Universidade de Brasília - UnB
Instituto de Biologia – IB
Departamento de Biologia Celular
Programa de Pós-Graduação em Biologia Molecular
Laboratório de Biologia Molecular

Predição de RNAs não-codificadores no transcriptoma do fungo *Paracoccidioides brasiliensis* usando aprendizagem de máquina

Roberto Ternes Arrial

Orientador: Prof. Dr. Marcelo de Macedo Brígido
Co-Orientador: Dr. Roberto Coiti Togawa

Brasília-DF
2008

PREDIÇÃO DE RNAS NÃO-CODIFICADORES NO TRANSCRIPTOMA DO FUNGO *PARACOCCIDIOIDES BRASILIENSIS* USANDO APRENDIZAGEM DE MÁQUINA

Roberto Ternes Arrial

Orientador: Prof. Dr. Marcelo de Macedo Brígido
Co-Orientador: Dr. Roberto Coiti Togawa

Universidade de Brasília - UnB
Instituto de Biologia – IB
Departamento de Biologia Celular
Programa de Pós-Graduação em Biologia Molecular
Laboratório de Biologia Molecular

Dissertação de Mestrado
apresentada como requisito parcial à
obtenção do título de Mestre em Biologia
Molecular.

Brasília – DF
Abril de 2008

Banca Examinadora

Dr. Georgios Joannis Pappas Júnior – EMBRAPA Recursos Genéticos e Biotecnologia – Examinador externo
Profa. Dra. Maria Emília Machado Telles Walter – CIC/UnB – Examinador externo
Prof. Dr. Marcelo de Macedo Brígido – IB/UnB - Orientador
Dr. Roberto Coiti Togawa – EMBRAPA Recursos Genéticos e Biotecnologia – Co-orientador

Membro Suplente

Profa. Dra. Andrea Queiroz Maranhão – IB/UnB

Trabalho desenvolvido no Laboratório de Biologia Molecular
da Universidade de Brasília, sob orientação do
prof. Dr. Marcelo de Macedo Brígido, e parcialmente na
EMBRAPA Recursos Genéticos e Biotecnologia (Brasília-DF),
sob co-orientação do Dr. Roberto Coiti Togawa,
com apoio financeiro do Conselho Nacional de Desenvolvimento
Científico e Tecnológico (CNPq).

“Crê nos que buscam a verdade.
Duvida dos que a encontraram.”
André Gide

Agradecimentos

Agradeço aos meus pais, já que sem eles nem eu nem meus agradecimentos estariam aqui ☺

-À minha mãe Sônia, pela atenção, carinho e auxílio em diversos sentidos;

-Ao meu pai, Tairo, e sua esposa, Cátia, pela companhia, carinho, estímulo e pela compreensão das ausências que o mestrado impõe;

Ao meu orientador, Marcelo Brígido, que além de ser um cientista e professor exemplar, é também uma ótima pessoa, um amigo de agradável convivência;

Ao meu co-orientador, Roberto Togawa, que por unanimidade é muito querido por todos, possui muito conhecimento e não abre mão de compartilhá-lo e ajudar quem lhe pede auxílio;

À banca examinadora: Dra. Maria Emília e Dr. Gergios Pappas que, com visões diferentes porém complementares, contribuíram de forma significativa para o aprimoramento tanto da forma quanto do conteúdo dessa dissertação, além de proporcionarem uma argüição no evento da defesa que por si só foi para mim um grande aprendizado;

À minha namorada, Brunna, pelos quase sete anos de amor, uma convivência maravilhosa e por sua paciência;

Aos colegas de laboratório da Computação e da Biologia, pelas discussões construtivas, além de tolerarem dividir seus trabalhos com algoritmos que consomem quase 100% dos recursos das máquinas;

Aos funcionários da UnB, especialmente os que mantêm o funcionamento dos laboratórios, por tornar possíveis as atividades acadêmicas;

À Instituição e aos funcionários da EMBRAPA Recursos Genéticos e Biotecnologia, por ceder computadores de altíssimo nível que catalisaram a execução desse trabalho;

À minha ex-orientadora Leila Barros, que me ensinou praticamente tudo que sei sobre experimentos de bancada, e que muito ajudou no amadurecimento do meu raciocínio científico e biológico;

Aos meus amigos do GISNO, pela amizade incondicional mesmo após longos períodos de ausência;
Aos amigos da graduação (Bioamigos), pelas experiências maravilhosas de vida, a amizade, as discussões e as viagens (mesmo que tenham sido só no planejamento e na imaginação!);

Aos amigos que não se encaixam em nenhuma das classes acima: muito obrigado pela amizade, e espero contar com vocês por toda vida!

Agradeço muito a todos!

Sumário

Índice de Tabelas

Índice de Figuras

Lista de Abreviaturas

Pb: *Paracoccidioides brasiliensis*
PCM: Paracoccidioidomicose
EST: *Expressed sequence tag*
ORF: *Open reading frame* (fase aberta de leitura)
mRNA: RNA mensageiro
ncRNA: *Non-coding* RNA (RNA não-codificador)
RNP: Complexo ribonucleoprotéico
miRNA: microRNA
siRNA: *small interfering* RNA (RNA de interferência)
eRNA: RNA eferente
UTR: *Untranslated region* (região não traduzida do mRNA, podendo ser 3' ou 5')
AM: Aprendizado de máquina
nB: Naive Bayes
FASTA: FAST Alignments (formato de arquivo)
MVS: Máquinas de vetores de suporte
ROC: *Receiver operating characteristic*
AAC: Área abaixo da curva
VP: Verdadeiro positivo
FP: Falso positivo
VN: Verdadeiro negativo
FN: Falso negativo
dbXXX_OP : Banco de Dados XXX de transcritos com ORF Presente (onde "XXX" é a designação do banco)
dbXXX_OA : Banco de Dados "XXX" de transcritos com ORF Ausente (onde "XXX" é a designação do banco)
dbTR : Banco de Dados de Treinamento
dbCOD: Banco de dados de transcritos codificadores
dbNC: Banco de dados de transcritos não-codificadores
dbPB: Banco de dados de transcritos de Pb
dbRD: Banco de Dados de Seqüências de DNA geradas aleatoriamente

RESUMO

Paracoccidioides brasiliensis (Pb) é um fungo saprófito e dimórfico de importância clínica, pois seus propágulos, quando inalados por humanos, desencadeiam a doença conhecida como paracoccidioidomicose. No ano de 2005 foi publicado o transcriptoma do Pb, apontando diversos alvos potenciais de drogas, mas ainda assim uma parte significativa dos transcritos seqüenciados não possui proteínas homólogas identificadas. Esse trabalho sugere que alguns desses RNAs possam ser não-codificadores (ncRNAs), uma classe de moléculas biologicamente funcionais que no entanto não codificam para nenhum produto protéico. Para tanto foi feita uma abordagem exclusivamente computacional, utilizando exemplos conhecidos de mRNAs e ncRNAs para treinamento de dois algoritmos de aprendizado de máquina: *naive* Bayes (nB) e Máquinas de Vetores de Suporte (MVS). Diversos programas descritos na literatura e desenvolvidos localmente foram usados para obter propriedades dos transcritos e de seus produtos protéicos, de forma que os algoritmos de aprendizado de máquina fossem capazes de diferenciar satisfatoriamente um mRNA de um ncRNA. O uso de várias medidas de eficiência mostra que ambos algoritmos, MVS e nB, induziram classificadores que discriminam as duas classes de RNAs de forma muito eficiente, mas também indicam que o MVS possui uma vantagem significativa em relação à sua detecção de ncRNAs. Acurácia média mensurada por validação cruzada de 10 vezes para o MVS foi de 92,4%, e para o nB, 75,3%. Quando usados no transcriptoma de Pb, o MVS e o nB detectam, respectivamente, 970 e 262 ncRNAs, dos quais a maior parte é de transcritos sem anotação e *singlets*, duas características que apóiam a possibilidade de que esses transcritos sejam realmente ncRNAs. Comparações a programas relacionados mostram que o programa aqui descrito apresenta um ganho em velocidade computacional sem perda de acurácia. Foi desenvolvido nesse trabalho um programa computacional de análise *ab initio*, designado PORTRAIT, especializado em detecção de ncRNAs em transcriptomas de organismos pouco caracterizados.

PALAVRAS-CHAVE: RNAs não-codificadores; Aprendizagem de máquina; Máquinas de vetores de suporte; *Paracoccidoides brasiliensis*; transcriptoma.

ABSTRACT

Paracoccidioides brasiliensis (Pb) is a saprophytic and dimorphic fungus of clinical importance because its propagules, when inhaled by humans, cause the disease known as paracoccidioidomycosis. In the year 2005 the Pb transcriptome was published, pointing out several potential drug targets, but still a significative amount of sequenced transcripts lack identified homologous proteins. This work suggests that these RNAs may be non-coding RNAs (ncRNAs), a class of biologically functional molecules that do not code for any protein product. Aiming this, a strictly computational approach was made, using known examples of mRNAs and ncRNAs for training two machine learning algorithms: naive Bayes (nB) and Support Vector Machines (SVM). Several programs available from literature and locally developed were used to obtain properties from transcripts and its corresponding protein products, in such a way that machine learning algorithms could successfully discriminate between mRNA and ncRNA. Several efficiency measurements show that both algorithms, SVM and nB, induced classifiers able to efficiently discriminate the two classes of RNAs, and also indicate that SVM has a significative advantage regarding ncRNA detection. Mean accuracy as estimated by 10-fold cross-validation procedure was 92.4% for SVM and 75.3% for nB. When used in the Pb transcriptome, SVM and nB detect, respectively, 970 and 262 ncRNAs, of which the majority is composed of singlets and unnanotated transcripts, two characteristics that support the possibility that these transcripts are real ncRNAs. Comparison to related works indicates that the described program offers a computational speed improvement without hindering accuracy. This work describes the design of a computational program for *ab initio* analysis, named PORTRAIT, specialized in detection of ncRNAs in transcriptomes from poorly characterized organisms.

KEYWORDS: Non-coding RNAs; Machine Learning; Support Vector Machines; *Paracoccidoides brasiliensis*; transcriptome.

1.1 O fungo *Paracoccidioides brasiliensis*, a paracoccidioidomicose e o transcriptoma

Paracoccidioides brasiliensis (Pb) é um fungo saprófito, encontrado em forma de micélio, com distribuição geográfica restrita à América Latina. Esse fungo apresenta dimorfismo, pois seus propágulos, inalados por humanos, podem sofrer diferenciação celular, assumindo forma de levedura multinucleada e induzindo uma doença denominada paracoccidioidomicose (PCM). Estima-se que mais de 10 milhões de pessoas estejam infectadas com o fungo, embora apenas 2% desse total desenvolvam a forma aguda fatal ou crônica. O perfil majoritário de pacientes manifestando PCM nesses estágios é adulto do sexo masculino, trabalhador de zonas rurais e imunodeprimido. A PCM afeta o sistema retículo-endotelial, sendo que a forma crônica tem alta freqüência de envolvimento pulmonar e/ou mucocutâneo. A forma crônica severa multifocal pode causar também lesões granulomatosas no sistema nervoso central do paciente. Independente do órgão afetado, a doença evolui para formação de seqüelas com danos permanentes no indivíduo (Felipe *et al*, 2003; Felipe *et al*, 2005).

Embora com uma taxa relativamente baixa de mortalidade, a PCM tem alta taxa de morbidade, diminuindo a qualidade e o tempo de vida de indivíduos em idade de trabalho. Por este motivo, torna-se um problema com importância econômica e social (Shikanai-Yasuda *et al*, 2006), o que impulsiona pesquisas que buscam elucidar a biologia do fungo e da doença, e que apontam alternativas terapêuticas.

Atualmente muito da biologia do fungo já é conhecida. Sabe-se que a transição *in vitro* de micélio para levedura é desencadeada por uma mudança da temperatura do meio onde o fungo se encontra, sendo que a 26^{o} C sua forma é de micélio e a 36^{o} C, levedura, e a transição é reversível para os dois estados (Franco, 1987). Esse dado é importante porque 36^{o} C é a temperatura do corpo humano, e a patogenicidade do fungo é necessariamente dependente da transição do estado de micélio a levedura (San-Blas e Nino-Vega, 2001). Portanto, os mecanismos envolvidos na transdução de sinal e as vias dessa diferenciação celular são os alvos mais promissores para o desenvolvimento de novos alvos terapêuticos para PCM (Felipe *et al*, 2005).

O projeto "Genoma Funcional e Diferencial do Fungo *P. brasiliensis*" representa um dos maiores esforços de estudo do Pb, tendo sido desenvolvido por uma rede de diversos laboratórios da região Centro-Oeste (**https://dna.biomol.unb.br/Pb/)**. O objetivo do projeto foi caracterizar o transcriptoma por meio do seqüenciamento de cDNAs denominados ESTs

("expressed sequence tags"), que representam apenas uma parte ou a totalidade de um gene expresso. A motivação do projeto foi a necessidade de identificar genes diferencialmente expressos nos dois estágios morfológicos, auxiliando na elucidação do mecanismo de aquisição de virulência e selecionando potenciais candidatos a vacinas e alvos de drogas (Felipe *et al*, 2003; Felipe *et al*, 2005).

A importância de um projeto como esse torna-se clara frente ao atual contexto clínico de combate às micoses: novas linhagens presentes em pacientes imunodeprimidos, mais agressivas e resistentes às drogas disponíveis, colocam em evidência o desafio do desenvolvimento de novos terapêuticos (Amaral *et al*, 2005). Outro desafio é a cura definitiva da PCM, uma vez que a erradicação completa do *P. brasiliensis* dos pacientes acometidos é considerada impossível porque os antifúngicos apenas reduzem a quantidade de fungos carreados pelo paciente, podendo ocorrer futuramente um ressurgimento da doença, obrigando os pacientes a fazerem um acompanhamento médico periódico (Shikanai-Yasuda *et al*, 2006). Uma estratégia para obtenção de novos alvos terapêuticos é isolar genes vitais para o fungo, ou aqueles relacionados à sua virulência, e desenhar drogas que interfiram com o metabolismo dessa molécula ou com sua atividade. Idealmente, deve-se selecionar genes específicos do fungo (ausentes no hospedeiro) para diminuir os efeitos colaterais de um medicamento potencial.

Dos 6.022 transcritos seqüenciados pelo projeto do transcriptoma do Pb, 44,1% são contigs (consensos montados por sobreposição de seqüências) e 55,9% são singletons (seqüências inteiriças, sem nenhuma outra seqüência se sobrepondo a ela). O projeto apontou genes com potencial a novos alvos terapêuticos, como por exemplo, proteínas de parede celular, de membrana plasmática, componentes da maquinaria celular, de vias de sinalização e de metabolismo (Amaral *et al*, 2005). Ainda assim, talvez o potencial de informação gerado pelo transcriptoma do Pb não tenha sido completamente explorado: isso porque dos 6.022 genes identificados, apenas cerca de 69,4% possuem homólogos no banco de dados GenBank (Felipe *et al*, 2005). Isso implica em 1.824 genes expressos completamente desconhecidos, sem similaridade a nenhuma seqüência conhecida.

Alguns desses transcritos sem homólogos identificados podem codificar proteínas que são inovadoras, inéditas nos bancos de dados de proteínas, e que por isso não são detectadas por análise comparativa. Uma hipótese alternativa é que esses transcritos não estejam conservados em nível de estrutura primária e sim secundária, por isso ferramentas tradicionais de análise comparativa, como o Blast (Altschul *et al*, 1997), que levam em conta apenas estrutura primária, não conseguem detectar transcritos similares nos bancos de dados

(Mattick, 2004). Não é esperado que tal conservação ocorra nos RNAs mensageiros, mas a similaridade de estruturas secundárias é relevante e pode ser encontrada em uma classe de RNAs que vem recebendo atenção da literatura - os RNAs não-codificadores (ncRNAs) - discutidos na seção seguinte. Os ncRNAs são estáveis *in vivo* e biologicamente ativos, embora não codificando para nenhum produto protéico. Evidências acumuladas sugerem que além dos RNAs de transferência e ribossomal, os outros ncRNAs exercem funções diversas, como regulação de metabolismo de outras moléculas, participam de *splicing*, auxiliam no transporte de proteínas, edição de nucleotídeos, regulação de *imprinting* e estado da cromatina, entre outros (Mattick e Makunin, 2006).

O estudo de ncRNAs é interessante especialmente no desenvolvimento de novas drogas contra organismos patogênicos, já que essas moléculas podem ser reguladas com alta precisão e especificidade por outros RNAs exógenos (Reynolds *et al*, 2004). Ao contrário do que se ocorre com as proteínas, nos ncRNAs freqüentemente observam-se significativas alterações da seqüência primária com conservação de estrutura secundária (Torarinsson *et al*, 2006), fato que caracteriza os ncRNAs como moléculas organismo-específicas e potencialmente alvos de drogas altamente específicas com efeitos colaterais minimizados. Por exemplo, (Chen *et al*, 2007) reportam a realização de *delivery* de *short hairpin* RNAs (shRNAs – que são ncRNAs similares a microRNAs – veja seção seguinte) mediado por vetores virais em aves.

A caracterização de ncRNAs a partir do transcriptoma do *P. brasiliensis* fornece uma contribuição importante para maior compreensão da biologia deste fungo. Esta caracterização pode vir a auxiliar no desenvolvimento de novas vacinas ou drogas, além de servir para a identificação de novos ncRNAs de outros fungos e organismos, já que o acúmulo de exemplos na literatura é essencial tanto para aprimoramento do modelo teórico geral dos ncRNAs como para programas de bioinformática que se utilizam de exemplos conhecidos, como aprendizagem de máquina (discutido na seção 1.3).

1.2 RNAs não-codificadores

A partir da identificação do DNA como carreador da informação genética das células, foi estabelecido um paradigma na literatura científica, onde cada molécula teria sua função: a informação que o DNA carrega seria transcrita em RNA, responsável por mediar o fluxo de informações. O RNA, por sua vez, seria decodificado pelos ribossomos, sendo a mensagem que carrega convertida em uma seqüência protéica. A geração da proteína no final dessa via seria o auge do que foi denominado, ainda em 1952, de Dogma Central da Biologia Molecular

(Judson *apud* Eddy, 2001). Realmente o papel das proteínas é fundamental em praticamente todos os processos biológicos conhecidos, sendo ator principal ou coadjuvante em inúmeras reações e montagem de estruturas complexas. No entanto, nem todas as moléculas de RNA são mensageiras, ou seja, nem todas codificam para uma proteína. De fato, os RNA não-codificadores já haviam sido preditos por um dos criadores do Dogma Central: ele propôs a hipótese que uma molécula deveria fazer a intermediação entre os códons e os respectivos aminoácidos codificados, e que haveria uma vantagem evolutiva para que essa molécula fosse o RNA, devido à possibilidade de interação por complementariedade e por sua versatilidade (Crick *apud* Eddy, 2001). Mais tarde a "hipótese do adaptador" de Crick viria a ser confirmada, caracterizando os RNAs de transferência, ou tRNAs (Hoagland *et al*, 1958). Uma outra classe de RNAs não-codificadores identificados foi a de RNAs ribossomais, ou rRNAs, que unem-se a um complexo protéico para constituir os ribossomos (Zimmermann e Dahlberg *apud* Eddy, 2001).

Apesar de identificados e a eles serem atribuídos papel de grande importância ainda no início dos projetos que envolviam o seqüenciamento de genomas inteiros, a caracterização em massa dos RNAs não-codificadores foi abandonada e relegada para um segundo plano, talvez devido ao grande impacto da identificação do código genético e das dificuldades técnicas relacionadas a identificar essas moléculas pequenas, instáveis e pouco abundantes (Eddy, 2001). Como conseqüência, os estudos subseqüentes passaram a ter como foco principal os mRNAs e as proteínas que eles codificam, mesmo sendo esses estudos necessariamente dependentes do conhecimento das duas classes de RNAs não-codificadores que se conhecia, os tRNAs e rRNAs (Eddy, 2001). A análise bioinformática de genomas por muito tempo ficou restrita a busca por regiões intrínsecas de fases abertas de leitura, como códons de início e de término, sítios de *splicing* e de poliadenilação, e de motivos genômicos clássicos (Guigó *et al*, 2000). Essa abordagem foi bem-sucedida, identificando diversos genes putativos posteriormente confirmados experimentalmente (Souza *et al*, 2000), apesar de ser conservativa e não explorar diversas outras regiões genômicas e motivos que posteriormente viriam a ser caracterizados como importantes (Schattner *et al*, 2005; Brent e Guigó, 2004; Mathé *et al*, 2002). Devido ao sucesso e relativa facilidade de encontrar genes dessa forma, foi estabelecido um paradigma de prioridade às regiões de transcrição de mRNA, sendo as demais regiões denominadas "ilhas intergênicas", consideradas desinteressantes, silenciosas (sem transcrição), e até "*junk* DNA" (resquícios de genes mal-sucedidos, transposons, pseudogenes, com função apenas de proteção dos exons ou sem nenhuma função aparente) (Shabalina e Spiridonov, 2004; Mattick e Gagen, 2001; Mattick e Makunin, 2006). Os íntrons,

componentes integrantes das regiões codificadoras, também recebiam denominações de insignificantes ou sem nenhum papel biológico, e o consenso era de que essas moléculas eram degradadas tão logo fossem excisadas por *splicing* (Williamson, 1977; Mattick e Gagen, 2001).

Com o avanço de novas técnicas de seqüenciamento, os transcriptomas de diversos organismos puderam ser obtidos (Adams *et al*, 1991). O estudo de um transcriptoma tem como objetivo identificar genes expressos em um organismo submetido a determinada condição ou estágio de desenvolvimento (Adams *et al*, 1991). Os grandes projetos transcriptoma foram desenhados ainda sob o paradigma do Dogma Central clássico, sendo seu foco os RNAs codificadores de proteínas (Jamet, 2004). Novos algoritmos, escritos especificamente para processar esses dados, também são muito embasados no Dogma Central, identificando genes principalmente com base em fases abertas de leitura (Jamet, 2004; Eddy, 2002).

Mesmo sendo os projetos transcriptoma desenhados ainda de forma "proteinocêntrica" (foco na detecção de mRNAs e seus produtos protéicos), o acúmulo de dados mostrou uma tendência cada vez mais evidente, principalmente para eucariotos: a maioria dos transcritos não possui um produto protéico putativo, tanto em estudos bioquímicos como de bioinformática (Shabaliana e Spiridonov, 2004; Costa, 2005; Hayashizaki e Kanamori, 2004; Claverie, 2005). Esse fato intrigante obrigou os pesquisadores a voltarem sua atenção naquela outra classe de transcritos, há muito esquecida e ignorada, apenas relembrada eventualmente por descobertas ao acaso, como por exemplo, isolamento de complexos ribonucleoprotéicos (RNPs) (Scharl *apud* Eddy, 2001; Kedersha e Rome, 1986). Estudos em paralelo reforçavam a retomada dos estudos dos ncRNAs. Em *Saccharomyces cerevisiae*, a sondagem do potencial transcricional pela técnica de "northern blot" de grandes porções intergênicas, que inicialmente supunha-se silenciosas, apresentaram níveis transcricionais significativos (Olivas *et al*, 1997).

Embora ainda não haja consenso até hoje quanto a uma definição formal do que seja um RNA não-codificador (Mattick e Makunin, 2006), uma característica é comum a todos: além de não codificarem para produtos protéicos, eles possuem uma atividade biológica relevante e não servem como intermediário entre DNA e proteína (Costa, 2007). A dificuldade em classificar os ncRNAs é justificável frente à sua variedade de estruturas, funções e interações que exercem *in vivo* (tabela 1). Por exemplo, eles podem ou não passar por processo de maturação de RNA (adição de 5´-cap, *splicing*, poliadenilação), podem se localizar em núcleo, citoplasma ou ambos, ligar-se a outros RNAs, a proteínas ou serem

ativos sozinhos (Wahlestedt, 2006; Laurent III e Wahlestedt, 2007). Essa grande variedade de funções e estruturas dificultam a geração de um consenso de classificação na literatura.

Tabela 1. Classificação dos ncRNAs de acordo com função e tamanho.

Tipo de ncRNA	Exemplos	Funções	Tamanho
RNAs pequenos	snoRNAs, snRNAs, piRNAs, smRNAs	Modificação de RNAs-alvo, síntese de DNA telomérico, dinâmica da estrutura de cromatina, modulação da transcrição, papel estrutural, gametogênese	~20–300 nt
RNAs estruturais	miRNAs, siRNAs	Silenciamento pós-transcricional e interferência por RNA	~18–25 nt
RNAs médios e grandes (similares a mRNA)	Xist, roX, SRA RNA	Imprinting de DNA, inativação de cromossomo (X), desmetilação de DNA, transcrição de genes, geração de outras classes de ncRNA, outras funções	~300–10.000nt

Abreviaturas: snoRNA, *small nucleolar RNA*; snRNA, *small nuclear RNA*; piRNA, PIWI-*interacting RNA*; smRNA, *small modulatory RNA*; miRNA, *microRNA*; siRNA, *small interfering RNA*. Dados adaptados de (Costa, 2007).

Independente de classificações, a quantidade de ncRNAs identificados cresce rapidamente na literatura. As descobertas mais notáveis envolvendo RNAs estruturais estão relacionadas ao desenvolvimento do sistema nervoso, corroborando a observação de que a quantidade de regiões não-codificadoras é proporcional à complexidade aparente dos organismos (Presutti *et al*, 2006; Mattick e Gagen, 2001; Mercer *et al*, 2008). Além dos siRNA e miRNA de ocorrência natural nos organismos, foram desenvolvidos novos métodos artificiais baseados nesse mecanismo, envolvendo principalmente silenciamento de RNAs mensageiros alvo, que já são empregados com sucesso como ferramenta de estudo de genes (Reynolds *et al*, 2004).

1.2.1 Exemplos de ncRNA – evidências experimentais

Os bancos de dados já acumulam também uma grande quantidade de ncRNA médios a grandes, isolados bioquimicamente ou identificados computacionalmente. A tabela 2 exemplifica alguns dos transcritos de ncRNA encontrados na literatura.

Tabela 2. Exemplos de transcritos de RNAs não-codificadores similares a mRNA caracterizados experimentalmente.

Transcrito	Função/Descrição	Referência
B2 RNA	Repressor da RNA polimerase II.	Espinoza *et al*, 2007.
as*PHO5*	Transcrito em direção antisenso a *PHO5*, em sobreposição, incluindo até seu promotor. Quando presente em *Saccharomyces cerevisiae*, ativa a transcrição de *PHO5* por recrutamento de RNA Polimerase II e remodelagem da cromatina.	Uhler *et al*, 2007.
PINC	Expresso em glândulas mamárias de grávidas. Localização alternada (citoplasma/núcleo), de acordo com etapa do ciclo celular. Estimula sobrevivência à morte celular por involução das glândulas mamárias e inibição da proliferação celular induzida por carcinógenos.	Ginger *et al*, 2006
PWRN1	Expresso de forma monoalélica em cérebro fetal. Possível participação na Síndrome de Prader-Willi (desordem neurogenética).	Buiting *et al*, 2007.
*FMR*4	Ausente em portadores da síndrome do X-frágil, expresso no cérebro adulto e no coração e rins fetais. Superexpressão causa proliferação celular, e *knockdown* causa apoptose	Khalil *et al*, 2008
SatIII	Diversos transcritos que participam da resposta a vários tipos de estresses, seqüestrando e liberando proteínas, principalmente fatores de transcrição, formando corpúsculos em resposta ao estresse.	Valgardsdottir *et al*, 2008

1.2.2 Bancos de dados específicos de ncRNA

Os exemplos de ncRNAs acumulam-se rapidamente na literatura, determinados tanto por meios experimentais como computacionais. Para catalogar e disponibilizar essas seqüências, diversos bancos de dados foram criados especificamente fornecendo seqüências anotadas de ncRNAs. Exemplos desses bancos de dados incluem:

- NONCODE (He *et al*, 2008);
- RNAdb (Pang *et al*, 2005);
- ncRNAdb (Szymanski *et al*, 2007);
- fRNAdb (Kin *et al*, 2007);
- NPInter (Wu *et al*, 2006b);
- Rfam (Griffiths-Jones *et al*, 2005);
- snoRNA-LBME-db (Lestrade *et al*, 2006);
- miRBase (Griffiths-Jones *et al*, 2006).

Apesar dessa grande variedade de bancos, já se observa integração entre alguns deles, enquanto outros se mantêm específicos para determinados organismos ou tipos de ncRNAs.

1.2.3 ncRNAs - revisão de paradigma

Os novos estudos sobre o papel biológico que as moléculas de RNA exercem dentro da célula obrigaram a uma revisão dos antigos conceitos. Diversas evidências apontam que, diferente do que o antigo dogma propunha, muitos íntrons podem exercer atividades

biológicas antes de serem degradados, associando-se a proteínas para montar os RNPs (Sharp e Burge, 1997), ou guiando edição de bases de outros RNAs (Bachellerie *et al*, 2002), dentre outras funções (Mattick e Gagen, 2001; Michalak, 2006; Nakaya *et al*, 2007).

A análise dos transcriptomas mostrou que os conhecimentos atuais sobre o processo de transcrição não conseguem explicar as "origens genômicas" de diversos transcritos, indicando que diversos elementos e regiões genômicas, fatores de transcrição, sítios de reconhecimento, RNA polimerases e até o mecanismo transcricional em si podem não estar completamente caracterizados para diversos organismos (Cheng et al, 2005). Tomando como premissa que existem muitos sítios transcricionais desconhecidos, um novo tipo de ensaio de microarranjos, chamado *genome tiling*, investiga os níveis transcricionais de um genoma sem nenhum tipo de viés, ou seja, as regiões genômicas são usadas como sondas e interrogadas quanto presença ou ausência de transcrição independentemente de presença de promotor, repetições, fase aberta de leitura, ou qualquer outro motivo específico (Mockler *et al*, 2005). Usando essa nova técnica, as ilhas intergênicas humanas, antes consideradas silenciosas, mostram-se altamente ativas transcricionalmente, podendo ser responsáveis por mais de 60% dos transcritos (Cheng *et al*, 2005).

Além disso, estima-se atualmente que os RNAs codificadores representem pelo menos metade do volume transcricional global humano (Ravasi *et al*, 2006). Esse elevado volume transcricional levantou a hipótese de que esses transcritos são meros produtos de transcrição promíscua, resultado da ativação aleatória das RNA polimerases em sítios inespecíficos (Costa, 2007), e que as regiões genômicas que lhes são molde são meros "escudos" dos exons (Bouaynaya e Schonfeld, 2006). O contra-argumento é composto por evidências que esses ncRNAs, além de ter uma função identificada, muitas vezes são expressos de forma específica de acordo com tecido, estágio e/ou condição, além de sofrer regulação de outras proteínas e RNAs, fatos incompatíveis com um modelo de transcrição promíscua (Ravasi *et al*, 2006; Mercer *et al*, 2008; Ginger *et al*, 2006; Valgardsdottir *et al*, 2007).

A própria concepção de "gene", já problemática e abstrata segundo a visão tradicional, sob a ótica das novas descobertas (principalmente dos ncRNAs) assume uma definição ainda mais complexa: além de ser um segmento de cromossomo que contém informação para uma proteína ou RNA funcional, e que possui *enhancers*, silenciadores, promotores, operadores, região codificadora e sítios de poliadenilação (Nelson e Cox, 2004), talvez deva passar a abarcar também os íntrons funcionais (microRNAs e *small nucleolar RNAs*), as regiões transcritas sem nenhuma ORF aparente, as regiões *cis* e *trans* de regulação, e os transcritos em sobreposição e em orientação antisenso (Mattick e Makunin, 2006).

As novas descobertas não só convidam a uma revisão de antigos conceitos, como indicam novas formas de pensar sobre as interações e regulações que acontecem em nível celular. Levantamentos recentes sugerem uma correlação direta entre a quantidade de ncRNAs de um organismo e sua complexidade (Hüttenhofer *et al*, 2005; Mattick e Makunin, 2006; Mattick, 2004; Laurent III e Wahlestedt, 2007). Foi sugerido que a ocorrência de grande quantidade de ncRNAs é uma aquisição evolutiva recente, e que essa classe pode estar relacionada a um controle fino de expressão, comparado a um sistema digital onde cada transcrito pode mediar uma rede de informações no sistema, emitindo e sendo alvo de moléculas eferentes (os chamados eRNAs) (Mattick e Gagen, 2001; Laurent III e Wahlestedt, 2007). Uma interação semelhante já é observada em diversos sistemas que envolvem miRNAs e proteínas (Costa, 2007). Em um inovador estudo computacional, as regiões não codificadoras do genoma humano foram investigadas quanto à presença de pequenos motivos que se repetem com uma freqüência maior do que a esperada ao acaso (Rigoutsos *et al*, 2006). Os pesquisadores descobriram que esses motivos, denominados *pyknons*, possuem uma forte tendência a possuírem instâncias principalmente nas regiões 5´ UTR e 3´ UTR, e também em regiões codificadoras de transcritos. Essa interação de regiões codificadoras com não-codificadoras sugere um mecanismo que pode representar uma nova camada de regulação, espécie-específica, denominada "pyknoma" (Rigoutsos *et al*, 2006; Meynert e Birney, 2006).

1.2.4 Abordagens experimentais para detecção de ncRNAs

Atualmente os RNAs já são estudados de forma sistemática a partir de estratégias desenhadas para deliberadamente detectar dessas moléculas. Um método de detecção de RNAs independente de cauda poli-A, denominado *Rnomics*, foi feito em cérebro de camundongos, revelando diversos ncRNAs novos, inclusive alguns exclusivos desse órgão (Hüttenhofer *et al*, 2001). Essa nova técnica já foi aplicada a diversos outros organismos, identificando quantidades razoáveis de novos ncRNAs que passariam despercebidos por métodos tradicionais de detecção de RNAs. Além do *Rnomics*, foram descritas outras abordagens experimentais novas, para isolamento e caracterização de novos ncRNAs (Hüttenhofer e Vogel, 2006; Mercer *et al*, 2008).

Mesmo com técnicas direcionadas para o descobrimento de novos ncRNAs, não existe nenhum método que caracterize um RNA definitivamente como não-codificador porque a definição de ncRNA é não-afirmativa, ou seja, sua principal característica é que não codifica para proteínas. Diante da complexidade do mecanismo transcricional, onde os transcritos são produzidos de acordo com condições ambientais e metabólicas de um indivíduo, não há como

garantir que um RNA não codifique uma proteína a não ser que se testem todas as condições fisiológicas possíveis em todos os tecidos, o que logicamente é impossível (Frith *et al*, 2006; McCutcheon e Eddy, 2003).

Mesmo quando um transcrito é caracterizado como ncRNA, há ainda a tarefa complexa de definir sua função por estudos genéticos, como em ensaios de perda de função. A dificuldade disso é que eliminar a função de um ncRNA é uma tarefa mais complicada que fazê-lo em mRNAs, porque os ncRNAs são imunes a mudança de fase e mutações *nonsense* (Hershberg *et al*, 2003; Eddy, 2002).

Já o processo de rotular um RNA como mensageiro (ou seja, codificador) não é uma tarefa simples, mas é bem definida experimentalmente e conclusiva. Uma abordagem possível é expressar a proteína *in vitro*, produzir um anticorpo contra ela e, por meio de ensaios com esse mesmo anticorpo, mostrar que a proteína está sendo produzida pelo organismo em determinada condição experimental ou tecido (Frith *et al*, 2006).

1.2.5 Abordagens computacionais para detecção de ncRNAs

A detecção computacional de ncRNAs tem problemas similares aos que os métodos experimentais enfrentam. A biologia computacional também não possui uma definição formal para ncRNA, no entanto um critério mais ou menos geral é de que não possuem ORFs longas, e há ao longo de sua seqüência uma ocorrência de códons de parada maior do que a esperada (Wahlestedt, 2006). Outra característica é de que teriam uma conservação em nível de estrutura secundária e não primária, o que invalida a detecção de ncRNAs por meio de ferramentas tradicionais que são usadas para caracterizar similaridade de DNA (Rivas *et al*, 2001; Torarinsson *et al*, 2006; Pang *et al*, 2006). Estudos comparativos que incorporam análise de uso de códons, substituições sinônimas e não-sinônimas e energia mínima de dobramento também são bem-sucedidos na identificação de ncRNAs (Badger e Olsen, 1999; Torarinsson *et al*, 2006; Xue *et al*, 2005).

Os pequenos RNAs estruturais possuem um certo grau de conservação e características de energia mínima de dobramento e estrutura secundária que permitem estabelecer um certo padrão que os diferencia dos demais RNAs (Pang *et al*, 2006). Os ncRNAs longos, no entanto, não possuem nenhum tipo de sinal ou característica específica e inequívoca que permita diferenciá-los dos RNAs codificadores (Hüttenhofer *et al*, 2005; Zhang *et al*, 2005; Pang *et al*, 2006). Por isso, a tendência atual das abordagens bioinformáticas é recorrer a uma combinação de diversos métodos computacionais que caracterizem os ncRNAs por meio de diferentes princípios, e depois analisar integrativamente

todos os dados gerados para decidir (usando estatística avançada e/ou anotação manual) quais RNAs são codificadores e quais são não-codificadores (Liu *et al*, 2006; Frith *et al*, 2006).

A escolha do método de descoberta de ncRNAs é dependente também dos dados que estão disponíveis sobre o organismo. Esses dados limitam as perguntas que podem ser respondidas e impõem necessidades diferentes de validação. Por exemplo, a análise do genoma fornece prováveis transcritos, e a vantagem é de que o contexto genômico da seqüência pode ser avaliado para determinar o potencial codificador de um transcrito, melhorando a acurácia das predições. A desvantagem é que os transcritos necessitam validação experimental (como *northern blot* ou *PCR*, por exemplo), e a ausência de transcrição não é conclusiva, já que esse transcrito poderia ser expresso quando exposto a outras condições ambientais e fisiológicas (Wang *et al*, 2006; Rivas *et al*, 2001).

Por outro lado, a busca de ncRNAs no transcriptoma é um problema essencialmente de classificação: já se sabe que as seqüências são transcritas, e o objetivo é determinar se ocorre a tradução delas ou não. Um aspecto positivo é que não há dúvidas de que as seqüências realmente são transcritas em RNAs. No entanto, um transcrito predito computacionalmente como codificador exige validação experimental (como *western blot*, por exemplo), e a ausência de tradução não é conclusiva, já que esse transcrito poderia ser traduzido quando exposto a outras condições ambientais e fisiológicas (Frith *et al*, 2006; Liu *et al*, 2006).

1.2.5.1 Programas especialistas (*genefinders*)

Diversos programas otimizados em técnicas específicas foram criados para o problema de identificação de ncRNAs em organismos bem caracterizados, do qual se tem disponíveis o genoma, e/ou transcriptoma e dados de espécies filogeneticamente próximas. A maior vantagem desses programas é sua facilidade de implementação, já que geralmente exigem a instalação de apenas um programa, e a otimização dos parâmetros é mais simples, sendo que as opções padrão podem já ser as ideais. Entre as desvantagens está a demanda de muitos dados sobre o organismo (por exemplo, nenhum realiza predições com base apenas no transcriptoma), a detecção limitada de ncRNAs, já que esses algoritmos usam poucos métodos para suas predições, e o tempo de processamento exigido pode ser restritivo, dependendo do programa e do tamanho do genoma analisado.

