

**UNIVERSIDADE DE SÃO PAULO**  
**Faculdade de Medicina de Ribeirão Preto**  
**Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto**

**LARIZA LAURA DE OLIVEIRA**

**SISTEMA DE APOIO AO ESTUDO DE PROTEÍNAS ATRAVÉS DE TÉCNICAS DE  
INTELIGÊNCIA ARTIFICIAL**

**RIBEIRÃO PRETO**

**2008**

UNIVERSIDADE DE SÃO PAULO  
Faculdade de Medicina de Ribeirão Preto  
Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto

**LARIZA LAURA DE OLIVEIRA**

**SISTEMA DE APOIO AO ESTUDO DE PROTEÍNAS ATRAVÉS DE TÉCNICAS DE  
INTELIGÊNCIA ARTIFICIAL**

Monografia apresentada à Faculdade Medicina de Ribeirão Preto e à Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto ambas da Universidade de São Paulo, como requisito parcial para obtenção do título de Bacharel em Informática Biomédica.

**ORIENTADOR: Prof. Dr. RENATO TINÓS**  
**CO-ORIENTADORA: Prof. Dra. SILVANA GIULIATTI**

**RIBEIRÃO PRETO**

**2008**

UNIVERSIDADE DE SÃO PAULO  
Faculdade de Medicina de Ribeirão Preto  
Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto

**SISTEMA DE APOIO AO ESTUDO DE PROTEÍNAS ATRAVÉS DE TÉCNICAS DE  
INTELIGÊNCIA ARTIFICIAL**

**LARIZA LAURA DE OLIVEIRA**

Monografia apresentada à Faculdade Medicina de Ribeirão Preto e à Faculdade de Filosofia Ciências e Letras de Ribeirão Preto ambas da Universidade de São Paulo, como requisito parcial para obtenção do título de Bacharel em Informática Biomédica.

Aprovado em: \_\_\_\_\_  
Conceito: \_\_\_\_\_

**BANCA EXAMINADORA**

---

Prof. Dr. Renato Tinós  
Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto  
Presidente

---

Profa. Dra. Silvana Giuliatti  
Faculdade de Medicina de Ribeirão Preto  
Membro Titular

---

Prof. Dr. Fernando Luís Barroso da Silva  
Faculdade de Ciências Farmacêuticas de Ribeirão Preto  
Membro Titular

Aos meus pais, Walter e Elizabeth

## **AGRADECIMENTOS**

Agradeço aos meus pais e a minha irmã Nadia pelo carinho durante todo o período de graduação.

Ao meu namorado e melhor amigo Hugo pelo apoio, incentivo e companheirismo.

A todos os amigos do Laboratório de Informática em Saúde (LIS), entre eles: Daniane, Juliana, Iuliana, Luís Henrique, Helder, Gustavo, Vinícius e Ricardo.

A todos os meus amigos, entre eles: Aline, Tiago e Valéria, pelo apoio.

Aos professores Renato Tinós e Silvana Giuliatti pela orientação, atenção e incentivo durante a realização deste projeto.

## RESUMO

As proteínas são moléculas essenciais para a maioria dos processos biológicos, desse modo, seu estudo é de extrema importância. Diversas técnicas são empregadas na análise de seqüências protéicas, investigando os vários níveis estruturais existentes. As proteínas são classificadas em famílias de acordo com características funcionais e estruturais, para esse fim diversas ferramentas são utilizadas. O presente trabalho envolve a implementação de um sistema de apoio ao estudo de proteínas em seus diferentes níveis estruturais. Esse apoio será provido mediante a implementação de ferramentas e sua integração com alguns softwares já existentes. O sistema conterà três módulos principais: um para classificação de proteínas, onde será implementada uma Rede Neural do tipo *Perceptron* Multicamadas, um para alinhamento e outro para visualização de estruturas tridimensionais, os quais utilizarão ferramentas disponíveis na Internet. Por meio de suas ferramentas, o sistema fornecerá subsídios para o estudo e classificação de novas proteínas.

**Palavras chaves:** Proteínas, Redes Neurais, Alinhamento.

## LISTA DE FIGURAS

FIGURA 1: EXEMPLO DE ARQUIVO NO FORMATO PDB.....	10
FIGURA 2: ALGORITMO <i>BACKPROPAGATION</i> . ....	12
FIGURA 3: ILUSTRAÇÃO DE UMA MLP. ....	13
FIGURA 4: SEQUÊNCIA NO FORMATO FASTA.....	14
FIGURA 5: INTERFACE INICIAL.....	16
FIGURA 6: INTERFACE DOS MÓDULOS DE VISUALIZAÇÃO E ALINHAMENTO – ABA 1: ALINHAMENTO. ....	17
FIGURA 7: INTERFACE PARA VISUALIZAÇÃO DAS SEQUÊNCIAS ALINHADAS. ....	18
FIGURA 8: INTERFACE PARA A VISUALIZAÇÃO DO ALINHAMENTO DE CADA SEQUÊNCIA.....	19
FIGURA 9: INTERFACE DOS MÓDULOS DE VISUALIZAÇÃO E ALINHAMENTO – ABA 2: VISUALIZAÇÃO. ....	20
FIGURA 10: VISUALIZAÇÃO FORNECIDA PELA FERRAMENTA Jmol.....	21
FIGURA 11: CLASSIFICAÇÃO DE PROTEÍNAS – ABA 1: TESTE. ....	22
FIGURA 12: CLASSIFICAÇÃO DE PROTEÍNAS – ABA 2: RESULTADOS.....	23
FIGURA 13: – TREINAR CLASSIFICADOR - ABA 1: DADOS.....	24
FIGURA 14: TREINAR CLASSIFICADOR – ABA 2: PARÂMETROS. ....	25
FIGURA 15: TREINAR CLASSIFICADOR – ABA 3: VALIDAÇÃO. ....	26
FIGURA 16: TREINAR CLASSIFICADOR – ABA 4: RESULTADOS.....	27
FIGURA 17: FOSFOLIPASE DE SERPENTE UTILIZADA NA REALIZAÇÃO DE ALINHAMENTO. ....	28
FIGURA 18: PDB: 1GMZ - PHOSPHOLIPASE A2 HOMOLOG 1 - CADEIAS A E B. ....	30
FIGURA 19: PDB: 1PC9 - BNSP-6- CADEIAS A E B. ....	31
FIGURA 20: PDB: 1QLL- PHOSPHOLIPASE A2- CADEIAS A E B. ....	31
FIGURA 21: PDB: 1PA0 - MYOTOXIC PHOSPHOLIPASE - CADEIAS A E B. ....	32
FIGURA 22: PDB: 2H8I - PHOSPHOLIPASE A2- CADEIAS A E B. ....	32
FIGURA 23: FREQUÊNCIA MÉDIA DOS AMINOÁCIDOS – CLASSE: L-AMINOÁCIDO OXIDASE.....	34
FIGURA 24: FREQUÊNCIA MÉDIA DOS AMINOÁCIDOS – CLASSE: FOSFOLIPASE. ....	35
FIGURA 25: FREQUÊNCIA MÉDIA DOS AMINOÁCIDOS – CLASSE: METALOPROTEINASE. ....	36
FIGURA 26: FREQUÊNCIA MÉDIA DOS AMINOÁCIDOS – CLASSE: SERINO PROTEASE.....	37
FIGURA 27: DIAGRAMA DE CASO DE USO. ....	71
FIGURA 28: DIAGRAMA DE SEQUÊNCIA: CLASSIFICAR SEQUÊNCIAS. ....	72
FIGURA 29: DIAGRAMA DE SEQUÊNCIA: VISUALIZAR ESTRUTURA PROTÉICA.....	72
FIGURA 30: DIAGRAMA DE SEQUÊNCIA: REALIZAR ALINHAMENTO. ....	73
FIGURA 31: DIAGRAMA DE CLASSES DO SISTEMA.....	74

## LISTA DE TABELAS

TABELA 1: SEQÜÊNCIAS COM ALINHAMENTO SIGNIFICANTE – BANCO DE DADOS <i>SWISS-PROT</i> .	28
TABELA 2: SEQÜÊNCIAS COM ALINHAMENTO SIGNIFICANTE – BANCO DE DADOS <i>PDB</i> .	30
TABELA 3: L-AMINOÁCIDO OXIDASE – AS 7 MAIORES FREQUÊNCIAS MÉDIAS.	35
TABELA 4: FOSFOLIPASE – AS 7 MAIORES FREQUÊNCIAS MÉDIAS.	36
TABELA 5: METALOPROTEINASE. – AS 7 MAIORES FREQUÊNCIAS MÉDIAS.	36
TABELA 6: SERINO PROTEASE. – AS 7 MAIORES FREQUÊNCIAS MÉDIAS.	37
TABELA 7: NÚMERO DE EXEMPLOS DE TREINAMENTO E TESTE.	38
TABELA 8: VARIAÇÃO DA TAXA DE APRENDIZADO.	39
TABELA 9: MATRIZ DE CONFUSÃO PARA TAXA IGUAL A 0.01.	39
TABELA 10: MATRIZ DE CONFUSÃO PARA TAXA IGUAL A 0.2.	39
TABELA 11: MATRIZ DE CONFUSÃO PARA TAXA IGUAL A 0.5 E 0.9.	40
TABELA 12: VARIAÇÃO DO NÚMERO DE NEURÔNIOS DA CAMADA OCULTA.	40
TABELA 13: MATRIZ DE CONFUSÃO PARA 20 NEURÔNIOS NA CAMADA OCULTA.	40
TABELA 14: MATRIZ DE CONFUSÃO PARA 50 NEURÔNIOS NA CAMADA OCULTA.	41
TABELA 15: MATRIZ DE CONFUSÃO PARA 100 NEURÔNIOS NA CAMADA OCULTA.	41
TABELA 16: ERRO DE CLASSIFICAÇÃO UTILIZANDO <i>10-FOLD CROSS-VALIDATION</i>	42
TABELA 17: MATRIZ DE CONFUSÃO PARA SEMENTE ALEATÓRIA 1.	42
TABELA 18: MATRIZ DE CONFUSÃO PARA SEMENTE ALEATÓRIA 2.	42
TABELA 19: MATRIZ DE CONFUSÃO PARA SEMENTE ALEATÓRIA 3.	42
TABELA 20: MATRIZ DE CONFUSÃO PARA SEMENTE ALEATÓRIA 4.	43
TABELA 21: MATRIZ DE CONFUSÃO PARA SEMENTE ALEATÓRIA 5.	43
TABELA 22: CLASSE PREDITA DO ALINHAMENTO COM O BANCO DE DADOS <i>SWISS-PROT</i> .	44
TABELA 23: PROTEÍNAS DE HUMANOS: CLASSE PREDITAS.	45
TABELA 24: CLASSE PREDITA DO ALINHAMENTO COM PROTEÍNAS DE HUMANOS.	46
TABELA 25: CLASSE PREDITA DO ALINHAMENTO COM DE <i>APIS MELLIFERA</i> .	47
TABELA 26: ALINHAMENTO DE FOSFOLIPASES DE <i>APIS MELLIFERA</i> COM PROTEÍNAS DE SERPENTES.	48
TABELA 27: NÚMERO DE SEQÜÊNCIAS UTILIZADAS.	49
TABELA 28: ERRO DE CLASSIFICAÇÃO UTILIZANDO <i>10-FOLD CROSS-VALIDATION</i>	50
TABELA 29: MATRIZ DE CONFUSÃO PARA SEMENTE ALEATÓRIA 1.	50
TABELA 30: MATRIZ DE CONFUSÃO PARA SEMENTE ALEATÓRIA 2.	50
TABELA 31: MATRIZ DE CONFUSÃO PARA SEMENTE ALEATÓRIA 3.	51
TABELA 32: MATRIZ DE CONFUSÃO PARA SEMENTE ALEATÓRIA 4.	51
TABELA 33: MATRIZ DE CONFUSÃO PARA SEMENTE ALEATÓRIA 5.	51
TABELA 34: CLASSIFICAÇÃO DE HEMOGLOBINAS DE HUMANOS.	52
TABELA 35: CLASSIFICAÇÃO DE FERRITINAS DE HUMANOS.	53
TABELA 36: CLASSIFICAÇÃO DE MIOSINAS.	54
TABELA 37: CLASSIFICAÇÃO DE QUERATINAS.	54
TABELA 38: CLASSIFICAÇÃO DE PROTEÍNAS G.	55

## LISTA DE SIGLAS

**BLAST** – *Basic Local Alignment Search Tool* (Ferramenta de Alinhamento Local).

**MLP** – *Multilayer Perceptron* ( Perceptron Multicamadas).

**HSP** - *High-scoring segment pair* (Alinhamento Local de alta pontuação).

**DNA** - Ácido desoxirribonucléico.

**RNA** - Ácido ribonucléico.

**IA** – Inteligência Artificial.

**AM** – Aprendizado de Máquina.

**PDB** – *Protein Data Bank* (Banco de Dados de Proteínas).

**HTTP** -*Hypertext Transfer Protocol* (Protocolo de Transferência de Hipertexto).

# SUMÁRIO

<b>RESUMO.....</b>	<b>IV</b>
<b>LISTA DE FIGURAS.....</b>	<b>V</b>
<b>LISTA DE TABELAS.....</b>	<b>VI</b>
<b>LISTA DE SIGLAS.....</b>	<b>VII</b>
<b>1. INTRODUÇÃO .....</b>	<b>1</b>
1.1. MOTIVAÇÃO .....	2
1.2. SOLUÇÃO PROPOSTA .....	3
1.3. ORGANIZAÇÃO DO DOCUMENTO .....	3
<b>2. O ESTADO DA ARTE.....</b>	<b>4</b>
2.1. CONSIDERAÇÕES INICIAIS .....	4
2.2. VISÃO GERAL DOS ARTIGOS.....	4
2.3. CONSIDERAÇÕES FINAIS .....	6
<b>3. METODOLOGIA E PROCEDIMENTOS .....</b>	<b>7</b>
3.1. CONSIDERAÇÕES INICIAIS .....	7
3.2. ESTUDO DOS TEMAS RELACIONADOS AO PROJETO.....	7
3.3. LEVANTAMENTO DOS DADOS .....	14
3.4. PRÉ-PROCESSAMENTO DOS DADOS .....	15
<b>4. IMPLEMENTAÇÃO DO SISTEMA .....</b>	<b>16</b>
<b>5. TESTES REALIZADOS.....</b>	<b>28</b>
5.1. MÓDULO DE ALINHAMENTO.....	28
5.2. MÓDULO DE VISUALIZAÇÃO.....	30
5.3. MÓDULO DE CLASSIFICAÇÃO .....	33
<b>6. CONCLUSÕES.....</b>	<b>57</b>
<b>7. REFERÊNCIAS .....</b>	<b>59</b>
<b>APÊNDICE A .....</b>	<b>61</b>

# 1. INTRODUÇÃO

Proteínas são macromoléculas responsáveis pelo funcionamento da maioria dos processos biológicos dos organismos vivos, sendo parte constituinte das estruturas e das atividades desempenhadas. Elas são compostas por aminoácidos unidos por ligações covalentes formando seqüências com diferentes tamanhos e constituições (LENINGER *et al*, 1998).

As proteínas possuem diferentes níveis estruturais complexos. Eles são definidos como:

a) Estrutura Primária

Nível mais básico composto pela seqüência de aminoácidos que compõe a proteína.

b) Estrutura Secundária

Disposição espacial dos aminoácidos mais próximos entre si que compõe a estrutura primária. Alguns desses arranjos bastante conhecidos são as folhas- $\beta$  e as hélices- $\alpha$ .

c) Estrutura Terciária

Disposição espacial dos átomos dos aminoácidos que compõe a estrutura primária. É a forma tridimensional enovelada de uma proteína.

d) Estrutura Quaternária

Algumas proteínas possuem mais de uma cadeia polipeptídica e a estrutura quaternária representa a disposição dessas cadeias dentro de estrutura protéica (LENINGER *et al*, 1998).

A complexidade existente nas estruturas protéicas é o fator responsável pelas diferenças funcionais entre elas. As proteínas são divididas em famílias ou classes, dependendo de características funcionais e estruturais. Muitas proteínas da mesma classe apresentam semelhanças entre seus níveis estruturais (TSUNODA, 2004).

De acordo com (LENINGER *et al*, 1998), entre principais funções biológicas das proteínas, destacam-se:

a) Enzimas

Constituem um grupo bastante variado de proteínas, que possuem ação catalítica e participam das reações químicas de biomoléculas orgânicas. As lipases são exemplos de enzimas.

b) Proteínas Transportadoras

Essas proteínas ligam-se a íons ou a outras moléculas, transportando-os de um lugar para outro. As proteínas do plasma sanguíneo, como a hemoglobina dos eritrócitos, são exemplos de proteínas transportadoras.

c) Proteínas de Armazenamento

Armazenam nutrientes necessários ao organismo ou célula. As proteínas presentes nas sementes de muitas plantas, que armazenam nutrientes necessários à germinação, são exemplos de proteínas de armazenamento. A ferritina, responsável pelo armazenamento de ferro, é um exemplo dessa classe de proteínas.

d) Proteínas Contráteis ou de Motilidade

Essas proteínas conferem motilidade e capacidade de contração as células e organismos. A actina e miosina, que são constituintes do músculo esquelético, são exemplos de proteínas contráteis.

e) Proteínas Estruturais

Fornecem proteção e resistência a organismos e células. Exemplos de proteínas estruturais são o colágeno e a queratina.

f) Proteínas de Defesa

Protegem os organismos da invasão de outras espécies. As imunoglobulinas ou anticorpos são exemplos de proteínas de defesa.

g) Proteínas Reguladoras

Contribuem para a regulação da atividade celular ou fisiológica. Entre elas estão os alguns hormônios como, por exemplo, a insulina.

A necessidade de técnicas para investigação de aspectos funcionais e estruturais das proteínas e classificação das mesmas em famílias tem sido um grande desafio para os pesquisadores (COSTA *et al.*, 2005). Este trabalho propõe a utilização de algumas dessas técnicas e a sua integração em um ambiente que facilite o estudo e pesquisa desse tema, facilitando o trabalho dos pesquisadores da área.

## **1.1. Motivação**

Muitos complexos protéicos deixam de ser estudados pela comunidade científica devido à dificuldade de coleta, como, por exemplo, os venenos de animais (AMUI, 2006). Além disso, em muitos casos, os custos dessa investigação podem ser altos. Desse modo, o uso de técnicas computacionais que proporcionem uma análise prévia de estruturas protéicas pode ser útil para guiar os cientistas, antes de uma possível análise laboratorial.

A Bioinformática, que lida com dados de seqüenciamento de DNA (Ácido desoxirribonucléico), RNA (Ácido Ribonucléico) e proteínas, auxiliando na investigação de

suas funções, freqüentemente, trabalha com grande volume de informação, o que torna necessário o uso de técnicas que facilitem a manipulação e interpretação de dados (LORENA & CARVALHO, 2003). Além disso, com o avanço de tecnologias aplicadas a biologia e medicina, cada vez mais ferramentas têm sido oferecidas para auxiliar no cotidiano de profissionais da área. A utilização dessas ferramentas, no entanto, nem sempre pode ser feita em um ambiente comum, ou seja, utilizando um único software que as agregue. Dessa forma, a proposta do presente trabalho é reunir e implementar ferramentas úteis aos profissionais da área, facilitando e otimizando suas tarefas diárias.

