas de aprendizado e algumas arquiteturas. Ao final são expostas aplicações que fazem uso das RNs para o reconhecimento de promotores em procariotos, procurando fornecer ao leitor um panorama recente das abordagens a este problema.

3.1 Algumas Comparações: Cérebro vs. Computador

Um dos principais fatores que conduzem ao estudo das RNs é o fato de o cérebro humano realizar muitas tarefas, como: reconhecimento de imagens e voz, com um desempenho muito superior ao de um computador.

No entanto, uma vez comparadas as características do cérebro com os computadores digitais algumas diferenças merecem destaque.

O tempo de processamento de um neurônio animal é de aproximadamente 1ms, enquanto que um simples computador doméstico é capaz de trabalhar a 500 MHz, isto representa a execução de uma instrução a cada 2 ns. Desta forma, coloca-se a dúvida de como o cérebro é tão superior na tarefa de reconhecimento tendo em vista que a máquina possui um grande poder de processamento sobre a informação. Isto pode ser justificado pela característica de o cérebro possuir um processamento paralelo e distribuído inexistente nos computadores atuais. Outra característica relevante é que o cérebro humano possui um número estimado de 10^{11} a 10^{14} neurônios, cada um com cerca de 10^3 a 10^4 conexões, representando uma unidade completa de computação (KOVÁCS, 2002). De outro lado dispõem os computadores, de apenas uma unidade de processamento central.

Além disto, deve ser exposta a questão do armazenamento do conhecimento em ambos, em um o conhecimento é armazenado em um local endereçável, caso dos computadores, e no outro está armazenado sob uma forma dispersa e adaptativa. Outro tópico relativo ao armazenamento é a forma distribuída como o cérebro guarda as informações que torna possível regenerar um conhecimento global a partir de apenas uma de suas partes. Tudo isto faz do cérebro um sistema tolerante a falhas, uma vez que danificada uma de suas partes o conhecimento global não é afetado (MACKAY, 2002).

Sendo assim, inspiradas nas características de paralelismo e processamento altamente distribuído do cérebro, as Redes Neurais Artificiais foram aprimoradas e aplicadas à solução de problemas relativos ao reconhecimento de padrões e predição.

3.2 O que são as RNs ?

Segundo Swingler (1996), as RNs representam modelos estatísticos de sistemas do mundo real construídas para ajustar um conjunto de parâmetros. Os pesos, como são chamados esses parâmetros, descrevem um modelo representacional construído a partir de um conjunto de valores, conhecido como entradas, que é mapeado em um outro conjunto de valores: as saídas. O processo de ajuste dos pesos para obtenção de valores adequados à solução de determinado problema é designado treinamento e é obtido através da submissão de pares de entrada e saída ao modelo e ajuste dos pesos no sentido de minimizar um valor de erro, entre a saída produzida pela rede e a saída que se espera que o modelo obtenha. Com os pesos ajustados, o modelo é capaz de produzir respostas adequadas a entradas que não estavam presentes na etapa de treinamento.

Por mapear um conjunto de entradas em um conjunto de saídas geralmente de tamanho inferior, as redes neurais são amplamente empregadas em tarefas de classificação, onde o objetivo é determinar a que classe pertence determinado objeto apresentado à rede dentre classes já estabelecidas; ou na tarefa de predição de valores, em que ocorre a busca por uma aproximação quantitativa a um determinado valor alvo.

A maioria dos problemas que as RNs se propõem resolver é de natureza estática, embora nada impeça seu uso na solução de problemas que apresentam dependências temporais.

Apesar de serem normalmente implementadas em programas de computador, elas podem ser realizadas, quando representadas por um conjunto limitado de pesos, sob forma física em hardware.

Basicamente as RNs são sistemas paralelos e distribuídos compostos por unidades simples de processamento, chamadas nós, responsáveis por realizar determinadas funções matemáticas (geralmente não lineares). Os nós estão dispostos em uma ou mais camadas e interligados por um número significativo de conexões, quase sempre unidirecionais. A maioria dos modelos apresenta conexões associadas a pesos que armazenam o conhecimento representado no modelo e servem para ponderar a entrada recebida da rede.

As características de paralelismo e a forma como as redes estão representadas contribuem para a solução de problemas, superando o desempenho de modelos convencionais da computação, os quais geralmente são projetados a partir da definição de um conjunto de regras e equações de um sistema e como ele deve ser programado para obter a solução (SWINGLER, 1996).

As Redes Neurais Artificiais não atuam somente como mapeadores de funções de entrada e saída. A sua capacidade de aprendizado e generalização da informação, associada à capacidade da rede de aprender através de um conjunto reduzido de exemplos fornecendo respostas coerentes para dados não conhecidos previamente, tornam as RNs uma ferramenta computacional poderosa e interessante na solução de problemas complexos (BRAGA et al., 2000).

3.3 Inspiração Biológica

O neurônio biológico possui um corpo celular chamado de soma e diversas ramificações conhecidas como dendritos, responsáveis por conduzir sinais das extremidades para o corpo celular. Outra ramificação existente, geralmente única, é o axônio, que transmite um sinal do corpo celular para sua extremidade.

Generalizando o funcionamento das redes biológicas, os dendritos recebem as informações, ou melhor os impulsos nervosos provenientes de outros nós e os conduzem ao soma; no corpo celular a informação é processada, produzindo novos impulsos que são transmitidos através do axônio até os dendritos dos nós seguintes. O ponto de contato entre a extremidade axônica do neurônio e o dendrito de outro é chamado de sinapse; é devido às sinapses que os nós se unem, constituindo as redes neurais cerebrais (KONAR, 2000). A fig. 3.1 apresenta um modelo do neurônio biológico.

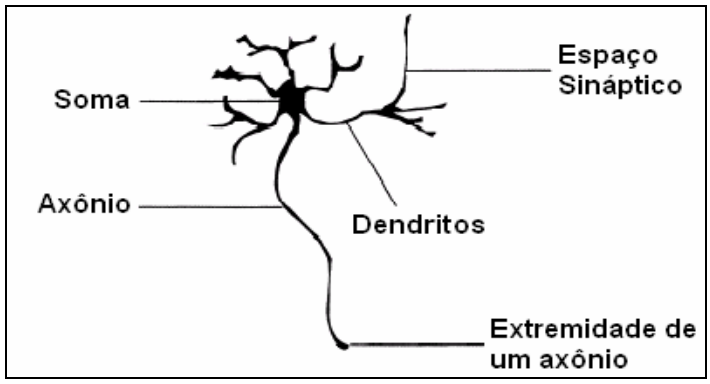


Figura 3.1: Neurônio biológico

3.3.1 Potencial de Ação

A ação da membrana dos nós é que cria a habilidade de produzir e transmitir os sinais, ela envolve o exterior do corpo do neurônio e tem a capacidade de gerar impulsos nervosos. O corpo é responsável por combinar os sinais recebidos, avaliando se o valor resultante está acima de um limiar de excitação do neurônio. Caso esteja, um impulso elétrico é produzido e propagado através do axônio para os nós seguintes (FREEMAN e SKAPURA, 1991).

A diferença de potencial (em *volts*) entre o interior e o exterior do neurônio, resultante da diferença de concentração de potássio (interna à célula) e sódio (externa à célula) é o que ocasiona o disparo do potencial de ação de um neurônio. A concentração de íons de potássio dentro da célula produz um potencial elétrico de –70 mv (potencial de repouso) em relação ao exterior; quando os impulsos das sinapses reduzem este nível para cerca de - 50 mv, o fluxo de sódio e potássio é invertido, tornando o interior da célula positivo em relação ao exterior. Isto faz com que o impulso nervoso seja transmitido pelo axônio até suas conexões sinápticas (BRAGA et al., 2000). Após o impulso chegar ao terminal de um axônio, canais controlados se abrem, permitindo a liberação de moléculas de vários tipos com o nome genérico de neurotransmissores, que se difundem no espaço entre o terminal do axônio e o dendrito de outro neurônio. Dependendo do tipo de neurotransmissores liberados, a sinapse poderá ser excitatória ou inibitória, estimulando ou restringindo determinada ação (FINE, 1999).

3.4 Histórico

O primeiro modelo de neurônio foi desenvolvido por McCulloch e Pitts (1943). Este modelo se caracterizou mais por descrever um modelo de um neurônio artificial e em apresentar suas capacidades computacionais, do que tratar técnicas de aprendizado.

Em 1949, Hebb, observando as mudanças nas sinapses dos neurônios, desenvolveu a "Teoria do Aprendizado Neural" onde determina que a conexão entre duas unidades ativadas ao mesmo tempo é reforçada (HEBB, 1949). A Regra de Hebb, como é conhecida a sua teoria na comunidade de RNs, foi interpretada do ponto de vista matemático, sendo utilizada atualmente em vários algoritmos de aprendizado. Outra regra de aprendizado, baseada no método do gradiente para a minimização do erro na saída de um neurônio com resposta linear, surgiu na década de 60, sendo elaborada por Widrow e Hoff e ficou conhecida como Regra Delta.

Frank Rosenblatt (1958) apresentou o seu modelo *Perceptron*, onde as RNs com nós MCP (originários de McCulloch e Pitts) poderiam ser treinadas para classificar certos tipos de padrões. O *Perceptron* mais simples descrito por Rosenblatt possui três camadas: a primeira recebe as entradas do exterior e possui conexões fixas, a segunda recebe os impulsos da primeira por meio das conexões, ocorrendo um ajuste nos pesos e por fim os valores são repassados à terceira camada produzindo uma resposta. Rosenblatt conseguiu provar a convergência de um algoritmo de aprendizado, como uma forma de aproximação dos pesos para valores adequados à solução de um problema específico. Sua descoberta incentivou muito as pesquisas, que no entanto foram contestadas por Minsky e Papert em 1969, no livro intitulado *Perceptrons*. Eles provaram que o teorema de Rosenblatt somente era válido para uma classe de problemas, que problemas elementares como a função XOR não eram capazes de ser computados pelo *Perceptron* e, portanto, se esta função não poderia ser solucionada também não poderia ser aplicada a Álgebra Booleana (MINSKY e PAPERT, 1969). Estes resultados e observações feitas por Minsky e Papert foram devastadores, deixando as pesquisas relativas às RNs em segundo plano durante a década de 70 até o início dos anos 80.

Em 1982, John Hopfield publicou um artigo que chamou a atenção para as propriedades associativas das RNs, o que incentivou as pesquisas na área. Hopfield (1982), mostrou a relação entre redes recorrentes auto-associativas e sistemas físicos. Mais tarde, em 1986, a impotência das redes *Perceptron* na solução do problema de associação de padrões para um conjunto de padrões não-lineares foi eliminada por Rumelhart, Hinton e Williams (1986), com a utilização da Regra Delta Generalizada. Eles redescobriram e divulgaram o algoritmo *Backpropagation*, chamando a atenção da comunidade de Inteligência Artificial (TVETER, 1998).

Um fator importante que forneceu interesse à área foi sem dúvida o avanço da tecnologia, sobretudo a microeletrônica, a qual vem permitindo a realização física de modelos de nós e sua interconexão de modo nunca antes proposto (BRAGA et al., 2000).

3.5 Neurônio Artificial

O neurônio é considerado a unidade básica de funcionamento da rede neural, na fig. 3.2 é apresentado um modelo para um neurônio artificial e a descrição de seu funcionamento corforme descrito em Haykin (2001).

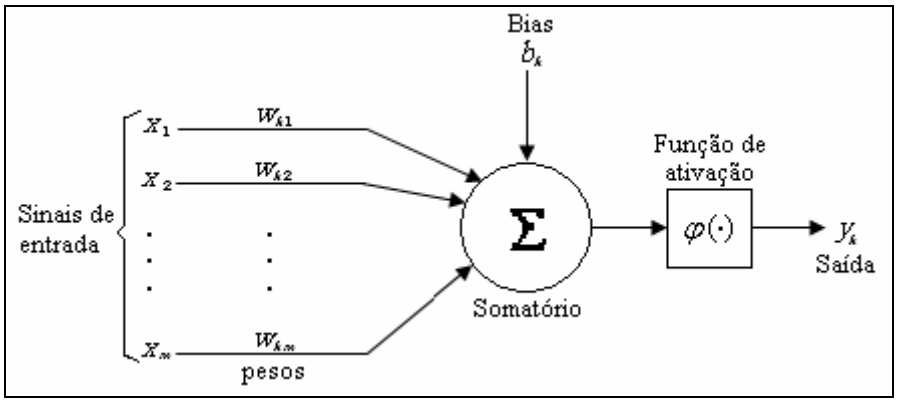


Figura 3.2: Modelo de um neurônio artificial

Neste modelo, são identificados três elementos básicos de funcionamento:

- Um conjunto de sinapses caracterizado por um peso próprio. Para cada neurônio k há uma entrada x_j conectada a ele que é multiplicada pelo peso sináptico w_{kj}. O peso sináptico geralmente está representado num intervalo de valores considerados pequenos podendo ser positivos ou negativos;
- Um somatório dos sinais de entrada, ponderados pelas respectivas sinapses do neurônio;
- Uma função de ativação utilizada para restringir a amplitude da saída do neurônio.Essa função representa uma normalização da amplitude da saída do neurônio a um valor finito.

O modelo ainda apresenta um bias, b_k, aplicado externamente. O papel deste bias é aumentar ou diminuir a entrada líquida da função de ativação, dependendo se ele fou positivo ou negativo.

Este modelo é descrito pelas equações 3.1 e 3.2.

$$net_k = \sum_{j=1}^{m} x_j w_{kj} \,, \tag{3.1}$$

$$y_k = \varphi\left(net_k + b_k\right) \tag{3.2}$$

onde x_1, x_2,..., x_m são os sinais de entrada; w_{k1}, w_{k2}, ..., w_{km} são os pesos do neurônio k; net_k é a saída do somatório das entradas multiplicadas pelos pesos; b_k é o bias; $\varphi(\cdot)$ é a função de ativação; y_k é o sinal de saída do neurônio.

3.6 Funções de Ativação

Diversas são as funções de ativação que podem ser aplicadas aos nós para produzir uma saída qualquer e não necessariamente zero ou um. A fig. 3.3 apresenta algumas funções utilizadas.

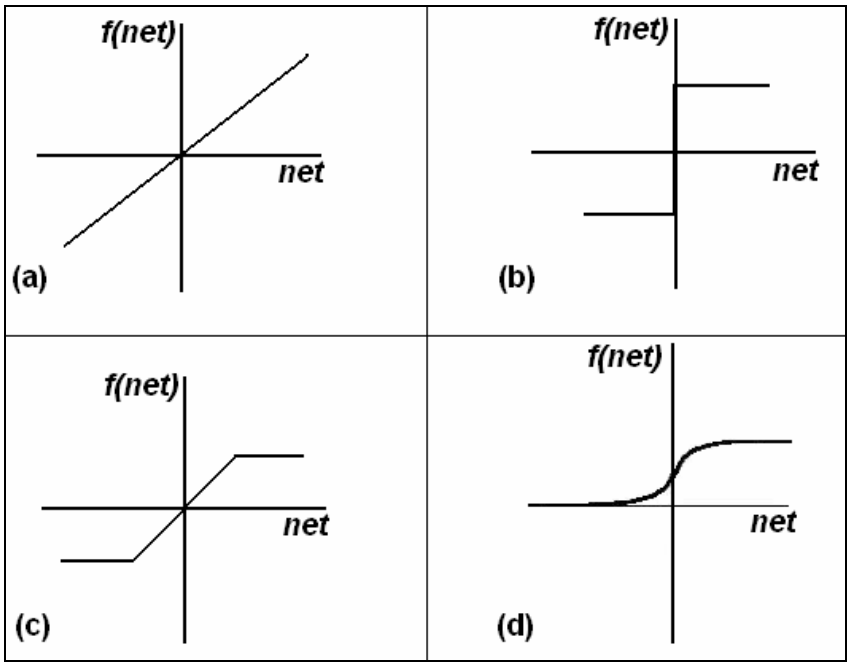


Figura 3.3: Funções de ativação

A função de ativação linear está representada na fig. 3.3(a) e é definida pela eq. 3.3, em que α um número real que define a saída linear, *y*, para os valores de entrada *x*.

$$y = \alpha net \tag{3.3}$$

A função degrau, fig. 3.3(b), produz uma saída $+\varphi$ para os valores de *x* maiores que zero e $-\varphi$ caso contrário. Está função está definida na eq. 3.4

$$y = \begin{cases} +\varphi & \text{se } net > 0 \\ -\varphi & \text{se } net \leq 0 \end{cases} \tag{3.4}$$

A função linear por partes, é uma derivação da função linear, que produz relações lineares em uma faixa restrita de $[-\varphi, +\varphi]$, como representada na eq. 3.5 e visualizada na fig. 3.3(c).

$$y = \begin{cases} +\varphi & \text{se } net \geq +\varphi \\ x, & \text{se } |net| < +\varphi \\ -\varphi & \text{se } net \leq -\varphi \end{cases} \tag{3.5}$$

Na fig. 3.3(d) é apresentada a função sigmóide, sob a forma logística limitada no intervalo [0,1], eq. 3.6, e que também pode assumir a forma de tangente hiperbólica o que estende seu intervalo para [-1,+1], eq. 3.7 (MACKAY, 1998).

$$y = \frac{1}{1 + e^{-net}} \tag{3.6}$$

$$y = \tanh(net) \tag{3.7}$$

3.7 Aprendizado

A aprendizagem das redes neurais é definida como o processo pelo qual os parâmetros da rede são ajustados, através de uma forma continuada de estímulo pelo ambiente no qual a mesma está operando, sendo o tipo de aprendizagem definido pela maneira particular como ocorrem os ajustes realizados nos parâmetros (MENDEL e McLAREN, 1970).

A idéia básica do aprendizado, indiferente da arquitetura da rede neural, é que a partir de um conjunto de pesos (geralmente pequenos valores randômicos), sejam aplicadas entradas para a rede, calculadas as saídas e observado o comportamento da mesma. Caso ela não desempenhe seu papel da maneira esperada os pesos são ajustados por algum algoritmo de aprendizado próprio da arquitetura da rede e o processo é repetido. Este processo é iterativo e realizado até que algum critério pré-estabelecido de parada seja obtido (WU e McLARTY, 2000), representando que a rede adquiriu conhecimento da situação proposta por meio de seus pesos.

O aprendizado pode ser definido como uma mudança de comportamento durante o procedimento de treinamento. Os métodos para treinamento estão divididos em duas abordagens que são vistas a seguir.

3.7.1 Aprendizado Supervisionado

A razão deste método possuir tal nome está ligada ao seu funcionamento, onde a entrada e a saída desejada para a rede são fornecidas por um "supervisor", que indica explicitamente um comportamento bom ou ruim, ou seja, seu papel é fornecer a resposta desejada, no sentido de encontrar uma ligação entre os pares de entrada e a saída fornecidos (BARRETO, 1997). O desempenho da rede é medido antes e depois do treinamento, sendo que a diferença na medida de desempenho será o fator responsável por indicar o quanto a rede aprendeu (LOESCH e SARI, 1996). A fig. 3.4 mostra o funcionamento do aprendizado supervisionado. Quando um padrão de entrada é apresentado para a rede, a saída desejada é comparada com a resposta calculada e os pesos sofrem um ajuste no sentido de minimizar uma medida de erro.

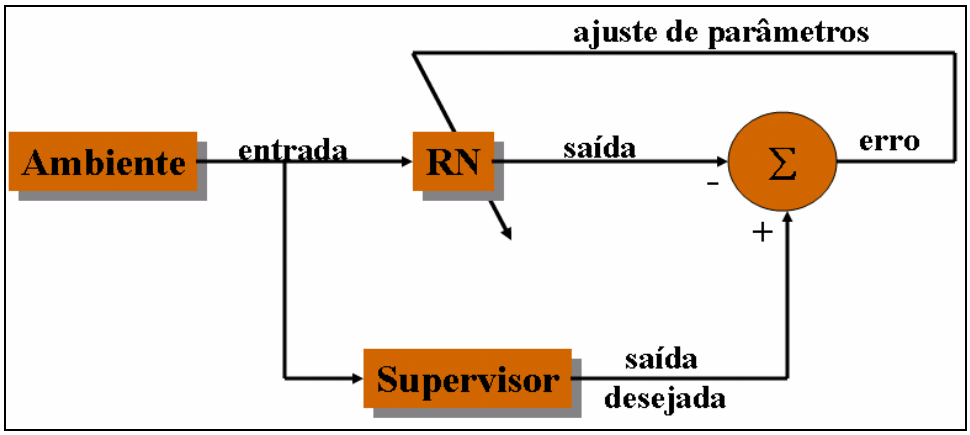


Figura 3.4: Aprendizado supervisionado

Na implementação deste tipo de aprendizado, o treinamento pode ser tratado sob duas formas: a *off-line*, em que uma vez treinados com os dados para resolução de determinado problema os parâmetros tornam-se fixos, desde que não sejam adicionados novos dados ao conjunto de treinamento, o que implicaria em um novo treinamento; e a

on-line, onde o conjunto de dados para treinamento muda continuamente, o que faz a rede estar sempre em processo de adaptação.

3.7.2 Aprendizado Não-Supervisionado

Este método é facilmente diferenciado do anterior pelo fato de não apresentar um "supervisor", conforme exposto na fig. 3.5. No aprendizado não-supervisionado não são usadas informações sobre se a resposta da rede foi correta ou não, no sentido de ajustar conexões. Mas sim, se a rede deve responder de modo semelhante a exemplos de amostras semelhantes, ou seja, no momento que a rede estabelece uma harmonia com regularidades estatísticas de entrada ela se torna capaz de formar representações internas para codificar características da entrada e produzir novas classes automaticamente (BARRETO, 1997). Com isto, conclui-se que este método somente é possível com a existência de dados redundantes na entrada da rede.

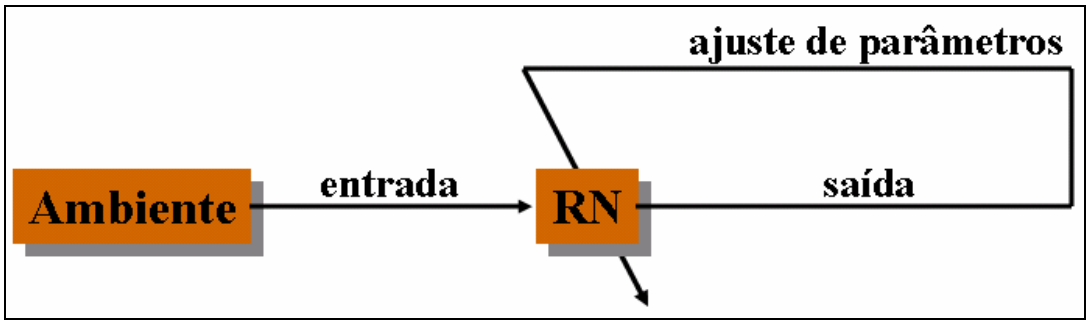


Figura 3.5: Aprendizado não-supervisionado

3.8 Definições de Arquiteturas

Existem diversas arquiteturas para as RNs e a definição da mais adequada está diretamente associada ao tipo de problema que a mesma será aplicada. Como no caso de redes com uma única camada de nós MCP, somente é possível a resolução de problemas linearmente separáveis. Para problemas não linearmente separáveis ou destinados à obtenção de agrupamentos outras arquiteturas são propostas. Nas subseções deste item são expostas as duas arquiteturas empregadas nos experimentos propostos. Alguns parâmetros que influenciam na definição da arquitetura são apresentados:

- **Número de camadas:**
  a) redes de camada única: existe somente um nó entre qualquer entrada e qualquer saída da rede;
  b) redes de múltiplas camadas: existe mais de um nó entre a camada de entrada e a camada de saída da rede.
- **Conexões dos nós:**
  a) *feedforward*, ou acíclica: a saída de um nó na *i*-ésima camada da rede não pode ser utilizada como entrada dos nós em camadas de índice menor ou igual a *i*;
  b) *feedback*, ou cíclica: a saída de algum nó na *i*-ésima camada da rede é utilizada como entradas dos nós de índice menor ou igual a *i*.

- **Conectividade da rede:**
  a) rede parcialmente conectada: nem todos os nós de uma camada para outra estão interligados;
  b) rede completamente conectada: todos os nós de uma camada para a seguinte estão conectados.

3.8.1 A Rede MLP

A arquitetura da *Multilayer Perceptron*, fig. 3.6, é baseada em um projeto hierárquico, formado por camadas compostas por vários elementos de processamento (neurônios). Tanto a quantidade de camadas como de neurônios que fazem parte da rede é determinada de acordo com a aplicação em que a rede é empregada. O número de elementos de processamento da primeira camada (entrada) está diretamente relacionado à dimensão do espaço de entrada, da mesma forma que a camada de saída está com o espaço de saída. Quanto à camada intermediária, ou às camadas intermediárias, também conhecida como escondida(s), a rede não possui uma regra fixa que estabeleça um número de neurônios, variando de forma empírica até a obtenção de resultados satisfatórios (SWINGLER, 1996).

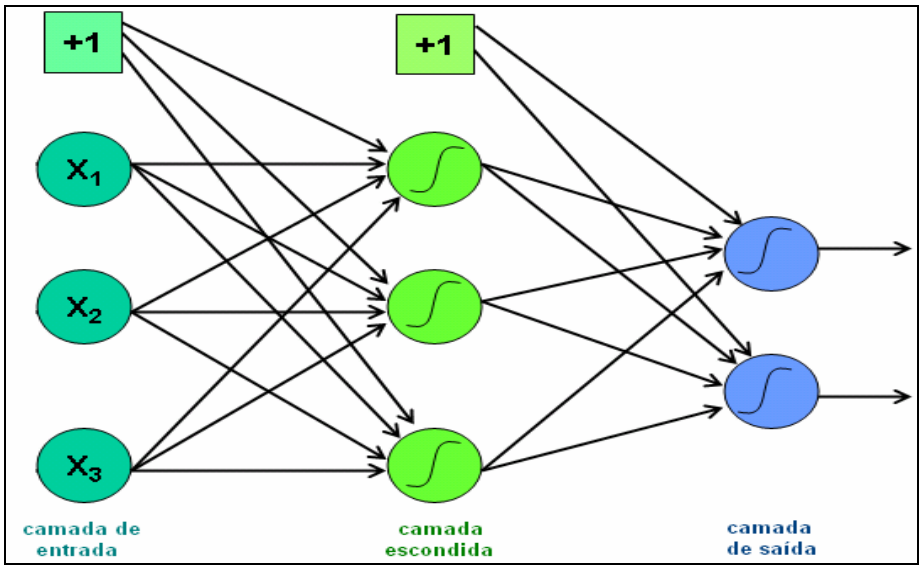


Figura 3.6: Arquitetura do MLP

Segundo o Teorema de Kolmogorov, o objetivo da rede é mapear uma função $\mathbf{y}=f(\mathbf{x})$ de R^m em R^n, através de um conjunto de treinamento de k pares de vetores $(\mathbf{x}_1,\mathbf{o}_1),\ldots,(\mathbf{x}_i,\mathbf{o}_i),\ldots,(\mathbf{x}_k,\mathbf{o}_k)$, onde para cada vetor de entrada $\mathbf{x}_i$ de m elementos, existe um vetor de saída correspondente $\mathbf{o}_i$ de n elementos (HECTH-NIELSEN, 1990).

O funcionamento da MLP se baseia na apresentação à rede de um padrão de entrada ($\mathbf{x}_i$), que é propagado pela(s) camada(s) intermediária(s) até chegar à camada de saída, onde a rede apresentará a sua resposta ($\mathbf{y}_i$), correspondente ao estímulo. O padrão de saída calculado pela rede ($\mathbf{y}_i$) é comparado ao padrão de saída desejado ($\mathbf{o}_i$), sendo então calculado o erro para cada unidade de saída (BARRETO, 1997). A função de erro que deve ser minimizada é apresentada na eq. 3.8, e é conhecida como Erro Médio Quadrado (EMQ).

$$E = \frac{1}{2}\sum_{p=1}^{M}\sum_{i=1}^{N}\left(o_{p,i} - y_{p,i}\right)^2, \tag{3.8}$$

onde:

M = número de padrões ou vetores de entrada;

N = número de neurônios na saída ou dimensão do vetor de saída;

$o_{p,i}$ = saída desejada do *i-ésimo* neurônio, para o *p-ésimo* padrão apresentado;

$y_{p,i}$ = saída obtida pela rede do *i-ésimo* neurônio, para o *p-ésimo* padrão apresentado.

A partir da função de erro são calculados recursivamente termos do gradiente do erro (gradientes locais ou δs) que representam a participação de cada unidade no erro da rede, obtidos segundo a eq. 3.9 conforme a camada da rede.

$$\delta_j^{(l)}(n) = \begin{cases} e_j^{(L)}(n)\varphi_j'(v_j^{(L)}(n)) & \text{para o neurônio } j \text{ da camada de saída } L \\ \varphi_j'(v_j^{(l)}(n))' \sum_k \delta_k^{(l+1)}(n) w_{kj}^{(l+1)}(n) & \text{para o neurônio } j \text{ da camada oculta } l \end{cases} \tag{3.9}$$

onde e representa o sinal de erro para o j-ésimo elemento para a camada de saída L, φ' representa a derivada, e v representa o *net* (eq. 3.1) sob influência do bias.

O gradiente local calculado é retropropagado para os neurônios da camada intermediária imediatamente anterior à camada de saída. O valor recebido por cada unidade é proporcional à sua contribuição para o erro total. Este valor é diretamente influenciado pelo peso (w_{ij}) associado à conexão entre o elemento da camada i e o da camada de saída j; a regra de aprendizado, eq. 3.10, atualiza os pesos das conexões por meio do erro produzido (LUGER, 2004).

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha\left[w_{ji}^{(l)}(n-1)\right] + \eta\delta_j^{(l)}(n) y_i^{(l-1)}(n) \tag{3.10}$$

onde η é o parâmetro da taxa de aprendizado e α é a constante de momento.

A regra de aprendizado da rede MLP utilizando o algoritmo de *Backpropagation* busca obter pesos que permitam a realização de mapeamentos não-lineares entre os padrões de entrada e saída, e é conhecida como Regra Delta Generalizada, em virtude de ser uma extensão da Regra Delta (TVETER, 1998).

A Regra Delta foi criada por Widrow e Hoff, que a aplicaram a um algoritmo adaptativo, *Least Mean Square*-LMS, destinado a redes neurais de apenas duas camadas com unidades de saída lineares (BARRETO, 1997). O objetivo principal desta regra era ajustar os pesos no sentido de minimizar o erro médio quadrático total.

A limitação no uso da Regra Delta sobre redes com mais de duas camadas surgiu devido ao fato de esta ser baseada na diferença entre os valores reais e os desejados para o ajuste dos pesos. Como em redes com mais de duas camadas os valores desejados pelos neurônios da camada intermediária são desconhecidos, torna-se impraticável o cálculo da diferença entre a saída real e a desejada. Em busca da superação dessa limitação é que foram intensificados os estudos sobre redes com uma ou mais camadas, surgindo assim a Regra Delta Generalizada (FREEMAN e SKAPURA, 1991), a qual utiliza o método de descida do gradiente em busca da minimização do erro.

A convergência é alcançada quando a rede atinge um mínimo local. Embora a Regra Delta Generalizada tenha superado certas limitações apresentadas por outros métodos aplicados para o mesmo fim, é possível ainda a sua permanência em mínimos locais, provocando oscilações nos pesos como também o abandono do processo de treinamento da rede (SIMPSON, 1990).

No sentido de evitar que a rede, durante o processo de treinamento, permaneça em mínimos locais; alguns fatores merecem uma atenção especial, como: taxa de aprendizado, número de neurônios na camada intermediária, número de camadas intermediárias e o número de padrões utilizados no treinamento. A aplicação das RNs está muito relacionada à heurística, onde a experimentação ainda é um bom fator para a determinação dos parâmetros para o treinamento da rede.

A definição básica do algoritmo *Backpropagation* para treinamento incremental da rede, adaptado de Haykin (2001), é descrita nos seguintes passos:

1) Inicialização dos pesos, bias, taxa de aprendizado, momento e definição dos critérios de parada;
2) Apresentação para a rede dos exemplos de treinamento. Apresente uma época[10] de treinamento à rede. Para cada exemplo do conjunto realize a série de computações para frente e para trás descritas nos passos 3 e 4;
3) Propagação: suponha que um exemplo de treinamento da época seja representado por ($\mathbf{x}(n)$, $\mathbf{o}(n)$), com o vetor $\mathbf{x}(n)$ aplicado a camada de entrada de nós sensoriais e o vetor de resposta desejada $\mathbf{o}(n)$ apresentado à camada de saída de nós computacionais. Calcule as saídas locais induzidas (*nets*) e os sinais funcionais da rede prosseguindo para frente através da rede, camada por camada;
4) Retropropagação: calcule os gradientes locais segundo a eq. 3.9. Ajuste os pesos sinápticos da rede na(s) camada(s) oculta(s) de acordo com a regra delta generalizada apresentada na eq. 3.10;
5) Iteração: itere as computações para frente e para trás dos passos 3 e 4, apresentando novas épocas de exemplos de treinamento para a rede, até que seja satisfeto o critério de parada.

É importante salientar que a ordem de apresentação dos exemplos de treinamento deve ser aleatória, de época para época. Os parâmetros de momento e taxa de aprendizado tipicamente são ajustados quando o número de iterações de treinamento aumenta.

A parada do algoritmo *Backpropagation* é estabelecida quando ocorrer ou um decréscimo do erro a níveis fixados, ou a execução de um determinado ciclo de treinamento iterativo ou mesmo quando a rede atingir um estado em que todos os padrões de treinamento estejam classificados corretamente (FREEMAN e SKAPURA, 1991).

Nas duas próximas subseções são brevemente descritos dois algoritmos para treinamento supervisionado e que foram empregados na realização de experimentos.

---

[10] Uma época de treinamento se refere à passagem de todo o conjunto reservado para treinamento uma vez pela rede.

3.8.1.1 Gradiente Descendente com Termo de Momento (GDX) e Taxa de Aprendizado Adaptativa

A implementação do aprendizado de retropropagação (*Backpropagation*) atualiza os pesos da rede e bias na direção em que a função de custo decresce rapidamente; isto é, o negativo do gradiente e é representado como:

$$\mathbf{v}_{k+1} = \mathbf{v}_k - \alpha_k \mathbf{g}_k, \tag{3.11}$$

onde $\mathbf{v}_k$ é o vetor com os pesos e bias atuais, $\mathbf{g}_k$ é o gradiente atual da função de custo e α_k é a taxa de aprendizado. Esta técnica é conhecida como "*steepest (gradient) descent*". A mudança nos pesos e no bias é obtida pela multiplicação da taxa de aprendizado pelo gradiente negativo. Quanto maior for a taxa de aprendizado maior é o passo no processo de descida (minimização). Se a taxa de aprendizado for muito grande o algoritmo fica instável. Se a taxa de aprendizado for muito pequena a convergência demora a ocorrer (ABDUL-KAREEM et al., 2001). Desta forma, quando a solução está próxima do ótimo o erro tende a ser próximo a zero. Assim, já que a evolução da solução depende do erro esta tende a ser muito reduzida. Uma forma de tentar evitar este problema consiste em definir um α_k adaptativo.

O termo de momento possui como principal função acelerar o aprendizado sem produzir oscilações. A probabilidade de a convergência esbarrar em mínimos locais é reduzida, uma vez que o termo ignora as variações de alta freqüência na superfície de erro. A inclusão do momento, baseia-se em fazer com que as mudanças nos pesos das conexões sejam somadas a uma fração da última alteração nestes pesos, determinada pela regra de aprendizagem. Assim, se a alteração anterior foi realizada num determinado sentido da superfície de erro, parte da atualização atual nos pesos das conexões será realizada no mesmo sentido (VALIATI, 2000).

3.8.1.2 Resilient Back-propagation (RPROP)

As redes neurais multicamadas normalmente utilizam funções de transferência sigmóides em suas camadas ocultas. Estas funções caracterizam-se pelo fato de sua inclinação ser muito próxima a zero em pontos afastados da origem. Isto causa um problema quando se emprega o algoritmo de treinamento "*steepest (gradient) descent*" com função sigmóide, onde o gradiente possui uma magnitude muito pequena o que acarreta pequenas mudanças nos pesos e bias.

O *Resilient Back-propagation* (RPROP) é um método que elimina o efeito desta inclinação pequena da sigmóide nos pontos mais afastados da origem, levando em consideração somente o sinal da derivada para a atualização dos pesos. O tamanho desta mudança de pesos é determinado por uma constante de atualização e incrementado, sempre que uma atualização não alterar o sinal do gradiente, ou decrementado caso contrário. Se a derivada é zero o valor de atualização permanece o mesmo. Tal método fornece uma convergência rápida, apesar de apresentar alguma oscilação em torno do mínimo ao final do processo.

3.8.2 Teoria da Ressonância Adaptativa

A rede neural ART (*Adaptive Resonance Theory*), proposta por Carpenter e Grossberg, representa um sistema que auto-organiza padrões de entrada em categorias de reconhecimento. O principal objetivo desta rede é solucionar o dilema de plasticidade e estabilidade, ou seja, como um sistema adaptativo pode permanecer flexível, quando padrões estranhos estimulam a rede criando novas categorias de

reconhecimento, e ainda assim continuar estável, quando agrupa padrões similares na mesma categoria de reconhecimento. A solução para este dilema foi a implantação de um mecanismo de realimentação entre uma camada competitiva e a camada de entrada, permitindo o aprendizado de uma nova informação relevante sem perda do conhecimento já adquirido (CARPENTER e GROSSBERG, 1987).

Com o passar dos anos a família ART se expandiu sendo composta pelos modelos:

ART 1: processa apenas padrões binários;

ART 2: processa padrões contínuos;

ART 3: processa padrões contínuos e aborda a ação de neurotransmissores nos mecanismos de sinapse;

Fuzzy ART: trata os conceitos nebulosos na arquitetura *ART 1*, permitindo o tratamento de padrões analógicos;

ARTMAP: apresenta uma arquitetura preditiva, composta por dois módulos ART;

Fuzzy ARTMAP: apresenta a arquitetura preditiva do *ARTMAP* utilizando conceitos nebulosos.

Neste trabalho é abordado somente o modelo conexionista *ART 1*, por ser uma das técnicas empregadas nos experimentos propostos no próximo capítulo.

3.8.2.1 ART 1

Este foi o modelo primordial da Teoria da Ressonância Adaptativa, baseado no uso de entradas binárias e na aplicação da tradicional teoria dos conjuntos, com o operador de intersecção ($\cap$) para a escolha de categorias e no processo de aprendizado.

Na escolha de categorias são calculados escores de casamento entre os padrões que entram na rede e avaliada uma medida de similaridade entre tais padrões, determinando se o padrão corrente é pertencente ou não a uma determinada categoria já existente, ou se ele deve dar origem a uma nova categoria. No cálculo do casamento entre padrões ocorrem dois processos: um conhecido como *bottom-up*, que diz respeito às conexões alimentadas adiante e que fazem o papel de filtro adaptativo, ampliando os contrastes de um padrão que entra na rede; e outro processo chamado de *top-down*, que é responsável pela realimentação da rede e visa encontrar similaridades entre as categorias já identificadas e os novos padrões. Como pode ser visto na figura abaixo, o processo *bottom-up* trata dos pesos de baixo para cima, enquanto que o processo *top-down* realiza a tarefa inversa, administrando os pesos de cima para baixo.

Conforme a fig. 3.7, a rede padrão ART é composta por campos de atividade dos vetores, incluindo: um campo F_0 de nós que representam o vetor de entrada corrente, um campo F_1 que recebe ambas as entradas *bottom-up* de F_0 e a entrada *top-down* do campo F_2, que representa o código de atividade ou categoria. O vetor ativo em F_0 é representado por $\mathbf{I}=(I_1,...,I_M)$, e cada componente I_i possui um valor no intervalo [0,1] com o índice $i=1,...,M$. O vetor de atividade em F_1 é representado por $\mathbf{x}=(x_1,...,x_M)$ e o vetor de atividade em F_2 é representado por $\mathbf{y}=(y_1,...,y_N)$. O número de nós de cada campo é arbitrário.

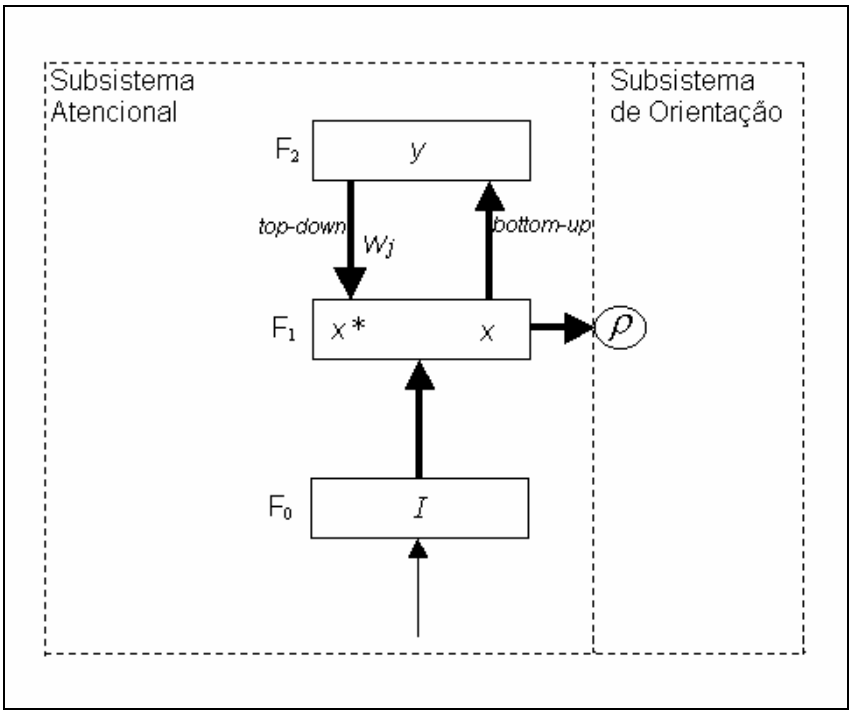


Figura 3.7: Arquitetura padrão da ART

Existem dois subsistemas que atuam em um módulo ART para processar os padrões de entrada:

- Subsistema Atencional: realiza o processamento dos padrões de entrada familiares, determinando respostas e representações internas mais precisas de tais padrões;
- Subsistema de Orientação: atua sobre os padrões estranhos, inibindo o subsistema atencional quando estes padrões são apresentados à rede (GUAZZELLI, 1993).

Na categoria F_2 existem nós j ($j=1,...,N$) que estão associados a um vetor de pesos adaptativos $\mathbf{w}_j \equiv (w_{j1},...,w_{jM})$, que inicialmente estão configurados para o valor 1, não possuindo nenhuma categoria comprometida. Quando uma categoria é selecionada ela se torna comprometida e o peso é atualizado, decrescendo monotonicamente através do tempo, convergindo para um limite.

O funcionamento do modelo ART ocorre da seguinte maneira:

1) Quando um padrão de entrada estimula a rede ele se encontra na camada F_0 e recebe a denominação de **I**;
2) Este padrão **I** é então repassado para a camada F_1 que pode executar algum tipo de normalização desse padrão **I**, recebendo o rótulo de **x**. Portanto, **x** é o padrão **I** normalizado;
3) Através das conexões *bottom-up* o padrão **x** tem seus contrastes ampliados, quando propagado pelas conexões entre as camadas F_1 e F_2 (*Long-Term Memory - LTM*), produzindo na camada F_2 o padrão **y**, resultante do processo *WTA* (*Winner Take All*), onde o neurônio com maior ativação recebe o valor 1 e aos demais neurônios é atribuído o valor zero;

4) Obtido o padrão **y** em F_2 as conexões *top-down* atuam no sentido de produzir a realimentação em F_1, gerando um padrão protótipo, **x***, nesta camada o qual determinará o grau de similaridade entre o padrão corrente e o protótipo produzido. Desta forma é definido, por meio de uma regra que avalie **x*** e **I**, se o padrão de entrada será estabilizado, ou seja, pertencerá ao neurônio vencedor, ou se outro neurônio, que representará as características deste novo padrão, deverá ser utilizado.

3.8.2.2 Aprendizado em ART 1

A rede ART 1 possui um processo de aprendizagem rápida que possibilita que padrões de entrada esporádicos e importantes sofram uma adaptação instantânea ao sistema. A aprendizagem rápida está associada a uma taxa de esquecimento (enfraquecimento da conexão representativa de alguma característica), a qual é conhecida como *fast commit slow recode* (CARPENTER et al., 1992).

Existem três parâmetros que guiam o aprendizado da rede ART 1: o parâmetro de escolha $\alpha > 0$; o parâmetro de vigilância $\rho \in [0,1]$ e a taxa de aprendizado $\beta \in [0,1]$.

Quando o valor de β é igual a 1 significa que a aprendizagem rápida está habilitada. No entanto, se a propriedade *fast commit slow recode* for desejada para neurônios que não estejam comprometidos por nenhuma categoria, o valor de β deve ser reduzido ($\beta < 1$) depois da primeira adaptação daquele neurônio (GUAZZELLI, 1994).

Os pesos das conexões entre as camadas F_1 e F_2 são inicializados com o valor 1 e representados pelo vetor $\mathbf{w}_j$, conforme:

$$w_{j1} = w_{j2} = \ldots = w_{jM} \,, \tag{3.12}$$

onde M indica o número de neurônios em F_1 e N o número de neurônios em F_2 com o valor de j variando de 1 até N.

Uma categoria j é escolhida em F_2, por um determinado padrão de entrada, segundo a eq. 3.13 que representa a função de escolha T_j

$$T_j(\mathbf{I}) = \frac{\left|\mathbf{I} \cap \mathbf{w}_j\right|}{\alpha + \left|\mathbf{w}_j\right|}, \tag{3.13}$$

onde o operador de intersecção ($\cap$) é definido por:

$$(\mathbf{x} \cap \mathbf{y})_i \equiv min(x_i, y_i) \tag{3.14}$$

e a norma | . | é definida por:

$$\left|\mathbf{x}\right| = \sum_{i=1}^{M} x_i \;. \tag{3.15}$$

O neurônio J com maior ativação de entrada, pelo processo *WTA* (eq. 3.16) será o escolhido:

$$T_J = max\{T_j : j = 1,\ldots,N\} \,, \tag{3.16}$$

onde J indica o índice do neurônio vencedor. Caso mais de um neurônio obtenha a ativação máxima, a regra determina que o nó escolhido deve ser aquele que apresentar o menor índice j.