Uma confusão facilmente encontrada na literatura é que esses programas especialistas seriam os identificadores definitivos de ncRNAs, e que seriam capazes de apontar todos os transcritos não-codificadores de dado organismo. No entanto, o que esses programas fazem é procurar por sinais específicos conservados entre espécies ou que pareçam favoráveis a

constituir estruturas secundárias estáveis. Os programas que exploram a estrutura secundária partem do pressuposto que esses motivos são importantes para a função e que por isso são conservados em ncRNA. O que se observa, no entanto, é a ocorrência dos mesmos motivos em RNAs codificadores (Frith *et al*, 2006). Além disso, a simples análise da estrutura secundária não é suficiente para detectar um ncRNA (Workman e Krogh, 1999), mesmo porque alguns ncRNAs, como por exemplo transcritos antisenso não-codificadores, atuam por complementaridade de base geralmente independente de motivos de estrutura secundária (Wahlestedt, 2006; Frith *et al*, 2006). Exemplos de algoritmos que usam esse racional são o ddbRNA (di Bernardo *et al*, 2003), QRNA (Rivas e Eddy, 2001), RNAprofile (Pavesi *et al*, 2004), MSARI (Coventry *et al*, 2004) e FastR (Zhang *et al*, 2005).

O programa GenoMiner (Castrignanò *et al*, 2006) utiliza uma abordagem diferente das citadas. A intenção do algoritmo é caracterizar um transcrito ou seqüência genômica submetida quanto ao potencial codificador. Isso é feito alinhando-se a seqüência fornecida a genomas completos já conhecidos, restringindo o espaço de busca. Em seguida, um alinhamento mais preciso é realizado entre as seqüências de entrada e os alvos caracterizados como homólogos mais prováveis. As seqüências alinhadas são caracterizadas por seu potencial codificador de acordo com uma adaptação do programa CSTminer (Castrignanò *et al*, 2004). A principal desvantagem do uso desse programa é a necessidade de disponibilidade de genomas de espécies relacionadas, limitando seu uso quase unicamente a organismos-modelo.

Diversos ncRNAs também foram identificados com o programa RNAGENiE (Carter *et al*, 2001), que usa duas camadas de aprendizagem de máquina treinadas no genoma de *Escherichia coli* para caracterização de seqüências. Um algoritmo de redes neurais e outro de máquinas de vetores de suporte são usados para classificar os dados de entrada de acordo com composição de nucleotídeos, energia de dobramento e motivos específicos que ocorrem em ncRNAs. A desvantagem desse programa é sua limitação quanto à generalização: como o algoritmo de aprendizagem foi treinado apenas no genoma de *E. coli*, e segundo ressalva dos próprios autores, seu uso é limitado a organismos procarióticos e archaea.

Apesar de não ser uma regra geral, é possível encontrar homologia entre ncRNAs analisando sua estrutura secundária. Com base nisso foi desenvolvido o algoritmo RSEARCH que, usando alinhamento local e estrutura secundária predita da seqüência de entrada, faz a busca por seqüências homólogas nos bancos de dados (Klein e Eddy, 2003). Esse programa é uma tentativa que visa implementar, para seqüências de RNA, um sistema de busca e alinhamento a banco de dados similar aos programas usados para proteínas e DNA, como

BLAST e FASTA (Altschul *et al*, 1997; Pearson, 1990). O alinhamento é feito por Gramática Estocástica Livre de Contexto (SCFG), que é uma formalização do algoritmo de modelos de Markov (Baldi e Brunak, 2001) adaptada ao problema da identificação dos ncRNAs. Incorporada ao algoritmo de alinhamento está uma matriz de substituição apropriada para RNAs denominada RIBOSUM, similar às matrizes usadas para proteínas, como por exemplo a BLOSUM (Henikoff e Henikoff, 1992). A idéia de gerar um programa para busca de homologia de RNAs é promissora; porém, conforme já foi discutido, informações sobre estrutura secundária não são suficientes para caracterizar um ncRNA, por isso pode-se inferir que esse programa poderia melhorar consideravelmente se outros atributos fossem incorporados à sua análise. Além disso, como a maioria dos programas que lida com estrutura secundária de RNA, o RSEARCH é bastante lento e oneroso computacionalmente.

Os algoritmos especializados na detecção de algumas classes de ncRNAs estruturais, como tRNAs e snoRNAs, são mais bem-sucedidos do que os programas citados acima, que são destinados à detecção de classes específicas de ncRNA. Uma possível razão para esse sucesso é a presença de sinais intrínsecos e extrínsecos dos RNAs estruturais, conforme discutido. Exemplos de algoritmos criados para a detecção de ncRNAs estruturais são o tRNAscan-SE, snoscan e snoGPS (Schattner *et al*, 2005). A detecção de miRNAs naturais, no entanto, é uma tarefa mais complicada, mas já existem algoritmos que o fazem com um grau aceitável de confiabilidade, como por exemplo o ProMIR-II (Nam *et al*, 2006). Espera-se que a precisão desses algoritmos aumente consideravelmente com o desenvolvimento de novos métodos e principalmente com o aumento dos bancos de dados, pois tanto a modelagem do problema como algumas abordagens e técnicas específicas se beneficiam de uma maior quantidade de exemplos disponíveis.

1.2.5.2 Alinhamento do transcriptoma ao genoma estrutural e filtragem

O alinhamento do transcriptoma ao genoma estrutural seguido de aplicação de filtros às seqüências constitui uma abordagem específica com métodos bioinformáticos conservadores que, usando informações de anotações, genomas estrutural e funcional, dados de organismos filogeneticamente relacionados e uma combinação de filtros e programas, escolhem entre os transcritos os candidatos mais prováveis a ncRNA. Apesar de algumas pequenas variações, um modelo metodológico comum pode ser delineado dessas abordagens: inicialmente o transcriptoma é alinhado ao genoma estrutural por programas específicos para esse fim, por exemplo, SIM4 (Florea *et al*, 1998) ou BLASTN (Altschul *et al*, 1997). Esse alinhamento fornece uma medida de qualidade do EST e também permite obter o contexto

genômico onde aquele transcrito está inserido, usualmente sendo extraídos 10Kb a montante e a jusante da seqüência. Esse fragmento genômico é analisado por programas localizadores de ORFs, por exemplo, GENSCAN (Burge e Karlin, 1997) ou GeneMark.hmm (Lukashin e Borodovsky, 1998). Se uma ORF for encontrada, essa seqüência é descartada como candidata a ncRNA. Os fragmentos restantes são submetidos à busca por similaridade a proteínas nos bancos de dados pelo programa BLASTX (Altschul *et al*, 1997), geralmente sendo mantidas as seqüências que não possuem *hits* para um *e-value* de 10^{-5}. A etapa final envolve uma análise mais fina, como anotação manual, busca por similaridades em bancos de ESTs, ou busca por padrões nos transcritos. As seqüências que passam por esse último filtro são anotadas como candidatos a ncRNA. Essa abordagem foi usada na anotação de camundongo (Numata *et al*, 2003), *Drosophila* (Inagaki *et al*, 2005), *Arabidopsis* (MacIntosh *et al*, 2001), e *Caenorhabditis elegans* (Deng *et al*, 2006), identificando, respectivamente, 4.280, 136, 39 e 100 seqüências de ncRNAs.

Essa abordagem tem como desvantagens a necessidade obrigatória do transcriptoma e genoma do organismo, a dependência de programas de predição gênica e também que espécies próximas sejam bem caracterizadas (para que proteínas homólogas sejam identificadas com precisão). Além disso, se o transcrito codificar para uma proteína que não possua similaridade com nenhuma outra nos bancos de dados, pode no final ser incorretamente classificado como não-codificador.

1.2.5.3 Abordagem de múltiplas variáveis

Os programas e protocolos atuais para detecção de proteínas em larga escala são bem confiáveis, com uma taxa de sucesso tão boa que essa é uma área já considerada bem estabelecida (Guigó e Brent, 2004; Mathé *et al*, 2002). Com a descoberta da larga distribuição e importância dos ncRNAs, houve um grande esforço em desenvolver programas para detecção de ncRNA com o mesmo desempenho daqueles destinados à detecção de RNAs codificadores (Wang *et al*, 2006). No entanto, o consenso hoje é que os ncRNAs não possuem um sinal intrínseco (contido no transcrito) nem extrínseco (presente no contexto genômico) forte e único que os distinga dos mRNAs, o que pode ser uma crítica aos programas especializados na detecção de um único sinal (Carter *et al*, 2001; Frith *et al*, 2006; Wang *et al*, 2006). Um método emergente hoje é a análise sinergística de diversas variáveis simultaneamente, por diversas técnicas e usando diferentes programas. A desvantagem dessa abordagem é que as técnicas são escolhidas *ad hoc*, de acordo com a "intuição" do pesquisador (Liu *et al*, 2006). A quantidade de programas que extraem os mesmos atributos é

grande, mas os atributos a extrair, em si, não são muitos, limitando a influência do fator "intuição do pesquisador" na pesquisa. Com isso, observa-se uma escolha mais ou menos uniforme dos critérios escolhidos para análise (Frith *et al*, 2006; Liu *et al*, 2006; Xue *et al*, 2005; Teramoto *et al*, 2005; Wang *et al*, 2006). Outra desvantagem é a necessidade de escolha e implementação de diversos programas e o tempo computacional dedicado ao processamento de cada um deles (Frith *et al*, 2006).

Após a escolha dos atributos, esses podem ser extraídos do conjunto pelos programas específicos, e a concordância dos programas é analisada estatisticamente, permitindo ao pesquisador atribuir o rótulo de transcrito potencialmente codificador ou não-codificador. Esse método foi usado para caracterizar os transcritos obtidos do projeto transcriptoma de camundongo versão 3, o FANTOM3 (Maeda *et al*, 2006). Todos os transcritos foram traduzidos, e nessas proteínas putativas foram usados como atributos o comprimento da ORF, um programa que detecta possíveis erros no seqüenciamento, dois programas para detecção de domínios funcionais de proteínas, dois programas para busca de proteínas homólogas, dois programas para detecção de região codificadora em cDNAs, e dois programas de genômica comparativa para busca de ncRNAs baseados em substituições sinônimas e não-sinônimas. Foram feitas análises de correlação entre os métodos, determinando a eficiência dos programas e a possibilidade de cada transcrito ser codificador ou não. Os autores encontraram muitos ncRNAs (um terço da quantidade total de transcritos), e a conclusão foi que essa abordagem não só é viável, como é preferível a usar apenas uma técnica, produzindo resultados com alta confiabilidade a partir do consenso gerado por diversos programas (Frith *et al*, 2006).

A abordagem multivariáveis pode ser também implementada em conjunto com algoritmos de aprendizagem de máquina (discutidos na seção seguinte). Nesse método também são usados dois conjuntos de exemplos extraídos da literatura, podendo ser, por exemplo, um conjunto composto por seqüências de mRNAs e outro, por seqüências de ncRNAs. Diversas propriedades independentes são obtidas desses exemplos, e essas são fornecidas ao algoritmo, que a partir desses exemplos deve "aprender" a diferenciar um conjunto do outro. Passada a fase de treinamento do algoritmo, esse pode agora ser usado para julgar a qual classe um transcrito desconhecido pertence, sendo que esse transcrito não foi apresentado ao algoritmo durante a fase de treinamento.

Essa estratégia foi usada para criar um programa discriminador de precursores verdadeiros e pseudo-precursores de miRNAs de humanos (Xue *et al*, 2005), para o auxílio no desenvolvimento racional de siRNAs como ferramenta para estudos de perda de função de

genes (Teramoto *et al*, 2005), e na diferenciação de mRNAs e ncRNAs longos e médios de camundongos (Liu *et al*, 2006). A taxa média de acerto nesses estudos foi de 90,9%, 72,3% e 94,5%, respectivamente.

1.3 Aprendizagem de máquina

O principal objetivo dos algoritmos de aprendizagem de máquina (AM) é melhorar o desempenho da acurácia de classificação de um programa a partir da experiência. Formalmente, um computador aprende a partir da experiência **E** com respeito a uma classe de tarefas **T** e a partir da medida de desempenho **D**, se seu desempenho em tarefas **T**, medido por **D**, melhora com a experiência **E** (Mitchell, 1997).

Algoritmos de AM são ideais para que se possa predizer ou classificar fenômenos automaticamente, a partir de dados de entrada volumosos e com ruídos, para os quais não estão estabelecidas regras discriminativas generalizáveis (Baldi e Brunak, 2001). O algoritmo de AM produz um modelo estatístico computacional a partir desse grande volume de dados, e esse modelo gerado é usado para fazer inferências sobre dados desconhecidos, produzindo predições sobre esses dados (Larrañaga *et al*, 2006). Os dados usados durante o treinamento ou "aprendizagem" devem inicialmente passar por um processo de uniformização e integração, seguido por eliminação de dados redundantes e *outliers* (Larrañaga *et al*, 2006). Em seguida, deve ser feita a seleção das características a extrair (os chamados atributos), que são dados quantitativos que de alguma forma descrevem o exemplo, selecionados de acordo com os objetivos do trabalho (Souto *et al*, 2003). Existem diversos algoritmos de AM, como por exemplo redes neurais artificiais, árvores de decisão, aprendizado Bayesiano, algoritmos de agrupamento, máquinas de vetores suporte, entre outros, cada qual com vantagens e desvantagens de uso (Souto *et al*, 2003).

Dependendo da necessidade de treinamento, o paradigma do algoritmo de AM pode ser classificado em supervisionado e não-supervisionado. O aprendizado supervisionado caracteriza-se pela necessidade de ser fornecido ao algoritmo um conjunto de atributos de entrada e de saída rotulados já conhecidos. Esses dados são usados para que o algoritmo possa estabelecer regras, criando assim um modelo que o permita classificar posteriormente novos dados desconhecidos (Souto *et al*, 2003; Witten e Frank, 2005). Os algoritmos de AM não-supervisionados necessitam apenas dos dados de entrada, a partir do qual o algoritmo procura tendências e padrões. Nesse caso os dados fornecidos não são rotulados (Souto *et al*, 2003).

Em resumo, os algoritmos de AM necessitam ser treinados previamente por um conjunto de treinamento com atributos quantitativos que representam características de cada

dado, podendo ser rotulados (aprendizado supervisionado) ou não (aprendizado não-supervisionado). É a partir desses dados do conjunto de treinamento que o algoritmo irá derivar regras que lhe permitam identificar ou classificar o conjunto de teste que será apresentado posteriormente. Antes do algoritmo já treinado ser submetido a dados desconhecidos, é necessário realizar uma estimativa da taxa de erro (acurácia) com o ajuste simultâneo de parâmetros internos, que pode ser feito por otimização em um conjunto não utilizado durante o treinamento, ou então por técnicas como validação cruzada (*cross-validation*) ou *bootstrap* (Souto *et al*, 2003; Mitchell, 1997; Witten e Frank, 2005).

1.3.1 Naive Bayes

A teoria de Bayes é um tópico de discussão ainda atual entre estatísticos, originando opiniões divergentes quanto à sua utilização. Ela lida com os chamados problemas de estatística inversa, onde estão disponíveis apenas exemplos (realizações) de uma variável, mas sua distribuição (modelo) é desconhecida. Assim, diferentemente de problemas tradicionais de estatística onde se deseja calcular o valor de uma variável *d* dado seu modelo *m*: P (*d* | *m*), a estatística inversa procura resolver o problema: P (*m* | *d*). O teorema de Bayes é uma resolução matemática direta desse problema:

$$P(m \mid d) = \frac{P(d \mid m)\, P(m)}{P(d)}$$

O modelo estatístico gerado é baseado em probabilidades inferidas a partir dos dados disponíveis previamente. Percebe-se a partir da equação do teorema de Bayes que a probabilidade do modelo *m* descrever corretamente a distribuição de *d* é proporcional à probabilidade dos dados *d* terem sido originados pelo modelo *m* e à probabilidade anterior (antes de *d* ter sido apresentado) de *m*, mas é inversamente proporcional à probabilidade dos dados *d* serem gerados independentes do modelo *m.*

Em aprendizagem de máquina, há interesse que o algoritmo determine qual modelo *m* dentre os diversos candidatos é o mais provável (modelo máximo *a posteriori*), fornecendo-se apenas os dados *d*. Na fase de aprendizagem são coletadas as diversas P(*d*) e P(*d* | *m*), de acordo com suas freqüências no conjunto de treinamento. Na fase de teste, ao ser desafiado por novas instâncias, o classificador Bayesiano emite probabilidades (“confiança”) do dado pertencer a uma classe ou outra de acordo com o modelo gerado durante o treinamento. O algoritmo de naive Bayes (nB) assume uma premissa simplista (e muitas vezes irreal) de independência condicional entre os atributos, o que lhe conferiu esse nome (*naive* é “ingênuo” na língua inglesa). Essa premissa é simultaneamente o ponto mais forte e mais fraco do

algoritmo, pois o algoritmo possui um baixo desempenho em conjuntos com atributos redundantes (que quantificam características relacionadas).

Apesar de considerados extremamente simples, empiricamente os algoritmos de aprendizagem Bayesianos não só são comparáveis aos algoritmos mais recentes, como algumas vezes apresentam eficiência superior a abordagens muito mais complexas (Mitchell, 1997; Eddy, 2004a).

1.3.2 Máquinas de Vetores de Suporte (MVS)

Devido à flexibilidade para aplicação a diversos problemas e à robustez ao se trabalhar com dados com ruídos e de grande dimensão, o uso de algoritmos de AM, em especial os de máquinas de vetores de suporte (MVS), teve um aumento substancial em diversas áreas (Souto *et al*, 2003; Larrañaga *et al*, 2006; Noble, 2006), por exemplo na classificação de textos ou reconhecimento facial (Lorena, 2006), entre outros. O algoritmo MVS tem sido aplicado também com sucesso em diversos problemas da bioinformática, como predição de estrutura secundária de proteínas (Birzele e Kramer, 2006), análise de sinais na região codificadora como sítios de metilação, de início de transcrição, de *splicing* e de poliadenilação (Larrañaga *et al*, 2006), identificação de microRNAs (Nam *et al*, 2006), localização subcelular (Lorena, 2006) e identificação de peptídeos-sinal (Lorena, 2006), processamento de dados de microarranjos (Noble, 2006; Lorena, 2006), análise filogenética (Larrañaga *et al*, 2006), entre diversos outros. Uma de suas características mais interessantes é o uso preferencial dos dados mais relevantes do conjunto (dados redundantes ou "supérfluos" tendem a ser ignorados).

O algoritmo MVS é uma adaptação do algoritmo de redes neurais, porém com uma abordagem mais estatística, e devido à sua robustez e base teórica mais sólida, tende a obter melhores resultados (Schölkopf e Smola, 2002; Abate *et al*, 2007; Lorena, 2006) ou pelo menos resultados comparáveis (Romero e Toppo, 2007). MVS é um algoritmo de AM, normalmente com aprendizado do tipo supervisionado, sendo seu uso mais popular em resolução de problemas de reconhecimento binário de padrões - apenas duas classes possíveis: -1 e +1 (Schölkopf e Smola, 2002), apesar de ser possível seu uso com outras configurações (Lorena, 2006). O algoritmo de MVS é muito útil, por exemplo, quando se tem um conjunto de dados misto com relação a duas classes e não se conhece uma regra geral que separe com precisão os dados do conjunto em função dessas duas classes, mas ainda assim estão disponíveis conjuntos disjuntos, suficientemente grandes, de exemplos independentes e relativamente confiáveis de ambas as classes (Larrañaga *et al*, 2006). Talvez por isso o

algoritmo de MVS tenha sido cada vez mais usado na detecção de ncRNAs, já que a quantidade de exemplos e bancos de dados de ncRNAs disponíveis estão crescendo rapidamente, embora ainda não se conheça uma regra única e universal que os identifique.

O MVS é baseado na Teoria do Aprendizado Estatístico descrita por Vapnik (1998), e o problema tratado é essencialmente de classificação. A partir de exemplos rotulados já conhecidos, o algoritmo gera uma função que faz predições de atribuição de classes a dados não-rotulados desconhecidos. Para um conjunto de classes de intervalo discreto 1…*k*, tem-se um problema de classificação; se o intervalo assume valores contínuos, tem-se um problema de regressão. Para $k > 2$ configura-se um problema multiclasses, e no caso especial $k = 2$ tem-se um problema binário (duas classes) de classificação (Lorena, 2006), que é a configuração usada nesse trabalho.

O funcionamento do MVS se dá da seguinte maneira: inicialmente, separam-se os exemplos em duas classes, a positiva e a negativa. Em seguida definem-se quais características (ou atributos, ou qualidades) serão extraídas dos exemplos: dessa forma é montado o vetor de características. Os dados quantitativos são então normalizados, organizados, rotulados e formatados para servirem de entrada ao MVS. Os dados não necessitam obedecer a nenhuma distribuição particular (modelo estatístico não-paramétrico): por exemplo, não é necessário que os valores assumam uma distribuição normal (Noble, 2006). O MVS então inicia o processamento do conjunto de atributos dos dados de treinamento, gerando o espaço de entradas composto de pontos, que pode ser entendido como um gráfico cartesiano de duas dimensões (Figura 1a). A tarefa do algoritmo é encontrar uma função que separe os pontos negativos e positivos de forma que essa função tenha o mínimo de sobreposição a pontos, e máxima distância a pontos fronteiriços, os chamados vetores de suporte. Em casos reais, no entanto, nem sempre é possível separar linearmente os pontos nesse espaço bidimensional; com isso incorpora-se uma função especial ao MVS, permitindo-o lidar com casos não-lineares. Essas funções fazem o mapeamento dos dados de entrada para espaços de dimensão mais elevada, chamados espaços de características (Figura 1b). A representação explícita dos pontos no espaço de características após o mapeamento é uma tarefa muito custosa computacionalmente. Porém, graças a funções especiais denominadas *Kernels*, o problema pode ser tratado por uma abordagem implícita, realizando apenas os cálculos necessários sem representar explicitamente os pontos do mapeamento de dimensão mais alta (Figura 1b) (Lorena, 2006). A função Kernel recebe os vetores x_i e x_j do espaço de entradas e calcula o produto escalar desses dois pontos no vetor de características, ou seja:

$$K(x_i, x_j) = \Phi(x_i)\ \Phi(x_j)$$

Dentre as funções Kernel mais utilizadas e conhecidas, pode-se citar:

1. Função polinomial:

   $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d$, $\gamma > 0$ (parâmetros para otimização: γ, r e ^{d}).

2. Função Radial Basis Function (RBF) (também chamada gaussiana):

   $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$, $\gamma > 0$ (parâmetro para otimização: γ).

3. Função sigmoidal:

   $K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$ (parâmetros para otimização: γ e r).

O processamento dos dados em dimensões mais altas tem o intuito de encontrar uma função que seja simples (Figura 1c) no espaço de características, mas que, quando analisada na dimensão original, seja eficiente em separar pontos negativos e positivos (Figura 2).

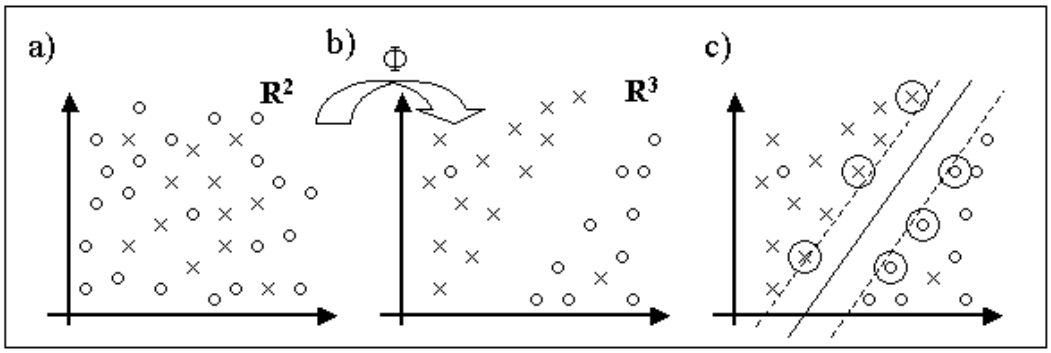


Figura 1. Esquema representativo do funcionamento do algoritmo MVS para duas classes. Os vetores de cada classe estão representados por círculos ou cruzes. **a)** Espaço de entrada bidimensional contendo os vetores do conjunto de treinamento. A separação das classes por uma função é impossível nessa condição. **b)** Uma função *kernel* (representada por Φ) mapeia os dados do espaço de entradas para o espaço de características, onde as classes já podem ser separadas. **c)** Um hiperplano (representado por uma linha contínua) é criado no espaço de características, permitindo a separação das classes. Os vetores de suporte estão circulados. O hiperplano ótimo determina a fronteira de decisão do SVM, que é acompanhada também por margens separadoras (representadas por linhas pontilhadas). Adaptado de (Lorena, 2006) e (Schölkopf e Smola, 2002).

A solução que se busca no espaço de características é encontrar um hiperplano ótimo que se sobrepõe à menor quantidade de vetores de suporte possível, ao mesmo tempo em que maximiza a fronteira de decisão do classificador (Figura 1c). Isso resulta em uma boa generalização, que é a base de uma predição acurada para dados desconhecidos (Chang e Lin, 2006). Nesse ponto, uma vantagem adicional do SVM é a existência de apenas um mínimo global (graças à convexidade do problema de otimização), em contraste aos múltiplos mínimos locais presente nas Redes Neurais Perceptron Multicamadas, por exemplo. Essa

característica é interessante porque garante uma melhor otimização do problema, maior confiabilidade dos resultados e menor tempo de execução computacional (Lorena, 2006).

Idealmente, os dados do conjunto de treinamento devem possuir baixa redundância e atributos que avaliem características diversas, não-redundantes, sob o risco de ser obtido um classificador "especialista" nos dados de treinamento que é incapaz de prever dados desconhecidos – o chamado *overfitting*. Deve-se também ter um conjunto de treinamento suficientemente grande e usar um *Kernel* compatível com o trabalho para evitar a geração de um modelo demasiadamente simples – o chamado *underfitting* (Lorena, 2006).

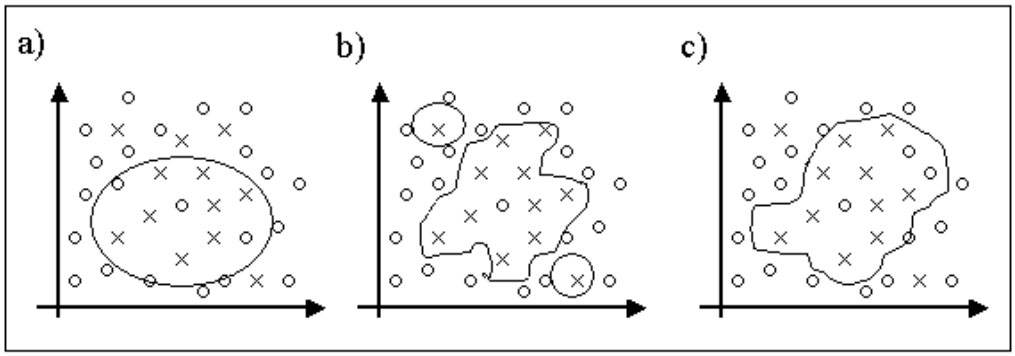


Figura 2. Possíveis configurações da fronteira de decisão. a) O modelo induzido é demasiado simples, caracterizando um sub-ajustamento (*underfitting*) do modelo ao conjunto de treinamento. **b)** O modelo induzido classifica corretamente todos os dados do conjunto de treinamento, inclusive *outliers* de ambas as classes, caracterizando o *overfitting*, ou super-ajustamento do modelo ao conjunto de treinamento. **c)** Esse caso ilustra um compromisso entre os casos a) e b), assumindo complexidade e ajustamento intermediários. Adaptado de (Lorena, 2006).

1.4 Aplicação do algoritmo MVS à identificação de ncRNAs

Uma abordagem possível para identificar ncRNAs em um transcriptoma é o uso de MVS treinado com dados de literatura, usando atributos que caracterizam os produtos protéicos putativos desses conjuntos de dados (Liu *et al*, 2006). O método, chamado de CONC (**C**oding **O**r **N**on-**C**oding), introduzido com o trabalho de Liu *et al* (2006), foi pioneiro ao aplicar MVS ao problema da identificação de ncRNAs médios e longos em transcriptomas e inspirou o método usado nesse projeto. A estratégia consiste em preparar dois conjuntos, um contendo exemplos de mRNAs conhecidos e outro possuindo exclusivamente ncRNAs. A redundância de cada conjunto é eliminada, os atributos são escolhidos e os programas que os extraem são implementados, e então, os dados são formatados para entrada no programa de MVS. Os autores trabalharam com a hipótese de que proteínas "reais" diferem de produtos protéicos putativos de ncRNAs, portanto as traduções de transcritos também devem ser analisadas. Após a proposição do CONC, outro grupo criou paralelamente o programa **C**oding

Potential **C**alculator - CPC (Kong *et al*, 2007), que usa o mesmo banco de dados de treinamento do CONC, porém com atributos diferentes, para indução de um classificador MVS. A proposta do CPC é otimizar a acurácia do CONC focando na análise de programas que estimam similaridade de proteínas.

Tanto o CPC como o CONC mostram-se excelentes para análise de transcriptomas de organismos bem caracterizados experimentalmente, do qual tem-se riqueza de informação de genoma, transcriptoma e proteínas, tanto para o organismo sendo analisado quanto para organismos filogeneticamente próximos a ele, já que ambos programas dependem muito da análise comparativa. Além disso, tanto o CPC como CONC presumem que a seqüência seja de boa qualidade e completa, não-truncada, já que a tradução é feita de forma simples, analisando apenas os sinais canônicos de um mRNA, motivos esses que nem sempre estão presentes em ESTs provenientes de projetos de seqüenciamento com *reads* de baixa qualidade (Nadershahi *et al*, 2004).

Nas seções 1.4.1 e 1.4.2, algumas etapas da operação com MVS são discutidas com mais detalhes, ilustradas pelo método de Liu *et al* (2006), que é também a estratégia usada no presente trabalho.

1.4.1 Construção do conjunto de treinamento

O algoritmo MVS faz uso do aprendizado supervisionado. Então, seu treinamento exige, para a formulação de seu modelo preditivo, que sejam fornecidos dois conjuntos de dados rotulados, um positivo e um negativo. Especificamente para a aplicação do MVS ao problema dos ncRNAs, o positivo pode ser construído por exemplo a partir do UniProt Knowledge base (Wu *et al*, 2006a), que consiste em um repositório de seqüências de proteínas com alto nível de anotação, baixa redundância e alta integração a outros bancos de dados. No entanto, mesmo com um baixo nível de redundância, qualquer banco de dados de seqüências de proteínas possui uma redundância intrínseca, que ocorre devido à diferença na representatividade que certos domínios (ou famílias) possuem (Baldi e Brunak, 2001). Por exemplo, a família PS01033 (perfil da família de globinas) aparece em 1.934 proteínas no banco de dados PROSITE (Hulo *et al*, 2006); já a família de proteínas PS00295 (assinatura de arrestinas) possui somente 87 representantes. Isso pode ser reflexo de ancestralidade comum na herança desses domínios (Pandit *et al*, 2004; Lesk, 2002) ou de convergência evolutiva (Lesk, 2002); qualquer que seja o caso, essa redundância em domínios funcionais é um potencial gerador de *overfitting* em algoritmos de aprendizagem de máquina, já que a

saturação de exemplos repetidos induz um padrão indesejável no conjunto de treinamento que o algoritmo pode interpretar como um sinal relevante (Schölkopf e Smola, 2002).

Existem programas que lidam com a eliminação dessa redundância em bancos de dados, gerando conjuntos de dados contendo representantes únicos de cada família de proteínas. Exemplos desses programas são o UniqueProt (Mika e Rost, 2003), TribeMCL (Enright *et al*, 2002), SYSTERS (Krause *et al*, 2000), e o CD-HIT (Li e Godzik, 2006). O CD-HIT é um programa bastante utilizado em trabalhos com bancos de dados de seqüências biológicas, sendo adotado, por exemplo, para gerar conjuntos não-redundantes de seqüências do banco de dados UniProt (Wu *et al*, 2006a) e de diversos outros (Li e Godzik, 2006).

A redundância do banco de dados de nucleotídeos também deve ser eliminada. Programas disponíveis para esse fim são o CD-HIT-EST, componente do pacote CD-HIT (Li e Godzik, 2006), CleanUP (Grillo *et al*, 1996), e BLASTCLUST (McGinnis e Madden, 2004). Esses programas normalmente são usados para agrupar fragmentos de DNA (cDNAs) similares em *contigs* durante projetos de seqüenciamento, mas também podem ser usados como eliminadores de redundância de um conjunto de dados.

Os conjuntos de dados de seqüências nucleotídicas não redundantes gerados são a seguir submetidos à predição de fases abertas de leitura (ORFs) por programas especializados. A seleção do algoritmo é uma etapa crucial já que diversas variáveis do vetor de características são obtidas a partir da seqüência protéica predita. A detecção de ORFs em ESTs é um problema antigo da bioinformática que, apesar de parecer simples, empiricamente mostra-se bastante complexo, apresentando diversas "armadilhas" para a correta predição (Nadershahi *et al*, 2004); por isso, nem todos os algoritmos disponíveis são apropriados para essa tarefa. Por exemplo, os programas ORFfinder (Wheeler *et al*, 2004), Diogenes (Crow e Retzel, 2005) e *getorf*, do pacote EMBOSS (Rice *et al*, 2000) utilizam uma abordagem simples de leitura nas 6 fases, sendo normalmente escolhido como produto protéico a maior cadeia polipeptídica predita. Essa abordagem é ineficaz para uso em ESTs, pois não considera a baixa qualidade das seqüências e o contexto como um todo, preocupando-se apenas com os sítios de início e término da ORF, podendo por exemplo confundir uma seqüência incompleta com um transcrito inteiro.

Os algoritmos ESTScan (Lottaz *et al*, 2003) e Diana-EST (Hatzigeorgiou, 2001) são eficientes para a tarefa a que se destinam: encontrar ORFs em um organismo específico, do qual se têm abundância de exemplos e informações *a priori* de seqüências que possuem e que não possuem ORFs do próprio organismo e de organismos aparentados filogeneticamente. Os exemplos são necessários porque esses programas fazem uso de aprendizagem de máquina,

como por exemplo redes neurais. Por isso o uso desses programas é inviável tanto para ESTs de organismos que não são modelos e com poucos dados de parentes próximos, quanto em predições em larga escala de diversos organismos diferentes, onde o modelo gerado para uma espécie é inválido para predição de ORFs de uma outra espécie. Já o programa OrfPredictor (Min *et al*, 2005) possui uma abordagem voltada especificamente para a complexidade da predição de ORFs em ESTs com *reads* de baixa qualidade. O programa explora todas as possibilidades que um EST pode representar: idealmente ele deve ser o transcrito completo que irá codificar a proteína; mas também, ele pode ser apenas a parte central de um transcrito, estando ausentes seu códon de início e sinal de poliadenilação; pode ocorrer o fato desse EST nem sequer apresentar uma ORF; e pode ser também que o EST seja uma combinação complexa dessas e de outras situações. Em teoria, para a identificação de ncRNAs em um transcriptoma, bastaria submeter todos os transcritos à predição de ORFs por esse programa, sendo que aqueles que não apresentaram ORF predita seriam rotulados não-codificadores. No entanto, durante a submissão de 265.691 exemplos de ncRNAs conhecidos, obtidos dos bancos de dados, cerca de 46% das seqüências tiveram uma ORF predita (dados não mostrados), e na submissão de 3.000 seqüências geradas aleatoriamente, mais de 97% tiveram uma ORF identificada (dados não mostrados). Isso leva a crer que o algoritmo é eficiente em identificar ORFs mas apresenta muitos falsos-positivos, podendo isso ser um indicativo de viés do programa em encontrar ORFs e uma deficiência em identificar transcritos não-codificadores. O programa ANGLE (Shimizu *et al*, 2006) foi construído considerando dois problemas em seqüências de ESTs com *reads* de baixa qualidade: os erros no seqüenciamento (inserções e deleções - *indels*) e o truncamento de seqüências. Para isso o programa usa uma modelagem que tenta prever e corrigir os *indels*, além de ser otimizado para lidar com seqüências curtas, valorizando ao máximo a informação de input. A abordagem é híbrida e composta por três etapas: na primeira, um classificador "fraco" (AdaBoost) de região codificadora avalia segmentos da seqüência por meio de uma janela deslizante; então, um modelo de Markov determina a estrutura secundária protéica ótima, por meio de um pontuador baseado em programação dinâmica; e finalmente, mudanças da fase de leitura são detectadas e consideradas. As seis fases de leitura são analisadas e o produto protéico mais provável codificado pelo transcrito é fornecido. ANGLE foi treinado em um conjunto de mRNAs humanos, no entanto ele mostra um desempenho excelente mesmo em organismos filogeneticamente distantes de humano, como por exemplo fungos (Dr. Kana Shimizu, comunicação pessoal). Além disso, ANGLE não apresenta um viés em identificar ORFs em

conjuntos de dados onde a maioria é não-codificador, contrariamente ao programa OrfPredictor, conforme discutido acima (dados não mostrados).

O conjunto negativo pode ser construído de forma similar, a partir de exemplos de ncRNAs depositados em bancos de dados específicos, como aqueles citados na seção 1.2.2. Vale ressaltar que esse conjunto também deve passar por processos de eliminação de redundância em nível de nucleotídeos, e as ORFs putativas de seus transcritos também podem ser obtidas pelos programas descritos acima.

1.4.2 Vetor de características

A montagem do vetor de características é uma etapa essencial da utilização do algoritmo de MVS. Denomina-se vetor de características o conjunto de atributos quantitativos que descrevem de alguma forma um objeto ou fenômeno (Souto *et al*, 2003). Os atributos que serão adotados - e que integram o vetor de características - são escolhidos pelo pesquisador de acordo com uma tentativa de erro-e-acerto (Noble, 2006), ou a partir da "intuição", embasado em seu conhecimento prévio sobre o tópico (Liu *et al*, 2006).

Intuitivamente pode-se dizer que um gene, por exemplo, passa a ser representado de forma que possa ser processado, ao ser codificado na forma de valores referentes aos diversos atributos, como: tamanho, porcentagem de nucleotídeos, presença de motivos etc., todos contidos no vetor de características (Wang *et al*, 2006).