Técnicas de Inteligência Artificial (IA) têm sido aplicadas com sucesso em diferentes problemas de Bioinformática, como na análise de seqüências de genomas, predição de estruturas secundárias de proteínas, alinhamento de seqüências, entre outras (BALDI & BRUNNAK, 1998).

## **1.2. Solução Proposta**

Este trabalho propõe a criação de um ambiente computacional que possibilite análise, classificação e visualização de proteínas, oferecendo mecanismos para atividades pré-laboratoriais. A escolha das ferramentas baseou-se nas necessidades de alguns profissionais que trabalham na investigação de proteínas de venenos do Departamento de Genética da Faculdade de Medicina de Ribeirão Preto (FMRP).

As tarefas desenvolvidas inicialmente foram divididas em três módulos: um módulo para classificação de proteínas, um para alinhamento e um para visualização. Maiores detalhes sobre os três módulos serão apresentados nas próximas seções.

## **1.3. Organização do Documento**

Esta monografia está organizada da seguinte maneira: a Seção 2 apresenta o estado da arte; a Seção 3, metodologia e procedimentos; a Seção 4, implementação do sistema; a Seção 5, testes realizados; a Seção 7, referências bibliográficas e a Seção 8, apêndice A.

## 2. O ESTADO DA ARTE

### 2.1. Considerações Iniciais

Nesta seção serão comentados alguns artigos científicos utilizados para a definição do escopo e da metodologia empregada neste trabalho. Os trabalhos apresentados serão: (SOUTO *et al.*, 2005), (COSTA *et al.*, 2005), (SANTOS *et al.*, 2006), (SANTOS, E.C., 2004) e (ALTSCHUL *et al.*, 1990).

### 2.2. Visão Geral dos Artigos

Em (SOUTO *et al.*, 2005), são descritas técnicas de aprendizado de máquina aplicadas a problemas de biologia molecular. Os problemas abordados são: predição de genes, análise de dados de expressão gênica e construção de filogenia. As técnicas descritas são representativas dos diversos tipos de aprendizado: conexionista, estatístico, evolutivo e simbólico.

Uma aplicação importante descrita neste artigo é a identificação dos sítios de início da tradução. Esses sítios são marcados pela presença do códon AUG, que codifica o aminoácido *Metionina*, no entanto, nem todo AUG é um ponto de início. Nos procariotos, existe outra informação relevante para a determinação dos sítios. Trata-se da presença de determinada seqüência anterior ao códon AUG, que se liga ao RNA mensageiro, durante a tradução. Essa seqüência, porém apresenta grandes variações de composição, fato que dificulta sua identificação por métodos tradicionais. A solução proposta é a aplicação de técnicas de aprendizado de máquina, como redes neurais.

Outra aplicação interessante apontada pelo artigo é a identificação de sítios de *splicing*. Nos organismos eucariotos nem todo o DNA é traduzido em proteínas. Algumas regiões chamadas de *introns* não são traduzidas, enquanto outras, chamadas de *exons*, são. As fronteiras entre essas regiões são os sítios de *splicing*. O artigo mostra algumas soluções para determinar se uma seqüência possui a fronteira *intron-exon*, *exon-intron* ou nenhuma delas. Dentre os métodos utilizados, as Redes Neurais obtiveram o melhor resultado na classificação.

Além disso, as Máquinas Vetores Suporte (SVM), que constituem uma técnica de aprendizado estatístico, são apontadas pela sua robustez diante dos problemas de bioinformática, que em geral tem grande dimensão. Diversos outros trabalhos encontrados na

literatura, entre eles (BALDI & BRUNNAK, 1998), também enfatizam o uso do paradigma estatístico.

Em (COSTA *et al.*, 2005), são apresentadas técnicas de aprendizado de máquina para determinação de similaridade entre proteínas sem considerar informações de sua estrutura primária. A similaridade, neste caso, é observada de acordo com o tipo de estrutura secundária encontrada, ou seja, verifica-se similaridade entre duas proteínas, isto é, se elas contêm a mesma estrutura secundária principal com arranjo semelhante.

Para tal problema, foram utilizados alguns classificadores e multiclassificadores disponíveis no pacote de aprendizado de máquina do Weka (WITTEN & FRANK, 2000) entre eles, uma rede neural. Os resultados mostraram que os classificadores e multiclassificadores obtiveram desempenho regular na classificação, sendo que os últimos obtiveram resultados melhores.

Em (SANTOS *et al.*, 2006), foi apresentado o problema de classificação de proteínas de venenos de serpentes, onde as classes eram as plantas medicinais inibidoras. O propósito do trabalho era encontrar plantas medicinais inibidoras para proteínas de venenos ainda não estudadas.

Neste trabalho, as proteínas foram classificadas considerando informações apenas de sua estrutura primária, ou seja, da cadeia de aminoácidos. Foram também feitos testes utilizando dois métodos de codificação de proteínas: o *n-gram* e o *6-letter exchange group*, descritos em (WANG *et al.*, 2001). Os resultados obtidos mostraram que a codificação *n-gram* obteve os melhores resultados.

Quanto à classificação, foram realizados testes com uma rede neural do tipo MLP. Os resultados mostraram que a MLP obteve bons resultados na classificação.

Em (SANTOS, E.C., 2004), foi realizada uma análise das ferramentas mais utilizadas em Bioinformática para a análise de alinhamentos e foram apresentados alguns bancos de dados biológicos públicos para a obtenção de seqüências protéicas. O BLAST é apontado como a ferramenta mais popular para a realização de alinhamentos, sendo que sua implementação mais conhecida é a fornecida pelo NCBI.

Em (ALTSCHUL *et al.*, 1990), a importância da realização e estudo dos alinhamentos é ressaltada. O artigo enfatiza o fato de que seqüências similares de genes ou proteínas têm maior probabilidade de possuírem funções semelhantes. Deste modo, também ressalta que as primeiras informações a respeito das funções de seqüências descobertas recentemente,

normalmente, são obtidas a partir dessa técnica. Assim, em muitos casos, o alinhamento obtém bons resultados podendo indicar inclusive a família de determinada seqüência.

### **2.3.Considerações Finais**

Os artigos acima apresentados mostram que técnicas de Inteligência Artificial têm sido usadas com sucesso nos diversos problemas de Bioinformática. Além disso, algumas ferramentas disponíveis na Internet são apontadas como indispensáveis para o estudo de proteínas.

Também, pode-se concluir que o estudo de similaridade e classificação de proteínas pode ser realizado utilizando informações de seus vários níveis estruturais. Assim, essas informações podem ser levadas em conta quando se deseja estudar uma seqüência protéica nova.

## **3. METODOLOGIA E PROCEDIMENTOS**

### **3.1. Considerações Iniciais**

As atividades realizadas neste projeto foram: estudo da bibliografia, levantamento e pré-processamento dos dados, escolha dos algoritmos e ferramentas empregados, análise de requisitos do sistema e implementação e teste do sistema.

O sistema é composto pelos módulos de classificação, visualização e alinhamento. As entradas são constituídas por uma ou mais seqüências protéicas, que contem estruturas primárias, ou seja, os aminoácidos que as compõe. As seqüências podem, então, ser classificadas de acordo com as classes de proteínas existentes. Além disso, as seqüências de entrada podem ser alinhadas e é possível observar as estatísticas do alinhamento. Também, a existência da estrutura terciária dessas seqüências é verificada para possível visualização.

O sistema foi desenvolvido na linguagem de programação Java. Essa linguagem foi escolhida por ser orientada a objetos, facilitando assim o desenvolvimento, favorecendo a reutilização das classes, a manutenibilidade, a portabilidade e a integração com outros softwares. O Java também fornece ferramentas para a geração de documentação em formato padrão.

### **3.2. Estudo dos Temas Relacionados ao Projeto**

A seguir serão descritos os principais temas relacionados a este trabalho, cada um deles importante para a compreensão dos módulos do sistema. Serão discutidos também os algoritmos e ferramentas escolhidas para cada módulo.

#### **3.2.1. Alinhamento de Seqüências**

O alinhamento de seqüências de DNA, RNA ou proteínas é uma técnica bastante conhecida, que tem como principal objetivo encontrar fragmentos conservados, ao longo da evolução, em diferentes seqüências.

Duas seqüências podem ser similares, quando possuem fragmentos em comum, ou homólogas, quando são similares devido a um ancestral comum.

Um alinhamento pode ser global ou local, sendo que é local, quando busca por regiões de alta similaridade, que compõem uma parte da seqüência e um alinhamento é global, quando se considera a similaridade ao longo de toda seqüência.

Existem vários métodos de alinhamento de seqüências: entre eles podem-se citar matrizes de pontos, programação dinâmica e métodos de palavras ou *k-tuplas*. Neste projeto,

abordaremos apenas o método de palavras por serem ideais para utilização em bancos de dados com grande volume de informação.

Um popular método de palavras que será utilizado neste projeto é o BLAST (ALSTCHUL *et al.*, 1990). Esta ferramenta realiza alinhamento local de fragmentos de uma seqüência de interesse com um banco de dados de seqüências, apresentando desempenho aceitável.

O BLAST identifica palavras significantes para a realização da busca. A significância é calculada com o auxílio de matrizes de ponderação, como, por exemplo, a PAM ou a BLOSSUM (KORF *et al.*, 2003).

Conforme apresentado em (ALSTCHUL *et al.*, 1990), o algoritmo do BLAST pode ser dividido em quatro fases ou etapas, as quais são descritas a seguir.

a) Identificação e construção da lista de palavras;

As palavras com maior significância são extraídas das seqüências de interesse para compor uma lista. No caso dos aminoácidos, as palavras de um tamanho  $x$  pré-definido são geradas por todas as combinações possíveis dos 20 aminoácidos existentes, desse modo, o número de palavras obtidas é  $20^x$ .

A seguir, as palavras com maior significância são selecionadas. Essa seleção ocorre de acordo com um ponto de corte para a pontuação, que também é pré-definido.

b) Busca de cada palavra da lista em todas as seqüências do banco;

A seguir, o algoritmo realiza uma busca de cada uma das palavras da lista em cada seqüência do banco de dados. As seqüências encontradas, cujo casamento é perfeito, são chamadas de *hits* e passam a ser pontos de referência para a continuação do alinhamento.

c) Extensão;

Os *hits* ou palavras encontradas agora são estendidos, de modo que novas identidades são buscadas para ambos as direções, partindo dos extremos de cada *hit*.

d) Alinhamento de seqüências;

A seqüência mais longa obtida, durante a extensão, é chamada HSP e tem sua significância estatisticamente avaliada. Por fim, seqüências com alinhamento mais significante são retornadas.

Encontrar regiões conservadas ao longo da evolução é uma técnica importante, pois possibilita inferir sobre a função de proteínas ainda não conhecidas, comparando-as com as mais similares (ALSTCHUL *et al.*, 1990).

Neste projeto, o BLAST foi utilizado para compor o módulo de alinhamento de seqüências. A implementação escolhida foi a fornecida pelo NCBI por tratar-se de uma ferramenta livre e não comercial.

### 3.2.2. Estruturas Protéicas Terciárias

Como descrito anteriormente, as proteínas possuem quatro níveis estruturais, neste tópico, daremos ênfase ao estudo das estruturas terciárias.

O estudo e a compreensão da estrutura terciária de proteínas é também um importante fator para classificação, pois a forma tridimensional de uma proteína é fundamental para determinação de sua atividade funcional (TSUNODA, 2004; PROSDOCIMI *et al.*, 2003).

Quanto à forma, as proteínas são classificadas de acordo com a conformação que adquirem em seu estado nativo. Dependendo de sua forma, as proteínas podem ser fibrosas ou globulares. As proteínas globulares e fibrosas são exemplos da influência da estrutura na função desempenhada. A maioria das proteínas globulares é solúvel em água e desempenha função enzimática e de transporte. Já as proteínas fibrosas são insolúveis e, geralmente, desempenham funções estruturais, um exemplo é a queratina (TSUNODA, 2004).

Assim, a observação da estrutura terciária é um fator extremamente relevante para investigação e pesquisa das estruturas protéicas.

O banco de dados públicos de proteínas PDB, disponível em <http://www.pdb.org>, é um repositório internacional de estruturas protéicas, que possibilita a busca por estruturas protéicas já conhecidas. O formato de arquivo padrão disponível no banco de dados do PDB, utilizado pela maioria dos softwares de visualização de estrutura, é o PDB.

A Figura 1 mostra um exemplo de arquivo no formato PDB. No cabeçalho do arquivo (HEADER), encontra-se o nome da classe da proteína, neste exemplo, trata-se de uma Imunoglobulina, a data e o código de quatro dígitos que representa a estrutura, que neste caso é 12E8. Em seguida, em SOURCE, apresenta-se também o organismo de onde a proteína provém, o tipo de célula e outras informações de origem. Em EXPDTA, podemos observar qual técnica foi utilizada para determinar esta estrutura. A palavra HELIX indica a presença de hélices  $\alpha$ , enquanto a palavra SHEET indica a presença de folhas  $\beta$ . Em seguida, após a palavra ATOM são apresentados todos os átomos e suas posições no espaço (TSUNODA, 2004).

```

HEADER IMMUNOGLOBULIN 14-MAR-98 12E8
TITLE 2E8 FAB FRAGMENT
(...) (...) (...)
SOURCE MOL_ID: 1;
SOURCE 2 ORGANISM_SCIENTIFIC: MUS MUSCULUS;
SOURCE 3 ORGANISM_COMMON: MOUSE;
SOURCE 4 STRAIN: BALB/C;
SOURCE 5 CELL_LINE: 2E8 HYBRIDOMA;
SOURCE 6 ORGAN: SPLEEN;
SOURCE 7 CELL: LYMPHOCYTE-PLASMA CELL
KEYWDS IMMUNOGLOBULIN
EXPDTA X-RAY DIFFRACTION
AUTHOR B.RUPP,S.TRAKHANOV
REVDAT 1 05-AUG-98 12E8 0 9
...
SEQRES 17 P 221 SER SER THR LYS VAL ASP LYS LYS ILE VAL PRO
ARG ASP
HELIX 1 1 SER L 80 ASP L 82 5 3
HELIX 2 2 SER L 122 SER L 127 1 6
...
SHEET 1 A 2 PHE L 10 THR L 13 0
SHEET 2 A 2 LYS L 103 LEU L 106 1 N LYS L 103 O MET L 11
SHEET 1 B 3 VAL L 19 LYS L 24 0
...
ATOM 1 N ASP L 1 74.982 33.405 -6.325 1.00 19.77 N
ATOM 2 CA ASP L 1 74.669 34.823 -6.680 1.00 20.74 C
ATOM 3 C ASP L 1 73.505 35.333 -5.830 1.00 19.56 C

```

**Figura 1: Exemplo de Arquivo no Formato PDB.**

Neste projeto, o software escolhido para visualização foi o Jmol (HERRÁEZ, 2006; CASS, 2005), disponível em <http://jmol.sourceforge.net/>. A escolha deveu-se ao fato do software ser livre, de código aberto e por estar implementado na linguagem Java, facilitando a integração com o sistema.

O módulo de visualização busca por uma dada estrutura terciária no banco de dados do PDB, como resposta, o sistema obtém algumas seqüências similares, cuja estrutura já foi determinada, experimentalmente. A estrutura, ou estruturas de interesse, pode(m), ser visualizada(s) através do software Jmol. Nenhum processamento será efetuado nos arquivos PDB obtidos, desse modo, eles serão visualizados da maneira com que foram retornados pelo banco de dados do PDB.

### 3.2.3. Redes Neurais Artificiais

O MLP é constituído por conjunto unidades sensoriais em uma camada de entrada, uma ou mais camadas ocultas (intermediárias) formadas por neurônios e uma camada de saída, também formada por neurônios.

O MLP pode ser visto como um veículo prático para realizar mapeamentos de funções não-lineares (HAYKIN, 1994).

Considerando uma MLP com uma única camada escondida, apresentando-se o padrão de entrada  $X(n)=[X_1(n), X_2(n)... X_p(n)]^T$ , a ativação de um neurônio  $j$  da camada oculta é dada por:

$$Z_j = \varphi(v_j(n)) = \varphi\left(\sum \omega_{ji}(n)X_i(n)\right) \quad (1)$$

A ativação do neurônio  $k$  da camada de saída será:

$$Y_k = \varphi(v_k(n)) = \varphi\left(\sum \omega_{kj}(n)Z_j(n)\right) \quad (2)$$

onde  $\varphi(\cdot)$  é a função de ativação não-linear (normalmente usa-se a função sigmoideal) do neurônio,  $v_k(\cdot)$  a ativação interna no neurônio  $k$ ,  $\omega_{ji}$  representa os pesos entre a camada de entrada e a camada oculta,  $\omega_{kj}$  são os pesos entre os neurônios da camada oculta e os neurônios da camada de saída,  $i=1, \dots, p$  o índice dos neurônios da camada de entrada,  $j = 1, \dots, m$  o índice dos neurônios da camada escondida, e  $k = 1, \dots, q$  o índice dos neurônios da camada de saída (HAYKIN, 1994).

Em uma MLP um sinal se propaga através das camadas em direção à camada de saída, cada sinal de entrada está associado a uma saída desejada. O treinamento possui duas fases:

**Fase Forward:** Uma entrada passa pela camada de entrada, e prossegue até o fim, de modo que após os neurônios de uma camada calcularem suas saídas, os neurônios da próxima camada utilizam estes valores como entrada. No final, a saída da ultima camada é comparada à saída desejada e o erro é calculado.

**Fase Backward:** Esta fase é baseada no gradiente descendente do erro. Cada neurônio ajusta seus pesos para reduzir o erro. Assim, cada neurônio das camadas anteriores tem seu erro igual ao erro das camadas seguintes ponderados pelos pesos das conexões entre eles.

O objetivo do treinamento é reduzir a função de custo baseada no erro instantâneo  $e(n)$ :

$$e(n) = d(n) - y(n) \quad (3)$$

onde  $d(n)$  é a saída desejada e  $y(n)$  a saída encontrada.

Assim, a energia do Erro  $E(n)$  é calculada da seguinte forma:

$$E(n) = \frac{1}{2} \sum e^2 (n) \quad (4)$$

Como o erro é uma função dos parâmetros livres da MLP, sua minimização ocorre através do ajuste nos pesos da rede.

Em uma MLP com uma camada oculta a classificação é realizada por hiperplanos no espaço de decisão formado pelos padrões de entrada, isto é, pelos exemplos a serem classificados. Uma MLP com uma única camada é capaz de tratar somente problemas linearmente separáveis, uma vez que é formada pela combinação de funções lineares. A adição de pelo menos mais uma camada, que combina as saídas da camada anterior e utiliza função de ativação não linear, visa facilitar o problema de classificação, pois utiliza uma série de transformações não-lineares para simplificar a separação dos dados em um novo espaço de classificação. Cada neurônio da rede forma um hiperplano que divide o espaço de decisão, sendo que a composição de todos os hiperplanos em uma MLP com pelo menos uma camada oculta forma regiões complexas responsáveis pela classificação dos exemplos.