Carpenter et al. (1991) descrevem que o sistema pode estar dentro de um *limite conservativo*, em que o valor de α, utilizado na escolha da função da categoria *j* em F_2 (eq. 3.13), deve receber um valor muito próximo a zero. Isto faz com que o padrão de entrada tente ativar uma determinada categoria, já existente, que represente nos seus respectivos pesos um subconjunto do padrão de entrada, evitando a modificação dos pesos da *LTM*, o que dá origem a expressão *limite conservativo*.

A ressonância que dá origem à aprendizagem ocorre quando a proposição da eq. 3.17 for verdadeira, indicando que uma categoria estável foi encontrada.

$$\frac{\left|\mathbf{I} \cap \mathbf{w}_j\right|}{\left|\mathbf{I}\right|} \geq \rho \tag{3.17}$$

Assim, a rede pode realizar a adaptação do padrão de entrada, alterando o valor das conexões da categoria ativada por:

$$\mathbf{w}_J(t+1) = \beta(\mathbf{I} \cap \mathbf{w}_J(t)) + (1-\beta)\mathbf{w}_J(t). \tag{3.18}$$

Entretanto, se a proposição da equação 3.17 for falsa o subsistema de orientação gera um sinal inibitório de *reset*, inibindo o neurônio *J* durante a apresentação do padrão de entrada corrente, para que este não seja novamente selecionado, fazendo com que outro neurônio em F_2 seja escolhido.

3.9 Estado da Arte - Redes Neurais no Reconhecimento de Promotores

Esta subseção expõe experimentos que utilizaram a métodos de RNs na busca de soluções ao problema de reconhecimento de regiões promotoras. Dentre os motivos que conduziram à escolha desses trabalhos estão: o fato de serem artigos muito referenciados; os dados são todos relativos a organismos procariotos, mesmo reino do organismo tratado nessa proposta; e por utilizarem RNs, ou a associação de RNs a outras estratégias no sentido de obterem melhores resultados. Para cada trabalho relatado são abordados: os dados empregados, a técnica de codificação dos dados para apresentação à rede neural, características da rede implementada e os resultados obtidos para cada experimento relatado.

3.9.1 Métricas de Avaliação dos Resultados

Antes de expor as aplicações que utilizam redes neurais na solução do problema de reconhecimento de regiões promotoras em procariotos, é importante apresentar as métricas estatísticas de avaliação dos resultados empregadas pela maioria dos experimentos relatados, as quais são também as métricas empregadas para avaliação dos experimentos descritos no próximo capítulo. Uma delas é o coeficiente de correlação (*CC*) (MATHEWS, 1975), eq. 3.19, que fornece uma estimativa dos resultados preditos corretamente pela rede neural em relação aos preditos incorretamente. As outras duas são: a sensibilidade (*SN*), probabilidade do modelo de predição reconhecer corretamente amostras positivas, eq. 3.20; e a especificidade (*SP*), eq. 3.21, probabilidade de identificar corretamente amostras negativas (BAJIĆ, 2000)

$$CC = \frac{(TP \times TN) - (FN \times FP)}{\sqrt{(TP + FN) \times (TN + FP) \times (TP + FP) \times (TN + FN)}} \tag{3.19}$$

$$SN = \frac{TP}{TP + FN} \tag{3.20}$$

$$SP = \frac{TN}{TN + FP} \tag{3.21}$$

Onde:

TP= verdadeiros positivos;

TN= verdadeiros negativos;

FP=falsos positivos;

FN=falsos negativos.

Quanto mais próximo ao valor 1 for o coeficiente de correlação melhor será a capacidade da rede em classificar a informação que lhe foi submetida. Assim como, os melhores resultados para sensibilidade e especificidade são valores próximos a 1.

3.9.2 Redes Neurais Bayesianas no Reconhecimento de Promotores

Este experimento proposto por Wang e Ma (1999), apresenta em conjunto dois níveis de classificadores para reconhecer seqüências de promotores da *E.coli*. Em um primeiro momento, os classificadores compostos por três Redes Neurais Bayesianas (*Bayesian Neural Networks*-BNN) são treinados por um conjunto de três características representativas. Numa segunda etapa ocorre a combinação da saída destes três classificadores produzindo um resultado final.

Os dados empregados neste experimento foram compilados por Lisser e Margalit (1993), destes foram extraídas 291 seqüências promotoras, utilizadas como dados positivos de treinamento. O conjunto de dados negativos foi recuperado do repositório de aprendizado de máquina (*Machine Learning Repository* - MLR) da Universidade da Califórnia em Irvine e é composto por 53 seqüências não-promotoras de DNA, cada uma com 57 nucleotídeos. Estas seqüências negativas foram concatenadas em uma única seqüência *S* com 3021 nucleotídeos e desta, escolhidas de forma aleatória 300 subseqüências, cada uma com 65 nucleotídeos, formando assim o conjunto negativo empregado aos classificadores.

3.9.2.1 Extração de Características

Dois métodos para extração de características foram utilizados sobre regiões previamente determinadas: –54, –44, –35, -29, -22, -10 e a região de início da transcrição. O método de Decomposição de Máxima Dependência (*Maximal Dependence Decomposition*-MDD) e o método baseado em motivos. Duas características extras se referem à distância em nucleotídeos entre as posições –35 e –10 e entre a posição –10 e o TSS, respectivamente.

O MDD é um método que foi proposto inicialmente para detectar locais de junção no DNA humano e estava atrelado ao software de predição de genes GENSCAN (BURGER e KARLIN, 1997), foi derivado da Matriz de Posição de Pesos (*Position Weight Matrix*-PWM) descrita por Staden (1984) para superar a limitação da seqüência consenso pela modelagem da distribuição dos nucleotídeos em cada posição. A desvantagem da PWM é que ela assume somente posições independentes. Para

solucionar esse problema foi utilizado o Modelo da Matriz de Pesos (*Weight Array Model*-WAM), uma generalização da PWM que permite dependências adjacentes entre posições. Basicamente, a WAM é uma cadeia de Markov (*Hidden Markov Models*-HMM) de primeira ordem, que pode ser generalizada para uma cadeia de segunda, de terceira, ou de mais alta ordem. Entretanto, quanto maior o número de dependências, maior é o número de parâmetros livres, o que requer um conjunto maior de treinamento para melhor ajuste do modelo. O MDD realiza um processo iterativo de clusterização do conjunto de dados baseado nas dependências mais significativas adjacentes ou não.

O funcionamento do MDD ocorre da seguinte maneira: dado um conjunto D de seqüências alinhadas, o primeiro passo é escolher o nucleotídeo consenso K_i a cada posição i (o conjunto D inclui todas as seqüências positivas de treinamento), após é calculado o qui-quadradro χ_{ij} para medir a dependência entre K_i e os nucleotídeos da posição j ($i \neq j$). Se não existem dependências significativas é utilizada uma simples PWM. Se existem dependências significativas, mas é somente entre posições adjacentes, é utilizada a WAM. Caso contrário um procedimento MDD é executado.

Este procedimento é um processo iterativo que calcula a soma $S_i = \Sigma_{i \neq j} \chi_{ij}$ para cada i, seleciona a posição m sendo S_m o máximo e decompõe o conjunto D em dois subconjuntos disjuntos: D_m que contém todas as seqüências que tem consenso com o nucleotídeio K_m da posição m e D-D_m que não tem consenso com este nucleotídeo. O procedimento MDD é aplicado recursivamente sobre D_m e D-D_m até que qualquer uma das seguintes condições seja satisfeita: nenhuma decomposição adicional seja possível, não existam dependências significativas entre posições dos subconjuntos resultantes, ou o número de seqüências nos subconjuntos resultantes esteja abaixo de um limiar, não sendo possível a estimação confiável de parâmetros após a decomposição adicional.

O outro método é baseado na extração de características de motivos pelo cálculo do valor característico de cada seqüência, com a aplicação da ferramenta Sdiscovery (WANG et al., 1994) sob os dados positivos de treinamento para encontrar motivos fracos nas regiões: -54, -44, -35, -22 e –10.

O Sdiscovery trabalha em uma primeira etapa criando uma árvore de sufixos generalizada (*Generalized Suffix Tree*-GST) da amostra de um dado conjunto de seqüências e percorre-a até encontrar todos os segmentos (candidatos a motivos) que satisfaçam um tamanho mínimo exigido. Em uma outra fase, o Sdiscovery pontua os candidatos a motivos de acordo com o número de ocorrência na amostra e então avalia os candidatos mais prováveis a motivos em relação ao conjunto inteiro de seqüências. O número de ocorrências de um motivo (segmento) refere-se ao número total de seqüências no conjunto ao qual o motivo ocorre. Para cada região o tamanho dos motivos é fixo.

3.9.2.2 Rede Neural Bayesiana

A BNN é a integração da inferência Bayesiana e das RNs, partindo da premissa que $D = \{\mathbf{x}^{(m)}, t_m\}$, $1 \leq m \leq N$, denota o conjunto de treinamento incluindo todas as seqüências de treinamento positivas e negativas, $\mathbf{x}^{(m)}$ é um vetor de entrada que representa os valores característicos do conjunto de treinamento e t_m indica a saída binária desejada para a ativação da respectiva classe, ou seja t_m é igual a 1 caso $\mathbf{x}^{(m)}$ represente uma região promotora e 0 caso contrário.

Considerando que $\mathbf{x}$ representa um vetor característico de entrada, que pode ser uma seqüência de treinamento ou teste, dado uma arquitetura A e pesos $\mathbf{w}$ para BNN, o

valor de saída *y* pode ser unicamente determinado pelo vetor de entrada **x**, uma vez que $y(\mathbf{x};\mathbf{w},\mathsf{A})$ pode ser interpretado como $P(t=1|\mathbf{x},\mathbf{w},\mathsf{A})$, ou seja, a probabilidade de **x** representar uma seqüência promotora dado os pesos **w** e uma arquitetura A. A função de probabilidade dos dados *D* de um dado modelo é calculada por:

$$P(D \mid w,\mathsf{A}\ ) = \prod_m y^{t_m}(1-y)^{1-t_m} = \exp G(D \mid w,\mathsf{A}\ ) \tag{3.22}$$

onde $G(D \mid \mathrm{w},\mathsf{A})$ é a função de erro da entropia cruzada,

$$G(D \mid w,\mathsf{A}) = \sum_m \left[t_m \log y + (1-t_m)\log(1-y)\right] \tag{3.23}$$

A função de erro da eq.3.23 é a função objetivo que deve ser minimizada no processo de treinamento de uma rede neural não-Bayesiana. O decaimento dos pesos é freqüentemente utilizado para evitar *overfitting* dos dados de treinamento e uma generalização pobre dos dados de teste pela adição do termo $\frac{\alpha}{2}\sum w^2$ na função objetivo, onde α é o parâmetro de decréscimo dos pesos (hiperparâmetro) e $\sum w^2$ é a soma dos quadrados de todos os pesos da rede. Esta função objetivo é minimizada no sentido de penalizar a grande magnitude dos pesos, penalizando um modelo complexo e favorecendo um modelo simples. Entretanto, não há uma forma precisa de especificar o valor de α, que é determinado antes do início do treinamento (*offline*).

Nas BNNs, ao contrário das RNs não-Bayesianas, o hiperparâmetro α é interpretado como um parâmetro de um modelo e é otimizado (*online*) durante o processo de treinamento. Este termo está associado com os pesos e bias em diferentes camadas e com diferentes discrepâncias, fazendo com que se torne um vetor α, o qual pode ser interpretado como a probabilidade *a priori* do vetor de pesos **w**, em uma distribuição Gaussiana com média zero e desvio padrão $\sqrt{\frac{1}{\alpha}}$. O que torna menos provável a existência de pesos grandes.

Na implementação descrita no artigo foram utilizados 3 classificadores básicos nomeados como: Classificador0, Classificador1, Classificador2, cada um representando uma BNN. O Classificador0 possui 5 nós na camada escondida e 9 unidades de entrada, sendo 7 características MDD e 2 de distância; o Classificador1 apresenta o mesmo número de nós na camada escondida que o Classificador0, no entanto tem 8 nós de entrada, sendo 6 características de motivos e 2 de distância; já o Classificador2 apresenta 6 nós na camada escondida e 15 na entrada, que representam a união das características MDD, motivos e distância, respectivamente, empregada nos classificadores anteriores. O número de nós escondidos foi determinado experimentalmente. Cada rede apresenta apenas uma camada escondida com uma função sigmóide de ativação e a camada de saída é composta de um único nó, sendo seu valor limitado entre 0 e 1 por uma função logística de ativação.

3.9.2.3 Combinação dos Classificadores

Os resultados dos classificadores descritos na sessão anterior foram combinados por dois métodos produzindo Comb0 e Comb1.

Em Comb0 foi utilizada a seguinte estratégia: se os três classificadores concordam com o resultado da classificação, o resultado final será o mesmo encontrado

pelos classificadores; se dois dos classificadores concordam no resultado, o resultado final será o mesmo destes dois; se nenhum classificador concordar com um resultado em comum, o resultado final será definido pelo mínimo entre a saída dos classificadores pela função $min(1\text{-}out_i, out_i)$.

A decisão em Comb1 é determinada pela avaliação em forma de pesos da soma da saída dos 3 classificadores, eq. 3.24. O peso de cada classificador é proporcional à taxa de classificação (1-taxa de erro) do classificador na fase de treinamento.

$$\frac{1}{weight_0 + weight_1 + weight_2} \sum_{i=0}^{2} weight_i * out_i \tag{3.24}$$

3.9.2.4 Avaliação dos Resultados

Os resultados dos três classificadores, apresentados na tab. 3.1, foram avaliados pela técnica de validação cruzada *leave one out* (ALLEN, 1974), por 10 subconjuntos, de mesmo tamanho, obtidos de uma quebra aleatória do conjunto de dados. A taxa de erro refere-se ao número total de classificações incorretas dividida pelo número total de seqüências do conjunto de teste.

Tabela 3.1: Resultados dos classificadores básicos

	Classificador0	Classificador1	Classificador2
Taxa de erro total	13%	11,2%	10,5%
Sensibilidade	83,8%	86,9%	89,3%
Especificidade	90%	90,7%	89,7%

A avaliação da combinação dos resultados é apresentada na tab. 3.2 e mostra as mesmas métricas aplicadas aos classificadores. Os resultados revelaram que o Comb1 atingiu o melhor desempenho com uma taxa de classificação de 92,2% (1-0,078).

Tabela 3.2: Combinações dos resultados dos classificadores básicos

	Comb0	Comb1
Taxa de erro total	8,8%	7,8%
Sensibilidade	89,7%	92,2%
Especificidade	92,7%	92%

3.9.3 EM e RNs para Reconhecer Promotores da *E. coli*

Este experimento foi proposto por Ma (2001) e seu grupo na tentativa de obter características relevantes de um conjunto de promotores da bactéria *E. coli* e posterior apresentação destas para treinamento e validação de uma RN. O objetivo do trabalho era a partir de uma seqüência de nucleotídeos *S* desconhecida, identificar se ela é ou não um promotor de *E. coli*. O que os autores consideram um problema de classificação binária de Mineração de Dados.

A extração de características foi baseada em uma variação dos algoritmos de Estimação-Maximização (*Expectation Maximization*–EM), os quais são utilizados em

problemas de estimação da Máxima Verossimilhança quando os dados que se possui são incompletos.

Segundo os autores, os dados utilizados nos experimentos são provenientes de outra aplicação que também empregou RNs (MAHADEVAN e GHOSH, 1994) para solução do problema possibilitando assim, a comparação de resultados. Entretanto em consulta ao artigo usado para comparação dos resultados, esses dados não apresentam a mesma composição dos empregados no experimento relatado. De qualquer forma, é exposto que foi utilizado um conjunto de 362 seqüências promotoras e 4500 seqüências randômicas, com 60% de pares de base AT, relativas a seqüências negativas para treinamento e um conjunto de teste composto por 126 seqüências positivas e 5000 seqüências randômicas negativas. Além disso, também expõe um resultado adicional relativo a uma composição de 441 seqüências promotoras, provenientes de Ozoline et al. (1997), nunca antes utilizada em outro experimento.

3.9.3.1 Obtenção das Características Via EM

As seqüências destinadas a treinamento para construção de um modelo representacional foram exploradas para obtenção de características relevantes. Inicialmente foram expostas as características padrão dos promotores, com suas duas regiões principais, seguido da descrição da falta de uma maior precisão do posicionamento destas nas várias seqüências que antecedem um gene e citados exemplos de motivos fracos, promotores de ocorrência menos freqüente, encontrados nas posições: +1, -22, -29 e -44, que contribuem para dificultar a identificação de um posicionamento padrão.

Em busca da extensão da variabilidade das principais regiões que constituem os promotores da *E.coli*, foi empregado o algoritmo EM Bayesiano (*Maximum a posteriori*-MAP). Este algoritmo é uma variação dos algoritmos EM padrão utilizados em outros experimentos que tentam desvendar promotores em *E. coli* (BAILEY e ELKAN, 1995) (CARDON e STORMO, 1992) (LAWRENCE e REILLY, 1990) e tenta considerar as regiões -35, -10 e o TSS e a influência dos pares de base que separam estas regiões.

Em geral, entre a região -10 e o TSS existem de 3 a 11 pares de base e entre as regiões -35 e -10 existem de 15 a 21 pares de base. Quando a composição deste posicionamento está bem definida, ou seja, as seqüências conhecidas são compostas por 6 pares de base em cada região, uma PWM de Staden (1984) fornece uma determinação probabilística dos nucleotídeos presentes em cada base, sendo possível a construção de um modelo. O algoritmo EM proposto pode realizar a mesma estimativa, entretanto sobre dados incompletos para estimar um modelo capaz de determinar a localização dos dois supostos sítios de ligação da RNA polimerase de qualquer seqüência de DNA.

O algoritmo EM executa iterativamente até convergir. Cada uma de suas iterações consiste de dois passos: um de estimação e outro de maximização. Em geral não há garantia que ele convirja para um máximo global, podendo cair em máximos locais. Por isto foi utilizada a variante EM (MAP), no sentido de evitar os máximos locais, uma vez que a função objetivo é mais côncava. As probabilidades *a posteriori* empregadas neste EM representam distribuições multinominais de Dirichlet (BERGER, 1985) (SANTNER, 1989).

A aplicação do algoritmo EM para obtenção das características resultou numa extensão das tradicionais regiões que representam o promotor. Com base nessa

ampliação de abrangência das regiões foi realizado um alinhamento das seqüências na tentativa de se estabelecer um padrão. Este alinhamento resultou na seleção de regiões com alto conteúdo informativo que representaram as características de cada seqüência, assim foram obtidos 17 pares de base na posição próxima a região -35, 11 pares de base na posição próxima a região -10 e 7 pares de base na posição próxima ao TSS.

Estes 35 nucleotídeos selecionados, sofreram a codificação BIN4, tratada no artigo como codificação ortogonal e foram utilizados para obtenção do modelo neural.

3.9.3.2 Configurações da RN

O conjunto característico obtido de amostras foi aplicado para treinamento de uma rede neural de arquitetura MLP completamente conectada com uma camada intermediária e função de ativação logística sigmóide. A camada de entrada possui 140 entradas, enquanto a camada de saída possui apenas 1 nó com seu valor limitado entre 0 e 1. O algoritmo de aprendizado utilizado foi o Gradiente Conjugado Escalonado (*Scaled Conjugated Gradient*-SCG).

Os métodos do gradiente conjugado possuem sua estratégia baseada no algoritmo *Backpropagation* padrão e descida do gradiente, entretanto ocorre uma seleção da direção de busca, do tamanho do passo e do termo de momento, de forma mais eficiente, utilizando informação de segunda ordem.

A primeira iteração de todos os algoritmos de gradiente inicia pela busca na direção do gradiente descendente para minimização do erro. Uma busca em linha é realizada para determinar a distância do deslocamento na direção atual. A próxima busca é realizada de modo que a direção anterior seja mantida. Quando realizada uma nova busca, esta é determinada pela combinação da nova direção do gradiente descendente com as direções anteriores. M∅LLER (1993) propôs o SCG, que é uma nova variação do gradiente conjugado, que evita a busca em linha a cada iteração utilizando uma abordagem de Levenberg-Marquardt cujo objetivo é escalar o passo de ajuste deste gradiente no sentido de obter uma aceleração da convergência.

Na realização dos experimentos com essa rede, foi constatado que com 20 nós na camada intermediária foram obtidos resultados satisfatórios.

3.9.3.3 Resultados Obtidos

As métricas usadas na avaliação dos resultados foram: precisão (eq. 3.25), sensitividade e especificidade.

$$\frac{TP}{TP + FP} * 100 \tag{3.25}$$

A comparação da abordagem proposta com o experimento de Mahadevan e Ghosh (1994) é apresentada na tab. 3.3. O método *Tenfold Cross Validation* foi aplicado, sobre amostras estratificadas, para obtenção dos resultados finais.

Tabela 3.3: Resultado comparativo do EM e o obtido por Mahadevan

	EM - RN	Mahadevan
Precisão	91,94%	90,4%
Sensibilidade	99,2%	98%
Especificidade	91,76%	90,2%

Como informação complementar, os autores relatam que obtiveram uma precisão de 96,29%, sensibilidade de 91,78% e especificidade de 96,68% sobre uma recente compilação de dados da *E. coli* com 441 seqüências.

3.9.4 RNs para Reconhecer Promotores em Micobactérias

Este experimento desenvolvido por Kalate et al. (2003) e seu grupo, inicia descrevendo as particularidades dos organismos da família Micobactéria que apresentam baixa taxa de transcrição, sendo seus genomas ricos em conteúdo G+C. Segundo Nakayama et al. (1989) e Ohama et al. (1987), isto dificulta o reconhecimento de regiões promotoras desses organismos, uma vez que aparentemente os sinais de transcrição e tradução são diferentes do padrão *E. coli*. Estudos revelam que em algumas Micobactérias existe certa similaridade com os promotores da *E. coli* na região -35, a qual é considerada essencial para transcrição, apesar de que sua exata localização seja um fator crítico, além de um alto conteúdo GC na região -10, que talvez seja uma característica das Micobactérias.

Entretanto, foi constatada uma ampla variação na constituição dos promotores Micobactériais caracterizados, o que sugere a inexistência de uma seqüência consenso representativa (MULDER et al., 1997).

São expostas as diversas tentativas de se construir ferramentas baseadas em análises estatísticas para descobrir promotores, mas no entanto não é obtido o resultado desejável. Assim são aplicadas as redes neurais por serem métodos aplicáveis a identificação de padrões não-lineares e por oferecerem a capacidade de generalização.

Como objetivos principais este experimento se preocupou em: usar as RNs para diferenciar seqüências promotoras Micobacteriais de seqüências randômicas (não promotoras) e determinar pela abordagem de RNs em conjunto com uma randomização calibrada a importância funcional e estrutural de sub-regiões das seqüências promotoras de Micobactérias.

3.9.4.1 Dados Empregados

As seqüências promotoras foram compiladas por Kalate et al. (2002), com base num amplo número de estudos investigativos (aproximadamente 80) relativo aos promotores de Micobactérias. O conjunto promotor foi inicialmente composto por 125 seqüências, destas 80 tinham seu TSS mapeado, enquanto 45 eram seqüências de supostos promotores. Em poucas seqüências foi detectada a existência de 2 ou mais seqüências que alternavam o consenso. Assim o conjunto total de promotores foi definido em 135 seqüências.

Este conjunto de 135 seqüências foi alinhado com o software ClustalW e então dividido em um subconjunto de 90 seqüências para treino e outro de 45 seqüências para teste. O conjunto negativo foi gerado randomicamente, onde a probabilidade de ocorrência de cada nucleotídeo foi estipulada em 0,25. A proporção de seqüências

negativas geradas foi de 3:1 sendo o conjunto final de treino estabelecido em 380 seqüências e o de teste em 160 seqüências. Sobre estes subconjuntos foi aplicada a codificação BIN4 e cada amostra obteve um tamanho de 284 elementos (correspondente a 71 pb).

3.9.4.2 Configurações da Rede Neural

A rede utilizada foi uma MLP com algoritmo de aprendizado *Backpropagation* padrão e uso da função de ativação logística sigmóide. A taxa de aprendizado foi fixada em 0,6 e a constante de momento em 0,4. A camada de entrada foi composta por 284 nós, a camada intermediária e a camada de saída possuem ambas apenas 1 nó cada. O nó de saída utiliza um valor limiar de 0,5 para classificar as seqüências como promotoras ou não, sendo que todo valor obtido abaixo deste limiar é considerado não promotor.

3.9.4.3 Resultados Obtidos

O subconjunto reservado para teste foi avaliado pela medida de acurácia, eq. 3.26, obtendo um desempenho de 96,9%, com a rede não predizendo nenhum falso positivo.

$$\frac{TP + TN}{N} * 100, \tag{3.26}$$

onde N é o numero total de amostras.

Como forma de complementar a validação, foi realizado um experimento adicional com a criação de 500 novas seqüências randômicas negativas com 56% de conteúdo G+C e submissão destas ao modelo neural gerado. Como resultado foi obtido 99,6% de acurácia.

Em virtude dos bons resultados alcançados foi proposto que a rede poderia ser utilizada na identificação de importantes sub-regiões das seqüências promotoras. Para tentar comprovar isso foi empregada uma estratégia de randomização calibrada, onde cada seqüência promotora Micobacteria foi randomizada em partes e submetida à rede neural treinada com o propósito de avaliar se esta seqüência ainda retinha suas características originais de promotora. Este experimento foi realizado somente com as amostras que tinham seu TSS mapeado.

Uma janela de tamanho fixo, em 10 pb, foi escolhida para randomização. Esta janela se deslocava do início ao final de seqüência em um par de base, os 71 pares de base constituintes de cada seqüência positiva serviram para a construção de 62 novas seqüências que se diferiam na sub-região de 10 pb randomizada.

Os resultados obtidos revelaram que a localização inicial da janela randomizada exerceu um grande papel na classificação. Foi descoberto que quando a posição de início da janela randomizada estava disposta sobre a região -42 a -35 as seqüências resultantes eram predominantemente classificadas como não promotoras. Esta observação sugeriu que o conteúdo dos nucleotídeos e sua disposição na localização calibrada da região -42 a -35 eram indicativos de fatores críticos para os promotores Micobacteriais.

A randomização da região espaçadora entre as regiões -10 e -35, assim como da própria região -10 também ocasionaram descaracterização das características promotoras. Entretanto, neste caso, o percentual de seqüências classificadas como não promotoras não foi tão elevado quanto o das localizadas na região -42 a -35.

Outro resultado, revelou que quando o ponto de início da janela randomizada foi a posição -38 obteve-se o resultado mais elevado (37%) de classificação de não promotores. Assim, foi inferido que a região de -38 a -29 exercia grande influência para identificar promotores e não promotores em Micobactérias.

Isto sugeriu uma investigação mais criteriosa, sendo realizado um alinhamento das seqüências originais pelo seu TSS e observadas as composições da região -38 a -29 na tentativa de identificar um padrão consenso. A seqüência consenso recuperada foi a seguinte: A_{31} C_{30} T_{43} T_{49} G_{44} G_{27} C_{34} C_{37} T_{37} C_{40}, sendo identificado que a sub-região -38 a -34, desta seqüência consenso, compreende um único 'A' e dois 'Ts' e a sub-região -33 a -29 é rica em 'GC'. Foi constatado um elevado conteúdo AT na primeira sub-região (-38 a -34) que representa uma contribuição para aumentar a significância de promotores Micobacteriais.

O processo de randomização revelou que: a região a montante da -35, a região -35, o espaço entre as regiões -35 e -10 e a região -10 apresentam semelhança ao tipo de promotores da *E. coli* em relação ao σ^{70}.

3.9.5 Incremento na Capacidade de Predição do NNPP2.2

O experimento proposto por Burden et al. (2005) expõe uma metodologia para ser incorporada a ferramentas de predição de promotores no sentido de reduzir o número de falsos positivos preditos.

Inicialmente o trabalho menciona algumas dificuldades inerentes na descoberta de promotores, tais como: as interações entre proteínas e a seqüência de DNA na região promotora são complexas e os vários elementos envolvidos são altamente degenerados; a característica mais distinguível de muitos promotores é frequentemente a presença de um ou mais motivos característicos *upstream* do TSS e o espaçamento e a distância entre esses elementos conservados; os motivos detectáveis de promotores ocorrem randomicamente ao longo de todo o genoma, sendo que os algoritmos de predição deveriam incorporar múltiplas características que podem estar presentes ou não num dado promotor.

Recorda que os promotores estão associados com pelo menos uma região codificante e que após a transcrição, estas regiões codificantes, na forma de RNA são traduzidas em uma proteína. Então imediatamente *downstream* à região promotora e ao TSS existe pelo menos um gene. O primeiro nucleotídeo *downstream* à região promotora a ser traduzido é denotado como o sítio de início da tradução (*Translation Start Site* – TLS).

O trabalho sugere que, para incrementar a capacidade preditiva dos algoritmos disponíveis, incluindo os que empregam RNs, é necessário introduzir outras medidas que contribuam na redução de falsas predições. Assim, aborda que a distância entre o TSS e o TLS não é explicitamente empregada nos algoritmos de predição de promotores existentes.

O experimento relatado incorpora a distância TSS-TLS calculada ao resultado de um programa de predição de promotores (NNPP2.2) que emprega redes neurais. Esta

metodologia, chamada de TLS-NNPP, é aplicada com sucesso somente quando existir um número considerável de seqüências promotoras experimentalmente determinadas.

3.9.5.1 A Distância TSS-TLS

A distância entre um promotor (definida pela associação ao TSS) e o TLS da região codificante subseqüente pode contribuir na predição de promotores, sendo possível estimar a probabilidade de ocorrência de um promotor a uma dada distância do TLS e posterior uso dessa informação para melhorar a capacidade preditiva das atuais ferramentas de predição de promotores, porque o TSS e o TLS podem ser definidos para todos os genes e a localização do TSS está estritamente relacionada à região promotora.

O TLS em um gene é prontamente identificável devido à estrutura das regiões codificantes do DNA, correspondendo ao primeiro nucleotídeo na região codificante do gene, sendo sua posição facilmente definida. Conforme exibido na fig. 3.8, a distância TSS-TLS é definida em termos do número de nucleotídeos entre essas posições.

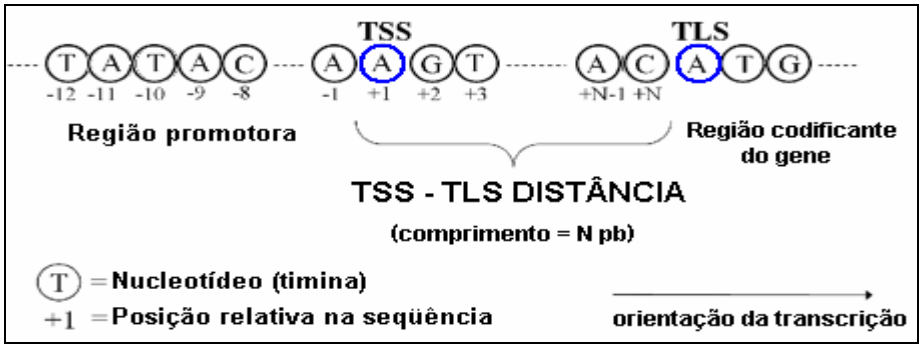


Figura 3.8: Distância TSS-TLS (BURDEN et al., 2005)

3.9.5.2 Dados Empregados

Um dos conjuntos utilizados é relativo a 771 promotores da *E. coli K12*, proveniente do banco de dados EcoCyc 7.1 (KARP et al., 2002), que está associado com 587 genes. Estes dados foram empregados somente na avaliação da distribuição da distância TSS-TLS e apontaram para as seguintes observações: intervalo de 0-920 pb, com média de 104.5 pb, mediana de 61 pb e um desvio padrão de 119 pb, sendo que 95% dos promotores possuem essa distância inferior a 325 pb.

O outro conjunto de dados foi empregado para avaliação da metodologia como um todo e refere-se a 510 seqüências do genoma da *E. coli* que contém 671 promotores conhecidos. Cada seqüência é constituída por 500 pb, iniciando na posição 501 *upstream* do TLS, terminando na posição -1 antes do TLS.

3.9.5.3 O Experimento e a Rede

Este experimento usou o programa NNPP2.2 (Neural Network Promoter Prediction) (REESE, 2000), inicialmente desenvolvido para predição de promotores em eucariotos, o programa foi treinado e ajustado para identificar promotores em procariotos com base em um conjunto de 272 promotores comprovados da *E. coli*.

Este programa utiliza uma série de TDNNs (*Temporal Delay Neural Networks*) (a qual é brevemente descrita no próximo sub-item) que incorporam múltiplos elementos promotores com espaçamentos variáveis.

Este programa requer como entrada uma seqüência de DNA com mais de 51 pb e produz como saída uma lista de predições com índices de classificação superiores a um limiar estipulado pelo usuário. Todos os índices estão em um intervalo de 0 a 1 e representam a probabilidade da amostra submetida ser um promotor. Devido ao fato dos elementos promotores poderem aparecer em diferentes posições relativas em relação ao TSS, o NNPP2.2 possui uma acurácia posicional na predição de ±3 pb.

O experimento proposto segue os seguintes passos:

- Cálculo da distribuição empírica das distâncias TSS-TLS para os promotores conhecidos;
- Submissão das seqüências de DNA ao NNPP2.2 para obter as localizações dos promotores preditos;
- Associação da predição com o próximo TLS na seqüência;
- Ao final, para cada predição, é calculada uma distribuição empírica das duas características associadas e criado um conjunto de predições usando uma probabilidade de corte apropriada.

3.9.5.3.1 A TDNN

A rede TDNN teve origem no trabalho de Lang e Hinton (1988) e Waibel et al. (1989) que a aplicaram ao problema de reconhecimento de fonemas em séries temporais com deslocamento local de tempo. A TDNN se caracteriza por ser uma rede *feedforward* de múltiplas camadas, onde tanto os nós da camada oculta como os nós da camada de saída são replicados através do tempo (HAYKIN, 2001).

A primeira característica da TDNN são as entradas com atraso temporal para os nós. Cada atraso temporal está conectado ao nó via um peso específico, representando o valor de entrada num momento passado. A fig. 3.9 apresenta um nó da TDNN com duas entradas, onde é possível visualizar as entradas que alimentam a rede (In1 e In2), com suas respectivas informações passadas e futuras. Isto força a TDNN a generalizar não somente a entrada corrente, mas também a informação passada e futura das entradas.

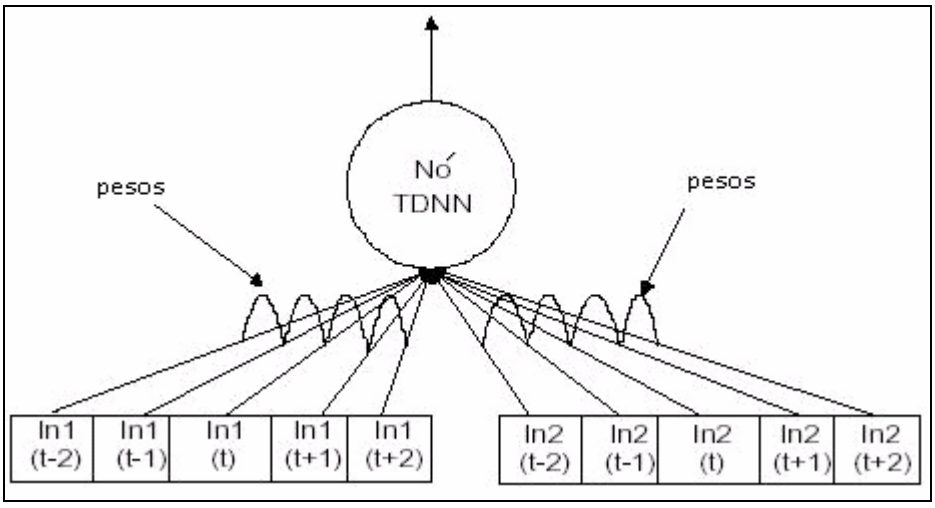


Figura 3.9: Nó da TDNN

Dentre as propriedades da TDNN destacam-se:

- **A rede é invariante a atraso temporal**: um determinado padrão pode ser corretamente reconhecido e classificado, desconsiderando a sua localização temporal;
- **A rede não é sensível a desalinhamentos**: a rede não só é capaz de aprender a partir de dados mal alinhados, bem como é capaz de corrigir estes desalinhamentos;
- **A rede requer um conjunto de treinamento reduzido**: quando apresentadas entradas com modificações temporais e pesos impostos (pré-determinados), a rede é capaz de generalizar muito bem, permitindo o uso de um conjunto limitado de dados para seu treinamento (YAN, 2000).

Outra característica da TDNN é possuir uma região em que estão completamente conectadas as unidades de entrada e seus atrasos temporais, denominados campos receptivos. Estes campos são usualmente, mas não necessariamente, tão amplos quanto o número de unidades características (unidades conectadas à característica a ser aprendida), estas unidades podem estar divididas entre vários campos receptivos. Os campos receptivos podem estar sobrepostos num plano de origem, mas tem que englobar todas as unidades características (WAIBEL et al., 1989).

3.9.5.4 Resultados

Os resultados computados referem-se a predição somente com o NNPP2.2, tab. 3.4, e a predição com o TLS-NNPP, tab. 3.5.

Tabela 3.4: Predição do NNPP2.2 para 671 promotores conhecidos

corte	***TP***	***TP+FP***	***Precisão%***	***P[Prom]***
>0,98	42	303	6,3	0,139
>0,9	135	1055	20,1	0,128
>0,8	208	1774	31	0,117
>0,7	255	2339	38	0,109
>0,6	297	2896	44,3	0,103
>0,5	333	3502	49,6	0,095
>0,4	356	4116	53,1	0,086
>0,3	401	4917	59,8	0,082
>0,2	446	6124	66,5	0,073
>0,1	470	7584	70	0,062

O valor ***corte*** na tab. 3.4 refere-se ao limiar definido pelo usuário no NNPP2.2, enquanto que na tab. 3.5 ele refere-se à distribuição empírica do resultado do NNPP2.2 associado a distância TSS-TLS; ***TP*** é o número de predições verdadeiras com o NNPP2.2 utilizando um limiar superior a 0,1; ***TP+FP*** é o número total de seqüências testadas com índice acima do limiar; ***Precisão*** é o percentual de predições corretas

obtidas no total (*TP*/671) e ***P[Prom]*** é a probabilidade estimada de uma seqüência teste ser um promotor verdadeiro.

Tabela 3.5: Predição do TLS-NNPP para 671 promotores conhecidos

corte	*TP*	*TP+FP*	*Precisão%*	*P[Prom]*
>0,0605	27	56	4	0,482
>0,0538	69	150	10,3	0,460
>0,0471	97	245	14,5	0,396
>0,0403	122	380	18,2	0,321
>0,0336	159	529	23,7	0,301
>0,0269	202	766	30,1	0,264
>0,0202	245	1097	36,5	0,223
>0,0134	303	1616	45,2	0,188
>0,0067	383	2638	57,1	0,145
>0,0034	430	3939	64,1	0,109

Comparadas as duas tabelas, os resultados demonstraram que o número de predições verdadeiras foi reduzido quando aplicada a metodologia TLS-NNPP, no entanto o número de falsas predições sofreu uma queda considerável.

3.9.6 Uma Metodologia Híbrida para a Identificação de Promotores em Procariotos

Este experimento descreve a proposta de (COTIK et al., 2005) de criar uma metodologia híbrida, denominada HPAM (*Hybrid Promoter Analysis Methodology*), para a descoberta de promotores que combina: a eficiência e a habilidade das RNs na representação de padrões imprecisos e incompletos; a flexibilidade e interpretabilidade dos modelos *Fuzzy*; e a capacidade dos algoritmos evolucionários multi-objetivos em identificar ótimos exemplos de um modelo pela procura relacionada a múltiplos critérios.

O HPAM é considerado uma abordagem híbrida de Aprendizado de Máquina por compor uma aplicação seqüencial com três métodos distintos. A rede TDNN (subseção 3.9.5.3.1) responsável por aprender motivos das regiões de ligação referentes a seqüências não específicas de DNA, decompondo um conjunto de motivos em módulos, onde cada módulo corresponde a uma característica. Em seqüência, no sentido de fornecer maior clareza aos resultados obtidos pela rede, foram usadas inferências da Lógica *Fuzzy* para associar os módulos identificados pela rede. E finalmente, empregado o método de reconhecimento de padrões MOSS (*Multi-objective Scatter Search*), que usa Algoritmos Genéticos para encontrar os motivos mais representativos previamente descobertos pela sinergia dos outros dois métodos aplicados.

3.9.6.1 O HPAM

A fig. 3.10 apresenta a estrutura implementada para o HPAM. Em (A) a TDNN recebe como entrada uma seqüência de DNA, a qual é quebrada em janelas de tamanho fixo.

Em (B) a saída da TDNN é o conjunto de promotores preditos com a localização de suas seqüências conservadas. Os sub-motivos reconhecidos por cada módulo da rede foram inspecionados para obter a freqüência de nucleotídeos, bem como a freqüência de distribuição das distâncias entre eles. A partir dessas distribuições de probabilidade para identificar sub-motivos, foram calculados histogramas das freqüências de seus nucleotídeos constituintes e suas distâncias. Os modelos *Fuzzy* foram adquiridos com base nestas distribuições prévias que serviram para a construção das funções de pertinência, determinadas pelo cálculo proposicional da freqüência de seus nucleotídeos consenso na forma de conjuntos *Fuzzy* discretos.

Cada predição da rede e os modelos *Fuzzy* são usados como entrada do método MOSS. Em (C) o MOSS é utilizado, especificamente, para reconhecer todos os ótimos motivos localizados em uma seqüência de DNA. O MOSS considera a relação de uma seqüência de DNA com cada sub-motivo promotor e as suas distâncias, como múltiplos objetivos que devem ser otimizados. Além disso o MOSS é capaz de tratar problemas multi-modais, como encontrar mais de uma solução para cada região promotora. O algoritmo evolucionário utilizado pelo MOSS é uma extensão da heurística original *scatter search* (LAGUNA e MARTI, 2003), que usa as regiões promotoras detectadas pela TDNN e os modelos *Fuzzy* identificados para cada módulo promotor como entradas e encontra todos os exemplos ótimos que satisfaçam as restrições do modelo.

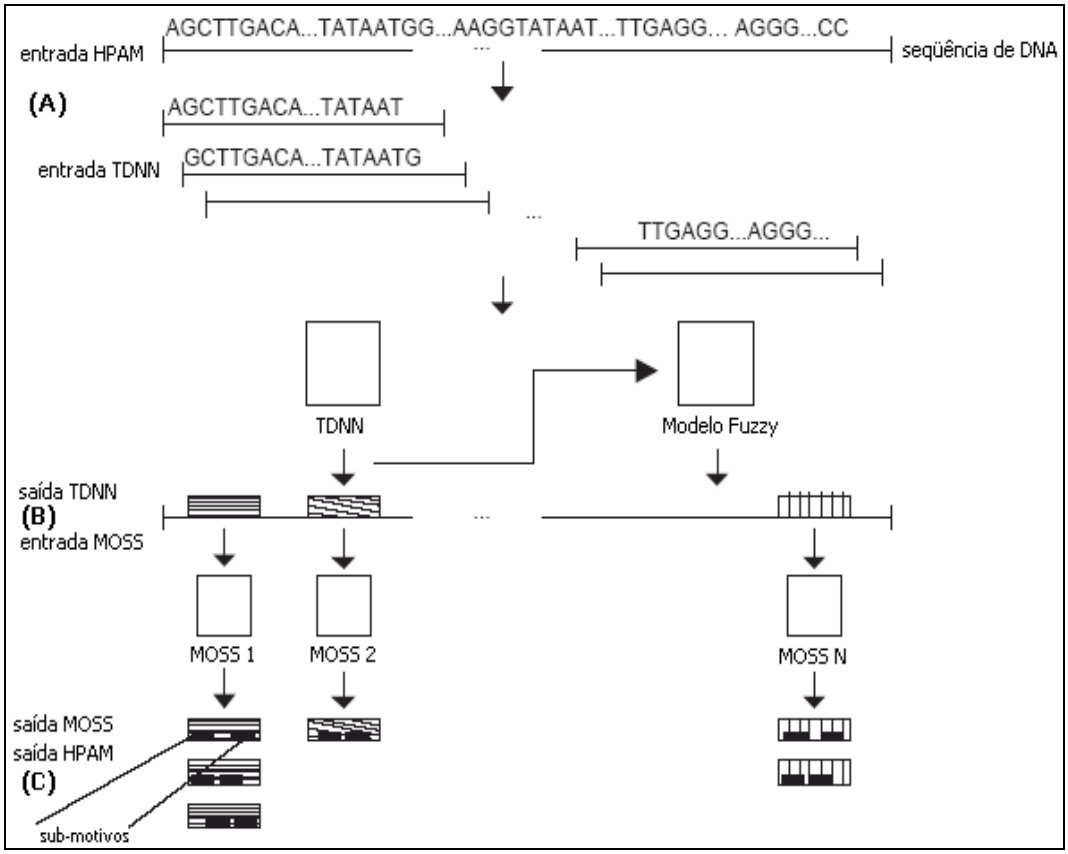


Figura 3.10: Metodologia do HPAM (COTIK et al., 2005)

3.9.6.2 Experimento

Ao HPAM foi submetido um conjunto de seqüências conhecido por conter as regiões de ligação com a RNA polimerase com mais de um motivo alternativo na mesma região regulatória (HARLEY e RAYNOLDS, 1987), 272 regiões promotoras

candidatas de *E. coli*, foram identificadas por meio de suas regiões -35 e -10. Randomicamente, foram selecionados 60% dos dados do conjunto para treinamento e o restante destinada para teste. As seqüências possuíam 46 pb.

O conjunto negativo foi gerado randomicamente com igual probabilidade para cada nucleotídeo. A proporção de positivos versus negativos foi de 1:10, respectivamente, para o conjunto de treinamento e 1:25, respectivamente, para o conjunto de teste.

A implementação da TDNN foi realizada no SNNS[11].

Os modelos *Fuzzy* foram modelados a partir da distribuição correspondente de probabilidades para as freqüências de nucleotídeos que compunham os sub-motivos e as distâncias entre os mesmos. O método MOSS foi executado 20 vezes com diferentes sementes para cada seqüência de entrada.