Ao escolher as características, é importante ter em mente que o MVS é um algoritmo que realiza cálculos complexos, envolvendo multiplicações em campos de dimensões elevadas, e dependendo da quantidade de produtos, pode-se ter um problema que exija muito processamento computacional, ou que demande muito tempo, possibilitando até chegar-se a um problema intratável computacionalmente (Noble, 2006). Outra consideração importante é não utilizar atributos com informações redundantes, o que poderia causar o *overfitting* na geração do modelo do MVS, fazendo com que o algoritmo se "especialize" nos dados de treinamento e possua um desempenho insatisfatório para dados desconhecidos (Wang *et al*, 2006; Lorena, 2006). É imprescindível que o atributo a extrair possa ser codificado em forma quantitativa, isto é, em forma de números reais ou inteiros. No caso de atributos qualitativos (não-quantitativos), deve-se procurar codificá-lo em forma de categorias, já que o algoritmo MVS aceita como entrada apenas valores numéricos.

Composição de nucleotídeos: O conteúdo nucleotídico é uma característica muito usada em análise de seqüências por algoritmos de aprendizagem de máquina. Além disso, também já é reconhecida há muito tempo como uma boa característica de identificação de

potencial codificador de um transcrito (Fickett e Tung, 1992). Em teoria, quanto maior a quantidade de palavras usadas, e quanto maior a quantidade de letras em cada palavra, mais preciso é o modelo gerado pelo MVS. No entanto, a codificação para o formato MVS dessas palavras demanda um grande número de variáveis: a composição de um nucleotídeo exige 4 variáveis (A, C, G e T), a de dinucleotídeos exige 16 variáveis (AA, AC, AG, AT, TA, etc) , a de trinucleotídeos exige 64 variáveis (AAA, AAC, AAG, AAT, ATT, etc), e assim por diante. A análise de nucleotídeos pode também ser feita em diversas fases de leitura, ou apenas uma. Ao selecionar quais fases e que tamanhos de palavras usar como atributos, o pesquisador deve considerar o compromisso entre custo computacional e a geração de características representativas. Um exemplo de programa que extrai a composição de nucleotídeos, di- e trinucleotídeos é o *compseq*, do pacote EMBOSS (Rice *et al*, 2000).

Comprimento da fase aberta de leitura: A presença de ORFs e seu comprimento é uma característica muito usada na análise computacional para classificar um transcrito como codificador ou não-codificador. Como consenso de literatura, considera-se que uma ORF que codifique para um produto protéico menor do que 100 aminoácidos pode ser considerado um transcrito não-codificador, um limite arbritário muito usado empiricamente e com evidências de ser efetivamente um valor adequado (Frith *et al*, 2006). Um programa que pode ser usado para calcular a extensão das ORFs é o *checktrans*, componente do pacote EMBOSS (Rice *et al*, 2000).

Composição de aminoácidos: A estrutura primária de uma proteína é o que irá determinar suas estruturas secundária e terciária, que por sua vez são essenciais na determinação da atividade biológica da proteína. Os aminoácidos presentes em uma proteína podem ser analisados para inferir diversas informações: paralogia e ortologia a outras proteínas, viés no uso de códons, segmentos repetitivos, presença de aminoácidos raros, assinaturas específicas, entre outros. Muitos dados estão codificados na estrutura primária de uma proteína, e à medida que as seqüências são tornadas disponíveis, o que limita o conhecimento sobre essas moléculas são novas formas de analisá-las (Nelson e Cox, 2004).

A composição de aminoácidos é uma análise derivada da estrutura primária: informa o conteúdo de resíduos de uma proteína, mas sem levar em conta posições ou seqüências desses aminoácidos. Embora um dado mais pobre que a estrutura primária, as preferências de resíduos e freqüência de aminoácidos mais raros permitem inferir a história evolutiva e até características ambientais sobre um organismo. Um programa que permite fazer a análise dessa característica é o *pepinfo*, componente do pacote EMBOSS (Rice *et al*, 2000).

Predição de ponto isoelétrico da proteína: Os resíduos de aminoácidos de uma proteína estão sujeitos à ionização de acordo com suas características intrínsecas de eletronegatividade, de interação com meio e de interação entre os átomos da própria molécula. Em decorrência disso, as moléculas apresentam diferentes perfis de carga em resposta a diferentes pHs do meio. Determina-se ponto isoelétrico o valor pontual de pH onde as cargas da proteína se anulam e assim a proteína apresenta carga global neutra. Esse atributo fornece informações quanto ao ambiente de atuação da proteína, além de resumir a composição de aminoácidos a apenas um valor (Nelson e Cox, 2004). Um programa que determina a carga global da proteína em cada pH, além de seu ponto isoelétrico, é o *iep*, componente do pacote EMBOSS (Rice *et al*, 2000).

Complexidade da proteína (entropia composicional): A seqüência de algumas proteínas às vezes apresenta viés de composição de alguns resíduos, caracterizada por uma repetição extensa e periódica desses resíduos. Segmentos que apresentam essa propriedade são classificados como de baixa complexidade (Promponas *et al*, 2000). Segmentos de baixa complexidade foram por muito tempo considerados lixos resultantes de tradução de RNA repetitivo, que por sua vez teve origem de elementos repetitivos de DNA, que também seriam lixo. No entanto, já existem evidências que indicam que esses segmentos são importantes para diversas atividades das proteínas, e parecem estar associados a condições patológicas neurais, como príons (Kuznetsov e Hwang, 2006; Kreil e Ouzounis, 2003). Programas que estimam a entropia composicional de uma seqüência protéica são: SEG (Wootton e Federhen, 1996), CAST (Promponas *et al*, 2000), BIAS (Kuznetsov e Hwang, 2006) e CARD (Shin e Kim, 2005).

Hidrofobicidade média da proteína: As cadeias laterais dos aminoácidos variam de não-polar e hidrofóbica (não interagem com a água) a polar ou hidrofílica (interagem com a água). O perfil de hidrofobicidade é importante ao determinar com quais átomos, dentro da própria proteína ou no ambiente, cada resíduo poderá interagir. Resíduos hidrofóbicos tendem a se aglomerar e permanecerem no núcleo de proteínas globulares, sem contato com o solvente; os hidrofílicos, por sua vez, aglomeram-se e ficam expostos ao meio. Esse esquema de disposição, no entanto, pode ser revertido para proteínas localizadas em membranas lipídicas, onde o solvente é apolar (Nelson e Cox, 2004).

O estudo da polaridade ou hidropatia, portanto, fornece inferências quanto ao ambiente onde a proteína normalmente atua, da composição de aminoácidos e também uma idéia geral do perfil de interação da proteína com o meio (Nelson e Cox, 2004).

Diversas tabelas de atribuição de valores de polaridade de cada resíduo foram confeccionadas. A tabela é escolhida de acordo com o método que irá avaliar a polaridade da proteína. Uma das tabelas mais utilizadas, principalmente em experimentos de bioinformática, é a constante do programa SOAP (Kyte e Doolittle, 1982). Outros programas que calculam a polaridade, mas também se baseiam no método do SOAP, são o *pepinfo* e *octanol*, componentes do pacote EMBOSS (Rice *et al*, 2000).

1.5 Medidas de eficiência

Avaliar a eficiência do classificador induzido é uma etapa essencial em experimentos de aprendizado de máquina, sem a qual é impossível determinar a validade do classificador. Em procedimentos de mineração de dados é comum que sejam encontrados exemplos classificados erroneamente, atributos com valores imprecisos e às vezes ausentes, *outliers*, e até mesmo um desbalanço nos conjuntos de treinamento, como nos casos em que há um excesso de exemplos de uma classe e uma grande escassez de exemplos de outra classe. No estudo dos ncRNAs, os principais empecilhos são a escassez de exemplos de ncRNAs (comparativamente à quantidade de exemplos de mRNAs), a qualidade duvidosa desses exemplos (principalmente porque os dados sobre esses transcritos são recentes e a grande maioria foi identificada apenas *in silico*, sem confirmação experimental), alguns transcritos terem apenas sua seqüência parcial representada, e também a imprecisão dos algoritmos que extraem atributos das seqüências.

Todas essas inconsistências do conjunto de treinamento constituem ruído, e a robustez do algoritmo de aprendizado de máquina é que definirá se esses exemplos errôneos serão usados na construção do classificador. Uma medida de eficiência é afetada se o classificador tiver sido construído com base em exemplos errôneos, e também se os atributos e instâncias do conjunto de treinamento são representativos o suficiente para que uma teoria generalizada possa ser gerada a partir deles. Empiricamente, é uma forma de julgar se o classificador é útil ou não para a tarefa a que se destina: rotular dados novos, desconhecidos, não apresentados durante a etapa de treinamento (Witten e Frank, 2005).

1.5.1 Matriz de confusão (tabela de contingência) e medidas derivadas

A medição da eficiência depende da disponibilidade de um conjunto previamente rotulado, no qual sejam conhecidas as classes de cada exemplo. Os exemplos que constituem esse conjunto, chamado conjunto de teste, não podem ter sido usados durante o treinamento do classificador. Após o treinamento, o classificador já pronto é então usado para fazer

predições individuais dos exemplos do conjunto de teste; se a classe predita é igual à previamente conhecida, diz-se que o classificador teve um acerto; se discordam, tem-se um erro.

Para um problema binário (apenas duas classes), só há quatro possibilidades de resultado durante uma análise de eficiência, que são representados por uma matriz de confusão (tabela 3).

Tabela 3. Possíveis resultados de um problema de classificação envolvendo duas classes (adaptado de Witten e Frank, 2005).

		Classe predita	
		Positiva	**Negativa**
Classe real	**Positiva**	Verdadeiro positivo (VP)	Falso negativo (FN)
	Negativa	Falso positivo (FP)	Verdadeiro negativo (VN)

A matriz de confusão é uma forma de representação dos resultados que permite a interpretação e derivação de diversas medidas de eficiência (Witten e Frank, 2005; Fawcett, 2004):

(1) Acurácia (ou Taxa geral de sucesso):

$$\frac{VP+VN}{VP+VN+FP+FN}$$

(2) Sensibilidade (ou Taxa de Verdadeiros Positivos, ou *Recall*):

$$\frac{VP}{VP+FN}$$

(3) Medida-F:

$$\frac{2 \times VP}{2 \times VP + FP + FN}$$

(4) Especificidade:

$$\frac{VN}{VN+FP}$$

Os valores de todas as medidas são restritos ao intervalo real [0,1], sendo comumente expressos também em forma porcentual. No contexto desse trabalho, a sensibilidade pode ser entendida como a probabilidade de que se um transcrito for predito como codificador, ele seja

realmente codificador. A especificidade é a probabilidade de que um transcrito predito como não codificador realmente seja um ncRNA. A acurácia indica simultaneamente a proporção de transcritos preditos como codificadores que realmente são codificadores, e a quantidade de preditos como não-codificadores que realmente são ncRNA. Logo, é uma medida geral do sucesso de um classificador. A medida-F é uma métrica harmônica ponderada entre sensibilidade e especificidade, permitindo uma análise simultânea das duas medidas, sendo um método satisfatório para avaliação do classificador (Witten e Frank, 2005).

1.5.2 Validação cruzada

Esse método é um dos mais usados em avaliação de eficiência em experimentos de mineração de dados. Seu funcionamento se dá da seguinte forma: inicialmente, o conjunto original é fracionado em *k* vezes (ou sub-partes) com quantidades de exemplos os mais similares possíveis e com uma distribuição homogênea de classes. O parâmetro *k* pode ser escolhido pelo usuário, no entanto k=10 é um valor tido como ideal tanto por aproximações teóricas como empíricas (Witten e Frank, 2005). O algoritmo consiste em uma quantidade de repetições (iterações) igual ao número *k* selecionado pelo usuário. A cada iteração, um subconjunto diferente é tomado como único conjunto de treinamento, o qual gera um modelo que é usado para realizar predições sobre o conjunto de teste (que é composto pelos k-1 subconjuntos restantes). Ao final do processo, cada subconjunto terá sido usado como conjunto de treinamento uma vez, e terá sido testado k-1 vezes pelos demais subconjuntos (Figura 3). Como a cada iteração o modelo criado fornece uma predição para instâncias cujos rótulos (classes) já são previamente conhecidos, pode-se fazer ao final do processo uma avaliação da acurácia desse modelo.

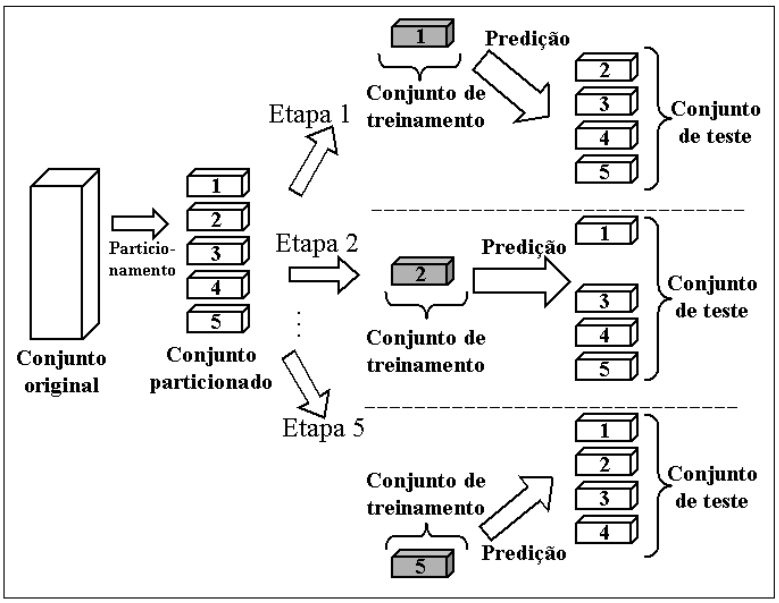


Figura 3. Esquema ilustrativo do processo de validação cruzada para k=5 (5 vezes). Após o particionamento do conjunto original, são feitas cinco iterações, e a cada iteração um subconjunto diferente é tomado como conjunto de treinamento (em cinza) e os demais *k*-1 subconjuntos são tomados como teste (em branco). A cada iteração, a acurácia do modelo é estimada de acordo com a contabilização dos acertos e erros da predição. Ao final de todas iterações, a acurácia global do modelo pode ser estimada fazendo a média aritmética das diversas acurácias obtidas.

Existe um esquema especial de validação cruzada, onde apesar das instâncias de cada subconjunto serem escolhidas aleatoriamente, as classes são distribuídas uniformemente, de forma que todos os conjuntos tenham uma quantidade aproximadamente igual de exemplos positivos e negativos. Esse tipo de validação cruzada é denominado estratificado, e é o mais recomendado atualmente por não causar desbalanços de classe e de viés nos conjuntos de treinamento e teste (Witten e Frank, 2005).

1.5.2 Curvas ROC

A avaliação do desempenho de um classificador pode ser feita por curvas ROC (do inglês *receiver operating characteristic*). Esse tipo de gráfico é originado de estudo de detecção de sinais, quando se deseja determinar a proporção entre sinais verdadeiros e alarmes falsos através de um canal com ruídos (por exemplo, na detecção de sinais em um radar), descrevendo o desempenho de um classificador independentemente de balanço na distribuição de classes ou custos de erros (que normalmente não são usados em algoritmos de MVS). No eixo das ordenadas tem-se a quantidade de exemplos positivos, expressos como

porcentagem do total de positivos, e nas abscissas, o número de exemplos negativos, expressos como o percentual do total de negativos.

Os dados usados no gráfico são obtidos por comparação das predições do classificador contra exemplos já previamente rotulados. Usando os dados da predição e os rótulos conhecidos *a priori*, uma nova tabela é feita em ordem decrescente de probabilidade de predição (*ranking* de "confiança" de predição), e para cada exemplo (instância), compara-se a classe que o classificador atribuiu com a classe "real" da instância, já previamente conhecida. Para cada instância, se a predição do classificador coincide com a classe atribuída *a priori*, um "sim" é contabilizado; caso contrário, contabiliza-se um "não". A partir dessa lista, começando do extremo inferior esquerdo do gráfico - ponto (0,0), para cada "sim" é traçada uma linha vertical e para cada "não", é traçada uma linha horizontal. Se estiverem disponíveis poucos pontos, tem-se um gráfico em forma de "escada"; conforme o número de pontos aumenta e tende a infinito, tem-se um gráfico que se aproxima a uma linha curvilínea (Figura 4) (Witten e Frank, 2005; Fawcett, 2004).

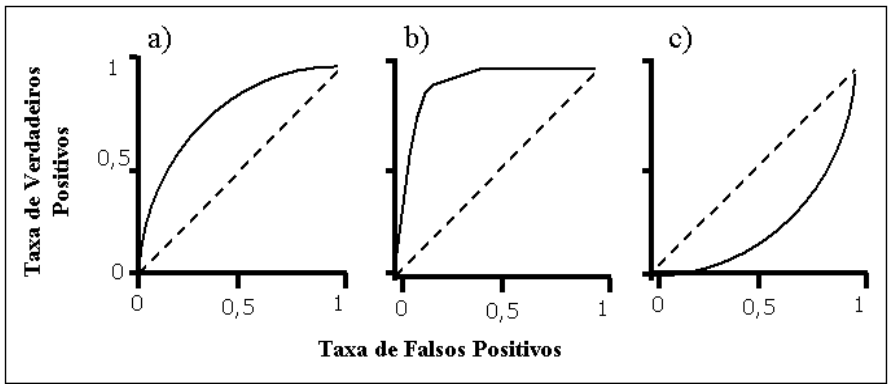


Figura 4. Possíveis disposições de uma curva ROC. Em todas as configurações, a linha pontilhada representa um classificador aleatório. a) Um classificador "liberal". b) Um classificador "conservador". c) Um classificador com desempenho inferior a uma classificação aleatória. Adaptado de (Fawcett, 2004).

Visto de outra forma, no eixo "y" tem-se a sensibilidade (taxa de verdadeiros positivos) do classificador, e no "x", os valores [1-especificidade] (taxa de falsos positivos), evidenciando que o ganho em um desses índices de eficiência implica necessariamente em decréscimo no outro, no sentido de que quanto maior a quantidade de positivos que se deseja, maior a quantidade de falsos negativos que se obtém como efeito colateral. Quanto maior a sensibilidade exigida de um classificador imperfeito, maior é o risco de serem obtidas predições positivas erradas.

Assim, um classificador randômico gera uma linha diagonal que intercepta o gráfico do canto esquerdo inferior ao direito superior. Quanto mais a curva se localize à esquerda e

acima, tanto melhor será o desempenho do classificador, sendo que um classificador que passe pelo ponto (0,1) e forme uma reta até o ponto (1,1) é dito perfeito (Witten e Frank, 2005). Uma curva próxima ao eixo "x" e mais acentuadamente à esquerda indica um classificador "conservador" (figura 4b), enquanto uma curva mais inclinada superior à direita é produzida por um classificador dito "liberal" (figura 4a). Um classificador conservador faz uma predição positiva apenas com uma forte evidência, por isso há pouca ocorrência de falsos positivos, mas a taxa de positivos verdadeiros tende a ser menor. Já o classificador liberal faz classificações positivas com evidência fraca, classificando quase todos os positivos corretamente mas com taxa elevada de falsos positivos (Fawcett, 2004).

A área abaixo de uma curva ROC (ou pontuação ROC, ou AAC) é a sensibilidade média dentre todos os valores de especificidade possíveis, a qual pode ser usada como medida para estimar o desempenho da predição para diferentes limites (Liu *et al*, 2006). Sendo a AAC uma porção de um gráfico quadrado, seu valor fica entre 0,0 e 1,0, sendo que se um classificador possui algum poder discriminativo deve ter sua AAC maior do que 0,5, que é o valor encontrado para um classificador que realiza predições aleatórias (Fawcett, 2004). Na figura 4, o classificador (b) tem uma AAC maior que os classificadores (a) e (c), evidenciando seu melhor desempenho para essa tarefa, sendo que o classificador (c) possui uma AAC < 0,5, indicando seu baixo desempenho, onde até mesmo predições geradas aleatoriamente são mais precisas que suas predições. A medida AAC já foi demonstrada como estatisticamente consistente, inclusive tendo um poder discriminatório maior do que a medida de acurácia (Ling *et al*, 2003), e é equivalente a outros testes estatísticos, como Wilcoxon e Gini (Fawcett, 2004).

1.6 Análise comparativa dos ncRNAs

Similarmente às proteínas, os RNAs também têm codificada em sua estrutura primária a informação necessária para seu dobramento para que atinja a estrutura terciária final (Zuker e Stiegler, 1981; Mount, 2004). Diferentemente do DNA, o RNA é uma molécula que não ocorre na forma de fitas duplas; logo, a tendência é que suas bases nitrogenadas façam ligações Watson-Crick entre si, o que dá origem a conformações secundárias ao longo da molécula (Mount, 2004). Considera-se também para os RNAs que sua estrutura terciária está em íntima associação com sua função (Hofacker, 2003), e que um RNA conservado pode ter alterações em sua estrutura primária, mas ainda assim ter uma estrutura secundária conservada (Zuker e Stiegler, 1981; Mount, 2004). Por isso, os programas que buscam similaridade por estrutura primária (como BLAST ou FASTA) são inadequados em estudos comparativos de

ncRNAs (Freyhult *et al*, 2007). Uma alternativa é usar programas que comparam estrutura secundária de RNA, atribuindo matrizes e esquemas de pontuação otimizados para identificar conservação de padrões de dobramento. Atualmente existem diversos programas que usam informações sobre estrutura secundária para inferir similaridade entre seqüências de RNA (revisto em Machado-Lima *et al*, 2007).

A maioria dos programas de alinhamento e comparação de seqüências de RNA utiliza-se da estrutura secundária predita por outros programas. O método atual mais usado para cálculo da estrutura secundária de um RNA é simular possíveis ligações entre diversos resíduos, atribuindo-lhes valores termodinâmicos calculados empiricamente; dentre as diversas estruturas geradas, a escolhida será a que tiver a menor energia de dobramento, obtida pela soma dos valores determinados individualmente para cada ligação (Machado-Lima *et al*, 2007). O programa RSmatch (Khaladkar *et al*, 2007), por exemplo, utiliza-se das estruturas preditas pelo programa RNAfold (Hofacker, 2003), considerando matrizes de pontuação distintas para porções pareadas e não-pareadas da estrutura secundária predita do RNA, dispensando o conhecimento ou fornecimento de um motivo a ser encontrado em um alinhamento múltiplo de RNAs, o que é uma exigência comum de programas similares. Suas configurações padrão são adaptadas à detecção de motivos curtos de RNAs, como aqueles compartilhados em miRNAs e snoRNAs. O principal atrativo do RSmatch, no entanto, é seu relativamente baixo custo computacional, o que permite que ele seja usado até na análise de grandes seqüências de RNAs (acima de 2.000 letras), sendo que programas similares são tão computacionalmente proibitivos que seu uso restringe-se a pequenas seqüências (em alguns casos, até 400 letras).

O consenso atual da literatura é que ncRNAs com padrões de dobramentos similares possuem ancestralidade comum (Machado-Lima *et al*, 2007; Freyhult *et al*, 2007), e corroborando essa premissa, já foram catalogadas famílias de dobramentos similares entre organismos filogeneticamente relacionados (Griffiths-Jones *et al*, 2005).

2. OBJETIVOS

2.1 Objetivo geral

Identificar no transcriptoma do fungo patógeno *Paracoccidioides brasiliensis* prováveis RNAs não-codificadores.

2.2 Objetivos específicos

a) Propor um método computacional eficiente que permita classificar seqüências de RNA quanto ao seu potencial de codificar uma proteína;

b) Gerar uma lista com os candidatos mais prováveis a RNAs não-codificadores do transcriptoma do fungo *Paracoccidioides brasiliensis*.

2.3 Justificativa

Apesar de já existirem programas para detecção de ncRNAs em seqüências de transcriptomas, esses não são adaptados a seqüências provenientes de projetos de seqüenciamento de transcriptoma de organismos negligenciados, que possuem diversas limitações se comparadas às provenientes de grandes projetos. Além disso, esses programas são computacionalmente onerosos, lentos, e necessariamente utilizam-se de análises comparativas, o que nem sempre é interessante para o estudo de organismos dos quais se têm pouca informação.

3. MATERIAIS E MÉTODOS

3.1 Estrutura física

Os programas usados nesse trabalho foram instalados no Laboratório de Bioinformática da Biologia Molecular da Universidade de Brasília, cujas máquinas e sistema estão descritas em (Brígido *et al*, 2005). Algumas etapas de processamento computacional foram executadas no Laboratório de Bioinformática da Empresa Brasileira de Pesquisa Agropecuária (EMBRAPA) – Recursos Genéticos e Biotecnologia, em Brasília-DF, sob supervisão do Dr. Roberto Coiti Togawa.

3.2 Conjuntos de treinamento

Como a estratégia do presente trabalho é classificação de duas classes por AM supervisionado, o conjunto de treinamento é subdividido em conjuntos positivo (RNA mensageiro) e negativo (RNAs não-codificadores). O positivo foi montado a partir das seqüências do banco de dados Swiss-Prot versão 50.8 (Wu *et al*, 2006a), de 3 de outubro de 2006, cujo arquivo de seqüências foi obtido no dia 5 de outubro de 2006. O conjunto negativo foi montado a partir dos bancos de dados NONCODE, Rfam e parte do RNAdb, todos obtidos no dia 7 de outubro de 2006. O subconjunto RNAdb-asoverlaps não foi incluído no conjunto final devido à tradução *in silico* ocorrer nas 6 fases de leitura e esses transcritos possuírem

muitas características similares a transcritos codificadores, e essa semelhança poderia influenciar de forma negativa o aprendizado do algoritmo MVS.

A Tabela 4 sumariza como o conjunto de treinamento é composto.

Tabela 4. Composição do conjunto de treinamento.

Conjunto positivo (RNAs codificadores)		
Componente	**Conteúdo**	**Quantidade de seqüências**
Seqüências do EMBL	cDNAs correspondentes às proteínas do Swiss-Prot	104.247
Conjunto negativo (RNAs não-codificadores)		
Componente	**Conteúdo**	**Quantidade de seqüências**
NONCODE	ncRNAs das classes estrutural e semelhante a mRNA, exceto RNAs ribossomais e transportadores.	6.200
Rfam	Exclusivamente RNAs não-codificadores estruturais.	45.644
RNAdb unificado	ncRNAs de todas as classes.	213.849

3.3 Programas geradores de atributos

O único programa desenvolvido por terceiros usado especificamente na geração de atributos foi o CAST (Promponas et al, 2000). Para tradução dos transcritos foram usados, paralelamente, os programas OrfPredictor (Min et al, 2005) e ANGLE (Shimizu et al, 2006). Também foram usados *scripts* desenvolvidos localmente em linguagem computacional PERL (*Practical Extraction and Report Language*).

3.4 Algoritmos de aprendizagem de máquina

MVS: Foi selecionada a biblioteca de Máquina de Vetores de Suporte LIBSVM versão 2.84, implementada em linguagem computacional C (Chang e Lin, 2006), atualização de abril de 2007, e algumas de suas ferramentas auxiliares. Scripts em linguagem PERL foram escritos para formatar entradas e analisar saídas do LIBSVM. Usou-se o formato recomendado pelos autores, em configuração esparsa (atributos com valor igual a zero são omitidos).

Naive Bayes: Foram utilizados o algoritmo Naive Bayes Classifier versão 2.7, atualização de fevereiro de 2007, e ferramentas auxiliares (Borgelt, 2007). Scripts em linguagem PERL foram escritos para formatar entradas e analisar saídas do NBC, e fazer a conversão de formatos entre NBC (configuração densa) e LIBSVM (configuração esparsa).

3.5 Algoritmo de comparação entre RNAs

Após uma busca dos programas disponíveis, o RSmatch2.0 (Khaladkar et al, 2007) foi selecionado por ser um programa que usa relativamente poucos recursos computacionais, permitindo com poucos ajustes fazer pesquisa de um FASTA múltiplo contra um banco de dados composto por outro FASTA múltiplo, além de não exigir que o usuário conheça o motivo que deve ser procurado. As estruturas secundárias, que são os arquivos de entrada do RSmatch, foram geradas pelo programa RNAfold, componente do pacote Vienna (Hofacker, 2003). Como o programa RSmatch é adaptado para análise de seqüências pequenas, os parâmetros foram reajustados para tratar seqüências de ncRNAs longos: -p pmatch (modo de busca *pairwise* local, onde supõe-se que a seqüência usada na busca (*query*) compartilhe um motivo de estrutura secundária desconhecido com as seqüências do banco de dados), -W 300 (tamanho da janela de leitura, baseado no fato que a maioria das famílias de ncRNAs descritas possuem entre 50 e 150 letras (Griffiths-jones et al, 2005)), -S 0.3 (tamanho da janela deslizante igual a 0,3 vezes o tamanho da janela de leitura), -g -3 (penalidade de abertura de lacuna diminuída para favorecer análise de trechos mais longos contíguos). São consideradas relevantes as regiões de *hits* que aparecem em pelo menos dois organismos diferentes, com um tamanho de janela de pelo menos 35 letras, considerando adjacências de 30 letras a jusante e a montante da seqüencia analisada, e em que regiões pareadas (dupla-fita) compreendam no mínimo 15% da estrutura secundária. Todos esses critérios foram escolhidos empiricamente, após análise dos arquivos de saída do RSmatch e rodadas-piloto.

4. RESULTADOS

4.1 Fluxo do programa PORTRAIT

O programa descrito nesse trabalho opera em ambiente Linux por linha de comando, não possuindo nenhuma interface gráfica. Foram desenvolvidas duas versões, uma para instalação e uso locais, e outra como *webserver* com acesso pela Internet, via *browser*, com interface CGI rodando scripts de PERL no lado do servidor. O programa foi batizado de **P**rediction **o**f T**r**anscrip**t**ome nc**R**NA by ***A**b **I**nitio* Me**t**hods, ou simplesmente PORTRAIT.

A partir da submissão de um arquivo de entrada contendo múltiplas seqüências no formato FASTA, é feita a tradução para obtenção das seqüências de aminoácidos, e a seguir as seqüências são divididas em transcritos com ORF ou sem ORF. Diversas características são extraídas das seqüências de entrada, que são formatadas e fornecidas a um classificador MVS dependendo do grupo que as seqüências se inserem (com ou sem ORF). Os resultados da

classificação são analisados e retornados ao usuário na forma de seqüências de prováveis ncRNAs em formato FASTA, e a confiança do MVS para cada predição. A figura 5 mostra um resumo do processo inteiro.

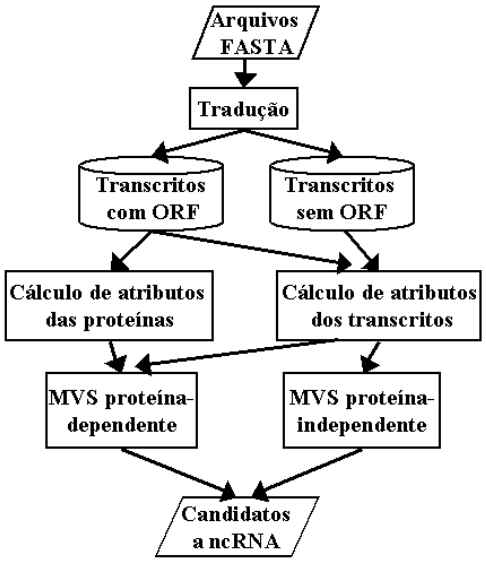


Figura 5. Fluxograma mostrando o fluxo de dados do programa, desde a recepção do arquivo do usuário até a emissão dos arquivos de saída.

4.2 Construção do conjunto de treinamento

O conjunto de treinamento (dbTR) é dividido em conjunto de exemplos positivos (dbCOD – Banco de Dados de transcritos Codificadores), constituído por seqüências de proteínas anotadas e bem caracterizadas, e pelo conjunto de exemplos negativos (dbNC - Banco de Dados de transcritos Não-Codificadores), composto por seqüências de RNAs não-codificadores. Ambos os conjuntos são constituídos de seqüências de nucleotídeos e suas respectivas traduções. Também estão presentes em ambos os conjuntos transcritos que não possuem proteína predita. O formato de trabalho dos conjuntos usado foi o FASTA múltiplo (Lesk, 2002).

4.2.1 Conjunto dbCOD

O arquivo do banco Swiss-Prot v.50.8 (formato FASTA), contendo 234.112 seqüências de proteínas, foi submetido ao programa CD-HIT para eliminar seqüências redundantes. O parâmetro –c 0.7 fornecido ao programa garante que seqüências com identidade igual ou maior que 70% (limite de agrupamento) terão apenas um representante dentro do conjunto de saída.

O arquivo de saída do programa CD-HIT é um arquivo FASTA com seqüências não-redundantes de proteínas. Um script de PERL (id_extractor.pl) foi usado para extrair de cada seqüência sua Swiss-Prot ID correspondente, que é um código único e não-ambíguo de cada

proteína. Dessa forma foram extraídos 115.115 códigos. A partir das IDs extraídas, um outro *script* (Swiss-Prot_parser.pl) foi usado para extrair do arquivo da versão completa e detalhada do Swiss-Prot, chamada *flatfile*, o código de acesso correspondente ao banco de dados de nucleotídeos EMBL (Cochrane *et al*, 2006). Como algumas proteínas não possuem anotadas suas referências cruzadas ao banco de dados de nucleotídeos EMBL, essas seqüências foram automaticamente descartadas. Além disso, algumas proteínas possuem mais de uma referência cruzada; nesse caso, todas as referências são analisadas.

O arquivo de entradas do EMBL, contendo inicialmente 220.202 seqüências, foi filtrado com o *script* seqs_repetidas.pl para exclusão de seqüências repetidas. O número de entradas foi reduzido então para 117.951 seqüências e usado para baixar as seqüências correspondentes do EMBL - versão de 11 de outubro de 2006 (Rodriguez-Tomé et al, 1996) - em formato FASTA, utilizando o serviço EBI dbfetch (Harte *et al*., 2004) por meio do script local my_dbfetch.pl. O script foi programado para rejeitar entradas EMBL inválidas (total 31 códigos), e seqüências de genomas e cromossomos inteiros ou segmentos desses (total 868 seqüências), gerando um total de 105.115 seqüências de cDNAs inteiras (incluindo ORFs, UTRs e introns) que codificam um conjunto não-redundante de proteínas, mas que *per se*, em forma de nucleotídeos, são redundantes.

Foram excluídas desse conjunto seqüências maiores do que 65.535 letras - para evitar problemas de alocação de memória em programas usados posteriormente - e menores do que 80 letras - um limite que, segundo Liu *et al* (2006), exclui transcritos que produzem um peptídeo pequeno demais para ser analisado - reduzindo a quantidade de seqüências para um total de 104.247. A seguir, as seqüências foram novamente submetidas a eliminação de redundância pelo programa BLASTCLUST (Altschul et al, 1997), configurado para agrupar seqüências nucleotídicas (-p F) com densidade de pontuação de 50% (-S 0.5) em pelo menos 50% do comprimento das sequências (-L 0.5), sendo a comparação iniciada ao encontrar uma palavra com pelo menos 18 letras (–W 18). Esses parâmetros usados são bastante estringentes, sendo que valores mais restritivos testados geraram quantidade de seqüências muito similar à configuração usada (dados não mostrados). Talvez pelo fato da redundância já ter sido eliminada no conjunto de proteínas, o programa BLASTCLUST reduziu o conjunto em apenas 2%. O conjunto não-redundante foi então submetido à predição de ORFs pelo programa ANGLE (Shimizu *et al*, 2006), sendo selecionado como produto protéico a proteína com maior pontuação pelo algoritmo de Programação Dinâmica dentre as 6 janelas de leitura. A interface com o programa ANGLE foi mediada pelo *script* multiANGLE.pl, escrito localmente.

Os transcritos que tiveram ORFs preditas pelo programa ANGLE são tomados como dados de treinamento que, em conjunto com suas respectivas seqüências codificadoras, foram denominados dbTR_OP (Banco de Dados de Treinamento de Transcritos com ORF Presente). Apesar das seqüências de transcritos do banco dbCOD terem se originado se seqüências protéicas, espera-se que alguns desses mRNAs não tenham suas ORFs detectadas por um erro do programa ANGLE, ou por esse transcrito estar anotado erroneamente no banco. Assim, os transcritos que não tiveram uma ORF detectada são integrados ao conjunto de treinamento dbTR_OA (Banco de Dados de Treinamento de Transcritos com ORF Ausente). Como a quantidade de seqüências ainda era muito alta, procedeu-se a uma segunda eliminação de redundâncias das ORFs com o programa CD-HIT configurado com o parâmetro –c 0.7. Dessa forma, a quantidade de seqüências foi reduzida em 46%. O esquema de montagem do conjunto dbCOD está ilustrado na Figura 6.

4.2.2 Conjunto dbNC

Os três bancos de dados de ncRNA (NONCODE, Rfam e RNAdb) foram uniformizados pelos scripts format_NONCODE.pl, format_Rfam.pl e format_RNAdb.pl, respectivamente, e então reunidos em um só e as seqüências com mais de 65.535 letras e menos de 80 letras foram excluídas para evitar os mesmos problemas descritos para o dbCOD. É importante ressaltar que a exclusão de seqüências menores do que 80 letras exclui diversos tipos de ncRNA, como miRNAs, siRNAs e piRNAs presentes nos bancos Rfam e RNAdb, por exemplo, restando no conjunto principalmente os ncRNAs médios e longos. O arquivo resultante, com 265.691 seqüências, foi submetido ao mesmo processo de montagem do dbCOD, de eliminação de redundância pelo programa BLASTCLUST com os parâmetros –L 0.5, –S 0.5 e –W 18, reduzindo o tamanho total do conjunto em 25%. O mesmo processo de predição de ORFs feito para o conjunto dbCOD foi usado no dbNC, sendo que transcritos com ORFs preditas integraram o conjunto dbTR_OP. Transcritos sem predição de proteína foram integrados ao conjunto de treinamento dbTR_OA. Foi feita uma segunda eliminação de redundância das ORFs, da mesma forma feita com o conjunto dbCOD. Observou-se uma redução de 8% na quantidade de seqüências.

O esquema de montagem do conjunto dbNC está ilustrado na Figura 6.

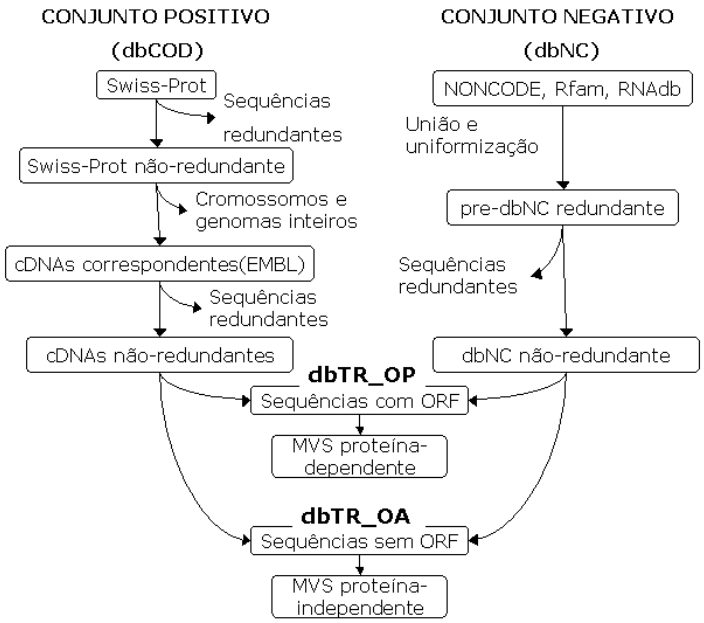


Figura 6. Esquema ilustrativo dos passos executados para obter os conjuntos de treinamento. Um classificador MVS dependente de proteína é induzido a partir de seqüências codificadoras putativas dos conjuntos positivo e negativo, e um classificador MVS independente é induzido apenas com as seqüências que não apresentaram ORFs nos dois conjuntos.