O algoritmo comumente utilizado para treinamento e que será empregado aqui é o *Backpropagation*. O algoritmo pode ser visto na Figura 2:

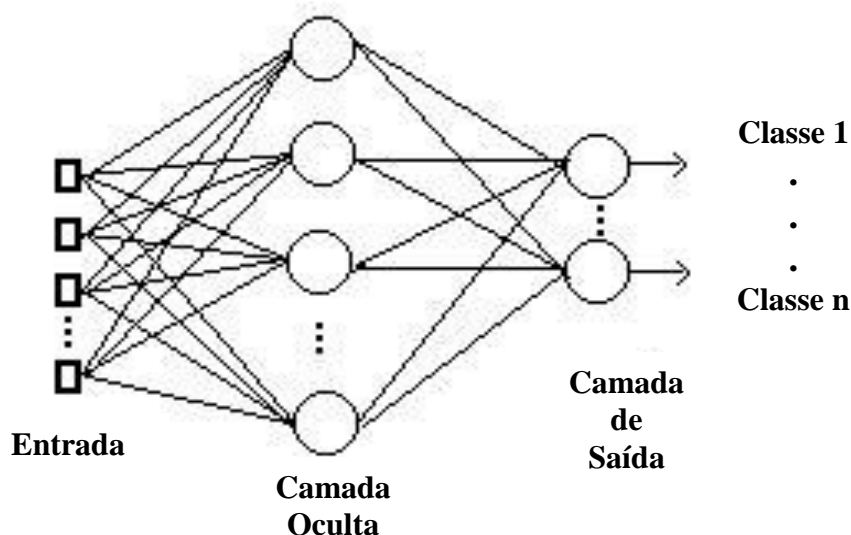
```
Algoritmo 1: Backpropagation  
Início  
  inicialize todas as conexões com valores aleatórios  
  n <- 1  
  faça  
    // fase forward  
    para cada camada k (começando da primeira)  
      para cada neurônio j da camada k  
        calcular a saída yj  
      fim para  
    fim para  
    Calcular a energia total do Erro E(n)  
    se E(n)>0 // fase backward  
      para cada camada k (começando da ultima)  
        para cada neurônio j da camada k  
          Atualizar pesos (w)  
        fim para  
      fim para  
    fim se  
  fim faça  
  n <- n + 1  
enquanto(critério de convergência não tenha sido satisfeito)
```

Figura 2: Algoritmo *Backpropagation*.

Neste projeto, uma rede MLP foi utilizada para classificar seqüências protéicas, utilizando apenas informações da estrutura primária.

A MLP foi escolhida, como algoritmo de classificação, devido ao seu bom desempenho em problemas semelhantes encontrados na literatura (SOUTO *et al.*, 2005), (SANTOS *et al.*, 2006).

A Figura 3 mostra a rede MLP implementada, com uma única camada oculta e com  $n$  neurônios na camada de saída, sendo  $n$  o número de classes utilizado na classificação.



**Figura 3: Ilustração de uma MLP.**

A MLP da Figura 3 foi implementada na linguagem de programação Java. As fases de sua construção foram: normalização das entradas da rede, separação dos exemplos em conjunto de treinamento e conjunto de testes, codificação da MLP, treinamento e teste.

A MLP implementada especializa cada neurônio da camada de saída como sendo classificador de uma determinada classe, por essa razão, o número de neurônio da camada de saída é sempre igual ao número de classes. Desse modo, um exemplo de determinada classe deve ativar seu neurônio classificador.

A validação dos resultados obtidos pela MLP foram obtidos através da utilização do método *10-fold cross-validation*, que será apresentado a seguir.

### **3.2.3.1. Validação dos resultados obtidos: método *10-fold cross-validation***

O método *10-fold cross-validation* consiste no particionamento do conjunto de dados em 10 grupos (*folds*), com distribuição de classes e número de exemplos aproximadamente iguais, 9 das 10 partições são utilizadas para o treinamento e a partição restante, que não participou do treino, é utilizada para o teste. O procedimento é repetido 10 vezes, ou seja, uma para cada

partição. O número de *folds* igual a 10 foi escolhido porque é comumente utilizado na literatura.

Para avaliar os resultados obtidos pela MLP, o método *10-fold cross-validation* foi implementado. A taxa de erro na classificação foi calculada realizando a média das taxas nas 10 execuções da rede para os diferentes conjuntos de treinamento e teste. A matriz de confusão final foi obtida somando as matrizes geradas em cada execução.

### 3.3. Levantamento dos Dados

Parte das seqüências protéicas utilizadas foram fornecidas pelo Laboratório de Bioinformática do Departamento de Genética da Faculdade de Medicina de Ribeirão Preto. Estas seqüências fazem parte de um banco de dados relacionado à investigação dos venenos de serpente. As pesquisas relacionadas promovem maior conhecimento de características funcionais e estruturais dessas seqüências, utilizando técnicas computacionais usadas em bioinformática. Além disso, são realizados estudos sobre as relações de inibição dessas proteínas de venenos com plantas medicinais (SANTOS *et al.*, 2006).

Algumas seqüências, no entanto, foram obtidas diretamente do banco de dados publico dados público mantido pelo *National Center for Biotechnology Information* (NCBI) disponível em [www.ncbi.nih.gov](http://www.ncbi.nih.gov). Todas as seqüências utilizadas neste projeto encontram-se no formato FASTA.

A Figura 4 apresenta uma seqüência no formato “fasta”. O início da seqüência é marcado com “>” seguido de um identificador da proteína e de uma descrição. Logo a seguir, tem-se a seqüência protéica. Normalmente, não são utilizados marcadores para indicar o final da seqüência, porém, aqui utilizaremos o “@” seguido pelo nome da classe da proteína.

```
>gi|82239742|Q71QJ2 Venom serine protease KN6precursor [Viridoviperastejnegeri]
MVLIRVLANLLILQLSYAQKSELVIGGDECNINEHRFLVALYDVSSGDFRSGTTLINPEWVLTAAHCETEEM
KLQFGLHSKRVPNKDKQTRVSKEKFFCESNKNYTKWNKDIMLIKLNRPVKNSAHIEPLSLPSSPPSVGSVCRI
MGWGTLSDTEMILPDVPHCANINLLNYSDCQAAYPELPAKSRTLCAILEGGKDTCSGDSGGPLICNGTFQGI
ASWGSTLCGYVREPGSYTKVFDHLDWIQSIHAGNTNVTCP
@serinoprotease
```

Figura 4: Seqüência no formato Fasta

Todas as seqüências obtidas foram verificadas manualmente para garantir que pertenciam realmente as famílias indicadas. Essa filtragem foi realizada através da observação

do identificador da seqüência presente no arquivo fasta, onde é possível obter uma descrição da proteína.

### 3.4. Pré-Processamento dos Dados

O uso das seqüências de aminoácidos não pode ser feito diretamente, uma vez que as seqüências têm tamanhos variados. Desse modo, para codificação das seqüências foi utilizada a seguinte abordagem: método de codificação *n-gram* ou *n-tuplas* (WANG *et al.*, 2001).

- ***n-gram* ou *n-tuplas***

Esse método consiste em extrair vários padrões de *n* aminoácidos de uma seqüência contando a ocorrência de cada um deles. Por exemplo, dada a seqüência de aminoácidos ACLVAC, a codificação *2-gram* seria a seguinte: 2 AC (AC ocorre 2 vezes), 1 CL (CL ocorre 1 vez), 1 LV (LV ocorre 1 vez) e 1 VA (VA ocorre 1 vez).

A seguir, calcula-se a freqüência dos padrões de aminoácidos ou grupos:

$$X_{AB} = \frac{c_{ab}}{len - 1} \quad (5)$$

onde,  $c_{ab}$  é o número de ocorrências de um determinado padrão formado pelos aminoácidos *a* e *b* (ou grupos *a* e *b*) e *len* representa o comprimento da seqüência de aminoácidos.

Os padrões, como descrito anteriormente, são todas as combinações dois a dois de aminoácidos, para este problema teremos:  $20^2 = 400$  padrões, sendo 20 o número de aminoácidos.

Neste projeto, as seqüências foram processadas segundo esta codificação, porém não foi realizado o cálculo da freqüência dos padrões, mas sim o valor absoluto do número de ocorrência dos padrões.

Essa variação do método foi testada em trabalhos anteriores (OLIVEIRA *et al.*, 2007). A conclusão obtida foi que levando em conta o fato de que os sítios ativos das proteínas não dependem da freqüência com que um aminoácido ou conjunto de aminoácidos aparece ao longo de sua estrutura, mas sim do número de vezes que aparece em um determinado local ou sítio, podemos considerar que uma abordagem com valores absolutos é mais condizente com as estruturas protéicas existentes.

A MLP implementada tem como entrada uma seqüência, a qual será constituída pelos 400 padrões (atributos) calculados pela codificação *2-gram*. Cada seqüência, durante o treinamento, está associada a uma classe, que representa sua família protéica. A MLP, durante a fase de teste, classifica as seqüências entre as famílias que foram utilizadas durante o treinamento.

## 4. IMPLEMENTAÇÃO DO SISTEMA

A implementação do Sistema de Apoio ao Estudo de Proteínas baseou-se na Especificação de Requisitos, que pode ser vista no Apêndice A. Conforme anteriormente mencionado, a linguagem de programação utilizada foi o Java. As interfaces gráficas do sistema, bem como as funcionalidades presentes, serão apresentadas a seguir.

A interface inicial, presente na Figura 5, apresenta duas opções: a primeira delas, constituindo o módulo de classificação de proteínas e a segunda, os módulos de visualização e alinhamento de proteínas. Os módulos do sistema serão apresentados abaixo:

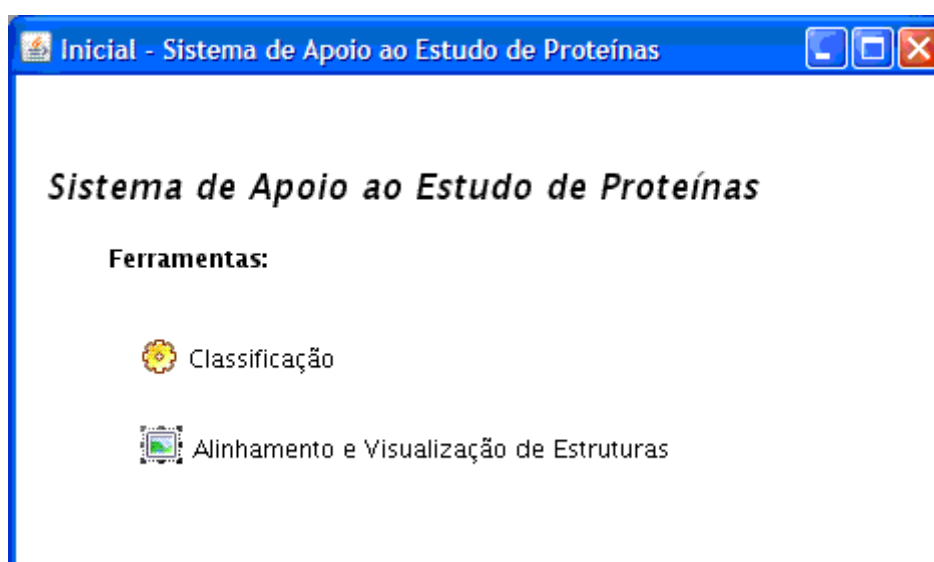


Figura 5: Interface Inicial.

- **Módulos de Alinhamento e Visualização**

Os módulos de visualização e alinhamento foram agrupados, para simplificar a realização de algumas tarefas, como, por exemplo, visualizar as estruturas de seqüências alinhadas.

Os dois módulos necessitam da Internet para funcionarem adequadamente, uma vez que o Alinhamento e a Busca de Estruturas são realizados remotamente.

A Figura 6 representa o módulo de alinhamento de seqüências, onde o usuário poderá selecionar uma seqüência no formato FASTA, preencher as informações necessárias para a realização do alinhamento. As opções de preenchimento são:

- *Email*: O endereço de *email* do usuário é requerido para envio de informações, caso ocorra algum problema com o servidor responsável pela realização do BLAST.
- Banco de Dados: o banco de dados com o qual deseja-se realizar o alinhamento. Como, por exemplo: PDB ou *Swiss-Prot*.
- *Score* Mínimo: pontuação mínima de *score* requerida para a visualização dos resultados do alinhamento.
- Matriz de ponderação: A matriz de ponderação utilizada pra a realização do alinhamento. Exemplo: PAM ou BLOSSUM.

Para realizar o alinhamento, o sistema envia uma requisição ao servidor do BLAST, juntamente com os parâmetros mencionados acima. O servidor realiza o alinhamento e retorna alguns arquivos com a saída do alinhamento realizado. O sistema processa o conjunto de arquivos recebidos para proporcionar a visualização dos resultados.

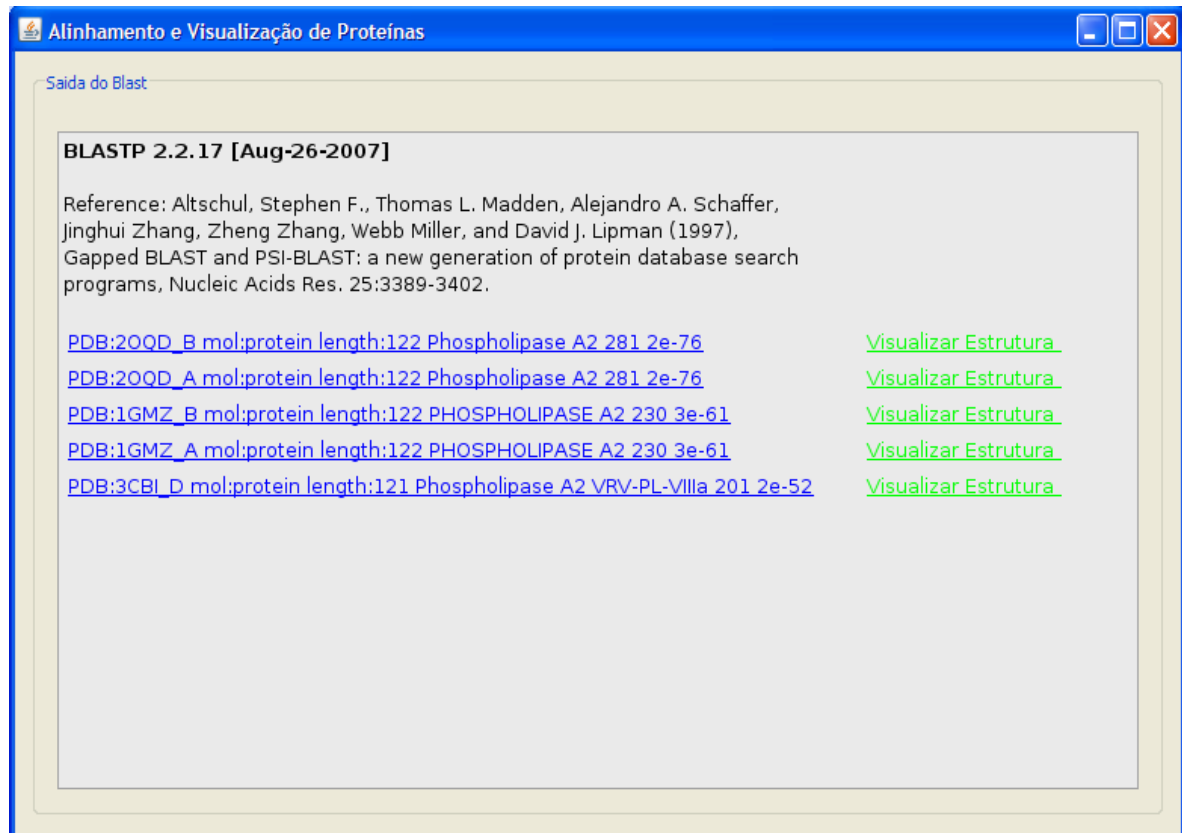
The screenshot shows a window titled "Alinhamento e Visualização de Proteínas" with a subtitle "Sistema de Apoio ao Estudo de Proteínas". It has two tabs: "Alinhamento - BLAST" (active) and "Visualizacao de Estruturas".

Under "Alinhamento - BLAST", there are several input fields and a button:

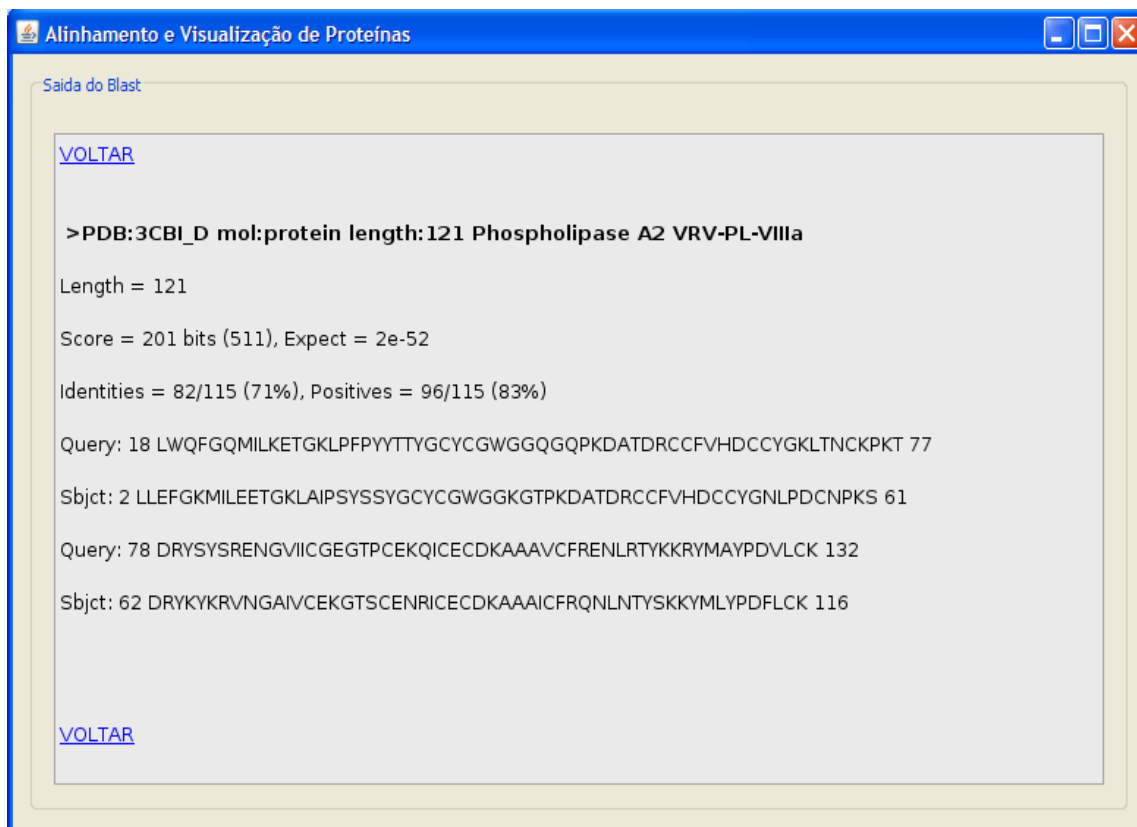
- "Entre com a sequencia Fasta" section: "Selecione o Arquivo Fasta:" with a text box containing "ktop\TCC\_testes\testes\Blast\_Atual\sequencias\_filtradas.fasta" and a file selection icon.
- "Selecione:" section: "Entre com o seu Email:" with a text box containing "larizalaura@gmail.com".
- "Entre com score minimo:" with a text box containing "20".
- "Selecione o Banco de Dados:" with a dropdown menu showing "pdb".
- "Selecione a Matriz:" with a dropdown menu showing "blosum62".
- A button labeled "Executar BLAST" with a gear icon.