3.9.6.3 Resultados

Ambos os métodos, TDNN e MOSS, foram comparados individualmente com outros métodos. A TDNN foi comparada com um método baseado em expressões, que considera descasamentos de um consenso como probabilidade *crisp* em uma representação (0,1) e também comparada com os programas *Consensus*/*Patser* (HERTZ e STORMO, 1999) que representam os motivos como sendo matrizes com uma escala [0,1]. O MOSS foi comparado com duas abordagens evolucionárias: o SPEA (*Strength Pareto Evolutionary Algorithm*) (ZITZLER e THIELE, 1999) e o (μ+λ) GA (SARKER et al., 2002).

A tab. 3.6 apresenta os resultados de um experimento que utilizou 5 taxas diferentes de verdadeiros positivos (*TP*) e falsos positivos (*FP*) com os dados de treinamento e teste submetidos à TDNN.

Tabela 3.6: Resultados da TDNN para dados de treino e teste

	0%FP		*1%FP*		*5%FP*		*100% TP*		*80%TP*		*Melhor CC*		
Método	*%TP*	*CC*	*%TP*	*CC*	*%TP*	*CC*	*%FP*	*CC*	*%FP*	*CC*	*CC*	*%TP*	*%FP*
Treino	41,45	0,51	83,55	0,84	93,42	0,89	17,83	0,82	0,72	0,81	0,9	91,45	1,91
Teste	13,33	0,27	76,67	0,78	93,33	0,88	23	0,79	1,4	0,8	0,9	91,83	5,83

Conforme os resultados apresentados na tab. 3.7, a TDNN obteve os melhores resultados frente às metodologias comparadas.

Tabela 3.7: Comparação dos resultados da TDNN com outros métodos a dados de teste

Método	***CC***	***%TP***	***%FP***
TDNN	0,90	95,83	5,83
Crisp	0,46	43,33	3,9
Consensus-Patser	0,68	74	7

[11] *Stuttgart Neural Network Simulator*: simulador de Redes Neurais disponível em: http://www-ra.informatik.uni-tuebingen.de/SNNS/

Como pode existir mais de uma possível descrição para cada região promotora, o método MOSS e seus concorrentes foram testados com as 261 regiões promotoras reconhecidas pela TDNN e mais 60 soluções alternativas (múltiplos promotores) definidas para seqüências correspondentes totalizando 329 regiões. A tab. 3.8 apresenta os resultados comparativos.

Tabela 3.8: Comparação dos resultados do MOSS e outros métodos evolucionários a dados de teste

	original	**Alternativo**	**%original**	**%alternativo**	**Total**	**%total**
MOSS	243	59	93,1	86,76	302	91,79
SPEA	217	43	83,14	63,24	260	79,03
(μ+λ) GA	223	52	85,44	76,47	275	83,59

3.9.7 Comentários dos Experimentos Descritos

Os dois experimentos iniciais relatados tratavam de compilações de dados com diferentes quantidades de amostras. No entanto, referiam-se ao mesmo organismo, a *E. coli*. Ambos utilizaram técnicas para a extração de características, que objetivavam uma busca adicional por supostas regiões de interesse, além das já tradicionais regiões -10 e -35. As técnicas de extração de características combinadas com as redes neurais produziram excelentes resultados ao problema com taxas de reconhecimento superiores a 90%.

O único trabalho relatado que não aborda dados do organismo *E. coli*, foi o que empregou RNs para reconhecer promotores em Micobactérias, outro organismo além da *E. coli*, que apresenta um número razoável de seqüências com as duas principais regiões de interesse bem definidas. Uma contribuição adicional desse artigo, foi uma seleção de um conjunto heterogêneo de seqüências com base em características como: distância de pares de base entre as principais regiões de interesse, distância entre a região -10 e o TSS e percentual de conteúdo (A+T) e (C+G). Outra observação é que foram usadas três vezes mais amostras negativas do que positivas para a construção dos modelos.

O penúltimo experimento abordado, tratou de melhorar o resultado preditivo de uma ferramenta tradicional de predição de promotores em organismos eucarióticos que utiliza as redes TDNNs. Entretanto, essa rede foi modelada para identificar promotores em procariotos, apresentando uma quantidade grande de falsos preditos. Uma métrica de cálculo da distância entre o ponto de início da transcrição e ponto de início da tradução do gene relativo ao promotor, foi a solução adotada pelo experimento para aumentar o poder preditivo do programa já existente, conseguindo bons resultados diante de 671 promotores. Por outro lado ocorreu uma pequena queda na capacidade de identificar amostras positivas.

O último experimento exposto, apresentou um modelo híbrido que empregou três abordagens da Inteligência Artificial, entre elas a rede TDNN. A essa arquitetura maciça de métodos foram submetidas seqüências de *E. coli*, que apresentaram resultados significativos.

Duas questões importantes observadas nestes experimentos são: a grande quantidade de seqüências comprovadas utilizadas nos experimento, daí a justificativa de

uso de dados da *E. coli* na maior parte dos experimentos; e, além do número superior de seqüências, o uso de estratégias para extração de características mais relevantes que colaborem com o poder preditivo das RNs.

Não foram encontrados na literatura trabalhos científicos que tratam da utilização de RNs para reconhecimento de promotores em *Mycoplasmas* e sim somente poucos artigos (Weiner et al. (2000) e Eskin et al. (2003)) que descrevem um número reduzido de seqüências de alguns organismos desta família e que motivaram a realização de uma investigação inicial ao problema.

4 EXPERIMENTOS E RESULTADOS

Neste capítulo são relatados seis experimentos realizados em ordem cronológica que demonstram o desenvolvimento de estratégias para obtenção de melhores resultados para o problema de reconhecimento de regiões promotoras em *Mycoplasmas*. As maiores dificuldades encontradas tratam da baixa caracterização e um número insuficiente de amostras representativas referentes ao problema. O primeiro e segundo experimentos abordam o uso de dados comprovados efetivamente por meios biológicos de seqüências promotoras; quanto aos demais experimentos a intenção foi obter um maior número de seqüências, no entanto sem a precisão da localização do promotor na seqüência e sim se estas continham ou não informação relativa a promotor. Conforme relatado, em cada experimento foram empregadas técnicas de aprendizado supervisionado e clusterização juntamente com testes estatísticos dos resultados obtidos. No processo de tratamento dos dados foram usados arquivos de texto e os softwares: Artemis 7 e EMBOSS; e o *Neural Network Toolbox* do Matlab tanto para preparação dos dados como para implementação das redes neurais.

4.1 Padronizações

Esta seção visa fornecer algumas padronizações que serão empregadas nos experimentos como codificação dos dados, testes de validação cruzada e algoritmos de aprendizado das RNs. As métricas utilizadas para avaliação dos testes de validação dos experimentos propostos são as mesmas descritas no capítulo anterior, subseção 3.9.1.

4.1.1 Codificação dos Pares de Base

Para tratamento digital, os pares de base foram transformados para forma binária, seguindo a codificação conhecida como BIN4 (WU e McLARTY, 2000), em que cada par de base recebeu a seguinte representação:

a = {1 0 0 0};

t = {0 1 0 0};

g = {0 0 1 0};

c = {0 0 0 1}.

Por exemplo, uma seqüência nucleotídica com 50 pb submetida à codificação BIN4 acaba sendo representada por um vetor de tamanho 200 elementos. Para formar uma amostra de treinamento, ao final de cada seqüência codificada foram acrescidos 2 bits {1 0} ou {0 1} que representam a saída conhecida da classe, promotor ou não promotor, respectivamente.

4.1.2 Testes de Validação Cruzada

4.1.2.1 Leave-one-out

Este método é computacionalmente dispendioso e freqüentemente é utilizado em conjuntos reduzidos de amostras. Para um conjunto de tamanho N uma hipótese é induzida utilizando $(N - 1)$ exemplos; a hipótese é então testada no único exemplo (amostra) remanescente. Este processo é repetido N vezes, cada vez induzindo uma hipótese e deixando de considerar um único exemplo (BARANAUSKAS, 2001). O erro de classificação é a soma dos erros de cada teste dividido por N (WHELAN e MOLLOY, 2001).

4.1.2.2 Tenfold Cross Validation

Essa técnica é semelhante à anterior, com a diferença de ser aplicada a conjuntos compostos por um maior número de amostras. Ao invés de ser retirada uma única amostra, todo o conjunto é particionado em 10 subconjuntos sendo reservado um dos subconjuntos para teste e os demais aplicados para treinamento do modelo. Este é um processo cíclico em que a cada rodada são utilizados 9 subconjuntos para treino e 1 subconjunto é reservado para teste executado até que o último subconjunto seja destinado para teste.

4.1.2.3 Partição em Treinamento e Teste

Outra metodologia de validação aplicada aos experimentos realizados foi o particionamento dos conjuntos de amostras em treino e teste, onde um percentual de amostras é reservado para cada conjunto, geralmente 80% para treino e os 20% restantes destinadas a teste.

4.1.3 Algoritmos de Treinamento Supervisionado

Os algoritmos de treinamento supervisionado utilizados nos experimentos I, III, V e VI referem-se ao Gradiente Descendente com termo de Momento (GDX), descrito na subseção 3.8.1.1, e ao *Resilient Back-propagation* (RPROP), descrito na subseção 3.8.1.2. Ambos estão presentes no *Neural Network Toolbox* do Matlab e são representados pelas funções ‘traingdx’ e ‘trainrp’, respectivamente.

4.2 Experimento I

Este experimento relata a obtenção de seqüências relativas aos *Mycoplasmas* que apresentam a região promotora definida, sendo descrita a composição de conjuntos de amostras de diferentes fontes com base na localização da região promotora e emprego desses dados em diferentes composições para treinamento de redes neurais supervisionadas e avaliação dos resultados.

4.2.1 Dados Utilizados

Os dados utilizados apresentam duas fontes: um primeiro conjunto composto por 32 seqüências foi obtido do artigo *Transcription in Mycoplasma pneumoniae* (WEINER et al., 2000), que descreve como obteve essas seqüências com a realização de experimentos laboratoriais que tratam desde a criação de culturas com a bactéria até a obtenção da lista de seqüências descritas no artigo.

O segundo conjunto de dados foi obtido por meio de consulta ao site do NCBI, por ser o BD mais completo, realizada em janeiro de 2004, sendo realizada uma busca por termos representativos presentes em promotores como : “-10”, “-35”, “rbs”, “adhesin”, em conjunto com as expressões: “promoters” e “mycoplasma family”.

A maneira como os dados do primeiro conjunto são apresentados no artigo, apresenta as seqüências em uma forma resumida, ou seja, com poucos pares de base e alguns cortes, não sendo adequadas para o tratamento computacional proposto.

Cada seqüência que referencia um gene apresenta um identificador que permite encontrar dentro do genoma completo da bactéria, disponível no NCBI, no caso o *Mycoplasma pneumoniae*, a posição de início e término do gene referenciado. Com base nessa informação foi efetuada uma busca na seqüência de nucleotídeos anterior ao início dos genes para encontrar as seqüências apresentadas no artigo que apresentavam este identificador e assim obter a referida região promotora.

Quando realizado esse levantamento e encontrada essa seqüência dentro do genoma, foi feito um recorte de 10 pares de base anteriores à região -10 até a posição -50, obtendo-se assim uma seqüência com 50 pares de base. A razão pela escolha dessa região -10, conforme Lewin (2001) e Zaha et al. (2003), é que esta apresenta um maior consenso em sua composição. Desta forma, pode ser utilizada como referência para obtenção das seqüências promotoras. Já o fato que levou a escolha de um tamanho de 50 pb faz menção a englobar ambas as características que compõem a região promotora e a fronteira de decisão entre o que seria considerado promotor e o que não seria promotor.

Ao contrário da composição do primeiro conjunto, o segundo conjunto não apresenta um artigo como referência e sim resultados retornados da consulta ao BD por meio de palavras chave, que conduzem a determinados genes que apresentam alguma relação com o objeto de busca. Num momento já refinado da busca, se obteve 46 seqüências, no entanto fatores como: determinação de somente uma região promotora (-10 ou -35), falta de um número adequado de pares de base definido como mínimo, ou alguma outra imprecisão acabaram limitando o conjunto a 21 seqüências.

4.2.2 Pré-processamento dos Dados

De posse das seqüências alvo de promotores algumas manipulações foram realizadas não somente para tornar viável seu uso nas técnicas utilizadas, mas também para proporcionar uma maior representatividade da informação e desta forma, as seqüências de nucleotídeos são transformadas em amostras.

Conforme visto no capítulo 2, os nucleotídeos de uma fita de DNA recebem marcações previamente estabelecidas para melhor manipulação da seqüência gênica, onde: o ponto de início do gene é determinado como +1 (um), os pares a sua direita seguem uma numeração positiva, enquanto os da esquerda uma numeração negativa. Isso ocorre a cada novo gene encontrado na fita.

Devido à característica do DNA em poder apresentar os genes ora numa fita, ora em outra, e o fato de os promotores poderem ser encontrados em qualquer uma destas, se optou, por simples escolha lógica de programação, tratar todas as seqüências na forma inversa, ou seja, todas as seqüências que já estavam naturalmente assim, permaneceram desta forma, quanto as demais seqüências, orientação 5'-3', tiveram suas seqüências invertidas. Assim, após esse processo, todas as seqüências ficaram alinhadas pela região -10.

Com os dados pré-processados e codificados foi efetuada a geração dos conjuntos de dados para treinamento e teste da rede neural.

Os dados que compõem as amostras presentes nos dois conjuntos de dados descritos referenciam somente dados relativos a promotores. No entanto, o problema exige uma solução para identificação do que é e o que não é promotor, uma vez que na maioria das seqüências de nucleotídeos é difícil existir uma determinação clara para essa classificação. Assim, se propôs um experimento que produz amostras sintéticas com base nas seqüências reais de dados referentes a promotores, para a composição de seqüências de não promotores.

A origem destas seqüências artificiais, está baseada na proposição que existem duas regiões que fazem menção à localização de pares de base indicativos da presença do promotor. Uma como já citado é a região -10 composta por 6 pares de base. E a outra região indicada por conter pares de base referentes a promotores é a região -35, porém essa região não apresenta grande consenso, ou seja, não apresenta uma definição clara e exata deste posicionamento.

Com base nessa afirmação se procurou delimitar duas janelas sobre cada uma das regiões indicativas. Uma janela foi posicionada na região -10 com o tamanho de 6 pb e a outra janela, de mesmo tamanho, foi posicionada sobre a possível região -35. Devido à imprecisão desta região foram realizados dois experimentos extras, variando o tamanho dessa segunda janela: um empregando uma janela com tamanho de 10 pb e outro com uma janela de 15 pb, englobando uma quantidade maior de nucleotídeos. A escolha do tamanho dessa segunda janela em 10 ou 15 pares de base foi devido à variação observada nas seqüências de promotores, onde a região -35 apresentava a menor e a maior distância, em pares de base, em relação à região -10, o que englobava 10 ou 15 pb. Desta forma, com a junção das duas janelas, foram obtidas somente informações relativas ao promotor.

A criação de amostras que caracterizassem seqüências não promotoras ocorreu pelo deslocamento das referidas janelas em uma posição, um par de base, à direita das posições indicativas da presença de promotor até atingir o final da seqüência; ou à esquerda das posições indicativas da presença de promotor até atingir o início da seqüência. Assim a cada deslocamento dessas janelas em uma posição à direita ou à esquerda, e a junção das referidas janelas, produziu-se uma seqüência que referencia algo que parece ser promotor, pelo posicionamento próximo à região indicativa, ou mesmo porque com um deslocamento de apenas um par de base ainda existem vários pares de base na janela que indicam a presença do promotor, mas que no entanto não consiste na presença de um promotor exato.

Essa metodologia foi adotada no sentido de aproximar a realidade da fronteira de decisão, onde existe um limiar que distingue as classes promotor e não promotor, mesmo não havendo uma precisão na região -35 da quantidade de pares de base que estão presentes na composição dessa característica.

Como o deslocamento em uma posição à direita ou à esquerda da localização indicativa de promotor produz uma amostra considerada não promotora e também devido ao tamanho das seqüências, 50 pb cada, foi possível a partir de uma amostra promotora a geração de várias amostras não promotoras. Por exemplo, a partir de cada seqüência promotora referente ao conjunto de dados do artigo, utilizando duas janelas de 6 pb, foi possível produzir 19 amostras negativas. Para compensar a grande

quantidade de amostras negativas produzidas, as amostras representativas de seqüências promotoras foram replicadas, até a quantidade de negativas e positivas ficar balanceada.

Assim foram gerados 3 conjuntos de dados, os quais foram estendidos pela aplicação de janelas:

- Ao conjunto de 32 amostras foram aplicadas janelas de 6 pb e 6 pb, 6 pb e 10pb e 6 pb e 15 pb, sobre as regiões -10 e -35, respectivamente. E produzidos 3 novos subconjuntos;
- Ao conjunto de 21 amostras foram aplicadas janelas de 6 pb e 10pb e 6 pb e 15 pb sobre as regiões -10 e -35, respectivamente. E produzidos 2 novos subconjuntos;
- Ao conjunto que é a junção dos dois anteriores com 53 amostras foram aplicadas janelas de 6 pb e 10pb e 6 pb e 15 pb, sobre as regiões -10 e -35, respectivamente. E produzidos 2 novos subconjuntos;

Cada um destes 7 subconjuntos foi submetido a treinamento de diversas redes neurais e aplicação de testes para averiguar a capacidade de classificação dos modelos.

4.2.3 Experimento

Com os subconjuntos estabelecidos efetuou-se o treinamento das redes neurais com cada um. Diante de diversos testes realizados variando-se o algoritmo de aprendizado, optou-se pelo uso de uma configuração padrão.

O algoritmo de aprendizado escolhido foi o RPROP, com 27 nós na camada escondida, taxa de aprendizado 0,01, constante de momento de 0,95, número máximo de 1000 épocas de treinamento e erro médio quadrado desejado de 10^{-6}.

Dependendo do subconjunto de dados aplicado, conforme exposto na tab. 4.1, ocorreu a seguinte variação na camada de entrada.

Tabela 4.1: Variação do número de entradas por tamanho das janelas de cada subconjunto

Janelas em pb		Tamanho da entrada
região-10	região -35	
6	6	48
6	10	64
6	15	84

A saída da rede é composta por somente 2 nós representando as classes: promotor ou não promotor.

Devido à falta de padronização nos dados do subconjunto de 21 amostras, em termos de tamanho e localização correspondente a região -35, optou-se por somente utilizar as duas janelas exatas de 6 pb sobre as regiões -10 e -35, respectivamente, somente no conjunto de 32 amostras. À junção dos dois subconjuntos, compondo o terceiro subconjunto de 53 amostras, também não foram empregadas as duas janelas de 6 pb.

4.2.4 Resultados Obtidos

Dentro dos critérios e métricas estabelecidos, as seguintes tabelas reportam os resultados obtidos.

Tabela 4.2: Resultados da aplicação de dados de teste ao modelo RPROP com o método *leave-one-out* ao Experimento I

Subconjuntos	*CC*	*SN*	*SP*
32 amostras			
6 pb e 6pb	0,42	33%	99%
6 pb e 10pb	0,29	19%	99%
6 pb e 15pb	0,20	13%	98%
21 amostras			
6 pb e 10pb	0,06	5%	97%
6 pb e 15pb	-0,11	0%	98%
53 amostras			
6 pb e 10pb	0,07	6%	97%
6 pb e 15pb	0,07	6%	97%

Tabela 4.3: Resultados da aplicação de dados de teste ao modelo RPROP com o método *Tenfold Cross Validation* ao Experimento I

Subconjuntos	*CC*	*SN*	*SP*
32 amostras			
6 pb e 6pb	0,64	63%	98%
6 pb e 10pb	0,47	41%	98%
6 pb e 15pb	0,52	47%	98%
21 amostras			
6 pb e 10pb	0,40	34%	96%
6 pb e 15pb	0,45	38%	98%
53 amostras			
6 pb e 10pb	0,45	40%	98%
6 pb e 15pb	0,31	27%	96%

Tabela 4.4: Resultados da aplicação de dados de teste ao modelo RPROP com o método treino 80% - teste 20% ao Experimento I

Subconjuntos	*CC*	*SN*	*SP*
32 amostras			
6 pb e 6pb	0,26	15%	99%
6 pb e 10pb	0,20	11%	98%
6 pb e 15pb	0,21	12%	98%
21 amostras			
6 pb e 10pb	0,11	10%	96%
6 pb e 15pb	0,05	6%	97%
53 amostras			

6 pb e 10pb	0,08	8%	96%
6 pb e 15pb	0,12	15%	92%

4.2.5 Discussão dos Resultados

Observando-se as tabelas de resultados é fácil detectar que em todos os experimentos realizados, o subconjunto de 32 amostras com as duas janelas posicionadas exatamente sobre as características de promotor e o deslocamento dessas janelas para a construção de amostras negativas obteve os melhores resultados. Embora a significância de um melhor resultado de 63% para a sensibilidade seja um tanto questionável.

Quando aplicadas somente as 21 amostras, provenientes do NCBI, foi perceptível uma queda no percentual de reconhecimento de amostras positivas, revelando não só inconsistências na composição dessas seqüências, como também um baixo número de amostras sem representatividade, o qual aparenta adicionar ruído às 32 amostras quando realizados experimentos com a junção dos dois subconjuntos.

Além disso, os resultados revelam que a extensão do número de pares de base, representantes da região -35 não contribuiu para uma melhor caracterização dessas seqüências.

Embora realizados outros experimentos, ocorrendo a alteração dos parâmetros e configuração da arquitetura das redes, bem como aplicação de outros algoritmos de treinamento, pouco foi a contribuição na obtenção de melhores resultados.

Os resultados desses experimentos conduziram à realização de uma investigação mais aprofundada na composição das seqüências do subconjunto com 32 amostras, o qual resultou no próximo experimento descrito.

4.3 Experimento II

Neste experimento é explorado o conjunto de dados relativos as 32 amostras utilizadas no Experimento I, esses dados foram escolhidos por apresentar melhores resultados e serem originários de um estudo mais acurado. A abordagem feita neste experimento explora a geração de categorias com o uso de uma metodologia neural não supervisionada.

4.3.1 Dados Utilizados

Nesse experimento, Valiati e Engel (2006) utilizaram os dados relativos aos promotores são das 32 seqüências descritas no experimento anterior.

Os dados negativos referem-se à junção de duas seqüências relativas a regiões intergênicas sem indicativos da presença de promotores, ou seja, regiões que não estão inseridas dentro das regiões codificantes e sim entre um gene e outro. Neste caso essas duas regiões referem-se ao final de dois genes que ocorrem em fitas distintas do DNA. A união das duas seqüências resultou em 4440 pares de base.

4.3.2 Pré-processamento dos Dados

A todas as seqüências que constituíram os conjuntos de dados foi empregada a codificação BIN4.

Em um primeiro momento, foram aplicadas janelas nas posições exatas de ocorrência das características promotoras, relativas a 6 pb na região -10 e 6 pb na região -35, respectivamente. Posteriormente, também se optou por aplicar janelas com 6 pb e 10 pb, respectivamente, sobre as regiões de interesse, conforme justificado no experimento anterior devido à falta de consenso na região -35.

A geração de amostras sobre os dados negativos ocorreu conforme a distribuição das regiões nas seqüências positivas, ou seja, região -10 espaçada da região -35, e deslocamento destas janelas ao longo da seqüência até seu comprimento final. Dos 4440 pb da junção das duas seqüências, foram geradas 2516 amostras negativas quando usadas 2 janelas de 6 pb com seu respectivo espaçamento, e 2220 amostras negativas quando usadas 2 janelas: uma de 6pb e outra de 10pb com seu respectivo espaçamento.

4.3.3 Experimentos

Os experimentos relatados neste item referem-se à aplicação da rede ART 1, descrita no capítulo de Redes Neurais, para a construção de agrupamentos que representassem a variabilidade dos nucleotídeos constituintes das regiões indicadas como promotoras.

Uma atenção especial foi dada ao parâmetro de vigilância (ρ), que exerce um controle do número mínimo de pares de bases constituintes dos agrupamentos que o modelo produziu. Assim, por exemplo, para a aplicação de seqüências representativas das regiões de interesse compostas por 12 pb em conjunto, com o ρ = 0,5, significa dizer que os agrupamentos formados serão constituídos por grupos compostos com pelo menos 6 pb.

Inicialmente foi empregado o conjunto das 32 amostras com suas exatas duas regiões de 6 pb na construção de um modelo. Os vários valores para o parâmetro de vigilância foram testados, sendo ρ = 0,5 o mais estável encontrado para este experimento, o que propiciou a geração de 11 agrupamentos, os quais são apresentados no APÊNDICE A, com suas respectivas saídas reconstruídas, e posterior aplicação das 2516 seqüências negativas sobre os *cluster*s encontrados.

Com base nos resultados obtidos e apresentados na próxima subseção, partiu-se para uma extensão desse experimento inicial. Desta forma, foi criado um modelo para as 32 amostras com janelas de 6 pb e 10 pb, respectivamente. A melhor configuração também foi obtida quando ρ = 0,5, o que possibilitou a geração de 14 agrupamentos, os quais são apresentados no APÊNDICE B, com suas respectivas saídas reconstruídas, e posterior aplicação das 2220 seqüências negativas sobre os *cluster*s encontrados.

Os resultados mais significativos foram obtidos para o segundo experimento de 6 pb e 10 pb, respectivamente para cada região. Desta forma, optou-se por realizar uma nova extensão do experimento com melhores resultados, onde, cada uma das janelas de 6 pb e 10 pb, respectivamente, foi utilizada para o treinamento de um modelo ART 1, correspondente a cada uma das duas regiões em separado.

Foram construídos vários modelos de agrupamentos, primeiro com o ρ variando em [0,15; 0,2; 0,4; 0,6; 0,7; 0,9] para o ART 1 destinado à região -10; e depois com o ρ variando em [0,1; 0,2; 0,3; 0,4; 0,5; 0,6; 0,7; 0,8] para o ART 1 da região -35. Uma vez construídos os modelos foi recuperado o valor de ρ para cada região em que se obtiveram os melhores resultados da validação cruzada com *leave-one-out*. Cada amostra reservada para teste, teve suas respectivas regiões submetidas ao referido

modelo. O processo decisório tratou de unir a classificação dos dois modelos e somente a classificação em ambos permitiu se estabelecer o agrupamento a que pertencia a amostra de teste.

Após a recuperação do ρ para cada modelo referente à determinada região, foi realizada a submissão das 2220 amostras negativas, cada região da amostra foi submetida a seu respectivo modelo e a classificação estabelecida segundo a associação do cumprimento da meta de ambos. As amostras negativas não foram utilizadas para geração dos agrupamentos, no entanto a avaliação de seus resultados contribuiu para ajustar o parâmetro ρ de ambos os modelos para um valor que balanceasse a classificação de amostras positivas e negativas a valores supostamente satisfatórios.

Assim, para o experimento que empregou um modelo ART 1 para cada região, foi estabelecido um $\rho = 0,7$ e obtenção de 10 agrupamentos para a região -10 e um $\rho = 0,3$ e obtenção de 12 agrupamentos para a região -35. Nos APÊNDICES C e D, respectivamente, são apresentados os agrupamentos gerados e as saídas reconstruídas para cada modelo.

4.3.4 Resultados Obtidos

A tab. 4.5 sumariza os resultados obtidos para os subconjuntos que utilizaram duas janelas associadas para obtenção dos agrupamentos e para o subconjunto que utilizou cada região em separado para a construção de modelos. Os resultados relativos à sensibilidade (SN) fazem menção à classificação obtida com o método *leave-one-out* sobre as amostras positivas, enquanto que os resultados relativos à especificidade (SP) representam a capacidade de rejeição das amostras negativas. O valor da sensibilidade para o subconjunto com as regiões separadas foi obtido pela média do *leave-one-out* para as duas regiões.

Tabela 4.5: Resultado da aplicação de 3 subconjuntos a agrupamentos com obtenção dos melhores parâmetros de vigilância

Subconjuntos	ρ		***CC***	***SN***	***SP***
6 pb e 6 pb juntos	0,5		0,43	53%	99%
6 pb e 10 pb juntos	0,5		0,75	65%	99%
	região -10	região -35			
6 pb e 10 pb separados	0,7	0,3	0,34	72%	95%

4.3.5 Discussão dos Resultados

Os resultados revelaram que a separação das regiões -10 e -35 para composição de modelos distintos produziu resultados melhores na identificação de promotores, apesar de ter obtido um coeficiente de correlação baixo e uma pequena queda na identificação de falsos positivos.

Embora, novamente as amostras referentes às 32 seqüências sejam relativamente representativas, obtendo-se bons resultados para a aplicação de 6 pb e 10 pb para as regiões -10 e -35, respectivamente, esses dados ainda são insuficientes em quantidade de amostras para poderem ser generalizados a todo o problema e serem capazes de descobrir e indicar possíveis promotores em uma seqüência desconhecida submetida ao modelo. Os indícios levam a crer, que quando submetidas amostras positivas um pouco

diferentes em conteúdo das que produziram os agrupamentos, conforme constatado em alguns testes preliminares, ocorreu a rejeição das mesmas. Sendo que o modelo não foi capaz de generalizar as positivas.

Considerando os apêndices, que apresentam o número de categorias geradas para as melhores configurações do parâmetro de vigilância aos subconjuntos aplicados, constatamos que a média da quantidade de agrupamentos por seqüências é de dois ou menos, dependendo da situação. Ou seja, não há uma representatividade dessas 32 seqüências para se obter uma boa padronização que colabore na identificação de novas seqüências promotoras.

Dessa forma, se buscou uma nova alternativa para tentar obter um número maior de seqüências que pudessem ser investigadas. Essa busca por mais seqüências resultou nos próximos experimentos que exploram uma nova abordagem, mas que também tentam explorar a metodologia empregada pelo ART 1.

4.4 Experimento III

Este experimento aborda o uso de regiões que contêm ou não um promotor inserido em suas seqüências sem saber a localização deste na seqüência. São descritos os organismos selecionados, a forma de obtenção das regiões de interesse e o processo de composição das amostras para consolidar um conjunto de treinamento de modelos neurais supervisionados. Também é exposta a realização de testes com os dados empregados neste experimento aos modelos obtidos com a abordagem não supervsionada do Experimento II.

4.4.1 Dados Utilizados

Na tentativa de suprir a deficiência da falta de dados relativos a promotores de *Mycoplasmas***,** optou-se pela exploração de duas regiões intergênicas de organismos da família *Mycoplasmataceae*. As duas regiões escolhidas se caracterizam: pela presença de dois promotores sem se saber sua localização de ocorrência, IGR-*B*, e pela ausência de promotor, IGR-*X*. Para tal experimento foram selecionados 12 genomas dos referidos organismos: *M. galliceticum*, *M. genitallium*, *M. hyopneumoniae J*, *M. hyopneumoniae 232*, *M. hyopneomoniae 7448*, *M. móbile*, *M. mycoides*, *M. penetrans*, *M. pneumoniae*, *M. pulmoniae*, *M. synoviae* e *Ureaplasma* (parente próximo dos *Mycoplasmas*) expostos em Vasconcelos et al. (2005).

4.4.1.1 Regiões Intergênicas

Muitas das diferenças entre espécies podem ser atribuídas a mudanças nos processos de transcrição e tradução, os quais são comandados por elementos que estão dispostos nas regiões intergênicas. Essas regiões são definidas como as seqüências que estão entre o ponto de parada da transcrição de um gene e o ponto de início da transcrição do próximo gene.

Segundo Camargo et al. (2005), autor da ferramenta IGR-Annot que usa a metodologia de Sistemas Multi-agentes para aquisição das regiões intergênicas, para a obtenção dessas regiões do genoma de um determinado organismo é necessário possuir o genoma completo do organismo e informações sobre suas regiões codificantes, compostas pelas posições de início e término do gene, a sua orientação e o seu nome. A ferramenta trabalha com a exploração de arquivos no formato GenBank, os quais

contêm todas as informações requisitadas e é considerado um formato padrão, sendo que todos os genomas depositados neste banco possuem essa formatação.

Cada arquivo com a seqüência de nucleotídeos de um organismo, em formato GenBank, fornece uma série de itens sob forma tabular que descrevem as características do genoma. Uma das partes de interesse desse arquivo, contém as informações relativas aos genes que compõem o organismo, geralmente os pontos de início e término do gene são indicados num campo referenciado como *CDS*, que também acompanham informações relativas ao seu produto e nome. Outras duas informações importantes a serem consideradas são: a fita em que o gene está presente, fita senso 5′->3′ (*foward*) ou fita anti-senso 3′->5′ (*reverse*); e a posição relativa entre genes adjacentes, onde dois genes adjacentes podem estar sobrepostos ou não. A existência de sobreposição indica que não há região intergênica.

Assim as regiões intergênicas são detectadas e nomeadas conforme a seguinte convenção:

IGR-*O*-*G1*-*G2*,

onde $O = \{F|R|X|B\}$, dependendo da orientação do gene anterior (*G1*) e posterior (*G2*). Conforme mostra a fig. 4.1, quatro situações distintas da aplicação da nomenclatura podem ocorrer:

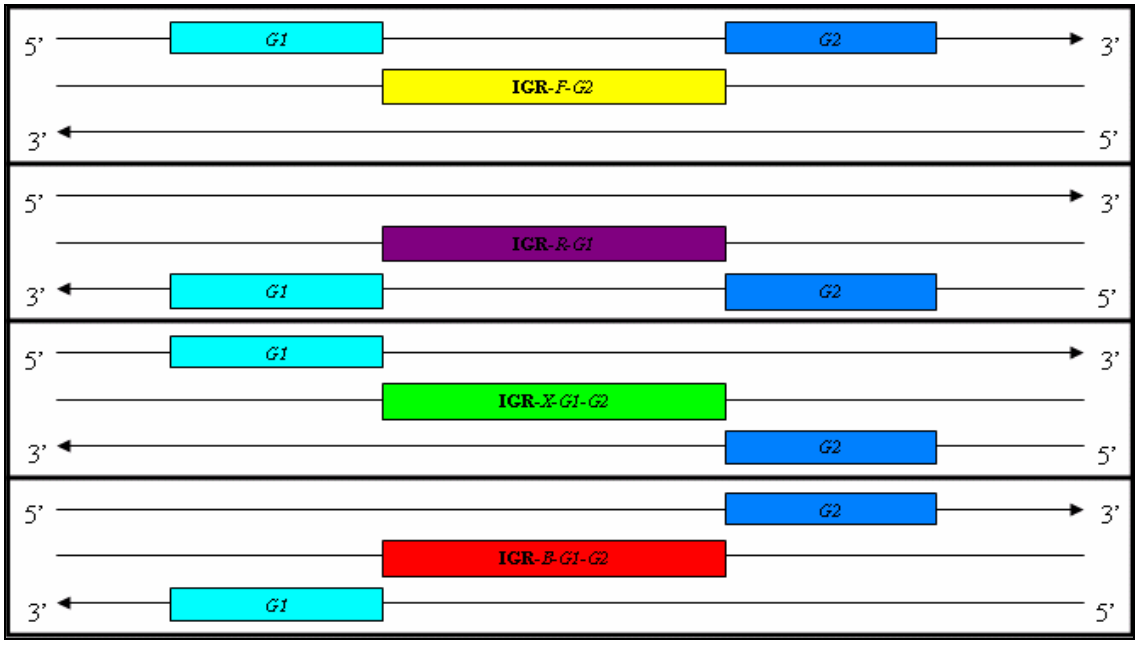


Figura 4.1: As quatro regiões intergênicas possíveis (CAMARGO et al., 2005)

- Se *G1* e *G2* estão na fita senso então a região intergênica é nomeada **IGR**-*F*-*G2*, porque é uma região intergênica que contém informação regulatória sobre o gene *G2*;
- Se *G1* e *G2* estão na fita anti-senso então a região intergênica é nomeada **IGR**-*R*-*G1*, porque é uma região intergênica que contém informação regulatória sobre o gene *G1*;
- Se *G1* está na fita senso e *G2* está na fita anti-senso então a região intergênica é nomeada **IGR**-*X*-*G1*-*G2*, porque é uma região intergênica que não contém informação regulatória de nenhum dos genes;

- Se *G1* está na fita anti-senso e *G2* está na fita senso então a região intergênica é nomeada **IGR**-*B*-*G1*-*G2*, porque é uma região intergênica que contém informação regulatória sobre ambos os genes.

Ao final da detecção de todas as regiões intergênicas do genoma de um organismo, a ferramenta IGR-Annot grava as informações em um novo arquivo, de mesmo formato GenBank, adicionando um campo chamado *misc_feature* que contém as informações de cada região intergênica (CAMARGO et al., 2005).

4.4.2 Pré-processamento dos Dados

Partindo dos arquivos com as regiões intergênicas dos 12 organismos, foi realizada a extração das regiões de interesse IGR-*X* e IGR-*B*, com o auxílio do software Artemis 7 que permite a manipulação das informações anotadas em arquivos GenBank. Assim, quando realizada uma busca no arquivo anotado de cada organismo, foram selecionadas somente as seqüências que apresentassem no campo *misc_feature* a expressão "IGR-*X*". O resultado dessa seleção foi gravado em arquivo formato FASTA[12] com nome representativo, a esta característica e ao organismo, para processamento posterior. O mesmo foi realizado para a obtenção das seqüências IGR-*B*.

Com base nas informações da subseção anterior, tanto as regiões IGR-*B* como IGR-*X* se referem à presença e ausência de promotores, respectivamente, em ambas as fitas. Como a ferramenta IGR-Annot realiza a anotação das informações somente da fita senso, um pré-processamento adicional foi realizado: a geração do complemento reverso de cada seqüência que compõe os arquivos com as regiões intergênicas selecionadas. Consequentemente, cada arquivo teve o número de seqüências duplicado em virtude dessa característica. A realização da inversão e complementação das seqüências dessas regiões foi realizada com o emprego da rotina Revseq, que compõe o pacote EMBOSS de ferramentas projetadas para Bioinformática.

Segundo consulta a um especialista do domínio, a maior parte das seqüências deveria apresentar o promotor distante do início do gene não mais do que 100 pb. Essa informação foi utilizada para limitar o tamanho das seqüências que compunham cada arquivo de regiões IGR-*X* e IGR-*B* a 100 pb. No sentido de padronizar as seqüências, foi estabelecido um corte dos nucleotídeos que excedessem os 100 pb e eliminação do conjunto de dados de seqüências com número inferior a 100 pb.

Após a realização dessa limitação no tamanho das seqüências foram extraídas as seqüências de nucleotídeos de cada arquivo, ignorando o cabeçalho informativo. Posteriormente, todos os arquivos IGR-*X* e IGR-*B* extraídos e pré-processados, conforme relatado acima, de cada um dos 12 organismos, foram unidos em um único arquivo relativo a cada região de interesse, o que resultou na composição de dois arquivos referentes às características exploradas.

A quantidade de seqüências obtidas relativas às regiões de cada organismo, assim como o número total de seqüências, é apresentado na tab. 4.6. Sobre cada um dos arquivos com as informações IGR-*X* (782 seqüências) e IGR-*B* (1318 seqüências) consolidado, foi realizada a codificação binária, sendo cada seqüência representada por

[12] Formato composto por um cabeçalho identificador seguido pela correspondente seqüência de nucleotídeos ou aminoácidos com 60 elementos por linha

um vetor de 400 elementos e adicionados dois elementos ao final de cada amostra relativos à saída desejada, conforme especificação da subseção 4.1.1.

Tabela 4.6: Quantidade de seqüências obtidas nos 12 organismos para regiões IGR-*X* e IGR-*B* já com a inserção do complemento reverso

Organismo	**IGR-*X***	**IGR-*B***
M. galliceticum	54	72
M. genitallium	12	38
M. hyopneumoniae J	82	130
M. hyopneumoniae 232	86	122
M. hyopneomoniae 7448	88	122
M. móbile	28	108
M. mycoides	104	174
M. penetrans	96	182
M. pneumoniae	40	56
M. pulmoniae	92	122
M. synoviae	56	92
Ureaplasma	44	100
Total de seqüências	782	1318

4.4.3 Experimentos

Inicialmente, o conjunto total de amostras obtido de cada região foi aplicado às RNs com arquitetura MLP e algoritmos de aprendizado GDX e RPROP. Dentre vários treinamentos realizados, se optou por 4 testes com o *Tenfold Cross Validation*, 2 com menor número de neurônios na camada escondida e 2 com maior número de neurônios na camada escondida para cada algoritmo de aprendizado. Para os testes com percentuais de dados para treino e teste se optou pela realização de 6 testes variando a quantidade de neurônios na camada escondida: 11, 20 e 38, respectivamente, para cada algoritmo de aprendizado.

A variação na quantidade de neurônios da camada escondida foi realizada para ver a sua influência na capacidade de classificação dos modelos, uma maior quantidade de neurônios nesta camada não contribuiu em se obter melhores resultados. Assim como a adição de mais camadas escondidas.

Os demais parâmetros das redes treinadas seguiram a seguinte configuração: taxa de aprendizado de 0,01, constante de momento de 0,95, número máximo de 1000 épocas de treinamento e erro médio quadrado desejado de 10^{-6}. Não foram identificados melhores desempenhos variando esses parâmetros, desta forma eles foram mantidos.

Os resultados apresentados no item seguinte, relativos ao teste com o *Tenfold Cross Validation* representam o resultado médio para a apresentação dos 10 subconjuntos, já os resultados relativos aos experimentos de treino e teste apresentam a média para as 5 execuções.

Além dos experimentos com redes de aprendizado supervisionado, foi realizada uma tentativa de submissão à rede ART 1 treinada com as 32 amostras, com janelas de 6 pb e 10 pb separadas, com seus parâmetros ρ configurados em 0,7 e 0,3 respectivamente, do experimento relatado na seção anterior, por ter encontrado o melhor resultado na avaliação da sensibilidade. Isso foi realizado porque as 32 amostras são comprovadas, sendo que para as regiões IGR-*B* somente sabemos que existe pelo menos um promotor inserido e nas IGR-*X* sabemos que não há promotor.

Na realização deste experimento, foram posicionadas as janelas de 6 pb e 10 pb, espaçadas pelo número padrão de pares de base estabelecidos. Sobre cada uma das seqüências dos arquivos com informações IGR-*X* e IGR-*B* de todos os organismos, e então deslocadas de seu início até o final de cada seqüência. Esse deslocamento produziu 70 novas amostras por cada seqüência, o que totalizou ao final um conjunto de 92400 amostras para a região intergênica IGR-*B* e 54880 amostras para a região intergênica IGR-*X*.

As amostras positivas (IGR-*B*) foram propagadas pelo modelo treinado e ao final foi fornecida uma lista de quais posições e sub-seqüências, dentro de cada uma das 1318 amostras, eram indicativas da presença de promotor. O mesmo procedimento foi realizado com as amostras IGR-*X* e ao final, fornecida uma lista com falsos promotores, uma vez que as seqüências sendo negativas não deveriam apresentar promotores.

Outro teste, com ART 1 foi a geração de agrupamentos com a submissão das 1318 seqüências referentes às regiões IGR-*B*, no sentido de tentar encontrar alguma regularidade na composição dessas seqüências. As seqüências foram utilizadas na íntegra, ou seja, com seus 100 pb originais. Na seção de resultados é apresentado o número de agrupamentos obtidos com os respectivos parâmetros ρ.

Em um último experimento com o ART 1, foram gerados agrupamentos utilizando as seqüências relativas à região intergênica IGR-*B* para cada organismo. Isto foi realizado em decorrência dos resultados obtidos no experimento anterior, visando a possibilidade de existir alguma regularidade na composição dos pares de base, relativos a cada organismo individualmente, com o intuito que cada um pudesse apresentar particularidades em sua composição nucleotídica nessas regiões.

4.4.4 Resultados Obtidos

4.4.4.1 RPROP e GDX

A tab. 4.7 apresenta os resultados dos experimentos realizados com os dados das regiões IGR-*B* e IGR-*X*, conforme as métricas de avaliação e algoritmos de aprendizado expostos. A tabela também apresenta a quantidade de neurônios na camada escondida de cada rede neural e a quantidade de amostras promotoras (***P***) e não promotoras (***NP***) utilizadas para teste. O número de amostras de teste utilizadas nos experimentos onde a métrica de avaliação foi o percentual de treino e teste, se refere a 5 execuções em que foram utilizados 20% (informação especificada junto com a métrica de avaliação) das amostras para teste e 80 % destinadas para treino em cada execução.

Tabela 4.7: Resultados da aplicação de algoritmos de aprendizado sobre métricas de avaliação das regiões IGR-*B* e IGR-*X*

Tipo de teste	**Algoritmo de aprendizado**	**Neurônios na camada escondida**	***P/NP***	***CC***	***SN***	***SP***
Tenfold Cross Validation	RPROP	13	1318/782	0,25	75%	50%
Tenfold Cross Validation	GDX	13	1318/782	0,38	77%	61%
Tenfold Cross Validation	RPROP	35	1318/782	0,37	82%	53%
Tenfold Cross Validation	GDX	35	1318/782	0,39	80%	59%
Treino e teste (20%)	RP	11	1280/825	0,34	76%	58%
Treino e teste (20%)	GDX	11	1280/825	0,40	78%	62%
Treino e teste (20%)	RP	20	1280/825	0,39	75%	63%
Treino e teste (20%)	GDX	20	1280/825	0,45	81%	63%
Treino e teste (20%)	RP	38	1280/825	0,39	79%	59%
Treino e teste (20%)	GDX	38	1280/825	0,48	83%	63%

Os resultados demonstram que o algoritmo de aprendizado GDX com maior quantidade de neurônios na camada escondida apresentou com maior freqüência os melhores resultados em todas as métricas analisadas. Destacam-se, o modelo GDX com 35 nós no teste *Tenfold Cross Validation* e os modelos GDX com 20 e 38 nós, respectivamente, no treino e teste.

É importante salientar que menores índices encontrados para a especificidade em relação à sensibilidade, provavelmente sejam decorrentes da quantidade inferior de amostras negativas.

4.4.4.2 ART 1com 32 seqüências comprovadas

No uso do ART 1, treinado com as 32 amostras e submissão das 92400 amostras geradas foram encontrados 5788 prováveis promotores. Nenhum provável promotor foi encontrado em 33 das 1318 seqüências, que deram origem ao conjunto submetido. Essa clusterização apresentou uma variação de ocorrência de 0-14 promotores por seqüência, ou seja, algumas seqüências não apresentaram nenhum promotor ao longo das 70 janelas, enquanto outras apresentaram, por exemplo, 5 promotores e outras até 14 promotores. O gráfico da fig. 4.2 apresenta o número de amostras encontradas para a freqüência de possíveis aparições na variação resultante. As ocorrências, em cada seqüência, de 2, 3, 4, 5 e 6 supostos promotores, respectivamente concentraram aproximadamente 66% de toda clusterização. Embora tenham sido detectadas mais de 10 ocorrências em uma mesma seqüência, estas aconteceram em menor número.