A tabela 5 mostra diversos aspectos estatísticos do conjunto de treinamento.

Tabela 5. Características dos conjuntos de treinamento.

	dbCOD_OA	dbCOD_OP	dbNC_OA	dbNC_OP
Letras	3.124.032	181.620.572	24.505.453	45.452.385
Quantidade de seqüências	4.043	51.329	47.887	22.647
Maior seqüência	31.586	64.165	8.351	34.635
Menor seqüência	80	184	80	183
Tamanho médio	773	3.538	512	2.007
%GC	0,40	0,45	0,42	0,44

4.3 Construção dos conjuntos de teste

Os conjuntos de teste constituem os dados desconhecidos que o algoritmo de MVS deve classificar. Foram construídos três conjuntos: um contendo seqüências de DNA geradas aleatoriamente, denominado dbRD, outro constituído por cDNAs do fungo *Paracoccidioides brasiliensis*, denominado dbPB, e outro composto de seqüências de fungos filogeneticamente próximos ao *P. brasiliensis*. Todos os conjuntos foram filtrados para seqüências menores do que 80 letras.

4.3.1 Conjunto dbRD

O conjunto dbRD (Banco de Dados de Seqüências de DNA geradas aleatoriamente) foi desenvolvido a partir de um script de PERL (fabrica_DNA.pl) que gerou aleatoriamente 3.000 seqüências de DNA com tamanhos podendo variar de 80 a 3.000 nucleotídeos. A freqüência de nucleotídeos e o tamanho das seqüências, respeitados os limites máximo e

mínimo, variaram aleatoriamente de acordo com o interpretador do código, sem nenhum tipo de interferência por parte do programa ou do usuário.

4.3.2 Conjunto dbPB

Outro conjunto de teste é composto pelas seqüências montadas dos cDNAs seqüenciados durante o projeto do Genoma Funcional e Diferencial do Fungo *P. brasiliensis* (PbAESTs) (Felipe *et al*, 2003). Esse conjunto possui 6.022 seqüências.

4.3.3 Conjunto dbFG

Fungos próximos filogeneticamente a *P. brasiliensis* foram determinados pela ferramenta Taxonomy Browser (http://www.ncbi.nlm.nih.gov/Taxonomy), e pela mesma ferramenta foram obtidas seqüências de ESTs dos seguintes organismos: *Aspergillus niger, Ajellomyces capsulatum, Schizosaccharomyces pombe, Saccharomyces cerevisiae* e *Cryptococcus neoformans*. Os três últimos citados são considerados *outgroups* da análise filogenética.

Obteve-se um total de 137.629 seqüências.

Aspectos estatísticos básicos dos conjuntos de teste estão mostrados na tabela 6.

Tabela 6. Características dos conjuntos de teste.

	dbRD_OA	dbRD_OP	dbPB_OA	dbPB_OP	dbFG_OA	dbFG_OP
Letras	1159646	3541380	329434	4374657	10110666	68311203
Qtde. de seqüências*	1059	1941	572	5449	30347	107244
Maior seqüência	2987	2999	2028	4079	1046	1146
Menor seqüência	82	203	107	228	81	184
Tamanho médio	1095	1824	576	804	333	637
%GC	0,50	0,50	0,48	0,50	0,41	0,48

*O somatório da quantidade de seqüências dos subconjuntos _OP e _OA não necessariamente totaliza um valor igual ao conjunto integral porque algumas seqüências podem ser excluídas da análise se tiverem menos de 80 letras ou se o conteúdo de caracteres "N" ultrapassar 20% do total de letras.

4.4 Vetor de características

Como os ncRNAs não possuem características únicas que possam diferenciá-los dos demais RNAs, a escolha dos atributos que compõem o vetor de características foi feita intuitivamente, orientada pela hipótese de que é possível induzir um classificador que faça a predição do potencial codificador de um RNA extraindo atributos das seqüências de transcritos não-codificadores e codificadores e de suas respectivas seqüências protéicas putativas preditas. Nesse trabalho, a hipótese testada é que as características extraídas tanto

das seqüências nucleotídicas quanto das protéicas podem ser usadas para classificá-las como codificadoras ou não-codificadoras.

Devido à grande quantidade de programas disponíveis para obter os dados de composição do vetor de características, a escolha foi orientada de acordo com prioridades. A primeira é que o programa seja de livre acesso para uso acadêmico. A segunda prioridade refere-se à implementação: são escolhidos os programas que aceitem processamento de larga escala, ou que possuam uma linguagem e implementação que permitam instalação estável nos sistemas operacionais e ambientes dos computadores do laboratório. A terceira prioridade considera a facilidade e conveniência da manipulação de dados de entrada e saída. Buscou-se utilizar as versões mais atualizadas disponíveis de cada programa. Todos os programas foram usados com os parâmetros padrão, a não ser quando mencionado.

Adotou-se um esquema de particionamento dos conjuntos para execução controlada e interruptível dos processos mais onerosos computacionalmente. Dessa forma evitou-se a perda de dados por motivos de falhas, como quedas de energia ou parada de execução por falta de memória disponível nas máquinas, uma preocupação que se justifica frente à grande quantidade de tempo exigida para execução de alguns programas, que em alguns casos passa de dois meses.

O vetor de características possui no total 8 características e 111 variáveis (modelo proteína-dependente), e 2 características e 88 variáveis (modelo proteína-independente). Os programas que só executam um processamento por vez foram adaptados para receber múltiplas entradas a cada execução, com controle também do arquivo de saída. O algoritmo de MVS tem seu funcionamento ideal quando os dados de atributos são normalizados, devendo o domínio dos valores ficar entre 0 e 1 ou –1 e 1 (Chang e Lin, 2006). Os dados qualitativos foram então transformados em quantitativos, e para cada atributo foi buscado um método de normalização para restringir o domínio de valores a [0, 1]. Todos os dados gerados foram incorporados a um *script* único que reúne todos os dados, normaliza alguns, e codifica-os no formato de entrada do MVS, sendo portanto a etapa final na construção dos conjuntos de treinamento e de teste.

Composição de nucleotídeos (inclusive di- e tri-nucleotídeos): O programa para cálculo das proporções relativas de nucleotídeos, dinucleotídeos e trinucleotídeos foi desenvolvido localmente, por ser relativamente simples, fornecendo também uma vantagem de manipulação já que foi concebido para aceitar múltiplas seqüências em uma única execução e já fornece os dados em formato de entrada do MVS. A quantificação de nucleotídeos é feita de uma só vez, calculando a freqüência de cada nucleotídeo

individualmente. Os dinucleotídeos e trinucleotídeos foram calculados apenas para a primeira fase de leitura, por meio de uma janela deslizante do tamanho de duas letras ou de três letras, sendo descartados o último ou os dois últimos nucleotídeos quando presentes, no caso da contagem de dinucleotídeos e de trinucleotídeos, respectivamente. A normalização foi feita dividindo-se a quantidade de um nucleotídeo, dinucleotídeo ou trinucleotídeo específico pelo total de elementos calculados. No total, esse atributo fornece um vetor de 84 dimensões (84 variáveis) para o MVS.

Comprimento da fase aberta de leitura: O programa que calcula o tamanho das ORFs foi desenvolvido localmente, já realizando a normalização e adequação ao formato de entrada para o MVS. Sendo uma variável quantitativa sem um método satisfatório de normalização, optou-se por transformá-la em uma variável qualitativa de quatro categorias, envolvendo os intervalos: menor que 20, entre 20 e 60, entre 60 e 100 e maior que 100 aminoácidos. Dessa forma, esse atributo contribui com um vetor de 4 dimensões (4 variáveis) para o MVS.

Composição de aminoácidos: O programa que calcula a composição de aminoácidos foi desenvolvido localmente, já realizando a normalização e adequação ao formato de entrada para o MVS. A normalização foi feita dividindo-se a freqüência individual de cada resíduo pelo total de resíduos da seqüência, sendo obtidas assim um vetor de 20 dimensões (20 variáveis) para o MVS.

Predição de ponto isoelétrico da proteína: Foi desenvolvido um *script* de PERL, adaptado de código em linguagem computacional C descrito por Kozlowski (2007). Esse *script* apresenta os mesmos resultados e aproximadamente a mesma velocidade que o programa *iep*, componente do pacote EMBOSS (Rice *et al*, 2000), mas com a vantagem de dispensar a instalação do pacote EMBOSS. Ele foi usado para gerar predições de ponto isoelétrico das seqüências protéicas. O valor de ponto isoelétrico foi analisado e normalizado por um *script* escrito localmente. Como o ponto isoelétrico é um valor que pode variar de 0 a 14, a normalização foi feita dividindo-se o valor obtido por 14, produzindo assim um vetor unidimensional (1 variável) para o MVS.

Complexidade da proteína (entropia composicional): O programa CAST (Promponas *et al*, 2000) foi instalado localmente para estimar a complexidade das seqüências protéicas. Alguns programas foram rejeitados, como o algoritmo SEG, que apresenta um desempenho inferior ao CAST na tarefa a que se propõe (Kreil e Ouzounis, 2003), o algoritmo BIAS (Kuznetsov e Hwang, 2006) é inconveniente para esse trabalho por necessitar que o usuário forneça os motivos que serão considerados de baixa complexidade, e o CARD (Shin e Kim, 2005), que apesar de ter desempenho similar à do programa escolhido, foi rejeitado por

demandar um tempo de processamento maior (é inadequado para processamento de larga escala) e por sua página de *download* estar fora do ar no tempo de realização desse trabalho. O algoritmo CAST mascara com "X" as letras pertencentes a um segmento considerado repetitivo, de baixa complexidade. A quantidade de resíduos mascarados, então, é determinada subtraindo-se a quantidade de "X" presentes na seqüência mascarada da quantidade de "X" na seqüência protéica original, já que a letra "X" é também encontrada na seqüência original e representa problemas no seqüenciamento. A normalização é feita dividindo a quantidade de "X" presentes exclusivamente na seqüência mascarada pelo tamanho total da seqüência, fornecendo um vetor unidimensional (1 variável) para o MVS.

Hidrofobicidade média da proteína: Foi incorporada ao programa *portrait*.pl uma sub-rotina que calcula a hidrofobicidade média da proteína, portando para PERL o código originalmente escrito em linguagem computacional C constante no trabalho de (Kyte e Doolittle, 1982). O algoritmo possui uma janela deslizante com o tamanho de três letras, e desloca-se de letra a letra, podendo deixar de ler um ou dois resíduos no final na seqüência devido ao tamanho da janela. O tamanho da janela foi escolhido por ser o menor possível, já que se espera seqüências protéicas muito pequenas traduzidas a partir de seqüências de transcritos truncadas e pequenas. A tabela de hidropaticidade de resíduos descrita em (Kyte e Doolittle, 1982) tem seu domínio dentro do intervalo [-4,5, 4,5]. Essa tabela foi normalizada pela fórmula: (X + 4,5) / 9, para que o domínio fique restrito ao intervalo [0, 1]. A trinca é lida pela janela, e os valores normalizados são somados e divididos por 3. Quando a seqüência inteira foi lida, todos os valores são somados e divididos pelo total de leituras de janela, normalizando a leitura e reduzindo-a a um valor único, representativo de uma hidrofobicidade média da proteína. Esse atributo contribui com um vetor unidimensional (1 variável) para o MVS.

Os atributos usados, assim como os programas selecionados para extração dos atributos e quantidade de variáveis estão sumarizados na tabela 7.

Tabela 7. Descrição dos atributos que compõem o vetor de características.

Atributo	Programa	Quantidade de variáveis
1. Composição de nucleotídeos	PERL – *portrait*.pl	84
2. Comprimento da fase aberta de leitura	PERL – *portrait*.pl	4
3. Composição de aminoácidos	PERL – *portrait*.pl	20
4. Predição de ponto isoelétrico da proteína	PERL – *portrait*.pl	1
5. Complexidade da proteína (entropia composicional)	CAST	1
6. Hidrofobicidade média da proteína	PERL – *portrait*.pl	1
7. Comprimento do transcrito*	PERL – *portrait*.pl	4

*Atributo calculado apenas para modelo proteína-independente.

4.5 Configuração do programa MVS

Existem diversas implementações do algoritmo de MVS na literatura. As duas mais usadas em problemas biológicos são SVM^{light} (Joachims, 1999) e LIBSVM (Chang e Lin, 2006). Nesse trabalho a implementação utilizada foi o LIBSVM pelo fato de ter uma documentação detalhada, diversos exemplos e ferramentas que facilitam seu uso, tornando-o mais acessível para o usuário novato. O problema solucionado pelo algoritmo foi de classificação binária (apenas duas classes: codificador [classe positiva] e não-codificadora [classe negativa]), com configuração C-SVM (domínio do parâmetro C é $\{0, +\infty\}$), sendo usado o *kernel* RBF (*Radial Basis Function*), pois mostrou-se que este é o ideal para o tipo de problema abordado nesse trabalho (Liu *et al*, 2006), além de ser recomendado pelo próprio autor do programa LIBSVM por ser uma função mais abrangente, podendo englobar outros *kernels*, como linear e polinomial (Chang e Lin, 2006).

4.5.1 Determinação dos parâmetros ótimos e treinamento do MVS

Para determinação dos parâmetros C (penalização do erro de generalização) e γ (gama - "curvilinidade" do *kernel* RBF) foi usado o *script* grid.py, que acompanha o pacote LIBSVM. A otimização dos parâmetros é feita por validação cruzada (*cross-validation)* estratificada (ilustrada na Figura 3) usando o *script* subset.py que acompanha o pacote LIBSVM. A determinação dos parâmetros ótimos foi feita seguindo-se a recomendação dos autores do LIBSVM (Chang e Lin, 2006): inicialmente foi montado aleatoriamente um subconjunto de 20.000 instâncias a partir do conjunto de treinamento original com número igual de representantes de cada classe, sendo nesse subconjunto feita uma busca "grosseira" do par ótimo (C; gama) por validação cruzada estratificada de 10 vezes, usando o intervalo de valores (C; gama) padrão do script *grid.py*. Determinou-se para o dbTR_OA que, no intervalo buscado, o par (C; gama) ótimo foi (8,0; 8,0), com o maior índice de acurácia: 92,9% (Figura 7a), e para o dbTR_OP, o (C; gama) ótimo foi também (8,0; 8,0) com acurácia 90,79% (Figura 7b) . De posse desse valor foi feita uma busca "fina" nos parâmetros vizinhos a esses: C entre 2 e 32, gama também entre 2 e 32, com incremento de 0,25 em C e gama a cada iteração para ambos conjuntos de dados. O par ótimo final para o conjunto dbTR_OA obtido foi (3,3; 26,9), com acurácia de 93,2% (Figura 7c), e para o dbTR_OP foi (2,8; 16,0), acurácia de 90,90% (Figura 7d).

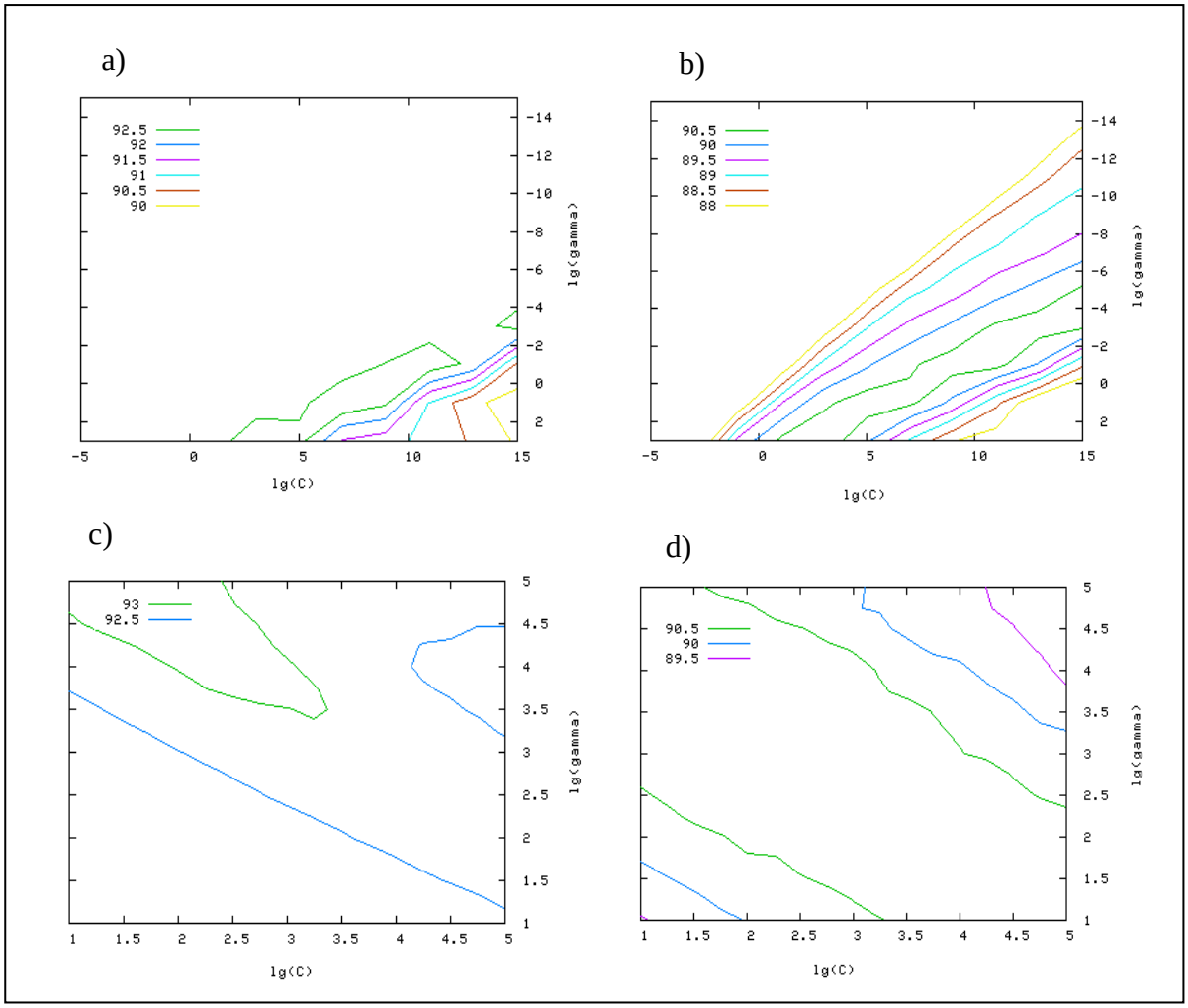


Figura 7. Busca dos parâmetros C e gama ótimos para o MVS. Busca inicial ou "grosseira" nos conjuntos dbTR_OA (a) e dbTR_OP (b). Busca avançada ou "fina" nos conjuntos dbTR_OA (c) e dbTR_OP (d). A legenda na porção superior esquerda, interna aos gráficos, indica a acurácia que cada cor de linha ("curvas de nível") representa. Devido à configuração do programa que gerou os gráficos, as acurácias estão arredondadas para uma casa decimal.

Os autores do LIBSVM recomendam que a busca "fina" seja feita no conjunto inteiro, diferente do que foi feito nesse trabalho. Isso não pôde ser realizado devido à alta demanda computacional que esse processo exigiria, uma vez que o processo realizado no subconjunto com apenas 20.000 instâncias demorava em média 4 dias de execução ininterrupta em máquinas de alto desempenho; o conjunto inteiro, com 126.039 instâncias, possivelmente demoraria até mais do que os esperados 25 dias.

Foram gerados dois modelos a partir do treinamento feito com os conjuntos de treinamento inteiros dbTR_OP e dbTR_OA usando os parâmetros C e gama ótimos. Durante o treinamento o programa LIBSVM foi configurado com a opção –b 1 para geração de estimativas de probabilidade.

Devido ao desbalanceamento na proporção de exemplos positivos e negativos no dbTR_OA (1 positivo : 12 negativos), aplicou-se no treinamento desse conjunto o esquema de penalização –w 1 –w-1 12, ou seja, toda classificação negativa possui uma penalidade de peso 12.

Os modelos MVS_OA e MVS_OP, treinados a partir dos parâmetros otimizados determinados nessa etapa, são os efetivamente usados como classificadores no programa PORTRAIT.

4.6 Configuração do programa naive Bayes (nB)

A formatação dos dados foi feita por um *script* que faz a conversão entre os formatos LIBSVM esparso e nB denso. A partir do arquivo de entrada já formatado foi gerado um arquivo com o domínio dos valores de atributos do conjunto de treinamento por meio do programa *dom*. Em seguida, foi induzido um classificador com o conjunto de treinamento por meio do programa *bci*. O programa *bcx* realiza predições para os conjuntos usando o modelo induzido. Todos os programas foram executados com as configurações padrão. As medidas de eficiência foram calculadas pelos programas *xmat* e *corr*. Os programas *dom, bci, bcx, xmat* e *corr* fazem parte do pacote nB (Borgelt, 2007).

4.7 Medidas de eficiência

4.7.1 Validação cruzada

Para avaliação do classificador gerado pelo MVS usou-se o próprio programa de treinamento do LIBSVM (*svm-train*), sendo feita uma validação cruzada estratificada de 10 vezes. Para o algoritmo de nB um *script* foi escrito localmente para fazer a geração dos subconjuntos e etapas de treinamento/teste, com 10 vezes. Em ambos processos foi usado o conjunto de treinamento completo (dbTR_OP e dbTR_OA), com 126.039 instâncias. Segundo essa técnica, a acurácia para o programa PORTRAIT é 92,4% e do naive Bayes é 75,3%.

4.7.2 Curvas ROC

As curvas ROC foram obtidas da mesma forma para todos os métodos de classificação. O classificador final, treinado, é usado para gerar predições sobre o conjunto de treinamento inteiro. As predições e respectivas confianças são comparadas aos rótulos do conjunto de treinamento e são uniformizados, sendo fornecidos ao programa *PERF* versão 5.10 (disponível em http://kodiak.cs.cornell.edu/kddcup/software.html). Esse programa

fornece os pontos para plotagem da curva ROC, além de diversas medidas de eficiência, incluindo AAC. Os pontos de plotagem são usados para gerar as curvas por meio do programa *gnuplot* versão 4.2 (disponível em http://www.gnuplot.info) compostas por linhas construídas a partir da interpolação linear dos pontos fornecidos (Figura 8). Não foi feita a curva ROC do classificador CPC porque seu esquema de pontuação não possui uma escala normalizada de probabilidade, o que impossibilita essa análise. O programa CONC também não foi analisado porque o tempo e poder computacional exigidos para a predição do conjunto dbTR inteiro são proibitivos para a estrutura física disponível.

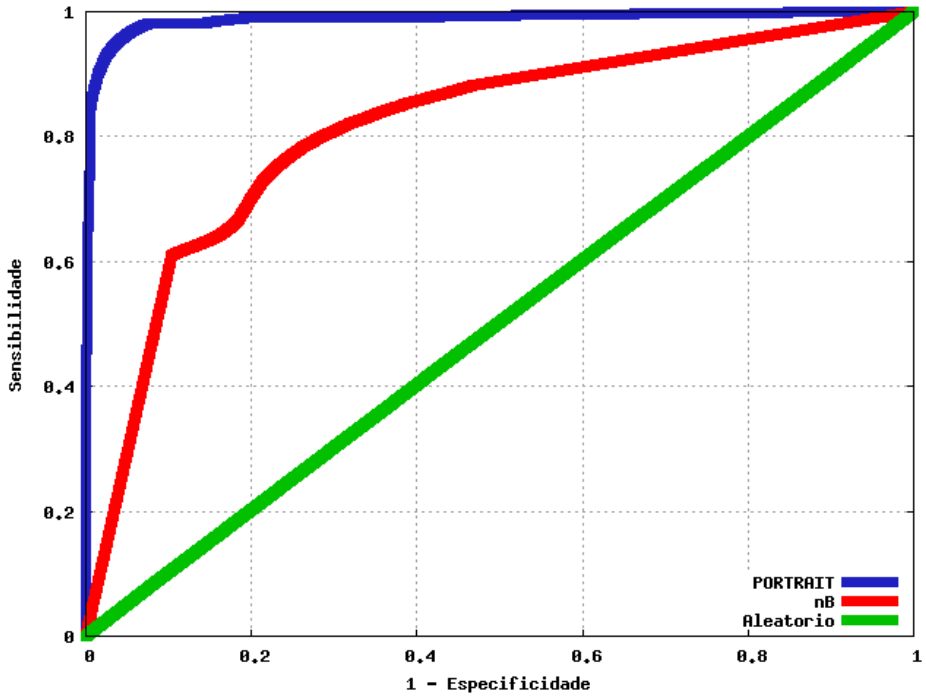


Figura 8. Curvas ROC plotadas a partir do desempenho dos classificadores no conjunto de treinamento inteiro (dbTR).

As AAC dos classificadores PORTRAIT, naive Bayes e aleatório foram, respectivamente, 0,988, 0,813, e 0,500, conforme determinado pelo programa *PERF*.

4.7.3 Matrizes de confusão e medidas derivadas

As matrizes de confusão (explicadas na Tabela 3), medidas derivadas e tempos de execução foram calculadas manualmente e confirmadas pelo programa *PERF* para os classificadores MVS, nB e CPC e para classificações aleatórias (Tabela 8 e Tabela 9).

Tabela 8. Matriz de confusão dos classificadores induzidos. Foi usado o conjunto de treinamento dbTR como objeto de avaliação.

		Classe predita					
		Classificador PORTRAIT		Classificador nB		Classificador aleatório	
		Proteína	**ncRNA**	**Proteína**	**ncRNA**	**Proteína**	**ncRNA**
Classe real	**Proteína**	51.789	2.033	41.992	13.380	30.188	25.182
	ncRNA	3.583	68.739	16.697	53.837	38.283	32.253

		Classe predita			
		Classificador CPC		Classificador CONC	
		Proteína	**ncRNA**	**Proteína**	**ncRNA**
Classe real	**Proteína**	50.992	4.453	*	*
	ncRNA	7.239	63.431	*	*

*****Parâmetros não estimados devido ao tempo e poder computacional elevados exigidos pela tarefa.

Tabela 9. Medidas de eficiência calculadas para o conjunto de treinamento dbTR e tempo de execução para análise do conjunto dbPB.

	Acurácia	Especificidade	Sensibilidade	Medida-F	Tempo (min.)*
Classificador PORTRAIT**	95,6%	93,5%	97,2%	94,9%	21,6
Classificador nB**	76,1%	75,8%	76,3%	73,6%	16,1
Classificador CPC	90,7%	89,7%	91,9%	89,7%	1.789,7
Classificador Aleatório	49,1%	45,6%	54,6%	49,7%	0,07
Classificador CONC	†	†	†	†	>17.280

*Todos programas foram executados na mesma máquina (processador duplo Intel® XEON™ 1.80MHz x86 (Dual Core 32 bits) com 512Mb RAM) .

**Conforme média das classificações dos modelos induzidos dependente (_OP) e independente (_OA) de proteína.

† Parâmetros não calculados devido ao tempo e poder computacional elevados exigidos pela tarefa.

4.7.4 Contribuição individual dos atributos

O valor discriminatório de cada variável de atributo pode ser obtido pela estimativa de pesos de atributos. Para isso, foi usado o *script* fselect.py, uma ferramenta do pacote LIBSVM. Os valores das 10 variáveis com os melhores e piores pesos estão mostrados na tabela 10. A tabela completa pode ser encontrada no anexo 1.

Tabela 10. Pontuação atribuída a cada variável por sua contribuição para a separação de classes feita pelo MVS, e sua colocação relativa às demais contribuições de outras variáveis. Na porção esquerda estão listados as 10 melhores variáveis, e na porção direita, as 10 piores.

As 10 melhores variáveis	Pontuação	As 10 piores variáveis	Pontuação
1.ORF maior que 100	0.583153	102.dinucleotídeo CC	0.000445
2.ORF entre 60 e 100	0.514314	103.aminoácido G	0.000430
3.dinucleotídeo CG	0.173249	104.trinucleotídeo GGT	0.000422
4.aminoácido A	0.135667	105.trinucleotídeo AGC	0.000351
5.trinucleotídeo TAG	0.124019	106.nucleotídeo A	0.000310
6.trinucleotídeo TGT	0.111791	107.aminoácido Y	0.000234
7.trinucleotídeo ACG	0.109074	108.trinucleotídeo GAG	0.000233
8.trinucleotídeo TCG	0.106840	109.trinucleotídeo TGC	0.000097
9.dinucleotídeo CT	0.103361	110.trinucleotídeo GCT	0.000033
10.trinucleotídeo CGA	0.098901	111.ORF menor que 20	0.000000

4.8 Análise de predições para os conjuntos de teste e treinamento

Ambos classificadores, nB e PORTRAIT, foram usados para detectar ncRNAs nos conjuntos de teste – dbPB, dbRD e dbFG – e nos de treinamento – dbCOD e dbNC. A figura 9 mostra as predições para o classificador MVS, e a figura 10 mostra predições para o classificador nB. A tabela 11 mostra a quantidade apenas de instâncias classificadas como negativas pelos classificadores nos conjuntos de teste.

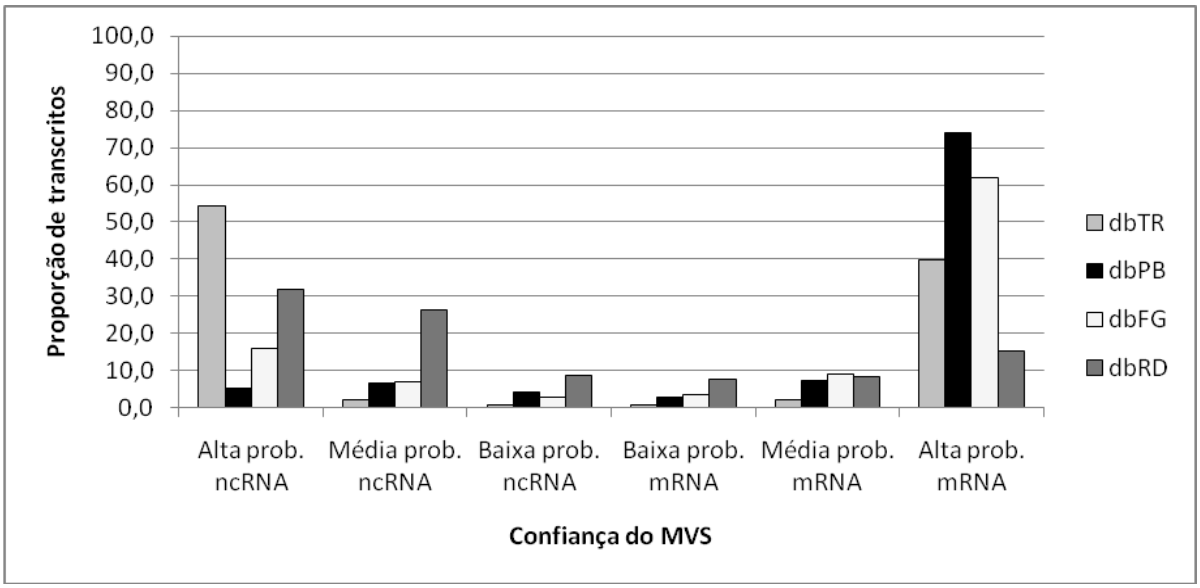


Figura 9. Distribuição de transcritos dos bancos de dados em função das probabilidades de predições (confiança) emitidas pelo MVS. Para ncRNA, uma alta probabilidade é a pontuação entre 0 e 20; média, 20 e 40; baixa, 40 e 50. Para mRNA, baixa probabilidade é a pontuação entre 50 e 60; média, entre 60 e 80; alta, entre 80 e 100.

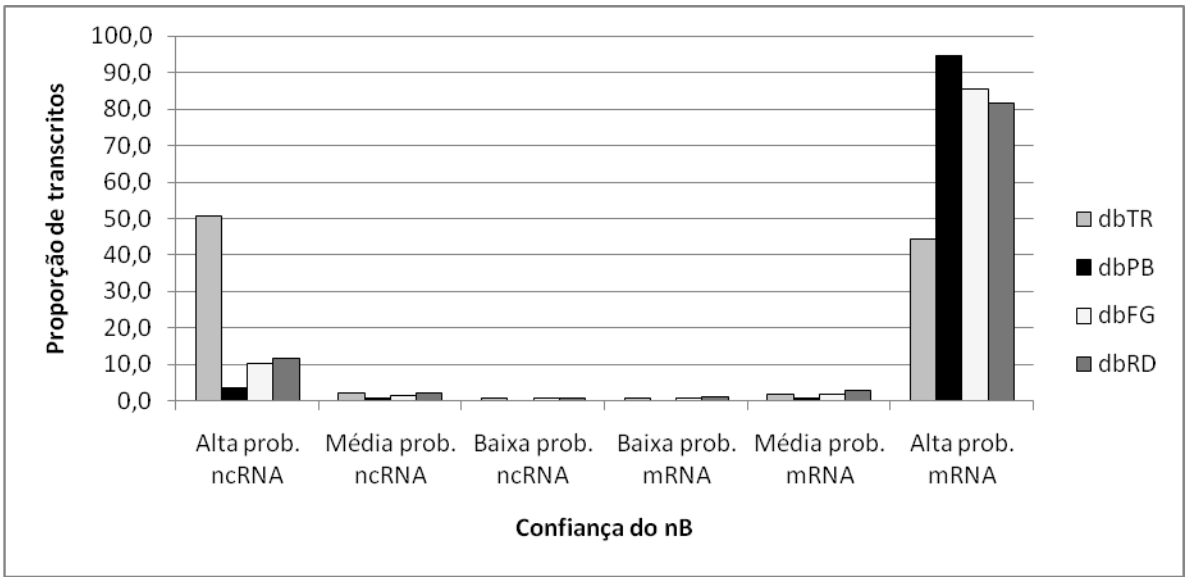


Figura 10. Distribuição de transcritos dos bancos de dados em função das probabilidades de predições (confiança) emitidas pelo nB. Para ncRNA, uma alta probabilidade é a pontuação entre 0 e 20; média, 20 e 30; baixa, 40 e 50. Para mRNA, baixa probabilidade é a pontuação entre 50 e 60; média, entre 60 e 80; alta, entre 90 e 100.

Tabela 11. Quantidade de instâncias (seqüências) classificadas como negativas (ncRNAs) pelos classificadores.

	dbPB	**dbRD**	**dbFG**
Classificador PORTRAIT	16,1%*	67,1%	25,9%
Classificador nB	4,3%	14,6%	12,1%
Classificador CPC	33,1%	100%	49,8%

* Uma seqüência foi ignorada por conter "N" em excesso.

As predições também foram avaliadas qualitativamente, pela contraposição às anotações geradas durante o projeto de seqüenciamento dos PbAESTs (Felipe *et al*, 2005) para cada um dos transcritos selecionados como ncRNAs (Figura 11).

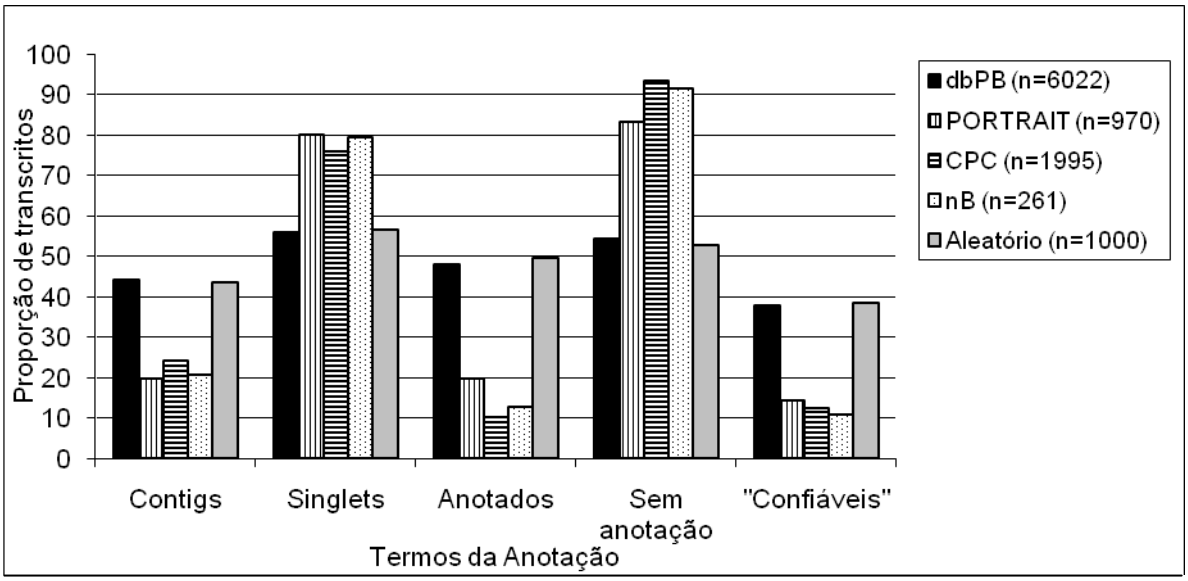
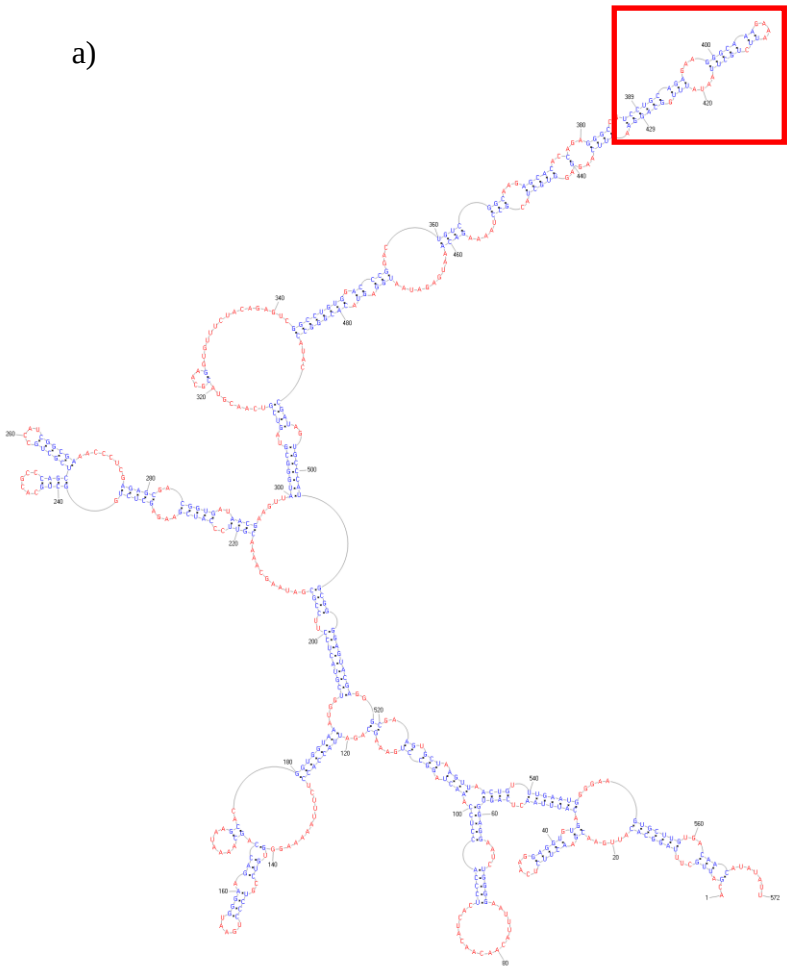


Figura 11. Distribuição dos transcritos classificados como ncRNA nesse trabalho, em função de anotações específicas a eles previamente atribuídas por (Felipe et al, 2005). O "n" (na legenda superior direita) indica a quantidade de transcritos preditos como ncRNAs (a não ser a classe de dbPB, que indica a totalidade de transcritos, incluindo mRNAs). O termo "Anotações confiáveis" refere-se a transcritos em cujas anotações estão ausentes as palavras: "*putative*", "*probable*" e "*hypothetical*".