**Figura 6: Interface dos Módulos de Visualização e Alinhamento – Aba 1: Alinhamento.**

As interfaces presentes nas Figuras 7 e 8 mostram a saída do BLAST para um exemplo de seqüência alinhada. Na Figura 7, caso o banco de dados para a realização do alinhamento seja o PDB, também será possível visualizar as estruturas protéicas das seqüências alinhadas.



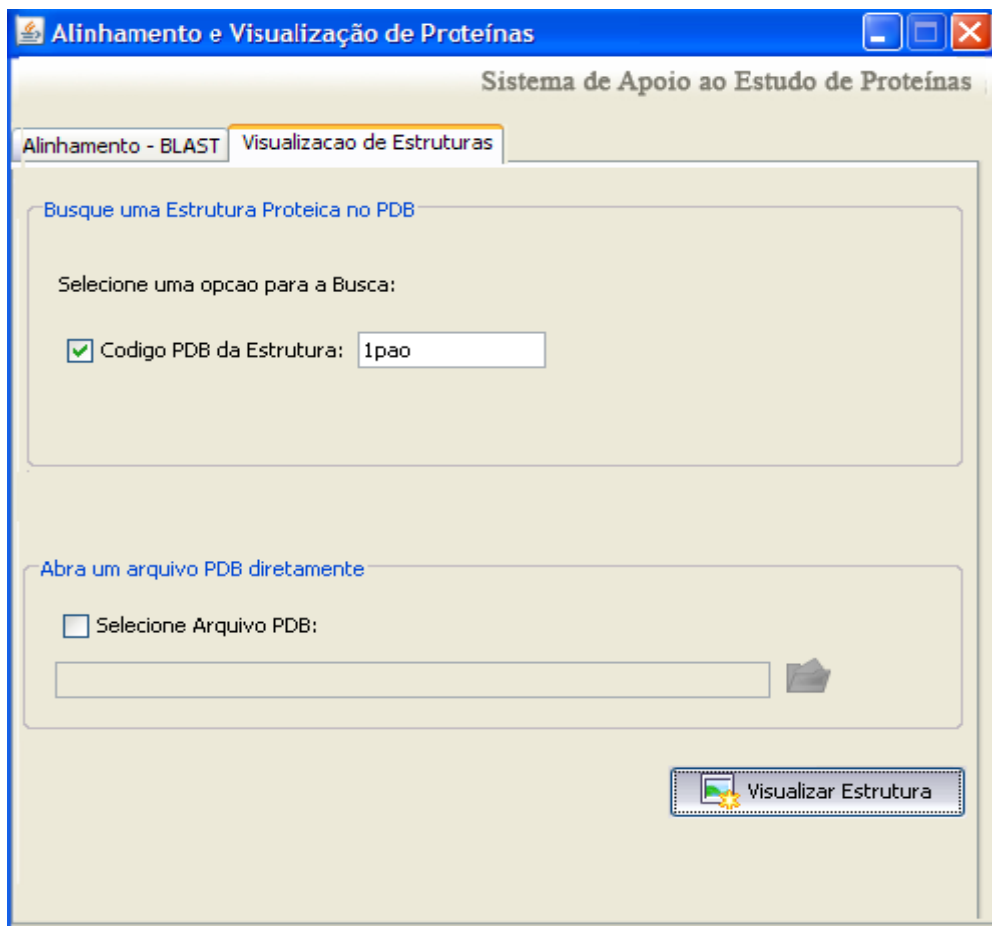
**Figura 7: Interface para visualização das seqüências alinhadas.**



**Figura 8: Interface para a visualização do alinhamento de cada seqüência.**

Na interface gráfica, presente na Figura 9, é possível realizar a visualização de estruturas protéicas. O usuário pode abrir um arquivo PDB diretamente, ou realizar busca por uma seqüência no Banco de Dados do PDB, empregando o código PDB da seqüência como chave da busca.

A busca de uma seqüência é realizada através de uma requisição HTTP (*Hypertext Transfer Protocol*) realizada ao Banco de Dados do PDB. Toda busca é efetuada utilizando o código PDB. A resposta obtida contém o arquivo PDB de interesse, que é salvo para posterior visualização através do JMol.



**Figura 9: Interface dos Módulos de Visualização e Alinhamento – Aba 2: Visualização.**

A Figura 10, mostra a visualização de um exemplo de estrutura tridimensional protéica, fornecida pelo JMol.



**Figura 10: Visualização fornecida pela ferramenta Jmol.**

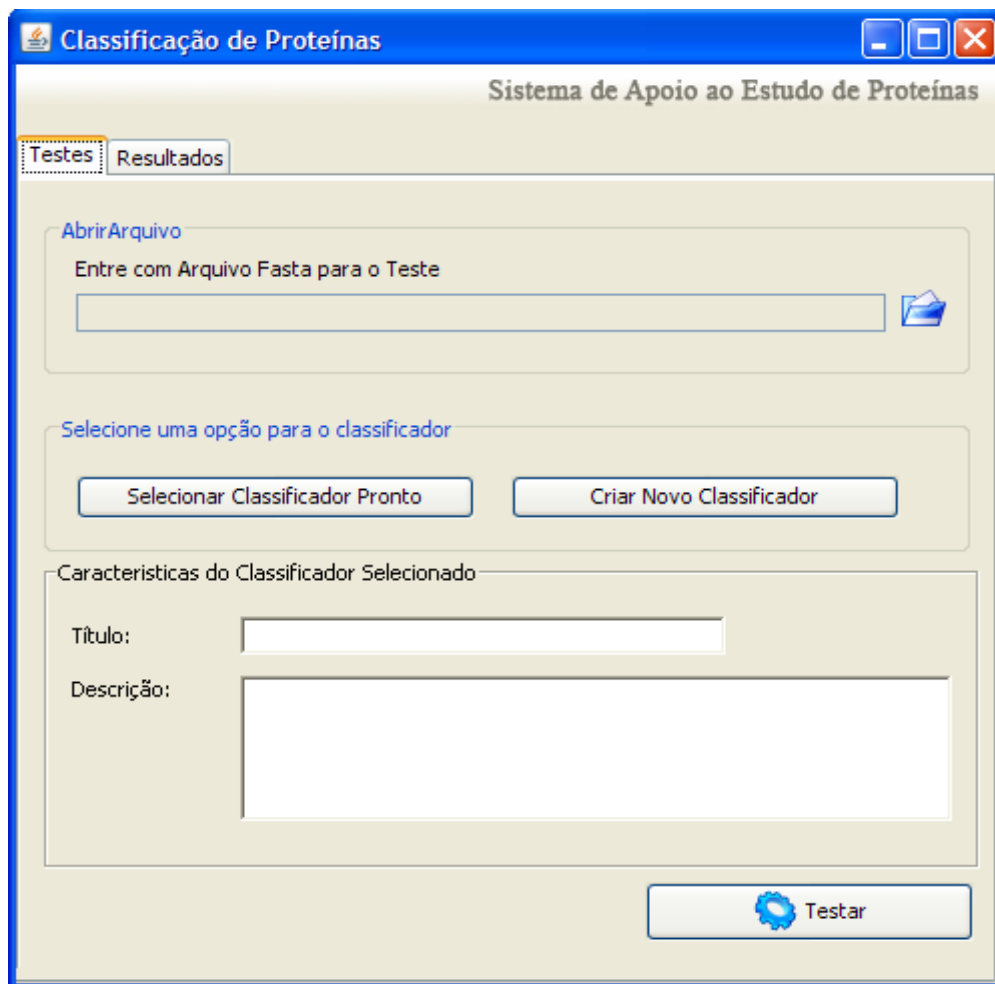
- **Módulo de Classificação**

O módulo de classificação, que pode ser acessado a partir da interface inicial (Figura 5), suas interfaces gráficas podem ser vistas a seguir.

A Figura 11 mostra a interface inicial para a realização de testes com um classificador. Para realizar um teste o usuário deverá selecionar um arquivo FASTA e em seguida um classificador para iniciar o teste. Assim que o usuário seleciona o classificador as informações como título e descrição do classificador são atualizadas na interface.

Além da possibilidade de selecionar um classificador pronto, o usuário também poderá criar seu próprio classificador, esta opção será apresentada posteriormente.

Ao selecionar arquivo e classificador o usuário poderá classificar suas seqüências FASTA, os resultados são exibidos na segunda aba desta interface.



**Figura 11: Classificação de Proteínas –Aba 1: Teste.**

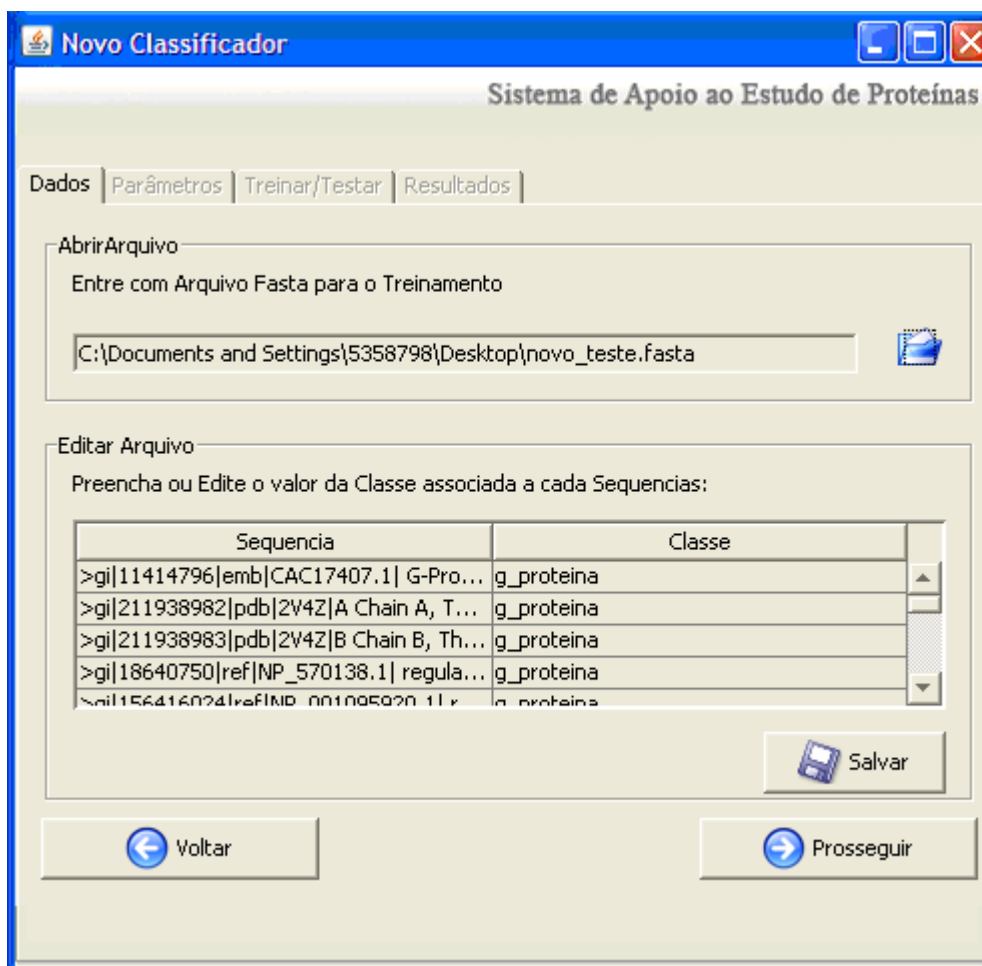
A Figura 12 mostra os resultados do teste, ou seja, a classe predita pelo classificador para cada uma das seqüências de entrada.

Sequencia	Classe Predita
> gi 73747887 NP_067673ADA...	@metaloproteinase
> gi 17865540 P58464Phosphol...	@phospholipase
> gi 73747885 NP_003465ADA...	@metaloproteinase
> gi 17433156 P82950Phosphol...	@phospholipase
> gi 117606341 NP_031426adi...	@metaloproteinase
> gi 17433157 Q90249Phosphol...	@phospholipase
> gi 143770741 NP_00107736...	@metaloproteinase
> gi 1171971 P45881Phospholi...	@phospholipase
> gi 143770755 NP_057447glyc...	@metaloproteinase
> gi 3914259 P81458Phospholi...	@phospholipase
> gi 56206987 CAI25036adisint...	@metaloproteinase
> gi 24638087 P59071Phosphol...	@phospholipase
> gi 115502351 O75078ADAM1...	@metaloproteinase
> gi 166215047 P24605Phosph...	@phospholipase

**Figura 12: Classificação de Proteínas –Aba 2: Resultados.**

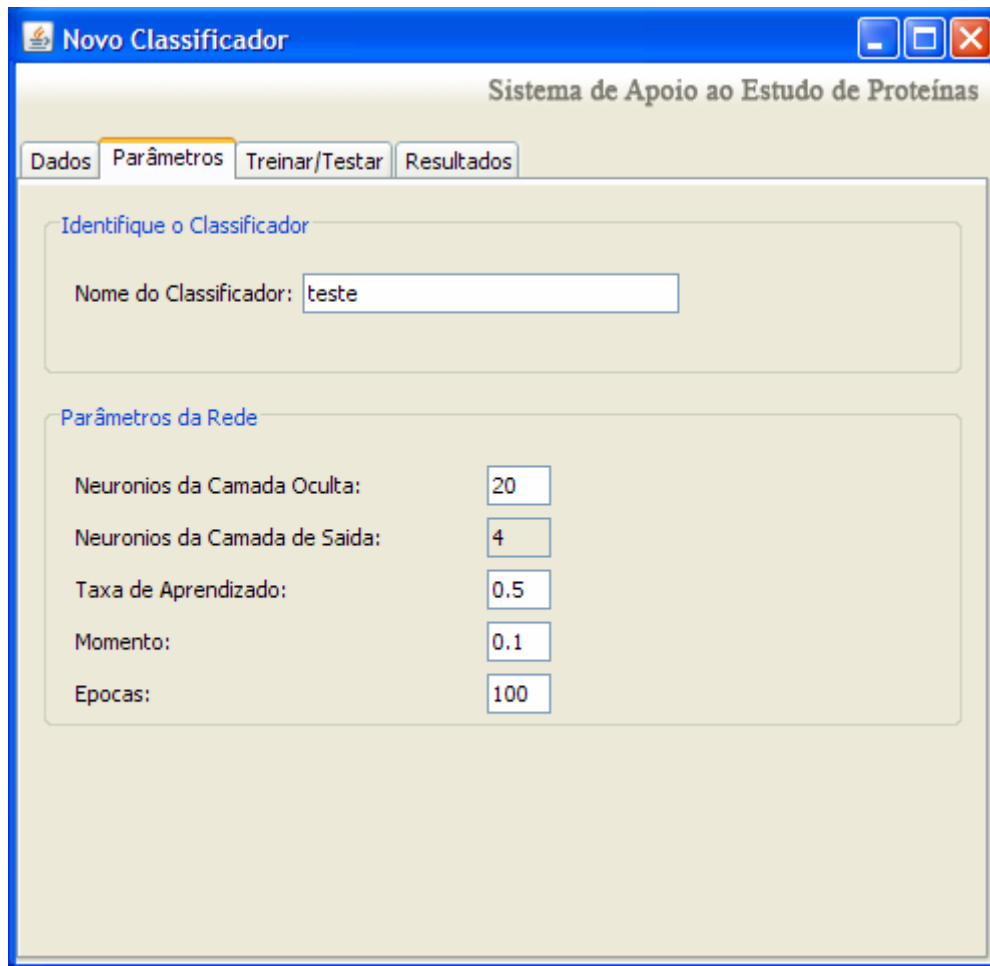
O sistema conterá classificadores prontos, porém, como mencionado anteriormente, o usuário poderá criar classificadores conforme desejar. As interfaces gráficas para criação desses classificadores são apresentadas a seguir.

A Figura 13 mostra a interface para pré-processamento de dados, onde é possível selecionar um arquivo FASTA e editá-lo acrescentando classes a cada seqüência de entrada. A adição de classes é uma tarefa necessária, uma vez que, é requerida para a realização do treinamento da rede neural.



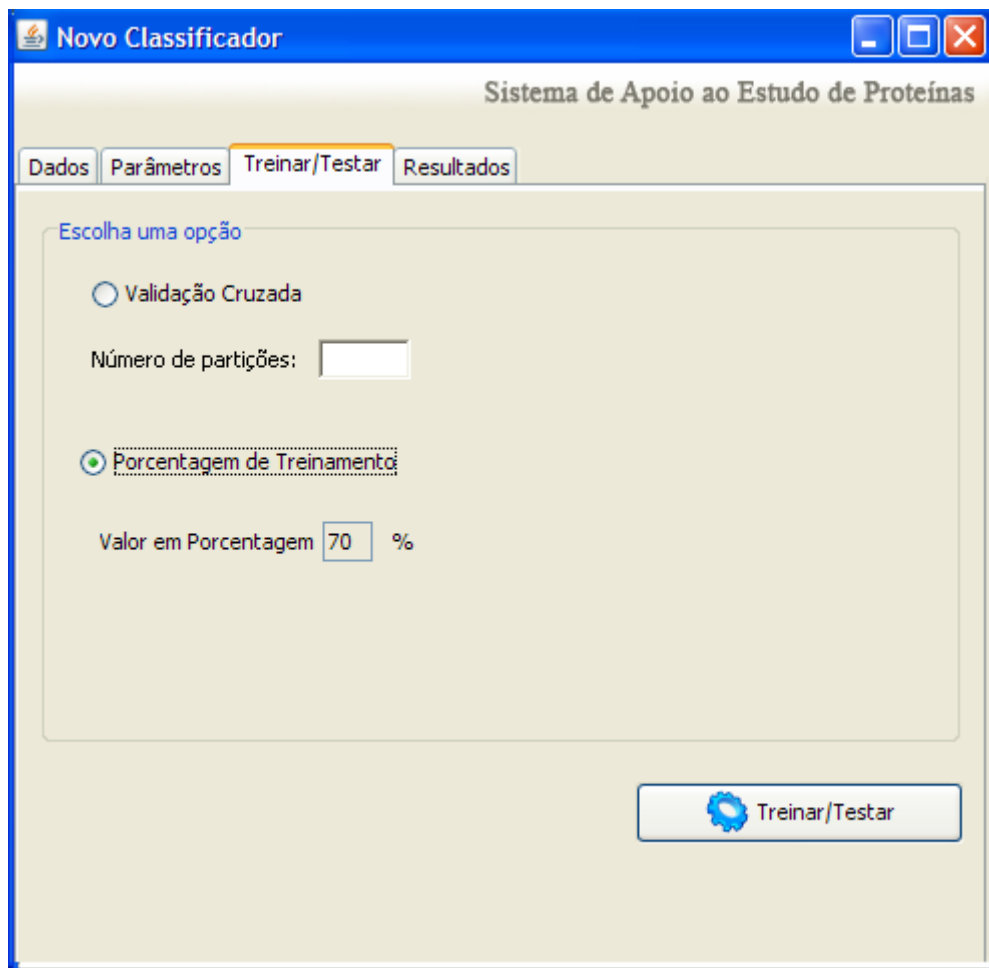
**Figura 13: – Treinar Classificador -Aba 1: Dados.**

A Figura 14 representa a interface para a configuração dos parâmetros da rede neural, tais como número de neurônios, taxa de aprendizado, momento e número de épocas de treinamento. Além disso, nesta etapa o usuário pode acrescentar um nome ao classificador.