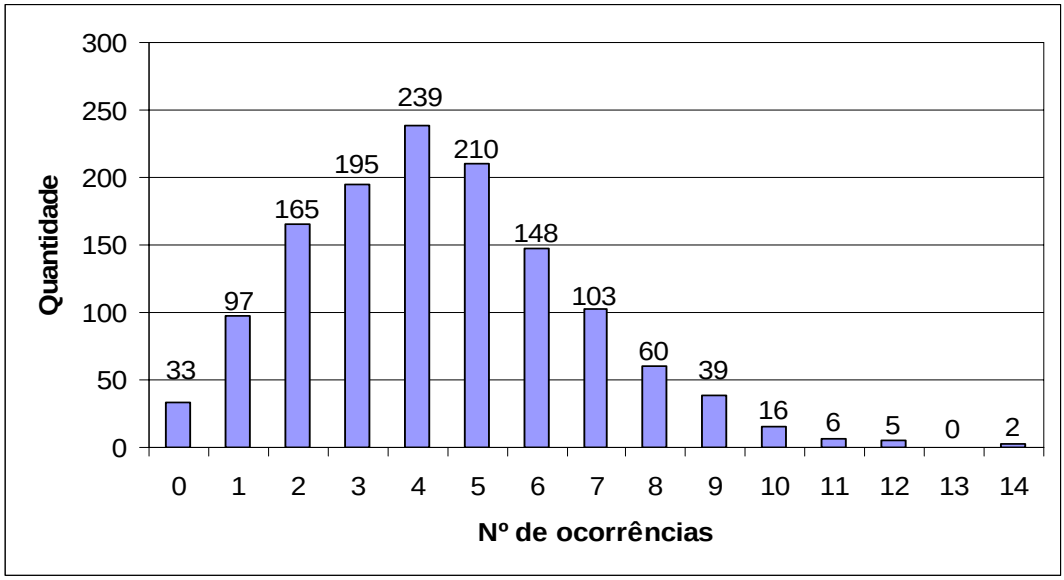


Figura 4.2: Gráfico com a quantidade de seqüências e a variação de ocorrência de possíveis promotores em seqüências IGR-*B*

Nos APÊNCICES E e F são apresentadas as possíveis regiões promotoras com seu conteúdo e exata localização, juntamente com a seqüência original de 100 pb, em que são sugeridos promotores uma e duas vezes, ou seja, em 97 e 165 seqüências, respectivamente.

Quanto à submissão das 54880 amostras, originárias das 782 seqüências IGR-*X*, ao modelo ART 1, referentes aos 32 promotores comprovados, os resultados revelaram que foram encontrados 2621 prováveis promotores. Nenhum promotor foi encontrado para 34 das 782 seqüências. A fig. 4.3 apresenta o gráfico com a distribuição da quantidade de amostras em cada possível número de ocorrências em uma seqüência. As altas concentrações, que sugerem supostos promotores, ocorreram entre 1 e 5 aparições, sendo que foram encontradas até 9, 10 e 11 aparições, mas com baixa incidência.

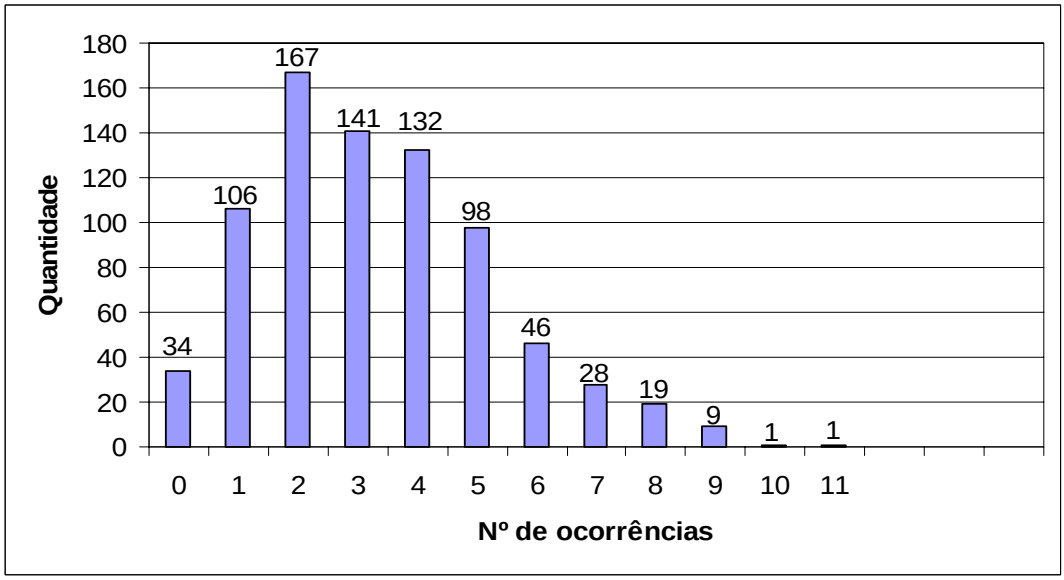


Figura 4.3: Gráfico com a quantidade de seqüências e a variação de ocorrência de possíveis promotores em seqüências IGR-*X*

4.4.4.3 ART 1 na obtenção de agrupamentos com as regiões IGR-B

Os resultados relativos da aplicação do modelo ART 1 às 1318 seqüências que compõem o conjunto IGR-*B* são apresentadas na tab. 4.8. Nas várias tentativas de se determinar um melhor parâmetro de vigilância (ρ) se esbarrou na grande variabilidade da composição nucleotídica presente ao longo dos 100 pb.

Tabela 4.8: Parâmetro de vigilância e o número de agrupamentos obtido para as regiões IGR-*B*

Parâmetro de vigilância (ρ)	**Nº de agrupamentos obtidos**
0,12	331
0,10	295
0,08	250
0,06	168

Lembrando, que o parâmetro ρ está relacionado com o número mínimo de pares de base que devem ser recuperados. O parâmetro ρ em 0,06 significa dizer que serão considerados pelo menos 6 pb nos 100 pb. Uma elevação do ρ, demonstrou a geração de um maior número de agrupamentos, por não haver uma similaridade adequada nas seqüências tratadas e da forma como estão dispostas no conjunto de amostras.

Na remontagem dos agrupamentos obtidos, somente para o parâmetro que obteve o menor número de agrupamentos, se constatou uma grande dispersão na ocorrência dos 6 pb recuperados. Adeninas, timinas, guaninas e citosinas foram encontradas isoladas ao longo dos 100 pb, apresentando na melhor das hipóteses duas bases uma ao lado da outra.

4.4.4.4 ART 1 na obtenção de agrupamentos com as regiões IGR-B para cada um dos 12 organismos

Na obtenção desses resultados foram aplicadas diferentes configurações para o parâmetro ρ, até se estabelecer um valor padrão que se mostrou mais conciso diante dos resultados obtidos. A tab. 4.9 apresenta a quantidade de seqüências utilizadas e o número de agrupamentos encontrados para cada organismo, com o $\rho = 0,1$.

Tabela 4.9: Número de agrupamentos encontrados para cada organismo em suas regiões IGR-*B* com parâmetro ρ estabelecido em 0,1

Organismo	**Nº de seqüências IGR-*B***	**Nº de agrupamentos encontrados**
M. galliceticum	72	21
M. genitallium	38	13
M. hyopneumoniae J	130	36
M. hyopneumoniae 232	122	32
M. hyopneomoniae 7748	122	32
M. móbile	108	29

M. mycoides	174	39
M. penetrans	182	51
M. pneumoniae	56	19
M. pulmoniae	122	33
M. synoviae	92	26
Ureaplasma	100	28

Em média cada agrupamento representou 3,6 seqüências. Isso demonstra a baixa capacidade de generalização propiciada pelo modelo, diante das seqüências IGR-*B* individual dos organismos.

4.4.5 Discussão dos Resultados

Neste experimento, foram tratadas duas abordagens: a capacidade do modelo supervisionado em distinguir seqüências que contém promotor (IGR-*B*), de seqüências que não contém promotor (IGR-*X*) e a capacidade de um modelo não supervisionado em encontrar uma ou mais supostas localizações e conteúdo de promotores, em uma seqüência IGR-*B* de 100pb, com base em seqüências comprovadas da presença de promotor (32 seqüências). Além disso, foi realizada uma extensão dos experimentos primários, na tentativa de encontrar agrupamentos mais relevantes em relação a possíveis promotores, a qual apresentou quase que total ineficiência, diante dos resultados obtidos em virtude da variabilidade da informação.

Embora a indicação de prováveis promotores, com suposta localização e conteúdo em uma seqüência, conforme apresentado nos apêndices E e F pareça ser útil e visualmente palpável, sua representatividade é duvidosa. Diante de mais de 1300 seqüências que apresentavam promotores, basear-se somente em 32 seqüências comprovadas, o que representa ~2,4% de supostos promotores, para referenciar um modelo genérico é uma suposição extremamente perigosa. No entanto, serve como uma metodologia futura de aplicação, a partir do momento que estudos biológicos determinarem um volume confiável de seqüências promotoras para os *Mycoplasmas*.

Mediante os resultados obtidos, os experimentos aplicando o modelo neural ART 1 foram abandonados e as investigações foram direcionadas no sentido de encontrar regularidades relevantes nos dados referentes às regiões intergênicas, que apresentaram resultados mais interessantes.

Assim o próximo experimento conduz à aplicação da ferramenta BLAST (*Basic Local Alignment Search Tool*) para alinhamento de seqüências e de um algoritmo para identificação de regras de associação nas regiões intergênicas IGR-*B* na tentativa de encontrar possíveis regularidades nas seqüências que pudessem representar a informação promotora.

4.5 Experimento IV

Este experimento trata do emprego de uma técnica de alinhamento, amplamente utilizada em Bioinformática, para tentar encontrar regularidades nas regiões intergênicas caracterizadas por apresentar promotor utilizadas no Experimento III. Em conjunto a ela, são aplicados filtros para refinamento da informação de interesse o que conduz ao

uso de um algoritmo de Mineração de Dados para encontrar relação nos resultados da filtragem do alinhamento.

4.5.1 Dados Utilizados

Os dados utilizados nesse experimento se referem às regiões intergênicas IGR-*B* descritas no experimento anterior, entretanto todas as seqüências independem do tamanho, ou seja, há seqüências com menos de 100 pb e que estão presentes somente na fita senso, não sendo consideradas as amostras com o complemento reverso.

4.5.2 Pré-processamento dos Dados

As regiões intergênicas IGR-*B* selecionadas totalizaram 854 seqüências, que num primeiro momento foram submetidas ao programa *formatdb* (rotina que acompanha o pacote do BLAST) que realiza uma listagem catalogada das seqüências envolvidas e então cria o banco de dados; e posterior aplicação ao programa BLAST (descrito na próxima subseção).

Os parâmetros para a execução do BLAST são vários, dentre eles se optou por uma configuração que produzisse resultados direcionados às características desejadas como a busca por trechos, palavras, com no mínimo 7 pb, devido ao suposto tamanho de trechos regulares representativos, análogo aos que compõem as regiões promotoras características - 10 e -35. Optou-se pelo uso da rotina *blastn*, por se tratarem de seqüências nucleotídicas. O valor de expectativa foi determinado em 1e-10, esse valor se mostrou adequado, valores mais baixos produziram resultados muito amplos e de difícil interpretação. O formato de saída escolhido para os resultados do BLAST foi o tabular (designado m8), que retorna 12 itens: a identificação da seqüência de consulta, a identificação da seqüência do banco de dados, a medida de identidade encontrada no alinhamento das duas seqüências, o número de *matchs*, o número de identidades, o número de espaços (*gaps*), a posição inicial e final da seqüência de consulta onde ocorreu o alinhamento, a posição inicial e final da seqüência do banco de dados onde ocorreu o alinhamento, o valor da expectativa (*e-value*) e o valor de pontuação (*score*) do alinhamento.

Desta forma as 854 seqüências foram alinhadas todas contra todas, aos pares, pelo programa BLAST, gerando uma tabela de resultados com os itens descritos totalizando mais de 725 mil registros. Devido à grande quantidade de registros encontrados pelo BLAST, foram aplicados filtros determinados inicialmente por valores médios. Assim, pelo fato de se considerar o *score* um dos campos mais importantes da tabela foi obtido a sua média, além de obter o tamanho médio dos alinhamentos, com base na diferença entre as posições iniciais e finais das seqüências envolvidas. Apesar da especificação de busca por palavras de tamanho 7 pb, não foi retornado nenhum registro com esse tamanho mediante os demais parâmetros requisitados. O valor médio encontrado para o *score* foi de 25,5 e o tamanho médio das seqüências foi de 13 pb.

Todos os registros que apresentaram um valor de *score* superior à média foram selecionados, encontrando-se 41150 registros, o corte pelo tamanho não foi necessário, se deu automaticamente pelo *score* médio encontrado. Entretanto, o número de registros resultantes ainda foi considerado excessivo, a aplicação de um novo filtro foi baseada na definição do percentual de identidade em 100%, o que implica na inexistência de *gaps* e no número de identidades, obtendo-se assim 1955 registros.

Sobre esses 1955 registros foi feita uma varredura para encontrar alinhamentos de uma seqüência de consulta com uma seqüência do banco de dados, e a situação inversa, ou seja, em determinado momento o conteúdo identificador da seqüência tido como pertencente ao banco de dados passou a ser o conteúdo identificador da seqüência de consulta e o conteúdo identificador da seqüência de consulta passou a ser o conteúdo identificador da seqüência do banco de dados, isto implicava em se ter dois registros com o mesmo resultado. Assim se eliminou essa redundância. Também foram eliminados registros em que a seqüência de consulta alinhava com determinado trecho da seqüência do banco de dados e sucessivamente apresentava demais alinhamentos com deslocamentos de um par de base com a mesma seqüência. Com a aplicação dessa filtragem foram obtidos 272 registros.

Como nosso interesse estava em descobrir a relação do alinhamento das seqüências de consulta com as seqüências do banco de dados, foram eliminados os demais campos da tabela, restando apenas os campos identificadores das seqüências. Sobre a identificação das seqüências relativas ao banco de dados, foram encontradas 84 ocorrências de identificação única, essa listagem é apresentada no APÊNDICE G. Desta forma, foi montada uma nova tabela, com a relação dos 272 registros contra as 84 ocorrências para se observar com quantas e quais seqüências do banco de dados, as seqüências de consulta apresentavam relação. Devido ao fato de os identificadores de consulta ocorrerem mais de uma vez, por apresentar relação com mais de uma seqüência, ou com demais trechos de várias seqüências do banco de dados, foi possível condensar os registros correspondentes ao mesmo identificador da seqüência de consulta em um só registro e fazer referência a que identificadores de seqüência do banco de dados estes apresentavam correspondência. Ao final restaram 70 registros.

A alternativa para explorar a forma como os registros remanescentes se relacionam foi a aplicação de uma técnica de Mineração de Dados, para encontrar Regras de Associação, conhecida como algoritmo *Apriori*. Esta técnica busca encontrar associação entre ocorrências que apresentam alguma relação e é amplamente utilizada para solucionar problemas em que é difícil a compreensão dos relacionamentos existentes.

4.5.2.1 BLAST

BLAST (ALTSCHUL et al., 1990) é o pacote de programas que realiza comparações entre pares de seqüências, procurando por regiões com similaridades locais. A popularidade do BLAST é conseqüência do seu funcionamento otimizado, que privilegia a eficiência computacional e determina da melhor forma possível uma medida de similaridade específica, a ponto de seus usuários criarem o verbo "*blastar*" (GIBAS e JAMBECK, 2001).

A função do BLAST é produzir uma lista com as mais altas pontuações (*score*) a partir de trechos de seqüências que não apresentam espaços em branco (*gaps*) entre a seqüência de consulta e as seqüências no banco de dados. Para encontrar as pontuações mais elevadas são determinadas certas "sementes", que são trechos muito curtos entre a seqüência de consulta e uma seqüência do banco de dados. Estas sementes são então estendidas em ambas as direções, sem incluir *gaps*, até que a maior pontuação possível para a extensão de uma determinada semente seja alcançada (SETUBAL e MEIDANIS, 1997).

Conforme apresentado na seção anterior, o BLAST possui uma série de parâmetros que estabelecem sua execução. Um desses parâmetros trata da rotina a ser utilizada. Conforme o formato das seqüências que se deseja alinhar, existem 5 rotinas diferentes: *blastn*, que compara nucleotídeos com nucleotídeos (e foi utilizada nesse experimento); *blastx*, que compara nucleotídeos traduzidos com aminoácidos; *blastp*, que compara aminoácidos com aminoácidos, *Tblastn*, que compara aminoácidos com nucleotídeos traduzidos; e *Tblastx*, que compara nucleotídeos traduzidos com nucleotídeos traduzidos. Outro parâmetro de interesse é o *e-value*, valor estatístico da probabilidade de o alinhamento acontecer por acaso, valores menores que 1e-5 são desejados, valores superiores a este limite são considerados pouco significativos estatisticamente. Quanto aos demais parâmetros para execução estão definições: das seqüências de entrada e saída envolvidas; especificação de um tamanho de palavra; determinação da orientação das seqüências envolvidas, além de outros parâmetros pré-definidos (*default*) mas que podem ser alterados.

Dentre as variáveis apresentadas nos resultados dos alinhamentos destacam-se: *score*, medida de pontuação dos alinhamentos; o percentual de identidade, relação entre o número de *matchs* (tamanho do alinhamento) com a existência ou não de *gaps*; e o valor de expectativa (*e-value*).

4.5.3 Experimento

Os 70 registros que indicam a relação entre as seqüências alinhadas foram submetidos ao algoritmo *Apriori* (descrito na próxima subseção), destinado à obtenção de regras de associação. No nosso caso, como desejamos obter somente os grupos de itens freqüentes foi necessária apenas a execução da primeira parte desse algoritmo sem a necessidade de se produzir regras associativas, para se obter esses grupos.

Para a execução do algoritmo foi definido um suporte mínimo que garante ser encontrado pelo menos dois itens para cada registro. O gráfico da fig. 4.4 apresenta a quantidade de itens freqüentes nos grupos encontrados. Cada grupo, em ordem crescente, indica a quantidade de seqüências do banco de dados que estão relacionadas aos pares, trincas e assim sucessivamente até não encontrar mais relação possível entre as seqüências.

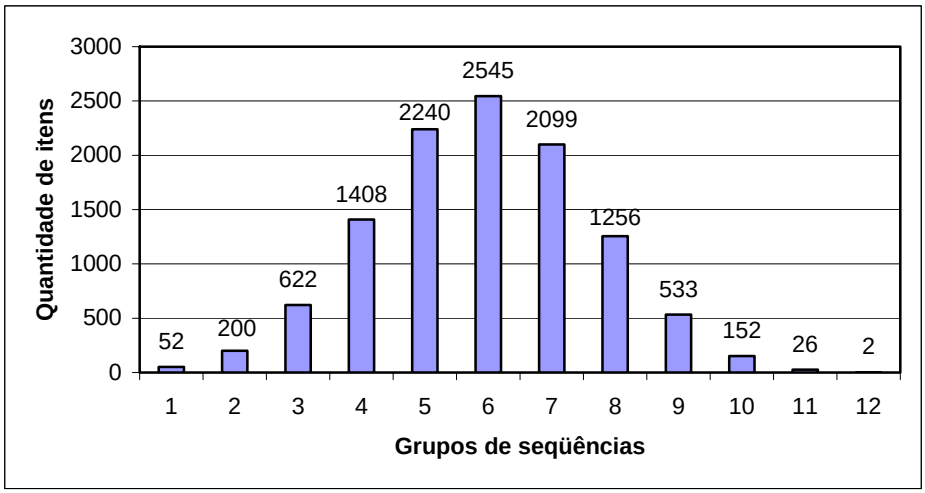


Figura 4.4: Gráfico com a quantidade de itens encontrados para cada grupo de seqüências

Os itens freqüentes encontrados a partir do segundo grupo até o último grupo representam sempre uma combinação dos itens freqüentes do grupo de ordem menor que atingiu a sua meta determinada pelo suporte mínimo.

Todos os itens freqüentes de um grupo são combinações dos itens freqüentes do grupo de ordem menor que a dele, caso este não seja o primeiro. Entretanto, nem todos os itens que compõem um grupo necessariamente produzem novas combinações no grupo subseqüente, com isto são obtidos conjuntos com itens únicos que são de interesse para nossa busca, porque geralmente, com grande freqüência, os conjuntos de itens dos grupos de ordem maior englobam combinações dos itens presentes no grupo de ordem menor que a sua. Esses conjuntos de itens únicos são considerados Conjuntos Fechados Freqüentes, ou Closed Frequent Itemsets (ZAKY, 2000), e a cada combinação de itens dentro de um conjunto fechado é dada a denominação de conceito.

4.5.3.1 Apriori

O *Apriori* (AGRAWAL e SRIKANT, 1994) é o algoritmo empregado para a mineração de regras de associação em grandes bancos de dados quando se deseja descobrir importantes associações entre itens que os compõem. O objetivo geral é obter todas as regras relevantes, de forma que um antecedente implique um conseqüente ($A \Rightarrow B$). A base matemática para efetivação deste processo está em duas estimativas: suporte e confiança (AGRAWAL et al., 1993). O suporte trata da freqüência com que ocorrem os itens do conseqüente com os do antecedente em toda a base de dados, ou seja, equivalente à probabilidade conjunta $P(A,B)$, enquanto a confiança representa a probabilidade condicional $P(B|A)$ que trata da garantia da freqüência dos itens do antecedente e do conseqüente da regra. Na prática são consideradas somente as regras que atendem a exigência de um suporte e confiança mínimos estabelecidos.

O algoritmo está dividido em duas etapas: num primeiro momento devem ser obtidos todos os conjuntos de itens (*itemsets*) com um suporte superior ao requisitado, os *itemsets* que atenderem a essa exigência são denominados *itemsets freqüentes*; a obtenção de regras, segundo passo, ocorre com base nos *itemsets freqüentes* obtidos na primeira etapa.

Conforme especificado na seção anterior do experimento, nosso objetivo é só a obtenção dos *itemsets freqüentes*. Para chegar a eles, o *Apriori* inicia com a geração dos 1-*itemsets*, *itemsets* com um único item, com freqüência maior que o suporte mínimo estabelecido. Esses 1-*itemsets* são utilizados para a geração dos 2-*itemsets* (com 2 itens) *candidatos*, os quais serão usados para gerar os 3-*itemsets candidatos* e assim sucessivamente até ser possível a obtenção de todas as associações de itens. O que torna os *itemsets candidatos* em *itemsets freqüentes* é satisfazerem o suporte mínimo exigido.

A idéia para a geração dos *itemsets candidatos* é partir de K iterações. Quando K=1 são gerados os 1-itemsets. Quando K=2, os K-1 (1-*itemsets*) são usados para gerar os 2-*itemsets candidatos*. Nos sucessivos incrementos de K, os K-1 *itemsets* são usados para gerar os *K-itemsets candidados*. A geração dos *itemsets candidatos* é feita pela junção dos K-1 *itemsets* conforme seu tamaho. Os candidatos que não atingirem o suporte mínimo estabelecido são desconsiderados.

4.5.4 Resultados Obtidos

Os resultados expostos retratam as seqüências presentes no campo com o identificador de seqüências do banco de dados que apresentam alguma relação, fruto do alinhamento produzido com o programa BLAST seguido da aplicação do *Apriori*.

Conforme relatado, na seção experimentos, foram encontrados grupos que relacionam de duas à doze seqüências. Para alguns grupos foram encontrados um, dois ou três conceitos, enquanto para outros não foi encontrado nenhum conceito.

As tabelas a seguir apresentam a identificação do grupo de seqüências a que se referem com o referido conceito encontrado.

Tabela 4.10: Conceito para a relação de 2 seqüências

Conceito
IGR-B-tnpAIS1296ds-MSC_0906
IGR-B-tnpAIS1296px-MSC_0233

Tabela 4.11: Conceitos para a relação de 3 seqüências

Conceito 1	Conceito 2	Conceito 3
IGR-B-locus_tag=MSC_1002-tnpIS1634bg	IGR-B-MH02172_MHJ0271-MH01375_pyrG	IGR-B-MH10370_rpsJ-MH13292_MHJ0192
IGR-B-MSC_0813-tnpIS1634chbz	IGR-B-MH19822_MHJ0264-MH02115_pheS	IGR-B-MP12640_rpsJ-MP10661_MHP0196
IGR-B-tnpIS1634cd-MSC_0922	IGR-B-MP06566_MHP0025-MP06560_sipS	IGR-B-MYPU_0050-MYPU_0060

Tabela 4.12: Conceitos para a relação de 4 seqüências

Conceito 1	Conceito 2	Conceito 3
IGR-B-MH10370_rpsJ-MH13292_MHJ0192	IGR-B-MH15047_MHJ0021-MH04541_sipS	IGR-B-tnpIS1634ab-MSC_0539
IGR-B-mhp263-tRNA-His-MH232	IGR-B-mhp263-tRNA-His-MH232	IGR-B-tnpIS1634ad-MSC_0521
IGR-B-MP18883_MHP0272-MP12636_pheS	IGR-B-MP18883_MHP0272-MP12636_pheS	IGR-B-tnpIS1634ax-glk
IGR-B-rpsF-mhp308-MH232	IGR-B-rpsF-mhp308-MH232	IGR-B-tnpIS1634ce-MSC_0872

Tabela 4.13: Conceitos para a relação de 6 seqüências

Conceito 1	Conceito 2
IGR-B-locus_tag=MSC_0059-tnpIS1634bk	IGR-B-MH02172_MHJ0271-MH01375_pyrG
IGR-B-locus_tag=MSC_1002-tnpIS1634bg	IGR-B-MH15047_MHJ0021-MH04541_sipS
IGR-B-mgtE-tnpIS1634bz	IGR-B-MH19822_MHJ0264-MH02115_pheS
IGR-B-MSC_0248-tnpIS1634ag	IGR-B-mhp263-tRNA-His-MH232
IGR-B-MSC_0784-tnpIS1634am	IGR-B-MP09790_MHP0279-MP12688_pyrG
IGR-B-rrnA-16S-tnpIS1634bv	IGR-B-rpsF-mhp308-MH232

Tabela 4.14: Conceitos para a relação de 8 seqüências

Conceito 1
IGR-B-arcB-abc
IGR-B-MH01488_rplJ-MH01491_MHJ0621
IGR-B-MH10362_pdhD-1-MH04436_nagA
IGR-B-MH12725_MHJ0424-MH03055_efp
IGR-B-MSC_0818-MSC_0819
IGR-B-MYPU_1030-MYPU_TRNA_LEU_1
IGR-B-rplJ-mhp639-MH232
IGR-B-tRNA-Ser-mhp460-MH232

Tabela 4.15: Conceitos para a relação de 9 seqüências

Conceito 1	Conceito 2
IGR-B-MH10362_pdhD-1-MH04436_nagA	IGR-B-MH12690_MHJ0398-MH12693_MHJ0400
IGR-B-MH12690_MHJ0398-MH12693_MHJ0400	IGR-B-MH12725_MHJ0424-MH03055_efp
IGR-B-MH12725_MHJ0424-MH03055_efp	IGR-B-MH15028_MHJ0502-MH07046_pdhC
IGR-B-MH16399_MHJ0481-MH06764_MHJ0482	IGR-B-MH16399_MHJ0481-MH06764_MHJ0482
IGR-B-mhp502-aceF-MH232	IGR-B-mhp415-asnS-MH232
IGR-B-MSC_0813-tnpIS1634chbz	IGR-B-mhp502-aceF-MH232
IGR-B-MSC_0818-MSC_0819	IGR-B-MP12655_rplJ-MP03313_MHP0623
IGR-B-MYPU_1030-MYPU_TRNA_LEU_1	IGR-B-MYPU_1030-MYPU_TRNA_LEU_1
IGR-B-tRNA-Ser-mhp460-MH232	IGR-B-rpsF-mhp308-MH232

Tabela 4.16: Conceitos para a relação de 10 seqüências

Conceito 1
IGR-B-MH01488_rplJ-MH01491_MHJ0621
IGR-B-MH12725_MHJ0424-MH03055_efp
IGR-B-MH16399_MHJ0481-MH06764_MHJ0482

IGR-B-mhp415-asnS-MH232
IGR-B-mhp429-efp-MH232
IGR-B-mhp502-aceF-MH232
IGR-B-MP12655_rplJ-MP03313_MHP0623
IGR-B-rplJ-mhp639-MH232
IGR-B-rRNA-16S-mhp688-MH232
IGR-B-tRNA-Ser-mhp460-MH232

Tabela 4.17: Conceitos para a relação de 11 seqüências

Conceito 1
IGR-B-MH01488_rplJ-MH01491_MHJ0621
IGR-B-MH10362_pdhD-1-MH04436_nagA
IGR-B-MH12690_MHJ0398-MH12693_MHJ0400
IGR-B-MH12725_MHJ0424-MH03055_efp
IGR-B-MH16399_MHJ0481-MH06764_MHJ0482
IGR-B-mhp415-asnS-MH232
IGR-B-mhp502-aceF-MH232
IGR-B-MP12655_rplJ-MP03313_MHP0623
IGR-B-MYPU_1030-MYPU_TRNA_LEU_1
IGR-B-rplJ-mhp639-MH232
IGR-B-tRNA-Ser-mhp460-MH232

Tabela 4.18: Conceitos para a relação de 12 seqüências

Conceito 1	**Conceito 2**
IGR-B-MH01488_rplJ-MH01491_MHJ0621	IGR-B-MH01488_rplJ-MH01491_MHJ0621
IGR-B-MH12690_MHJ0398-MH12693_MHJ0400	IGR-B-MH12725_MHJ0424-MH03055_efp
IGR-B-MH12725_MHJ0424-MH03055_efp	IGR-B-MH15020_MHJ0322-MH21628_MHJ0323
IGR-B-MH16399_MHJ0481-MH06764_MHJ0482	IGR-B-MH16399_MHJ0481-MH06764_MHJ0482
IGR-B-mhp415-asnS-MH232	IGR-B-mhp415-asnS-MH232
IGR-B-mhp502-aceF-MH232	IGR-B-mhp502-aceF-MH232
IGR-B-MP12655_rplJ-MP03313_MHP0623	IGR-B-MP12655_rplJ-MP03313_MHP0623
IGR-B-MSC_0813-tnpIS1634chbz	IGR-B-MSC_0813-tnpIS1634chbz
IGR-B-MYPU_1030-MYPU_TRNA_LEU_1	IGR-B-rplJ-mhp639-MH232
IGR-B-rplJ-mhp639-MH232	IGR-B-rRNA-16S-mhp688-MH232
IGR-B-rRNA-16S-mhp688-MH232	IGR-B-tRNA-Ser-mhp460-MH232
IGR-B-tRNA-Ser-mhp460-MH232	IGR-B-tRNA-Tyr-mhp399-MH232

Para os grupos com 5 e 7 seqüências, respectivamente, não foram encontrados conceitos.

4.5.5 Discussão dos Resultados

Os resultados expõem várias séries de seqüências, aos grupos de quantidade, que apresentam alguma relação, obtida com um refinamento do algoritmo *Apriori*, perante o alinhamento realizado, mediante diversos parâmetros especificados na execução do programa BLAST.

Devido ao amplo resultado produzido pelo programa de alinhamento e à escala heurística de aplicação dos filtros, consideramos bastante complexo o processo de descarte de supostas seqüências irrelevantes e manutenção de um número restrito de seqüências supostamente mais significativas. Apesar destes fatores, foi obtido um conjunto consolidado de registros sobre o qual se aplicou o algoritmo de mineração de dados.

Para estender o uso dos conceitos obtidos como uma forma de pré-processamento de dados, no sentido de compor um conjunto de informações mais significativas destinadas para treinamento de redes neurais, seria desejado possuir uma quantidade superior de seqüências envolvidas nos grupos obtidos e conseqüente uma maior representatividade. Apesar de não obtermos um número considerado relevante de seqüências alinhadas que pudessem contribuir para geração de um modelo neural com maior poder preditivo, acreditamos que os conceitos obtidos podem ser de interesse dos especialistas que investigam seqüências promotoras dos *Mycoplasmas*, vindo direcionar a análise e pesquisa das seqüências indicadas nas relações dos conceitos.

Desta forma, devido à complexidade e sutilezas no pré-processamento dos dados e aos resultados obtidos, o experimento tentando detectar relevância no alinhamento de seqüências de regiões intergênicas foi deixado de lado e se partiu para uma extensão dos dos experimentos iniciais de classificação de regiões que continham ou não promotores, do Experimento III, com a adição de regiões intergênicas até então não empregadas. O que resultou no próximo experimento descrito.

4.6 Experimento V

Este experimento apresenta uma extensão do Experimento III, ampliando a quantidade de seqüências com o uso de regiões intergênicas não exploradas anteriomente. São realizadas composições de conjuntos e treinamento de vários modelos supervisionados e descritos diversos teste que comparam os resultados obtidos para cada conjunto.

4.6.1 Dados Utilizados

Os dados representativos de promotores utilizados neste experimento referem-se às regiões intergênicas IGR-*F* e IGR-*R*, conforme especificação do item 4.4.1.1, relativos aos 12 organismos considerados no Experimento III, representando seqüências que contém um promotor antecipando o gene na fita senso ou anti-senso, respectivamente, conforme a orientação da seqüência for *forward* ou *reverse*.

As seqüências IGR-*F* e IGR-*R* foram anotadas com a ferramenta IGR-Annot.

As regiões IGR-*B*, descritas em experimento anterior e que contém um promotor em suas seqüências, foram utilizadas como amostras de teste de regiões promotoras.

Para a realização de testes com amostras representativas de seqüências não promotoras foram usadas regiões codificantes e amostras negativas sintéticas, geradas a partir das seqüências positivas.

4.6.2 Pré-processamento dos Dados

A extração das regiões de interesse IGR-*F* e IGR-*R*, anotadas no arquivo padrão GenBank modificado com a ferramenta IGR-Annot, foi realizada com o software Artemis 7. Nos arquivos GenBank referentes a cada organismo foi realizada uma busca no campo *misc_feature* pela expressão IGR-*F* ou IGR-*R* para obtenção das seqüências com essa característica e que apresentasse no mínimo 100 pb em sua composição.

Quando encontradas as seqüências de interesse, as mesmas foram armazenadas em arquivo formato FASTA para posteriormente serem extraídos exatos 100 pb de cada seqüência conforme padronização estabelecida para os experimentos.

A tab. 4.19 apresenta a quantidade de seqüências obtidas para cada organismo. As seqüências obtidas para cada região de interesse de cada organismo foram unidas em um único arquivo e posteriormente foi efetivada a codificação binária. Ao final foi realizada a união de todas as amostras IGR-*F* e IGR-*R*, em um único arquivo, que totalizou 2002 amostras.

Tabela 4.19: Quantidade de seqüências obtidas nos 12 organismos para regiões IGR-*F* e IGR-*R*

Organismo	**IGR-*F***	**IGR-*R***
M. galliceticum	93	68
M. genitallium	27	32
M. hyopneumoniae J	75	87
M. hyopneumoniae 232	68	84
M. hyopneomoniae 7448	80	88
M. móbile	34	58
M. mycoides	140	162
M. penetrans	171	200
M. pneumoniae	110	58
M. pulmoniae	83	77
M. synoviae	44	54
Ureaplasma	59	50
Total de seqüências	984	1018

Quanto à obtenção de seqüências de regiões codificantes, o processo de sua obtenção foi semelhante ao de aquisição das regiões IGR-*F* e IGR-*R*. Entretanto a diferença está no campo de consulta no arquivo GenBank, com o uso do Artemis em que se buscou: regiões codificantes com a designação *CDS*, determinação da orientação das seqüências (*forward* ou *reverse*) e especificado que fossem seqüências com pelo

menos 100 pb. Com isso foi possível a produção de arquivos FASTA com essas características para cada um dos 12 organismos.

Assim foram extraídas subseqüência com exatos 100 pb das seqüências presentes nos arquivos FASTA e aplicada a codificação binária. A tab. 4.20 apresenta o número de regiões codificantes de cada organismo. Por fim se deu a junção de todas as amostras codificantes selecionadas em um único arquivo com 8232 amostras.

Tabela 4.20: Quantidade de seqüências codificantes nos 12 organismos

Organismo	***Forward***	***Reverse***
M. galliceticum	383	338
M. genitallium	279	205
M. hyopneumoniae J	322	343
M. hyopneumoniae 232	172	84
M. hyopneomoniae 7448	323	340
M. móbile	323	310
M. mycoides	469	547
M. penetrans	504	533
M. pneumoniae	405	284
M. pulmoniae	406	376
M. synoviae	267	405
Ureaplasma	330	284
Total de seqüências	4183	4049

Além do uso de regiões codificantes, também foram utilizadas amostras sintéticas negativas, representando não promotores, para realização de testes. A geração dessas amostras ocorreu pelo processo de aleatorização do posicionamento de cada nucleotídeo que constituí cada uma das seqüências tratadas como positivas, ou seja, seqüências referentes às regiões IGR-*B*, IGR-*F* e IGR-*R*. Com esse processo se preservou as proporções dos pares de base constituintes das seqüências, mas a aleatorização descaracterizou a informação fundamental. A junção das três regiões totalizou 3320 seqüências que foram sintetizadas e em seguida binarizadas, obtendo-se amostras negativas sintéticas.

4.6.3 Experimentos

Nos moldes dos testes iniciais realizados no Experimento III, com o *Tenfold Cross Validation* e percentuais de dados destinados para treino e teste, foram realizados diversos experimentos.

No treinamento dos modelos foi empregada a arquitetura MLP e algoritmos de aprendizado GDX e RPROP. A sistemática de criação de modelos e realização de testes seguiu a descrição do item 4.4.3, para a abordagem supervisionada, com a diferença que para os testes com percentuais de dados para treino e teste se optou pela realização de 6 testes variando a quantidade de neurônios na camada escondida em: 11, 20 e 35,

respectivamente, para cada algoritmo de aprendizado. Os parâmetros de aprendizado mantiveram a mesma configuração: taxa de aprendizado de 0,01, constante de momento de 0,95, número máximo de 1000 épocas de treinamento e erro médio quadrado desejado de 10^{-6}.

Dois conjuntos de dados foram utilizados para geração de modelos neurais conforme o processo descrito no parágrafo anterior. Um utilizando o novo conjunto de amostras referenciando as regiões IGR-*F*, IGR-*R* e IGR-*X* e outro referenciando o mesmo conjunto de amostras com adição de 1318 amostras referente às regiões IGR-*B*.

Além dos testes realizados com *Tenfold Cross Validation* ou partições dos dados em treinamento e teste aos dados reservados para construção de modelos nesse experimento, submissões de conjuntos de teste com dados desconhecidos aos modelos foram realizados, assim:

- Aos modelos treinados com dados referentes às regiões IGR-*B* e IGR-*X* obtidos no Experimento III, foram submetidos dados referentes às regiões IGR-*F* e IGR-*R* representando amostras que continham promotor e amostras provenientes das regiões codificantes e amostras sintéticas negativas, representando amostras que não continham promotor;
- Aos modelos treinados com dados referentes às regiões IGR-*F*, IGR-*R* e IGR-*X* obtidos nesse experimento, foram submetidos dados referentes às regiões IGR-*B*, do Experimento III, representando amostras que continham promotor e amostras provenientes das regiões codificantes e amostras sintéticas negativas, representando amostras que não continham promotor;
- E aos modelos treinados com dados referentes às regiões IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X* obtidos nesse experimento, foram submetidos dados referentes às amostras provenientes das regiões codificantes e amostras sintéticas negativas, representando amostras que não continham promotor.

Seguindo a padronização empregada no Experimento III, os resultados relativos ao teste com o *Tenfold Cross Validation* representam o resultado médio para a apresentação dos 10 subconjuntos, já os resultados relativos aos experimentos de treino e teste apresentam a média para as 5 execuções. Isso quando realizados teste com o conjunto composto por amostras representativas das regiões IGR-*F*, IGR-*R* e IGR-*X* e com o conjunto composto por amostras representativas das regiões IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*.

4.6.4 Resultados Obtidos

A tab. 4.21 apresenta os resultados dos experimentos realizados com os dados das regiões IGR-*F*, IGR-*R* e IGR-*X*, conforme as métricas de avaliação e algoritmos de aprendizado expostos. A tabela também apresenta a quantidade de neurônios na camada escondida de cada rede neural e a quantidade de amostras promotoras (***P***) e não promotoras (***NP***) utilizadas para teste. O número de amostras de teste utilizadas nos experimentos onde a métrica de avaliação foi percentual de treino e teste, se refere a 5 execuções em que foram utilizados 20% (informação especificada junto com a métrica de avaliação) das amostras para teste e 80 % destinadas para treino em cada execução.

Tabela 4.21: Resultados da aplicação de algoritmos de aprendizado sobre métricas de avaliação das regiões IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	Algoritmo de aprendizado	Neurônios na camada escondida	*P/NP*	*CC*	*SN*	*SP*
Tenfold Cross Validation	RPROP	13	2002/782	0,17	76%	41%
Tenfold Cross Validation	GDX	13	2002/782	0,23	78%	44%
Tenfold Cross Validation	RPROP	35	2002/782	0,18	78%	40%
Tenfold Cross Validation	GDX	35	2002/782	0,29	81%	44%
Treino e teste (20%)	RP	11	1925/865	0,24	80%	44%
Treino e teste (20%)	GDX	11	1925/865	0,25	78%	47%
Treino e teste (20%)	RP	20	1925/865	0,27	82%	44%
Treino e teste (20%)	GDX	20	1925/865	0,31	82%	47%
Treino e teste (20%)	RP	35	1925/865	0,27	82%	43%
Treino e teste (20%)	GDX	35	1925/865	0,31	85%	43%

Os resultados demonstram que o algoritmo de aprendizado GDX com maior quantidade de neurônios na camada escondida apresentou com maior freqüência os melhores resultados em todas as métricas analisadas. Destacam-se, o modelo GDX com 35 nós no teste *Tenfold Cross Validation* e o modelo GDX com 20 nós que obteve o maior percentual para especificidade e o modelo GDX com 35 nós que obteve o maior percentual para sensibilidade, respectivamente, no treino e teste.

A tab. 4.22 expõe os resultados obtidos somente para a métrica de sensibilidade, quando submetidas amostras relativas à região IGR-*B* como dados de teste, para os melhores modelos obtidos no treinamento com dados IGR-*F*, IGR-*R* e IGR-*X*. Os melhores modelos se referem aos testes que obtiveram os melhores índices percentuais na classificação relativa à sensibilidade, dentre os 10 modelos de cada teste realizado com o *Tenfold Cross Validation* e ao melhor modelo dentre as 5 execuções do teste com percentuais de treino e teste.

O número ao final do nome do modelo descrito na tabela é relativo à qual execução, mediante o tipo de teste realizado, corresponde o referido modelo.

Tabela 4.22: Resultados da submissão de amostras IGR-*B*, sobre os melhores modelos obtidos referentes às regiões IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	Modelo	*SN*
Tenfold Cross Validation	RP13_2	74%
Tenfold Cross Validation	GDX13_2	75%
Tenfold Cross Validation	RP35_2	78%
Tenfold Cross Validation	GDX35_3	81%
Treino e teste (20%)	RP11_FRX4	75%
Treino e teste (20%)	GDX11_FRX3	75%
Treino e teste (20%)	RP20_FRX4	76%
Treino e teste (20%)	GDX20_FRX2	78%

Treino e teste (20%)	RP35_FRX1	79%
Treino e teste (20%)	GDX35_FRX1	78%

A tab. 4.23 apresenta o resultado da submissão de amostras referentes às regiões codificantes dos 12 organismos considerados, aos melhores modelos obtidos no treinamento com dados IGR-*F*, IGR-*R* e IGR-*X*. Os resultados expõem a especificidade, uma vez que tratamos as amostras codificantes como não contendo promotor. Os melhores modelos se referem aos testes que obtiveram os melhores índices percentuais na classificação relativa à especificidade, dentre os 10 modelos de cada teste realizado com o *Tenfold Cross Validation* e ao melhor modelo dentre as 5 execuções do teste com percentuais de treino e teste.

Tabela 4.23: Resultados da submissão de amostras codificantes, sobre os melhores modelos referentes às regiões IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	**Modelo**	***SP***
Tenfold Cross Validation	RP13_3	27%
Tenfold Cross Validation	GDX13_2	31%
Tenfold Cross Validation	RP35_2	19%
Tenfold Cross Validation	GDX35_2	26%
Treino e teste (20%)	RP11_FRX1	30%
Treino e teste (20%)	GDX11_FRX1	28%
Treino e teste (20%)	RP20_FRX4	28%
Treino e teste (20%)	GDX20_FRX2	25%
Treino e teste (20%)	RP35_FRX1	18%
Treino e teste (20%)	GDX35_FRX1	26%

A tab. 4.24 apresenta o resultado da submissão de amostras sintéticas negativa, geradas segundo especificação da seção de pré-processamento dos dados desse experimento, aos melhores modelos obtidos no treinamento com dados IGR-*F*, IGR-*R* e IGR-*X*. Os resultados expõem a especificidade, uma vez que tratamos as amostras sintéticas negativas como uma deturpação da informação referente às regiões intergênicas IGR-*B*, IGR-*F* e IGR-*R*.

Tabela 4.24: Resultados da submissão de amostras sintéticas negativas sobre os melhores modelos referentes às regiões IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	**Modelo**	***SP***
Tenfold Cross Validation	RP13_3	27%
Tenfold Cross Validation	GDX13_2	25%
Tenfold Cross Validation	RP35_2	21%
Tenfold Cross Validation	GDX35_2	23%
Treino e teste (20%)	RP11_FRX1	28%
Treino e teste (20%)	GDX11_FRX1	28%
Treino e teste (20%)	RP20_FRX4	25%
Treino e teste (20%)	GDX20_FRX2	24%
Treino e teste (20%)	RP35_FRX1	24%
Treino e teste (20%)	GDX35_FRX1	22%

A tab. 4.25 expõe os resultados obtidos somente para a métrica de sensibilidade, quando submetidas amostras relativas às regiões IGR-*F* e IGR-*R* como dados de teste, para os melhores modelos obtidos no treinamento com dados IGR-*B* e IGR-*X* descritos no Experimento III. Novamente, os melhores modelos se referem aos testes que

obtiveram os melhores índices percentuais na classificação relativa à sensibilidade, dentre os 10 modelos de cada teste realizado com o *Tenfold Cross Validation* e ao melhor modelo dentre as 5 execuções do teste com percentuais de treino e teste.