4.9 Análise comparativa entre ncRNAs do dbPB e do dbFG

Como saída do programa RSmatch, obteve-se um total de 75299 *hits* estruturais. A partir desse arquivo, usando os *scripts* de análise da saída, foram identificados 3598 *hits* considerados relevantes, segundo os critérios explicitados na seção 1.6. Um exemplo de *hit* (ID: gi|50017 de Pb) com região considerada relevante está mostrado na Figura 12.

a)



b)

```
                                                 (((((((...(((((((....)).)))))....))).))))
                                                 (((((((...(((((((....)) ))))) ...))) ))))
gi|50017:                                     389 CUGCAGAGAAGGGCAAAGAAAUUCUGCUUAAUAUUUGGCAG 429
                                                  :::||:| ||:::|||| |  || ||::: |  |:| |:::
gi|622848|gb|T39031.1|,Scerevisiae:          77  AAGCAAAAAAUUUCAAAAACGUU-UGAAA-AGCUUU-GUUU 114


                                              ((((((((((...(((((  ((....)).)))))  ....))).))))))
                                              ((((((  ((  ..((((((.((....)).)))))).....))  ))))))
gi|50017:                                     387 UCCUGCAGAGAAGGGCA-AAGAAAUUCUGCUU-AAUAUUUGGCAGGA 431
                                                  |::||| |: | ::|:| ||| ||||||:|::   | :|  |||::|
gi|10181748|gb|BE759111.1|,Aniger:           83  UGAUGC-GC-AUUUGUAUAAGGAAUUCUGCAGUUCUGGU--GCAUUA 125


                                                   (((((((...(((((((....)).)))))).... .))).))))
                                                   (((((((...(((((((....)) ))))).....))) ))))
gi|50017:                                      389 CUGCAGAGAAGGGCAAAGAAAUUCUGCUUAAU-AUUUGGCAG 429
                                                   ::::|:: ||:::::|| || || ::::::||| |::| ::::
gi|54618492|gb|CV625628.1|,Acapsulatus:        79  AAAAAUGAAAAUGAUAAUAAUUU-GUUAUAAUAAUAU-UUUU 118


                                                ((((((((((...((((  (((... .)).)))))).....))).))))))
                                                ((((((((((...((((.(((.....)) ))))).. .))) ))))))
gi|50017:                                     387 UCCUGCAGAGAAGGGC-AAAGAA-AUUCUGCUUAAUAUUUGGCAGGA 431
                                                  :|:::||:: ||::|: :|| ||  || ::|::||  ::| |:::|:
gi|3344825|gb|AU008367.1|,Spombe:             203 ACGACCAUUCAACUGGAUAACAAUUUU-GCCAGAA-CGGU-GGUCGU 246
```

Figura 12. Exemplo de um hit estrutural de Pb considerado relevante. Em a), representação gráfica da estrutura secundária do transcrito inteiro de Pb (gi|50017) conforme desenhada pelo programa Pseudoviewer3 (Byun e Han, 2006). A região analisada está destacada por um quadro vermelho. Em b), o *hit* estrutural mostrando a similaridade em ocorrências nos fungos *S. cerevisiae*, *A. niger*, *A. capsulatus* e *S. pombe*. As estruturas secundárias estão representadas por parênteses (indicando estruturas pareadas) e pontos (estruturas livres, sem ligação).

5. DISCUSSÃO

A revelação da importância dos ncRNAs nas células justifica e estimula o surgimento de novos algoritmos computacionais e métodos experimentais para caracterização dessas moléculas. Assim, a identificação dos ncRNAs em genomas e transcriptomas desponta como uma tendência nos grandes projetos, e é de se esperar que a anotação dessas seqüências pode vir a se tornar tão essencial quanto anotação dos mRNAs e seus produtos protéicos.

Apesar de já existirem programas para detecção de ncRNAs em transcriptomas, esses são claramente adaptados a seqüências de organismos modelo, organismos dos quais diversas informações de literatura, como por exemplo o genoma, ou seqüências protéicas e nucleotídicas anotadas do próprio organismo e de organismos relacionados, podem ser encontrados com abundância. Esse não é o caso para muitos organismos, principalmente negligenciados, já que muitas vezes sequencia-se parcialmente o genoma de um patógeno e há pouco conhecimento de organismos similares. Nesses casos são mais interessantes as análises *ab initio*, que não fazem uso da comparação de seqüências.

Uma outra limitação dos programas existentes é o tempo de processamento e o custo computacional exigido. Se o pesquisador quiser incluir um desses programas a seu *pipeline* de análise do transcriptoma, terá de disponibilizar máquinas muito poderosas e dedicadas a essa tarefa, tamanha a exigência desses programas. A realidade contrasta com essa exigência, pois o *pipeline* de processamento e montagem dos dados gerados por projetos de seqüenciamento de transcriptoma por si só já são muito onerosos computacionalmente, além do que muitos projetos de transcriptoma não possuem máquinas em número suficiente de forma que se possa dedicar uma máquina exclusivamente a uma etapa do processamento.

O programa descrito nesse trabalho é uma tentativa de adequação desses programas estado-da-arte à realidade dos projetos transcriptoma de baixo custo e/ou de organismos negligenciados, considerando a escassez de máquinas, a carência de dados sobre o organismo na literatura e limitações das seqüências, freqüentemente truncadas.

A estratégia é buscar na literatura programas (ou criá-los, conforme o caso) que extraem propriedades das seqüências, representando-as quantitativamente. Já em forma de "valores", as seqüências podem ser distinguidas estatisticamente. Para essa tarefa podem ser usados os programas de aprendizagem de máquina (ou aprendizagem estatística), que geram modelos que permitem diferenciar os dados do conjunto de treinamento. O modelo induzido pode ser avaliado por diversos medidores de eficiência, e se seu desempenho for considerado adequado, este pode ser usado para classificar dados desconhecidos, ausentes do conjunto de treinamento. Nesse trabalho foi utilizada uma abordagem com uso do algoritmo Máquinas de

Vetores de Suporte (MVS), por este ter um reconhecido desempenho em diversas situações, especialmente em problemas de biologia computacional.

O conjunto de treinamento foi construído a partir dos bancos de dados de proteínas e ncRNAs mais confiáveis e abrangentes disponíveis. Os dados passaram por uma eliminação de seqüências indesejáveis e repetidas/similares para evitar vício na etapa de treinamento.

Em um primeiro momento, como projeto piloto, foi usado como tradutor de ESTs o programa OrfPredictor. Um classificador MVS foi induzido usando o programa OrfPredictor, além de um programa de comparação de seqüências protéicas, outro de análise de domínios funcionais protéicos e outro de desdobramento intrínseco protéico. Esse classificador apresentou uma acurácia de 92,1% e uma AAC de 0,97.

Nas etapas seguintes de aprimoramento do programa, decidiu-se pela eliminação da comparação a bancos de dados protéicos para evitar o viés que essa análise impõe, e também do desdobramento intrínseco, que mostrou ser um atributo de baixo valor discriminatório (dados não mostrados). Além disso, o programa OrfPredictor mostrou ser inconveniente ao não disponibilizar uma versão *standalone,* além de seu viés em encontrar ORFs na maioria dos transcritos. Por isso optou-se pela adoção de um programa de tradução de transcritos mais conveniente e melhor adaptado a ESTs. Assim, um novo classificador foi induzido usando o programa ANGLE e excluindo-se esses três programas (OrfPredictor, análise comparativa e desdobramento intrínseco).

Após extensa busca na literatura o programa ANGLE foi considerado o melhor preditor de ORFs para seqüências desse tipo, e compatível com o programa desenvolvido nesse trabalho, por fazer uma abordagem que modela explicitamente todas os possíveis erros de seqüenciamento, mudanças de fase de leitura e fragmentação da seqüência. Isso garante a melhor tradução possível de um EST, o que é fundamental para uma predição confiável proteína-dependente de potencial codificador.

Apesar de existir uma aparente diferença de desempenho entre os métodos nB e MVS (Figuras 8 e Tabelas 8 e 9), deve-se ressaltar que o nB é um algoritmo de uso muito mais simples e rápido, conforme observado na Tabela 9. O que mais diferencia o desempenho global superior do MVS com relação ao nB, no entanto, é sua alta especificidade, ou seja, a exatidão em suas predições de verdadeiros negativos, ou seja, ncRNAs verdadeiros, integrantes do subconjunto de exemplos negativos do conjunto de treinamento dbTR. Pode-se inferir, portanto, que ao menos no presente trabalho a predição de mRNAs foi uma tarefa mais simples do que a predição de ncRNAs.

Outra diferença marcante entre os algoritmos é a baixa variedade de probabilidades do classificador nB (Figura 10) se comparado ao MVS (Figura 9). O classificador nB parece ter tendência a emitir predições com confiança absoluta, como +1 ou –1. Isso dificulta bastante uma análise humana posterior, já que as nuances de probabilidade representam a confiança do classificador, que o anotador pode comparar à sua própria confiança. Isso não ocorre com o MVS, que explora melhor o espectro de probabilidades de predição, conforme pode ser observado na figura 8. Esse fato, em conjunto com os resultados de eficiência, sugere o MVS como o melhor classificador a ser usado para auxiliar um pesquisador, mesmo tendo maior exigência computacional e temporal.

A rotulagem de transcritos do conjunto positivo como ncRNA (índice de falsos positivos) feita tanto pelo MVS quanto pelo nB, conforme observado na Tabela 8, era esperada não só por um erro dos classificadores induzidos, mas também por corroboração a dados levantados por outros trabalhos, que apontam a possibilidade de que algumas proteínas do banco de dados Swiss-Prot possam na verdade serem ncRNAs erroneamente anotados como mRNAs, tanto por erros experimentais como por erros dos anotadores (Frith *et al*, 2006; Liu *et al*, 2006).

Pela figura 11 pode-se perceber uma tendência de todos os programas em selecionar como ncRNA os *singlets* de Pb. Esse fato corrobora dados experimentais de Ravasi *et al* (2006), que determinam que para camundongos os ncRNAs tendem a ser *singlets* (no referido estudo, apenas 9% dos mRNAs apresentam-se como *singlets*, enquanto 48% dos ncRNAs são *singlets*), e análise *in silico* do transcriptoma de porco, onde essa tendência é novamente observada (Seemann *et al*, 2007). Essa discrepância pode ser explicada em parte pelos baixos níveis transcricionais que os ncRNAs apresentam, se comparados aos altos níveis dos mRNAs. Sendo transcritos com função primordialmente regulatória, poucas cópias de transcritos de ncRNAs seriam necessárias nas células, em contraste com mRNAs, que podem ser exigidos em grandes quantidades para que quantidades suficientes de proteína possam ser traduzidas.

Pela figura 11 nota-se que as seqüências selecionadas como ncRNAs por todos classificadores são em sua maioria aquelas sem uma anotação confiante e precisa. A maioria dos transcritos que o PORTRAIT estaria selecionando como ncRNAs seriam, portanto, transcritos sem homólogos nos bancos de dados, sendo isso uma corroboração paralela feita apenas pelo PORTRAIT, já que os programas CPC e CONC fazem uso dessa informação para sua classificação, não sendo portanto um dado independente.

A tabela 10 e anexo 1 mostram que o tamanho da ORF é uma característica muito importante na identificação de ncRNAs, confirmando dados da literatura (Frith *et al*, 2006; Liu *et al*, 2006). Dos aminoácidos classificados como melhores discriminadores, a alanina (4º. colocado) e ácido aspártico (17º. colocado) não possuem uma justificativa clara do porquê de seu poder discriminador. Já para os aminoácidos triptofano (73º. colocado) e cisteína (18º. colocado) ocorreu o contrário: eram esperados como bons discriminadores, pois são raros e geralmente ocupam posições estratégicas nas proteínas, mas para o MVS essa discriminação não foi relevante. É possível imaginar que as proteínas codificadas pelo conjunto dbNC, por exemplo, possua seqüências anômalas desses aminoácidos, incluindo repetições extensas ou combinações não ortodoxas e irreais, características que poderiam ser exploradas pelo algoritmo.

A presença massiva de dinucleotídeos e trinucleotídeos de alto poder discriminativo com a dupla "CG" em sua composição (colocações 3, 7, 8, 10) é um fato interessante, já que esses nucleotídeos formam uma ligação mais estável que "A" e "U", garantindo uma estrutura secundária definida que é característica de muitos ncRNAs. Paradoxalmente, a dupla "GC" recebeu pontuações bem mais baixas (o dinucleotídeo "GC" aparece em 28º. colocado, enquanto o dinucleotídeo "CG" aparece em 3º). O trinucleotídeo TAG, que é um códon de parada, mostra-se como um atributo importante (5º. lugar), enquanto os outros códons de parada, TAA e TGA, não recebem uma colocação tão importante (34ª. e 70ª. posições, respectivamente). Observa-se que a característica "tamanho de ORF menor que 20 letras" é o único atributo com poder discriminativo nulo, e que portanto o vetor de características, à exceção desse atributo, foi composto por atributos relevantes.

O programa PORTRAIT descrito nesse trabalho está disponível como um software livre e de código aberto para uso pela Internet (Figura 13), via *browser* (URL: http://bioinformatics.cenargen.embrapa.br/portrait), e também para instalação e execução local (versão *standalone*).

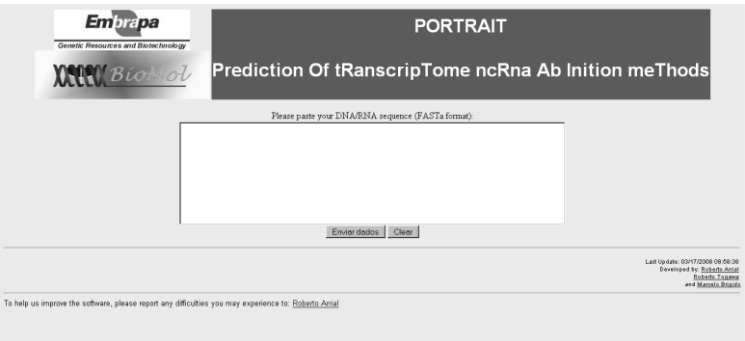


Figura 13. Página da Internet que disponibiliza a versão webserver do programa PORTRAIT.
Atualização de abril de 2008.

5.1 Comparação a trabalhos relacionados

No trabalho de Liu *et al* (2006), o desempenho do programa CONC é avaliado em comparação ao programa ESTscan (Lottaz *et al*, 2003). Essa comparação na verdade é imprecisa, já que o ESTscan precisa necessariamente ser treinado anteriormente em um conjunto de exemplos de cDNAs codificadores e não-codificadores específicos dos organismos, processo no qual é gerado um modelo organismo-específico (arquivo .SMAT) que só é eficiente para predições específicas daquele organismo. Por padrão, o ESTscan usa o .SMAT de *Homo sapiens*, que gera predições imprecisas se usado em outros organismos, que foi exatamente o que ocorreu na comparação ao CONC, já que os pesquisadores não geraram .SMAT para cada organismo e usaram apenas o padrão humano para todos (Jinfeng Liu, comunicação pessoal). É mostrado que CONC possui um desempenho muito superior ao ESTscan.

No trabalho de (Liu *et al*, 2006) foi induzido também um classificador bayesiano. Esse classificador teve um bom desempenho, sendo similar ao desempenho do MVS naquele trabalho. No presente trabalho, o desempenho do nB foi notavelmente reduzido em comparação ao MVS. O baixo desempenho do nB pode ser explicado pela redundância do conjunto de treinamento, maior e mais complexo do que o trabalho comparado, ou então porque alguns dos atributos usados podem ter alta interdependência (alta covariância) (Witten e Frank, 2005), o que viola uma premissa essencial do naive Bayes e é reconhecidamente um fator de degradação de desempenho desse algoritmo, mas não do MVS, que é mais robusto a redundâncias, conforme discutido anteriormente.

O programa CPC é mais veloz e possui maior acurácia que o CONC (Kong *et al*, 2007), ao preço de uma dependência extrema de análises comparativas a bancos de dados de proteínas, já que nenhuma outra característica é analisada (CONC utiliza 180 variáveis enquanto CPC utiliza apenas 6). Se as seqüências analisadas possuírem poucas proteínas homólogas, ou o banco possuir ausência ou excesso de determinados tipos de seqüências ou seqüências com anotação errônea, é gerada uma tendência que pode alterar completamente a decisão do classificador.

Comparado aos programas CONC e CPC, o programa PORTRAIT é notavelmente mais rápido (tabela 9). O maior tempo de execução ocorre simultaneamente com um maior uso dos recursos da máquina: observou-se, durante o processamento de ambos programas, um alto uso de RAM e de CPU do início ao fim (dados não mostrados). O mesmo não se observa com o programa PORTRAIT, que além de mais veloz, utiliza programas menos onerosos computacionalmente (dados não mostrados). Isso representa um ganho em liberação de

máquinas que pequenos projetos de transcriptomas podem ter. Quanto à acurácia no banco dbTR, o programa PORTRAIT mostrou ser superior pelo menos ao CPC (Tabela 9), mas sua acurácia em validação cruzada foi inferior (PORTRAIT: 92,4%. CPC: 95,7%; CONC: 97,0%). No entanto, deve-se notar que a comparação direta das acurácias no procedimento de validação cruzada é válida entre o CONC e CPC, mas imprecisa entre esses e o PORTRAIT, porque os conjuntos de treinamento são muito diferentes: a acurácia reportada para o CONC e CPC refere-se a uma validação cruzada realizada em um conjunto de dados com 5.610 entradas, enquanto o programa PORTRAIT foi avaliado em um conjunto com 126.039 entradas. Um conjunto de dados maior significa indução de um modelo de classificação mais generalista e menos especialista, ao preço da ocorrência de uma quantidade maior de erros na validação cruzada porque a presença de *outliers* é maior.

Outras diferenças entre os bancos de dados podem explicar a maior acurácia do CONC e CPC: o conjunto de treinamento usado por esses programas tem proteínas do SwissProt oriundas apenas de eucariotos. O presente trabalho não aplica nenhum tipo de filtro, envolvendo proteínas de potencialmente todos os Reinos. Além disso, no treinamento do PORTRAIT foi usada uma versão mais recente desse banco, e que portanto possui uma quantidade sensivelmente maior de exemplos. Um conjunto de treinamento positivo maior potencialmente implica em caracterização de uma quantidade maior e mais variada de casos positivos, ao custo de maior ocorrência de falsos positivos. O conjunto negativo, por sua vez, é composto apenas de transcritos oriundos de eucariotos do conjunto NONCODE, sendo a parte de seqüências de procariotos do NONCODE e os dois outros bancos reservados apenas para teste de acurácia. No PORTRAIT, para a montagem do conjunto negativo foram usados os três conjuntos negativos inteiros: NONCODE, RNAdb e Rfam, com filtro apenas contra redundâncias. Além disso, esse trabalho beneficia-se de uma grande atualização feita no RNAdb apenas recentemente (RNAdb v.2.0). Com isso tem-se, também, um conjunto de treinamento negativo maior, que potencialmente implica em caracterização de uma quantidade maior e mais variada de casos negativos, ao custo de maior ocorrência de falsos negativos.

Com relação a atributos, uma diferença importante entre os programas é que no PORTRAIT não foram usados atributos que usam comparação de seqüências. Por exemplo, não foi incluído como atributo a entropia de alinhamento de proteína. Segundo a discussão dos próprios autores do CONC, apesar de esse atributo ser um dos que mais contribui para a alta pontuação do algoritmo, é um potencial "viciador" do MVS em propriedades já conhecidas de proteínas, resultando em viés de classificar transcritos com muitos homólogos

próximos como proteína. Com isso, tem-se no CONC, ao menos teoricamente, maior obtenção de falsos negativos, já que um transcrito que codifique para uma proteína nova será classificado como não-codificador. Com a ausência desse atributo, o algoritmo descrito nesse trabalho apresenta um decréscimo em sensibilidade, mas um ganho na caracterização de proteínas novas, sem homólogos caracterizados. Isso privilegia a análise de transcritos oriundos de organismos negligenciados, para os quais não existem muitas seqüências de homólogos depositadas nos bancos de dados. Uma propriedade importante do programa PORTRAIT é portanto que sua análise é estritamente *ab initio*, sem buscar informação de similaridade entre a seqüência de entrada e os bancos de dados. Isso evita o uso excessivo de análise comparativa ao analisar seqüências, já que o pesquisador irá utilizar essa ferramenta também para fazer anotação de seus dados.

A tendência do CPC em classificar seqüências sem proteínas similares nos bancos de dados como ncRNA (tendência a falsos-negativos) está claramente demonstrado na Tabela 11. O conjunto dbRD é inteiramente classificado como sendo composto por ncRNAs com pontuações confiantes (dados não mostrados). Aparentemente é um resultado positivo, já que essas seqüências foram geradas de forma aleatória. No entanto, dentre essas seqüências aleatórias, é de se esperar que algumas sejam similares a mRNAs reais. Essa análise pode ser interpretada como a simulação do surgimento de uma proteína nova em um organismo, e por ser ainda não identificada, tal proteína não possui nenhum homólogo nos bancos de dados. Como o programa CPC faz toda sua análise baseada em comparação de seqüências, é natural que tal proteína não será identificada. Em constraste, o programa PORTRAIT chega a encontrar mRNAs entre os transcritos. Isso dá a impressão de que o programa CPC é um mero rebuscamento dos métodos usados por (Numata et al, 2003; MacIntosh et al, 2001), o que pode ser um retrocesso na metodologia de identificação de ncRNAs. Outra evidência desse viés pode ser observada na tabela 11. Uma grande quantidade de ncRNAs é encontrada para o conjunto dbPB, uma quantidade muito maior do que a estimada pelo programa PORTRAIT. Pode-se inferir que essa alta quantidade de ncRNAs é reflexo da quantidade reduzida de proteínas similares nos bancos de dados, ou seja, devido ao fato das seqüências de *P. brasiliensis* possuírem pouca representatividade nos bancos de proteínas, o programa CPC tem uma tendência a classificar essas proteínas como ncRNAs.

5.2 Perspectivas

-Com a publicação da seqüência do genoma estrutural de *P. brasiliensis* (disponível no site: http://www.broad.mit.edu/annotation/genome/paracoccidioides_brasiliensis/Info.html), será possível não apenas identificar o contexto genômico dos transcritos preditos como ncRNA, reforçando ou enfraquecendo o status de ncRNA, como também, permitirá o uso de *genefinders* genômicos e genômica comparativa, podendo inclusive indicar novos ncRNAs ou confirmar os citados nesse trabalho.

-A partir da finalização da etapa de comparação de estruturas secundárias de seqüências de RNA de *P. brasiliensis* e dos outros fungos do conjunto dbFG, segue-se a análise e anotação de seqüências que apresentarem alguma conservação estrutural. Com isso, pode ser possível identificar transcritos conservados entre os fungos, e que representam bons candidatos a estudos bioquímicos e futuramente, potenciais alvos de drogas.

-Os algoritmos de MVS e nB são extremamente sensíveis à fase de pré-processamento dos dados, incluindo normalização, discretização e codificação das variáveis. É possível que mantendo-se os atributos e alterando apenas esses processamentos, obtenha-se um desempenho superior ao relatado nesse trabalho.

-Integração de outros atributos, principalmente estrutura secundária de RNA, com isso espera-se uma classificação mais confiante dos ncRNAs. Suportando essa hipótese está o fato de que nesse trabalho a dupla de nucleotídeos "CG" apresentou elevado poder discriminativo (ver Discussão), do qual infere-se que a análise estrutural dos transcritos pode contribuir na identificação mais confiante dos ncRNAs. Com o aumento do poder computacional e a simplificação desses algoritmos de predição, isso pode ser possível no futuro; atualmente, a complexidade desses algoritmos não permite o uso para todas as seqüências, especialmente as muito longas.

-É possível a integração do PORTRAIT a *pipelines* de análise de transcriptoma, fornecendo subsídios para que um anotador humano tenha uma opinião a mais ao decidir sobre a anotação de um transcrito (fornecendo predições por MVS). Ressalta-se que o algoritmo não só detecta ncRNAs putativos, como também é útil para eventualmente corroborar que dado transcrito realmente codifica para um produto protéico, podendo caracterizar novos transcritos e também reconsiderar anotações de transcritos já anotados, como foi feito nesse trabalho com o transcriptoma de Pb. Como não foram excluídas seqüências de grupos filogenéticos específicos, espera-se que PORTRAIT funcione bem para transcriptomas oriundos de quaisquer organismos, tanto procariotos como eucariotos.

-Flexibilização e dinamização do processo de treinamento, com criação de um *pipeline* que facilite atualizações do próprio algoritmo a partir da expansão dos bancos de dados de proteínas e de ncRNAs, de acordo com demanda e interesse da comunidade científica.

-Nesse trabalho utilizou-se AM do tipo supervisionado. Um trabalho futuro pode ser utilizar algoritmos de AM não-supervisionado, com os quais se espera a identificação de mais de uma classe de ncRNAs (em constraste a esse trabalho, onde apenas duas classes são possíveis), agrupando e distinguindo, por exemplo, UTRs, ncRNAs médios e longos, pré-miRNAs, dentre outras classes específicas de ncRNAs.

6. REFERÊNCIAS

ABATE, A., DALMORO, F.D. e LANCKRIET, G.R.G. Response to 'Support vector machines versus artificial neural network: Who is the winner?'. **Kidney Int.** 71:83-86, 2007.

ADAMS, M., ADAMS, M.D., KELLEY, J.M., GOCAYNE, J.D., DUBNICK, M., POLYMEROPOULOS, M.H., H. XIAO, H., MERRIL, C.R., WU A., OLDE B. E MORENO, R.F. Complementary DNA sequencing: expressed sequence tags and human genome project. **Science** 252:1651–1656, 1991.

ALTSCHUL, S.F., MADDEN, T.L., SCHÄFFER, A.A., ZHANG, J., ZHANG, Z., MILLER, W. e LIPMAN, D.J. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. **Nucleic Acids Res.** 25:3389-3402, 1997.

AMARAL, A.C., FERNANDES, L., GALDINO, A.S., FELIPE, M.S.S., SOARES, C.M.A. e PEREIRA, M.A. Therapeutic targets in *Paracoccidioides brasiliensis*: post-transcriptome perspectives. **Genet. Mol. Res.** 4(2):430-449, 2005.

BACHELLERIE, J-P., CAVAILLÉ, J. e HÜTTENHOFER, A. The expanding snoRNA world. **Biochimie** 84:775–790, 2002.

BADGER, J.H. e OLSEN, G.J. CRITICA: Coding region identification tool invoking comparative analysis. **Mol. Biol. Evol.** 16(4):512–524, 1999.

BALDI, P. e BRUNAK, S. **Bioinformatics – the machine learning approach**. 2ª edição. MIT Press, England, 2001.

BIRZELE, F. e KRAMER, S. A new representation for protein secondary structure prediction based on frequent patterns. **Bioinformatics** 22(21): 2628-2634, 2006.

BORGELT, C. Full and Naive Bayes classifiers. Software disponível em [**http://www.borgelt.net/bayes.html**], 2007.

BOUAYNAYA, N. e SCHONFELD, D. The Genomic Structure: Proof of the Role of Non-Coding DNA. **EMBS 28th Ann. Intl. Conf. IEEE** 4544-4547, 2006.

BRENT, M.R. E GUIGÓ, R. Recent advances in gene structure prediction. **Curr. Opin. Struct. Biol**.14:264-272, 2004.

BRÍGIDO, M.M., WALTER, M.E.M.T., OLIVEIRA, A.G., INOUE, M.K., ANJOS, D.S., SANDES, E.F.O., GONDIM, J.J., CARVALHO, M.J.A., ALMEIDA JR., N.F. e FELIPE, M.S.S. Bioinformatics of the Paracoccidioides brasiliensis EST Project**. Genet. Mol. Res.** 4(2):203-215, 2005.

BUITING, K., NAZLICAN, H., GALETZKA, D., WAWRZIK, M., GROβ , S. e HORSTHEMKE, B. C15orf2 and a novel noncoding transcript from the Prader–Willi/Angelman syndrome region show monoallelic expression in fetal brain. **Genomics** 89(5):588-595, 2007.

BURGE, C. e KARLIN, S. Prediction of complete gene structures in human genomic DNA. **J. Mol. Biol.** 268:78–94, 1997.

BYUN, Y. e HAN, K. PseudoViewer: web application and web service for visualizing RNA pseudoknots and secondary structures. **Nucleic Acids Res**. 34:W416-W422, 2006.

CARTER, R.J., DUBCHAK, I. e HOLBROOK, S.R. A computational approach to identify genes for functional RNAs in genomic sequences. **Nucleic Acids Res**. 29:3928-38, 2001.

CASTRIGNANÒ, T., CANALI, A., GRILLO, G., LIUNI, S., MIGNONE, F., PESOLE, G. CSTminer: a web tool for the identification of coding and noncoding conserved sequence tags through cross-species genome comparison. **Nucleic Acids Res.** 32:W624–W627, 2004.

CASTRIGNANÒ, T., DE MEO, P.D., GRILLO, G., LIUNI, S., MIGNONE, F., TÁLAMO, I.G. e PESOLE, G. GenoMiner: a tool for genome-wide search of coding and non-coding conserved sequence tags. **Bioinformatics** 22(4):497–499, 2006.

CLAVERIE, J.M. Fewer genes, more noncoding RNA. **Science** 309: 1529–1530, 2005.

CHANG, C.C. e LIN, C.J. LIBSVM: a library for support vector machines. Software disponível em [**http://www.csie.ntu.edu.tw/~cjlin/libsvm**], 2006.

CHEN, M., GRANGER, A.J., VANBROCKLIN, M.W., PAYNE, W.S., HUNT, H., ZHANG, H., DODGSON, J.B. e HOLMEN, S.L. Inhibition of avian leukosis virus replication by vector-based RNA interference. **Virology** 365(2):464-472, 2007.

CHENG, J., KAPRANOV, P., DRENKOW, J., DIKE, S., BRUBAKER, S., PATEL, S., LONG, J., STERN, D., TAMMANA, H., HELT, G., SEMENTCHENKO, V., PICCOLBONI, A., BEKIRANOV, S., BAILEY, D.K., GANESH, M., GHOSH, S., BELL, I., GERHARD, D.S. e GINGERAS, T.R. Transcriptional maps of 10 human chromosomes at 5-nucleotide resolution. **Science** 308: 1149–1154, 2005.

COCHRANE, G., ALDEBERT, P., ALTHORPE, N., ANDERSSON, M., BAKER, W., BALDWIN, A., BATES, K., BHATTACHARYYA, S., BROWNE, P., VAN DEN BROEK, A., CASTRO, M., DUGGAN, K., EBERHARDT, R., FARUQUE, N., GAMBLE, J., KANZ, C., KULIKOVA, T., LEE, C., LEINONEN, R., LIN, Q., LOMBARD, V., LOPEZ, R., MCHALE, M., MCWILLIAM, H., MUKHERJEE, G., NARDONE, F., PASTOR, M.P.G., SOBHANY, S., STOEHR, P., TZOUVARA, K., VAUGHAN, R., WU, D., ZHU, W. e APWEILER, R. EMBL nucleotide sequence database: developments in 2005. **Nucleic Acids Res.** 34:D10-D15, 2006.

COSTA, F.F. Non-coding RNAs: new players in eukaryotic biology. **Gene** 357:83–94, 2005.

COSTA, F.F. Non-coding RNAs: Lost in translation?. **Gene** 386(1-2):1-10, 2007.

COVENTRY, A., KLEITMAN, D.J. e BERGER, B. MSARI: Multiple sequence alignments for statistical detection of RNA secondary structure. **Proc Natl Acad Sci USA** 101:12102–12107, 2004.

CROW, J.A. e RETZEL, E.F. Diogenes -- Reliable prediction of protein-encoding regions in short genomic sequences. [**http://analysis.ccgb.umn.edu/diogenes]**, 2005.

DENG, W., ZHU, X., SKOGERBO, G., ZHAO, Y., FU, Z., WANG, Y., HE, H., CAI, L., SUN, H., LIU, C., LI, B., BAI, B., WANG, J., JIA, D., SUN, S., HE, H., CUI, Y., WANG, Y., BU, D. e CHEN, R. (2006) Organization of the *Caenorhabditis elegans* small non-coding transcriptome: genomic features, biogenesis, and expression. **Genome Res** 16:20–29, 2006.

di BERNARDO, D., DOWN, T. e HUBBARD, T. ddbRNA: detection of conserved secondary structures in multiple alignments. **Bioinformatics** 19:1606-1611, 2003.

EDDY, S.R. Non-coding RNA genes and the modern RNA world. **Nat. Rev**. 2:919-929, 2001.

EDDY, S.R. Computational genomics of noncoding RNA genes. **Cell** 109:137–140, 2002.

EDDY, S.R. What is Bayesian statistics? **Nat. Biotech.** 22(9):1177-1178, 2004a.

ENRIGHT A.J., VAN DONGEN, S. e OUZOUNIS, C.A. An efficient algorithm for large-scale detection of protein families. **Nucleic Acids Res.** 30:1575–1584, 2002.

ESPINOZA, C.A., GOODRICH, J.A. e KUGEL, J.F. Characterization of the structure, function, and mechanism of B2 RNA, an ncRNA repressor of RNA polymerase II transcription. **RNA** 13(4):583-96, 2007.

FAWCETT, T. ROC graphs: Notes and practical considerations for researchers. **Tech Report HPL-2003-4**, HP Laboratories, 2004.

FELIPE, M.S., ANDRADE, R.V., ARRAES, F.B.M., NICOLA, A.M., MARANHÃO, A.Q., TORRES, F.A.G., SILVA-PEREIRA, I., POÇAS-FONSECA, M.J., CAMPOS, E.G., MORAES, L.M.P., ANDRADE, P.A., TAVARES, A.H.F.P., SILVA, S.S., KYAW, C.M., SOUZA, D.P., PBGENOME NETWORK, PEREIRA, M., JESUÍNO, R.S.A., ANDRADE, E.V., PARENTE, J.A., OLIVEIRA, G.S., BARBOSA, M.S., MARTINS, N.F., FACHIN, A.L., CARDOSO, R.S., PASSOS, G.A.S., ALMEIDA, N.F., WALTER, M.E.M.T., SOARES, C.M.A., CARVALHO, M.J.A. e BRÍGIDO, M.M. Transcriptional profiles of the human pathogenic fungus *Paracoccidioides brasiliensis* in mycelium and yeast cells. **J. Biol. Chem.** 280:24706-24714, 2005.

FELIPE, M.S., ANDRADE, R.V., PETROFEZA, S.S., MARANHAO, A.Q., TORRES, F.A., ALBUQUERQUE, P., ARRAES, F.B., ARRUDA, M., AZEVEDO, M.O., BAPTISTA, A.J., BATAUS, L.A., BORGES, C.L., CAMPOS, E.G., CRUZ, M.R., DAHER, B.S., DANTAS, A., FERREIRA, M.A., GHIL, G.V., JESUINO, R.S., KYAW, C.M., LEITAO, L., MARTINS, C.R., MORAES, L.M., NEVES, E.O., NICOLA, A.M., ALVES, E.S., PARENTE, J.A., PEREIRA, M., POCAS-FONSECA, M.J., RESENDE, R., RIBEIRO, B.M., SALDANHA, R.R., SANTOS, S.C., SILVA-PEREIRA, I., SILVA, M.A., SILVEIRA, E., SIMOES, I.C., SOARES, R.B., SOUZA, D.P., DE-SOUZA, M.T., ANDRADE, E.V., XAVIER, M.A., VEIGA, H.P., VENANCIO, E.J., CARVALHO, M.J., OLIVEIRA, A.G., INOUE, M.K., ALMEIDA, N.F., WALTER, M.E., SOARES, C.M. e BRÍGIDO, M.M. Transcriptome characterization of the

dimorphic and pathogenic fungus *Paracoccidioides brasiliensis* by EST analysis. **Yeast** 20: 263-271, 2003.

FICKETT, J.W. e TUNG, C.-S. Assessment of protein coding measures. **Nucleic Acids Res**. 20(24):6441-6450, 1992.

FLOREA, L., HARTZELL, G., ZHANG, Z., RUBIN, G.M., e MILLER, W. A computer program for aligning a cDNA sequence with a genomic DNA sequence. **Genome Res.** 8: 967-974, 1998.

FRANCO, M. Host–parasite relationships in paracoccidioidomycosis. **J. Med. Vet. Mycol**. 25: 5–18, 1987.

FREYHULT, E.K., BOLLBACK, J.P. e GARDNER, P.P. Exploring genomic dark matter: A critical assessment of the performance of homology search methods on noncoding RNA. **Genome Res.** 17:117-125, 2007.

FRITH, M.C., BAILEY, T.L., KASUKAWA, T., MIGNONE, F., KUMMERFELD, S.K., MADERA, M., SUNKARA, S., FURUNO, M., BULT, C.J., QUACKENBUSH, J., KAI, C., KAWAI, J., CARNINCI, P., HAYASHIZAKI, Y., PESOLE, G. e MATTICK, J.S. Discrimination of non-protein-coding transcripts from protein-coding mRNA. RNA Biol. 3(1):40-48, 2006.

GINGER, M.R., SHORE, A.N., CONTRERAS, A., RIJNKELS, M., MILLER, J., GONZALEZ-RIMBAU, M.F. e ROSEN, J.M. A noncoding RNA is a potential marker of cell fate during mammary gland development. **Proc Natl Acad Sci USA** 103(15): 5781–5786, 2006.

GRIFFITHS-JONES, S., GROCOCK, R.J., VAN DONGEN, S., BATEMAN, A. e ENRIGHT, A.J. miRBase: microRNA sequences, targets and gene nomenclature. **Nucleic Acids Res.** 32:140-144, 2006.