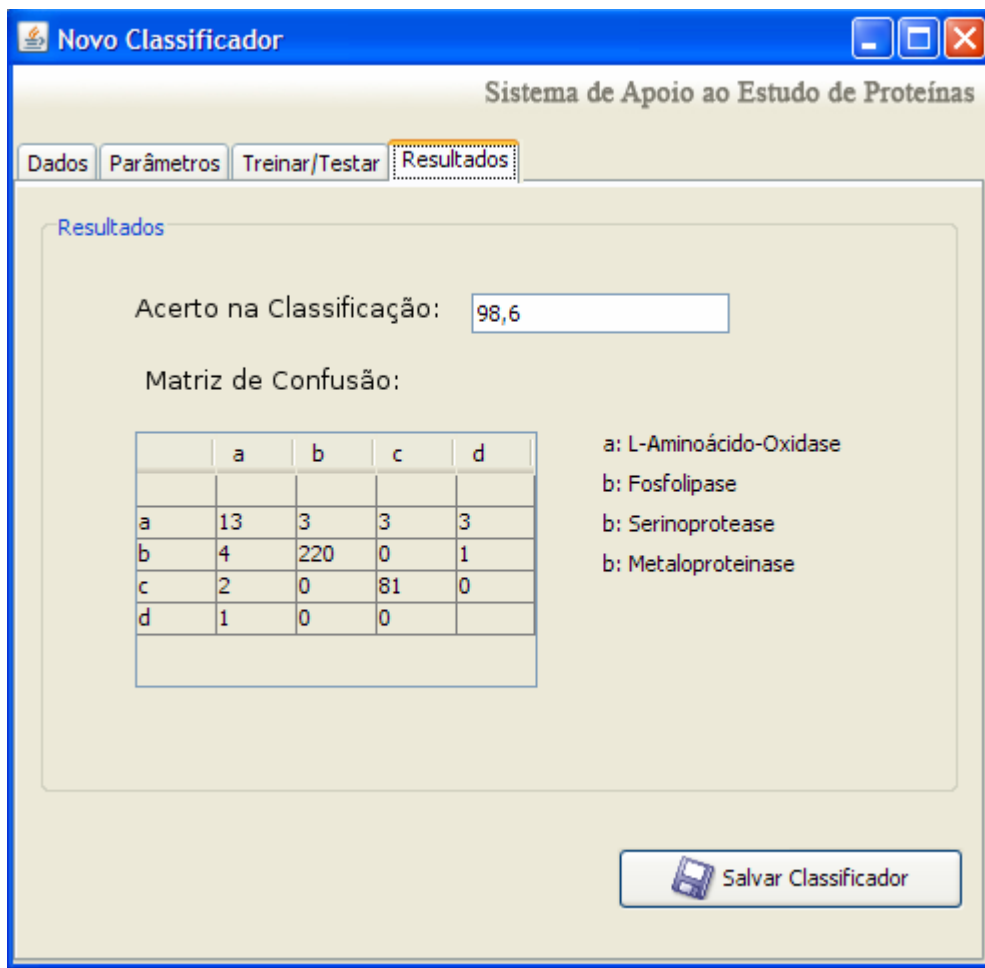
**Figura 14: Treinar Classificador–Aba 2: Parâmetros.**

Para a realização do treinamento da rede, foram disponibilizadas duas opções (Figura 15): a primeira delas utiliza o método *crossvalidation* e possibilita a escolha do número de *folds* para a realização da validação, já a segunda opção disponível separa o conjunto de exemplos em duas partes: uma delas para treinamento e outra para teste, sendo possível também determinar que porcentagem de exemplos será utilizada para o treinamento.



**Figura 15: Treinar Classificador –Aba 3: Validação.**

Após o treinamento, os resultados são expostos na interface presente na Figura 16. É possível observar a matriz de confusão do treinamento e a porcentagem de acerto obtida. Caso desejado, o classificador poderá também ser salvo para posterior utilização.



**Figura 16: Treinar Classificador –Aba 4: Resultados.**

## 5. TESTES REALIZADOS

A seguir, serão apresentados os testes efetuados para cada um dos módulos do sistema.

### 5.1. Módulo de Alinhamento

Para mostrar o funcionamento desse módulo, foi escolhida uma seqüência para a realização de alinhamento. A escolha não obedeceu nenhum critério específico. A seqüência escolhida pode ser vista na Figura 17.

```
>gi|17865540|P58464PhospholipaseA2-3 (Phosphatidylcholine2-acylhydrolase) [Bothrospirajai]  
DLWQFGQMILKETGKLPFPYYTYGGCYCGVGGRRGLGTKDDRCCYVHDCCYKKLTGCPKTDDRYSSWLDLT  
IVCGEDDPCKELCECDKAIIVCFRENLTYNKKYRYHLKPKCKADKPC
```

**Figura 17: Fosfolipase de Serpente utilizada na realização de alinhamento.**

Os bancos de dados utilizados para a realização do alinhamento foram: *Swiss-Prot* e *PDB*. Não serão exibidas todas as seqüências retornadas, devido ao grande número, desse modo, apenas as com maior *Score* no alinhamento, serão apresentadas, também, por serem mais relevantes para este estudo.

Para facilitar a visualização os resultados obtidos foram dispostos em tabelas. A identificação da seqüência foi mantida conforme retornada pelo BLAST, ou seja, obedecendo à nomenclatura que a seqüência possui no banco onde foi realizado o alinhamento. Os resultados podem ser vistos abaixo:

**Tabela 1: Seqüências com alinhamento significativo – Banco de Dados *Swiss-Prot*.**

Seqüência	Score	E-Value
SW:PA23_BOTPI P58464 Phospholipase A2-3 OS=Bothrops pirajai	279	2e-75
SW:PA2B1_BOTJR Q90249 Phospholipase A2 homolog bothropstoxin-1 OS=Bothrops jararacussu	178	8e-45
SW:PA22_BOTPI P82287 Phospholipase A2 homolog 2 OS=Bothrops pirajai	176	3e-44
SW:PA2H_BOTNE Q9IAT9 Phospholipase A2 homolog (Fragment) OS=Bothrops neuwiedi pauloensis	174	1e-43
SW:PA21B_BOTPI P58399 Phospholipase A2 homolog 1 OS=Bothrops pirajai	173	2e-43
SW:PA22_BOTMO Q9I834 Phospholipase A2 homolog 2 OS=Bothrops moojei	169	3e-42
SW:PA2H2_BOTAS P24605 Phospholipase A2 homolog 2 OS=Bothrops asper	168	6e-42

**Continuação da Tabela 1.**

SW:PA2B2_BOTJR P45881 Phospholipase A2 bothropstoxin-2 OS=Bothrops	166	2e-41
SW:PA2HA_BOTAS P0C616 Phospholipase A2 homolog 4a OS=Bothrops asper	158	8e-39
SW:PA2H1_BOTAT Q6JK69 Phospholipase A2 homolog 1 OS=Bothrops atrox	158	8e-39
SW:PA2H3_BOTAS Q9PVE3 Phospholipase A2 homolog M1-3-3 OS=Bothrops	157	1e-38
SW:PA2H_AGKAC O57385 Phospholipase A2 homolog acutohaemolysin	157	2e-38
SW:PA25_TRIGA P70090 Phospholipase A2 isozyme 5 OS=Trimeresurus flavoviridis	156	2e-38
SW:PA2H_CERGO Q8UVU7 Phospholipase A2 homolog Pgo-K49 OS=Cerrophidiongodmani	154	1e-37
SW:PA22_CERGO P81165 Phospholipase A2 homolog, myotoxin II OS=Cerrophidiongodmani	152	3e-37
SW:PA2H_ZHAMA P84776 Phospholipase A2 homolog zhaoermiatoxin	151	8e-37
SW:PA2H_ATRNM P82950 Phospholipase A2 homolog OS=Atropoides nummifer	149	3e-36
SW:PA21B_BOTMO P82114 Phospholipase A2 homolog 1 OS=Bothrops moojeni.	149	5e-36
SW:PA21_BOTAS P20474 Phospholipase A2 OS=Bothrops asper PE=1 SV=2	146	2e-35
SW:PA24_AGKHP O42187 Phospholipase A2 B OS=Agkistrodon halys	145	7e-35
SW:PA2F_AGKRH Q9PVF3 Phospholipase A2 homolog G6K49 OS=Agkistrodrodon	143	2e-34
SW:PA29_AGKHP O42188 Phospholipase A2 homolog OS=Agkistrodon halys	142	6e-34
SW:PA25_AGKHP O42189 Phospholipase A2 BA1 OS=Agkistrodon halys	142	6e-34
SW:PA2J_TRIFL P20381 Phospholipase A2 isozyme BP-I/BP-II OS= Trimeresurus flavoviridis	141	8e-34
SW:PA2H_CROAT Q8UVZ7 Phospholipase A2 homolog Cax-K49 OS=Crotalus	141	1e-33
SW:PA27_TRIGA P70089 Phospholipase A2 isozyme 7 OS= Trimeresurus flavoviridis	141	1e-33
SW:PA2H_AGKPI P04361 Phospholipase A2 homolog OS=Agkistrodon	140	1e-33
SW:PA21B_AGKHA P04417 Phospholipase A2, basic OS=Agkistrodon halys	140	2e-33
SW:PA2M_AGKCL P49121 Phospholipase A2 homolog MT1 OS=Agkistrodon	140	2e-33

De acordo com a Tabela 1, pode-se observar que as seqüências retornadas possuem alta similaridade e são também de serpentes, algumas delas pertencentes à mesma espécie da seqüência submetida. Também é possível perceber que todas as seqüências pertencem à família das Fosfolipases.

**Tabela 2: Seqüências com alinhamento significativo – Banco de Dados PDB.**

Seqüência	Score	E-Value
PDB: 1GMZ_B mol:protein length:122 PHOSPHOLIPASE A2	187	3e-48
PDB: 1GMZ_A mol:protein length:122 PHOSPHOLIPASE A2	187	3e-48
PDB: 1PC9_B mol:protein length:121 BnSP-6	181	2e-46
PDB: 1PC9_A mol:protein length:121 BnSP-6	181	2e-46
PDB: 1QLL_B mol:protein length:121 PHOSPHOLIPASE A2	178	2e-45
PDB: 1QLL_A mol:protein length:121 PHOSPHOLIPASE A2	178	2e-45
PDB: 1PA0_B mol:protein length:121 Myotoxic phospholipase A2-like	178	2e-45
PDB:1PA0_A mol:protein length:121 Myotoxic phospholipase A2-like	178	2e-45
PDB:2H8I_B mol:protein length:121 Phospholipase A2 homolog 1	177	2e-45
PDB:2H8I_A mol:protein length:121 Phospholipase A2 homolog 1	177	2e-45

A Tabela 2 indica a existência de seqüências similares, ou seja, que apresentaram alinhamento significativo com as proteínas existentes no PDB. Todas as seqüências retornadas possuem estruturas tridimensionais disponíveis no PDB. A realização do alinhamento possibilita a localização de seqüências similares, o que por sua vez, possibilitará a localização de estruturas tridimensionais possivelmente similares também.

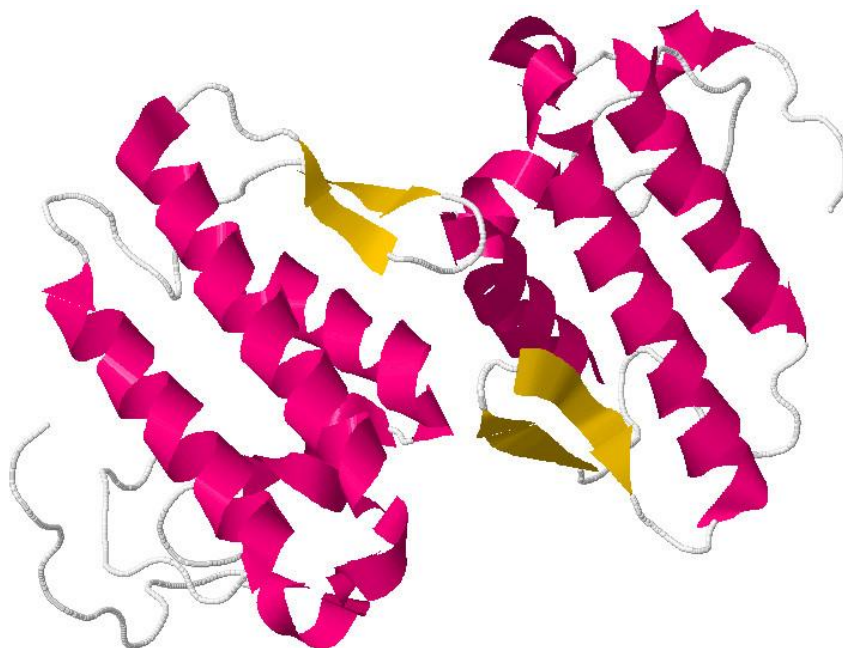
## 5.2. Módulo de Visualização

Para ilustrar o funcionamento do módulo de visualização, as estruturas tridimensionais das seqüências retornadas pelo alinhamento na Tabela 2 foram visualizadas. As visualizações podem ser vistas a seguir.

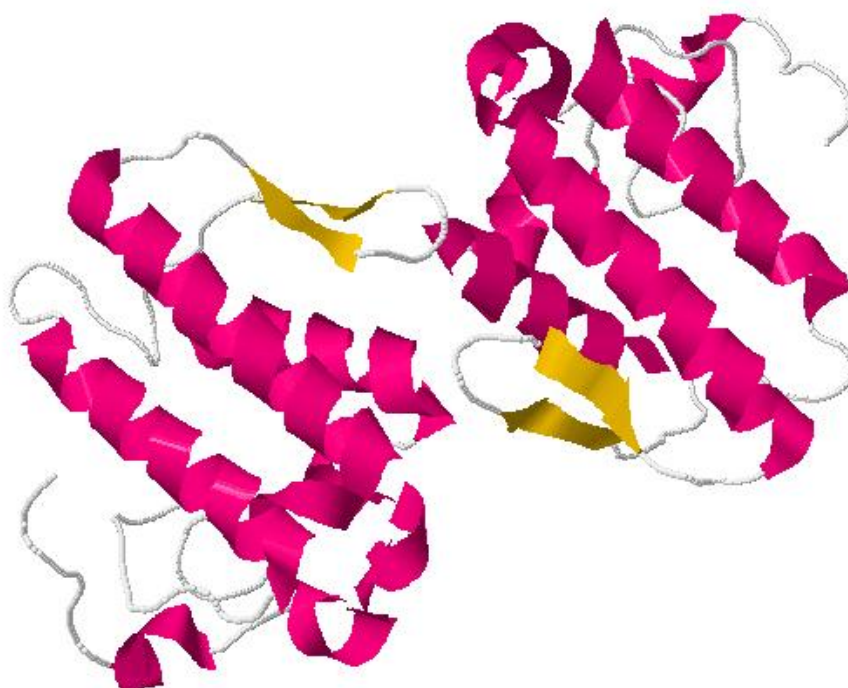


**Figura 18: PDB: 1GMZ - Phospholipase A2 homolog 1 - Cadeias A e B.**

A Figura 18 mostra as duas cadeias de um Fosfolipase A2. Trata-se das duas primeiras seqüências retornadas pelo alinhamento com o Banco de Dados do PDB presentes Tabela 2. O *Score* obtido para ambas as cadeias foi 187, enquanto que o *E-Value* foi de  $3e-48$ .

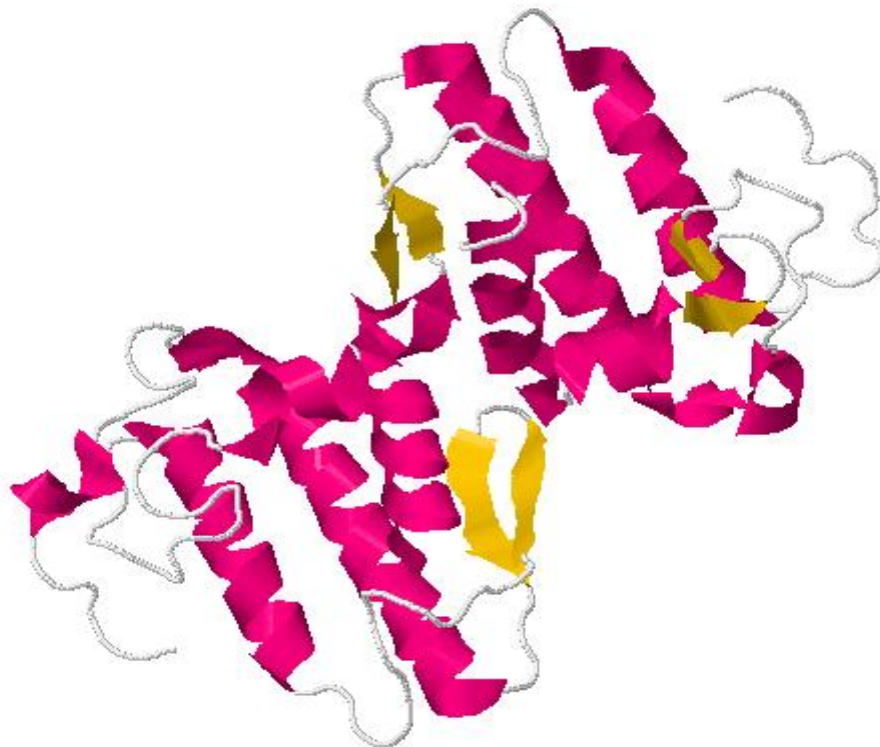


**Figura 19: PDB: 1PC9 - BnSP-6- Cadeias A e B.**



**Figura 20: PDB: 1QLL- Phospholipase A2- Cadeias A e B.**

A Figura 19 também mostra as duas cadeias de uma BnSP-6, cujo *Score* obtido para ambas as cadeias foi 181, enquanto que o *E-Value* foi de  $2e-46$ . Já a Fosfolipase encontrada na Figura 20 obteve *Score* de 181 e *E-Value* de  $2e-46$ , para as duas cadeias.



**Figura 21: PDB: 1PA0 - Myotoxic phospholipase - Cadeias A e B.**



**Figura 22: PDB: 2H8I - Phospholipase A2- Cadeias A e B.**

A estrutura protéica de Fosfolipase presente na Figura 21 obteve *Score* de 178 e *E-Value* foi de  $2e-45$  para suas duas cadeias. Já a estrutura, também de Fosfolipase, da Figura 22 obteve *Score* 177 e *E-Value* de  $2e-45$ , também para ambas as cadeias.