Tabela 4.25: Resultados da submissão de amostras IGR-*F* e IGR-*R* sobre os melhores modelos referentes às regiões IGR-*B* e IGR-*X*

Tipo de teste	Modelo	*SN*
Tenfold Cross Validation	RP13_2	57%
Tenfold Cross Validation	GDX13_2	64%
Tenfold Cross Validation	RP38_2	67%
Tenfold Cross Validation	GDX35_2	66%
Treino e teste (20%)	RP11_BX5	57%
Treino e teste (20%)	GDX11_BX3	59%
Treino e teste (20%)	RP20_BX1	59%
Treino e teste (20%)	GDX20_BX2	61%
Treino e teste (20%)	RP38_BX4	65%
Treino e teste (20%)	GDX35_BX1	63%

A tab. 4.26 apresenta o resultado da submissão de amostras referentes às regiões codificantes dos 12 organismos considerados, aos melhores modelos obtidos no treinamento com dados IGR-*B* e IGR-*X*. Os resultados expõem a especificidade, uma vez que tratamos as amostras codificantes como não contendo promotor.

Tabela 4.26: Resultados da submissão de amostras de regiões codificantes sobre os melhores modelos referentes às regiões IGR-*B* e IGR-*X*

Tipo de teste	Modelo	*SP*
Tenfold Cross Validation	RP13_2	49%
Tenfold Cross Validation	GDX13_2	41%
Tenfold Cross Validation	RP38_2	37%
Tenfold Cross Validation	GDX35_2	45%
Treino e teste (20%)	RP11_BX1	40%
Treino e teste (20%)	GDX11_BX1	33%
Treino e teste (20%)	RP20_BX3	41%
Treino e teste (20%)	GDX20_BX3	42%
Treino e teste (20%)	RP38_BX1	41%
Treino e teste (20%)	GDX35_BX4	40%

A tab. 4.27 apresenta o resultado da submissão de amostras sintéticas negativa, geradas segundo especificação da seção de pré-processamento dos dados desse experimento, aos melhores modelos obtidos no treinamento com dados IGR-*B* e IGR-*X*.

Tabela 4.27: Resultados da submissão de amostras sintéticas negativas sobre os melhores modelos referentes às regiões IGR-*B* e IGR-*X*

Tipo de teste	Modelo	*SP*
Tenfold Cross Validation	RP13_2	40%
Tenfold Cross Validation	GDX13_2	35%
Tenfold Cross Validation	RP38_2	32%
Tenfold Cross Validation	GDX35_2	34%
Treino e teste (20%)	RP11_BX1	40%

Treino e teste (20%)	GDX11_BX1	36%
Treino e teste (20%)	RP20_BX3	37%
Treino e teste (20%)	GDX20_BX3	36%
Treino e teste (20%)	RP38_BX1	40%
Treino e teste (20%)	GDX35_BX4	33%

A tab. 4.28 apresenta os resultados dos experimentos realizados com os dados de todas as regiões tratadas, ou seja, IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*, conforme as métricas de avaliação e algoritmos de aprendizado expostos.

Tabela 4.28: Resultados da aplicação de algoritmos de aprendizado sobre métricas de avaliação das regiões IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	**Algoritmo de aprendizado**	**Neurônios na camada escondida**	***P/NP***	***CC***	***SN***	***SP***
Tenfold Cross Validation	RPROP	13	3320/782	0,15	83%	33%
Tenfold Cross Validation	GDX	13	3320/782	0,19	88%	33%
Tenfold Cross Validation	RPROP	35	3320/782	0,18	86%	31%
Tenfold Cross Validation	GDX	35	3320/782	0,25	84%	34%
Treino e teste (20%)	RP	11	3345/765	0,17	85%	32%
Treino e teste (20%)	GDX	11	3345/765	0,25	78%	47%
Treino e teste (20%)	RP	20	3345/765	0,27	82%	44%
Treino e teste (20%)	GDX	20	3345/765	0,31	82%	47%
Treino e teste (20%)	RP	35	3345/765	0,27	82%	43%
Treino e teste (20%)	GDX	35	3345/765	0,31	85%	43%

Os resultados demonstram que o algoritmo de aprendizado GDX apresentou os melhores resultados nas métricas analisadas. Destacam-se, o modelo GDX com 13 nós no teste *Tenfold Cross Validation* que obteve a maior sensibilidade; e os modelos GDX com 11 e 20 nós que obtiveram o maior percentual para especificidade e os modelos GDX com 20 e 35 nós que obtiveram a maior correlação entre os índices de sensibilidade e especificidade, respectivamente, no treino e teste.

A tab. 4.29 apresenta o resultado da submissão de amostras referentes às regiões codificantes dos 12 organismos considerados para os melhores modelos obtidos no treinamento com dados IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*. Os resultados expõem a especificidade, uma vez que tratamos as amostras codificantes como não contendo promotor.

Tabela 4.29: Resultados da submissão de amostras referentes às regiões codificantes sobre os melhores modelos referentes às regiões IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	**Modelo**	***SP***
Tenfold Cross Validation	RP13_3	24%
Tenfold Cross Validation	GDX13_2	22%
Tenfold Cross Validation	RP38_2	14%

Tenfold Cross Validation	GDX35_3	21%
Treino e teste (20%)	RP11_BFRX2	23%
Treino e teste (20%)	GDX11_BFRX1	19%
Treino e teste (20%)	RP20_BFRX4	20%
Treino e teste (20%)	GDX20_BFRX4	17%
Treino e teste (20%)	RP38_BFRX3	19%
Treino e teste (20%)	GDX35_BFRX5	15%

A tab. 4.30 apresenta o resultado da submissão de amostras sintéticas negativa, geradas segundo especificação da seção de pré-processamento dos dados desse experimento, aos melhores modelos obtidos no treinamento com dados IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*. Os resultados expõem a especificidade, uma vez que tratamos as amostras sintéticas negativas como uma deturpação da informação referente às regiões intergênicas IGR-*B*, IGR-*F* e IGR-*R*.

Tabela 4.30: Resultados da submissão de amostras sintéticas negativas sobre os melhores modelos referentes às regiões IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	**Modelo**	***SP***
Tenfold Cross Validation	RP13_3	19%
Tenfold Cross Validation	GDX13_2	17%
Tenfold Cross Validation	RP38_2	14%
Tenfold Cross Validation	GDX35_3	19%
Treino e teste (20%)	RP11_BFRX2	20%
Treino e teste (20%)	GDX11_BFRX1	16%
Treino e teste (20%)	RP20_BFRX4	19%
Treino e teste (20%)	GDX20_BFRX4	14%
Treino e teste (20%)	RP38_BFRX3	18%
Treino e teste (20%)	GDX35_BFRX5	13%

4.6.5 Discussão dos Resultados

Este experimento apresentou resultados relativos a uma extensão da primeira parte do Experimento III, com a adição de 2002 novas seqüências que continham promotor, referentes às regiões IGR-*F* e IGR-*R*. E uso de amostras alternativas para representar informação não promotora.

Uma comparação entre os resultados obtidos quando avaliados os testes realizados com as regiões: IGR-*B* e IGR-*X* (tab. 4.7); IGR-*F*, IGR-*R* e IGR-*X* (tab. 4.21); IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X* (tab. 4.28), aos 10 melhores testes a cada conjunto, com a variação da sua modalidade e número de nós na camada escondida, revelou que se considerados os resultados percentuais obtidos para a sensibilidade e especificidade em conjunto, os melhores resultados foram encontrados nos modelos obtidos no Experimento III, apresentando melhores índices para o coeficiente de correlação. Se considerarmos os testes realizados com os dados relativos às regiões IGR-*F*, IGR-*R* e IGR-*X*, percebemos um pequeno aumento no percentual de sensibilidade, no entanto ocorreu uma queda mais significativa no percentual de especificidade em relação ao Experimento III, acompanhado de menores índices para o coeficiente de correlação. Conforme salientado no Experimento III, o motivo para menores percentuais nos resultados de especificidade pode estar associado ao fato desse experimento apresentar uma maior quantidade de dados positivos (+ 684 amostras) e a mesma quantidade de dados negativos do Experimento III. O mesmo se observa, de

forma mais acentuada, no experimento com a junção das três regiões IGR-*B*, IGR-*F* e IGR-*R* com 3320 amostras positivas e as mesmas 782 amostras representando a informação negativa.

A comparação entre resultados obtidos quando avaliados os teste realizados com a submissão de amostras relativas às regiões IGR-*B* aos melhores modelos treinados com dados das regiões IGR-*F*, IGR-*R* e IGR-*X* (tab. 4.22) e o inverso, ou seja, a submissão de amostras relativas às regiões IGR-*F* e IGR-*R* aos melhores modelos treinados com dados das regiões IGR-*B* e IGR-*X* (tab. 4.25), em que foi avaliada somente a métrica de sensibilidade, por se considerar todas as amostras positivas, revelou que a submissão de amostras IGR-*B* aos melhores modelos obtidos no treinamento com dados relativos às regiões IGR-*F*, IGR-*R* e IGR-*X* apresentou resultados com percentuais de ~15% melhores que a situação inversa. Novamente podemos considerar a influência de um maior número de amostras positivas de treinamento ao modelo que obteve melhores percentuais de sensibilidade, supondo que esse aumento na quantidade de dados represente uma maior abrangência da informação relativa à variabilidade das seqüências que contêm um promotor.

Já a submissão de amostras negativas, tanto quando consideradas as regiões codificantes para tal fim ou quando utilizadas amostras geradas sinteticamente, não apresentou resultados significativos em nenhum dos testes realizados (tab. 4.23, tab 4.24, tab. 4.26, tab. 4.27, tab. 4.29 e tab. 4.30), demonstrando que nenhuma dessas informações é adequada para representar dados referentes a não promotores. Apesar de poucos dados, a informação referente à região IGR-*X* ainda é mais representativa para caracterizar a ausência de promotor. Outra constatação, foi que quanto mais amplo o modelo, ou seja, maior número de positivas frente às negativas utilizadas no treinamento dos modelos, piores foram os resultados referentes à especificidade encontrados.

4.7 Experimento VI

Este último experimento segue a metodologia empregada no experimento anterior, com o uso de regiões intergênicas e uma metodologia neural supervisionada, entretanto para explorar os dados referentes ao organismo *E. coli*. A realização deste experimento se deve à popularidade do organismo em questão, utilizado como base para investigação do funcionamento de diversos mecanismos envolvidos nos processos biológicos.

4.7.1 Dados Utilizados

Os dados utilizados neste experimento referem-se às regiões intergênicas IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*, conforme especificação do item 4.4.1.1, relativos ao organismo *E. coli*.

Para a realização de testes com amostras representativas de seqüências não promotoras foram usadas regiões codificantes da *E. coli*.

4.7.2 Pré-processamento dos Dados

A obtenção das regiões de interesse, anotadas no arquivo padrão GenBank modificado com a ferramenta IGR-Annot, foi realizada com o software Artemis 7. Nos arquivos GenBank referentes a cada organismo foi realizada uma busca no campo

misc_feature pela expressão referente a região desejada e que apresentasse no mínimo 100 pb em sua composição.

Quando encontradas as seqüências de interesse, as mesmas foram armazenadas em arquivo formato FASTA para posteriormente serem extraídos exatos 100 pb de cada seqüência conforme padronização estabelecida para os experimentos anteriores.

Foram encontradas as seguintes quantidades de seqüências para as referidas regiões: 1048 para IGR-*B*, 508 para IGR-*F*, 512 para IGR-*R* e 450 para IGR-*X*. O próximo passo foi a aplicação da codificação binária seguida pela formação dos conjuntos de amostras para treinamento e teste dos modelos neurais. Os conjuntos criados correspondem aos existentes no Experimento V formados por: amostras IGR-*B* e IGR-*X*, amostras IGR-*F*, IGR-*R* e IGR-*X*, e amostras IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*.

A obtenção de seqüências de regiões codificantes sobre o organismo *E. coli* seguiu a forma padrão de busca pela expressão *CDS* no software Artemis, junto com a designação da orientação desejada e tamanho mínimo da seqüência. Assim foram obtidas 4243 amostras representativas de regiões codificantes. Nesse experimento se optou pela não utilização de regiões sintéticas negativas em virtude dos resultados encontrados no experimento anterior.

4.7.3 Experimentos

A sistemática como foram realizados os testes desse experimento foi similar à realizada no Experimento V, com o uso do *Tenfold Cross Validation* e percentuais de dados destinados para treino e teste.

Os mesmo parâmetros e configurações das redes se mostraram adequados para realização dos testes.

Os três conjuntos de dados, definidos na seção de pré-processamento, foram utilizados para geração de modelos neurais referentes a cada conjunto estabelecido e aos quais foram aplicados os testes.

Além dos testes realizados com as partições dos dados, efetivados pelo *Tenfold Cross Validation* ou percentual de dados de treino e teste, foram também realizadas as submissões de conjuntos de dados desconhecidos aos respectivos modelos obtidos, com a aplicação de:

- Dados referentes às regiões IGR-*B* submetidos aos modelos obtidos para dados referentes às regiões IGR-*F*, IGR-*R* e IGR-*X*;
- Dados referentes às regiões IGR-*F* e IGR-*R* submetidos aos modelos obtidos para dados referentes às regiões IGR-*B* e IGR-*X*;
- Dados referentes às regiões codificantes submetidos aos modelos obtidos para dados referentes às regiões: IGR-*B* e IGR-*X*; IGR-*F*, IGR-*R* e IGR-*X*; e IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*.

4.7.4 Resultados Obtidos

A tab. 4.31 apresenta os resultados dos experimentos realizados com os dados das regiões IGR-*B* e IGR-*X*, conforme as métricas de avaliação e algoritmos de aprendizado expostos.

Tabela 4.31: Resultados da aplicação de algoritmos de aprendizado sobre métricas de avaliação das regiões IGR-*B* e IGR-*X*

Tipo de teste	**Algoritmo de aprendizado**	**Neurônios na camada escondida**	***P/NP***	***CC***	***SN***	***SP***
Tenfold Cross Validation	RPROP	13	1050/450	0,22	79%	43%
Tenfold Cross Validation	GDX	13	1050/450	0,23	88%	44%
Tenfold Cross Validation	RPROP	35	1050/450	0,20	80%	40%
Tenfold Cross Validation	GDX	35	1050/450	0,23	79%	44%
Treino e teste (20%)	RP	11	1030/475	0,29	83%	44%
Treino e teste (20%)	GDX	11	1030/475	0,32	80%	52%
Treino e teste (20%)	RP	20	1030/475	0,31	80%	50%
Treino e teste (20%)	GDX	20	1030/475	0,30	79%	51%
Treino e teste (20%)	RP	35	1030/475	0,32	84%	47%
Treino e teste (20%)	GDX	35	1030/475	0,32	80%	51%

Os resultados demonstram que o algoritmo de aprendizado GDX obteve melhores percentuais para especificidade para ambas as configurações, em todos os testes realizados, se destacando os modelos com 11 e 35 nós, respectivamente, com bons índices para ambas as métricas. Também o modelo RPROP com 20 nós por apresentar resultados interessantes para ambas as métricas.

A tab. 4.32 expõe os resultados obtidos somente para a métrica de sensibilidade, quando submetidas amostras relativas à região IGR-*F* e IGR-*R* como dados de teste, para os melhores modelos obtidos no treinamento com dados IGR-*B* e IGR-*X*. A escolha dos melhores modelos para essa submissão, se refere aos modelos que obtiveram os melhores resultados na classificação relativa à sensibilidade, dentre os 10 modelos de cada teste realizado com o *Tenfold Cross Validation* e ao melhor modelo dentre as 5 execuções do teste com percentuais de treino e teste.

O número ao final do nome do modelo descrito na tabela é relativo à qual execução, mediante o tipo de teste realizado, corresponde o referido modelo.

Tabela 4.32: Resultados da submissão de amostras IGR-*F* e IGR-*R* sobre os melhores modelos obtidos referentes às regiões IGR-*B* e IGR-*X*

Tipo de teste	Modelo	*SN*
Tenfold Cross Validation	RP13_7	87%
Tenfold Cross Validation	GDX13_1	73%
Tenfold Cross Validation	RP35_3	73%
Tenfold Cross Validation	GDX35_5	74%
Treino e teste (20%)	RP11_BX4	86%
Treino e teste (20%)	GDX11_BX3	74%
Treino e teste (20%)	RP20_BX4	75%
Treino e teste (20%)	GDX20_BX4	74%
Treino e teste (20%)	RP35_BX1	81%
Treino e teste (20%)	GDX35_BX2	72%

A tab. 4.33 apresenta o resultado da submissão de amostras referentes às regiões codificantes da *E. coli*, aos melhores modelos obtidos no treinamento com dados IGR-*B* e IGR-*X*. Os resultados expõem a especificidade, uma vez que tratamos as amostras codificantes como não contendo promotor.

Tabela 4.33: Resultados da submissão de amostras codificantes sobre os melhores modelos referentes às regiões IGR-*B* e IGR-*X*

Tipo de teste	Modelo	*SP*
Tenfold Cross Validation	RP13_5	41%
Tenfold Cross Validation	GDX13_1	45%
Tenfold Cross Validation	RP35_1	43%
Tenfold Cross Validation	GDX35_1	45%
Treino e teste (20%)	RP11_BX3	39%
Treino e teste (20%)	GDX11_BX5	52%
Treino e teste (20%)	RP20_BX5	44%
Treino e teste (20%)	GDX20_BX5	42%
Treino e teste (20%)	RP35_BX5	39%
Treino e teste (20%)	GDX35_BX1	43%

A tab. 4.34 apresenta os resultados dos experimentos realizados com os dados das regiões IGR-*F*, IGR-*R* e IGR-*X*, conforme as métricas de avaliação e algoritmos de aprendizado expostos.

Tabela 4.34: Resultados da aplicação de algoritmos de aprendizado sobre métricas de avaliação das regiões IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	Algoritmo de aprendizado	Neurônios na camada escondida	*P/NP*	*CC*	*SN*	*SP*
Tenfold Cross Validation	RPROP	13	1020/450	0,13	78%	37%
Tenfold Cross Validation	GDX	13	1020/450	0,17	75%	42%
Tenfold Cross Validation	RPROP	35	1020/450	0,15	80%	34%
Tenfold Cross Validation	GDX	35	1020/450	0,15	78%	37%

Treino e teste (20%)	RP	11	1085/395	0,18	78%	40%
Treino e teste (20%)	GDX	11	1085/395	0,20	71%	51%
Treino e teste (20%)	RP	20	1085/395	0,20	74%	47%
Treino e teste (20%)	GDX	20	1085/395	0,19	72%	48%
Treino e teste (20%)	RP	35	1085/395	0,19	78%	41%
Treino e teste (20%)	GDX	35	1085/395	0,20	74%	47%

Os resultados apresentados revelam que o modelo GDX com 13 nós obteve os melhores percentuais considerando ambas as métricas para o teste com o *Tenfold Cross Validation*, entretanto resultados pouco melhores foram encontrados nos testes de treinamento e teste, com destaque para os modelos GDX com 11 nós com o melhor resultado obtido para a métrica de especificidade e RPROP com 11 e 35 nós, respectivamente, para a métrica de sensibilidade.

A tab. 4.35 expõe os resultados obtidos somente para a métrica de sensibilidade, quando submetidas amostras relativas à região IGR-*B* como dados de teste, para os melhores modelos obtidos no treinamento com dados IGR-*F*, IGR-*R* e IGR-*X*. A escolha dos melhores modelos para essa submissão se refere ao modelo que obteve melhor resultado na classificação relativa à sensibilidade, dentre os 10 modelos de cada teste realizado com o *Tenfold Cross Validation* e ao melhor modelo dentre as 5 execuções do teste com percentuais de treino e teste.

O número ao final do nome do modelo descrito na tabela é relativo à qual execução, mediante o tipo de teste realizado, corresponde o referido modelo.

Tabela 4.35: Resultados da submissão de amostras IGR-*B* sobre os melhores modelos obtidos referentes às regiões IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	**Modelo**	***SN***
Tenfold Cross Validation	RP13_1	82%
Tenfold Cross Validation	GDX13_3	73%
Tenfold Cross Validation	RP35_3	78%
Tenfold Cross Validation	GDX35_3	75%
Treino e teste (20%)	RP11_FRX2	85%
Treino e teste (20%)	GDX11_FRX3	73%
Treino e teste (20%)	RP20_FRX2	75%
Treino e teste (20%)	GDX20_FRX5	74%
Treino e teste (20%)	RP35_FRX1	79%
Treino e teste (20%)	GDX35_FRX4	74%

A tab. 4.36 apresenta o resultado da submissão de amostras referentes às regiões codificantes dos 12 organismos considerados, aos melhores modelos obtidos no treinamento com dados IGR-*F*, IGR-*R* e IGR-*X*. Os resultados expõem a especificidade, uma vez que tratamos as amostras codificantes como não contendo promotor.

Tabela 4.36: Resultados da submissão de amostras codificantes sobre os melhores modelos referentes às regiões IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	Modelo	*SP*
Tenfold Cross Validation	RP13_3	51%
Tenfold Cross Validation	GDX13_1	51%
Tenfold Cross Validation	RP35_1	44%
Tenfold Cross Validation	GDX35_4	44%
Treino e teste (20%)	RP11_FRX3	44%
Treino e teste (20%)	GDX11_FRX1	55%
Treino e teste (20%)	RP20_FRX3	44%
Treino e teste (20%)	GDX20_FRX4	51%
Treino e teste (20%)	RP35_FRX4	48%
Treino e teste (20%)	GDX35_FRX1	48%

A tab. 4.37 apresenta os resultados dos experimentos realizados com os dados das regiões IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*, conforme as métricas de avaliação e algoritmos de aprendizado expostos.

Tabela 4.37: Resultados da aplicação de algoritmos de aprendizado sobre métricas de avaliação das regiões IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	Algoritmo de aprendizado	Neurônios na camada escondida	*P/NP*	*CC*	*SN*	*SP*
Tenfold Cross Validation	RPROP	13	2070/450	0,09	86%	22%
Tenfold Cross Validation	GDX	13	2070/450	0,15	88%	26%
Tenfold Cross Validation	RPROP	35	2070/450	0,16	90%	23%
Tenfold Cross Validation	GDX	35	2070/450	0,18	91%	24%
Treino e teste (20%)	RP	11	2115/415	0,18	88%	29%
Treino e teste (20%)	GDX	11	2115/415	0,15	87%	28%
Treino e teste (20%)	RP	20	2115/415	0,14	89%	24%
Treino e teste (20%)	GDX	20	2115/415	0,15	89%	26%
Treino e teste (20%)	RP	35	2115/415	0,19	91%	25%
Treino e teste (20%)	GDX	35	2115/415	0,15	89%	25%

Os resultados demonstram maior capacidade dos modelos, nos teste realizados em classificar amostras características de conterem promotor, atingindo percentuais acima de 90%, entretanto é clara uma queda no percentual de especificidade e consequentemente para os índices do coeficiente de correlação. Novamente, ocorreu um aumento na quantidade de amostras positivas com a junção dos dois conjuntos, enquanto o número de negativas permaneceu igual. Mesmo assim, destacamos o modelo GDX com 35 nós para o teste *Tenfold Cross Validation* e os modelos RPROP com 11 e 35 nós, respectivamente, entre os testes com percentuais de dados destinados a treinamento e teste.

A tab. 4.38 apresenta o resultado da submissão de amostras referentes às regiões codificantes dos 12 organismos considerados, aos melhores modelos obtidos no treinamento com dados IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*. Os resultados expõem a especificidade, uma vez que tratamos as amostras codificantes como não contendo promotor.

Tabela 4.38: Resultados da submissão de amostras codificantes sobre os melhores modelos referentes às regiões IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*

Tipo de teste	**Modelo**	***SP***
Tenfold Cross Validation	RP13_1	32%
Tenfold Cross Validation	GDX13_6	29%
Tenfold Cross Validation	RP35_1	25%
Tenfold Cross Validation	GDX35_6	27%
Treino e teste (20%)	RP11_BFRX1	28%
Treino e teste (20%)	GDX11_BFRX3	31%
Treino e teste (20%)	RP20_BFRX2	23%
Treino e teste (20%)	GDX20_BFRX2	21%
Treino e teste (20%)	RP35_BFRX1	19%
Treino e teste (20%)	GDX35_BFRX5	26%

4.7.5 Discussão dos Resultados

Este experimento apresentou resultados relativos ao emprego de dados da bactéria *E. coli* à sistemática utilizada nos Experimento III e V, com a divisão das seqüências referentes às várias regiões consideradas e composição de conjuntos de amostras para a realização de testes.

Uma comparação entre os resultados obtidos quando avaliados os testes realizados com as regiões: IGR-*B* e IGR-*X* (tab. 4.31); IGR-*F*, IGR-*R* e IGR-*X* (tab. 4.34); IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X* (tab. 4.37), aos 10 melhores testes a cada conjunto, com a variação da sua modalidade e número de nós na camada escondida, revelou que se considerados os resultados percentuais obtidos para a sensibilidade e especificidade em conjunto, os melhores resultados foram encontrados nos modelos obtidos com dados referentes às regiões IGR-*B* e IGR-*X*, apresentando melhores índices para o coeficiente de correlação. Se considerarmos os testes realizados com os dados relativos às regiões IGR-*F*, IGR-*R* e IGR-*X*, percebemos uma redução no percentual para todas as métricas em relação ao conjunto de dados IGR-*B* e IGR-*X*. Para o experimento utilizando a junção das quatro regiões IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*, se observou um resultado satisfatório para a classificação de amostras que contêm promotor acompanhada da incapacidade dos modelos em predizer amostras que não contêm promotor, certamente em decorrência da proporção de menos de 1:4 negativas.

A comparação entre resultados obtidos quando avaliados os testes realizados com a submissão de amostras relativas às regiões IGR-*B* aos melhores modelos treinados com dados das regiões IGR-*F*, IGR-*R* e IGR-*X* (tab. 4.35) e o inverso, ou seja, a submissão de amostras relativas às regiões IGR-*F* e IGR-*R* aos melhores modelos treinados com dados das regiões IGR-*B* e IGR-*X* (tab. 4.32), em que foi avaliada somente a métrica de sensibilidade, por se considerar todas as amostras positivas, revelou que ambos os resultados encontrados foram similares com percentuais de sensibilidade na faixa de 75% a 85%.

A submissão das regiões codificantes consideradas como exemplo de informação não promotora apresentou resultados regulares nos testes realizados (tab. 4.33, tab. 4.36 e tab. 4.38) em comparação aos resultados obtidos nos testes com a região IGR-*X*. Mesmo assim, ainda consideramos os resultados para predição de seqüências que não contêm promotores não satisfatórios mediante os resultados expostos, uma vez que a quantidade de seqüências informativas dessa característica ainda é considerada baixa.

4.8 Discussão Geral dos Resultados

Os resultados relatam uma evolução cronológica dos experimentos, sempre na tentativa de encontrar melhores índices de distinção entre promotores e não promotores.

Inicialmente, o Experimento I expôs investigações sobre seqüências comprovadas da presença e localização exata de promotores. Embora a quantidade de informações disponíveis seja pequena, se constatou que o conjunto de 32 amostras era mais representativo frente aos outros conjuntos utilizados ou em associação destes.

Os resultados obtidos no Experimento II, conduziram a uma investigação mais acurada com o conjunto de sucesso do Experimento I, com a aplicação de uma metodologia não supervisionada sobre as regiões, caracterizadas por posições, representativas de promotores para a obtenção de modelos de categorias, capazes de referenciar as características principais apresentadas por estas seqüências. Apesar de se obter resultados interessantes, foi considerado que um conjunto com 32 seqüências correspondia a uma quantidade insuficiente de amostras a ponto de gerar um modelo confiável.

Assim foi desenvolvido o Experimento III, que buscou compensar a inexistência de informações características disponíveis com o uso das regiões intergênicas IGR-*B* e IGR-*X*. Entretanto, a localização do promotor nessas seqüências era desconhecida, considerando-se apenas que as mesmas o continham ou não. Os resultados obtidos com modelos supervisionados foram animadores fazendo com que os Experimentos IV e V fossem uma extensão do que foi obtido no Experimento III.

Tendo como base as seqüências IGR-*B* que continham promotor, no Experimento IV se investigou possíveis similaridades que pudessem conduzir a trechos regulares dentro dessas seqüências, que correspondessem a promotores e conseqüentemente a uma maior caracterização e direcionamento da busca. Para implementar essa idéia, foi empregado o algoritmo de alinhamento de seqüências BLAST que produziu resultados muito esparsos sendo necessária a implementação de vários filtros para refinamento da informação, em busca de algo mais consistente. Deste pré-processamento se obteve um conjunto de registros indicativos do relacionamento entre as seqüências envolvidas e foram desenvolvidos algoritmos de busca dos possíveis encadeamentos entre as seqüências alinhadas. Entretanto, se chegou a soluções que conduziam a um problema exponencial. Assim se optou pelo algoritmo *Apriori* para identificação de associações entre os registros. Ao final foram obtidos conjuntos de conceitos considerados relevantes, mas novamente, com baixa quantidade de amostras os representando.

A conclusão do Experimento IV, conduziu à realização do Experimento V que tratou da adição de seqüências com inserção das regiões IGR-*F* e IGR-*R*, ambas também contendo promotores sem determinação da posição onde se encontravam. Essa

adição, ocasionou uma melhor caracterização das amostras que continham promotor frente às regiões IGR-*X* que não continham, obtendo-se bons índices para a sensibilidade. Na tentativa de se obter mais amostras que compensassem a falta de dados que não contivessem promotores, geramos amostras negativas sintéticas com base nas positivas e também usamos as regiões codificantes para representá-las. Entretanto, nenhuma dessas alternativas se mostrou viável indicando que a melhor representação para tal característica ainda era o uso das seqüências IGR-*X*.

Além do uso das técnicas padronizadas de treinamento empregadas, experimentamos usar o treinamento com parada antecipada (*early stop*), baseado na interrupção antecipada do processo de treinamento de um modelo de rede neural com a utilização de dados de validação, em conjunto com a estratégia de Máquinas de Comitê, em que são criados modelos especialistas direcionados para as classes envolvidas, sendo que a maneira como esses especialistas são combinados, a fim de produzir um único resultado, ser uma alternativa para se obter melhores resultados. Ambas as técnicas são descritas em Haykin (2001). Apesar da realização de incansáveis experimentos fazendo uso dessa metodologia não se obteve resultados satisfatórios, optando-se pelo uso dos modelos pradrão empregados até o momento.

Devido à ampla abrangência das investigações sobre a *E. coli*, considerada como organismo padrão em grande parte dos estudos direcionados. O Experimento VI expôs uma intervenção com a realização de experimentos, utilizando a mesma metodologia empregada no Experimento V, mas com o uso das regiões intergênicas da *E. coli*. Apesar de os resultados obtidos serem considerados similares aos obtidos com os *Mycoplasmas*, não é viável uma comparação desse último experimento, com os resultados relatados na seção Estado da Arte do capítulo anterior. Neles são expostos experimentos que retratam o uso de redes neurais e informações precisas da localização de promotores, diferente do nosso caso em que temos o promotor contido ou não numa seqüência. Conforme investigação realizada por Eskin et al. (2002), com base nas regiões intergênicas de 20 genomas bacteriais, os sinais promotores nas seqüências intergênicas analisadas para a *E. coli* foram considerados fracos, enquanto que para os *Mycoplasmas* estes mesmos sinais foram considerados muito fracos.

5 UM FRAMEWORK PARA RECONHECIMENTO DE PROMOTORES EM MYCOPLASMAS

Este capítulo apresenta a proposta de um framework englobando todo o processo desenvolvido nos experimentos realizados para o reconhecimento de promotores em *Mycoplasmas*, sendo descritas as etapas principais do processo para detecção de promotores e especificação das abordagens propostas. Além da apresentação do framework é apresentada uma abordagem simbólia alternativa.

Este framework é pensado como um guia de procedimentos para a geração de modelos de previsão de promotores, a partir das bases de dados biológicos, atualizadas dinâmicamente.

5.1 O Framework

A fig. 5.1 apresenta de forma abstrata a metodologia de trabalho adotada para a construção de modelos computacionais conexionistas representativos para efetuar o reconhecimento de promotores em seqüências de DNA relativos à *Mycoplasmas*.

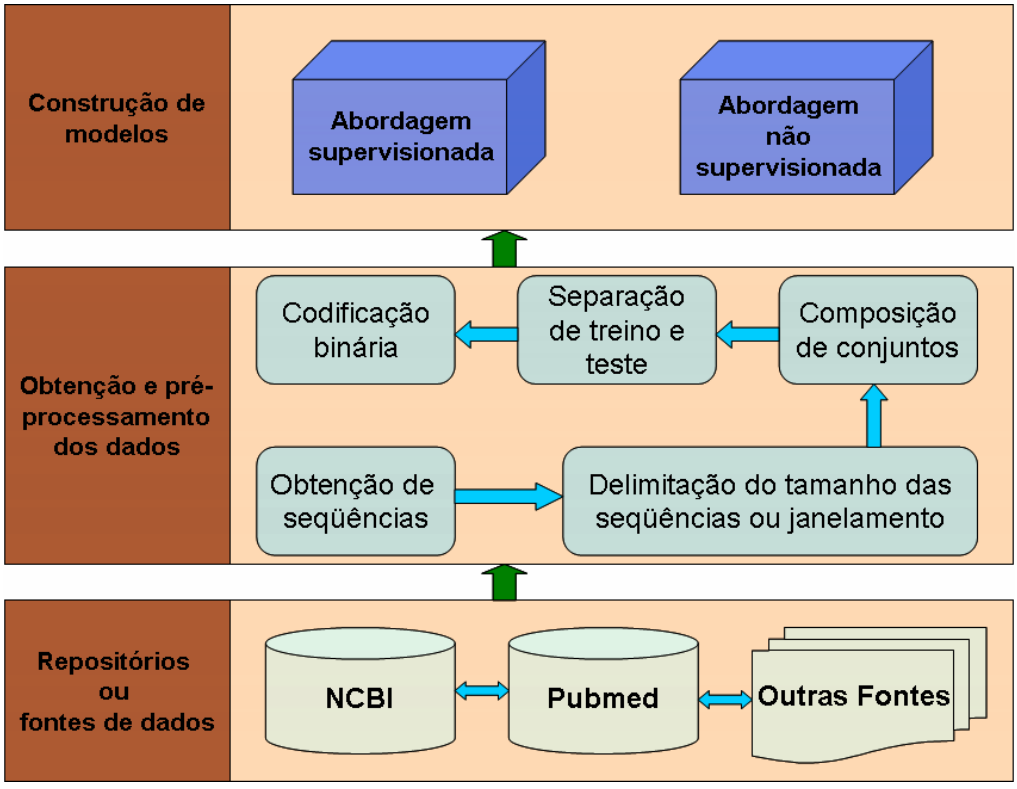


Figura 5.1: Framework para reconhecimento de promotores

O framework apresenta três principais esquemas que são: os repositórios ou fontes de dados, a obtenção e pré-processamento das seqüências obtidas e a construção de modelos computacionais conexionistas. A execução desta estrutura segue uma filosofia *bottom-up*, partindo da obtenção de seqüências nucleotidicas das bases de dados para a construção de modelos neurais.

A operacionalização desta metodologia deve inicialmente partir de uma consulta aos repositórios de seqüências de DNA, tais como NCBI, ou mesmo artigos científicos que tratem do estudo comprobatório da localização dos promotores nas seqüências. As fontes de dados estão interligadas devido ao fato de uma complementar a outra, por exemplo, caso exista um artigo que cita a identificação de seqüências relacionada mas não as apresenta na íntegra ou caso existam seqüências já caracterizadas nos bancos de dados que conduzam aos trabalhos científicos.

Uma vez encontrada a informação que caracterize o problema é dado início à próxima etapa que é a obtenção dessas seqüências e tratamento da informação para deixá-las num formato adequado à submissão das técnicas de aprendizado conexonista. Note que esta etapa segue um fluxo de tarefas que devem ser sucessivamente cumpridas para transformar as seqüências de DNA em amostras simbólicas discretizadas. Para isto ocorrer é necessário que as seqüências nucleotídicas sejam delimitadas a um tamanho fixo pré-estabelecido, ou sofram um processo de janelamento dependendo da abordagem que será adotada na construção dos modelos. Tendo-se as seqüências delimitadas a uma quantidade de pares de base é possível a composição de conjuntos, positivos e negativos, representantes de promotores e não promotores, respectivamente. Com os conjuntos compostos, deve-se separar proporcionalmente subconjuntos destinados para realização de treinamento e teste dos modelos a serem construídos. Finalmente, a última tarefa desta etapa é a codificação dos dados em formato representativo adequado para a criação de modelos.

A última etapa deste framework apresenta as abordagens supervisionada e não supervisionada, descritas nos capítulo 3 desta tese, que objetivam a obtenção de modelos computacionais genéricos e precisos para a separação de classes ou identificação de similaridades mediante o problema apresentado.

Cada uma dessas abordagens requer detalhamentos próprios para construção dos modelos que vão desde a especificação das seqüências a serem utilizadas, passando pelo direcionamento do pré-processamento empregado até a configuração dos modelos. Tais particulariadades são descritas nas próximas duas subsecções e visam indicar caminhos para a construção de modelos adequados ao reconhecimento de promotores nos *Mycoplasmas*.

5.1.1 Abordagem Supervisionada

Nesta metodologia, conforme o diagrama da fig. 5.2, estão definidas as etapas necessárias para a obtenção de um modelo supervisionado, capaz de identificar se uma dada seqüência submetida ao modelo gerado, em uma etapa de teste, contém ou não informação relativa a promotores inseridos em seu conteúdo.

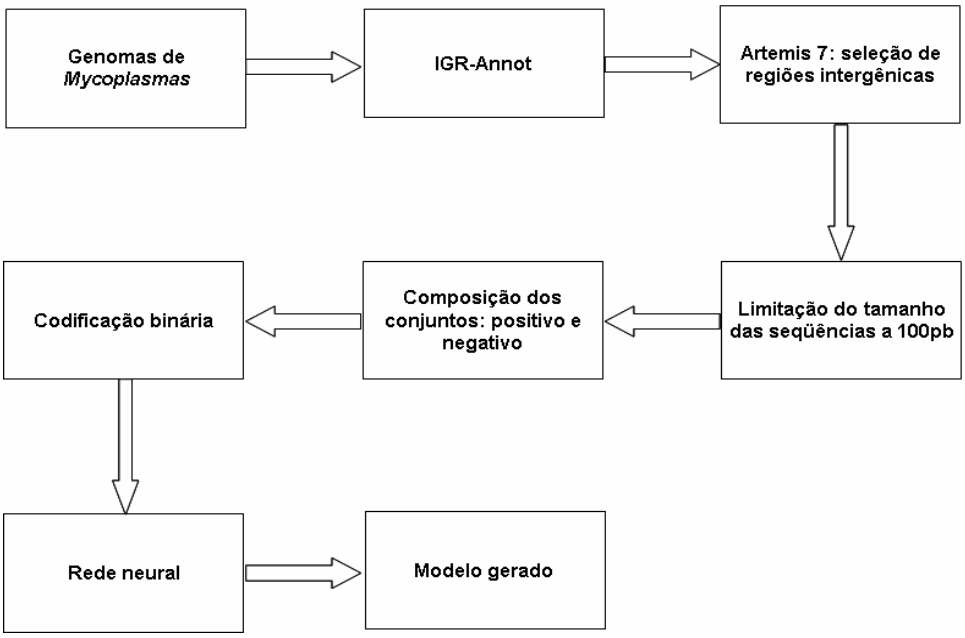


Figura 5.2: Diagrama das etapas para geração de um modelo supervisionado

Partindo do genoma completo dos organismos de interesse, no caso, relativos à *Mycoplasmas*, obtidos em sua maioria do NCBI, são anotadas as regiões intergênicas, com o uso da ferramenta IGR-Annot. Conforme relatado, a anotação produz quatro regiões diferentes: IGR-*B*, IGR-*F*, IGR-*R* e IGR-*X*, sendo as três primeiras características por conter promotor e a quarta por não conter promotor em suas seqüências. A seleção dessas regiões anotadas é realizada por uma busca ao identificador da mesma, com o emprego da ferramenta Artemis 7, que também permite a delimitação do comprimento da região de interesse. Nesse caso é indicada a seleção das regiões de interesse com pelo menos 100 pb, as seqüências que ultrapassarem esse tamanho devem ter seu comprimento, posteriormente, limitado a 100 pb.

Uma vez selecionadas as seqüências de interesse, deve ocorrer uma divisão para compor conjuntos representados aqui por: positivo e negativo, corespondentes por conter promotor e não conter promotor, respectivamente. Após essa definição, todas as amostras dos referidos conjuntos, sofrem a codificação binária. Assim, como todas as seqüências que possuem tamanho de 100 pb, suas respectivas amostras discretas são compostas por 400 elementos. Ao final de cada amostra são adicionados dois bits referentes à saída desejada: [1 0] para promotor, [0 1] para não promotor.

Com essas etapas cumpridas é possível a apresentação dos conjuntos de amostras a rede neural devidamente configurada. Após a rede atingir determinados patamares pré-estabelecidos, é considerado que a mesma teve seus pesos ajustados, estando apta a realizar a identificação de seqüências desconhecidas.

Conforme mostra o diagrama da fig. 5.3, uma seqüência desconhecida de DNA deve sofrer um processo de codificação binária seguida pela delimitação de seu tamanho correspondente a 100 pb. Para seqüências de tamanho superior, sugerimos a aplicação de uma janela correspondente aos 100 pb, ou seja, 400 elementos, que deslize do início ao fim da seqüência e que a cada deslocamento produza nova uma amostra com o tamanho definido para submissão ao respectivo modelo neural. O papel do modelo é realizar a classificação, identificando se a dada seqüência contém ou não um promotor inserido.

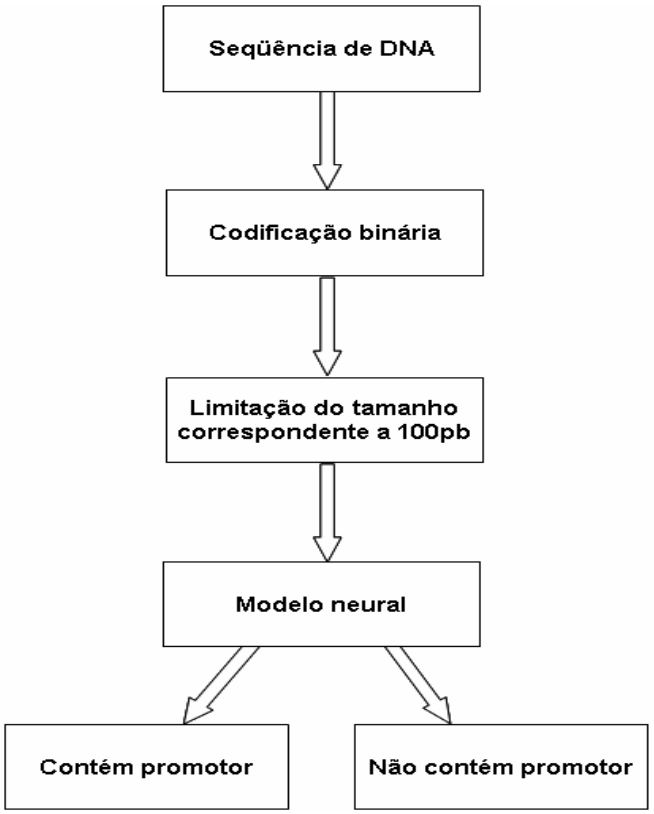


Figura 5.3: Diagrama das etapas para submissão de uma seqüência desconhecida para modelo supervisionado obtido

5.1.2 Abordagem Não Supervisionada

Esta metodologia, conforme o diagrama da fig. 5.4, define as etapas necessárias para a obtenção de categorias, com base na similaridade de seqüências comprovadas da presença de promotores.

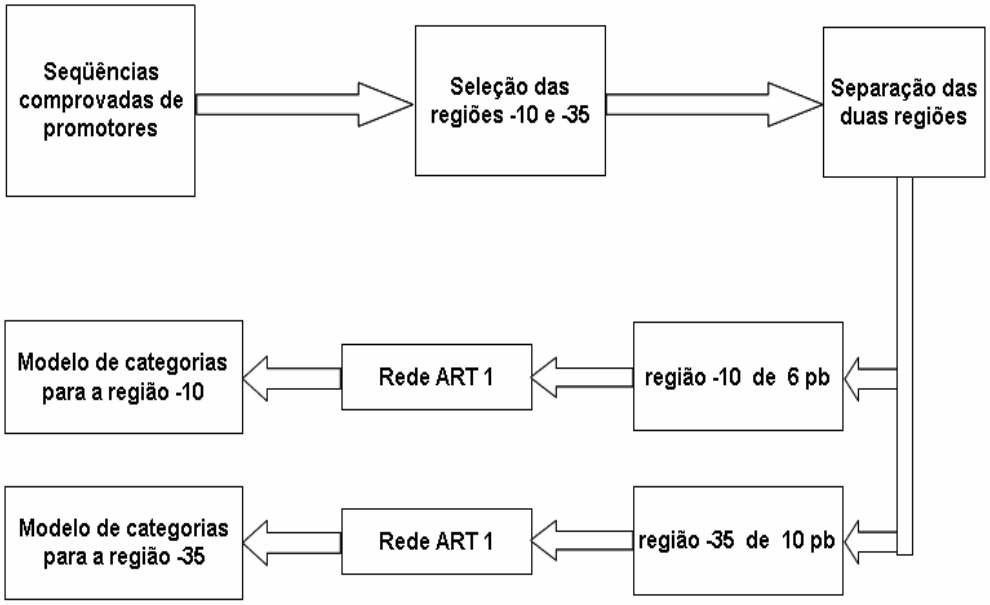


Figura 5.4: Diagrama das etapas para geração de categorias representativas de promotores com base na similaridade das seqüências comprovadas

A partir de seqüências comprovadas da presença de promotores em *Mycoplasmas*, é realizada a seleção da região -10, constituída por 6 pb, e a seleção da área de abrangencia da região -35 nas seqüências. Com base nos experimentos realizados, sugerimos que com 10 pb é possível englobar a região -35 de todas as seqüências utilizadas.