GRIFFITHS-JONES, S., MOXON, S., MARSHALL, M., KHANNA, A., EDDY, S.R., e BATEMAN, A. Rfam: annotating non-coding RNAs in complete genomes. **Nucleic Acids Res.** 33:D121–D124, 2005.

GRILLO, G., ATTIMONELLI, M., LIUNI, S. e PESOLE, G. CLEANUP: a fast computer program for removing redundancies from nucleotide sequence databases. **Comput. Appl. Biosci.** 12:1–8, 1996.

GUIGÓ, A,R., AGARWAL,P., ABRIL,J., BURSET,M. e FICKETT,J. An assessment of gene prediction accuracy in large DNA sequences. **Genome Res.** 10:1631-1642, 2000.

GUIGÓ, A.R. e BRENT, M.R. Recent advances in gene structure prediction. **Curr. Op. Struct. Biol.** 14:264-272, 2004.

HARTE, N., SILVENTOINEN, V., QUEVILLON, E., ROBINSON, S., KALLIO, K., FUSTERO, X., PATEL, P., JOKINEN, P. e LOPEZ, P. Public web-based services from the European Bioinformatics Institute. **Nucleic Acids Res.** 32:W3–W9, 2004.

HATZIGEORGIOU A.G., FIZIEV, P. e RECZKO, M. DIANA-EST: A statistical analysis. **Bioinformatics** 17(10):913-919, 2001

HAYASHIZAKI, Y. e KANAMORI, M. Dynamic transcriptome of mice. **Trends Biotechnol**. 22:161–167, 2004.

HE, S., LIU, C., SKOGERBØ, G., ZHAO, Y., WANG, J., LIU, T., BAI, B., ZHAO, Y., e CHEN, R. NONCODE v2.0: decoding the non-coding. Nucleic Acids Res. 36:D170-D172, 2008.

HENIKOFF, S. e HENIKOFF, J.G. Amino acid substitution matrices from protein blocks. **Proc. Natl. Acad. Sci. USA** 89:10915–10919, 1992.

HERSHBERG, R., ALTUVIA, S. e MARGALIT, H. A survey of small RNA-encoding genes in *Escherichia coli***. Nucleic Acids Res.**, 31(7):1813-1820, 2003.

HOAGLAND, M. B., STEPHENSON, M. L., SCOTT, J. F., HECHT, L. I. e ZAMECNIK, P. C. A soluble ribonucleic acid intermediate in protein synthesis. **J. Biol. Chem.** 231, 241–257, 1958.

HOFACKER, I.L. Vienna RNA secondary structure server. **Nucleic Acids Res.** 31(13):3429–3431, 2003.

HULO, N., BAIROCH, A., BULLIARD, V., CERUTTI, L., DE CASTRO, E., LANGENDIJK-GENEVAUX, P.S., PAGNI, M. e SIGRIST, C.J.A. The PROSITE database. **Nucleic Acids Res.**, 34:D227-D230, 2006.

HÜTTENHOFER, A., KIEFMANN, M., MEIER-EWERT, S., O'BRIEN, J., LEHRACH, H., BACHELLERIE, J.-P., e BROSIUS, J. RNomics: an experimental approach that identifies 201 candidates for novel, small, non-messenger RNAs in mouse **EMBO J.** 20(11): 2943–2953, 2001.

HÜTTENHOFER, A. e VOGEL, J. Experimental approaches to identify non-coding RNAs. **Nucleic Acids Res.** 34(2): 635–646, 2006.

HÜTTENHOFER, A., SCHATTNER, P. E POLACEK, N. Non-coding RNAs: hope or hype? **Trends Genet.** 21: 289–297, 2005.

INAGAKI, S., NUMATA, K., KONDO, T., TOMITA, M., YASUDA, K., KANAI, A. e KAGEYAMA, Y. Identification and expression analysis of putative mRNA-like non-coding RNA in *Drosophila*. **Genes Cells** 10:1163–1173, 2005.

JAMET, E. Bioinformatics as a critical prerequisite to transcriptome and proteome studies. **J. of Exp. Bot.** 55:1977–1979, 2004.

JOACHIMS, T. Making large-Scale SVM Learning Practical. In: **Advances in Kernel Methods - Support Vector Learning**, Schölkopf, B., Burges, C. e Smola, A. (eds.), MIT Press, 1999.

KEDERSHA, N.L., e ROME, L.H.. Isolation and characterization of a novel ribonucleoprotein particle: large structures contain a single species of small RNA. **J. Cell Biol.** 103:699-709, 1986.

KHALADKAR, M., BELLOFATTO, V., WANG, J.T.L., TIAN, B. e SHAPIRO, B.A. RADAR: a web server for RNA data analysis and research. **Nucleic Acids Res.** 35:W300-W304, 2007.

KHALIL, A.M., FAGHIHI, M.A., MODARESSI, F., BROTHERS, S.P. e WAHLESTEDT, C. A Novel RNA Transcript with Antiapoptotic Function Is Silenced in Fragile X Syndrome. **PloS ONE** 3(1):e1486, 2008.

KIN, T., YAMADA, K., TERAI, G., OKIDA, H., YOSHINARI, Y., ONO, Y., KOJIMA, A., KIMURA, Y., KOMORI, T. e ASAI, K. fRNAdb: a platform for mining/annotating functional RNA candidates from non-coding RNA sequences. **Nucleic Acids Res.** 35:D145-D148, 2007.

KLEIN, R.J. e EDDY, S.R. RSEARCH: Finding homologs of single structured RNA sequences. **BMC Bioinformatics** 4:44-60, 2003.

KONG, L., ZHANG, Y., YE, Z.-Q., LIU, X.-O., ZHAO, S.-O., WEI, L. e GAO, G. CPC: assess the protein-coding potential of transcripts using sequence features and support vector machine. **Nucleic Acids Res**. 35:W345-W349, 2007.

KOZLOWSKI, L (2007). **Calculation of protein isoelectric point**. Disponível em: < http://isoelectric.ovh.org/>. Acesso em: 20 de janeiro de 2008.

KRAUSE, A., STOYE, J., e VINGRON, M. The SYSTERS protein sequence cluster set. Nucleic Acids Res. 28(1):270-272, 2000.

KREIL, D.P. e OUZOUNIS, C.A. Comparison of sequence masking algorithms and the detection of biased protein sequence regions. Bioinformatics 19(13):1672–1681, 2003.

KUZNETSOV, I.B. e HWANG, S. A novel sensitive method for the detection of user-defined compositional bias in biological sequences. Bioinformatics 22(9):1055–1063, 2006.

KYTE, J. e DOOLITTLE, R.F. A Simple Method for Displaying the Hydropathic Character of a Protein. J. Mol. Biol. 157:105-132, 1982.

LARRAÑAGA, P., CALVO, B., SANTANA, R., BIELZA, C., GALDIANO, J., INZA, I., LOZANO, J.A., ARMAÑANZAS, R., SANTAFÉ, G., PÉREZ, A., e ROBLES, V. Machine learning in bioinformatics. Briefings in Bioinformatics 7(1):86-112, 2006.

LAURENT III, G.S. e WAHLESTEDT, C. Noncoding RNAs: couplers of analog and digital information in nervous system function? **Trends Neurosci.** 30(12):612-621, 2007.

LESK, A. Introduction to bioinformatics. Oxford University Press, Oxford, United States, 2002.

LESTRADE, L. e WEBER, M.J. snoRNA-LBME-db, a comprehensive database of human H/ACA and C/D box snoRNAs. Nucleic Acids Res. 34:D158–D162, 2006.

LI, W. e GODZIK, A. CD-HIT: a fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics 22(13):1658–1659, 2006.

LING, C.X., HUANG, J. e ZHANG, H. AUC: a statistically consistent and more discriminating measure than accuracy. Proceedings of IJCAI 2003 519-526, Morgan Kaufmann, 2003.

LIU, J., GOUGH, J. e ROST, B. Distinguishing protein-coding from non-coding RNAs through support vector machines. PLoS Genet. 2:e29-e36, 2006.

LORENA, A.C. **Investigação de estratégias para geração de máquinas de vetores de suporte multiclasses**. Tese de Doutorado, Instituto de Ciências da Computação e Matemática Computacional, Universidade de São Paulo, São Carlos, 2006.

LOTTAZ, C., ISELI, C., JONGENEEL, C.V. e BUCHER, P. Modeling sequencing errors by combining Hidden Markov models. **Bioinformatics** 19(2):103-112, 2003.

LUKASHIN, A.V. e BORODOVSKY, M. GeneMark.hmm: new solutions for gene finding. **Nucleic Acids Res.** 26:1107–1115, 1998.

MACINTOSH, G.C., WILKERSON, C., GREEN, P.J. Identification and analysis of Arabidopsis expressed sequence tags characteristic of non-coding RNAs. **Plant. Physiol.** 127: 765–776, 2001.

MAEDA, N., KASUKAWA, T., OYAMA, R., GOUGH, J., FRITH, M., ENGSTRÖM, P.G., LENHARD, B., ATURALIYA, R.N., BATALOV, S., BEISEL, K.W., BULT, C.J., FLETCHER, C.F., FORREST, A.R.R., FURUNO, M., HILL, D., ITOH, M., KANAMORI-KATAYAMA, M., KATAYAMA, S., KATOH, M., KAWASHIMA, T., QUACKENBUSH, J., RAVASI, T., RING, B.Z., SHIBATA, K., SUGIURA, K., TAKENAKA, Y., TEASDALE, R.D., WELLS, C.A., ZHU, Y., KAI, C., KAWAI, J., HUME, D.A., CARNINCI, P., e HAYASHIZAKI, Y. Transcript annotation in FANTOM3: Mouse gene catalog based on physical cDNAs. **PLoS Genet.** 2:e62, 2006.

MACHADO-LIMA, A., DEL PORTILLO, H.A. e DURHAM, A.M. Computational methods in noncoding RNA research. **J. Math. Biol.** 56(1-2):15-52, 2007.

MATHÉ, C., SAGOT, M.-F., SCHIEX, T., e ROUZE, P. Current methods of gene prediction, their strengths and weaknesses. **Nucleic Acids Res.** 30:4103–4117, 2002.

MATTICK, J.S. e MAKUNIN, I.V. Non-coding RNA. **Hum. Mol. Genet.** 15(1): R17–R29, 2006.

MATTICK, J.S. e GAGEN, M.J. The evolution of controlled multitasked gene networks: the role of introns and other noncoding RNAs in the development of complex organisms. **Mol. Biol. Evol.** 18(9): 1611-1630, 2001.

MATTICK, J.S. RNA regulation: a new genetics? **Nat. Rev. Genet.** 5:316–323, 2004.

McCUTCHEON, J.P. e EDDY, S.R. Computational identification of non-coding RNAs in *Saccharomyces cerevisiae* by comparative genomics. **Nucleic Acids Res.** 31(14):4119-4128, 2003.

McGINNIS, S. E MADDEN, T.L. BLAST: At the core of a powerful and diverse set of sequence analysis tools. **Nucleic Acids Res.** 32:W20–W25, 2004.

MERCER, T.R., DINGER, M.E., SUNKIN, S.M., MEHLER, M.F. e MATTICK, J.S. Specific expression of long noncoding RNAs in the mouse brain. **Proc. Natl. Acad. Sci. USA** 105(2):716-721, 2008.

MEYNERT, A. e BIRNEY, E. Picking pyknons out of the human genome. **Cell** 125:836-838, 2006.

MICHALAK, P. RNA world – the dark matter of evolutionary genomics. **J. Evol. Biol.** 19:1768-1774, 2006.

MIKA, S. e ROST, B. UniqueProt: creating representative protein sequence sets. **Nucleic Acids Res**. 31(13):3789–3791, 2003.

MIN, X.J., BUTLER, G., STORMS, R. e TSANG, A. OrfPredictor: predicting protein-coding regions in EST-derived sequences. **Nucleic Acids Research** 33:W677–W680, 2005.

MITCHELL, T.M. **Machine Learning**. McGraw-Hill, 1997.

MOCKLER, T.C., CHAN, S., SUNDARESAN, A., CHEN, H., JACOBSEN, S.E. e ECKER, J.R. Applications of DNA tiling arrays for whole-genome analysis. Genomics 85:1–15, 2005.

MOUNT, D.W. Bioinformatics: Sequence and genome analysis. 2 ed. Cold Spring Harbor Laboratory Press: EUA, 2004.

NADERSHAHI, A., FAHRENKRUG, S.C. e ELLIS, L.B.M. Comparison of computational methods for identifying translation initiation sites in EST data. BMC Bioinformatics 5:14, 2004.

NAKAYA, H.I., AMARAL, P.P., LOURO, R., LOPES, A., FACHEL, A.A., MOREIRA, Y.B., EL-JUNDI, T.A., SILVA, A.M., REIS, E.M. e VERJOVSKI-ALMEIDA, S. Genome mapping and expression analyses of human intronic noncoding RNAs reveal tissue-specific patterns and enrichment in genes related to regulation of transcription. Genome Biol., 8(3):R43, 2007.

NAM, J.-W., KIM, J., KIM, S.-K. e ZHANG, B.-T. ProMiR II: a web server for the probabilistic prediction of clustered, nonclustered, conserved and nonconserved microRNAs. **Nucleic Acids Res.** 34:W455–W458, 2006.

NELSON, D.L. e COX, M.M. **Lehninger Principles of Biochemistry.** 4ª edição. W. H. Freeman, 2004.

NOBLE, W.S. What is a support vector machine? **Nat. Biotech.** 24(12):1565-1567, 2006.

NUMATA, K., KANAI, A., SAITO, R., KONDO, S., ADACHI, J., WILMING, L.G., HUME, D.A., HAYASHIZAKI, Y., TOMITA, M., RIKEN GER Group, membros do GSL. Identification of putative noncoding RNAs among the RIKEN mouse full-length cDNA collection. **Genome Res**. 3:1301-1306, 2003.

OLIVAS, W. M., MUHLRAD, D. e PARKER, R. Analysis of the yeast genome: identification of new non-coding and small ORF-containing RNAs. **Nucleic Acids Res.** 25:4619–4625, 1997.

PANDIT, S.B., BHADRA, R., GOWRI, V.S., BALAJI, S., ANAND, B. e SRINIVASAN, N. SUPFAM: A database of sequence superfamilies of protein domains. **BMC Bioinformatics** 5:28, 2004.

PANG, K.C., FRITH, M.C. e MATTICK, J.S. Rapid evolution of noncoding RNAs: lack of conservation does not mean lack of function. **Trends in Genetics** 22:1-5. 2006.

PANG, K.C., STEPHEN, S., ENGSTRÖM, P.G., TAJUL-ARIFIN, K., CHEN, W., WAHLESTEDT, C., LENHARD, B., HAYASHIZAKI, Y. e MATTICK, J.S. RNAdb—a comprehensive mammalian noncoding RNA database. **Nucleic Acids Res.** 33:D125–D130, 2005.

PAVESI, G., MAURI, G., STEFANI, M. e PESOLE, G. RNAProfile: an algorithm for finding conserved secondary structure motifs in unaligned RNA sequences. **Nucleic Acids Res.** 32(10):3258–3269, 2004.

PEARSON, W.R. Rapid and sensitive sequence comparison with FASTP and FASTA. **Methods Enzymol.** 183:63-98, 1990.

POLLASTRI,G., BALDI, P., FARISELLI, P. e CASADIO, R. Prediction of Coordination Number and Relative Solvent Accessibility in Proteins. **PROTEINS: Struct., Funct., and Genet.** 47:142–153, 2002a.

PRESUTTI, C., ROSATI, J., VINCENTI, S. e NASI, S. Non coding RNA and brain. **BMC Neurosci.** 7(1):S5-S17, 2006.

PROMPONAS, V.J., ENRIGHT, A.J., TSOKA, S., KREIL, D.P., LEROY, C., HAMODRAKAS, S., SANDER, S. e OUZOUNIS, C. CAST: an iterative algorithm for the complexity analysis of sequence tracts. **Bioinformatics** 16(10):915–922, 2000.

RAVASI, T., SUZUKI, H., PANG, K.C., KATAYAMA, S., FURUNO, M., OKUNISHI, R., FUKUDA, S., RU1, K., FRITH, M.C., GONGORA, M.M., GRIMMOND, S.M., HUME, D.A., HAYASHIZAKI, Y. e MATTICK, J.S. **Experimental validation of the regulated expression of large numbers of non-coding RNAs from the mouse genome.** Genome Res. **16:11-19, 2006.**

REYNOLDS, A., LEAKE, D., BOESE, Q., SCARINGE, S., MARSHALL, W.S., e KHVOROVA, A. Rational siRNA design for RNA interference. **Nat. Biotech.** 22:326–330, 2004.

RICE, P., LONGDEN, I. e BLEASBY, A. EMBOSS: The European molecular biology open software suite. **Trends Genet.** 16:276-277, 2000.

RIGOUTSOS, I., HUYNH, T., MIRANDA, K., TSIRIGOS, A., McHARDY, A., e PLATT, D. Short blocks from the noncoding parts of the human genome have instances within

nearly all known genes and relate to biological processes. **Proc. Natl. Acad. Sci. USA** 17:6605-6610, 2006.

RIVAS, E. e EDDY, S.R. Noncoding RNA gene detection using comparative sequence analysis. **BMC Bioinformatics** 2:8, 2001.

RIVAS, E., KLEIN, R.J., JONES, T.A. e EDDY, S. R. Computational identification of noncoding RNAs in *E. coli* by comparative genomics. **Curr. Biol**. 11:1369-1373, 2001.

RODRIGUEZ-TOMÉ, P., STOEHR, P.J., CAMERON, G.N. e FLORES, T.P. The European Bioinformatics Institute (EBI) databases. **Nucleic Acids Research** 24(1):D6-D12, 1996

ROMERO, E. e TOPPO, D. Comparing Support Vector Machines and Feed-forward Neural Networks with Similar Hidden-layer Weights. **IEEE Transactions on Neural Networks**, 18 (3), 959-963. 2007.

SAN-BLAS, G. e NINO-VEGA, G. *Paracoccidioides brasiliensis*: virulence and host response. In **Fungal Pathogenesis: principles and clinical applications**. Marcel Dekker, New York, 2001.

SCHATTNER, P., BROOKS, A.N. e LOWE, T.M. The tRNAscan-SE, snoscan and snoGPS web servers for the detection of tRNAs and snoRNAs. **Nucleic Acids Res.** 33:W686–W689, 2005.

SCHÖLKOPF, B. e SMOLA, A. J. **Learning with Kernels**: Support Vector Machines, Regularization, Optimization, and Beyond. The MIT Press, London – England, 2002.

SEEMANN, S.E., GILCHRIST, M.J., HOFACKER, I.L., STADLER, P.F. e GORODKIN, J. Detection of RNA structures in porcine EST data and related mammals. **BMC Genomics** 8:316, 2007

SHABALINA, S. A. e SPIRIDONOV, N. A. The mammalian transcriptome and the function of non-coding DNA sequences. **Genome Biol**. 5:105-113, 2004.

SHARP, P. A. e BURGE, C. B. Classification of introns: U2- type or U12-type. Cell 91, 875–879, 1997.

SHIKANAI-YASUDA, M.A., FILHO, F.Q.T., MENDES, R.P., COLOMBO, A.L., MORETTI, M.L. e Grupo de Consultores do Consenso em Paracoccidioidomicose. Consenso em paracoccidioidomicose. **Rev. Soc. Bras. Med. Trop.** 39(3):297-310, 2006.

SHIMIZU, K., ADACHI, J. e MURAOKA, Y. ANGLE: a sequencing errors resistant program for predicting protein coding regions in unfinished cDNA. **J. Bioinfo. Comp. Biol.** 4(3):649-664, 2006.

SHIN, S.W. e KIM, S.M. A new algorithm for detecting low-complexity regions in protein sequences. **Bioinformatics** 21(2):160–170, 2005.

SOUTO, M.C.P., LORENA, A.C., DELBEM, A.C.B. e CARVALHO, A.C.P.L.F. Técnicas de Aprendizado de Máquina para Problemas de Biologia Molecular. In: **III Jornada de Mini-Curso de Inteligência Artificial** 103-152, Editora SBC, 2003.

SOUZA, S.J., CAMARGO, A.A., BRIONES, M.R., COSTA, F.F., NAGAI, M.A., VERJOVSKI-ALMEIDA, S., ZAGO, M.A., ANDRADE, L.E., CARRER, H., EL-DORRY, H.F. *et al*. Identification of human chromosome 22 transcribed sequences with ORF expressed sequence tags. **Proc. Natl. Acad. Sci.** 97:12690-12693, 2000.

SZYMANSKI, M., ERDMANN, V.A. e BARCISZEWSKI, J. Noncoding RNAs database (ncRNAdb). **Nucleic Acids Res.** 35:D162–D164, 2007.

TERAMOTO, R., AOKI, M., KIMURA, T. e KANAOKA, M. Prediction of siRNA functionality using generalized string kernel and support vector machine. **FEBS Lett**. 579(13):2878-2882, 2005.

TORARINSSON, E., SAWERA, M., HAVGAARD, J.H., FREDHOLM, M. E GORODKIN, J. Thousands of corresponding human and mouse genomic regions unalignable in primary sequence contain common RNA structure. **Genome Res.** 16**:**885–889, 2006.

UHLER, J.P., HERTEL, C. e SVEJSTRUP, J.Q. A role for noncoding transcription in activation of the yeast PHO5 gene. **Proc Natl Acad Sci USA.** 104(19):8011-8016, 2007.

VALGARDSDOTTIR, R., CHIODI, I., GIORDANO, M., ROSSI, A., BAZZINI, S., GHIGNA, C., RIVA, S., e BIAMONTI, G. Transcription of Satellite III non-coding RNAs is a general stress response in human cells. **Nucleic Acids Res.** 36(2):423-434, 2008.

VAPNIK, V.N. **Statistical learning theory**. John Wiley and Sons, 1998.

WAHLESTEDT, C. Natural antisense and noncoding RNA transcripts as potential drug targets. **Drug Disc. Today** 11(11/12):503-508, 2006.

WANG, C., DING, C., MERAZ, R.F. e HOLBROOK, S.R. PSoL: A Positive Sample only Learning algorithm for finding non-coding RNA genes. **Bioinformatics** 22(21):2590-2596, 2006.

WHEELER, D.L., CHURCH, D.M., EDGAR, R., FEDERHEN, S., HELMBERG, W., MADDEN, T.L., PONTIUS, J.U., SCHULER, G.D., SCHRIM, L.M., SEQUEIRA, E., SUZEK, T.O., TATUSOVA, T.A. e WAGNER, L. Database resources of the National Center for Biotechnology Information: update. **Nucleic Acids Res.** 32:D35-D40, 2004.

WILLIAMSON, B. DNA insertions and gene structure. **Nature** 270:295–297, 1977.

WITTEN, I.A. e FRANK, E. **Data Mining:** Practical machine learning tools and techniques. 2. ed. Elsevier: EUA, 2005.

WORKMAN, C. e KROGH, A. No evidence that mRNAs have lower free folding energies than random sequences with the same dinucleotide distribution. **Nucleic Acids Res.** 27(24):4816-4822, 1999.

WOOTTON J.C. e FEDERHEN, S. Analysis of compositionally biased regions in sequence databases. **Methods Enzymol**. 266: 554–571, 1996.

WU, C.H., APWEILER, R., BAIROCH, A., NATALE, D.A., BARKER, W.C., BOECKMANN, B., FERRO, S., GASTEIGER, E., HUANG, H., LOPEZ, R., MAGRANE, M., MARTIN,MAZUMDER, R., O'DONOVAN, C., REDASCHI, N. e SUZEK, B. The Universal Protein Resource (UniProt): an expanding universe of protein information. **Nucleic Acids Res.** 34:D187–D191, 2006a.

WU, T., WANG, J., LIU, C., ZHANG, Y., SHI, B., ZHU, X., ZHANG, Z., SKOGERBØ, G., CHEN, L., LU, H., ZHAO, Y. e CHEN, R. NPInter: the noncoding RNAs and protein related biomacromolecules interaction database. **Nucleic Acids Res.** 34:D150–D152, 2006b.

ZHANG, S., HASS, B., ESKIN, E. e BAFNA, V. Searching genomes for non-coding RNA using FastR. **IEEE/ACM Trans. on Comput. Biol. and Bioinfo.** 2(4):366–379, 2005.

XUE, C., LI, F., HE,T., LIU, G.-P., LI, Y. e ZHANG, X. Classification of real and pseudo microRNA precursors using local structure-sequence features and support vector machine. **BMC Bioinformatics** 6:310-317, 2005.

ZUKER, M. e STIEGLER, P. Optimal computer folding of larger RNA sequences using thermodynamics and auxiliary information. **Nucleic Acids Res.** 9:133–148, 1981.

7. ANEXOS

7.1 Anexo 1 – Pontuação e colocação relativa de cada variável alocada para cada um dos atributos do vetor de características.

Classificação/Atributo	Pontuação	Classificação/Atributo	Pontuação
1.ORF maior que 100	0.583153	57.trinucleotídeo GGC	0.011018
2.ORF entre 60 e 100	0.514314	58.aminoácido L	0.010582
3.dinucleotídeo CG	0.173249	59.aminoácido I	0.010482
4.aminoácido A	0.135667	60.nucleotídeo T	0.009894
5.trinucleotídeo TAG	0.124019	61.trinucleotídeo CAG	0.009639
6.trinucleotídeo TGT	0.111791	62.trinucleotídeo TTA	0.009610
7.trinucleotídeo ACG	0.109074	63.trinucleotídeo GGG	0.009567
8.trinucleotídeo TCG	0.106840	64.aminoácido K	0.009339
9.dinucleotídeo CT	0.103361	65.trinucleotídeo CAT	0.008997
10.trinucleotídeo CGA	0.098901	66.aminoácido V	0.008979
11.trinucleotídeo CGT	0.096770	67.trinucleotídeo ACC	0.008032
12.trinucleotídeo CGC	0.077155	68.aminoácido P	0.007738
13.trinucleotídeo GCG	0.076935	69.trinucleotídeo GTA	0.007563
14.trinucleotídeo TCT	0.075718	70.trinucleotídeo TGA	0.007417
15.trinucleotídeo CCG	0.070610	71.trinucleotídeo TCC	0.006809
16.trinucleotídeo CGG	0.063574	72.aminoácido Q	0.006789
17.aminoácido D	0.062826	73.aminoácido W	0.005522
18.nucleotídeo C	0.052079	74.dinucleotídeo AA	0.005294
19.trinucleotídeo CTT	0.044940	75.trinucleotídeo TAC	0.004931
20.dinucleotídeo AG	0.042967	76.Hidropatia	0.004866
21.aminoácido S	0.042579	77.dinucleotídeo CA	0.004715
22.trinucleotídeo GAT	0.041596	78.trinucleotídeo CTG	0.004397
23.trinucleotídeo ATC	0.037703	79.trinucleotídeo AAG	0.004110
24.Ponto isoeletrico	0.037553	80.trinucleotídeo TTG	0.003852
25.aminoácido E	0.037118	81.trinucleotídeo AAA	0.003598
26.trinucleotídeo TTT	0.034880	82.trinucleotídeo GCA	0.003578
27.trinucleotídeo AGT	0.034387	83.dinucleotídeo AC	0.003546
28.dinucleotídeo GC	0.032545	84.trinucleotídeo ATA	0.003476
29.aminoácido C	0.031548	85.trinucleotídeo AAT	0.003415
30.dinucleotídeo GA	0.030203	86.trinucleotídeo ATG	0.003215
31.dinucleotídeo TT	0.029373	87.trinucleotídeo TGG	0.002419
32.dinucleotídeo GT	0.029247	88.dinucleotídeo TC	0.002375
33.aminoácido N	0.026766	89.dinucleotídeo AT	0.002231
34.trinucleotídeo TAA	0.026575	90.trinucleotídeo GTG	0.001972
35.trinucleotídeo ACA	0.026424	91.trinucleotídeo CCC	0.001959
36.trinucleotídeo AGG	0.025849	92.trinucleotídeo CCA	0.001882
37.trinucleotídeo CCT	0.024429	93.trinucleotídeo TCA	0.001809
38.dinucleotídeo TG	0.024130	94.trinucleotídeo GTT	0.001391
39.trinucleotídeo CTA	0.020725	95.trinucleotídeo GTC	0.001242
40.Entropia composicional	0.020579	96.aminoácido M	0.001207

41.aminoácido F	0.019343	97.trinucleotídeo TTC	0.000974
42.aminoácido T	0.019058	98.trinucleotídeo ATT	0.000716
43.dinucleotídeo TA	0.018631	99.trinucleotídeo TAT	0.000677
44.trinucleotídeo GAC	0.018080	100.trinucleotídeo GGA	0.000599
45.aminoácido R	0.016802	101.dinucleotídeo GG	0.000518
46.trinucleotídeo CTC	0.016780	102.dinucleotídeo CC	0.000445
47.trinucleotídeo AGA	0.016613	103.aminoácido G	0.000430
48.ORF entre 20 e 60	0.016466	104.trinucleotídeo GGT	0.000422
49.trinucleotídeo CAA	0.015906	105.trinucleotídeo AGC	0.000351
50.nucleotídeo G	0.015625	106.nucleotídeo A	0.000310
51.trinucleotídeo GAA	0.014312	107.aminoácido Y	0.000234
52.trinucleotídeo ACT	0.014245	108.trinucleotídeo GAG	0.000233
53.aminoácido H	0.013947	109.trinucleotídeo TGC	0.000097
54.trinucleotídeo GCC	0.013923	110.trinucleotídeo GCT	0.000033
55.trinucleotídeo CAC	0.011848	111.ORF menor que 20	0.000000
56.trinucleotídeo AAC	0.011498		

7.2 Anexo 2. Relação dos títulos e respectivas probabilidades das 970 seqüências do transcriptoma de *Paracoccidioides brasiliensis* classificadas como ncRNA pelo programa PORTRAIT.

As seqüências não estão organizadas na mesma ordem em que aparecem no arquivo original com as 6.022 seqüências. Legenda: <Título da sequencia> ::: <Probabilidade de ser codificador> <Probabilidade de ser não-codificador>

```
gi|50017 Contig17 eukaryotic translation initiation factor  ::: 0.211271 0.788729
gi|50024 Contig24 ::: 0.249304 0.750696
gi|50035 Contig35 hypothetical protein; extensin-like; with SH3 Src homology domain  :::
0.494811 0.505189
gi|50041 Contig41 ::: 0.490891 0.509109
gi|50091 Contig91 ::: 0.353285 0.646715
gi|50109 Contig109 ::: 0.386923 0.613077
gi|50127 Contig127 ::: 0.376779 0.623221
gi|50144 Contig144 ::: 0.109227 0.890773
gi|50178 Contig178 Inexistente  ::: 0.28786 0.71214
gi|50181 Contig181 ::: 0.339714 0.660286
gi|50233 Contig233 mitochondrial ribosome small subunit component ::: 0.326981 0.673019
gi|50261 Contig261 ::: 0.150279 0.849721
gi|50292 Contig292 ::: 0.12023 0.87977
gi|50306 Contig306 ::: 0.196105 0.803895
gi|50307 Contig307 ::: 0.265856 0.734144
gi|50361 Contig361 ::: 0.493192 0.506808
gi|50414 Contig414 Cysteine desulfurase ::: 0.332428 0.667572
gi|50418 Contig418 ::: 0.263594 0.736406
gi|50433 Contig433 ::: 0.273733 0.726267
gi|50475 Contig475 ::: 0.1864 0.8136
gi|50498 Contig498 ::: 0.308315 0.691685
gi|50503 Contig503 ::: 0.188649 0.811351
gi|50727 Contig727 26S proteasome regulatory subunit  ::: 0.218236 0.781764
gi|50824 Contig824 ::: 0.427788 0.572212
gi|50853 Contig853 ::: 0.30306 0.69694
gi|50884 Contig884 ::: 0.163593 0.836407
gi|50911 Contig911 ::: 0.128221 0.871779
gi|50918 Contig918 ::: 0.320029 0.679971
gi|50969 Contig969 ::: 0.131128 0.868872
gi|50981 Contig981 ran/spi1 binding protein ::: 0.491918 0.508082
gi|50994 Contig994 ::: 0.395713 0.604287
gi|51010 Contig1010 ::: 0.414261 0.585739
gi|51016 Contig1016 fibrillarin ::: 0.479485 0.520515
gi|51075 Contig1075 ::: 0.164352 0.835648
gi|51111 Contig1111 ::: 0.318411 0.681589
gi|51123 Contig1123 putative DNA repair and recombination protein  ::: 0.478427 0.521573
gi|51190 Contig1190 ::: 0.106664 0.893336
gi|51203 Contig1203 ::: 0.413772 0.586228
gi|51228 Contig1228 ::: 0.117397 0.882603
gi|51298 Contig1298 ::: 0.395291 0.604709
gi|51307 Contig1307 glyoxylate pathway regulator GPR1 ::: 0.314754 0.685246
gi|51350 Contig1350 ::: 0.323558 0.676442
gi|51368 Contig1368 ::: 0.35144 0.64856
gi|51415 Contig1415 Similar to fission yeast UV induced protein ::: 0.249134 0.750866
gi|51428 Contig1428 ADP-ribosylation factor-like protein ::: 0.0855858 0.914414
gi|51445 Contig1445 ::: 0.14771 0.85229
gi|51466 Contig1466 Chain A, Structure Of Ubiquitin-Like Protein ::: 0.493138 0.506862
gi|51467 Contig1467 Probable prefoldin subunit 3  ::: 0.459623 0.540377
gi|51505 Contig1505 ::: 0.231169 0.768831
gi|51517 Contig1517 ::: 0.102621 0.897379
gi|51529 Contig1529 acetyltransferase ::: 0.443144 0.556856
gi|51532 Contig1532 ::: 0.393487 0.606513
gi|51582 Contig1582 NADH-UBIQUINONE OXIDOREDUCTASE 10.5 KD SUBUNIT (COMPLEX I) ::: 0.312437
0.687563
gi|51584 Contig1584 L-iditol 2-dehydrogenase ::: 0.152177 0.847823
gi|51595 Contig1595 ::: 0.346206 0.653794
gi|51605 Contig1605 ::: 0.490105 0.509895
gi|51620 Contig1620 ::: 0.397367 0.602633
gi|51624 Contig1624 ::: 0.179111 0.820889
gi|51638 Contig1638 ::: 0.363591 0.636409
gi|51656 Contig1656 ::: 0.480487 0.519513
gi|51696 Contig1696 phosphatidylserine decarboxylase ::: 0.330561 0.669439
gi|51702 Contig1702 ::: 0.191464 0.808536
gi|51708 Contig1708 ::: 0.222337 0.777663
gi|51724 Contig1724 ::: 0.177462 0.822538
gi|51732 Contig1732 ::: 0.488523 0.511477
```

gi|51733 Contig1733 ::: 0.40922 0.59078
gi|51812 Contig1812 ::: 0.427338 0.572662
gi|51833 Contig1833 ::: 0.452065 0.547935
gi|51861 Contig1861 RETROTRANSPOSABLE L1 ELEMENT ::: 0.454211 0.545789
gi|51869 Contig1869 ::: 0.160963 0.839037
gi|51949 Contig1949 ::: 0.285861 0.714139
gi|51950 Contig1950 ::: 0.439689 0.560311
gi|51955 Contig1955 ::: 0.321822 0.678178
gi|51977 Contig1977 probable GPR/FUN34 family protein - fission yeast (Schizosaccharomyces pombe) ::: 0.456712 0.543288
gi|52035 Contig2035 Serine hydroxymethyltransferase, mitochondrial; Shm1p ::: 0.35326 0.64674
gi|52060 Contig2060 ::: 0.251659 0.748341
gi|52065 Contig2065 ::: 0.41708 0.58292
gi|52074 Contig2074 ::: 0.352063 0.647937
gi|52144 Contig2144 glutamyl-trna synthetase, mitochondrial ::: 0.233191 0.766809
gi|52177 Contig2177 ::: 0.468224 0.531776
gi|52244 Contig2244 ::: 0.458523 0.541477
gi|52284 Contig2284 ::: 0.463665 0.536335
gi|52288 Contig2288 ::: 0.41159 0.58841
gi|52316 Contig2316 Bud23p ::: 0.324533 0.675467
gi|52339 Contig2339 related to flavin-containing monooxygenase ::: 0.395468 0.604532
gi|52341 Contig2341 ::: 0.457071 0.542929
gi|52407 Contig2407 ::: 0.18362 0.81638
gi|52449 Contig2449 ::: 0.452659 0.547341
gi|52525 Contig2525 probable purine nucleoside phosphorylase ::: 0.385194 0.614806
gi|52529 Contig2529 ::: 0.164076 0.835924
gi|52544 Contig2544 ::: 0.397375 0.602625
gi|52567 Contig2567 ::: 0.212267 0.787733
gi|52602 Contig2602 Hypothetical ORF ::: 0.101598 0.898402
gi|52603 Contig2603 P-type ATPase ::: 0.418766 0.581234
gi|52667 PBDAN-M1-001t_A02 ::: 0.443678 0.556322
gi|52670 PBDAN-M1-001t_C07 negative regulator of sexual conjugation and meiosis (EC 2.7.1.-) [Schizosaccharomyces pombe] ::: 0.353344 0.646656
gi|52679 PBDAN-M1-002t_B10 ::: 0.275995 0.724005
gi|52684 PBDAN-M1-002t_E09 ::: 0.22514 0.77486
gi|52688 PBDAN-M1-002t_F11 ::: 0.273658 0.726342
gi|52698 PBDBD-M1-010t_D11 ::: 0.345504 0.654496
gi|52708 PBDBR-M1-007t_D05 ::: 0.419013 0.580987
gi|52715 PBDBR-M1-008t_B03 60s ribosomal protein l32 mitochondrial precursor ::: 0.461127 0.538873
gi|52723 PBDBR-M1-008t_E05 May be pathogenicity protein ::: 0.393298 0.606702
gi|52725 PBDBR-M1-008t_G01 Threonyl-tRNA synthetase, cytoplasmic ::: 0.285809 0.714191
gi|52731 PBDCR-M1-011t_A12 mitochondrial ribosomal protein ::: 0.383631 0.616369
gi|52749 PBDCR-M1-012t_D10 ::: 0.0934245 0.906576
gi|52774 PBDEC-M1-014t_C09 ::: 0.243207 0.756793
gi|52791 PBDEC-M1-066t_F12 ::: 0.213295 0.786705
gi|52817 PBDEV-M1-142t_F04 ::: 0.253152 0.746848
gi|52818 PBDEV-M1-142t_G06 putative allantoicase ::: 0.420339 0.579661
gi|52820 PBDEV-Y1-032t_B01 ::: 0.353557 0.646443
gi|52827 PBDEV-Y1-032t_D06 Inexistente ::: 0.345622 0.654378
gi|52832 PBDEV-Y1-032t_G12 Inexistente ::: 0.254173 0.745827
gi|52838 PBDEV-Y1-033t_C05 Inexistente ::: 0.431393 0.568607
gi|52841 PBDEV-Y1-033t_H01 ::: 0.424034 0.575966
gi|52852 PBDEV-Y1-034t_G04 ::: 0.044556 0.955444
gi|52856 PBDEX-M1-001t_B09 ::: 0.405766 0.594234
gi|52896 PBDEX-M1-004t_E06 ::: 0.369918 0.630082
gi|52920 PBDEX-M1-006t_B01 ::: 0.263306 0.736694
gi|52936 PBDEX-M1-006t_G01 possible campothecin resistance conferring protein rcaA ::: 0.0866532 0.913347
gi|52953 PBDEX-M1-007t_H07 ::: 0.337182 0.662818
gi|52974 PBDEX-M1-008t_G06 ::: 0.194877 0.805123
gi|52976 PBDEX-M1-008t_H02 ::: 0.478648 0.521352
gi|52987 PBDEX-M1-009t_D10 ::: 0.31101 0.68899
gi|53006 PBDEX-M1-010t_D01 putative peptide synthetase; with 3 Phosphopantetheine attachment sites ::: 0.245334 0.754666
gi|53014 PBDEX-M1-010t_F01 hypothetical protein; possible RNA binding ::: 0.433059 0.566941
gi|53026 PBDEX-M1-011t_D05 ::: 0.23249 0.76751
gi|53045 PBDEX-M1-012t_G01 ::: 0.353784 0.646216
gi|53063 PBDEX-M1-014t_C05 ::: 0.256927 0.743073
gi|53079 PBDEX-M1-016t_D10 ::: 0.363642 0.636358
gi|53085 PBDEX-M1-016t_G08 ::: 0.262903 0.737097
gi|53101 PBDEX-M1-017t_F09 ::: 0.311741 0.688259
gi|53107 PBDEX-M1-018t_A11 ::: 0.0178232 0.982177
gi|53117 PBDEX-M1-018t_G03 ::: 0.259157 0.740843
gi|53128 PBDEX-M1-019t_D04 threonine dehydratase ::: 0.192581 0.807419
gi|53135 PBDEX-M1-020t_A12 ::: 0.494924 0.505076
gi|53149 PBDEX-M1-021t_E12 ::: 0.381831 0.618169
gi|53167 PBDEX-M1-022t_F03 ::: 0.417499 0.582501