As estruturas apresentadas nas Figuras 18 a 22 possuem regiões conservadas, a mesma semelhança é também observada em suas estruturas tridimensionais. Apenas através da observação dessas estruturas, é possível perceber a presença de macroestruturas em regiões semelhantes, como a presença de folhas- $\beta$  invertidas em todas as estruturas, também as hélices- $\alpha$  encontram-se em posições similares, principalmente, para as três primeiras estruturas apresentadas.

A utilização do alinhamento para posterior visualização de estruturas pode trazer uma idéia da conformação de uma estrutura de interesse, uma vez que seqüências conservadas costumam apresentar estruturas semelhantes.

### **5.3.Módulo de Classificação**

Os testes do módulo de classificação objetivaram a criação de classificadores, que pudessem ser disponibilizados no sistema. Dois classificadores foram gerados e os testes de validação serão apresentados a seguir.

Além disso, a partir dos classificadores gerados, foram escolhidas seqüências que não participaram das etapas de treinamento e teste, para mostrar como poderá ser feita a classificação de seqüências desconhecidas.

A realização de alinhamentos, também foi empregada para verificar a similaridade de seqüências e buscar por seqüências que fossem similares para a geração do classificador. Os dois classificadores gerados e seus testes serão apresentados nas seções 5.3.1 e 5.3.2.

#### **5.3.1. Testes realizados com proteínas de Venenos**

Para a realização deste teste foram utilizadas quatro classes de seqüências, sendo todas de Venenos de Serpentes, são elas: *Serino protease*, *Fosfolipase*, *Metaloproteinase* e *L-aminoácido oxidase*. O número total de seqüências selecionadas foi 706 sendo o número de exemplos de cada classe distribuído da seguinte maneira:

- *Serino protease*: 83 exemplos;
- *Fosfolipase*: 225 exemplos;
- *Metaloproteinase*: 376 exemplos;
- *L-aminoácido oxidase*: 22 exemplos;

A criação de um classificador de proteínas de venenos de serpentes pode ser útil para a verificar a classe de proteínas ainda não classificadas.

### 5.3.1.1. Análise da frequência média de atributos

Para observar a frequência média de aparecimento das combinações de aminoácidos geradas pela codificação *2-gram* foram gerados gráficos. Cada um deles representa o número médio de aparições de determinada combinação de aminoácidos, dentro de uma determinada classe. Os gráficos foram obtidos a partir aplicação da codificação *2-gram* a cada classe de exemplos, em seguida, foi feita uma média para todas as seqüências de uma dada classe. Os gráficos são apresentados a seguir.

Frequência Média de Aminoácidos: L-aminoácido oxidase

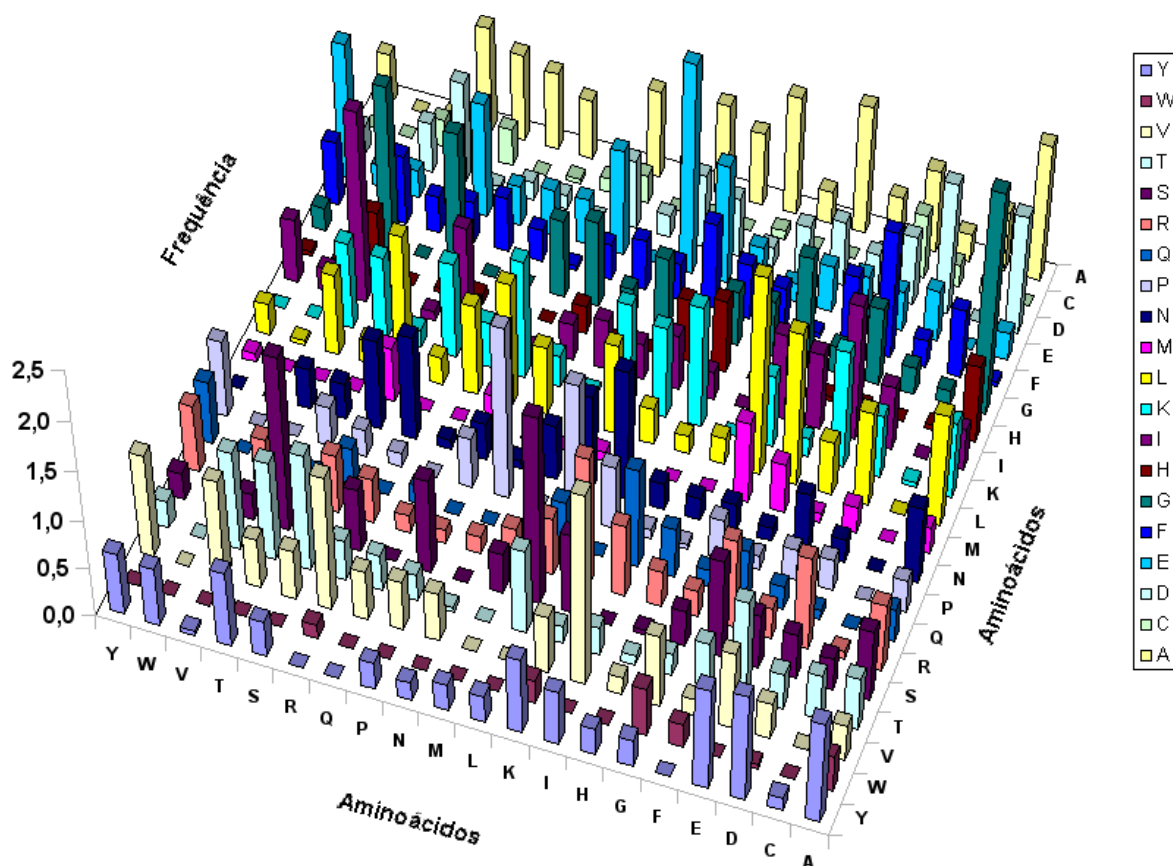


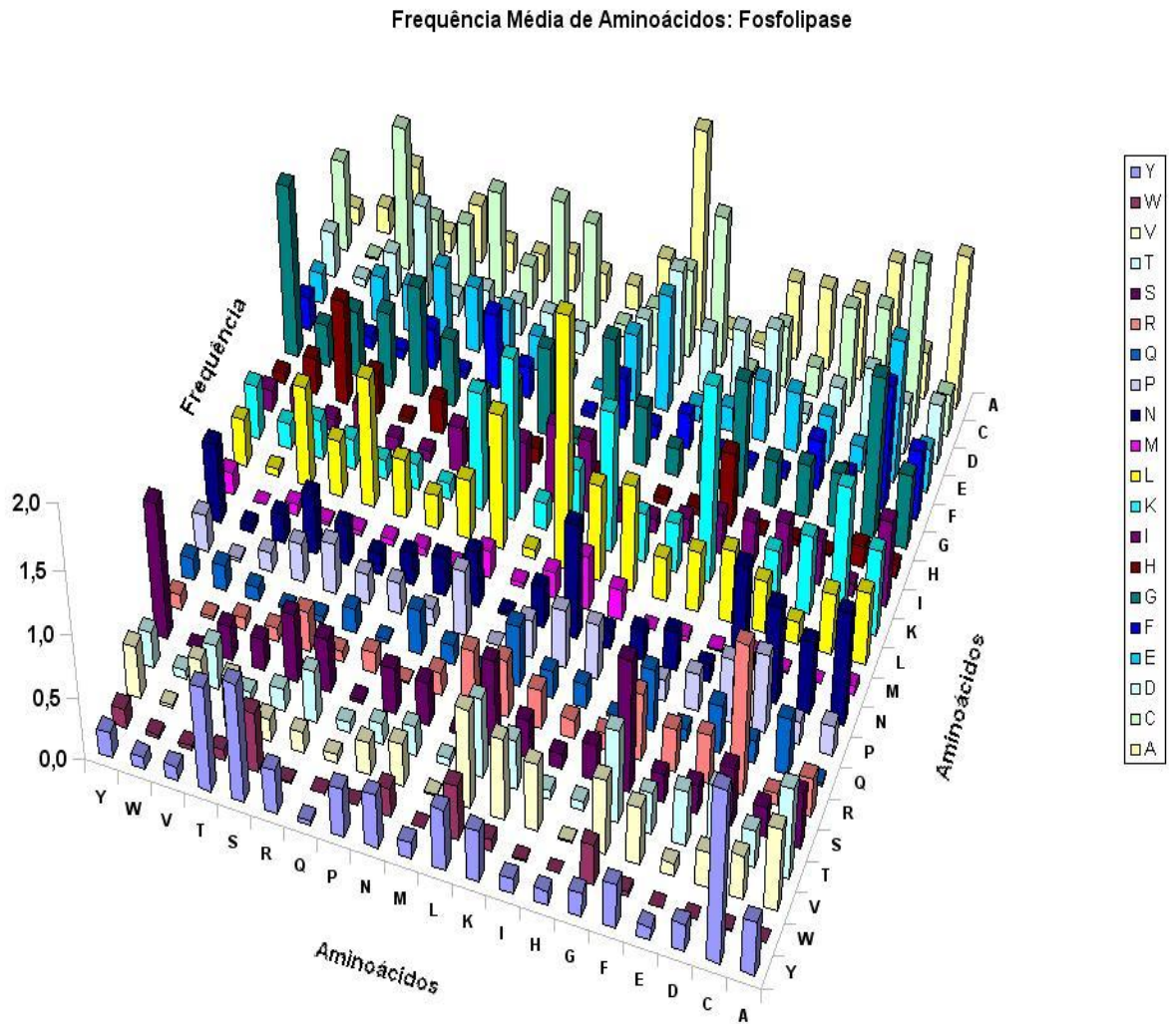
Figura 23:Frequência Média dos Aminoácidos – classe: L-aminoácido oxidase.

De acordo com o gráfico apresentado na Figura 23, podemos perceber que alguns aminoácidos apresentam frequências médias maiores que outros. Devido à angulação do gráfico, não é possível definir quais são exatamente as maiores frequências. Desse modo, para cada um dos gráficos apresentados foi gerada uma tabela contendo as 7 maiores frequências.

Para esta classe protéica, Figura 22, as 7 maiores freqüências observadas foram:

**Tabela 3: L-aminoácido oxidase – As 7 Maiores Freqüências Médias.**

VI	VG	TS	NP	LS	LE	GL
1,9091	1,6364	1,7727	1,7273	1,9091	2,0909	2,0000



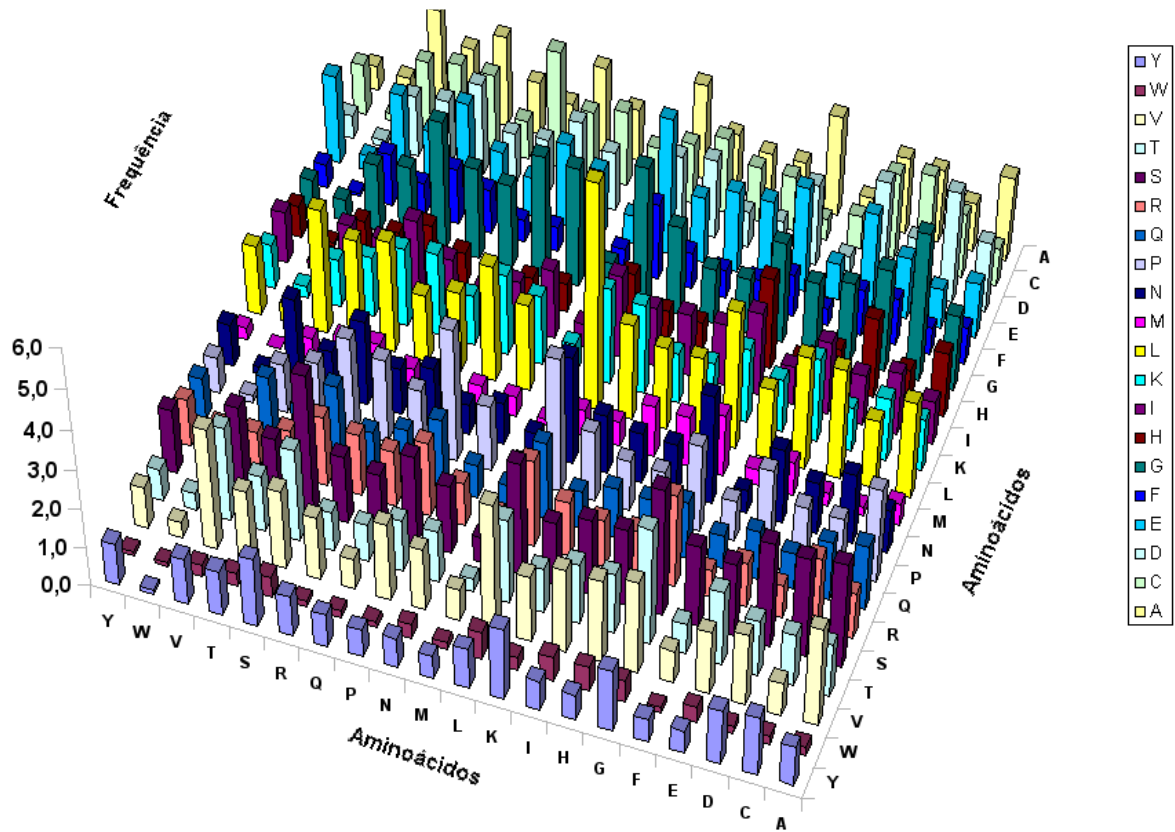
**Figura 24: Freqüência Média dos Aminoácidos – classe: Fosfolipase.**

Para o gráfico apresentado na Figura 24, os 7 maiores valores de freqüência média foram:

**Tabela 4: Fosfolipase – As 7 Maiores Frequências Médias.**

YG	NK	LL	KA	IC	GK	CY
1,3155	1,2577	1,9555	1,5244	1,1556	1,5289	1,4044

**Frequência Média de Aminoácidos: Metaloproteínase**



**Figura 25: Frequência Média dos Aminoácidos – classe: Metaloproteínase.**

Para o gráfico da Figura 25, as 7 maiores frequências foram:

**Tabela 5: Metaloproteínase. – As 7 Maiores Frequências Médias.**

ST	PP	LP	LL	LE	GL	CG
3,5130	3,3333	3,3633	5,6467	3,3693	3,4750	3,7006

### Frequência Média de Aminoácidos: Serino Protease

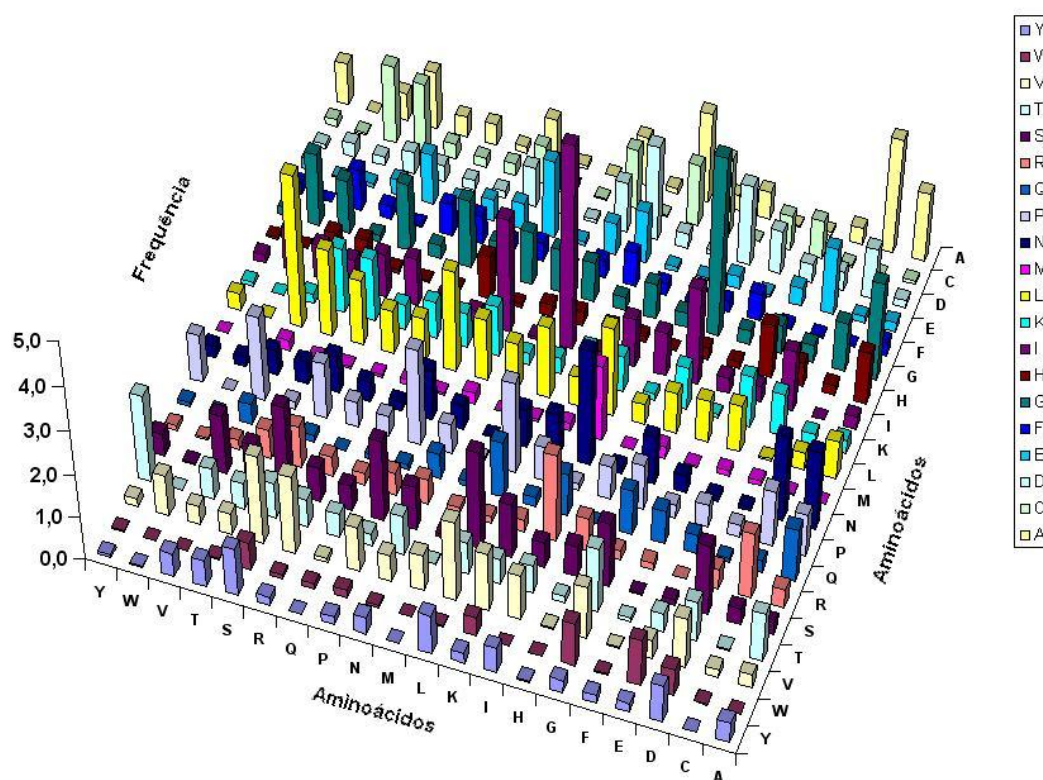


Figura 26: Frequência Média dos Aminoácidos – classe: Serino protease.

Para o gráfico da Figura 26, as 7 maiores frequências foram:

Tabela 6: Serino Protease. – As 7 Maiores Frequências Médias.

LI	GG	VL	NI	CA	PL	LS
4,6024	4,0602	3,4699	2,5783	2,5783	2,3012	2,2530

Observando os gráficos e tabelas anteriores, podemos perceber que a distribuição das combinações de aminoácidos é diferente para cada uma das quatro classes, ou seja, as combinações encontram-se distribuídas de forma diferente, sendo possível observar as combinações mais frequentes, que são mostradas nas tabelas. Também é possível perceber que algumas combinações raramente aparecem nas quatro classes, enquanto outras aparecem em quase todos os exemplos.

Os classificadores construídos, que serão utilizados no módulo de classificação, deverão ser capazes de classificar seqüências protéicas, separando regiões do espaço de atributos e rotulando classes para estas regiões.

### 5.3.1.2. Definição de Parâmetros

Para definir os parâmetros utilizados na validação da MLP implementada foram realizados alguns testes preliminares.

Os exemplos utilizados foram divididos em conjunto de treinamento e conjunto de testes, sendo que o conjunto de treinamento foi formado por 70% do total de exemplos. O número de exemplos de cada classe para os dois conjuntos pode ser visto na Tabela 7:

**Tabela 7: Número de Exemplos de Treinamento e Teste.**

Conjuntos	Classes utilizadas			
	Serinoproteases	Fosfolipases	Metaloproteinase	L-aminoácido oxidase
<b>Treinamento</b> (n° de exemplos)	61	148	275	13
<b>Testes</b> (n° de exemplos)	22	77	101	9

Testes foram realizados com variação da taxa de aprendizado e do número de neurônios da camada oculta. Os resultados obtidos podem ser vistos abaixo:

#### A. Variação da Taxa de Aprendizado

- Neurônios na camada oculta: 20
- Neurônios na camada de saída: 4 (um para cada uma das classes)
- Momento: 0,9
- Épocas de treinamento: 100

Os resultados obtidos podem ser vistos nas tabelas a seguir.

O erro na classificação representa a porcentagem de exemplos incorretamente classificados.