Para as duas regiões de interesse, formadas por 16 pb, é aplicada a codificação binária e produzidas amostras constituídas por 64 elementos. Ao final de cada amostra são inseridos dois bits relativos à saída desejada.

As amostras constituintes do conjunto positivo são separadas em dois subconjuntos: um representando a região -10 e outro a região -35. Cada um desses subconjuntos é aplicado a uma rede neural ART 1, a qual produz categorias conforme a identificação de similaridades entre as amostras do conjunto de dados, com base no parâmetro de vigilância configurado para a rede. Um valor alto desse parâmetro de vigilância implica numa restrição maior na identificação de similaridades entre as seqüências, caso contrário a restrição é menos seletiva na criação de categorias.

Conforme apresentado na fig. 5.5, quando submetida uma seqüência nucleotídica de teste ao modelo a mesma deve sofrer um processo de janelamento. Duas janelas devem ser fixadas na seqüência: uma de 6 pb a partir do primeiro par de base da seqüência e outra de 10 pb, distante 15 pb do final da primeira janela. A área tracejada da fig. 5.5 refere-se ao processo de propagação das amostras produzidas pelos deslocamentos ao longo da seqüência, o deslocamento das duas janelas fixadas deve ter suas regiões separadas e submetidas ao respectivo modelo.

A combinação dos melhores modelos encontrados, para cada região em separado, considera similaridades entre as seqüências e a capacidade preditiva do modelo em relação às particularidades de cada região. Ou seja, somente quando a amostra de teste submetida às redes cumprir os critérios estabelecidos para cada um dos dois modelos é que a amostra será considerada uma região promotora, indicando qual é a provável localização do promotor dentro da seqüência.

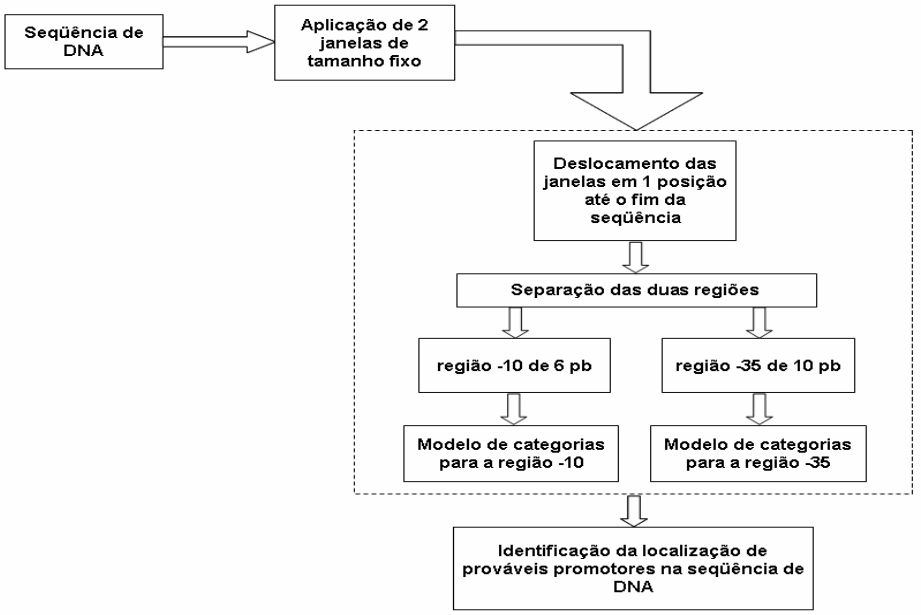


Figura 5.5: Diagrama das etapas para submissão de amostras deslocadas aos modelos de categorias

5.2 Abordagem Simbólica Alternativa

Em complementação aos modelos preditivos gerados com o framework apresentado na seção anterior, apresentamos aqui um procedimento para geração de um modelo simbólico da região promotora.

Apesar de não terem sido utilizados os conceitos finais obtidos no Experimento IV para processamento de modelos neurais, é exposta a abordagem simbólica alternativa. A ausência desta abordagem no framework apresentado se deve ao fato de ela apresentar algumas pecularied ades de pré-processamento e por ser considerada uma abordagem que necessita de aprimoramentos. O diagrama da fig. 5.6 apresenta as etapas que conduziram à obtenção de conceitos a partir de um conjunto de seqüências de DNA.

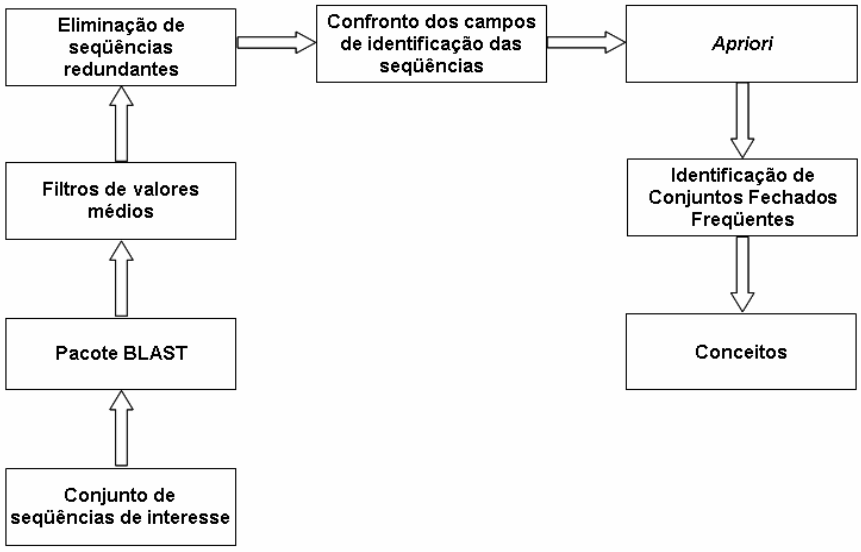


Figura 5.6: Diagrama das etapas para obtenção de conceitos a partir de um conjunto de seqüências

Partindo de um conjunto de seqüências deve ocorrer a submissão destas ao pacote BLAST para identificação dos alinhamentos. A especificação dos diversos parâmetros do BLAST é um ponto crucial que merece cuidado, os resultados produzidos podem ser muito concisos ou muito esparsos, dificultando assim o tratamento posterior. Por mais direcionada que possa ser a execução do BLAST os resultados produzidos necessitam da aplicação de filtros. Em nosso caso, os filtros identificáveis foram valores médios relativos ao *score* e tamanho das seqüências. Em continuidade às etapas definidas devem ser identificados alinhamentos únicos, ou seja, somente uma vez uma seqüência de consulta deve apresentar uma correspondência com uma seqüência do BD com o intuito de acabar com redundâncias indesejadas, assim como eliminação de sucessivos alinhamentos no decorrer de uma mesma seqüência do BD. Com base nesse refinamento são selecionados os campos de interesse: os identificadores das seqüências de consulta e do BD, e confrontados estes campos para detecção de múltiplas ocorrências da identificação das seqüências do BD em relação à identificação da seqüência de consulta. Ao final desse intenso pré-processamento são obtidos registros relevantes que podem ser aplicados ao algoritmo *Apriori* para identificação de grupos e itens freqüentes e posteriormente identificação dos conjuntos fechados freqüentes resultando na obtenção dos conceitos.

6 CONCLUSÕES

Este capítulo apresenta as conclusões da tese mediante a exposição do problema, da metodologia investigativa adotada e dos experimentos desenvolvidos, visando fornecer uma consolidação do trabalho desenvolvido por meio de um framework e possibilidades de extensão das propostas relatadas.

Nesta tese, foi relatado o estudo e desenvolvimento de metodologias para a geração de modelos capazes de efetuar o reconhecimento de regiões promotoras em organismos da família *Mycoplasmataceae*. Diante do que foi encontrado na literatura, relativo a trabalhos relacionados ao tema, concebeu-se a possibilidade de realização de tais experimentos para explorar um organismo do mesmo reino, mas com algumas particularidades um tanto desconhecidas.

Um dos principais empecilhos no desenvolvimento do trabalho foi sem dúvida, a escassez de dados comprovados, ou seja, um número significativo de seqüências elucidativas que já tivessem seus promotores muito bem caracterizados e que pudessem servir de indicativos para a geração de modelos computacionais. Embora seja relativamente bem caracterizada a definição dos promotores na *E. coli*, a mesma não se aplica completamente aos *Mycoplasmas*.

Conforme relatado, na seção 3.9, relativa ao estado da arte da pesquisa, os experimentos que utilizaram um volume maior de dados comprovados e bem definidos da presença de promotores obtiveram, de uma forma geral, bons resultados. Isso demonstra a viabilidade do uso das RNs, caso se possua um conjunto de amostras bem caracterizadas, assim como um número superior dessas amostras.

Os experimentos realizados demonstraram a geração de modelos com base nas poucas amostras comprovadas da presença do promotor, relativas a um dos *Mycoplasmas*. Esses modelos apresentaram uma baixa capacidade preditiva de generalização nos testes, não identificando com a certeza desejada novos possíveis promotores. Diante do número ínfimo de amostras, alternativas foram propostas no intuito de obter uma maior quantidade de seqüências informativas, capazes de oferecer melhores subsídios para a criação de modelos mais precisos e genéricos.

Os *Mycoplasmas*, apesar de sua simplicidade, em tamanho do genoma e número de genes, possuem detalhes próprios relativos aos seus promotores. Esses detalhes ainda não foram suficientemente elucidados para fornecer uma caracterização adequada, a fim de que os métodos computacionais de predição apresentem soluções mais palpáveis.

Os demais experimentos realizados revelaram a capacidade de obtenção de melhores resultados com base em dados indicativos da presença ou ausência de promotor nas seqüências tratadas. É coerente afirmar, tendo em vista o panorama atual, ou seja, em virtude da baixa caracterização das poucas seqüências promotoras

comprovadas de *Mycoplasmas*, que os melhores resultados encontrados são significativos diante das investigações realizadas, perante a necessidade de maior exploração cientifica do problema abordado.

6.1 Contribuições

A especificação do problema, abordada no segundo capítulo e a descrição da metodologia empregada, juntamente com a exposição de experimentos correlatos, abordada no capítulo 3, forneceram a base para o desenvolvimento de diversos experimentos que envolveram processos de: composição dos conjuntos de amostras, seleção de características, implementação de redes neurais e avaliação dos resultados obtidos sobre as seqüências relativas aos *Mycoplasmas*.

Os trabalhos relatados no capítulo 3 visaram demonstrar a aplicabilidade das RNs ao problema de reconhecimento de promotores em procariotos. A comparação dos resultados expostos nesses experimentos, com os resultados obtidos nos experimentos desenvolvidos nesta tese não é coerente, porque as abordagens propostas por eles não se aplicam aos experimentos com *Mycoplasmas*. Além disso, esses trabalhos foram baseados em uma quantidade muito superior de seqüências comprovadas da presença de promotor, situação totalmente oposta à encontrada nos *Mycoplasmas*.

Outra questão importante, é a existência de um número pequeno de trabalhos na literatura que abordem o reconhecimento de promotores em *Mycoplasmas*, muito menos ainda que empreguem RNs para tal fim. Isto torna o presente trabalho inédito, por explorar um problema de natureza extrema, onde se possui um conjunto restrito de amostras, com baixa caracterização que fornece poucos subsídios para construção de um modelo genérico, capaz de ser aplicável no reconhecimento de promotores em *Mycoplasmas*.

Perante a inexistência de outros trabalhos relacionados ao tema em questão, o desenvolvimento desse trabalho se tornou desafiante. Trabalhos como este tendem a abrir as portas para a criação de soluções a problemas dessa natureza, dando vazão à busca de novas metodologias.

A realização dos experimentos descritos no capítulo 4, bem como os resultados obtidos permitiram a identificação de duas principais abordagens para solução do problema: uma para identificar se uma dada seqüência apresenta informação relativa ao promotor ou não em sua composição; e outra que busca identificar categorias representativas de promotores com base na similaridade das seqüências comprovadas. Ambas as abordagens foram consolidadas na composição de um framework, que agrega as principais etapas e as tarefas envolvidas para obtenção de modelos computacionais conexionistas capazes de efetuar o reconhecimento de promotores.

Também foi exposta a investigação a uma abordagem simbólica alternativa, que utilizou um pré-processamento intenso em conjunto com uma técnica de Mineração de Dados, obtendo-se listagens de seqüências que apresentam algum relacionamento.

Embora o foco desta tese tenha sido um grupo de organimos referentes à família *Mycoplasmataceae*, o trabalho desenvolvido é modular e flexível estando aberta a novas investigações com o emprego de outros organismos procariotos.

Abaixo, são expostas sugestões que visam colaborar no aprimoramento dessa pesquisa.

6.2 Trabalhos Futuros

Com a intenção de se obter modelos mais sensíveis e específicos, ou seja, com maior poder de predição de promotores e não promotores, em seqüências de DNA relativas aos *Mycoplasmas*, são apresentadas algumas sugestões de extensão desse trabalho que podem vir a ser investigadas:

- Ampliação do número de organismos referentes à família *Mycoplasmataceae* e consequentemente aumento no número de seqüências envolvidas;
- Utilização de seqüências relativas à *operons*. Os *operons* são considerados regiões do DNA, em que o promotor dispara o processo de transcrição de uma série de genes muito próximos fisicamente, onde o promotor está localizado somente antes do primeiro gene a ser transcrito, enquanto os demais genes são transcritos automaticamente pela RNA-polimerase. O uso desses dados para a realização de testes implicará a identificação de somente um promotor, e que as regiões antecessoras dos demais genes do *operon*, posteriores ao primeiro, não apresentarão promotores;
- Refinamento no uso do pacote BLAST, na tentativa de direcionar os parâmetros de execução com melhores subsídios biológicos. Ou mesmo uma pré-seleção das seqüências submetidas ao BLAST, no sentido de obter alinhamentos mais representativos e assim um maior número de seqüências constituindo conceitos obtidos a partir da extensão do algoritmo *Apriori*;
- Investigação de abordagens híbridas de redes neurais para construção de modelos mais robustos, que ampliem sua capacidade preditiva;
- Com a obtenção de modelos mais precisos e exatos sugere-se a elaboração de um portal WWW para submissão de seqüências de interesse da comunidade científica.

REFERÊNCIAS

ABDUL-KAREEM, S. et al. Back Propagation Neural Network for Medical Prognosis: A Comparison of Different Training Algorithms. **Electronic Journal of School of Advanced Technologies - AIT**, Thailand, v.3, n.1, Apr. 2001.

AGRAWAL, R.; IMIELINSKI, T.; SWAMI, A. Mining Association Rules between Sets of Items in Large Databases. In: INTERNATIONAL CONFERENCE ON MANANGEMENT OF DATA, SIGMOD, 1993. **Proceedings…** New York: ACM Press, 1993. p. 207-216.

AGRAWAL, R.; SRIKANT, R. Fast Algorithms for Mining Association Rules. In: INTERNATIONAL CONFERENCE ON VERY LARGE DATABASES, VLDB, 20., 1994. **Proceedings…** Hove: Morgan Kaufmann, 1994. p. 487-499.

ALBERTS, B. et al. **Biología Molecular de la Célula**. Barcelona: Omega, 1996. 1232p.

ALBERTS, B. DNA Replication and Recombination. **Nature**, London, v.421, p. 431-435, Jan. 2003.

ALLEN, D.M. The Relationship between Variable Selection and Data Augmentation and a Method for Prediction. **Technometrics**, Washington, v.16, n.1, p. 125-127, 1974.

ALTSCHUL, S. et al. Basic Local Alignment Search Tool. **Journal of Molecular Biology**, London, v. 215, p. 403-410, 1990.

AMABIS, J.M.; MARTHO, G.R. **Biologia dos Organismos:** Classificação, Estrutura e Função nos Seres Vivos. São Paulo: Moderna, 1995.

ATTWOOD, T.K. **Introduction to Bioinformatics**. Harlow, England: Prentice Hall, 1999. 218p.

BAILEY, T.L.; ELKAN, C.P. Unsupervised Learning of Multiple Motifs in Biopolymers Using Expectation Maximization. **Machine Learning**, Boston, v.21, p. 51–83, 1995.

BAJIĆ, V.B. Comparing the Success of Different Prediction Software in Sequence Analysis: A Review. **Briefings in Bioinformatics**, [S.l.], v.1, n.3, p.214-228, 2000.

BALL, P. Portrait of a Molecule. **Nature**, London, v.421, p. 421-422, Jan. 2003.

BARANAUSKAS, J.A. **Extração Automática de Conhecimento por Múltiplos Indutores**. 2001. Tese (Doutorado em Ciências da Computação e Matemática Computacional) - Instituto de Ciências Matemáticas e de Computação, USP. São Carlos.

BARRETO, J.M. Introdução as Redes Neurais Artificiais. In: ESCOLA REGIONAL DE INFORMÁTICA DA SBC – REGIONAL SUL, 5., 1997. **Anais...** [S.l.:s.n.], 1997. p. 41-71.

BERGER, J.O. **Statistical Decision Theory and Bayesian Analysis**. New York: Springer-Verlag, 1985.

BONATO, C.M. **Moldes, Módulos e Forma:** do DNA às Proteínas. Disponível em: <http://www.biologianaweb.com/Livro2/Moldes.htm>. Acesso em: jun. 2005.

BRAGA, A.P.; CARVALHO, A.C.P.L.F.; LUDERMIR, T.B. **Redes Neurais Artificiais:** Teorias e Aplicações. Rio de Janeiro: LTC, 2000. 262p.

BRITANICA.COM. **Encyclopædia Britannica Online**. Disponível em: < http://www.britannica.com/search?ref=A01015&query=ligand> . Acesso em: jun. 2005.

BROWNING, D.F.; BUSBY, S.J.W. The Regulation of Bacterial Transcription Initiation. **Nature Reviews**, **Microbiology**, London, v.2, p. 1-9, Jan. 2004.

BURDEN, S.; LIN, X.Y; ZHANG, R. Improving Promoter Prediction for the NNPP2.2 Algorithm: a Case Study Using *Escherichia coli* DNA Sequences. **Bioinfomatics**, Oxford, v.21, n.5, p. 601-607, 2005.

BURGER, C.; KARLIN, S. Prediction of Complete Gene Structures in Human Genomic DNA. **Journal of Molecular Biology**, London, v.268, n.1, p. 78-94, 1997.

CAMARGO, S.S. et al. IGR-ANNOT: A Multiagent System for InterGenic Regions Annotation. In: BRAZILIAN SYMPOSIUM ON MATHEMATICAL AND COMPUTATIONAL BIOLOGY, 4., 2004, Ilhéus. **Proceedings…** Rio de Janeiro : E-papers, 2005. p. 107-121.

CARDON, L.R.; STORMO, G.D. Expectation Maximization Algorithm for Identifying Protein-binding Sites with Variable Lengths from Unaligned DNA Fragments. **Journal of Molecular Biology**, London, v.223, n.1, p. 159–170, 1992.

CARPENTER, G.A.; GROSSBERG, S. A Massively Parallel Architecture for a Self-organizing Neural Networks. **Computer Vision, Graphics, and Image Processing**, New York, v.37, p. 54-115, 1987.

CARPENTER, G.A.; GROSSBERG, S.; ROSEN, D.B. Fuzzy ART: Fast Stable Learning and Categorization of Analog Patterns by Adaptive Resonance System. **Neural Networks**, New York, v. 4, p. 759-771, 1991.

CARPENTER, G.A. el al. Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidemensional Maps. **IEEE Transactions on Neural Networks**, New York, v.3, n.5, p. 698-713, Sept. 1992.

COOPER, G.M.; HAUSMAN R.E. **The Cell:** a Molecular Approach. 3rd ed. Sunderland, MA: Sinauer, 2003. 713p.

COTIK, V.; ZALIZ, R.R.; ZWIR, I. A Hybrid Promoter Analysis Methodology for Prokaryotic Genomes. **Fuzzy Sets and Systems**, Amsterdam, v.152, n.1, p. 83-102, 2005.

DULBECCO, R. **Os Genes e o Nosso Futuro:** O Desafio do Projeto Genoma. São Paulo: Best Seller, 1997.

ESKIN, E. et al. Genome-Wide Analysis of Bacterial Promoter Regions. In: PACIFIC SIMPOSIUM ON BIOCOMPUTING, PSB, 8., 2003, Kauai, Hawaii. **Proceedings…** Singapore: World Scientific, 2002. p. 29-40.

FELSENFELD, G.; GROUDINE, M. Controlling the Double Helix. **Nature**, London, v.421, p. 448-453, Jan. 2003.

FINE, T.L. **Feedforward Neural Network Methodology**. New York: Spring-Verlag, 1999.

FREEMAN, J.A.; SKAPURA, D.M. **Neural Networks:** Algorithms, Applications and Programming Techniques. New York: Addison-Wesley, 1991. 401p.

FRIEDBERG, E.C. DNA. Damage and Repair. **Nature,** London, v.421, p. 436-440, Jan. 2003.

GIBAS, C.; JAMBECK, P. **Developing Bioinformatics Computer Skills**. Sebastopol, CA: O´Reilly, 2001. 427p.

GOWDAK, D.; MATTOS, N. S. **Biologia II**. São Paulo: FTD, 1995. 192p.

GUAZZELLI, A. **Do ART 1 ao Fuzzy ARTMAP:** um Estudo sobre Modelos de Redes Neurais Artificiais Baseados na Teoria da Adaptação Ressonante (ART). 1993. Trabalho Individual (Mestrado em Ciência da Computação) – Instituto de Informática, UFRGS, Porto Alegre.

GUAZZELLI, A. **Aprendizagem em Sistemas Híbridos**. 1994. Dissertação (Mestrado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

HARLEY, C.B.; REYNOLDS, R.P. Analysis of *E. coli* Promoter Sequences. **Nucleic Acids Research**, Oxford, v.15, n.5, p. 2343–2361, 1987.

HAYKIN, S. **Redes Neurais Princípios e Prática**. 2 ed. Porto Alegre: Bookman, 2001.

HEBB, D.O. **The Organization of Behavior**. New York: Wiley, 1949.

HECTH-NIELSEN, R. **Neurocomputing**. Massachusetts: Addison Wesley, 1990.

HERTTZ, G.Z.; STORMO, G.D. Identifying DNA and Protein Patterns with Statistically Significant Alignments of Multiple Sequences. **Bioinformatics**, Oxford, v.15, p. 563–577, 1999.

HOPFIELD, J.J. Neural Networks and Physical Systems with Emergent Collectives Computational Abilities. **Proceedings of the National Academy of Sciences - PNAS**, USA, v.79, n.8, p. 2554-2558, Apr. 1982.

HOOD, L.; GALAS, D. The Digital Code of DNA. **Nature**, London, v.421, p. 444-448, Jan. 2003.

HUNTER, L. **Artificial Intelligence and Molecular Biology**. Menlo Park, CA: AAAI/MIT Press, 1993. 500p.

HUTCHISON 3rd, C.A.; MONTAGUE, M.G. Mycoplasmas and the Minimal Genome Concept. In: RAZIN, S.; HERRMANN, R. (Ed.). **Molecular Biology and Pathogenicity of Mycoplasmas**. New York: Kluwer Academic/Plenum, 2002. p. 221-254.

KALATE, R.N.; KULKARNI, B.D.; NAGARAJA, V. Analysis of DNA Curvature Distribution in Mycobacterial Promoters Using Theoretical Models. **Biophysical Chemistry**, Amsterdam, v.99, n.1, p. 77-97, Sept. 2002.

KALATE, R.N.; TAMBE, S.S.; KULKARNI, B.D. Artificial Neural Networks for Prediction of Mycobacterial Promoter Sequences. **Computational Biology and Chemistry**, London, v.27, p. 555-564, 2003.

KARP, P.D. et al. The EcoCyc Database. **Nucleic Acids Research**, Oxford, v.30, n.1, p. 56–58, 2002.

KONAR, A. **Artificial Intelligence and Soft Computing:** Behavior and Cognitive Modeling of the Human Brain. Florida: CRC Press LLC, 2000. 788p.

KOVÁCS, Z.L. **Redes Neurais Artificiais:** Fundamentos e Aplicações. 3 ed. São Paulo: Livraria da Física, 2002.

LAGUNA, M.; MARTI, R. **Scatter Search:** Methodology and Implementations in C. Boston: Kluwer, 2003.

LANG, K.J.; HINTON G.E. **The Development of the Time-delay Neural Network Architeture for Speech Recognition**. Pittsburgh, Carnegie-Mellon University, 1988. (Technical Report CMU-CS-88-152).

LAWRENCE, C.E.; REILLY, A.A. An Expectation Maximization (EM) Algorithm for the Identification and Characterization of Common Sites in Unaligned Biopolymer Sequences. **Proteins: Structure, Function, and Genetics**, New York, v.7, p. 41–51, 1990.

LEWIN, B. **Genes VII**. Porto Alegre: Artes Médicas, 2001. 955p.

LISSER, S.; MARGALIT, H. Compilation of *E. coli* mRNA Promoter Sequences. **Nucleic Acid Research**, Oxford, v.21, p.1507-1156, 1993.

LODISH, H. et al. **Molecular Cell Biology**. 4th ed. New York: W. H. Freeman and Company, 2000. 1184p.

LOESCH, C.; SARI, S.T. **Redes Neurais Artificiais Fundamentos e Modelos**, Blumenau: FURB, 1996.

LUGER, G.F. **Inteligência Artificial Estruturas e Estratégias para a Solução de Problemas Complexos**. 4 ed. Porto Alegre: Bookmann, 2004. 774p.

MA, Q. et al. DNA Sequence Classification Via an Expectation Maximization Algorithm and Neural Networks: A Case Study. **IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Reviews**, New York, v.31, n.4, p. 468-475, Nov. 2001.

MACKAY, D.J.C. **Information Theory:** Inference, and Learning Algorithms. Cambridge: Cambridge University, 2003. 640p.

MAHADEVAN, I.; GHOSH, I. Analysis of Promoter *E. coli* Structures Using Neural Networks. **Nucleic Acids Research**, Oxford, v.22, n.11, p. 2158-2165, 1994.

MANILOFF, J. Phylogeny and Evolution. In: RAZIN, S.; HERRMANN, R. (Ed.). **Molecular Biology and Pathogenicity of Mycoplasmas**. New York: Kluwer Academic/Plenum, 2002. p. 31-44.

MATHEWS, B.W. Comparison of the Predicted and Observed Secondary Structure of T4 Phage Lysozyme. **Biochim. Biophys. Acta**, Amsterdam, v.405, n.2, p. 442-451, Oct. 1975.

McCULLOCH, W.S.; PITTS, W. A Logical Calculus of the Ideas Immanent in Nervous Activity. **Bulletin of Mathematical Biophysics**, [S.l.], v.5, p. 115-133, 1943.

MENDEL, J.M.; McLAREN, R.W. Reinforcement Learning Control and Pattern Recognition Systems. In: MENDEL, J. M.; FU, K. S. (Ed.). **Adaptative, Learning and Pattern Recognition Systems:** Theory and Applications. New York: Academic, 1970. p. 287-318.

MINATTI, E. Lipídios: As Biomoléculas Hidrofóbicas. **Revista Eletrônica de Química – QMCWEB,** Florianópolis, v.4, 2003. Disponível em: <http://qmc.ufsc.br/qmcweb/artigos/lipidios/lipidios.html >. Acesso em: jun. 2005.

MINSKY, M.; PAPERT, S. **Perceptrons:** An Introduction to Computational Geometry. Masschusetts: MIT, 1969.

MØLLER, M.F. A Scaled Conjugate Gradient Algorithm for Fast Supervised Learning. **Neural Networks**, New York, v.6, n.4, p. 525-533, 1993.

MULDER, M.A.; ZAPPE, H.; STEYN, L.M. Mycobacterial Promoters. **Tuberculosis Lung Disease**, Edinburgh, v.78, p. 211-223, 1997.

MURAKAMI, K. S. et al. Structural Basis of Transcription Initiation: an RNA Polimerase Holoenzime-DNA Complex. **Science**, New York, n.296, p. 1285-1290, May 2002.

NAKAYAMA, M. et al .*Micrococcus luteus*, a Bacterium with a High Genomic G+C Content, Contains *Escherichia coli*-type Promoters. **Molecular & General Genetics**, Berlin, v.218, n.3, p. 384-389, 1989.

NYGARD, O. **DNA - RNA – Protein (advanced level - Protein Translation)**. Disponível em: <http://www.nobel.se/medicine/educational/dna/b/translation/>. Acesso em: jun. 2005.

OHAMA, T. et al. Organization and Codon Usage of the Streptomycin Operon in *Micrococcus luteus*, a Bacterium with a High Genomic G+C Content. **Journal of Bacteriology**,Washington, v.169, p. 4770-4777, 1987.

OLBY, R. Quiet Debut for the Double Helix. **Nature**, London, v.421, p. 402-405, Jan. 2003.

OZOLINE, O.N.; DEEV, A.A.; ARKHIPOVA, M.V. Non-canonical Sequence Elements in the Promoter Structure. Cluster Analysis of Promoters Recognized by *E. coli* RNA Polymerase. **Nucleic Acids Research**, Oxford, v.25, n.23, p. 4703–4709, 1997.

RAZIN, S. et al. Molecular Biology and Pathogenicity of Mycoplasmas. **Microbiology and Molecular Biology Reviews**, Washington, v.62, n.4, p. 1094–1156, Dec. 1998.

REESE, M.G. **Computational Prediction of Gene Structure and Regulation in the Genome of Drosophila melanogaster**. 2000. PhD Thesis, Fakultät II - Biologie – Universität Hohenheim, Göttingen.

ROSENBLATT, F. The Perceptron: a Probabilistic Model for Information Storage and Organization in the Brain. **Psycological Review**, Washington, v.65, 386-408, 1958.

RUMELHART, D.E.; HINTON, G.E.; WILLIAMS, R.J. Learning Internal Representations by Error Propagation. In: RUMELHART, D.; McCLELLAND, J.(Ed.). **Parallel Distributed Processing Explorations in the Microstructure of Cognition**. Cambridge, MA: Mit, 1986. v.1, p. 318-362.

SANTNER, T.J. **The Statistical Analysis of Discrete Data**. New York: Springer-Verlag, 1989.

SARKER, R.; LIANG, K.; NEWTON, C. A New Multiobjective Evolutionary Algorithm. **European Journal of Operational Research**, Berlin, v.140, n.1, p. 12–23, 2002.

SETUBAL, J.C.; MEIDANIS, J. **Introduction to Computational Molecular Biology**. Boston, MA: PWS, 1997. 296p.

SIMPSON, P. K. **Artificial Neural Systems**. New York: Pergamon, 1990.

STADEN, R. Computer Methods to Locate Signals in Nucleic Acid Sequences. **Nucleic Acid Research**, Oxford, v.12, n.1, p. 505-519, 1984.

SWINGLER, K. **Applying Neural Networks:** A Pratical Guide. San Diego: Academic Press, 1996. 303p.

TVETER, D.R. **The Pattern Recognition Basis of Artificial Intelligence**. Los Alamitos, CA: IEEE Computer Society, 1998. 369p.

USSERY, D.W.; HALLIN, P.F. Genome Update: AT Content in Sequenced Prokaryotic Genomes. **Microbiology**, London, v.150, p. 749–752, Apr. 2004.

VALIATI, J.F. **Reconhecimento de Voz para Comandos de Direcionamento por meio de Redes Neurais**. 2000. Dissertação (Mestrado em Ciência da Computação) - Instituto de Informática, UFRGS, Porto Alegre.

VALIATI, J.F.; ENGEL, P.M. Discovering Similarities in Mycoplasma Promoters: A Clustering Strategy. In: BRAZILIAN SYMPOSIUM ON MATHEMATICAL AND COMPUTATIONAL BIOLOGY, 5., 2005, Petrópolis. **Proceedings…** Rio de Janeiro : E-papers, 2006. p. 141-156.

VASCONCELOS, A.T.R. et al. Swine and Poultry Pathogens: the Complete Genome Sequences of Two Strains of *Mycoplasma hyopneumoniae* and a Strain of *Mycoplasma synoviae*. **Journal of Bacteriology**, Washington, v.187, n.16, p. 5568-5577, 2005.

WAIBEL, A. et al. Phoneme Recognition Using Time-delay Neural Networks. **IEEE Transactions on Acoustics, Speech, and Signal Processing**, San Francisco, v.37, p. 328-339, 1989.

WALDO 3rd, R.H. et al. Transcription Analysis of the hmw Gene Cluster of *Mycoplasma pneumoniae*. **Journal of Bacteriology**, Washington, v.186, n.16, p. 4978-4985, 1999.

WANG, J.T. et al. Discovering Active Motifs in Sets of Related Protein Sequences and Using them for Classification. **Nucleic Acid Research**, Oxford, v.22, n.14, p. 2769-2775, 1994.

WANG, J.T.; MA, Q. Recognizing Promoters in DNA Using Bayesian Neural Networks, In: INTERNATIONAL CONFERENCE OF ARTIFICIAL INTELLIGENCE AND SOFT COMPUTING, IASTED, 1999, Honolulu. **Proceedings…** Calgary, Canada: ACTA Press, 1999. p. 301-305.

WEINER 3rd, J.; HERRMANN, R.; BROWNING, G.F. Transcription in *Mycoplasma pneumoniae*. **Nucleic Acids Research**, Oxford, v.28, n.22, p. 4488-4496, 2000.

WERNER, T. Promoter Analysis. In: MEWES, H.-W.; SEIDEL, H.; WEISS, B. (Ed). **Bioinformatics and Genome Analysis**. Berlin: Springer, 2002. p. 65-82. (Ernst Shering Research Foundation Workshop, v.38).

WHELAN, P.F.; MOLLOY, D. **Machine Vision Algorithms in Java**. London:Springer-Verlag, 2001.

WIESLANDER, L. **DNA - RNA – Protein (advanced level - RNA Processing)**. Disponível em: <http://www.nobel.se/medicine/educational/dna/b/translation/ >. Acesso em: jun. 2005.

WRANGE, O. **DNA - RNA – Protein (advanced level - RNA Transcription)**. Disponível em: <http://www.nobel.se/medicine/educational/dna/b/translation/ >. Acesso em: jun. 2005.

WU, C.H.; McLARTY, J.M. **Neural Networks and Genome Informatics**. New York: Elsevier Science, 2000.

YAN, C.T.K. **Speaker Adaptative Phoneme Recognition Using Time Delay Neural Networks**. 2000. Thesis in Computer Science - National University of Singapore, Singapore.

ZAHA, A. et al. **Biologia Molecular Básica**. 3. ed. Porto Alegre: Mercado Aberto, 2003. 421p.

ZAKI, M.J. Generating Non-Redundant Association Rules. In: INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, SIGKDD, 6., 2000. **Proceedings…** New York: ACM Press, p. 34-43.

ZITZLER, E.; THIELE, L. Multiobjetive Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. **IEEE Transactions on Evolutionary Computation**, New York, v.4, n.3, p. 257-271, Nov. 1999.

APÊNDICE A ONZE AGRUPAMENTOS PARA DUAS JANELAS DE 6 PB

A Tab. A1 apresenta os 11 agrupamentos encontrados com o parâmetro de vigilância (ρ = 0,5) sobre informação relativa a duas janelas de 6 pb cada uma, relativos às 32 amostras. A coluna **Agrupamentos** se refere aos agrupamentos encontrados pela ART 1. A coluna **Seqüências** apresenta as 32 seqüências numeradas de 1 a 32. As colunas **Seqüência desejada** e **Seqüência obtida**, apresentam os pares de base que compõem a região de interesse e a reconstrução do que o modelo ART 1 recuperou, respectivamente.

Tabela A1: Agrupamentos encontrados com ρ = 0,5 sobre duas janelas de 6 bp

Agrupamento	Seqüências	Seqüência desejada		Seqüência obtida	
		-10	-35	-10	-35
1	1	attata	tgtact	att-ta	t-----
	2	attcta	tagcag		
	11	attcta	tttcaa		
	13	atttta	tcgcat		
	18	attata	tggctt		
	19	atttta	tgctaa		
	21	attcta	tgaccg		
	22	attcta	tgacaa		
2	3	atttta	ctttaa	att-ta	----a-
	4	attata	ggccac		
	10	attcta	aatcaa		
	17	atttta	ggtgaa		
	27	atttta	attaaa		
3	5	taatag	aaaggg	ta--a-	-a-gg-
	14	tagaat	tatggt		

4	6	tataat	caaatt	-a-aat	----tt
	7	taaaat	ttagtt		
	8	taaaat	acgttt		
	9	aacaat	cgggtt		
	25	tagaat	tcattt		
5	12	atttta	attccc	att-ta	-t-c--
	20	attcta	ctgcga		
6	15	tagaat	aaatta	ta--at	a-a---
	16	tacaat	aaattg		
	24	tatcat	aaagtt		
	30	taatat	acaaat		
7	23	atctcc	tgacca	atct--	t-ac-a
	29	atctta	taacta		
8	26	aaccac	acattc	aaccac	acattc
9	28	aagtta	tcccaa	aagtta	tcccaa
10	31	tcaaat	aacaac	tcaaat	aacaac
11	32	tatcat	ttggct	tatcat	ttggct

APÊNDICE B QUATORZE AGRUPAMENTOS PARA DUAS JANELAS: UMA DE 6 PB E OUTRA DE 10 PB

A Tab. B1 apresenta os 14 agrupamentos encontrados com o parâmetro de vigilância (ρ = 0,5) sobre informação relativa a duas janelas, uma de 6 pb sobre a região -10 e outra de 10 pb sobre a região -35, relativos às 32 amostras. A coluna **Agrupamentos** se refere aos agrupamentos encontrados pela ART 1. A coluna **Seqüências** apresenta as 32 seqüências numeradas de 1 a 32. As colunas **Seqüência desejada** e **Seqüência obtida**, apresentam os pares de base que compõem a região de interesse e a reconstrução do que o modelo ART 1 recuperou, respectivamente.

Tabela B1: Agrupamentos encontrados com ρ = 0,5 sobre duas janelas, uma de 6 pb sobre a região -10 e outra de 10 pb sobre a região -35

Agrupamentos	Seqüências	Seqüência desejada		Seqüência obtida	
		-10	-35	-10	-35
1	1	attata	tgtacttagg	att-ta	--t---t-g-
	3	atttta	ctttaattgc		
	18	attata	attggcttga		
2	2	attcta	tagcagttaa	att-ta	------tt-a
	4	attata	ggccacttta		
	13	atttta	atcgcattaa		
3	5	taatag	aaagggaaaa	taata-	aaa---aaa-
	30	taatat	aaatacaaat		
4	6	tataat	caaattggga	ta-aat	--a-tt----
	7	taaaat	ttagttgaac		
	15	tagaat	caaattaagg		
	16	tacaat	aaaattgagt		
5	8	taaaat	acgtttactt	-a-aat	--g-tt--tt
	9	aacaat	cgggttcttt		

6	10	attcta	taatcaatta	att-ta	----c-a-t-
	11	attcta	atttcaaatg		
	19	atttta	attgctaata		
7	12	atttta	aattccccac	atttta	aa-------c
	17	atttta	aaggtgaacc		
8	14	tagaat	gtatggtcgt	tagaat	-t-t—t--t
	25	tagaat	ctttcatttt		
9	20	attcta	aatctgcgaa	attcta	--t---c---
	21	attcta	ctttgaccgg		
	22	attcta	ttttgacaat		
10	23	atctcc	agttgaccaa	a—-t--	--tt-ccaa
	28	aagtta	tttttcccaa		
11	24	tatcat	ctaaaagttt	tatcat	-t----g--t
	32	tatcat	gtttttggct		
12	26	aaccac	gcaacattcg	aaccac	gcaacattcg
13	27	atttta	caagattaaa	at-tta	--ag-----a
	29	atctta	atagtaacta		
14	31	tcaaat	tgacaacaac	tcaaat	tgacaacaac

APENDICE C DEZ AGRUPAMENTOS PARA UMA JANELA DE 6 PB SOBRE A REGIÃO -10

A Tab. C1 apresenta os 10 agrupamentos encontrados, para a aplicação do modelo ART 1 com o parâmetro de vigilância (ρ = 0,7) sobre informação relativa à janela de 6 pb sobre a região -10, relativos as 32 amostras. A coluna **Agrupamentos** se refere aos agrupamentos encontrados pela ART 1. A coluna **Seqüências** apresenta as 32 seqüências numeradas de 1 a 32. As colunas **Seqüência desejada** e **Seqüência obtida**, apresentam os pares de base que compõem a região de interesse e a reconstrução do que o modelo ART 1 recuperou, respectivamente.

Tabela C1: Agrupamentos encontrados com ρ = 0,7 sobre a região -10 de 6 pb

Agrupamentos	Seqüências	Saída desejada	Saída obtida
		-10	-10
1	1	attata	att-ta
	2	attcta	
	3	atttta	
	4	attata	
	10	attcta	
	11	attcta	
	12	atttta	
	13	atttta	
	17	atttta	
	18	attata	
	19	atttta	
	20	attcta	
	21	attcta	
	22	attcta	

	27	atttta	
2	5	taatag	taata-
	30	taatat	
3	6	tataat	ta-aat
	7	taaaat	
	8	taaaat	
	14	tagaat	
	15	tagaat	
	16	tacaat	
	25	tagaat	
4	9	aacaat	aacaat
5	23	atctcc	atctcc
6	24	tatcat	tatcat
	32	tatcat	
7	26	aaccac	aaccac
8	28	aagtta	aagtta
9	29	atctta	atctta
10	31	tcaaat	tcaaat

APENDICE D DOZE AGRUPAMENTOS PARA UMA JANELA DE 10 PB SOBRE A REGIÃO -35

A Tab. D1 apresenta os 12 agrupamentos encontrados, para a aplicação do modelo ART 1 com o parâmetro de vigilância ($\rho = 0,3$) sobre informação relativa à janela de 10 pb sobre a região -35, relativos as 32 amostras. A coluna **Agrupamentos** se refere aos agrupamentos encontrados pela ART 1. A coluna **Seqüências** apresenta as 32 seqüências numeradas de 1 a 32. As colunas **Seqüência desejada** e **Seqüência obtida**, apresentam os pares de base que compõem a região de interesse e a reconstrução do que o modelo ART 1 recuperou, respectivamente.