gi|53192 PBDEX-M1-024t_E05 ::: 0.241739 0.758261
gi|53222 PBDEX-M1-026t_E12 ::: 0.168333 0.831667
gi|53237 PBDEX-M1-027t_G06 ::: 0.367688 0.632312
gi|53241 PBDEX-M1-028t_A05 60S ribosomal protein L3 ::: 0.395343 0.604657
gi|53242 PBDEX-M1-028t_C12 isoleucyl-trna synthetase ::: 0.249542 0.750458
gi|53254 PBDEX-M1-029t_A07 ribosomal large subunit assembly and maintenance ::: 0.365988
0.634012
gi|53258 PBDEX-M1-029t_C08 ::: 0.390889 0.609111
gi|53283 PBDEX-M1-030t_F03 ::: 0.347487 0.652513
gi|53311 PBDEX-M1-033t_D10 ::: 0.180205 0.819795
gi|53320 PBDEX-M1-033t_G12 ::: 0.485732 0.514268
gi|53345 PBDEX-M1-035t_A10 Peptidyl-prolyl cis-trans isomerases ::: 0.0401906 0.959809
gi|53356 PBDEX-M1-035t_G02 ::: 0.0287957 0.971204
gi|53357 PBDEX-M1-035t_G11 ::: 0.459847 0.540153
gi|53358 PBDEX-M1-035t_H03 ::: 0.28981 0.71019
gi|53365 PBDEX-M1-036t_C09 PUTATIVE TRANSPORTER ::: 0.295396 0.704604
gi|53388 PBDEX-M1-037t_D03 GTP cyclohydrolase ::: 0.230253 0.769747
gi|53392 PBDEX-M1-037t_G12 ::: 0.335158 0.664842
gi|53393 PBDEX-M1-037t_H02 probable membrane protein YOL130w ::: 0.153003 0.846997
gi|53425 PBDEX-M1-040t_E04 ::: 0.379948 0.620052
gi|53427 PBDEX-M1-040t_F01 ::: 0.477042 0.522958
gi|53447 PBDEX-M1-042t_A05 ::: 0.165574 0.834426
gi|53492 PBDEX-M1-045t_F06 ::: 0.0921422 0.907858
gi|53497 PBDEX-M1-046t_A01 ::: 0.386246 0.613754
gi|53505 PBDEX-M1-046t_D06 ubiquitin fusion degradation protein ::: 0.222355 0.777645
gi|53509 PBDEX-M1-046t_F07 ::: 0.204826 0.795174
gi|53523 PBDEX-M1-047t_E04 ::: 0.128798 0.871202
gi|53532 PBDEX-M1-047t_H11 ATP-binding cassette, sub-family A, member 4 ::: 0.458463 0.541537
gi|53554 PBDEX-M1-050t_C03 [NID] ::: 0.429414 0.570586
gi|53590 PBDEX-M1-052t_F01 [NID] ::: 0.285246 0.714754
gi|53591 PBDEX-M1-052t_F11 [NID] ::: 0.389023 0.610977
gi|53592 PBDEX-M1-052t_H02 [ncRNA] ::: 0.00986767 0.990132
gi|53596 PBDEX-M1-053t_A11 [NID] ::: 0.384599 0.615401
gi|53641 PBDEX-M1-056t_B03 [NID] ::: 0.0995009 0.900499
gi|53645 PBDEX-M1-056t_C09 [NID] ::: 0.373623 0.626377
gi|53652 PBDEX-M1-056t_E10 [NID] ::: 0.376278 0.623722
gi|53655 PBDEX-M1-056t_G04 ::: 0.166646 0.833354
gi|53659 PBDEX-M1-092t_A01 Hypothetical ORF ::: 0.151287 0.848713
gi|53660 PBDEX-M1-092t_A06 conserved hypothetical protein ::: 0.276637 0.723363
gi|53669 PBDEX-M1-092t_F03 ::: 0.288627 0.711373
gi|53687 PBDEX-M1-094t_A02 ::: 0.386224 0.613776
gi|53690 PBDEX-M1-094t_B04 ::: 0.283222 0.716778
gi|53695 PBDEX-M1-094t_D08 UTP-glucose-1-phosphate uridylyltransferase ::: 0.272166 0.727834
gi|53699 PBDEX-M1-094t_E12 Severe Depolymerization of Actin ::: 0.267354 0.732646
gi|53708 PBDEX-M1-095t_B04 ::: 0.106739 0.893261
gi|53710 PBDEX-M1-095t_B09 hypothetical protein ::: 0.0395417 0.960458
gi|53711 PBDEX-M1-095t_C05 ::: 0.45291 0.54709
gi|53718 PBDEX-M1-095t_H11 ::: 0.424975 0.575025
gi|53720 PBDEX-M1-096t_B03 ZNF127 ::: 0.205466 0.794534
gi|53722 PBDEX-M1-096t_B12 ::: 0.280663 0.719337
gi|53723 PBDEX-M1-096t_C04 ::: 0.0263734 0.973627
gi|53726 PBDEX-M1-096t_D12 ::: 0.484892 0.515108
gi|53730 PBDEX-M1-096t_F05 tryptophanyl tRNA synthetase ::: 0.0342576 0.965742
gi|53743 PBDEX-M1-097t_E10 sterol carrier protein 2 ::: 0.265085 0.734915
gi|53745 PBDEX-M1-097t_F08 ::: 0.363319 0.636681
gi|53751 PBDEX-M1-098t_A01 ::: 0.439759 0.560241
gi|53752 PBDEX-M1-098t_A04 ::: 0.155149 0.844851
gi|53753 PBDEX-M1-098t_A08 ::: 0.298358 0.701642
gi|53767 PBDEX-M1-099t_C03 ::: 0.420401 0.579599
gi|53778 PBDEX-M1-100t_A05 probable flavoprotein ::: 0.257978 0.742022
gi|53779 PBDEX-M1-100t_A10 anthranilate synthase ::: 0.411346 0.588654
gi|53782 PBDEX-M1-100t_D06 ::: 0.217789 0.782211
gi|53788 PBDEX-M1-100t_F08 nitrate reductase ::: 0.137969 0.862031
gi|53794 PBDEX-M1-101t_A03 ::: 0.159583 0.840417
gi|53797 PBDEX-M1-101t_B03 ::: 0.315908 0.684092
gi|53806 PBDEX-M1-101t_F07 glucan synthase ::: 0.458151 0.541849
gi|53824 PBDEX-M1-102t_F03 ::: 0.125678 0.874322
gi|53825 PBDEX-M1-102t_F04 ::: 0.260486 0.739514
gi|53835 PBDEX-M1-103t_F07 Cell division control protein 4 ::: 0.340849 0.659151
gi|53838 PBDEX-M1-103t_G09 ::: 0.300771 0.699229
gi|53841 PBDEX-M1-103t_H07 Involved in lysine biosynthesis, oxidative stress protection :::
0.374045 0.625955
gi|53855 PBDEX-M1-105t_G06 ::: 0.331607 0.668393
gi|53885 PBDEX-Y1-002t_D04 ::: 0.14771 0.85229
gi|53912 PBDEX-Y1-004t_E12 ATP synthase epsilon chain, mitochondrial ::: 0.384078 0.615922
gi|53943 PBDEX-Y1-006t_F05 ::: 0.180669 0.819331
gi|53954 PBDEX-Y1-008t_B02 ::: 0.189254 0.810746
gi|53962 PBDEX-Y1-008t_E03 ::: 0.150362 0.849638

gi|53975 PBDEX-Y1-009t_E06 ::: 0.352182 0.647818
gi|53980 PBDEX-Y1-010t_D05 ::: 0.311304 0.688696
gi|53997 PBDEX-Y1-012t_A01 dynein light intermediate chain 1 ::: 0.336081 0.663919
gi|54029 PBDEX-Y1-014t_B11 ::: 0.322134 0.677866
gi|54032 PBDEX-Y1-014t_E06 eukaryotic translation initiation factor EIF-2B subunit 3 ::: 0.117
0.883
gi|54039 PBDEX-Y1-015t_A06 ::: 0.43042 0.56958
gi|54057 PBDEX-Y1-016t_G01 ::: 0.485032 0.514968
gi|54063 PBDEX-Y1-017t_C03 zinc finger protein [Schizosaccharomyces pombe] ::: 0.392607
0.607393
gi|54080 PBDEX-Y1-018t_G01 ::: 0.29606 0.70394
gi|54091 PBDEX-Y1-019t_G12 multiple ankyrin repeat single KH domain protein ::: 0.272463
0.727537
gi|54094 PBDEX-Y1-020t_B07 ::: 0.137211 0.862789
gi|54102 PBDEX-Y1-020t_H11 ::: 0.21106 0.78894
gi|54106 PBDEX-Y1-021t_D03 ::: 0.40666 0.59334
gi|54111 PBDEX-Y1-021t_H04 ::: 0.168725 0.831275
gi|54119 PBDEX-Y1-022t_F06 ::: 0.488632 0.511368
gi|54123 PBDEX-Y1-023t_B10 ::: 0.24108 0.75892
gi|54149 PBDEX-Y1-024t_G11 ::: 0.11958 0.88042
gi|54150 PBDEX-Y1-025t_A07 ::: 0.441924 0.558076
gi|54153 PBDEX-Y1-025t_A11 ::: 0.102283 0.897717
gi|54164 PBDEX-Y1-026t_B03 SCHPO HYPOTHETICAL 103.2 KDA PROTEIN C24B11.10C IN CHROMOSOME I :::
0.101617 0.898383
gi|54175 PBDEX-Y1-027t_A10 ::: 0.119676 0.880324
gi|54182 PBDEX-Y1-027t_F05 ::: 0.387411 0.612589
gi|54183 PBDEX-Y1-027t_G08 ::: 0.287683 0.712317
gi|54207 PBDEX-Y1-029t_C11 ::: 0.452521 0.547479
gi|54209 PBDEX-Y1-029t_E03 ::: 0.302453 0.697547
gi|54239 PBDEX-Y1-032t_A12 ::: 0.296018 0.703982
gi|54246 PBDEX-Y1-032t_F01 ::: 0.228111 0.771889
gi|54270 PBDEX-Y1-034t_C01 ::: 0.395495 0.604505
gi|54271 PBDEX-Y1-034t_C03 ::: 0.413365 0.586635
gi|54272 PBDEX-Y1-034t_C07 ::: 0.420526 0.579474
gi|54286 PBDEX-Y1-035t_D04 ::: 0.210913 0.789087
gi|54288 PBDEX-Y1-035t_E12 ::: 0.318855 0.681145
gi|54292 PBDEX-Y1-035t_G12 ::: 0.245761 0.754239
gi|54296 PBDEX-Y1-036t_A11 ::: 0.453845 0.546155
gi|54323 PBDEX-Y1-039t_A11 ::: 0.271953 0.728047
gi|54327 PBDEX-Y1-039t_D11 ::: 0.266017 0.733983
gi|54342 PBDEX-Y1-040t_G08 chloride channel ::: 0.33008 0.66992
gi|54352 PBDEX-Y1-041t_D11 ::: 0.327196 0.672804
gi|54356 PBDEX-Y1-041t_H01 ::: 0.041519 0.958481
gi|54359 PBDEX-Y1-041t_H08 ::: 0.0540448 0.945955
gi|54360 PBDEX-Y1-042t_A10 ATP-binding cassette (ABC) transporter family member ::: 0.385476
0.614524
gi|54367 PBDEX-Y1-042t_G03 ::: 0.473047 0.526953
gi|54377 PBDEX-Y1-043t_D02 ::: 0.174295 0.825705
gi|54378 PBDEX-Y1-043t_D04 ::: 0.227974 0.772026
gi|54395 PBDEX-Y1-044t_C12 ::: 0.383869 0.616131
gi|54397 PBDEX-Y1-044t_D03 ::: 0.434395 0.565605
gi|54406 PBDFA-M1-023t_A06 ::: 0.104976 0.895024
gi|54416 PBDFM-M1-058t_A06 ::: 0.0980774 0.901923
gi|54424 PBDFM-M1-058t_C10 ::: 0.0646014 0.935399
gi|54434 PBDFM-Y1-206t_D10 formamidase ::: 0.387593 0.612407
gi|54449 PBDHV-M1-054t_B07 ::: 0.2761 0.7239
gi|54472 PBDIP-M1-026t_C03 ::: 0.200001 0.799999
gi|54477 PBDIP-M1-026t_E09 ::: 0.314631 0.685369
gi|54487 PBDLP-M1-029t_F10 calcium transporting ATPase ::: 0.365343 0.634657
gi|54488 PBDLP-M1-029t_F11 ::: 0.0595933 0.940407
gi|54496 PBDLP-M1-030t_E11 ::: 0.206887 0.793113
gi|54498 PBDLP-M1-030t_G12   ::: 0.260562 0.739438
gi|54505 PBDMA-Y1-006t_F01 ::: 0.461815 0.538185
gi|54506 PBDMA-Y1-006t_F10 ::: 0.0891999 0.9108
gi|54507 PBDMF-Y1-007t_A01 ::: 0.235721 0.764279
gi|54515 PBDMF-Y1-007t_D09 ::: 0.0463157 0.953684
gi|54532 PBDMO-Y1-009t_B09 ::: 0.198138 0.801862
gi|54543 PBDMO-Y1-010t_F08 hypothetical protein 2SCK31.14c [Streptomyces coelicolor] :::
0.309214 0.690786
gi|54548 PBDMP-M1-031t_A06 METHYLENETETRAHYDROFOLATE REDUCTASE 2 ::: 0.260392 0.739608
gi|54556 PBDMP-M1-031t_F01 Nitrilase 4 ::: 0.356163 0.643837
gi|54558 PBDMP-M1-031t_G08 ::: 0.190369 0.809631
gi|54565 PBDMP-M1-032t_F10 ::: 0.479372 0.520628
gi|54567 PBDMP-M1-032t_H04 ::: 0.322044 0.677956
gi|54570 PBDMP-M1-033t_A11 ::: 0.421595 0.578405
gi|54573 PBDMP-M1-033t_C08 ::: 0.206287 0.793713
gi|54583 PBDMP-M1-033t_F10 ::: 0.196526 0.803474
gi|54594 PBDMP-M1-034t_H07 ::: 0.264779 0.735221

gi|54595 PBDMP-Y1-035t_D07 ::: 0.203621 0.796379
gi|54599 PBDMP-Y1-035t_H07 dimeric dihydrodiol dehydrogenase ::: 0.441963 0.558037
gi|54609 PBDMP-Y1-036t_H04 ::: 0.21748 0.78252
gi|54618 PBDMP-Y1-037t_E11 cell division control protein cdc14 ::: 0.218399 0.781601
gi|54626 PBDMP-Y1-039t_G03 ::: 0.473426 0.526574
gi|54633 PBDMP-Y1-040t_G03 GATA transcription factor ::: 0.0563562 0.943644
gi|54637 PBDMP-Y1-168t_D08 hypothetical protein ::: 0.152173 0.847827
gi|54641 PBDMP-Y1-169t_A06 ::: 0.0700669 0.929933
gi|54651 PBDMS-Y1-003t_C03 ::: 0.0700547 0.929945
gi|54662 PBDMS-Y1-004t_A07 ::: 0.340928 0.659072
gi|54666 PBDMS-Y1-004t_F03 ::: 0.361189 0.638811
gi|54680 PBDPA-Y1-021t_H05 ::: 0.257653 0.742347
gi|54714 PBDRS-M1-028t_E09 ::: 0.467091 0.532909
gi|54715 PBDRS-M1-028t_G02 ::: 0.445566 0.554434
gi|54733 PBDRV-M1-041t_F03 ::: 0.348066 0.651934
gi|54738 PBDRV-M1-042t_A05 ::: 0.389531 0.610469
gi|54754 PBDRV-M1-042t_G07 ::: 0.207156 0.792844
gi|54757 PBDRV-M1-043t_A01 Protein required for cell viability ::: 0.315694 0.684306
gi|54765 PBDRV-M1-043t_G08 ::: 0.394269 0.605731
gi|54773 PBDRV-M1-044t_E10 ::: 0.143654 0.856346
gi|54774 PBDRV-M1-044t_F03 ::: 0.354028 0.645972
gi|54781 PBDRV-M1-044t_G11 ::: 0.14073 0.85927
gi|54784 PBDRV-M1-045t_E06 ::: 0.164845 0.835155
gi|54790 PBDRV-Y1-041t_B04 chromatin maintenance and transcriptional regulation ::: 0.242565
0.757435
gi|54807 PBDRV-Y1-042t_B07 ::: 0.298924 0.701076
gi|54812 PBDRV-Y1-042t_E03 ::: 0.166601 0.833399
gi|54818 PBDRV-Y1-042t_H07 ::: 0.47402 0.52598
gi|54820 PBDRV-Y1-043t_A02 ::: 0.181301 0.818699
gi|54834 PBDRV-Y1-044t_D01 ::: 0.472375 0.527625
gi|54844 PBDRV-Y1-045t_A06 ::: 0.302806 0.697194
gi|54848 PBDRV-Y1-045t_G03 ::: 0.476718 0.523282
gi|54851 PBDRV-Y1-100t_B04 ::: 0.302377 0.697623
gi|54862 PBDRV-Y1-102t_H10 ::: 0.141871 0.858129
gi|54866 PBDRV-Y1-104t_D04 ::: 0.43281 0.56719
gi|54891 PBDSP-M1-048t_C04 ::: 0.216693 0.783307
gi|54903 PBDSP-M1-049t_E06 ::: 0.476448 0.523552
gi|54906 PBDSP-M1-049t_F09 dna repair protein rad13 ::: 0.402145 0.597855
gi|54908 PBDSP-M1-049t_G11 ATP-binding cassette (ABC) transporter, peroxisomal long-chain
fatty acid import ::: 0.231383 0.768617
gi|54916 PBDSP-M1-050t_C11 ::: 0.157852 0.842148
gi|54928 PBDSP-Y1-046t_F02 ::: 0.455029 0.544971
gi|54932 PBDSP-Y1-046t_G01 ::: 0.20913 0.79087
gi|54940 PBDSP-Y1-047t_F05 ::: 0.178335 0.821665
gi|54941 PBDSP-Y1-047t_F07 probable 20S proteasome subunit Y7 ::: 0.189412 0.810588
gi|54958 PBDSP-Y1-048t_G09 UBA1 "ubiquitin activating enzyme, similar to Uba2p" ::: 0.126035
0.873965
gi|54966 PBDSP-Y1-049t_B01 ::: 0.0453578 0.954642
gi|54972 PBDSP-Y1-049t_E07 ::: 0.477462 0.522538
gi|54979 PBDSP-Y1-050t_A04 ::: 0.385027 0.614973
gi|54982 PBDSP-Y1-050t_C04 ::: 0.115559 0.884441
gi|54983 PBDSP-Y1-050t_C07 ::: 0.150015 0.849985
gi|54987 PBDSP-Y1-050t_G08 ::: 0.28883 0.71117
gi|54991 PBGAC-M1-015t_D01 ::: 0.0665792 0.933421
gi|54992 PBGAC-M1-015t_D06 Sulfur metabolite repression control protein ::: 0.432191 0.567809
gi|54994 PBGAC-M1-015t_E11 ::: 0.106586 0.893414
gi|54995 PBGAC-M1-015t_F01 LIPOIC ACID SYNTHETASE ::: 0.411803 0.588197
gi|55017 PBGCB-M1-036t_E02 ::: 0.467395 0.532605
gi|55018 PBGCB-M1-036t_E08 Glucosamine-phosphate N-acetyltransferase ::: 0.196962 0.803038
gi|55022 PBGCB-M1-036t_F06 ::: 0.37638 0.62362
gi|55025 PBGCM-M1-017t_B01 ::: 0.471574 0.528426
gi|55040 PBGCM-M1-018t_C03 ::: 0.311726 0.688274
gi|55132 PBGEV-M1-500t_C10 ::: 0.223227 0.776773
gi|55135 PBGEV-M1-500t_H06 ::: 0.296049 0.703951
gi|55159 PBGEV-Y1-503t_E03 ::: 0.121873 0.878127
gi|55170 PBGEX-M1-061t_B07 Protoplasts-SecreTed protein ::: 0.124009 0.875991
gi|55192 PBGEX-M1-064t_A05 ::: 0.271699 0.728301
gi|55199 PBGEX-M1-064t_D02 ::: 0.373627 0.626373
gi|55200 PBGEX-M1-064t_D07 ::: 0.408313 0.591687
gi|55249 PBGEX-M1-068t_G06 ::: 0.156408 0.843592
gi|55252 PBGEX-M1-068t_H11 ::: 0.371293 0.628707
gi|55257 PBGEX-M1-069t_C12 hypothetical protein SPBC31F10.03 - fission yeast
(Schizosaccharomyces ::: 0.206313 0.793687
gi|55260 PBGEX-M1-069t_E07 membrane dipeptidase ::: 0.173235 0.826765
gi|55266 PBGEX-M1-070t_B09 ::: 0.240149 0.759851
gi|55309 PBGEX-M1-072t_G09 ::: 0.164684 0.835316
gi|55322 PBGEX-M1-074t_E04 GTP cyclohydrolase I ::: 0.450431 0.549569
gi|55328 PBGEX-M1-075t_D09 tubulin alpha-1 chain ::: 0.346274 0.653726

gi|55338 PBGEX-Y1-061t_D05 ::: 0.296408 0.703592
gi|55346 PBGEX-Y1-061t_H09 ::: 0.345906 0.654094
gi|55361 PBGEX-Y1-065t_C05 ::: 0.436895 0.563105
gi|55376 PBGEX-Y1-066t_F06 ::: 0.225202 0.774798
gi|55386 PBGEX-Y1-068t_G02 ::: 0.137891 0.862109
gi|55399 PBGEX-Y1-069t_H01 ::: 0.474637 0.525363
gi|55401 PBGEX-Y1-069t_H10 ::: 0.489447 0.510553
gi|55439 PBGEX-Y1-073t_B09 ::: 0.448556 0.551444
gi|55450 PBGEX-Y1-074t_C02 ::: 0.228588 0.771412
gi|55462 PBGEX-Y1-075t_A04 related to microfibril-associated protein ::: 0.484929 0.515071
gi|55466 PBGEX-Y1-075t_F04 related to sphingoid base-phosphate phosphatase ::: 0.251672
0.748328
gi|55505 PBGEX-Y1-079t_A11 ::: 0.395102 0.604898
gi|55514 PBGEX-Y1-079t_G12 ::: 0.487019 0.512981
gi|55521 PBGEX-Y1-080t_B10 ::: 0.171256 0.828744
gi|55532 PBGEX-Y1-081t_C01 ::: 0.478914 0.521086
gi|55538 PBGEX-Y1-081t_F02 ::: 0.0741284 0.925872
gi|55545 PBGEX-Y1-082t_C03 ::: 0.160729 0.839271
gi|55554 PBGEX-Y1-082t_G03 ::: 0.126141 0.873859
gi|55570 PBGEX-Y1-083t_G12 ::: 0.323371 0.676629
gi|55582 PBGEX-Y1-084t_E07 ::: 0.457982 0.542018
gi|55583 PBGEX-Y1-084t_E08 catalase isozyme P ::: 0.313538 0.686462
gi|55607 PBGEX-Y1-086t_D03 ::: 0.168396 0.831604
gi|55616 PBGEX-Y1-086t_H04 ::: 0.484721 0.515279
gi|55632 PBGEX-Y1-087t_F08 ::: 0.309322 0.690678
gi|55644 PBGEX-Y1-088t_C10 ::: 0.412121 0.587879
gi|55645 PBGEX-Y1-088t_C12 ::: 0.467927 0.532073
gi|55692 PBGEX-Y1-093t_C12 ::: 0.223362 0.776638
gi|55694 PBGEX-Y1-093t_F02 Branched-chain amino acid aminotransferase ::: 0.42123 0.57877
gi|55698 PBGEX-Y1-093t_H06 ::: 0.485736 0.514264
gi|55713 PBGEX-Y1-095t_F09 ::: 0.335089 0.664911
gi|55718 PBGEX-Y1-096t_B07 ::: 0.323904 0.676096
gi|55721 PBGEX-Y1-096t_C07 ::: 0.344365 0.655635
gi|55725 PBGEX-Y1-096t_F10 ::: 0.144623 0.855377
gi|55727 PBGEX-Y1-096t_H01 ::: 0.119593 0.880407
gi|55728 PBGEX-Y1-096t_H02 ::: 0.0965649 0.903435
gi|55738 PBGEX-Y1-098t_E02 DNA binding protein NsdD ::: 0.154989 0.845011
gi|55755 PBGEX-Y1-100t_D12 ::: 0.0267105 0.973289
gi|55774 PBGEX-Y1-102t_B04 ::: 0.456765 0.543235
gi|55781 PBGEX-Y1-103t_B08 Hypothetical ORF ::: 0.31015 0.68985
gi|55782 PBGEX-Y1-103t_B11 ::: 0.471384 0.528616
gi|55783 PBGEX-Y1-103t_C02 ::: 0.181816 0.818184
gi|55794 PBGEX-Y1-104t_D01 ::: 0.114395 0.885605
gi|55809 PBGEX-Y1-105t_H04 ::: 0.362943 0.637057
gi|55812 PBGEX-Y1-106t_B03 probable translation elongation factor ::: 0.326251 0.673749
gi|55817 PBGEX-Y1-106t_H08 ::: 0.48212 0.51788
gi|55821 PBGEX-Y1-108t_D08 ::: 0.128889 0.871111
gi|55836 PBGEX-Y1-110t_G04 ::: 0.271004 0.728996
gi|55837 PBGEX-Y1-111t_A01 ::: 0.479592 0.520408
gi|55846 PBGEX-Y1-111t_G08 ::: 0.376928 0.623072
gi|55848 PBGEX-Y1-112t_C02 ::: 0.343672 0.656328
gi|55849 PBGEX-Y1-112t_D12 ::: 0.102863 0.897137
gi|55857 PBGEX-Y1-113t_A11 ::: 0.492005 0.507995
gi|55865 PBGEX-Y1-114t_E04 ::: 0.349603 0.650397
gi|55866 PBGEX-Y1-114t_E07 Ribosomal protein ::: 0.262859 0.737141
gi|55870 PBGEX-Y1-115t_A11 MFS transporter of unknown specificity ::: 0.242102 0.757898
gi|55875 PBGEX-Y1-115t_E02 ::: 0.393617 0.606383
gi|55883 PBGEX-Y1-115t_H10 Methionyl-tRNA synthetase ::: 0.27803 0.72197
gi|55888 PBGEX-Y1-117t_D08 chitinase ::: 0.483549 0.516451
gi|55892 PBGEX-Y1-117t_G08 ::: 0.434592 0.565408
gi|55896 PBGEX-Y1-118t_F11 ::: 0.150371 0.849629
gi|55902 PBGEX-Y1-119t_A12 ::: 0.183988 0.816012
gi|55915 PBGEX-Y1-121t_C02 ::: 0.351609 0.648391
gi|55922 PBGEX-Y1-121t_D09 ::: 0.357522 0.642478
gi|55925 PBGEX-Y1-121t_F02 putative protein involved in autophagy yeast apg7 homolog
[Schizosaccharomyces pombe] ::: 0.415968 0.584032
gi|55931 PBGEX-Y1-124t_A06 ::: 0.116256 0.883744
gi|55934 PBGEX-Y1-124t_C09 ::: 0.337438 0.662562
gi|55952 PBGGO-M1-038t_D10 ::: 0.17164 0.82836
gi|55970 PBGEX-M1-062t_B08 ::: 0.45657 0.54343
gi|55975 PBGEX-M1-062t_F05 ::: 0.378454 0.621546
gi|55988 PBGJP-M1-407t_C12 ::: 0.469602 0.530398
gi|55989 PBGJP-M1-407t_D09 ::: 0.421725 0.578275
gi|55996 PBGJP-Y1-014t_D12 ::: 0.410989 0.589011
gi|55997 PBGJP-Y1-014t_F07 Alcohol dehydrogenase ::: 0.212463 0.787537
gi|55999 PBGJP-Y1-015t_C03 ::: 0.359074 0.640926
gi|56000 PBGJP-Y1-015t_C05 ::: 0.433645 0.566355
gi|56008 PBGLA-Y1-016t_G06 Putative protein ::: 0.148761 0.851239

gi|56018 PBGMG-M1-420t_C08 short chain dehydrogenase/reductase family ::: 0.292128 0.707872
gi|56024 PBGMG-Y1-018t_F02 ::: 0.143429 0.856571
gi|56026 PBGRJ-M1-422t_D06 ::: 0.413519 0.586481
gi|56029 PBGRJ-M1-422t_E05 ::: 0.380996 0.619004
gi|56051 PBGRS-Y1-027t_F08 ::: 0.454233 0.545767
gi|56056 PBGEX-Y1-062t_B05 ::: 0.136538 0.863462
gi|56090 PBGEX-Y1-070t_A12 conserved hypothetical protein ::: 0.43726 0.56274
gi|56091 PBGEX-Y1-070t_C07 Putative role in early maturation of pre-rRNA and mitochondrial maintenance ::: 0.286569 0.713431
gi|56099 PBGEX-Y1-070t_F12 ::: 0.489041 0.510959
gi|56100 PBGEX-Y1-070t_G04 ::: 0.175128 0.824872
gi|56104 Contig C1162_2 citocromo P450 ::: 0.0117899 0.98821
gi|56110 PBDEX-Y1-045t_C10 [nid] ::: 0.22995 0.77005
gi|56116 PBDEX-M1-107t_E06 hypothetical protein ::: 0.463816 0.536184
gi|56119 PBDEX-M1-107t_F11 putative aspartate aminotransferase ::: 0.297297 0.702703
gi|56139 PBDEX-M1-109t_A02 [nid] ::: 0.153962 0.846038
gi|56143 PBDEX-M1-109t_B01 [nid] ::: 0.122332 0.877668
gi|56146 PBDEX-M1-109t_C08 [nid] ::: 0.371999 0.628001
gi|50015 Contig15 ::: 0.16192 0.83808
gi|50022 Contig22 ::: 0.18803 0.81197
gi|50046 Contig46 oxalyl-CoA decarboxylase [Schizosaccharomyces pombe] ::: 0.0804435 0.919557
gi|50135 Contig135 ::: 0.0944235 0.905577
gi|50193 Contig193 lysyl-tRNA synthetase ::: 0.0264664 0.973534
gi|50289 Contig289 conserved hypothetical protein ::: 0.0670907 0.932909
gi|50343 Contig343 ::: 0.181818 0.818182
gi|50353 Contig353 CONSERVED HYPOTHETICAL ::: 0.293654 0.706346
gi|50411 Contig411 ::: 0.231955 0.768045
gi|50485 Contig485 ::: 0.383852 0.616148
gi|50502 Contig502 ::: 0.451899 0.548101
gi|50520 Contig520 ::: 0.0267257 0.973274
gi|50554 Contig554 ::: 0.208643 0.791357
gi|50563 Contig563 ::: 0.0915945 0.908406
gi|50593 Contig593 hypothetical protein ::: 0.205517 0.794483
gi|50672 Contig672 ::: 0.0393634 0.960637
gi|50709 Contig709 ::: 0.123977 0.876023
gi|50722 Contig722 ::: 0.0694052 0.930595
gi|50752 Contig752 [NID] ::: 0.079225 0.920775
gi|50753 Contig753 [NID] ::: 0.281845 0.718155
gi|50768 Contig768 [NID] ::: 0.166016 0.833984
gi|50787 Contig787 [NID] ::: 0.126995 0.873005
gi|50805 Contig805 conserved hypothetical protein ::: 0.218694 0.781306
gi|50837 Contig837 ::: 0.0113653 0.988635
gi|50854 Contig854 ::: 0.0426117 0.957388
gi|50856 Contig856 ::: 0.0490557 0.950944
gi|50874 Contig874 ::: 0.0632032 0.936797
gi|50958 Contig958 ::: 0.30286 0.69714
gi|50985 Contig985 3-oxoacid CoA transferase ::: 0.0255029 0.974497
gi|50986 Contig986 ras-like protein 1 ::: 0.106008 0.893992
gi|51000 Contig1000 ::: 0.129061 0.870939
gi|51006 Contig1006 ::: 0.106977 0.893023
gi|51021 Contig1021 ::: 0.214797 0.785203
gi|51051 Contig1051 ::: 0.0413083 0.958692
gi|51070 Contig1070 ::: 0.17664 0.82336
gi|51128 Contig1128 ::: 0.288776 0.711224
gi|51166 Contig1166 ::: 0.446609 0.553391
gi|51189 Contig1189 ::: 0.0526272 0.947373
gi|51214 Contig1214 ::: 0.166491 0.833509
gi|51238 Contig1238 ::: 0.36678 0.63322
gi|51241 Contig1241 ::: 0.118406 0.881594
gi|51245 Contig1245 ::: 0.0995849 0.900415
gi|51255 Contig1255 HSP104 ::: 0.374497 0.625503
gi|51257 Contig1257 ::: 0.052639 0.947361
gi|51275 Contig1275 ::: 0.338946 0.661054
gi|51283 Contig1283 LIPOCALIN family protein ::: 0.104643 0.895357
gi|51349 Contig1349 involved in the propagation of functional mitochondria yeast ::: 0.470884 0.529116
gi|51356 Contig1356 ::: 0.0449652 0.955035
gi|51365 Contig1365 ::: 0.151787 0.848213
gi|51374 Contig1374 ::: 0.325644 0.674356
gi|51391 Contig1391 ::: 0.135832 0.864168
gi|51400 Contig1400 ::: 0.214396 0.785604
gi|51403 Contig1403 ::: 0.0390854 0.960915
gi|51405 Contig1405 ::: 0.233436 0.766564
gi|51406 Contig1406 related to zinc finger protein 1 [imported] - Neurospora crassa ::: 0.485469 0.514531
gi|51471 Contig1471 ::: 0.173616 0.826384
gi|51477 Contig1477 ::: 0.0193185 0.980681
gi|51639 Contig1639 ::: 0.264603 0.735397