**Tabela 8:Variação da Taxa de Aprendizado.**

<b>Taxa de Aprendizado</b>	<b>Erro na Classificação</b>
<b>0,01</b>	5,68720%
<b>0,2</b>	1,89573%
<b>0,5</b>	1,42180 %
<b>0,9</b>	1,42180 %

Como foi possível observar o erro na classificação é maior para taxa de aprendizado igual a 0,01. A seguir, as matrizes de confusão obtidas podem ser observadas:

**Tabela 9: Matriz de Confusão para Taxa igual a 0,01.**

<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>Classificado como:</b>
0	7	0	2	<b>A = L-aminoácido oxidase</b>
0	77	0	0	<b>B = Fosfolipase</b>
0	2	22	0	<b>C = Serinoproteases</b>
0	1	0	100	<b>D = Metaloproteinase</b>

A matriz de confusão, apresentada na Tabela 9, mostra que para taxa de aprendizado igual a 0,01 são classificados incorretamente 12 exemplos.

**Tabela 10: Matriz de Confusão para Taxa igual a 0,2.**

<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>Classificado como:</b>
7	1	0	1	<b>A = L-aminoácido oxidase</b>
0	77	0	0	<b>B = Fosfolipase</b>
0	0	22	2	<b>C = Serinoproteases</b>
0	0	0	101	<b>D = Metaloproteinase</b>

A matriz de confusão, apresentada na Tabela 10, mostra que para taxa de aprendizado igual a 0,2 são classificados incorretamente 4 exemplos.

**Tabela 11: Matriz de Confusão para Taxa igual a 0,5 e 0,9.**

<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>Classificado como:</b>
8	0	0	1	<b>A = L-aminoácido oxidase</b>
0	77	0	0	<b>B = Fosfolipase</b>
0	0	22	2	<b>C = Serinoproteases</b>
0	0	0	101	<b>D = Metaloproteinase</b>

A matriz de confusão, apresentada na Tabela 11 para as taxas de aprendizado igual a 0,5 e 0,9, mostra que apenas 3 exemplos foram incorretamente classificados.

### **B. Variação dos Neurônios da Camada Oculta**

- Taxa de Aprendizado: 0,5
- Neurônios na camada de saída: 4 (um para cada uma das classes)
- Momento: 0,9
- Épocas de treinamento: 100

Os resultados obtidos podem ser vistos a seguir.

O erro na classificação representa a porcentagem de exemplos incorretamente classificados.

**Tabela 12: Variação do número de neurônios da camada oculta.**

<b>Número de Neurônios na Camada Oculta</b>	<b>Taxa de Erro na Classificação</b>
<b>20</b>	1,42180 %
<b>50</b>	4,26540%
<b>100</b>	4,26540%

**Tabela 13: Matriz de Confusão para 20 neurônios na Camada Oculta.**

<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>Classificado como:</b>
8	0	0	1	<b>A = L-aminoácido oxidase</b>
0	77	0	0	<b>B = Fosfolipase</b>
0	0	22	2	<b>C = Serinoproteases</b>
0	0	0	101	<b>D = Metaloproteinase</b>

A matriz de confusão, apresentada na Tabela 13, mostra que para 20 neurônios na camada oculta são classificados incorretamente 3 exemplos.

**Tabela 14: Matriz de Confusão para 50 neurônios na Camada Oculta.**

<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>Classificado como:</b>
2	4	0	3	<b>A = L-aminoácido oxidase</b>
0	77	0	0	<b>B = Fosfolipase</b>
0	0	22	2	<b>C = Serinoproteases</b>
0	0	0	101	<b>D = Metaloproteinase</b>

A matriz de confusão, apresentada na Tabela 14, mostra que para 50 neurônios na camada oculta são classificados incorretamente 9 exemplos.

**Tabela 15: Matriz de Confusão para 100 neurônios na Camada Oculta.**

<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>Classificado como:</b>
2	2	3	2	<b>A = L-aminoácido oxidase</b>
0	77	0	0	<b>B = Fosfolipase</b>
0	0	22	2	<b>C = Serinoproteases</b>
0	0	0	101	<b>D = Metaloproteinase</b>

A matriz de confusão, apresentada na Tabela 15, mostra que para 100 neurônios na camada oculta são classificados incorretamente 9 exemplos.

### **C. Conclusões**

Os melhores resultados obtidos foram para 20 neurônios na camada de saída e taxa de aprendizado igual a 0,5, onde apenas 3 exemplos foram classificados incorretamente.

#### **5.3.1.3. Testes de Validação utilizando Proteínas de Venenos**

Utilizando os parâmetros que obtiveram os melhores resultados nos testes anteriores foram realizados alguns testes de validação empregando o algoritmo *10-fold cross-validation*, que foi implementado.

Os testes realizados são importantes, pois diminuem o efeito da distribuição dos exemplos nos conjuntos de treinamento e teste, além disso, como os pesos da rede são iniciados aleatoriamente será feita uma média dos resultados obtidos para diferentes sementes aleatórias.

A MLP utilizando o *10-fold cross-validation* foi executada 5 vezes para diferentes valores de semente aleatória. O erro de classificação e as matrizes de confusão obtidos serão apresentados a seguir.

**Tabela 16: Erro de Classificação utilizando *10-fold cross-validation***

Semente Aleatória	Taxa de Erro na Classificação
1	2,1246 %
2	2,8328%
3	2,8328%
4	2,2662%
5	2,4079%

A seguir, serão apresentadas as matrizes de confusão geradas para os diferentes as 5 sementes aleatórias:

**Tabela 17: Matriz de Confusão para semente aleatória 1.**

A	B	C	D	Classificado como:
15	3	3	1	<b>A = L-aminoácido oxidase</b>
4	220	0	1	<b>B = Fosfolipase</b>
2	0	81	0	<b>C = Serinoproteases</b>
1	0	0	375	<b>D = Metaloproteinase</b>

**Tabela 18: Matriz de Confusão para semente aleatória 2.**

A	B	C	D	Classificado como:
9	4	6	3	<b>A = L-aminoácido oxidase</b>
3	221	0	1	<b>B = Fosfolipase</b>
2	0	81	0	<b>C = Serinoproteases</b>
1	0	0	375	<b>D = Metaloproteinase</b>

**Tabela 19: Matriz de Confusão para semente aleatória 3.**

A	B	C	D	Classificado como:
9	4	6	3	<b>A = L-aminoácido oxidase</b>
3	221	0	1	<b>B = Fosfolipase</b>
2	0	81	0	<b>C = Serinoproteases</b>
1	0	0	375	<b>D = Metaloproteinase</b>

**Tabela 20: Matriz de Confusão para semente aleatória 4.**

A	B	C	D	Classificado como:
13	3	3	3	<b>A = L-aminoácido oxidase</b>
4	221	0	0	<b>B = Fosfolipase</b>
2	0	81	0	<b>C = Serinoproteases</b>
1	0	0	375	<b>D = Metaloproteinase</b>

**Tabela 21: Matriz de Confusão para semente aleatória 5.**

A	B	C	D	Classificado como:
13	3	3	3	<b>A = L-aminoácido oxidase</b>
4	220	0	1	<b>B = Fosfolipase</b>
2	0	81	0	<b>C = Serinoproteases</b>
1	0	0	375	<b>D = Metaloproteinase</b>

O valor médio do erro de classificação para os 5 testes obtidos foi:

**Erro Médio de Classificação = 2,4929 %**

**Desvio Padrão = 0,2678 %**

De acordo com os resultados obtidos foi possível observar que a MLP implementada apresentou ótimo desempenho na classificação, em geral com taxas de acerto superiores a 90%. Dessa maneira, a MLP mostrou-se capaz de classificar corretamente entre as diferentes classes de proteínas.

#### **5.3.1.4. Testes realizados com o Classificador de Venenos de Serpentes**

A partir do classificador gerado, com proteínas de venenos de serpentes, testes foram realizados com seqüências protéicas que não participaram das etapas anteriores de treinamento e teste. Algumas das seqüências escolhidas para este teste, diferentemente, das utilizadas no treinamento, não pertencem a serpentes.

##### **a) Classificação de Seqüências obtidas a partir de um Alinhamento**

Este experimento utilizou o resultado do alinhamento da seqüência protéica de veneno presente na Figura 17, que utilizou o banco de dados o *Swiss-Prot*. O resultado deste alinhamento foi apresentado nos testes do Módulo de Alinhamento e podem ser vistos na Tabela 1. A partir da identificação das seqüências retornadas pelo alinhamento, foram selecionadas as seqüências FASTA correspondentes. A seguir, o resultado da classificação das seqüências retornadas por esse alinhamento é apresentado:

**Tabela 22: Classe Predita do Alinhamento com o Banco de Dados SWISS-Prot.**

<b>ID Sequência FASTA</b>	<b>Score</b>	<b>E-Value</b>	<b>Classe Predita</b>
>gi 17865540 sp P58464.1 PA23_BOTPI Phospholipase A2-3	279	2e-75	<b>Fosfolipase</b>
>gi 209572966 sp Q90249.3 PA2B1_BOTJR Phospholipase A2 homolog bothropstoxin-1	178	8e-45	<b>Fosfolipase</b>
>gi 17368328 sp P82287.1 PA22_BOTPI Phospholipase A2 homolog 2	176	3e-44	<b>Fosfolipase</b>
>gi 25453172 sp Q9IAT9.1 PA2H_BOTNE Phospholipase A2 homolog	174	1e-43	<b>Fosfolipase</b>
>gi 17433154 sp P58399.2 PA21B_BOTPI Phospholipase A2 homolog 1	173	2e-43	<b>Fosfolipase</b>
>gi 17865560 sp Q9I834.2 PA22_BOTMO Phospholipase A2 homolog 2	169	3e-42	<b>Fosfolipase</b>
>gi 166215047 sp P24605.3 PA2H2_BOTAS Phospholipase A2 homolog 2	168	6e-42	<b>Fosfolipase</b>
>gi 1171971 sp P45881.1 PA2B2_BOTJR Phospholipase A2 bothropstoxin-2	166	2e-41	<b>Fosfolipase</b>
>gi 166216293 sp P0C616.1 PA2HA_BOTAS Phospholipase A2 homolog 4a	158	8e-39	<b>Fosfolipase</b>
>gi 82201805 sp Q6JK69.1 PA2H1_BOTAT Phospholipase A2 homolog 1	158	8e-39	<b>Fosfolipase</b>
>gi 17433168 sp Q9PVE3.1 PA2H3_BOTAS Phospholipase A2 homolog M1-3-3	157	1e-38	<b>Fosfolipase</b>
>gi 26397573 sp O57385.1 PA2H_AGKAC Phospholipase A2 homolog acutohaemolysin	157	2e-38	<b>Fosfolipase</b>
>gi 3914265 sp P70090.1 PA25_TRIGA Phospholipase A2 isozyme 5	156	2e-38	<b>Fosfolipase</b>
>gi 26397687 sp Q8UVU7.1 PA2H_CERGO Phospholipase A2 homolog Pgo-K49	154	1e-37	<b>Fosfolipase</b>
>gi 3122600 sp P81165.1 PA22_CERGO Phospholipase A2 homolog	152	3e-37	<b>Fosfolipase</b>
>gi 115502551 sp P84776.1 PA2H_ZHAMA Phospholipase A2 homolog zhaermiatoxin	151	8e-37	<b>Fosfolipase</b>
>gi 17433156 sp P82950.1 PA2H_ATRNM Phospholipase A2 homolog	149	3e-36	<b>Fosfolipase</b>
>gi 17368325 sp P82114.1 PA21B_BOTMO Phospholipase A2 homolog 1	149	5e-36	<b>Fosfolipase</b>
>gi 166214965 sp P20474.2 PA21_BOTAS Phospholipase A2	146	2e-35	<b>Fosfolipase</b>
>gi 27151647 sp O42187.2 PA24_AGKHP Phospholipase A2 B	145	7e-35	<b>Fosfolipase</b>
>gi 27151658 sp Q9PVF3.1 PA2F_AGKRRH Phospholipase A2 homolog G6K49; AltName: CRV/TMV/DAV-K49; Flags: Precursor	143	2e-34	<b>Fosfolipase</b>
>gi 27151648 sp O42188.1 PA29_AGKHP Phospholipase A2 homolog	142	6e-34	<b>Fosfolipase</b>
>gi 27151649 sp O42189.1 PA25_AGKHP Phospholipase A2 BA1	142	6e-34	<b>Fosfolipase</b>
>gi 400717 sp P20381.2 PA2J_TRIFL Phospholipase A2 isozyme BP-I/BP-II	141	8e-34	<b>Fosfolipase</b>
>gi 26397690 sp Q8UVZ7.1 PA2H_CROAT Phospholipase A2 homolog Cax-K49	141	1e-33	<b>Fosfolipase</b>
>gi 20177995 sp P70089.1 PA27_TRIGA Phospholipase A2 isozyme 7	141	1e-33	<b>Fosfolipase</b>
>gi 129478 sp P04361.1 PA2H_AGKPI Phospholipase A2 homolog	140	1e-33	<b>Fosfolipase</b>
>gi 129398 sp P04417.1 PA21B_AGKHA Phospholipase A2, basic	140	2e-33	<b>Fosfolipase</b>
>gi 1352702 sp P49121.1 PA2M_AGKCL Phospholipase A2 homolog MT1	140	2e-33	<b>Fosfolipase</b>

De acordo com a Tabela 22, todas as seqüências retornadas pelo alinhamento foram classificadas corretamente com Fosfolipases. Ressalta-se que as seqüências obtidas desse alinhamento apresentam alta similaridade.

## b) Classificação Proteínas de Humanos

### • Experimento 1

Foram selecionadas 8 Fosfolipases de Humanos para testar o classificador gerado. A seleção dessas seqüências foi feita a partir do NCBI e não foi usado nenhum critério específico para a seleção. As seqüências utilizadas e a classe predita pela rede podem ser observadas na Tabela 23.

**Tabela 23: Proteínas de Humanos: classe preditas.**

<b>ID da Seqüência Fasta</b>	<b>Classe Predita</b>
>gi 4760647 dbj BAA77392.1  phospholipase [Homo sapiens]	Metaloproteinase
>gi 387025 gb AAA60107.1  phospholipase [Homo sapiens]	<b>Fosfolipase</b>
>gi 6453793 gb AAF09020.1 AF188625_1 phospholipase A2 [Homo sapiens]	Metaloproteinase
>gi 2392416 pdb 1KVO F Chain F, Human Phospholipase A2 Complexed With A Highly Potent Substrate Analogue	<b>Fosfolipase</b>
>gi 129483 sp P14555.2 PA2GA_HUMAN Phospholipase A2, membrane associated	<b>Fosfolipase</b>
>gi 39654124 pdb 1KQU A Chain A, Human Phospholipase A2 Complexed With A Substrate Analogue	<b>Fosfolipase</b>
>gi 7767002 pdb 1CJY B Chain B, Human Cytosolic Phospholipase A2	Metaloproteinase
>gi 17433737 sp O15496.2 PA2GX_HUMAN Group 10 secretory phospholipase A2	<b>Fosfolipase</b>

De acordo com os resultados apresentados na Tabela 23, podemos perceber que a MLP conseguiu classificar corretamente 5 das 8 Fosfolipases de humanos selecionadas, ou seja, 62,5%.

### • Experimento 2

Neste teste, a seqüência protéica de Veneno da Figura 17, anteriormente utilizada, foi alinhada com um banco de dados de proteínas de humanos. Em seguida, as 18 primeiras seqüências retornadas pelo alinhamento foram selecionadas para classificação. Os resultados do alinhamento e da classificação podem ser vistos a seguir.

**Tabela 24: Classe Predita do Alinhamento com proteínas de humanos.**

<b>Seqüência</b>	<b>Score</b>	<b>E-Value</b>	<b>Classe Predita</b>
>gi 119615312 gb EAW94906.1  phospholipase A2, group IIE [Homo sapiens]	94,7	6e-20	<b>Fosfolipase</b>
>gi 7657461 ref NP_055404.1  phospholipase A2, group IIE [Homo sapiens]	94,7	6e-20	<b>Fosfolipase</b>
>gi 4505849 ref NP_000291.1  phospholipase A2, group IIA [Homo sapiens]	94,4	7e-20	<b>Fosfolipase</b>
>gi 50295448 gb AAT73043.1  platelet phospholipase A2 [Homo sapiens]	94,4	9e-20	<b>Fosfolipase</b>
>gi 443191 pdb 1POE A Chain A, Structures Of Free And Inhibited Human Secretory Phospholipase A2 From Inflammatory Exudate	94,4	9e-20	<b>Fosfolipase</b>
>gi 38492484 pdb 1N29 A Chain A, Crystal Structure Of The N1a Mutant Of Human Group Iia Phospholipase A2	93,6	1e-19	<b>Fosfolipase</b>
>gi 46575624 gb AAH69116.1  PLA2G2E protein [Homo sapiens]	93,2	2e-19	<b>Fosfolipase</b>
>gi 50295450 gb AAT73044.1  platelet phospholipase A2 [Homo sapiens]	92,4	3e-19	<b>Fosfolipase</b>
>gi 38492482 pdb 1N28 B Chain B, Crystal Structure Of The H48q Mutant Of Human Group Iia Phospholipase A2	90,9	9e-19	<b>Fosfolipase</b>
>gi 6912596 ref NP_036532.1  phospholipase A2, group IID [Homo sapiens]	79,3	3e-15	Metaloproteinase
>gi 5771420 gb AAD51390.1 AF112982_1 group IID secretory phospholipase A2 [Homo sapiens]	78,6	4e-15	Metaloproteinase
>gi 194376344 dbj BAG62931.1  unnamed protein product [Homo sapiens]	78,6	5e-15	Metaloproteinase
>gi 189053222 dbj BAG34844.1  unnamed protein product [Homo sapiens]	78,2	6e-15	Metaloproteinase
>gi 167882816 gb ACA06110.1  phospholipase A2, group IID [Homo sapiens]	77,0	1e-14	<b>Fosfolipase</b>
>gi 119615319 gb EAW94913.1  phospholipase A2, group V, isoform CRA_c [Homo sapiens]	66,6	2e-11	Metaloproteinase
>gi 4505853 ref NP_000920.1  phospholipase A2, group V precursor [Homo sapiens]	66,6	2e-11	<b>Fosfolipase</b>
>gi 193783605 dbj BAG53516.1  unnamed protein product [Homo sapiens]	63,9	1e-10	<b>Fosfolipase</b>
>gi 4505845 ref NP_003552.1  phospholipase A2, group X [Homo sapiens]	63,2	2e-10	<b>Fosfolipase</b>

De acordo com a Tabela 24, pode se observar que o Classificador acertou 13 das 18 seqüências testadas, ou seja, 72,2 %. As 9 primeiras seqüências retornadas, que apresentam maior *Score*, foram todas classificadas corretamente.