Tabela D1: Agrupamentos encontrados com $\rho = 0,3$ sobre a região -35 de 10 pb

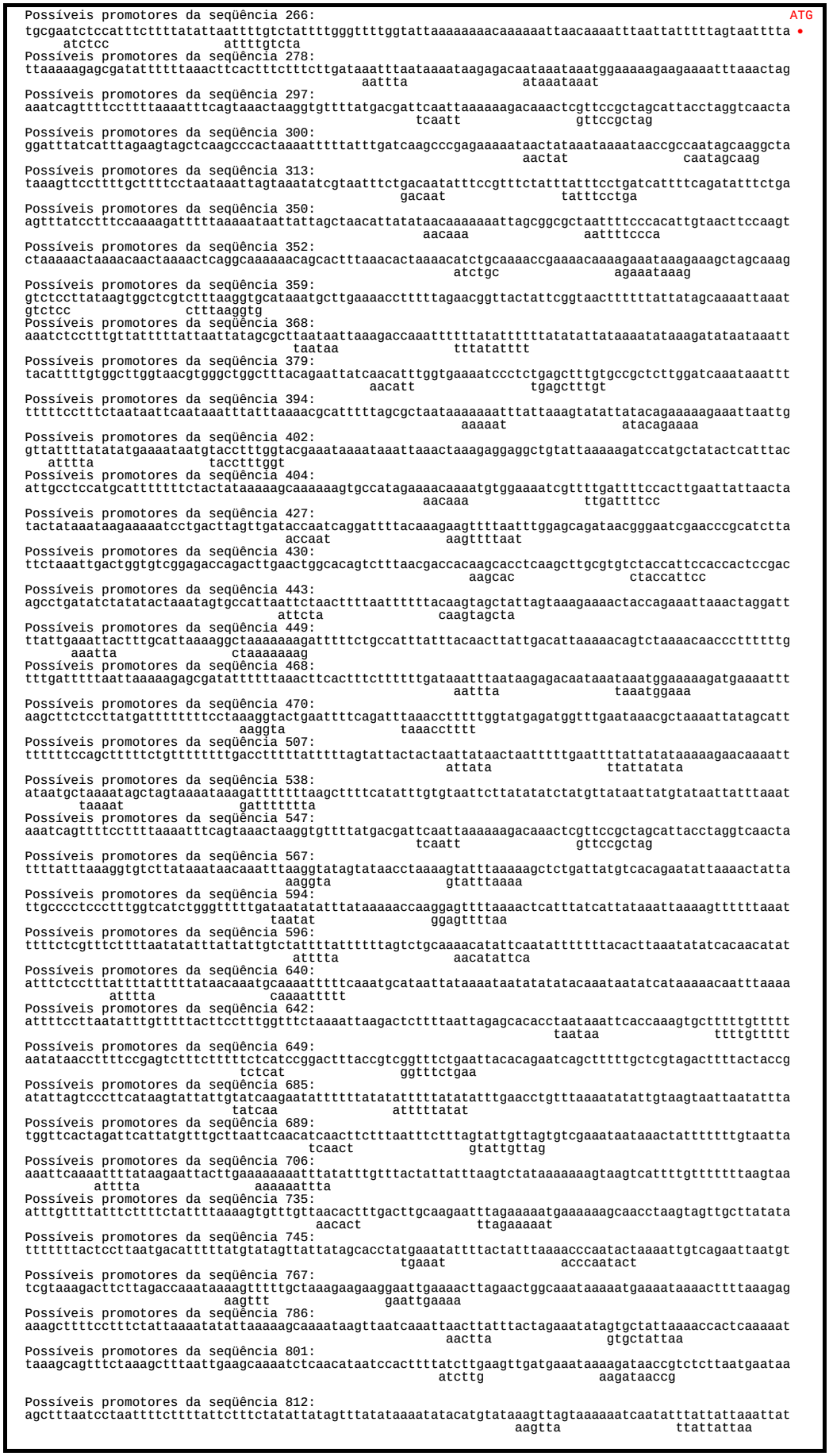
Agrupamentos	Seqüências	Saída desejada	Saída obtida
		-35	-35
1	1	tgtacttagg	--t---t-g-
	3	ctttaattgc	
	18	aatggcttga	
2	2	tagcagttaa	------tt-a
	4	ggccacttta	
	13	atcgcattaa	
3	5	aaagggaaaa	-aa------a
	6	caaattggga	
	10	taatcaatta	
	27	caagattaaa	
4	7	ttagttgaac	--a-tt-a--
	15	caaattaagg	
	16	aaaattgagt	
5	8	acgtttactt	----tt---t
	9	cgggttcttt	
	32	gtttttggct	

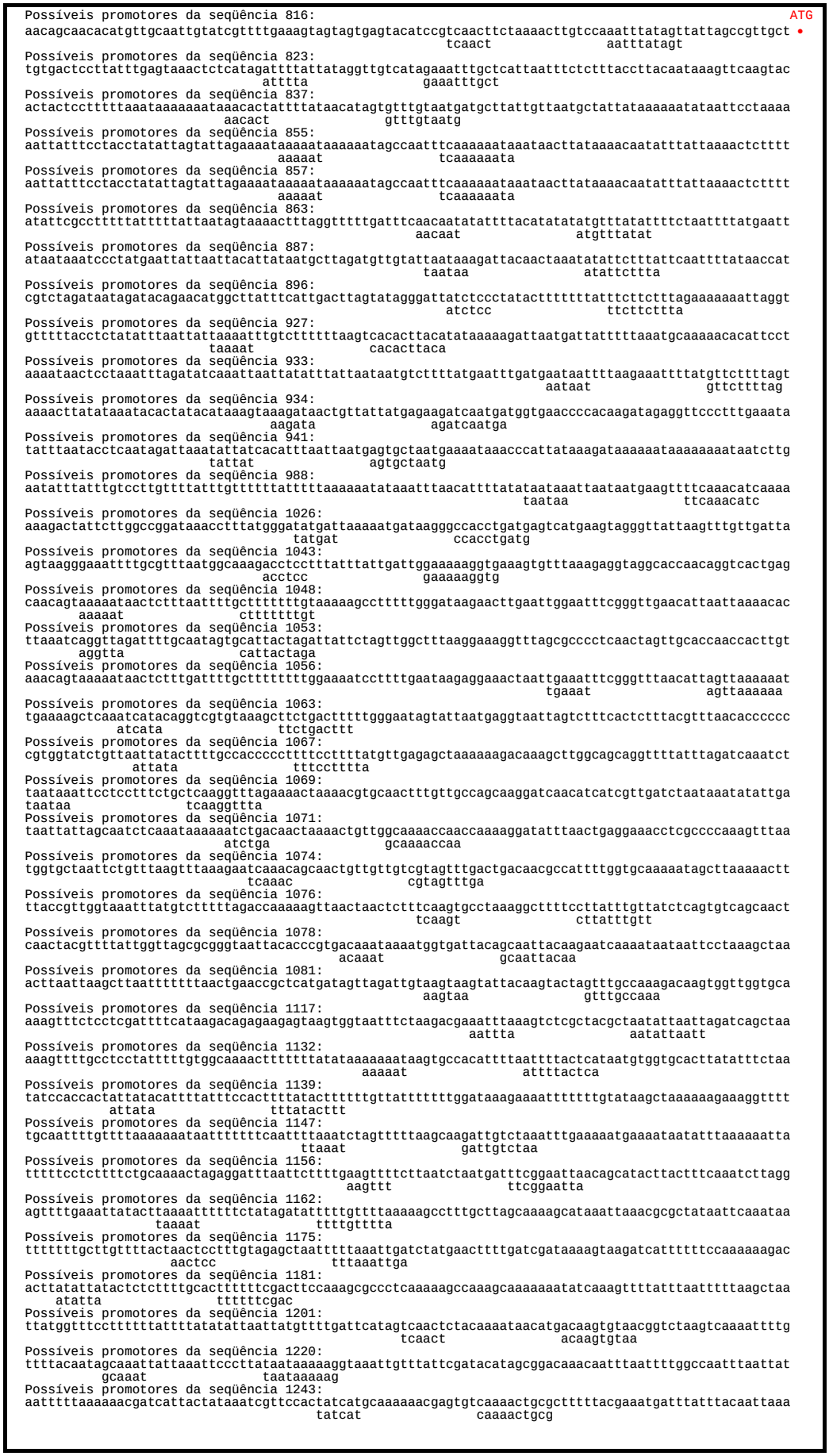
6	11	atttcaaatg	a-t-c-----
	12	aattccccac	
	19	attgctaata	
7	14	gtatggtcgt	-t-tg-----
	21	ctttgaccgg	
	22	ttttgacaat	
8	17	aaggtgaacc	aa--tg----
	20	aatctgcgaa	
9	23	agttgaccaa	--tt--ccaa
	28	tttttcccaa	
10	24	ctaaaagttt	-t---a--t-
	25	ctttcatttt	
	29	atagtaacta	
11	26	gcaacattcg	gcaacattcg
12	30	aaatacaaat	--a-a--aa-
	31	tgacaacaac	

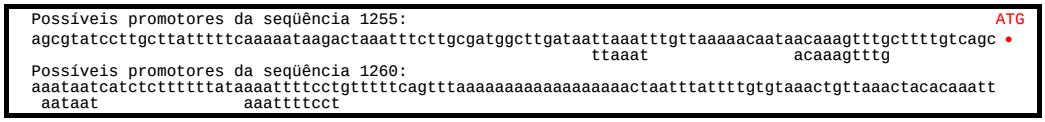
APÊNDICE E UMA POSSÍVEL POSIÇÃO E SEQUÊNCIA DE PROMOTOR EM UMA SEQUÊNCIA ORIGINAL

O quadro abaixo, apresenta a saída do programa que utilizou a combinação do modelo ART 1 treinado com ρ = 0,7 sobre a região -10 de 6 pb com outro modelo ART 1 treinado com ρ = 0,3 sobre a região -35 de 10 pb. Foram identificadas possíveis localizações de promotor em 97 seqüências das 1318 seqüências positivas e apresentadas somente seqüências em que foi encontrado somente 1 possível promotor. A numeração das seqüências é um identificador seqüencial referente às 1318 seqüências. Todas as seqüências possuem 100 pb e o ATG está situado no canto direito.

```
Possíveis promotores da seqüência 12:                                                                ATG
Gttaagattttaaaatatataattacctttttataaatgttaaaatattatcagtattatgtgagatagagctacttttttagacagttttagctctaa •
      atttta               ttttttataa
Possíveis promotores da seqüência 13:
tttttatttctttttaaaatattttacaaaattaacctgagtttccaaagtggaagctctttttataccttttaccttattacttatatatacactatata
                           aaatta                gtggaagctc
Possíveis promotores da seqüência 14:
attgttttaacaatatgtttatataaataaatatttgccttttttatcgataaaatcaataaaaactggaaaatgacacacttaaaaatttaactgttcat
                                                 atcaat               acacacttaa
Possíveis promotores da seqüência 18:
tctttaattttgttggagagaagataattctttcaccagtacgaatatttttttcacaatagactaattgaagtaagtgtaaatacgatagcagctgaatc
                    aagata              cgaatatttt
Possíveis promotores da seqüência 21:
ttaaagtcgaattgacgaatttgtaaaacgaacaagttcatcaagaaagaatatataaagttcaataagtaataaatcggcttaattttcgccgatcaaa
                               aagttc               ataaagttca
Possíveis promotores da seqüência 56:
gctaattaattttgataattaattataactaatttaactttagataagatttagtttttaagcttaactaggatttctattttaagcaacttgttttctt
                          aattta               tagtttttaa
Possíveis promotores da seqüência 73:
attttctttgcccgaataaaacccgtttgcattttcatttctaatagaaattgttgttgtggttgttgtaagaattgtacataaagaaccaaaaattccg
                                                                  aagaat               aaaaattccg
Possíveis promotores da seqüência 75:
tgatttttcgttaatttcagtcataattgtttccttttttaatttttcggttttggtttgttgcttattctaatttaatatttttttaaacattttaata
                                                                        attcta               aacattttaa
Possíveis promotores da seqüência 78:
tgcaccataaaaactacgtttttggttaatttgtttcaaaagccggtatgcgagccgtaattcggatgaggttttaaaaaagttaaatagatttttcatc
          aactac              gtttcaaaag
Possíveis promotores da seqüência 85:
aaaaaccaatacacgataacaactcactctcgcgtactgcaaccttatgcatcatcatcctttaacggtaagattaaattatacacaaaaaataaaaatt
                aacaac              gcaaccttat
Possíveis promotores da seqüência 103:
gagcgcctaagcaaaagttttagacatatgcgaaatttcgttgtgaaaatctccttttttaatttttatgatcagcgtattttatgataaaatttatca
                                                                  tatgat               taaaatttat
Possíveis promotores da seqüência 113:
tagattctggttatttctggtttggctggattttgctatcacttgttctaaatttggattttcctgagtttcccagtttgatggcgaaaattcacttttt
                                  tatcac               attttcctga
Possíveis promotores da seqüência 120:
ttttttctccttttttttttttttttttttactaaaaaaataaaaaaactactattttgaatagtagttttaaatttaaatggtgaatttactatatcaaa
                                        aactac               ttttaaattt
Possíveis promotores da seqüência 129:
tttttccagctttttctgttttttttttttttttttttgacttttttattttagtattactactaattataactaatttttgaattttattatataaaa
                                                                  attata               ttattatata
Possíveis promotores da seqüência 151:
ttaaaaagagcgatatttttttaaacttcactttcttttttgataaatttaataagagacaataaataaatggaaaaagatgaaaatttaaactagttatt
                                        aattta               taaatggaaa
Possíveis promotores da seqüência 171:
aaatcagttttccttttaaaatttcagtaaactaaggtgttttatgacgattcaattaaaaaaagacaaactcgttccgctagcattacctaggtcaact
                                              tcaatt               cgttccgcta
Possíveis promotores da seqüência 176:
ggatttatcatttagaagtagctcaagaccactaaaatttttatttaatcaagcccgagaaaaataactatataaataaaataaccgccaatagcaaggcta
                                                                        aactat               caatagcaag
Possíveis promotores da seqüência 206:
tgatttttcgttaatttcagtcataattgtttccttttttttaatttttcggttttggtttgttgcttattctaatttaatatttttttaaacattttaata
                                                                        attcta               aacattttaa
Possíveis promotores da seqüência 209:
tgccccataaaaactacgtttttggttaatttgtttcaaaagccggtatgcgagccgtaattcggatgaggttttaaaaaagttaaatagatttttcatc
          aactac              gtttcaaaag
Possíveis promotores da seqüência 216:
aaaaaccaatacacgataacaactcactctcgcgtactgcaaccttatgcatcatcatcctttaacggtaagattaaattatacacaaaaaataaaaatt
                aacaac              gcaaccttat
Possíveis promotores da seqüência 247:
ttttttctccttttttttttttttttttactaaaaaaataaaaaaactactattttgaatagtagttttaaatttaaatggtgaatttactatatcaaaa
                                        aactac               ttttaaattt
```


Possíveis promotores da seqüência 266:
ATG
tgcgaatctccatttcttttatattaattttgtctattttgggttttggtattaaaaaaaacaaaaaattaacaaaatttaattatttttagtaatttta
atctcc
attttgtcta
Possíveis promotores da seqüência 278:
ttaaaaagagcgatatttttttaaacttcactttctttcttgataaatttaataaaataagagacaataaataaatggaaaaagaagaaaatttaaactag
aattta
ataaataaat
Possíveis promotores da seqüência 297:
aaatcagttttccttttaaaatttcagtaaactaaggtgttttatgacgattcaattaaaaaagacaaactcgttccgctagcattacctaggtcaacta
tcaatt
gttccgctag
Possíveis promotores da seqüência 300:
ggatttatcatttagaagtagctcaagcccactaaaatttttatttgatcaagcccgagaaaaataactataaataaaataaccgccaatagcaaggcta
aactat
caatagcaag
Possíveis promotores da seqüência 313:
taaagttccttttgcttttcctaataaattagtaaatatcgtaatttctgacaatatttccgtttctatttatttcctgatcattttcagatatttctga
gacaat
tatttcctga
Possíveis promotores da seqüência 350:
agtttatcctttccaaaagatttttaaaaataattattagctaacattatataacaaaaaaattagcggcgctaatttttcccacattgtaacttccaagt
aacaaa
aattttccca
Possíveis promotores da seqüência 352:
ctaaaaactaaaacaactaaaactcaggcaaaaaacagcactttaaacactaaaacatctgcaaaaccgaaaacaaaagaaataaagaaagctagcaaag
atctgc
agaaataaag
Possíveis promotores da seqüência 359:
gtctccttataagtggctcgtctttaaggtgcataaatgcttgaaaacctttttagaacggttactattcggtaacttttttattatagcaaaattaaat
gtctcc
ctttaaggtg
Possíveis promotores da seqüência 368:
aaatctcctttgttatttttattaattatagcgcttaataattaaagaccaaatttttatatttttatatattataaaatataaagatataataaatt
taataa
tttatatttt
Possíveis promotores da seqüência 379:
tacattttgtggcttggtaacgtgggctggctttacagaattatcaacatttggtgaaaatccctctgagctttgtgccgctcttggatcaaataaattt
aacatt
tgagctttgt
Possíveis promotores da seqüência 394:
tttttcctttctaataattcaataaatttatttaaaacgcatttttagcgctaataaaaaaatttattaaagtatattatacagaaaaagaaattaattg
aaaaat
atacagaaaa
Possíveis promotores da seqüência 402:
gttattttatatatgaaaataatgtacctttggtacgaaataaaataaattaaactaaagaggaggctgtattaaaaagatccatgctatactcatttac
atttta
tacctttggt
Possíveis promotores da seqüência 404:
attgcctccatgcatttttttctactataaaaagcaaaaaagtgccatagaaaacaaaatgtggaaaatcgttttgattttccacttgaattattaacta
aacaaa
ttgattttcc
Possíveis promotores da seqüência 427:
tactataaataagaaaaatcctgacttagttgataccaatcaggattttacaaagaagttttaatttggagcagataacgggaatcgaacccgcatctta
accaat
aagttttaat
Possíveis promotores da seqüência 430:
ttctaaattgactggtgtcggagaccagacttgaactggcacagtctttaacgaccacaagcacctcaagcttgcgtgtctaccattccaccactccgac
aagcac
ctaccattcc
Possíveis promotores da seqüência 443:
agcctgatatctatatactaaatagtgccattaattctaacttttaatttttacaagtagctattagtaaagaaaactaccagaaattaaactaggatt
attcta
caagtagcta
Possíveis promotores da seqüência 449:
ttattgaaattactttgcattaaaaggctaaaaaaagatttttctgccatttatttacaacttattgacattaaaaacagtctaaaacaacccttttttg
aaatta
ctaaaaaaag
Possíveis promotores da seqüência 468:
tttgatttttaattaaaaagagcgatatttttttaaacttcactttctttttgataaatttaataagagacaataaataaatggaaaaagatgaaaattt
aattta
taaatggaaa
Possíveis promotores da seqüência 470:
aagcttctccttatgatttttttttcctaaaggtactgaattttcagatttaaaccttttggtatgagatggtttgaataaacgctaaaattatagcatt
aaggta
taaacctttt
Possíveis promotores da seqüência 507:
ttttttccagcttttttctgttttttttgaccttttattttttagtattactactaattataactaatttttgaatttttattatataaaaagaacaaaatt
attata
ttattatata
Possíveis promotores da seqüência 538:
ataatgctaaaatagctagtaaaataaagatttttttaagctttcatatttgtgtaattcttatatatctatgttataattatgtataattatttaaat
taaaat
gatttttta
Possíveis promotores da seqüência 547:
aaatcagttttccttttaaaatttcagtaaactaaggtgttttatgacgattcaattaaaaaagacaaactcgttccgctagcattacctaggtcaacta
tcaatt
gttccgctag
Possíveis promotores da seqüência 567:
ttttatttaaaggtgtcttataaataacaaatttaaggtatagtataacctaaaagtatttaaaaagctctgattatgtcacagaatattaaaactatta
aaggta
gtatttaaaa
Possíveis promotores da seqüência 594:
ttgcccctccctttggtcatctgggtttttgataatatatttataaaaaccaaggagttttaaaactcatttatcattataaattaaaagttttttaaat
taatat
ggagttttaa
Possíveis promotores da seqüência 596:
ttttctcgtttctttttaatatatttattattgtctatttttatttttttagtctgcaaaacatattcaatatttttttacacttaaatatatcacaacatat
atttta
aacatattca
Possíveis promotores da seqüência 640:
atttctccttttatttttatttttataacaaatgcaaaatttttcaaatgcataatattataaaataatatatatacaaataatatcataaaaacaatttaaaa
atttta
caaaattttt
Possíveis promotores da seqüência 642:
attttccttaatatttgtttttacttcctttggtttctaaaattaagactcttttaattagagcacacctaataaattcaccaaagtgctttttgttttt
taataa
ttttgttttt
Possíveis promotores da seqüência 649:
aatataacctttttccgagtctttctttttctcatccggactttaccgtcggtttctgaattacacagaatcagctttttgctcgtagacttttactaccg
tctcat
ggtttctgaa
Possíveis promotores da seqüência 685:
atattagtcccttcataagtattattgtatcaagaatatttttttatatatttttatatatttgaacctgtttaaaatatattgtaagtaattaatattta
tatcaa
atttttatat
Possíveis promotores da seqüência 689:
tggttcactagattcattatgtttgcttaattcaacatcaacttctttaatttctttagtattgttagtgtcgaaataataaactatttttttgtaatta
tcaact
gtattgttag
Possíveis promotores da seqüência 706:
aaattcaaaatttataagaattacttgaaaaaaaatttatatttgtttactattatttaagtctataaaaaaagtaagtcattttgttttttttaagtaa
atttta
aaaaaattta
Possíveis promotores da seqüência 735:
atttgttttatttcttttctattttaaaagtgtttgttaacactttgacttgcaagaatttagaaaaatgaaaaaagcaacctaagtagttgcttatata
aacact
ttagaaaaat
Possíveis promotores da seqüência 745:
ttttttttactccttaatgacatttttatgtatagttattatagcacctatgaaatattttactatttaaaacccaatactaaaattgtcagaattaatgt
tgaaat
acccaatact
Possíveis promotores da seqüência 767:
tcgtaaagacttcttagaccaaataaaagtttttgctaaagaagaaggaattgaaaacttagaactggcaaataaaaatgaaaataaaacttttaaagag
aagttt
gaattgaaaa
Possíveis promotores da seqüência 786:
aaagcttttccttttctattaaaatatattaaaaagcaaaataagttaatcaaattaacttatttactagaaatatagtgctattaaaaccactcaaaaat
aactta
gtgctattaa
Possíveis promotores da seqüência 801:
taaagcagtttctaaagctttaattgaagcaaaatctcaacataatccactttatcttgaagttgatgaaataaaagataaccgtctcttaatgaataa
atcttg
aagataaccg
Possíveis promotores da seqüência 812:
agctttaatcctaattttcttttattctttctatattatagtttatataaaatatacatgtataaagttagtaaaaaatcaatatttattattaaattat
aagtta
ttattattaa

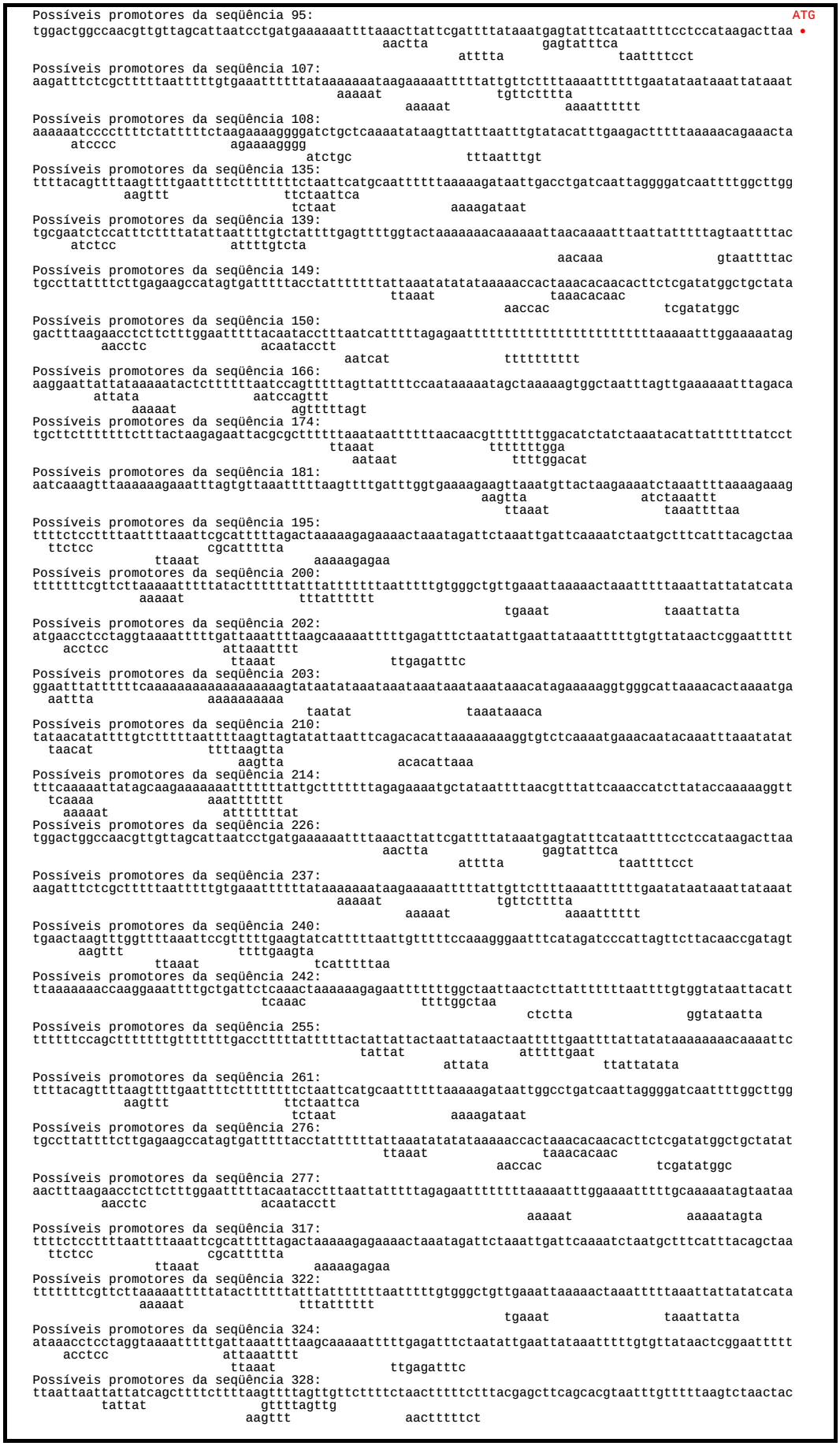

Possíveis promotores da seqüência 816:
ATG
aacagcaacacatgttgcaattgtatcgttttgaaagtagtagtgagtacatccgtcaacttctaaaacttgtccaaatttatagttattagccgttgct
tcaact
aatttatagt
Possíveis promotores da seqüência 823:
tgtgactccttatttgagtaaactctcatagattttattataggttgtcatagaaatttgctcattaatttctctttaccttacaataaagttcaagtac
atttta
gaaatttgct
Possíveis promotores da seqüência 837:
actactccttttttaaataaaaaaataaacactattttataacatagtgtttgtaatgatgcttattgttaatgctattataaaaaatataattcctaaaa
aacact
gtttgtaatg
Possíveis promotores da seqüência 855:
aattatttcctacctatattagtattagaaaataaaaataaaaaatagccaatttcaaaaaataaataacttataaaacaatatttattaaaactctttt
aaaaat
tcaaaaaata
Possíveis promotores da seqüência 857:
aattatttcctacctatattagtattagaaaataaaaataaaaaatagccaatttcaaaaaataaataacttataaaacaatatttattaaaactctttt
aaaaat
tcaaaaaata
Possíveis promotores da seqüência 863:
atattcgccttttttatttttattaatagtaaaactttaggtttttgatttcaacaatatattttacatatatatgtttatattttctaattttatgaatt
aacaat
atgtttatat
Possíveis promotores da seqüência 887:
ataataaatccctatgaattattaattacattataatgcttagatgttgtattaataaagattacaactaaatatattctttattcaattttataaccat
taataa
atattcttta
Possíveis promotores da seqüência 896:
cgtctagataatagatacagaacatggcttatttcattgacttagtatagggattatctccctatacttttttttatttcttctttagaaaaaaaattaggt
atctcc
ttcttcttta
Possíveis promotores da seqüência 927:
gtttttacctctatatttaattattaaaatttgtctttttttaagtcacacttacatatataaaaagattaatgattatttttaaatgcaaaaacacattcct
taaaat
cacacttaca
Possíveis promotores da seqüência 933:
aaaataactcctaaatttagatatcaaattaattatatttattaataatgtcttttatgaatttgatgaataatttttaagaaatttttatgttcttttagt
aataat
gttcttttag
Possíveis promotores da seqüência 934:
aaaacttatataaatacactatacataaagtaaagataactgttattatgagaagatcaatgatggtgaaccccacaagatagaggttccctttgaaata
aagata
agatcaatga
Possíveis promotores da seqüência 941:
tatttaatacctcaatagattaaatattatcacatttaattaatgagtgctaatgaaaataaacccattataaagataaaaaataaaaaaaataatcttg
tattat
agtgctaatg
Possíveis promotores da seqüência 988:
aatatttatttgtccttgttttatttgttttttattttttaaaaaatataaatttaacatttatataataaattaataatgaagttttcaaacatcaaaa
taataa
ttcaaacatc
Possíveis promotores da seqüência 1026:
aaagactattcttggccggataaacctttatgggatatgattaaaaatgataagggccacctgatgagtcatgaagtagggttattaagtttgttgatta
tatgat
ccacctgatg
Possíveis promotores da seqüência 1043:
agtaagggaaattttgcgtttaatggcaaagacctccttttatttattgattggaaaaaggtgaaagtgtttaaagaggtaggcaccaacaggtcactgag
acctcc
gaaaaaggtg
Possíveis promotores da seqüência 1048:
caacagtaaaaataactctttaattttgctttttttgtaaaaagccttttgggataagaacttgaattggaatttcgggttgaacattaattaaaacac
aaaaat
ctttttttgt
Possíveis promotores da seqüência 1053:
ttaaatcaggttagattttgcaatagtgcattactagattattctagttggctttaaggaaaggtttagcgcccctcaactagttgcaccaaccacttgt
aggtta
cattactaga
Possíveis promotores da seqüência 1056:
aaacagtaaaaataactctttgattttgctttttttttggaaaatccttttgaataagaggaaactaattgaaatttcgggtttaacattagttaaaaaat
tgaaat
agttaaaaaa
Possíveis promotores da seqüência 1063:
tgaaaagctcaaatcatacaggtcgtgtaaagcttctgacttttttgggaatagtattaatgaggtaattagtctttcactctttacgtttaacacccccc
atcata
ttctgacttt
Possíveis promotores da seqüência 1067:
cgtggtatctgttaattatacttttgccacccccttttccttttatgttgagagctaaaaaagacaaagcttggcagcaggttttatttagatcaaatct
attata
tttcctttta
Possíveis promotores da seqüência 1069:
taataaattcctcctttctgctcaaggtttagaaaactaaaacgtgcaactttgttgccagcaaggatcaacatcatcgttgatctaataaatatattga
taataa
tcaaggttta
Possíveis promotores da seqüência 1071:
taattattagcaatctcaaataaaaaatctgacaactaaaactgttggcaaaaccaaccaaaaggatatttaactgaggaaacctcgccccaaagtttaa
atctga
gcaaaaccaa
Possíveis promotores da seqüência 1074:
tggtgctaattctgtttaagtttaaagaatcaaacagcaactgttgttgtcgtagtttgactgacaacgccattttggtgcaaaaatagcttaaaaactt
tcaaac
cgtagtttga
Possíveis promotores da seqüência 1076:
ttaccgttggtaaatttatgtctttttagaccaaaaagttaactaactctttcaagtgcctaaaggctttttccttatttgttatctcagtgtcagcaact
tcaagt
cttatttgtt
Possíveis promotores da seqüência 1078:
caactacgttttattggttagcgcgggtaattacacccgtgacaaatataaatggtgattacagcaattacaagaatcaaaataataattcctaaagctaa
acaaat
gcaattacaa
Possíveis promotores da seqüência 1081:
acttaattaagcttaatttttttaactgaaccgctcatgatagttagattgtaagtaagtattacaagtactagtttgccaaagacaagtggttggtgca
aagtaa
gtttgccaaa
Possíveis promotores da seqüência 1117:
aaagtttctcctcgattttcataagacagagaagagtaagtggtaatttctaagacgaaatttaaagtctcgctacgctaatattaattagatcagctaa
aattta
aatattaatt
Possíveis promotores da seqüência 1132:
aaagttttgcctcctatttttgtggcaaaacttttttatataaaaaaataagtgccacatttaattttactcataatgtggtgcacttatatttctaa
aaaaat
attttactca
Possíveis promotores da seqüência 1139:
tatccaccactattatacatttattttccacttttatacttttttgttatttttttggataaagaaaatttttttgtataagctaaaaaagaaaggtttt
attata
tttatacttt
Possíveis promotores da seqüência 1147:
tgcaattttgttttaaaaaaataatttttttcaattttaaatctagtttttaagcaagattgtctaaatttgaaaaatgaaaataatatttaaaaaatta
ttaaat
gattgtctaa
Possíveis promotores da seqüência 1156:
tttttcctcttttctgcaaaactagaggatttaattcttttgaagttttcttaatctaatgatttcggaattaacagcatacttactttcaaatcttagg
aagttt
ttcggaatta
Possíveis promotores da seqüência 1162:
agttttgaaattatacttaaaatttttttctatagatatttttgttttaaaaagcctttgcttagcaaaagcataaattaaacgcgctataattcaaataa
taaaat
ttttgttta
Possíveis promotores da seqüência 1175:
ttttttttgcttgttttactaactccttttgtagagctaatttttaaattgatctatgaacttttgatcgataaaagtaagatcatttttttccaaaaaagac
aactcc
tttaaattga
Possíveis promotores da seqüência 1181:
acttatattatactctcttttgcacttttttcgacttccaaagcgccctcaaaaagccaaagcaaaaaaatatcaaagttttatttaatttttaagctaa
atatta
ttttttcgac
Possíveis promotores da seqüência 1201:
ttatggtttccttttttatttttatattaattatgttttgattcatagtcaactctacaaaataacatgacaagtgtaacggtctaagtcaaaatttttg
tcaact
acaagtgtaa
Possíveis promotores da seqüência 1220:
ttttacaatagcaaattattaaattcccttataataaaaaggtaaattgtttattcgatacatagcggacaaacaatttaatttttggccaatttaattat
gcaaat
taataaaaag
Possíveis promotores da seqüência 1243:
aatttttaaaaaacgatcattactataaatcgttccactatcatgcaaaaaacgagtgtcaaaactgcgctttttacgaaatgatttatttacaattaaa
tatcat
caaaactgcg

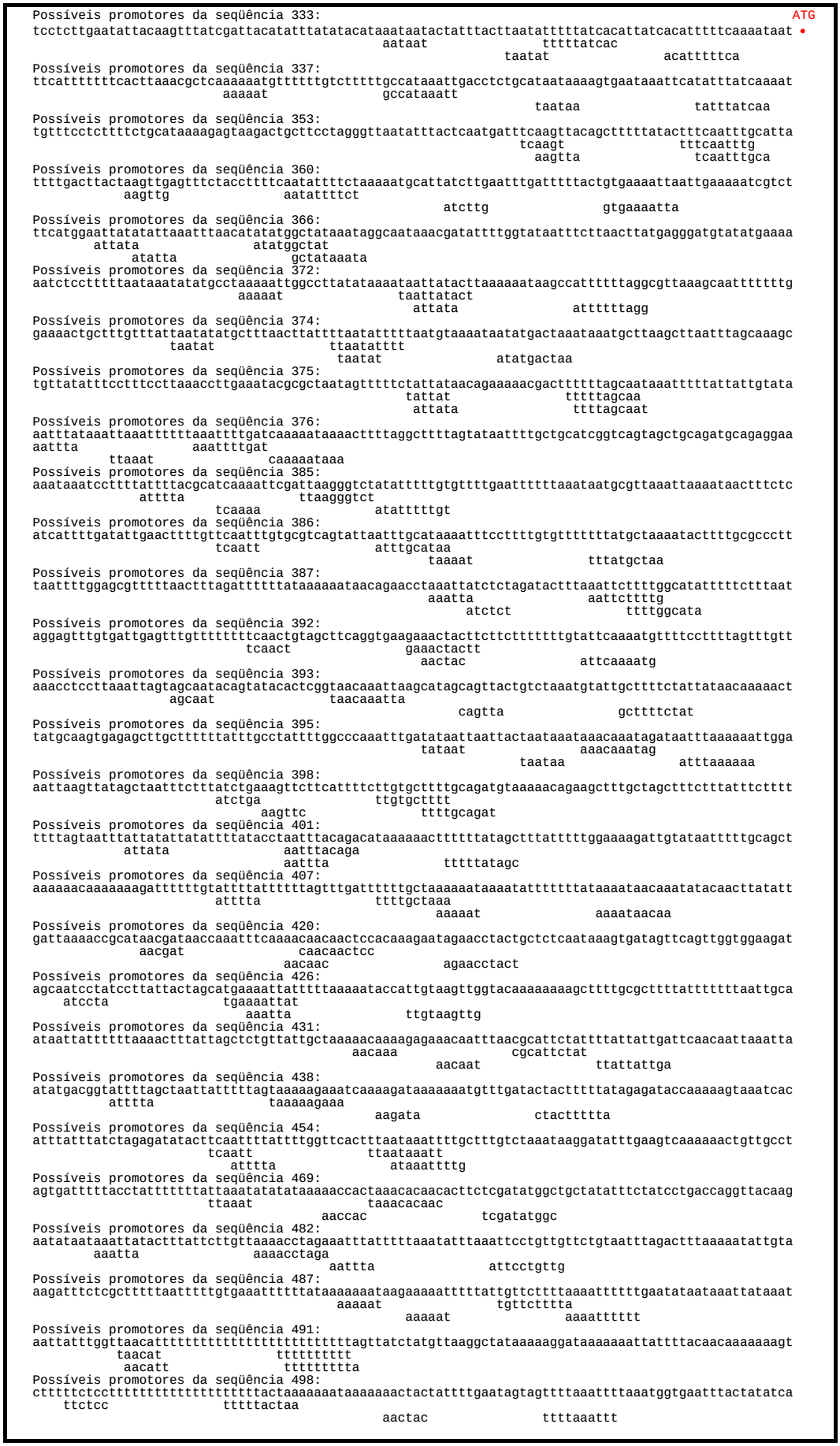

Possíveis promotores da seqüência 1255:
ATG
agcgtatccttgcttatttttcaaaaataagactaaatttcttgcgatggcttgataattaaatttgttaaaaacaataacaaagtttgcttttgtcagc
ttaaat
acaaagtttg
Possíveis promotores da seqüência 1260:
aaataatcatctctttttataaaatttcctgtttttcagtttaaaaaaaaaaaaaaaaaactaatttatttgtgtaaactgttaaactacacaaatt
aataat
aaattttcct

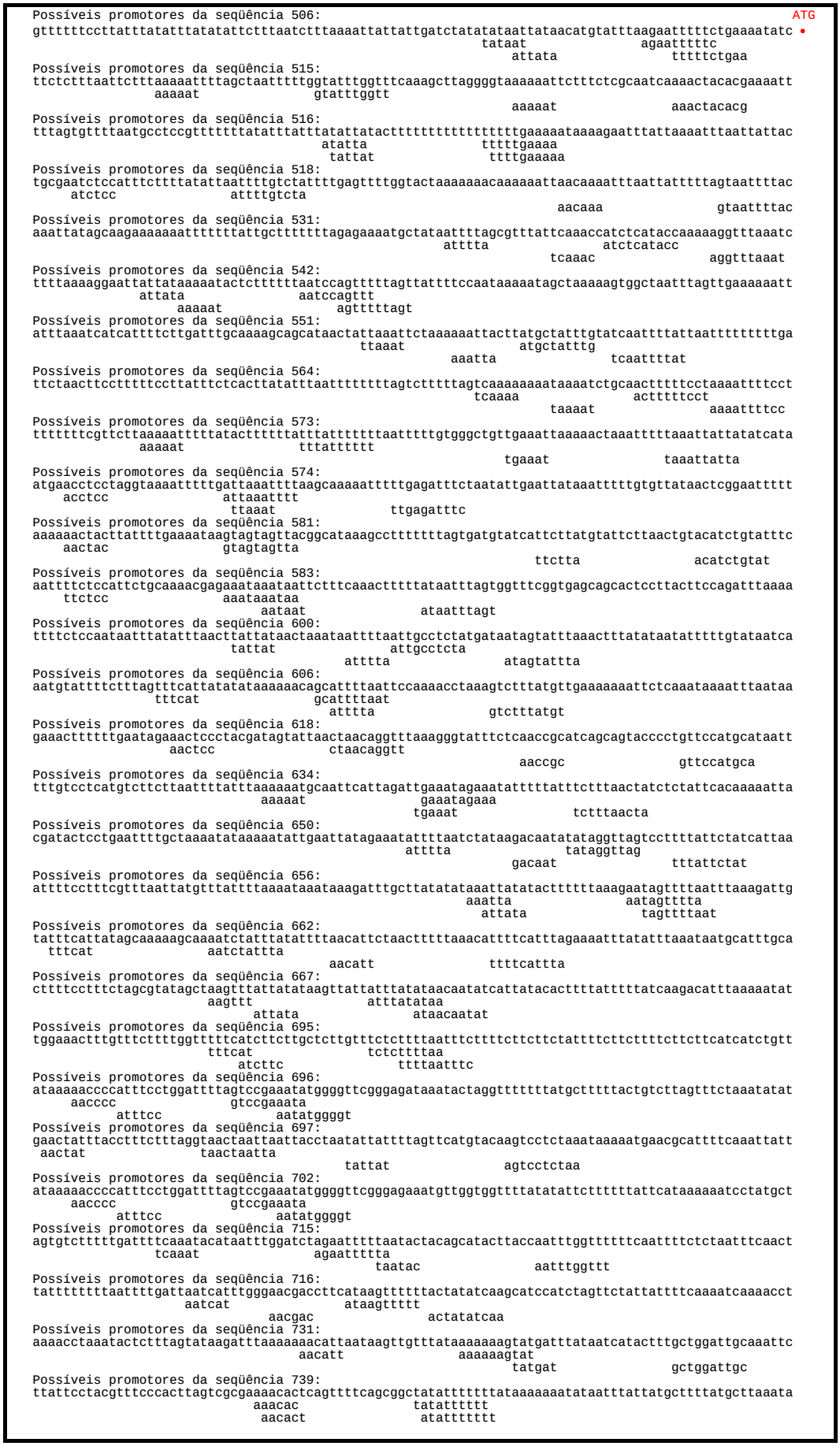
APÊNDICE F DUAS POSSÍVEIS POSIÇÕES E SEQUÊNCIAS DE PROMOTOR EM UMA SEQUÊNCIA ORIGINAL

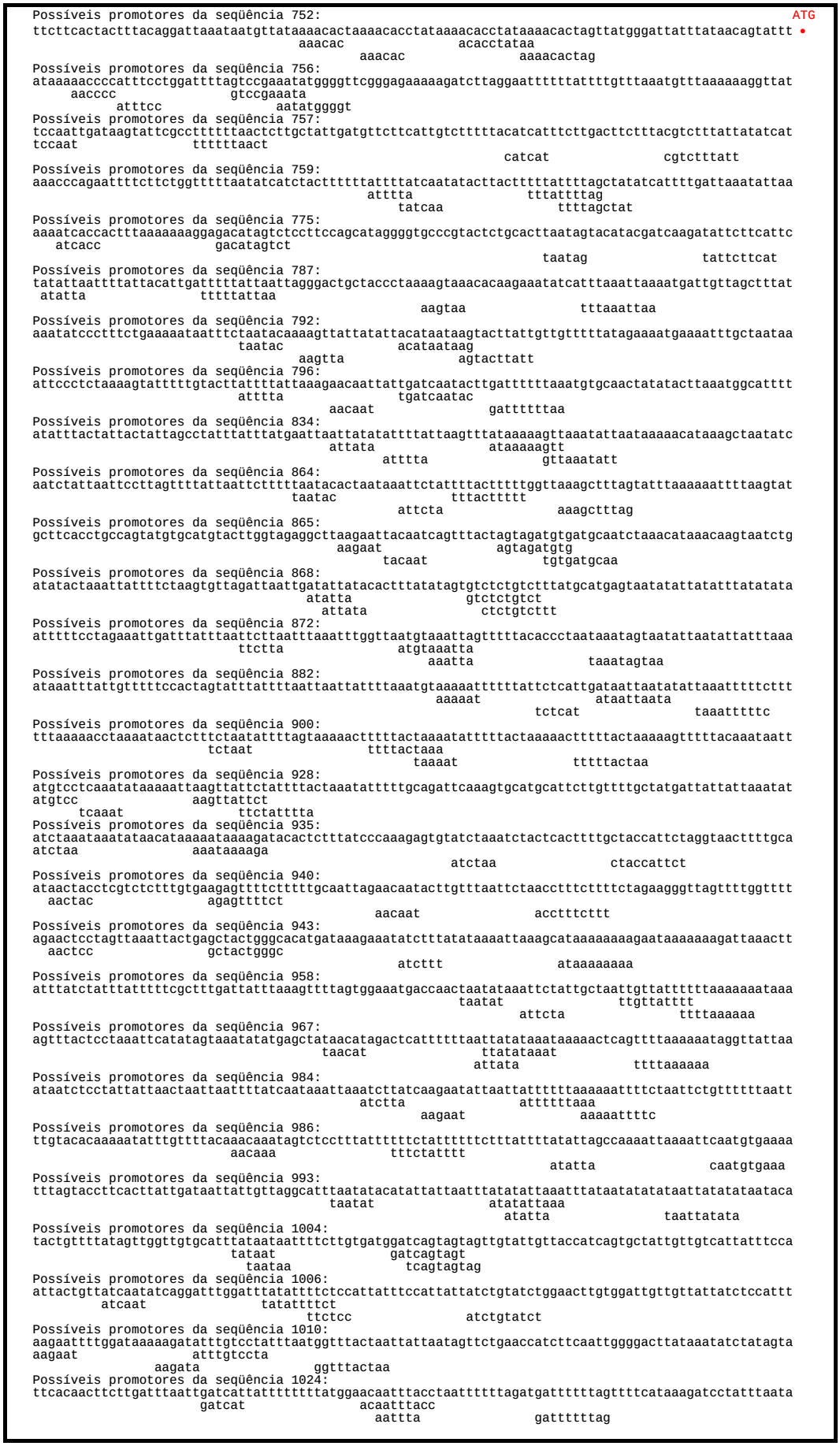
O quadro abaixo, apresenta a saída do programa que utilizou a combinação do modelo ART 1 treinado com ρ = 0,7 sobre a região -10 de 6 pb com outro modelo ART 1 treinado com ρ = 0,3 sobre a região -35 de 10 pb. Foram identificadas possíveis localizações de promotor em 165 seqüências das 1318 seqüências positivas e apresentadas somente seqüências em que foram encontradas 2 ocorrências de possíveis promotores. A numeração das seqüências é um identificador seqüencial referente às 1318 seqüências. Todas as seqüências possuem 100 pb e o ATG está situado no canto direito.

```
Possíveis promotores da seqüência 11:                                                                ATG
ggatcaactagaacgttaaccaacatgtggttatcacctagttcaaacaccctagaaggttgcatcttcactcttaagaaaagatttcccactagattct •
   tcaact               atgtggttat
                              atcacc               agaaggttgc
Possíveis promotores da seqüência 22:
ttattaaaaaaggttcaagttaatttccttaagatgttaacgataatggttgtcaatagtttttgttaagatatcttattatatataaaaattaagtaat
                                    aacgat               tttttgttaa
                                                               aagata               aaattaagta
Possíveis promotores da seqüência 25:
tctttaaattataggtaattatattatatttagtccttagattttccttatagagatgtaaaacgctatttgttgttgtcaaaattttacaaaagattat
   ttaaat               tatatttagt
                attata               agattttcct
Possíveis promotores da seqüência 28:
ttgttaattaatttcaaaaatctgtttcttaataatgccaaaaaacaaaattatttggaacaataagcatattctaatcaaatattgcttcagtttgtat
               aaaaat               gccaaaaaac
                          ttctta               aaaattattt
Possíveis promotores da seqüência 30:
ttgttattttaatcctcgggtatgaaaattttgacttatttatttttctgattaattaattaaatttttaattacattagacaaagcaaatatgtcttgtat
     atttta               aattttgact
                                                         atttta               gcaaatatgt
Possíveis promotores da seqüência 32:
gaactaatcttgttgtttcaagtttagatcttgttgttgtgtagttacgatttaagcaatgctcacaaatcaaaatatctttttacgcattaggcaaat
                                       tagtta               tcacaaatca
                                                  agcaat               taatcttttt
Possíveis promotores da seqüência 36:
ttaattatatttttaaaagttggataagcgataaataagaaatattggttagagtatgttggttagtatttgccgtgttttttggttgatctaaaaaac
   attata               taagcgataa
               aagttg               agaaatattg
Possíveis promotores da seqüência 37:
ctctcaaggctaatccatttataatttaatatgaatattttatccatttgatcatttataaatgttaatttataatatatcaatatgattattgttatt
                     atttta               ccatttgatc
                                                         atgtta               atatgattat
Possíveis promotores da seqüência 43:
ttaaaaccaacgttgtatcagcattagtcaaagctaatggtttgtaaatctggtcttttactattaataaaatctacactaaagtttataatattgaatt
                                                            taataa               ttataatatt
                                                               taaaat               taatattgaa
Possíveis promotores da seqüência 49:
ttaaaaaatgtacttatatagtatgtatataagtaataaggtaaaaggtataaaaagagcttccactttggaaactcaggttaattttgtaaaatatttt
                                  taataa               aagagcttcc
                                             aaggta               ctttggaaac
Possíveis promotores da seqüência 55:
ttaacgcataaaattataccaaattaagtgaagatttaaaaaggctagtgagattgtttagtttttggtagagggatttttaagataaataaaagttaga
        taaaat               gaagatttaa
          aaatta               agatttaaaa
Possíveis promotores da seqüência 63:
ccgattgaaattaacgaactaaagacacctataaaaataattgcttttaaacctcttgggttgaatgtgtaggtactaagcgcattattttcgatatatt
      aaatta               ctataaaaat
                                 aataat               tgggttgaat
Possíveis promotores da seqüência 71:
ttatttaatcaaaaagtaaaagctcattaaataactataaatcgattaatatacttaattataaaggaaaatttagattgatgcaaaggggtgcattaac
            aagtaa               ctataaatcg
                                          taatat               aaaatttaga
Possíveis promotores da seqüência 72:
tctttaatttagaatgtttagaaaaaaacgtaaaaaaggaaaatttagtcttggttaaatataggggataaaaagctttaaaatatagcgattatgttt
         tagaat               taaaaaagga
                                                ttaaat               ctttaaaata
Possíveis promotores da seqüência 79:
tttaagttagtatattaatttcagacacattaaaaaaaaggtgtctcaaaatgaaacaatacaaatttaaatatatcaaaattgctgaaactaaatggtt
   aagtta               acacattaaa
                                                aacaat               tcaaaattgc
```