gi|51663 Contig1663 Similar to yeast Ygr223cp ::: 0.180394 0.819606
gi|51672 Contig1672 ::: 0.104606 0.895394
gi|51681 Contig1681 ::: 0.282628 0.717372
gi|51703 Contig1703 conserved hypothetical protein ::: 0.351819 0.648181
gi|51726 Contig1726 ::: 0.0700306 0.929969
gi|51747 Contig1747 Peptidyl-prolyl cis-trans isomerases (cyclophillin) ::: 0.158954 0.841046
gi|51755 Contig1755 ::: 0.0970726 0.902927
gi|51777 Contig1777 ::: 0.0844469 0.915553
gi|51791 Contig1791 ::: 0.0802285 0.919772
gi|51814 Contig1814 conserved hypothetical protein SPBC3F6.04c - fission yeast
(Schizosaccharomyces pombe) ::: 0.0706033 0.929397
gi|51862 Contig1862 ::: 0.233491 0.766509
gi|51952 Contig1952 ::: 0.457495 0.542505
gi|52030 Contig2030 T41653 probable transcription or splicing factor - fission yeast
(Schizosaccharomyces pombe) ::: 0.17468 0.82532
gi|52031 Contig2031 ::: 0.464593 0.535407
gi|52072 Contig2072 Acetyl-CoA acetyltransferases ::: 0.323315 0.676685
gi|52077 Contig2077 ::: 0.027412 0.972588
gi|52113 Contig2113 ::: 0.317111 0.682889
gi|52153 Contig2153 ::: 0.430298 0.569702
gi|52208 Contig2208 ::: 0.105499 0.894501
gi|52232 Contig2232 ::: 0.0858922 0.914108
gi|52317 Contig2317 ::: 0.277869 0.722131
gi|52328 Contig2328 ::: 0.11587 0.88413
gi|52350 Contig2350 ::: 0.0534984 0.946502
gi|52380 Contig2380 ::: 0.20141 0.79859
gi|52398 Contig2398 ::: 0.0692659 0.930734
gi|52413 Contig2413 ::: 0.449397 0.550603
gi|52425 Contig2425 ::: 0.27883 0.72117
gi|52429 Contig2429 U4/U6 splicing factor PRP24 homolog ::: 0.0392336 0.960766
gi|52457 Contig2457 putative acetyltransferase ::: 0.0315654 0.968435
gi|52475 Contig2475 H+-transporting ATP synthase lipid-binding protein ::: 0.36082 0.63918
gi|52489 Contig2489 ::: 0.281861 0.718139
gi|52498 Contig2498 ::: 0.133854 0.866146
gi|52501 Contig2501 ::: 0.482942 0.517058
gi|52520 Contig2520 ::: 0.127029 0.872971
gi|52556 Contig2556 ::: 0.319671 0.680329
gi|52583 Contig2583 ::: 0.0195105 0.98049
gi|52586 Contig2586 Ribulose-5-phosphate 4-epimerase and related epimerases and aldolases :::
0.267914 0.732086
gi|52589 Contig2589 Hypothetical ORF ::: 0.0431477 0.956852
gi|52659 PBDAM-M1-005t_A06 dihydroorotate dehydrogenase [Schizosaccharomyces pombe] :::
0.284543 0.715457
gi|52660 PBDAM-M1-005t_A11 ::: 0.139713 0.860287
gi|52661 PBDAM-M1-005t_H09 ::: 0.106267 0.893733
gi|52665 PBDAM-M1-006t_F12 ::: 0.0318109 0.968189
gi|52678 PBDAN-M1-002t_B08 ::: 0.0533669 0.946633
gi|52680 PBDAN-M1-002t_C02 fatty acid coa ligase [Schizosaccharomyces pombe] ::: 0.0146116
0.985388
gi|52683 PBDAN-M1-002t_E08 ::: 0.111136 0.888864
gi|52686 PBDAN-M1-002t_F05 histidinol dehydrogenase ::: 0.0296839 0.970316
gi|52692 PBDBD-M1-009t_E06 ::: 0.0369112 0.963089
gi|52701 PBDBD-M1-010t_G03 ::: 0.0463731 0.953627
gi|52732 PBDCR-M1-011t_B10 Hypothetical ORF ::: 0.239433 0.760567
gi|52756 PBDEC-M1-013t_A01 ::: 0.287409 0.712591
gi|52758 PBDEC-M1-013t_C06 ::: 0.0104739 0.989526
gi|52773 PBDEC-M1-014t_C06 ::: 0.0653537 0.934646
gi|52814 PBDEV-M1-022t_F06 ::: 0.188177 0.811823
gi|52823 PBDEV-Y1-032t_B08 Inexistente ::: 0.109608 0.890392
gi|52837 PBDEV-Y1-033t_B02 Inexistente ::: 0.0734451 0.926555
gi|52844 PBDEV-Y1-034t_A10 ::: 0.150458 0.849542
gi|52850 PBDEV-Y1-034t_F03 ::: 0.130711 0.869289
gi|52875 PBDEX-M1-003t_E05 ::: 0.165008 0.834992
gi|52880 PBDEX-M1-003t_F10 ::: 0.14807 0.85193
gi|52939 PBDEX-M1-007t_A04 ::: 0.103745 0.896255
gi|52943 PBDEX-M1-007t_B11 ::: 0.0171052 0.982895
gi|52948 PBDEX-M1-007t_E09 ::: 0.055472 0.944528
gi|52951 PBDEX-M1-007t_G01 ::: 0.0516822 0.948318
gi|52970 PBDEX-M1-008t_E11 ::: 0.143557 0.856443
gi|53011 PBDEX-M1-010t_E01 ::: 0.0290983 0.970902
gi|53015 PBDEX-M1-010t_F03 ::: 0.111194 0.888806
gi|53021 PBDEX-M1-011t_A02 ::: 0.0474047 0.952595
gi|53023 PBDEX-M1-011t_B08 ::: 0.162249 0.837751
gi|53034 PBDEX-M1-012t_A08 ::: 0.0962279 0.903772
gi|53040 PBDEX-M1-012t_E10 ::: 0.0699613 0.930039
gi|53047 PBDEX-M1-013t_A07 ::: 0.330193 0.669807
gi|53054 PBDEX-M1-013t_E07 ::: 0.154785 0.845215
gi|53074 PBDEX-M1-016t_C08 ::: 0.182874 0.817126

gi|53082 PBDEX-M1-016t_F03 aminotransferase ::: 0.165874 0.834126
gi|53092 PBDEX-M1-017t_C03 sphingosine-1-phosphate lyase ::: 0.37507 0.62493
gi|53121 PBDEX-M1-018t_H05 ::: 0.381001 0.618999
gi|53125 PBDEX-M1-019t_A10 threonine synthase ::: 0.124495 0.875505
gi|53137 PBDEX-M1-020t_F05 ::: 0.141195 0.858805
gi|53174 PBDEX-M1-022t_H03 ::: 0.0793435 0.920656
gi|53184 PBDEX-M1-023t_E03 ::: 0.171863 0.828137
gi|53185 PBDEX-M1-023t_E11 ::: 0.0360761 0.963924
gi|53196 PBDEX-M1-024t_H11 ::: 0.124014 0.875986
gi|53206 PBDEX-M1-025t_D10 ::: 0.040661 0.959339
gi|53220 PBDEX-M1-026t_D09 ::: 0.352987 0.647013
gi|53239 PBDEX-M1-028t_A02 ::: 0.218417 0.781583
gi|53244 PBDEX-M1-028t_D04 eukaryotic translation initiation factor 2 gamma subunit :::
0.0741069 0.925893
gi|53252 PBDEX-M1-028t_H08 ::: 0.14977 0.85023
gi|53261 PBDEX-M1-029t_F05 ::: 0.0919827 0.908017
gi|53264 PBDEX-M1-029t_G05 ::: 0.134435 0.865565
gi|53272 PBDEX-M1-030t_A11 ::: 0.0236815 0.976318
gi|53301 PBDEX-M1-032t_F05 ::: 0.0546947 0.945305
gi|53304 PBDEX-M1-032t_H07 ::: 0.139372 0.860628
gi|53316 PBDEX-M1-033t_E08 ::: 0.0977269 0.902273
gi|53318 PBDEX-M1-033t_G03 ::: 0.0906487 0.909351
gi|53333 PBDEX-M1-034t_E10 ::: 0.0705809 0.929419
gi|53346 PBDEX-M1-035t_A11 ::: 0.426502 0.573498
gi|53347 PBDEX-M1-035t_B02 ::: 0.106761 0.893239
gi|53359 PBDEX-M1-035t_H05 DEAD/DEAH box RNA helicase ::: 0.181594 0.818406
gi|53362 PBDEX-M1-036t_A09 ::: 0.00867991 0.99132
gi|53370 PBDEX-M1-036t_E04 ::: 0.29041 0.70959
gi|53373 PBDEX-M1-036t_G01 ::: 0.0392853 0.960715
gi|53376 PBDEX-M1-036t_H04 ::: 0.0575183 0.942482
gi|53380 PBDEX-M1-037t_A06 ::: 0.27301 0.72699
gi|53382 PBDEX-M1-037t_B04 ::: 0.0213501 0.97865
gi|53391 PBDEX-M1-037t_F12 ::: 0.0432497 0.95675
gi|53400 PBDEX-M1-038t_F02 ::: 0.477868 0.522132
gi|53417 PBDEX-M1-039t_G01 ::: 0.0581701 0.94183
gi|53432 PBDEX-M1-041t_A10 PEROXISOMAL MEMBRANE PROTEIN PER10 ::: 0.0308763 0.969124
gi|53451 PBDEX-M1-042t_C09 ::: 0.13609 0.86391
gi|53457 PBDEX-M1-042t_E08 ::: 0.0779298 0.92207
gi|53462 PBDEX-M1-042t_G10 ::: 0.187491 0.812509
gi|53464 PBDEX-M1-043t_A07 ::: 0.0819837 0.918016
gi|53471 PBDEX-M1-043t_D06 ::: 0.061177 0.938823
gi|53486 PBDEX-M1-045t_C03 ::: 0.0579027 0.942097
gi|53501 PBDEX-M1-046t_B09 ::: 0.0422756 0.957724
gi|53520 PBDEX-M1-047t_B06 ::: 0.0329706 0.967029
gi|53524 PBDEX-M1-047t_F02 ::: 0.311373 0.688627
gi|53540 PBDEX-M1-048t_C11 ::: 0.120362 0.879638
gi|53548 PBDEX-M1-048t_F03 ::: 0.162054 0.837946
gi|53553 PBDEX-M1-050t_B09 [NID] ::: 0.03828 0.96172
gi|53557 PBDEX-M1-050t_D11 [NID] ::: 0.0453929 0.954607
gi|53619 PBDEX-M1-054t_C05 [NID] ::: 0.131753 0.868247
gi|53644 PBDEX-M1-056t_C06 [NID] ::: 0.370253 0.629747
gi|53651 PBDEX-M1-056t_E08 [NID] ::: 0.229943 0.770057
gi|53653 PBDEX-M1-056t_F06 [NID] ::: 0.365906 0.634094
gi|53668 PBDEX-M1-092t_F01 farnesyltransferase beta subunit protein ::: 0.0679019 0.932098
gi|53675 PBDEX-M1-093t_B05 ::: 0.0281559 0.971844
gi|53677 PBDEX-M1-093t_C06 ::: 0.200033 0.799967
gi|53678 PBDEX-M1-093t_D03 ::: 0.131816 0.868184
gi|53688 PBDEX-M1-094t_A08 ::: 0.0274098 0.97259
gi|53701 PBDEX-M1-094t_G03 ::: 0.0874413 0.912559
gi|53719 PBDEX-M1-096t_A06 ::: 0.15994 0.84006
gi|53729 PBDEX-M1-096t_F04 ::: 0.192045 0.807955
gi|53731 PBDEX-M1-096t_F12 Kynureninase ::: 0.181682 0.818318
gi|53765 PBDEX-M1-098t_G06 ::: 0.086271 0.913729
gi|53772 PBDEX-M1-099t_D11 ::: 0.206134 0.793866
gi|53776 PBDEX-M1-099t_H03 ::: 0.212867 0.787133
gi|53777 PBDEX-M1-100t_A04 ::: 0.0884926 0.911507
gi|53785 PBDEX-M1-100t_E10 Hypothetical ORF ::: 0.330698 0.669302
gi|53789 PBDEX-M1-100t_F09 isoamyl acetate hydrolytic enzyme homolog ::: 0.0830263 0.916974
gi|53801 PBDEX-M1-101t_C06 nuclease O ::: 0.146518 0.853482
gi|53807 PBDEX-M1-101t_G06 ::: 0.132139 0.867861
gi|53808 PBDEX-M1-101t_G09 ::: 0.0835971 0.916403
gi|53817 PBDEX-M1-102t_C10 ::: 0.163764 0.836236
gi|53820 PBDEX-M1-102t_E05 ::: 0.0558216 0.944178
gi|53822 PBDEX-M1-102t_E08 ::: 0.0782117 0.921788
gi|53828 PBDEX-M1-102t_H06 aromatic amino acid aminotransferase ::: 0.169598 0.830402
gi|53830 PBDEX-M1-103t_B04 ::: 0.162574 0.837426
gi|53836 PBDEX-M1-103t_G04 ::: 0.306556 0.693444
gi|53844 PBDEX-M1-104t_B07 ::: 0.0394107 0.960589

gi|53854 PBDEX-M1-105t_F10 Glutamine synthetase ::: 0.035741 0.964259
gi|53859 PBDEX-M1-106t_B04 ::: 0.302928 0.697072
gi|53864 PBDEX-M1-106t_F05 ::: 0.141081 0.858919
gi|53868 PBDEX-Y1-001t_C05 ::: 0.0565847 0.943415
gi|53871 PBDEX-Y1-001t_E09 ::: 0.142757 0.857243
gi|53875 PBDEX-Y1-001t_F05 ::: 0.291797 0.708203
gi|53878 PBDEX-Y1-001t_H07 ::: 0.240614 0.759386
gi|53890 PBDEX-Y1-002t_E07 ::: 0.0218535 0.978146
gi|53894 PBDEX-Y1-002t_H09 ::: 0.0074746 0.992525
gi|53908 PBDEX-Y1-004t_C09 ::: 0.185303 0.814697
gi|53917 PBDEX-Y1-004t_H03 N-acetyl-beta-glucosaminidase ::: 0.415265 0.584735
gi|53928 PBDEX-Y1-005t_F04 ::: 0.121868 0.878132
gi|53931 PBDEX-Y1-005t_G05 ::: 0.279116 0.720884
gi|53932 PBDEX-Y1-005t_G09 ::: 0.136861 0.863139
gi|53935 PBDEX-Y1-006t_C04 ::: 0.0831944 0.916806
gi|53950 PBDEX-Y1-007t_B11 GTP cyclohydrolase II ::: 0.209264 0.790736
gi|53953 PBDEX-Y1-007t_H04 ::: 0.0113979 0.988602
gi|53960 PBDEX-Y1-008t_D06 serine/threonine protein phosphatase ::: 0.159874 0.840126
gi|53963 PBDEX-Y1-008t_F03 ::: 0.136148 0.863852
gi|53967 PBDEX-Y1-008t_G08 ::: 0.0171328 0.982867
gi|53970 PBDEX-Y1-008t_H11 ::: 0.0674597 0.93254
gi|53976 PBDEX-Y1-009t_E07 ::: 0.16254 0.83746
gi|53985 PBDEX-Y1-010t_F01 ::: 0.289627 0.710373
gi|53991 PBDEX-Y1-011t_C04 ::: 0.0501876 0.949812
gi|53995 PBDEX-Y1-011t_E07 ::: 0.0288271 0.971173
gi|53996 PBDEX-Y1-011t_F02 ::: 0.191656 0.808344
gi|54004 PBDEX-Y1-012t_B07 Phenylalanyl-tRNA synthetase beta chain ::: 0.266661 0.733339
gi|54007 PBDEX-Y1-012t_C06 ::: 0.0175383 0.982462
gi|54011 PBDEX-Y1-012t_H02 ::: 0.340265 0.659735
gi|54013 PBDEX-Y1-013t_A06 ::: 0.0410214 0.958979
gi|54015 PBDEX-Y1-013t_C04 ::: 0.0276507 0.972349
gi|54021 PBDEX-Y1-013t_E07 ::: 0.189344 0.810656
gi|54030 PBDEX-Y1-014t_D06 ::: 0.237799 0.762201
gi|54034 PBDEX-Y1-014t_F09 MUS38 ::: 0.340756 0.659244
gi|54035 PBDEX-Y1-014t_G05 ::: 0.445471 0.554529
gi|54036 PBDEX-Y1-014t_H10 ::: 0.0558873 0.944113
gi|54040 PBDEX-Y1-015t_B06 putative sucrose carrier [Schizosaccharomyces pombe] ::: 0.053472
0.946528
gi|54041 PBDEX-Y1-015t_E04 ::: 0.447253 0.552747
gi|54044 PBDEX-Y1-015t_G05 ::: 0.0283647 0.971635
gi|54046 PBDEX-Y1-016t_A05 ::: 0.0303235 0.969676
gi|54071 PBDEX-Y1-017t_G09 ::: 0.107913 0.892087
gi|54072 PBDEX-Y1-017t_G10 ::: 0.0360991 0.963901
gi|54085 PBDEX-Y1-019t_C03 ::: 0.166702 0.833298
gi|54087 PBDEX-Y1-019t_C12 ::: 0.128797 0.871203
gi|54103 PBDEX-Y1-021t_C01 ::: 0.184484 0.815516
gi|54104 PBDEX-Y1-021t_C04 ::: 0.0705526 0.929447
gi|54110 PBDEX-Y1-021t_G09 ::: 0.087468 0.912532
gi|54115 PBDEX-Y1-022t_B09 ::: 0.14309 0.85691
gi|54130 PBDEX-Y1-023t_D12 ::: 0.0686851 0.931315
gi|54146 PBDEX-Y1-024t_E06 ::: 0.21444 0.78556
gi|54147 PBDEX-Y1-024t_E08 ::: 0.050668 0.949332
gi|54193 PBDEX-Y1-028t_D04 ::: 0.127023 0.872977
gi|54201 PBDEX-Y1-029t_A03 cell division control protein 5; Myb family DNA-binding
[Schizosaccharomyces pombe] ::: 0.0398191 0.960181
gi|54226 PBDEX-Y1-030t_F01 ::: 0.17467 0.82533
gi|54234 PBDEX-Y1-031t_C04 ::: 0.0343925 0.965607
gi|54236 PBDEX-Y1-031t_F01 ::: 0.085979 0.914021
gi|54238 PBDEX-Y1-032t_A11 ::: 0.0431884 0.956812
gi|54244 PBDEX-Y1-032t_D10 ::: 0.393298 0.606702
gi|54249 PBDEX-Y1-032t_G08 ::: 0.344454 0.655546
gi|54255 PBDEX-Y1-033t_C01 ::: 0.372359 0.627641
gi|54259 PBDEX-Y1-033t_D08 ::: 0.0804024 0.919598
gi|54260 PBDEX-Y1-033t_D09 ::: 0.119339 0.880661
gi|54262 PBDEX-Y1-033t_F08 ::: 0.146815 0.853185
gi|54265 PBDEX-Y1-033t_H05 ::: 0.488503 0.511497
gi|54266 PBDEX-Y1-033t_H08 hypothetical trp-asp repeats containing protein ::: 0.254528
0.745472
gi|54267 PBDEX-Y1-034t_A02 ::: 0.385495 0.614505
gi|54268 PBDEX-Y1-034t_B04 ::: 0.282335 0.717665
gi|54269 PBDEX-Y1-034t_B09 ::: 0.0982691 0.901731
gi|54276 PBDEX-Y1-034t_D12 ::: 0.299887 0.700113
gi|54277 PBDEX-Y1-034t_F08 ::: 0.0881814 0.911819
gi|54281 PBDEX-Y1-035t_A03 ::: 0.0946624 0.905338
gi|54283 PBDEX-Y1-035t_A08 D-mandelate dehydrogenase ::: 0.357816 0.642184
gi|54291 PBDEX-Y1-035t_F10 ::: 0.0429383 0.957062
gi|54295 PBDEX-Y1-036t_A05 ::: 0.134787 0.865213
gi|54297 PBDEX-Y1-036t_B05 SEC23 ::: 0.0559199 0.94408

gi|54305 PBDEX-Y1-037t_C01 ::: 0.0409547 0.959045
gi|54307 PBDEX-Y1-037t_C11 ::: 0.0427731 0.957227
gi|54313 PBDEX-Y1-037t_F05 ::: 0.263625 0.736375
gi|54314 PBDEX-Y1-037t_F12 ::: 0.161787 0.838213
gi|54317 PBDEX-Y1-037t_H11 histone h2a ::: 0.183596 0.816404
gi|54319 PBDEX-Y1-038t_B05 ::: 0.125066 0.874934
gi|54324 PBDEX-Y1-039t_B12 ::: 0.116164 0.883836
gi|54326 PBDEX-Y1-039t_D05 ::: 0.0811817 0.918818
gi|54330 PBDEX-Y1-039t_F05 ::: 0.282345 0.717655
gi|54341 PBDEX-Y1-040t_F11 ::: 0.0147751 0.985225
gi|54348 PBDEX-Y1-041t_A10 ::: 0.357801 0.642199
gi|54350 PBDEX-Y1-041t_D05 ::: 0.0199764 0.980024
gi|54351 PBDEX-Y1-041t_D09 ::: 0.0989766 0.901023
gi|54353 PBDEX-Y1-041t_F04 ::: 0.0625169 0.937483
gi|54355 PBDEX-Y1-041t_F10 ::: 0.042566 0.957434
gi|54357 PBDEX-Y1-041t_H05 ::: 0.146792 0.853208
gi|54358 PBDEX-Y1-041t_H06 ::: 0.29078 0.70922
gi|54365 PBDEX-Y1-042t_B10 ::: 0.018159 0.981841
gi|54374 PBDEX-Y1-043t_B10 ::: 0.0240031 0.975997
gi|54379 PBDEX-Y1-043t_D12 ::: 0.091872 0.908128
gi|54384 PBDEX-Y1-043t_G11 ::: 0.37439 0.62561
gi|54386 PBDEX-Y1-043t_H06 ::: 0.173155 0.826845
gi|54387 PBDEX-Y1-043t_H08 Conserved hypothetical protein ::: 0.0520902 0.94791
gi|54388 PBDEX-Y1-044t_A01 ::: 0.315661 0.684339
gi|54389 PBDEX-Y1-044t_A04 conserved hypothetical protein ::: 0.106648 0.893352
gi|54396 PBDEX-Y1-044t_D02 ::: 0.144007 0.855993
gi|54402 PBDEX-Y1-044t_H02 ::: 0.201294 0.798706
gi|54435 PBDFM-Y1-206t_G01 ::: 0.236858 0.763142
gi|54438 PBDHV-M1-051t_D09 ::: 0.0701209 0.929879
gi|54444 PBDHV-M1-052t_E11 ::: 0.0665946 0.933405
gi|54459 PBDIP-M1-025t_D06 ::: 0.128962 0.871038
gi|54475 PBDIP-M1-026t_D08 carnitine acetyl transferase ::: 0.228745 0.771255
gi|54478 PBDIP-M1-026t_E11 ::: 0.0644289 0.935571
gi|54483 PBDLP-M1-029t_B08 ::: 0.111744 0.888256
gi|54486 PBDLP-M1-029t_F01 ::: 0.0997333 0.900267
gi|54509 PBDMF-Y1-007t_A08 ::: 0.0669242 0.933076
gi|54511 PBDMF-Y1-007t_B05 ::: 0.248379 0.751621
gi|54517 PBDMF-Y1-007t_F03 ::: 0.160959 0.839041
gi|54523 PBDMF-Y1-008t_C09 ::: 0.0247787 0.975221
gi|54526 PBDMF-Y1-008t_F10 ::: 0.0804596 0.91954
gi|54527 PBDMF-Y1-008t_G07 ::: 0.113936 0.886064
gi|54528 PBDMF-Y1-008t_G12 ::: 0.149954 0.850046
gi|54536 PBDMO-Y1-009t_E12 ::: 0.094356 0.905644
gi|54571 PBDMP-M1-033t_B05 ::: 0.492826 0.507174
gi|54590 PBDMP-M1-034t_F05 ::: 0.111171 0.888829
gi|54596 PBDMP-Y1-035t_E04 ::: 0.055986 0.944014
gi|54634 PBDMP-Y1-040t_G06 ::: 0.0348368 0.965163
gi|54640 PBDMP-Y1-168t_H04 ::: 0.0672262 0.932774
gi|54652 PBDMS-Y1-003t_C05 ::: 0.424441 0.575559
gi|54657 PBDMS-Y1-003t_F09 ::: 0.112074 0.887926
gi|54671 PBDPA-Y1-013t_D03 ::: 0.0720541 0.927946
gi|54693 PBDRS-M1-027t_D05 ::: 0.159101 0.840899
gi|54708 PBDRS-M1-028t_D06 uroporphyrin methyltransferase - fission yeast (Schizosaccharomyces pombe) ::: 0.0182407 0.981759
gi|54720 PBDRV-M1-041t_A04 ::: 0.0664864 0.933514
gi|54737 PBDRV-M1-041t_H05 ::: 0.0496701 0.95033
gi|54747 PBDRV-M1-042t_D03 ::: 0.0463455 0.953655
gi|54769 PBDRV-M1-044t_B05 ::: 0.144564 0.855436
gi|54791 PBDRV-Y1-041t_B06 putative transcriptional activator ::: 0.0948114 0.905189
gi|54797 PBDRV-Y1-041t_F02 ::: 0.152102 0.847898
gi|54803 PBDRV-Y1-041t_H05 ::: 0.12553 0.87447
gi|54806 PBDRV-Y1-042t_A12 ::: 0.0403001 0.9597
gi|54808 PBDRV-Y1-042t_B10 ::: 0.150625 0.849375
gi|54819 PBDRV-Y1-042t_H08 ::: 0.046632 0.953368
gi|54856 PBDRV-Y1-101t_D09 ::: 0.0224812 0.977519
gi|54875 PBDSP-M1-046t_E06 ::: 0.0892512 0.910749
gi|54883 PBDSP-M1-047t_C10 ribosomal protein L4, mitochondrial ::: 0.36148 0.63852
gi|54900 PBDSP-M1-048t_H10 ::: 0.189205 0.810795
gi|54904 PBDSP-M1-049t_E07 ::: 0.332062 0.667938
gi|54930 PBDSP-Y1-046t_F07 ::: 0.282233 0.717767
gi|54957 PBDSP-Y1-048t_G07 ::: 0.169259 0.830741
gi|54978 PBDSP-Y1-049t_G06 ::: 0.283063 0.716937
gi|54999 PBGAC-M1-016t_E11 ::: 0.0828664 0.917134
gi|55003 PBGCB-M1-035t_B02 ::: 0.388105 0.611895
gi|55020 PBGCB-M1-036t_F01 ::: 0.206427 0.793573
gi|55048 PBGCM-M1-018t_G02 ::: 0.339148 0.660852
gi|55051 PBGCS-M1-019t_C08 ::: 0.0624295 0.93757
gi|55054 PBGCS-M1-019t_G05 ::: 0.0322959 0.967704

gi|55130 PBGEV-M1-500t_B05 pop-interacting protein 1 ::: 0.247659 0.752341
gi|55143 PBGEV-M1-502t_D08 ::: 0.0537469 0.946253
gi|55148 PBGEV-M1-502t_H12 ::: 0.0688571 0.931143
gi|55157 PBGEV-Y1-503t_D04 ::: 0.11928 0.88072
gi|55180 PBGEX-M1-063t_B03 ::: 0.487511 0.512489
gi|55183 PBGEX-M1-063t_B06 type I transmembrane protein, component of COPII-coated, ER-derived transport vesicles; Emp24p ::: 0.110747 0.889253
gi|55188 PBGEX-M1-063t_F04 ::: 0.0946845 0.905316
gi|55196 PBGEX-M1-064t_C03 Homoserine O-acetyltransferase (Homoserine O-trans-acetylase) ::: 0.2124 0.7876
gi|55201 PBGEX-M1-064t_D08 ::: 0.162878 0.837122
gi|55205 PBGEX-M1-064t_E06 ::: 0.228916 0.771084
gi|55211 PBGEX-M1-064t_G11 Ydr399p [Candida glabrata] ::: 0.0180687 0.981931
gi|55251 PBGEX-M1-068t_H09 ::: 0.038485 0.961515
gi|55256 PBGEX-M1-069t_C04 ::: 0.240171 0.759829
gi|55265 PBGEX-M1-070t_B07 DigA protein [Aspergillus nidulans] ::: 0.0266882 0.973312
gi|55275 PBGEX-M1-071t_A06 ::: 0.344837 0.655163
gi|55280 PBGEX-M1-071t_C07 ::: 0.0640127 0.935987
gi|55321 PBGEX-M1-074t_D05 ::: 0.0668172 0.933183
gi|55331 PBGEX-M1-075t_G07 [NC] ::: 0.158602 0.841398
gi|55347 PBGEX-Y1-064t_A03 [NID] ::: 0.0842828 0.915717
gi|55348 PBGEX-Y1-064t_A11 ::: 0.0179783 0.982022
gi|55349 PBGEX-Y1-064t_B07 ::: 0.481939 0.518061
gi|55359 PBGEX-Y1-065t_B02 ::: 0.331573 0.668427
gi|55360 PBGEX-Y1-065t_B08 ::: 0.140461 0.859539
gi|55366 PBGEX-Y1-065t_G02 ::: 0.0445894 0.955411
gi|55379 PBGEX-Y1-068t_A12 ::: 0.163644 0.836356
gi|55381 PBGEX-Y1-068t_C06 ::: 0.106735 0.893265
gi|55385 PBGEX-Y1-068t_E06 ::: 0.408055 0.591945
gi|55388 PBGEX-Y1-068t_G11 ::: 0.159821 0.840179
gi|55389 PBGEX-Y1-069t_A06 ::: 0.36588 0.63412
gi|55395 PBGEX-Y1-069t_C05 putative retrotransposon ::: 0.122408 0.877592
gi|55398 PBGEX-Y1-069t_G06 ::: 0.014368 0.985632
gi|55429 PBGEX-Y1-072t_E12 ::: 0.0135636 0.986436
gi|55438 PBGEX-Y1-073t_A06 ::: 0.153679 0.846321
gi|55440 PBGEX-Y1-073t_B10 hypothetical zf-C3HC4 zinc finger protein ::: 0.0876201 0.91238
gi|55442 PBGEX-Y1-073t_D03 ::: 0.023663 0.976337
gi|55443 PBGEX-Y1-073t_D08 ::: 0.0727675 0.927232
gi|55457 PBGEX-Y1-074t_F03 ::: 0.465328 0.534672
gi|55459 PBGEX-Y1-074t_G02 ::: 0.0497069 0.950293
gi|55467 PBGEX-Y1-075t_G03 ::: 0.014381 0.985619
gi|55473 PBGEX-Y1-076t_C05 ::: 0.324808 0.675192
gi|55482 PBGEX-Y1-077t_B02 ::: 0.0636114 0.936389
gi|55491 PBGEX-Y1-078t_A04 hypothetical protein ::: 0.172279 0.827721
gi|55496 PBGEX-Y1-078t_C07 ::: 0.0711974 0.928803
gi|55507 PBGEX-Y1-079t_B07 ::: 0.0225526 0.977447
gi|55516 PBGEX-Y1-079t_H06 ::: 0.02957 0.97043
gi|55523 PBGEX-Y1-080t_D01 ::: 0.0721778 0.927822
gi|55527 PBGEX-Y1-080t_F08 ::: 0.0154138 0.984586
gi|55542 PBGEX-Y1-081t_G01 [Nid] ::: 0.0253428 0.974657
gi|55543 PBGEX-Y1-082t_A08 ::: 0.331246 0.668754
gi|55563 PBGEX-Y1-083t_D08 ::: 0.0206531 0.979347
gi|55571 PBGEX-Y1-083t_H08 ::: 0.114153 0.885847
gi|55574 PBGEX-Y1-084t_B01 ::: 0.3157 0.6843
gi|55576 PBGEX-Y1-084t_C08 ::: 0.0422202 0.95778
gi|55584 PBGEX-Y1-084t_E10 ::: 0.266024 0.733976
gi|55594 PBGEX-Y1-085t_C11 protein kinase C-like protein ::: 0.0755595 0.924441
gi|55604 PBGEX-Y1-086t_A11 ::: 0.0759334 0.924067
gi|55610 PBGEX-Y1-086t_F02 ::: 0.0980887 0.901911
gi|55611 PBGEX-Y1-086t_F09 ::: 0.268444 0.731556
gi|55619 PBGEX-Y1-087t_B03 ::: 0.315977 0.684023
gi|55631 PBGEX-Y1-087t_F02 ::: 0.40904 0.59096
gi|55633 PBGEX-Y1-087t_F11 ::: 0.159926 0.840074
gi|55641 PBGEX-Y1-088t_B12 ::: 0.343705 0.656295
gi|55647 PBGEX-Y1-088t_D09 ::: 0.0145595 0.985441
gi|55657 PBGEX-Y1-089t_F08 ::: 0.0336808 0.966319
gi|55661 PBGEX-Y1-090t_A08 ::: 0.0269741 0.973026
gi|55662 PBGEX-Y1-090t_B02 ::: 0.138633 0.861367
gi|55666 PBGEX-Y1-090t_E11 ::: 0.400047 0.599953
gi|55669 PBGEX-Y1-090t_G12 ::: 0.280907 0.719093
gi|55670 PBGEX-Y1-090t_H01 ::: 0.0717605 0.92824
gi|55671 PBGEX-Y1-090t_H11 ::: 0.0660738 0.933926
gi|55673 PBGEX-Y1-091t_C04 ::: 0.207251 0.792749
gi|55675 PBGEX-Y1-091t_D04 ::: 0.0165489 0.983451
gi|55677 PBGEX-Y1-091t_E03 ::: 0.10779 0.89221
gi|55686 PBGEX-Y1-093t_A06 ::: 0.0525198 0.94748
gi|55687 PBGEX-Y1-093t_A10 ::: 0.0399747 0.960025
gi|55696 PBGEX-Y1-093t_G04 ::: 0.256281 0.743719

gi|55697 PBGEX-Y1-093t_H02 ::: 0.301671 0.698329
gi|55699 PBGEX-Y1-093t_H07 ::: 0.026959 0.973041
gi|55700 PBGEX-Y1-094t_A06 ::: 0.0792973 0.920703
gi|55708 PBGEX-Y1-095t_D03 ::: 0.0349156 0.965084
gi|55709 PBGEX-Y1-095t_D08 ::: 0.00662585 0.993374
gi|55710 PBGEX-Y1-095t_D10 ::: 0.37847 0.62153
gi|55722 PBGEX-Y1-096t_C09 ::: 0.0770471 0.922953
gi|55724 PBGEX-Y1-096t_E10 ::: 0.055247 0.944753
gi|55740 PBGEX-Y1-098t_H08 ::: 0.174767 0.825233
gi|55743 PBGEX-Y1-099t_D10 ::: 0.104902 0.895098
gi|55746 PBGEX-Y1-099t_F11 ::: 0.0722655 0.927735
gi|55762 PBGEX-Y1-101t_A02 putative cytochrome p450 ::: 0.0438135 0.956186
gi|55768 PBGEX-Y1-101t_E06 ::: 0.0748127 0.925187
gi|55772 PBGEX-Y1-102t_A07 ::: 0.0713602 0.92864
gi|55775 PBGEX-Y1-102t_C08 ::: 0.187573 0.812427
gi|55777 PBGEX-Y1-102t_D08 ::: 0.0715704 0.92843
gi|55786 PBGEX-Y1-103t_E06 ::: 0.07075 0.92925
gi|55787 PBGEX-Y1-103t_F03 ::: 0.203079 0.796921
gi|55797 PBGEX-Y1-104t_H05 ::: 0.16689 0.83311
gi|55802 PBGEX-Y1-105t_C11 ::: 0.0194503 0.98055
gi|55803 PBGEX-Y1-105t_E01 ::: 0.181812 0.818188
gi|55805 PBGEX-Y1-105t_E12 ::: 0.0986572 0.901343
gi|55813 PBGEX-Y1-106t_C08 ::: 0.094384 0.905616
gi|55814 PBGEX-Y1-106t_D05 ::: 0.185804 0.814196
gi|55819 PBGEX-Y1-108t_A09 ::: 0.0197247 0.980275
gi|55822 PBGEX-Y1-108t_E11 ::: 0.108711 0.891289
gi|55824 PBGEX-Y1-108t_G05 ::: 0.129868 0.870132
gi|55825 PBGEX-Y1-108t_H11 ::: 0.0863648 0.913635
gi|55847 PBGEX-Y1-112t_B01 ::: 0.0957718 0.904228
gi|55851 PBGEX-Y1-112t_F10 ::: 0.151918 0.848082
gi|55853 PBGEX-Y1-112t_G02 ::: 0.0399802 0.96002
gi|55856 PBGEX-Y1-113t_A05 ::: 0.025167 0.974833
gi|55858 PBGEX-Y1-113t_C02 ::: 0.202036 0.797964
gi|55859 PBGEX-Y1-113t_G07 ::: 0.0570549 0.942945
gi|55860 PBGEX-Y1-113t_H04 ::: 0.25356 0.74644
gi|55861 PBGEX-Y1-114t_B08 ::: 0.264642 0.735358
gi|55868 PBGEX-Y1-114t_H09 ::: 0.1367 0.8633
gi|55869 PBGEX-Y1-115t_A05 ::: 0.235618 0.764382
gi|55872 PBGEX-Y1-115t_B07 Probable cation-transporting ATPase ::: 0.102465 0.897535
gi|55874 PBGEX-Y1-115t_C08 ::: 0.311695 0.688305
gi|55881 PBGEX-Y1-115t_H01 ::: 0.0954243 0.904576
gi|55882 PBGEX-Y1-115t_H02 ::: 0.428398 0.571602
gi|55884 PBGEX-Y1-117t_A04 ::: 0.0482872 0.951713
gi|55886 PBGEX-Y1-117t_C06 ::: 0.374012 0.625988
gi|55889 PBGEX-Y1-117t_E02 ::: 0.166135 0.833865
gi|55893 PBGEX-Y1-117t_G12 ::: 0.127402 0.872598
gi|55894 PBGEX-Y1-117t_H06 ::: 0.262356 0.737644
gi|55897 PBGEX-Y1-118t_G09 ::: 0.0845905 0.91541
gi|55898 PBGEX-Y1-118t_H05 ::: 0.076551 0.923449
gi|55903 PBGEX-Y1-119t_B04 meiotic recombination protein ::: 0.0605009 0.939499
gi|55904 PBGEX-Y1-119t_G08 NADH-UBIQUINONE OXIDOREDUCTASE 9.5 KD SUBUNIT ::: 0.358297 0.641703
gi|55909 PBGEX-Y1-120t_D06 ::: 0.274658 0.725342
gi|55912 PBGEX-Y1-120t_F12 ::: 0.0418086 0.958191
gi|55918 PBGEX-Y1-121t_D01 ::: 0.311583 0.688417
gi|55921 PBGEX-Y1-121t_D08 ::: 0.0697089 0.930291
gi|55926 PBGEX-Y1-121t_H07 Multicopy suppressor of ypt6 null mutation ::: 0.0691174 0.930883
gi|55933 PBGEX-Y1-124t_C08 cytochrome-c oxidase chain VIIc-like protein ::: 0.401529 0.598471
gi|55944 PBGGO-M1-037t_B06 ::: 0.0187217 0.981278
gi|55965 PBGJO-M1-039t_F09 Hemolysin ::: 0.0593548 0.940645
gi|55982 PBGJP-M1-407t_A09 golgi-specific brefeldin A resistance factor 1 ::: 0.125648
0.874352
gi|56017 PBGMG-M1-420t_C05 ::: 0.0170926 0.982907
gi|56023 PBGMG-Y1-018t_E10 ::: 0.265021 0.734979
gi|56036 PBGRJ-M1-422t_H05 hypothetical protein ::: 0.453506 0.546494
gi|56058 PBGEX-Y1-062t_D12 ::: 0.12415 0.87585
gi|56059 PBGEX-Y1-062t_E02 ::: 0.0955631 0.904437
gi|56066 PBGEX-Y1-063t_B09 ::: 0.0216992 0.978301
gi|56069 PBGEX-Y1-063t_F06 ::: 0.104805 0.895195
gi|56074 PBGEX-Y1-063t_H08 ::: 0.405159 0.594841
gi|56078 PBGEX-Y1-067t_B03 Member of family of mitochondrial carrier proteins; Pet8 :::
0.310306 0.689694
gi|56088 PBGEX-Y1-070t_A07 ::: 0.058764 0.941236
gi|56089 PBGEX-Y1-070t_A11 ::: 0.13511 0.86489
gi|56095 PBGEX-Y1-070t_D05 ::: 0.115536 0.884464
gi|56098 PBGEX-Y1-070t_F10 hypothetical protein ::: 0.0830725 0.916928
gi|56101 PBGEX-Y1-070t_H01 RAS-LIKE GTP-BINDING PROTEIN ::: 0.085115 0.914885
gi|56105 Contig C1162_3 monooxigenase ::: 0.0235329 0.976467
gi|56106 Contig C1162_4 monooxigenase ::: 0.17838 0.82162

gi|56108 PBDEX-Y1-045t_B01 Cytochrome b5 ::: 0.0776906 0.922309
gi|56109 PBDEX-Y1-045t_C07 [NID] ::: 0.109408 0.890592
gi|56115 PBDEX-M1-107t_D12 [nid] ::: 0.11607 0.88393
gi|56132 PBDEX-M1-108t_E12 hypothetical protein ::: 0.258728 0.741272
gi|56136 PBDEX-M1-108t_G03 [nid] ::: 0.115709 0.884291