Possíveis promotores da seqüência 95:
ATG
tggactggccaacgttgttagcattaatcctgatgaaaaaatttaaacttattcgattttataaatgagtatttcataattttcctccataagacttaa
aactta gagtatttca
atttta taatttcct
Possíveis promotores da seqüência 107:
aagatttctcgctttttaatttttgtgaaatttttataaaaaaataagaaaaatttttattgttcttttaaaatttttgaatataataaattataaat
aaaaat tgttctttta
aaaaat aaaatttttt
Possíveis promotores da seqüência 108:
aaaaaatccccttttctatttttctaagaaaaggggatctgctcaaaatataagttatttaatttgtatacatttgaagacttttaaaaacagaaacta
atcccc agaaaagggg
atctgc tttaatttgt
Possíveis promotores da seqüência 135:
ttttacagttttaagttttgaatttttctttttttttctaattcatgcaatttttttaaaaagataattgacctgatcaattaggggatcaattttggcttgg
aagttt ttctaattca
tctaat aaaagataat
Possíveis promotores da seqüência 139:
tgcgaatctccatttctttttatattaatttttgtctattttgagttttggtactaaaaaaacaaaaaattaacaaaatttaattatttttagtaattttac
atctcc attttgtcta
aacaaa gtaatttttac
Possíveis promotores da seqüência 149:
tgccttattttcttgagaagccatagtgatttttacctatttttttattaaatatatataaaaaccactaaacacaacacttctcgatatggctgctata
ttaaat taaacacaac
aaccac tcgatatggc
Possíveis promotores da seqüência 150:
gactttaagaacctcttctttggaatttttacaatacctttaatcatttttagagaatttttttttttttttttttttttttaaaaatttggaaaaatag
aacctc acaatacctt
aatcat ttttttttt
Possíveis promotores da seqüência 166:
aaggaattattataaaaatactctttttaatccagttttagttattttccaataaaaatagctaaaaagtggctaatttagttgaaaaaatttagaca
attata aatccagttt
aaaaat agtttttagt
Possíveis promotores da seqüência 174:
tgcttctttttttctttactaagagaattacgcgctttttttaaataatttttttaacaacgttttttttggacatctatctaaatacattatttttttatcct
ttaaat ttttttgga
aataat ttttggacat
Possíveis promotores da seqüência 181:
aatcaaagtttaaaaaagaaatttagtgttaaatttttaagttttgatttggtgaaaagaagttaaatgttactaagaaaatctaaatttttaaaagaaag
aagtta atctaaattt
ttaaat taaattttaa
Possíveis promotores da seqüência 195:
ttttctccttttaatttttaaattcgcatttttagactaaaaagagaaaactaaatagattctaaattgattcaaaatctaatgctttcatttacagctaa
ttctcc cgcattttta
ttaaat aaaaagagaa
Possíveis promotores da seqüência 200:
ttttttttcgttcttaaaaatttttatacttttttatttatttttttaatttttgtgggctgttgaaattaaaaactaaatttttaaattattatatcata
aaaaat tttattttt
tgaaat taaattatta
Possíveis promotores da seqüência 202:
atgaacctcctaggtaaaatttttgattaaatttttaagcaaaaatttttgagatttctaatattgaattataaatttttgtgttataactcggaatttttt
acctcc attaaatttt
ttaaat ttgagatttc
Possíveis promotores da seqüência 203:
ggaatttatttttttcaaaaaaaaaaaaaaaaaaagtataatataaataaataaataaataaataaacatagaaaaaggtgggcattaaaacactaaaatga
aattta aaaaaaaaaa
taatat taaataaaca
Possíveis promotores da seqüência 210:
tataacatattttgtctttttaatttttaagttagtatattaatttcagacacattaaaaaaaaaggtgtctcaaaatgaaacaatacaaatttaaatatat
taacat ttttaagtta
aagtta acacattaaa
Possíveis promotores da seqüência 214:
tttcaaaaattatagcaagaaaaaaatttttttattgctttttttagagaaaatgctataatttttaacgtttattcaaaccatcttatataccaaaaaggtt
tcaaaa aaatttttt
aaaaat attttttat
Possíveis promotores da seqüência 226:
tggactggccaacgttgttagcattaatcctgatgaaaaaatttaaacttattcgattttataaatgagtatttcataattttcctccataagacttaa
aactta gagtatttca
atttta taatttcct
Possíveis promotores da seqüência 237:
aagatttctcgctttttaatttttgtgaaatttttataaaaaaataagaaaaatttttattgttcttttaaaatttttgaatataataaattataaat
aaaaat tgttctttta
aaaaat aaaatttttt
Possíveis promotores da seqüência 240:
tgaactaagtttggttttaaattccgtttttgaagtatcatttttaattgtttttccaaagggaatttcatagatcccattagttcttacaaccgatagt
aagttt ttttgaagta
ttaaat tcatttttaa
Possíveis promotores da seqüência 242:
ttaaaaaaaccaaggaaattttgctgattctcaaactaaaaaagagaatttttttggctaattaactcttatttttttaattttgtggtataattacatt
tcaaac ttttggctaa
ctctta ggtataatta
Possíveis promotores da seqüência 255:
ttttttccagctttttttgttttttgaccttttattttactattattactaattataactaatttttgaattttattatatataaaaaaaacaaaattc
tattat attttgaat
attata ttattatata
Possíveis promotores da seqüência 261:
ttttacagtttaagttttgaatttttctttttttttctaattcatgcaatttttttaaaaagataattggcctgatcaattaggggatcaattttggcttgg
aagttt ttctaattca
tctaat aaaagataat
Possíveis promotores da seqüência 276:
tgccttattttcttgagaagccatagtgatttttacctatttttttattaaatatatataaaaaccactaaacacaacacttctcgatatggctgctatat
ttaaat taaacacaac
aaccac tcgatatggc
Possíveis promotores da seqüência 277:
aactttaagaacctcttctttggaatttttacaatacctttaattatttttagagaatttttttttaaaaatttggaaaatttttgcaaaaatagtaataa
aacctc acaatacctt
aaaaat aaaaatagta
Possíveis promotores da seqüência 317:
ttttctccttttaatttttaaattcgcatttttagactaaaaagagaaaactaaatagattctaaattgattcaaaatctaatgctttcatttacagctaa
ttctcc cgcattttta
ttaaat aaaaagagaa
Possíveis promotores da seqüência 322:
ttttttttcgttcttaaaaatttttatacttttttatttatttttttaatttttgtgggctgttgaaattaaaaactaaatttttaaattattatatcata
aaaaat tttattttt
tgaaat taaattatta
Possíveis promotores da seqüência 324:
ataaacctcctaggtaaaatttttgattaaatttttaagcaaaaatttttgagatttctaatattgaattataaatttttgtgttataactcggaatttttt
acctcc attaaatttt
ttaaat ttgagatttc
Possíveis promotores da seqüência 328:
ttaattaattattatcagcttttctttttaagttttagttgttctttttctaacttttttctttacgagcttcagcacgtaatttgtttttaagtctaactac
tattat gttttagttg
aagttt aacttttttct


Possíveis promotores da seqüência 333:
ATG
tcctcttgaatattacaagtttatcgattacatatttatatacataaataatactatttacttaatatttttatcacattatcacatttttcaaaataat
aataat
tttttatcac
taatat
acatttttca
Possíveis promotores da seqüência 337:
ttcatttttttcacttaaacgctcaaaaaatgtttttttgtctttttgccataaattgacctctgcataataaaagtgaataaattcatatttatcaaaat
aaaaat
gccataaatt
taataa
tatttatcaa
Possíveis promotores da seqüência 353:
tgtttcctctttctgcataaaagagtaagactgcttcctagggttaatatttactcaatgatttcaagttacagctttttatactttcaatttgcatta
tcaagt
tttcaatttg
aagtta
tcaatttgca
Possíveis promotores da seqüência 360:
ttttgacttactaagttgagtttctacctttttcaatattttctaaaaatgcattatcttgaatttgatttttactgtgaaaattaattgaaaaatcgtct
aagttg
aatattttct
atcttg
gtgaaaatta
Possíveis promotores da seqüência 366:
ttcatggaattatatattaaatttaacatatatggctataaataggcaataaacgatattttggtataatttcttaacttatgagggatgtatatgaaaa
attata
atatggctat
atatta
gctataaata
Possíveis promotores da seqüência 372:
aatctccttttttaataaatatatgcctaaaaattggccttatataaaataattatacttaaaaaataagccatttttttaggcgttaaagcaatttttttg
aaaaat
taattatact
attata
atttttagg
Possíveis promotores da seqüência 374:
gaaaactgctttgtttattaatatatgctttaacttattttaatatttttaatgtaaaataatatgactaaataaatgcttaagcttaatttagcaaagc
taatat
ttaatatttt
taatat
atatgactaa
Possíveis promotores da seqüência 375:
tgttatatttccttttccttaaaccttgaaatacgcgctaatagtttttctattataacagaaaaacgacttttttagcaataaatttttattattgtata
tattat
tttttagcaa
attata
ttttagcaat
Possíveis promotores da seqüência 376:
aatttataaattaaatttttttaaattttgatcaaaaataaaactttaggcttttagtataatttttgctgcatcggtcagtagctgcagatgcagaggaa
aattta
aaatttttgat
ttaaat
caaaaataaa
Possíveis promotores da seqüência 385:
aaataaatccttttattttacgcatcaaaattcgattaagggtctatatttttgtgttttgaatttttaaataatgcgttaaattaaaataactttctc
atttta
ttaagggtct
tcaaaa
atatttttgt
Possíveis promotores da seqüência 386:
atcattttgatattgaacttttgttcaatttgtgcgtcagtattaatttgcataaaatttccttttgtgtttttttatgctaaaatacttttgcgcccctt
tcaatt
atttgcataa
taaaat
tttatgctaa
Possíveis promotores da seqüência 387:
taattttggagcgtttttaactttagatttttttataaaaaaataacagaacctaaattatctctagatacttaaattcttttggcatatttttctttaat
aaatta
aattcttttg
atctct
ttttggcata
Possíveis promotores da seqüência 392:
aggagtttgtgattgagtttgttttttttcaactgtagcttcaggtgaagaaactacttttcttctttttttgtattcaaaatgttttccttttagtttgtt
tcaact
gaaactactt
aactac
attcaaaatg
Possíveis promotores da seqüência 393:
aaacctccttaaattagtagcaatacagtatacactcggtaacaaattaagcatagcagttactgtctaaatgtattgctttttctattataacaaaaact
agcaat
taacaaatta
cagtta
gcttttctat
Possíveis promotores da seqüência 395:
tatgcaagtgagagcttgctttttttatttgcctattttggcccaaatttgatataattaattactaataaataaacaaatagataatttaaaaaattgga
tataat
aaacaaatag
taataa
atttaaaaaa
Possíveis promotores da seqüência 398:
aattaagttatagctaatttctttatctgaaagttcttcattttcttgtgcttttgcagatgtaaaaacagaagctttgctagctttctttatttctttt
atctga
ttgtgctttt
aagttc
ttttgcagat
Possíveis promotores da seqüência 401:
ttttagtaatttattatattatattttatacctaatttacagacataaaaaactttttatagctttatttttggaaaagattgtataatttttgcagct
attata
aatttacaga
aattta
tttttatagc
Possíveis promotores da seqüência 407:
aaaaaacaaaaaaagatttttgtattttatttttagtttgatttttgctaaaaaataaaatatttttttataaaataacaaatatacaacttatatt
atttta
ttttgctaaa
aaaaat
aaaataacaa
Possíveis promotores da seqüência 420:
gattaaaaccgcataacgataaccaaatttcaaaacaacaactccacaaagaatagaacctactgctctcaataaagtgatagttcagttggtggaagat
aacgat
caacaactcc
aacaac
agaacctact
Possíveis promotores da seqüência 426:
agcaatcctatccttattactagcatgaaaattatttttaaaaataccattgtaagttggtacaaaaaaaagcttttgcgcttttatttttttaattgca
atccta
tgaaaattat
aaatta
ttgtaagttg
Possíveis promotores da seqüência 431:
ataattatttttaaaacttattagctctgttattgctaaaaacaaaagagaaacaatttaacgcattctattttattattgattcaacaattaaatta
aacaaa
cgcattctat
aacaat
ttattattga
Possíveis promotores da seqüência 438:
atatgacggtattttagctaattatttttagtaaaaagaaatcaaaagataaaaaaatgtttgatactacttttttatagagataccaaaaagtaaatcac
atttta
taaaaagaaa
aagata
ctacttttta
Possíveis promotores da seqüência 454:
atttatttatctagagatatacttcaatttttattttggttcactttaataaattttgctttgtctaaataaggatatttgaagtcaaaaaactgttgcct
tcaatt
ttaataaatt
atttta
ataaattttg
Possíveis promotores da seqüência 469:
agtgatttttacctatttttttattaaatatatataaaaaccactaaacacaacacttctcgatatggctgctatatttctatcctgaccaggttacaag
ttaaat
taaacacaac
aaccac
tcgatatggc
Possíveis promotores da seqüência 482:
aatataataaattatactttattcttgttaaaacctagaaatttatttttaaatatttaaattcctgttgttctgtaatttagactttaaaaatattgta
aaatta
aaaacctaga
aattta
attcctgttg
Possíveis promotores da seqüência 487:
aagatttctcgctttttaatttttgtgaaatttttttataaaaaaaataagaaaaatttttattgttctttttaaaatttttttgaatataataaattataaat
aaaaat
tgttctttta
aaaaat
aaaatttttt
Possíveis promotores da seqüência 491:
aattatttggttaacatttttttttttttttttttttttttttttagttatctatgttaaggctataaaaaggataaaaaaattattttacaacaaaaaaagt
taacat
ttttttttt
aacatt
ttttttttta
Possíveis promotores da seqüência 498:
ctttttctccttttttttttttttttttttactaaaaaaataaaaaaaactactattttgaatagtagttttaaatttaaatggtgaatttactatatca
ttctcc
tttttactaa
aactac
ttttaaattt


Possíveis promotores da seqüência 506:
ATG
gtttttccttatttatatttatatattctttaatctttaaaattattattgatctatatataattataacatgtatttaagaatttttctgaaaatatc
tataat agaatttttc
attata tttttctgaa
Possíveis promotores da seqüência 515:
ttctctttaattctttaaaaattttagctaatttttggtatttggtttcaaagcttaggggtaaaaaattctttctcgcaatcaaaactacacgaaaatt
aaaaat gtatttggtt
aaaaat aaactacacg
Possíveis promotores da seqüência 516:
tttagtgttttaatgcctccgttttttatatttatttatattatacttttttttttttttttttgaaaaataaaagaatttattaaaatttaattattac
atatta tttttgaaaa
tattat ttttgaaaaa
Possíveis promotores da seqüência 518:
tgcgaatctccatttcttttatattaattttgtctattttgagttttggtactaaaaaaacaaaaaattaacaaaatttaattatttttagtaattttac
atctcc attttgtcta
aacaaa gtaattttac
Possíveis promotores da seqüência 531:
aaattatagcaagaaaaaaattttttttattgctttttttagagaaaatgctataatttagcgtttattcaaaccatctcataccaaaaaggtttaaatc
atttta atctcatacc
tcaaac aggtttaaat
Possíveis promotores da seqüência 542:
ttttaaaaggaattattataaaaatactctttttaatccagtttttagttatttccaataaaaatagctaaaaagtggctaatttagttgaaaaaatt
attata aatccagttt
aaaaat agtttttagt
Possíveis promotores da seqüência 551:
atttaaatcatcatttttcttgatttgcaaaagcagcataactattaaattctaaaaaattacttatgctatttgtatcaatttttattaatttttttttga
ttaaat atgctatttg
aaatta tcaattttat
Possíveis promotores da seqüência 564:
ttctaacttccttttttccttatttctcacttatatttaatttttttttagtctttttagtcaaaaaaaataaaatctgcaacttttttcctaaaattttcct
tcaaaa acttttttcct
taaaat aaaatttttcc
Possíveis promotores da seqüência 573:
ttttttttcgttcttaaaaatttttatacttttttattattttttttaatttttgtgggctgttgaaattaaaaactaaatttttaaattattatatcata
aaaaat tttattttttt
tgaaat taaattatta
Possíveis promotores da seqüência 574:
atgaacctcctaggtaaaatttttgattaaatttttaagcaaaaatttttgagatttctaatattgaattataaatttttgtgttataactcggaatttttt
acctcc attaaatttt
ttaaat ttgagatttc
Possíveis promotores da seqüência 581:
aaaaaactacttattttgaaaataagtagtagttacggcataaagccttttttttagtgatgtatcattcttatgtattcttaactgtacatctgtatttc
aactac gtagtagtta
ttctta acatctgtat
Possíveis promotores da seqüência 583:
aattttctccattctgcaaaacgagaaataaataattcttttcaaacttttataatttagtggtttcggtgagcagcactccttacttccagatttaaaa
ttctcc aaataaataa
aataat ataatttagt
Possíveis promotores da seqüência 600:
ttttctccaataatttatatttaacttattataactaaataatttttaattgcctctatgataatagtatttaaacttttatataatatttttgtataatca
tattat attgcctcta
atttta atagtattta
Possíveis promotores da seqüência 606:
aatgtattttctttagtttcattatatataaaaaacagcattttaattccaaaacctaaagtctttatgttgaaaaaaattctcaaataaaatttaataa
tttcat gcattttaat
atttta gtctttatgt
Possíveis promotores da seqüência 618:
gaaacttttttgaatagaaactccctacgatagtattaactaacaggtttaaagggtatttctcaaccgcatcagcagtacccctgttccatgcataatt
aactcc ctaacaggtt
aaccgc gttccatgca
Possíveis promotores da seqüência 634:
tttgtcctcatgtcttcttaattttatttaaaaaatgcaattcattagattgaaatagaaatatttttatttctttaactatctctattcacaaaaatta
aaaaat gaaatagaaa
tgaaat tctttaacta
Possíveis promotores da seqüência 650:
cgatactcctgaattttgctaaaatataaaaatattgaattatagaaatattttaatctataagacaatatataggttagtccttttattctatcattaa
atttta tataggttag
gacaat tttattctat
Possíveis promotores da seqüência 656:
attttcctttcgtttaattatgtttatttttaaaataaataaagatttgcttatatataaattatatacttttttaaagaatagttttaatttaaagattg
aaatta aatagtttta
attata tagttttaat
Possíveis promotores da seqüência 662:
tatttcattatagcaaaaagcaaaatctatttatattttaacattctaacttttttaaacattttcatttagaaaatttatatttaaataatgcatttgca
tttcat aatctattta
aacatt ttttcattta
Possíveis promotores da seqüência 667:
cttttccttttctagcgtatagctaagtttattatataaagttattatttatataacaatatcattatacacttttattttttatcaagacatttaaaaatat
aagttt atttatataa
attata ataacaatat
Possíveis promotores da seqüência 695:
tggaaactttgtttcttttggtttttcatcttcttgctcttgtttctcttttaatttctttttcttcttcttttttcttcttttcttcttcatcatctgtt
tttcat tctcttttaa
atcttc ttttaatttc
Possíveis promotores da seqüência 696:
ataaaaaccccatttcctggattttagtccgaaatatggggttcgggagataaatactaggttttttttatgcttttttactgtcttagtttctaaatatat
aacccc gtccgaaata
atttcc aatatggggt
Possíveis promotores da seqüência 697:
gaactatttacctttctttaggtaactaattaattacctaatattatttttagttcatgtacaagtcctctaaataaaaatgaacgcattttcaaattatt
aactat taactaatta
tattat agtcctctaa
Possíveis promotores da seqüência 702:
ataaaaaccccatttcctggattttagtccgaaatatggggttcgggagaaatgttggtggttttatatattcttttttattcataaaaaatcctatgct
aacccc gtccgaaata
atttcc aatatggggt
Possíveis promotores da seqüência 715:
agtgtctttttgattttcaaatacataatttggatctagaatttttaatactacagcatacttaccaatttggttttttcaattttctctaatttcaact
tcaaat agaattttta
taatac aatttggttt
Possíveis promotores da seqüência 716:
tatttttttttaattttgattaatcatttgggaacgaccttcataagttttttactatatcaagcatccatctagttctattattttcaaaatcaaaacct
aatcat ataagttttt
aacgac actatatcaa
Possíveis promotores da seqüência 731:
aaaacctaaatactctttagtataagatttaaaaaaacattaataagttgtttataaaaaaagtatgatttataatcatactttgctggattgcaaattc
aacatt aaaaaagtat
tatgat gctggattgc
Possíveis promotores da seqüência 739:
ttattcctacgtttcccacttagtcgcgaaaacactcagttttcagcggctatatttttttataaaaaaatataatttattatgcttttatgcttaaata
aaacac tatatttttt
aacact atatttttttt


Possíveis promotores da seqüência 752:
ATG
ttcttcactactttacaggattaaataatgttataaaacactaaaacacctataaaacacctataaaacactagttatgggattatttataacagtattt
aaacac
acacctataa
aaacac
aaaacactag
Possíveis promotores da seqüência 756:
ataaaaaccccatttcctggattttagtccgaaatatggggttcgggagaaaaagatcttaggaatttttattttgtttaaatgtttaaaaaaggttat
aacccc
gtccgaaata
atttcc
aatatggggt
Possíveis promotores da seqüência 757:
tccaattgataagtattcgccttttttaactcttgctattgatgttcttcattgtctttttacatcatttcttgacttctttacgtctttattatatcat
tccaat
ttttttaact
catcat
cgtctttatt
Possíveis promotores da seqüência 759:
aaacccagaattttcttctggtttttaatatcatctacttttttattttatcaatatacttacttttttattttagctatatcattttgattaaatattaa
atttta
tttattttag
tatcaa
ttttagctat
Possíveis promotores da seqüência 775:
aaaatcaccactttaaaaaaaggagacatagtctccttccagcataggggtgcccgtactctgcacttaatagtacatacgatcaagatattcttcattc
atcacc
gacatagtct
taatag
tattcttcat
Possíveis promotores da seqüência 787:
tatattaattttattacattgatttttattaattagggactgctaccctaaaagtaaacacaagaaatatcatttaaattaaaatgattgttagctttat
atatta
ttttattaa
aagtaa
tttaaattaa
Possíveis promotores da seqüência 792:
aaatatccctttctgaaaaataatttctaatacaaaagttattatattacataataagtacttattgttgttttatagaaaatgaaaatttgctaataa
taatac
acataataag
aagtta
agtacttatt
Possíveis promotores da seqüência 796:
attccctctaaaagtatttttgtacttatttattaaagaacaattattgatcaatacttgatttttaaatgtgcaactatatacttaaatggcatttt
atttta
tgatcaatac
aacaat
gatttttaa
Possíveis promotores da seqüência 834:
atatttactattactattagcctatttatttatgaattaattatatatttttattaagtttataaaaagttaaatattaataaaaacataaagctaatatc
attata
ataaaaagtt
atttta
gttaaatatt
Possíveis promotores da seqüência 864:
aatctattaattccttagttttattaattctttttaatacactaataaattctattttactttttggttaaagctttagtatttaaaaaatttaagtat
taatac
tttacttttt
attcta
aaagctttag
Possíveis promotores da seqüência 865:
gcttcacctgccagtatgtgcatgtacttggtagaggcttaagaattacaatcagtttactagtagatgtgatgcaatctaaacataaacaagtaatctg
aagaat
agtagatgtg
tacaat
tgtgatgcaa
Possíveis promotores da seqüência 868:
atatactaaattattttctaagtgttagattaattgatattatacactttatatagtgtctctgtctttatgcatgagtaatatattatatttatatata
atatta
gtctctgtct
attata
ctctgtcttt
Possíveis promotores da seqüência 872:
atttttcctagaaattgatttatttaattcttaatttaaatttggttaatgtaaattagtttttacaccctaataaatagtaatattaatattatttaaa
ttctta
atgtaaatta
aaatta
taaatagtaa
Possíveis promotores da seqüência 882:
ataaatttattgtttttccactagtatttatttttaattaattattttaaatgtaaaaatttttattctcattgataattaatatattaaatttttcttt
aaaaat
ataattaata
tctcat
taaatttttc
Possíveis promotores da seqüência 900:
tttaaaaacctaaaataactctttctaatatttagtaaaaactttttactaaaatatttttactaaaaactttttactaaaaagtttttacaaataatt
tctaat
ttttactaaa
taaaat
tttttactaa
Possíveis promotores da seqüência 928:
atgtcctcaaatataaaaattaagttattctattttactaaatatttttgcagattcaaagtgcatgcattcttgttttgctatgattattattaaatat
atgtcc
aagttattct
tcaaat
ttctatttta
Possíveis promotores da seqüência 935:
atctaaataaatataacataaaaataaaagatacactctttatcccaaagagtgtatctaaatctactcacttttgctaccattctaggtaacttttgca
atctaa
aaataaaaga
atctaa
ctaccattct
Possíveis promotores da seqüência 940:
ataactacctcgtctctttgtgaagagttttcttttgcaattagaacaatacttgtttaattctaacctttctttttctagaagggttagttttggtttt
aactac
agagttttct
aacaat
acctttcttt
Possíveis promotores da seqüência 943:
agaactcctagttaaattactgagctactgggcacatgataaagaaatatctttatataaaattaaagcataaaaaaaaagaataaaaaaagattaaactt
aactcc
gctactgggc
atcttt
ataaaaaaaa
Possíveis promotores da seqüência 958:
atttatctatttatttttcgctttgattatttaaagtttagtggaaatgaccaactaatataaattctattgctaattgttatttttttaaaaaaataaa
taatat
ttgttatttt
attcta
ttttaaaaaa
Possíveis promotores da seqüência 967:
agtttactcctaaattcatatagtaaatatatgagctataacatagactcatttttttaattatataaataaaaactcagttttaaaaaaataggttattaa
taacat
ttatataaat
attata
ttttaaaaaa
Possíveis promotores da seqüência 984:
ataatctcctattattaactaattaatttttatcaataaattaaatcttatcaagaatattaattatttttttaaaaaatttttctaattctgttttttaatt
atctta
atttttaaa
aagaat
aaaaattttc
Possíveis promotores da seqüência 986:
ttgtacacaaaaatatttgttttacaaacaaatagtctccttttatttttttctatttttttctttatttatattagccaaaattaaaattcaatgtgaaaa
aacaaa
tttctatttt
atatta
caatgtgaaa
Possíveis promotores da seqüência 993:
tttagtaccttcacttattgataattattgttaggcatttaatatacatattattaatttatatattaaatttataatatatataattatatataataca
taatat
atatattaaa
atatta
taattatata
Possíveis promotores da seqüência 1004:
tactgttttatagttggttgtgcatttataataatttttcttgtgatggatcagtagtagttgtattgttaccatcagtgctattgttgtcattatttcca
tataat
gatcagtagt
taataa
tcagtagtag
Possíveis promotores da seqüência 1006:
attactgttatcaatatcaggatttggatttatatttttctccattatttccattattatctgtatctggaacttgtggattgttgttattatctccattt
atcaat
tatatttttct
ttctcc
atctgtatct
Possíveis promotores da seqüência 1010:
aagaattttggataaaaagatatttgtcctattaatggtttactaattattaatagttctgaaccatcttcaattggggacttataaatatctatagta
aagaat
atttgtccta
aagata
ggtttactaa
Possíveis promotores da seqüência 1024:
ttcacaacttcttgatttaattgatcattatttttttttatggaacaatttacctaatttttttagatgatttttttagttttcataaagatcctatttaata
gatcat
acaatttacc
aattta
gatttttttag


Possíveis promotores da seqüência 1027:
ATG
attatcaattattaaaccttatatctatgcactatatgtattaattttcttttcaattcgtttacctcataccataatcattatttttaaattataaaaa
atcaat
ctatgcacta
acctta
tgtattaatt
Possíveis promotores da seqüência 1031:
tatatattctattcttttaattaaatttttattacataatttagattacaagaaaagatatttaacacatacaaatatcaattagagttaaataaaaaac
aattta
tatttaacac
aagata
tatcaattag
Possíveis promotores da seqüência 1032:
acctttaaccctcttttcaccctattttatgtaattaaaattaagatatctgttgttatttctacatttcttgtaatttatatagataaaagaaaggta
aaatta
tttctacatt
aagata
acatttcttg
Possíveis promotores da seqüência 1042:
atgtgctagttcaaagcttataaaactgaaaaggtaatagctttatctaactgttacccgacggtattagtttactatcttgatcgttttgaaaacttaa
agctta
aatagcttta
aaggta
tgttacccga
Possíveis promotores da seqüência 1045:
aaaggcttaactaattaatagttacagaaatgccttgacacgccgatgcttttatcgtttaaagttttgcagagctgaaaaagtggtttttttaaaaaaca
tatcgt
gctgaaaaag
aagttt
gtggtttttt
Possíveis promotores da seqüência 1050:
taattagaaattgtgtttaaagttaaattaaatctactttttttttcccaatatttgccaaaaaaatgcaaaatttttatttgcaatttgacctaacagcac
aagtta
tttttcccaa
aaaaat
caatttgacc
Possíveis promotores da seqüência 1052:
caacagtaaaaatagttcatcaattctgctttttttggaaaagcacgctcaaaaaagaacttcattgaaattttcggttgatttttttcttaaaaaagaa
aaaaat
ctttttttgg
atcaat
aaagcacgct
Possíveis promotores da seqüência 1054:
ttttagttaggcaactgtagatgtgtttaaagaaattaaggtaccgttatgaagactcactgaaccttgaccctgggtgctgacttgctgtttaaaggta
tagtta
gtttaaagaa
aaatta
gactcactga
Possíveis promotores da seqüência 1058:
caacagtaaaaataactctttaattttgttttttttggaaaagcaactttaaaaacggaaagctaattgaaatttcggtttgaacattaattaaaaaac
aagcta
tgaacattaa
tgaaat
aattaaaaaa
Possíveis promotores da seqüência 1068:
gttatagctaccatcagatagatcttcgattcaggcctttgctgttttaatcttatctttttttcaaatttcttttgaacaaagtaattgaaatttcgcgt
tatctt
tgaacaaagt
atcttt
gaacaaagta
Possíveis promotores da seqüência 1070:
caataaatagcaatcgtttatggcattaaagttgggtacactaccactagcaacagaggaaatttaaacgagatcttgcaatctcacctaagtttagcaa
aagttg
gcaacagagg
aattta
tctcacctaa
Possíveis promotores da seqüência 1082:
taaactagaccgactgggtcaagagaaggatgagctactcgaatttaaggagctcaaagatgaccttaacaacgtgattgtgattgataagagtagtgat
tcgaat
atgaccttaa
aattta
accttaacaa
Possíveis promotores da seqüência 1089:
tacacatttattcgctcctatctaaatgaatgtgtaaggagaatttaaagcgccaaacttttttatttgactacgtttctttttttgaacaggtaaataa
atctaa
gaatttaaag
aattta
ttatttgact
Possíveis promotores da seqüência 1090:
cagaaattatttttgcaatagcacaaacgatgataatcctcgataacttttttaaagttaaaaactcgttatttttctttggtttattgcaggttaataat
aacgat
ctttttttaaa
aagtta
ctttggttta
Possíveis promotores da seqüência 1091:
cctctttttcgttttttcttagtggttttttgtttgctttaaactatcgtgcaaacaacataaggttgtggctaaactcaaaaaagaagtaaagaaataa
ttctta
ctttaaacta
tatcgt
ttgtggctaa
Possíveis promotores da seqüência 1092:
agatgactttgatagttattagtaattaattagttaacagcccacttaaggtggttgttttggtttatttattataagtttgggggctgattttgtgggg
tagtta
ttaacagccc
tattat
ttttgtgggg
Possíveis promotores da seqüência 1093:
tactgatttttgaccaacaatttttgtaaatgcttcggtttgaccagcagctattacagcaatggcaataatgacagcaatcacaccaccacgtatttca
aacaat
ggtttgacca
aataat
accacgtatt
Possíveis promotores da seqüência 1094:
tggtaatttaagtaaataaagttaggcaacacactgatcaagctaaaaaaaccacttgtcacccttctaactaatgaattgcatgagtagagaaaaagtg
aattta
gcaacacact
aagtaa
acactgatca
Possíveis promotores da seqüência 1096:
caacgaacgtttagataggtttgaatacaagaaataacatttcttcaacagaaaacgccaacagttaacattcagttagctgttggtattttcattcttt
cagtta
ttggtatttt
taacat
tattttcatt
Possíveis promotores da seqüência 1103:
ttttttttgttgtagcttaaaatttttcatagcttttgtttaatgtgaaaaaggtctaaattatatatttttcaccaaattctttagcaagagtttttactca
agctta
tttgtttaat
aaatta
tctttagcaa
Possíveis promotores da seqüência 1106:
atcttttactattttacaaaatatcccataaaaaggccttaaataattaaggcaaaactagctttaaaacttaaaaacaactatttttagaatatagaa
atttta
aaaggcctta
aataat
tttaaaactt
Possíveis promotores da seqüência 1107:
atcatcttttgatatttctgaagcaaactttgatttgattaaaacttcacttttatttaaacttaaagctttaatttttctatttcgtttttttttctag
atcttt
aaactttgat
aactta
ctatttcgtt
Possíveis promotores da seqüência 1114:
ctttttttattaatataataggagttatttaaaacttgattttgattgatttttaagaaataaaaaagccaatctataaatatttatttaagataagta
taatat
aaacttgatt
taatag
tgattttgat
Possíveis promotores da seqüência 1115:
aatttatcttggaaaaaagctagcaaaaataaaagctactagcaaaaactaaagacaataaaactcttgctctttttaggttaaaaaacttagctttta
tatctt
aaaataaaag
aaaaat
aaactaaaga
Possíveis promotores da seqüência 1133:
aatataaatttttttcctttcttagaattattttgcttatgttttagtactaaaaccaaccaatgtatagactttcaactatatcctatccaattagtcaa
aaccaa
tcaactatat
aaccaa
ctatatccta
Possíveis promotores da seqüência 1135:
tgttctccttttgatttgataaataaatagttttaatatatcaataattgatattaaataatttttaacaatgtgactataataacacattttataaaggt
taatat
taaataattt
aacaat
cattttataa
Possíveis promotores da seqüência 1143:
taattccttttattatttgattttaaatatatttttttagtttgttgaatttcttcccaatggatcatttgactctagcaatttttctcaaaccacacccaa
tattat
ttttttagtt
ttaaat
ttgaatttct
Possíveis promotores da seqüência 1153:
cttgttcctttttttaatattgaaaataaatttctagttcgataaatatttttatactaaaatattgatccttctttgcccaataaaaaagaaaaatttcaa
taatat
agttcgataa
atttta
ccttctttgc


Possíveis promotores da seqüência 1200:
ATG
aattagtaatatttctcctcgtttttgtaatttagacagagaagataaaagattgatgtaattaaaattattttaccaaagactcgctacgctaatatta
aaatta
gctacgctaa
atttta
gctaatatta
Possíveis promotores da seqüência 1213:
agtttgaatatttgcttatatttctttttatttttgatgaaaaacaaatagttttactaaaaaaatttatattttagttattctttctttgatttaattt
aaaaat
tctttctttg
aattta
ttctttgatt
Possíveis promotores da seqüência 1226:
taattttggatttcctgatgcttgatttacatgtttttttagtaatttttctcaaaataaagctcaaatattttgtgtttaaattagaattaatgaacttg
atttcc
atgtttttta
tcaaaa
ttgtgtttaa
Possíveis promotores da seqüência 1238:
ctctaactcctatattaatataattagggaaaaataattattttgtatgaatataaaacataataacctttaaatacataataatgctaataattaggt
taatat
attattttgt
aataat
taaaacataa
Possíveis promotores da seqüência 1268:
ttattttactcctcttatggatattaatagctaattttgttgttttaaaattagcaagagtgtttagttcaataattattgttgaataaacgttaaaatt
atttta
ttaatagcta
aaatta
tcaataatta
Possíveis promotores da seqüência 1279:
accttcttgtattccttatcttttgttttaataattcatttaatatttattagtgattattctttaataaaaagcactaatttattatagttataataca
tatctt
atttaatatt
taatat
ctttaataaa
Possíveis promotores da seqüência 1290:
cttctatatccttactatttttatttctatgttaatgatactaattatttctaagtatattttattggtaatcaattatttttatcaaatataaaataag
atatcc
ctatgttaat
atttta
ttttatcaaa
Possíveis promotores da seqüência 1291:
attttctccttgattatgtgcaaaattcgctaaattgtcgtaaatatttttaacatactaagcatgttagaaaaatttttactatttttatattatataa
taacat
aaaatttta
atgtta
tttttatatt
Possíveis promotores da seqüência 1299:
actctctttttccttttttgtttgaattatttacttaaaacaacattactaagactcgtaataatgttgtaatataattattaaagttaaacattaatt
taataa
attaaagtta
aataat
ttaaagttaa
Possíveis promotores da seqüência 1308:
aattaaccaccacttattaactaaacttaaaaccaaaaataaacaattctggcttttgatcagcactttatttattattttactatatgtgcatatgct
caccac
taaaaccaaa
aacaat
gcactttatt

APÊNDICE G IDENTIFICADORES DE 84 SEQÜÊNCIAS ÚNICAS RELATIVAS AO BD OBTIDAS DA APLICAÇÃO DE FILTROS SOBRE O RESULTADO DO BLAST E A QUE ORGANISMOS REPRESENTAM

Tabela G1: Relação dos identificadores da seqüência do campo banco de dados, resultante da aplicação de filtros ao resultado do alinhamento com o programa BLAST sobre regiões IGR-*B*

Identificador da Seqüência do BD	**Organismo**
IGR-B-609788-610068-forward	*M. penetrans*
IGR-B-arcB-abc	*M. mycoides*
IGR-B-aspS-MSC_0433	*M. mycoides*
IGR-B-locus_tag=MSC_0059-tnpIS1634bk	*M. mycoides*
IGR-B-locus_tag=MSC_1002-tnpIS1634bg	*M. mycoides*
IGR-B-mgtE-tnpIS1634bz	*M. mycoides*
IGR-B-MH01202_MHJ0289-MH10073_MHJ0290	*M. hyopneumoniae J*
IGR-B-MH01488_rplJ-MH01491_MHJ0621	*M. hyopneumoniae J*
IGR-B-MH01522_dam-MH21622_MHJ0624	*M. hyopneumoniae J*
IGR-B-MH01776_parE-MH01768_gap	*M. hyopneumoniae J*
IGR-B-MH02172_MHJ0271-MH01375_pyrG	*M. hyopneumoniae J*
IGR-B-MH10358_MHJ0631-MH05103_plsC	*M. hyopneumoniae J*
IGR-B-MH10362_pdhD-1-MH04436_nagA	*M. hyopneumoniae J*
IGR-B-MH10370_rpsJ-MH13292_MHJ0192	*M. hyopneumoniae J*
IGR-B-MH12690_MHJ0398-MH12693_MHJ0400	*M. hyopneumoniae J*
IGR-B-MH12725_MHJ0424-MH03055_efp	*M. hyopneumoniae J*
IGR-B-MH15020_MHJ0322-MH21628_MHJ0323	*M. hyopneumoniae J*
IGR-B-MH15028_MHJ0502-MH07046_pdhC	*M. hyopneumoniae J*
IGR-B-MH15047_MHJ0021-MH04541_sipS	*M. hyopneumoniae J*
IGR-B-MH16399_MHJ0481-MH06764_MHJ0482	*M. hyopneumoniae J*

IGR-B-MH19654_rplK-tRNA-Leu	*M. hyopneumoniae J*
IGR-B-MH19822_MHJ0264-MH02115_pheS	*M. hyopneumoniae J*
IGR-B-MH21626_trpS-MH12995_MHJ0591	*M. hyopneumoniae J*
IGR-B-MH21632_MHJ0092-tRNA-Trp	*M. hyopneumoniae J*
IGR-B-mhp016-mhp017-MH232	*M. hyopneumoniae 232*
IGR-B-mhp027-mhp028-MH232	*M. hyopneumoniae 232*
IGR-B-mhp263-tRNA-His-MH232	*M. hyopneumoniae 232*
IGR-B-mhp311-mhp312-MH232	*M. hyopneumoniae 232*
IGR-B-mhp415-asnS-MH232	*M. hyopneumoniae 232*
IGR-B-mhp429-efp-MH232	*M. hyopneumoniae 232*
IGR-B-mhp502-aceF-MH232	*M. hyopneumoniae 232*
IGR-B-mhp641-mhp642-MH232	*M. hyopneumoniae 232*
IGR-B-MP02505_MHP0095-tRNA-Trp	*M. hyopneomoniae 7448*
IGR-B-MP02569_clpB-MP02573_tpiA	*M. hyopneomoniae 7448*
IGR-B-MP03337_dam-MP10391_MHP0626	*M. hyopneomoniae 7448*
IGR-B-MP03937_trpS-MP10334_MHP0594	*M. hyopneomoniae 7448*
IGR-B-MP05666_ktrA-MP05668_ktrB	*M. hyopneomoniae 7448*
IGR-B-MP06566_MHP0025-MP06560_sipS	*M. hyopneomoniae 7448*
IGR-B-MP09790_MHP0279-MP12688_pyrG	*M. hyopneomoniae 7448*
IGR-B-MP12640_rpsJ-MP10661_MHP0196	*M. hyopneomoniae 7448*
IGR-B-MP12655_rplJ-MP03313_MHP0623	*M. hyopneomoniae 7448*
IGR-B-MP12656_MHP0299-MP12690_MHP0300	*M. hyopneomoniae 7448*
IGR-B-MP12679_parE-MP12678_gap	*M. hyopneomoniae 7448*
IGR-B-MP12691_MHP0016-MP06686_MHP0017	*M. hyopneomoniae 7448*
IGR-B-MP18883_MHP0272-MP12636_pheS	*M. hyopneomoniae 7448*
IGR-B-MP18913_rplK-tRNA-Leu	*M. hyopneomoniae 7448*
IGR-B-MPN467-MPN468	*M. pneumoniae*
IGR-B-MPN505-MPN506	*M. pneumoniae*
IGR-B-MPN654-MPN655	*M. pneumoniae*
IGR-B-MS05312_MS0127-MS05308_secA	*M. synoviae*
IGR-B-MSC_0179-oppB	*M. mycoides*
IGR-B-MSC_0248-tnpIS1634ag	*M. mycoides*
IGR-B-MSC_0574-MSC_0575	*M. mycoides*

IGR-B-MSC_0784-tnpIS1634am	*M. mycoides*
IGR-B-MSC_0813-tnpIS1634chbz	*M. mycoides*
IGR-B-MSC_0818-MSC_0819	*M. mycoides*
IGR-B-MYPU_0050-MYPU_0060	*M. pulmoniae*
IGR-B-MYPU_0120-MYPU_0130	*M. pulmoniae*
IGR-B-MYPU_1030-MYPU_TRNA_LEU_1	*M. pulmoniae*
IGR-B-MYPU_4940-MYPU_4950	*M. pulmoniae*
IGR-B-parE-gap-MH232	*M. hyopneumoniae 232*
IGR-B-pdhD-nagA-MH232	*M. hyopneumoniae 232*
IGR-B-pheS-mhp107-MH232	*M. hyopneumoniae 232*
IGR-B-polC-1-MG032	*M. genitallium*
IGR-B-pyrG-mhp101-MH232	*M. hyopneumoniae 232*
IGR-B-rplJ-mhp639-MH232	*M. hyopneumoniae 232*
IGR-B-rpsF-mhp308-MH232	*M. hyopneumoniae 232*
IGR-B-rRNA-16S-mhp688-MH232	*M. hyopneumoniae 232*
IGR-B-rrnA-16S-tnpIS1634bv	*M. mycoides*
IGR-B-tnpAIS1296ds-MSC_0906	*M. mycoides*
IGR-B-tnpAIS1296mp-fbaA2	*M. mycoides*
IGR-B-tnpAIS1296px-MSC_0233	*M. mycoides*
IGR-B-tnpBIS1296ji-MSC_0214	*M. mycoides*
IGR-B-tnpIS1634ab-MSC_0539	*M. mycoides*
IGR-B-tnpIS1634ad-MSC_0521	*M. mycoides*
IGR-B-tnpIS1634ax-glk	*M. mycoides*
IGR-B-tnpIS1634cd-MSC_0922	*M. mycoides*
IGR-B-tnpIS1634ce-MSC_0872	*M. mycoides*
IGR-B-tRNA-Ser-mhp460-MH232	*M. hyopneumoniae 232*
IGR-B-tRNA-Trp-mhp284-MH232	*M. hyopneumoniae 232*
IGR-B-trpS-mhp610-MH232	*M. hyopneumoniae 232*
IGR-B-ushA-mhp652-MH232	*M. hyopneumoniae 232*
IGR-B-UU479-UU480	*Ureaplasma*
IGR-B-tRNA-Tyr-mhp399-MH232	*M. hyopneumoniae 232*

Informática UFRGS

PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

"Redes Neurais Aplicadas ao Reconhecimento de Regiões Promotoras na Família Mycoplasmataceae"

por

João Francisco Valiati

Tese apresentada aos Senhores:

Prof. Dr. Luis Otavio Campos Alvares

Profa. Dra. Irene Silveira Schrank (CBIOT/UFRGS)

Prof. Dr. Diego Gervasio Frías Suárez (UESC)

Vista e permitida a impressão.
Porto Alegre, 04 / 08 / 06.

Prof. Dr. Paulo Martins Engel,
Orientador.

Prof. Carlos Alberto Heuser
Coordenador do Programa de
Pós-Graduação em Computação-PPGC
Instituto de Informática - UFRGS