### c) Testes Realizados com Proteínas de Abelha

Além da comparação com proteínas de humanos, foram realizadas testes com proteínas da abelha *apis mellifera*. As seqüências obtidas do NCBI são todas Fosfolipases. O resultado da classificação pode ser visto na Tabela 25:

**Tabela 25: Classe Predita do Alinhamento com de *apis mellifera*.**

ID da Seqüência Fasta	Classe Predita
>gi 58585172 ref NP_001011614.1  phospholipase A2 [Apis mellifera]	Metaloproteinase
>gi 47117012 sp Q7M4I5.1 PA2_APIDO Phospholipase A2 (Phosphatidylcholine 2-acylhydrolase)	Metaloproteinase
>gi 24638082 sp Q9BMK4.1 PA2_APICC Phospholipase A2 (Phosphatidylcholine 2-acylhydrolase)	Metaloproteinase
>gi 146400061 gb ABQ28728.1  phospholipase A2 [Apis mellifera]	Metaloproteinase
>gi 16904372 gb AAL30844.1 AF438408_1 phospholipase A2 [Apis mellifera]	Metaloproteinase
>gi 24418862 sp P00630.3 PA2_APIME Phospholipase A2 precursor (Phosphatidylcholine 2-acylhydrolase) (Allergen Api m 1) (Api m I)	Metaloproteinase
>gi 157833543 pdb 1POC A Chain A, Crystal Structure Of Bee-Venom Phospholipase A2 In A Complex With A Transition-State Analogue	Metaloproteinase
>gi 7435005 pir  A59055 phospholipase A2 (EC 3.1.1.4), venom - Indian honeybee	Metaloproteinase
>gi 110758297 ref XP_001120293.1  PREDICTED: similar to secretory Phospholipase A2 CG11124-PA, isoform A [Apis mellifera]	Metaloproteinase
>gi 66532600 ref XP_392743.2  PREDICTED: similar to phospholipase A2, activating protein [Apis mellifera]	Metaloproteinase
>gi 66521517 ref XP_395319.2  PREDICTED: similar to Peroxiredoxin-6 (Antioxidant protein 2) (1-Cys peroxiredoxin) (1-Cys PRX) (Acidic calcium-independent phospholipase A2) (aiPLA2) (Non-selenium glutathione peroxidase) (NSGPx) [Apis mellifera]	Serino Protease
>gi 48109902 ref XP_393116.1  PREDICTED: similar to GXIVsPLA2 CG17035-PA, isoform A [Apis mellifera]	<b>Fosfolipase</b>
>gi 5627 emb CAA34681.1  phospholipase A-2 [Apis mellifera]	Metaloproteinase
>gi 110773141 ref XP_395595.3  PREDICTED: similar to Kinesin-like protein at 3A CG8590-PA [Apis mellifera]	Metaloproteinase
>gi 110761217 ref XP_392825.3  PREDICTED: similar to radish CG4346-PA [Apis mellifera]	Serino Protease

**Continuação da Tabela 25.**

>gi 66555018 ref XP_392798.2  PREDICTED: similar to CG3009-PA, partial [Apis mellifera]	Metaloproteinase
>gi 66545346 ref XP_624621.1  PREDICTED: similar to CG14507-PB [Apis mellifera]	Metaloproteinase
>gi 68150168 emb CAJ09946.1  unnamed protein product [Apis mellifera]	Metaloproteinase
>gi 229378 prf 711678A phospholipase A	Metaloproteinase

Os resultados da Tabela 25 mostram que o Classificador de Venenos de Serpentes não conseguiu classificar as Fosfolipases de *Apis Mellifera*. Apenas 1 das 19 seqüências (5,3%) foi classificada corretamente.

Para investigar a possível razão para o baixo desempenho da rede na classificação das Fosfolipases da abelha *Apis Mellifera*, foi realizado um alinhamento das seqüências da Tabela 25 apenas com proteínas de Serpentes. As 5 primeiras seqüências retornadas pelo alinhamento podem ser vistas a seguir.

**Tabela 26: Alinhamento de Fosfolipases de *apis mellifera* com Proteínas de Serpentes.**

Seqüência	Score	E-Value
sp Q8JIY9.1 PA2_TRIJE RecName: Full=Phospholipase A2 jerdoxin	31,2	0,040
gb AAP48891.1  phospholipase A2 isozyme Ts-R6 [Trimeresurus stejnegeri]	29,3	0,19
pdb 1OZY A Chain A, Crystal Structure Of Phospholipase A2 (Mipla3) From Micropechis Ikaheka	28,9	0,20
sp Q9I968.1 PA22_TRIMU RecName: Full=Phospholipase A2 2 [Protobothrops mucrosquamatus]	28,9	0,25
gb AAL36975.1 AF269132_1 Lys-49 phospholipase A2 precursor [Deinagkistrodon acutus]	27,7	0,52

Como é possível observar na Tabela 26 as Fosfolipases de *apis mellifera* possuem baixa similaridade com proteínas de Serpentes, fato que pode justificar o baixo desempenho obtido pelo Classificador Venenos de Serpentes.

### 5.3.1.5. Conclusões

O classificador gerado a partir de seqüências protéicas de serpentes mostrou-se eficiente para a classificação de seqüências que possuíam maior similaridade com as utilizadas no treinamento. Seqüências com baixa similaridade apesar de serem da mesma classe protéica, não possuem muitas regiões conservadas ou em comum com as proteínas utilizadas no treinamento. Isso ocorre porque as seqüências utilizadas são todas de serpentes, desse modo, para melhorar o classificador gerado, seqüências de outras espécies poderiam ser acrescentadas a cada uma das classes.

O classificador gerado será disponibilizado no sistema. Nota-se que a realização de alinhamentos podem nortear e auxiliar o entendimento dos resultados obtidos através da classificação.

### 5.3.2. Testes realizados com outras classes protéicas

A fim disponibilizar um classificador protéico mais geral, foram realizados novos testes, desta vez, utilizando protéicas de classes diferentes e de espécies diferentes. A escolha das classes foi baseada na função biológica exercida pelas proteínas. Assim, cinco classes protéicas foram escolhidas, um total de 516 seqüências. São elas:

- a) Hemoglobina: componente do plasma sanguíneo que desempenha função de transporte de oxigênio.
- b) Ferritina: responsável pelo armazenamento de Ferro em vários organismos.
- c) Miosina: constituinte do músculo esquelético exerce função de motilidade.
- d) Queratina: desempenha função de sustentação e é constituinte de diversas estruturas como unhas, cabelo, entre outros.
- e) Proteína G: desempenha função reguladora.

A distribuição das 516 seqüências entre as classes selecionadas pode ser vista na Tabela 27.

**Tabela 27: Número de Seqüências utilizadas.**

	Classes utilizadas				
	Hemoglobina	Ferritina	Miosina	Queratina	Proteína G
n° de exemplos	99	100	100	98	119

#### 5.3.2.1. Parâmetros utilizados

Os parâmetros utilizados foram os mesmos anteriormente testados. São eles:

- Taxa de Aprendizado: 0,5

- Neurônios na camada de saída: 4 (um para cada uma das classes)
- Momento: 0,9
- Épocas de treinamento: 100

### 5.3.2.2. Validação

A MLP utilizando o *10-fold cross-validation* foi executada 5 vezes para diferentes valores de semente aleatória. O erro de classificação e as matrizes de confusão obtidos são apresentados a seguir.

**Tabela 28: Erro de Classificação utilizando *10-fold cross-validation***

Semente Aleatória	Taxa de Erro na Classificação
1	13,0449%
2	13,4615%
3	13,6057%
4	12,4839%
5	16,7467%

A seguir, serão apresentadas as matrizes de confusão geradas para os diferentes as 5 sementes aleatórias:

**Tabela 29: Matriz de Confusão para semente aleatória 1.**

A	B	C	D	E	Classificado como:
96	2	0	1	1	<b>A = Ferritina</b>
9	67	17	5	1	<b>B = Hemoglobina</b>
0	2	114	2	1	<b>C = Proteína G</b>
6	2	4	82	4	<b>D = Queratina</b>
1	2	1	6	90	<b>E = Miosina</b>

**Tabela 30: Matriz de Confusão para semente aleatória 2.**

A	B	C	D	E	Classificado como:
95	2	0	2	1	<b>A = Ferritina</b>
6	69	17	6	1	<b>B = Hemoglobina</b>
0	3	113	1	2	<b>C = Proteína G</b>
5	2	3	81	7	<b>D = Queratina</b>
1	0	2	9	88	<b>E = Miosina</b>

**Tabela 31: Matriz de Confusão para semente aleatória 3.**

<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>Classificado como:</b>
93	2	0	3	2	<b>A = Ferritina</b>
5	79	7	8	0	<b>B = Hemoglobina</b>
1	2	109	3	4	<b>C = Proteína G</b>
3	2	2	80	11	<b>D = Queratina</b>
1	0	1	7	91	<b>E = Miosina</b>

**Tabela 32: Matriz de Confusão para semente aleatória 4.**

<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>Classificado como:</b>
93	2	2	3	0	<b>A = Ferritina</b>
14	68	7	2	8	<b>B = Hemoglobina</b>
3	5	81	9	0	<b>C = Proteína G</b>
1	1	5	92	1	<b>D = Queratina</b>
2	1	5	4	88	<b>E = Miosina</b>

**Tabela 33: Matriz de Confusão para semente aleatória 5.**

<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>Classificado como:</b>
91	5	0	2	2	<b>A = Ferritina</b>
12	56	20	8	3	<b>B = Hemoglobina</b>
0	3	111	4	1	<b>C = Proteína G</b>
9	1	3	81	4	<b>D = Queratina</b>
1	0	1	7	91	<b>E = Miosina</b>

O valor médio do erro de classificação para os 5 testes obtidos foi:

**Erro Médio de Classificação = 13,8686%**

**Desvio Padrão = 1,1512 %**

De acordo com os resultados obtidos foi possível observar que a MLP implementada apresentou bom desempenho na classificação. A taxa média de acerto para as cinco sementes foi de 86,1314 % . Dessa maneira, a MLP mostrou-se capaz de classificar corretamente entre as diferentes classes de proteínas.

Destaca-se que a maior quantidade de erros foi obtida para a classe de proteínas Hemoglobina. As seqüências de Hemoglobina incorretamente classificadas foram em sua maioria classificados como sendo Ferritinas.

### 5.3.2.3. Testes realizados com o Classificador

A partir do classificador gerado, testes foram realizados com seqüências protéicas que não participaram do treinamento. As seqüências testadas foram escolhidas de acordo com as classes utilizadas no treinamento. Todas as seqüências escolhidas são de Humanos. Para cada uma das classes utilizadas no treinamento foram escolhidas 15 seqüências para a realização dos testes. A seguir, são apresentados os resultados dos testes para cada uma das classes utilizadas.

#### a) Hemoglobina

Tabela 34: Classificação de Hemoglobinas de Humanos.

Seqüência Testada	Classe Predita
>gi 56749856 sp P68871.2 HBB_HUMAN Hemoglobin subunit beta	Hemoglobina
>gi 109893891 gb ABG47031.1  hemoglobin [Homo sapiens]	Hemoglobina
>gi 57013850 sp P69905.2 HBA_HUMAN Hemoglobin subunit alpha	Hemoglobina
>gi 56749860 sp P69891.2 HBG1_HUMAN Hemoglobin subunit gamma-1 (Hemoglobin gamma-1 chain) (Gamma-1-globin) (Hemoglobin gamma-A chain) (Hb F Agamma)	Hemoglobina
>gi 56749861 sp P69892.2 HBG2_HUMAN Hemoglobin subunit gamma-2	Hemoglobina
>gi 122713 sp P02042.2 HBD_HUMAN Hemoglobin subunit delta	Hemoglobina
>gi 239718 gb AAB20440.1  hemoglobin beta chain; beta-globin [Homo sapiens]	Queratina
>gi 71370292 gb AAZ30391.1  hemoglobin beta chain [Homo sapiens]	Hemoglobina
>gi 4378804 gb AAD19696.1  hemoglobin beta chain [Homo sapiens]	Hemoglobina
>gi 13958153 gb AAK50822.1 AF363956_1 hemoglobin alpha 2 [Homo sapiens]	Ferritina
>gi 576044 pdb 1CBM D Chain D, The 1.8 Angstrom Structure Of Carbonmonoxy-Beta4 Hemoglobin: Analysis Of A Homotetramer With The R Quaternary Structure Of Liganded Alpha2beta2 Hemoglobin	Hemoglobina
>gi 576043 pdb 1CBM C Chain C, The 1.8 Angstrom Structure Of Carbonmonoxy-Beta4 Hemoglobin: Analysis Of A Homotetramer With The R Quaternary Structure Of Liganded Alpha2beta2 Hemoglobin	Hemoglobina
>gi 576042 pdb 1CBM B Chain B, The 1.8 Angstrom Structure Of Carbonmonoxy-Beta4 Hemoglobin: Analysis Of A Homotetramer With The R Quaternary Structure Of Liganded Alpha2beta2 Hemoglobin	Hemoglobina

**Continuação da Tabela 34.**

>gi 576041 pdb 1CBM A Chain A, The 1.8 Angstrom Structure Of Carbonmonoxy-Beta4 Hemoglobin: Analysis Of A Homotetramer With The R Quaternary Structure Of Liganded Alpha2beta2 Hemoglobin	<b>Hemoglobina</b>
>gi 13650074 gb AAK37554.1 AF349571_1 hemoglobin alpha-1 globin chain [Homo sapiens]	<b>Hemoglobina</b>

Na Tabela 34, é possível observar que 13 das 15 Hemoglobinas (86,7%) foram classificadas corretamente.

**b) Ferritina**

**Tabela 35: Classificação de Ferritinas de Humanos.**

<b>Seqüência Testada</b>	<b>Classe Predita</b>
>gi 306744 gb AAA35832.1  ferritin	<b>Ferritina</b>
>gi 20149498 ref NP_000137.2  ferritin, light polypeptide [Homo sapiens]	<b>Ferritina</b>
>gi 29126241 ref NP_803431.1  mitochondrial ferritin [Homo sapiens]	Miosina
>gi 56682959 ref NP_002023.2  ferritin, heavy polypeptide 1 [Homo sapiens]	<b>Ferritina</b>
>gi 13994244 ref NP_114100.1  ferritin, heavy polypeptide-like 17 [Homo sapiens]	<b>Ferritina</b>
>gi 42794548 gb AAS45711.1  ferritin light polypeptide variant [Homo sapiens]	<b>Ferritina</b>
>gi 9621744 gb AAF89523.1 AF088851_1 ferritin heavy chain subunit [Homo sapiens]	<b>Ferritina</b>
>gi 109172121 gb AAI00771.1  Ferritin, heavy polypeptide-like 17 [Homo sapiens]	<b>Ferritina</b>
>gi 109172117 gb AAI00770.1  Ferritin, heavy polypeptide-like 17 [Homo sapiens]	<b>Ferritina</b>
>gi 109171999 gb AAI00769.1  Ferritin, heavy polypeptide-like 17 [Homo sapiens]	<b>Ferritina</b>
>gi 119619466 gb EAW99060.1  ferritin, heavy polypeptide-like 17 [Homo sapiens]	<b>Ferritina</b>
>gi 119594407 gb EAW74001.1  ferritin, heavy polypeptide 1, isoform CRA_a [Homo sapiens]	<b>Ferritina</b>
>gi 119594406 gb EAW74000.1  ferritin, heavy polypeptide 1, isoform CRA_h [Homo sapiens]	<b>Ferritina</b>
>gi 119594405 gb EAW73999.1  ferritin, heavy polypeptide 1, isoform CRA_g [Homo sapiens]	Miosina
>gi 119594404 gb EAW73998.1  ferritin, heavy polypeptide 1, isoform CRA_a [Homo sapiens]	<b>Ferritina</b>

A Tabela 35 mostra que 13 das 15 seqüências da proteína Ferritina (86,7%) foram classificadas corretamente.

c) **Miosina**

**Tabela 36: Classificação de Miosinas.**

Seqüência Testada	Classe Predita
>gi 558669 emb CAA86293.1  Myosin [Homo sapiens]	<b>Miosina</b>
>gi 531143 gb AAA20901.1  myosin	<b>Miosina</b>
>gi 531142 gb AAA20912.1  myosin	<b>Miosina</b>
>gi 531141 gb AAA20911.1  myosin	<b>Miosina</b>
>gi 531140 gb AAA20910.1  myosin	<b>Miosina</b>
>gi 531139 gb AAA20909.1  myosin	<b>Miosina</b>
>gi 531138 gb AAA20908.1  myosin	Queratina
>gi 531137 gb AAA20907.1  myosin	<b>Miosina</b>
>gi 531136 gb AAA20906.1  myosin	<b>Miosina</b>
>gi 531135 gb AAA20905.1  myosin	<b>Miosina</b>
>gi 531134 gb AAA20904.1  myosin	<b>Miosina</b>
>gi 531133 gb AAA20903.1  myosin	<b>Miosina</b>
>gi 531132 gb AAA20902.1  myosin	<b>Miosina</b>
>gi 531130 gb AAA20900.1  myosin	<b>Miosina</b>
>gi 116284396 ref NP_001070654.1  myosin, heavy chain 14 isoform 1 [Homo sapiens]	<b>Miosina</b>

A Tabela 36 mostra que 14 das 15 seqüências da proteína Ferritina (93,3%) foram classificadas corretamente.

d) **Queratina**

**Tabela 37: Classificação de Queratinas.**

Seqüência Testada	Classe Predita
>gi 7717238 gb AAB30058.2  keratin [Homo sapiens]	<b>Queratina</b>
>gi 386848 gb AAB59562.1  keratin	<b>Queratina</b>
>gi 1816454 dbj BAA09320.1  keratin [Homo sapiens]	<b>Queratina</b>
>gi 2190400 emb CAA73943.1  keratin [Homo sapiens]	<b>Queratina</b>
>gi 1200072 emb CAA31695.1  keratin [Homo sapiens]	<b>Queratina</b>
>gi 6685564 sp O76011.1 KRT34_HUMAN Keratin, type I cuticular Ha4 (Hair keratin, type I Ha4) (Keratin-34)	<b>Queratina</b>
>gi 6685563 sp O76009.1 KT33A_HUMAN Keratin, type I cuticular Ha3-I (Hair keratin, type I Ha3-I) (Keratin-33A)	<b>Queratina</b>
>gi 244509 gb AAB21315.1  keratin 10 V2 subdomain 142 amino acid variant [human, Peptide Partial, 142 aa]	<b>Queratina</b>

**Continuação da Tabela 37.**

>gi 244508 gb AAB21314.1  keratin 10 V2 subdomain 128 amino acid variant [human, Peptide Partial, 128 aa]	<b>Queratina</b>
>gi 244507 gb AAB21313.1  keratin 10 V2 subdomain 117 amino acid variant [human, Peptide Partial, 117 aa]	<b>Queratina</b>
>gi 461090 gb AAB29431.1  type II keratin K5 {Arg331Cys mutation, L12 linker domain} [human, EBS-WC patient, keratinocyte culture, Peptide Partial Mutant, 21 aa]	<b>Queratina</b>
>gi 399678 gb AAB27402.1  keratin 5 {exon 7} [human, Peptide Partial Mutant, 19 aa]	Proteína G
>gi 386644 gb AAB27488.1  type Ia hair keratin a3 [human, Peptide, 404 aa]	Queratina
>gi 1245902 gb AAB35666.1  keratin 13, K13 {Leu15Pro, alpha-helical rod-domain} [human, white sponge nevus patient, Peptide Partial Mutant, 35 aa]	Miosina
>gi 1332731 gb AAB36224.1  keratin 14, K14 {Y129D, rod domain} [human, Dowling-Meara epidermolysis bullosa simplex patient, blood, Peptide Partial Mutant, 16 aa]	<b>Queratina</b>

Na Tabela 37, para as seqüências protéicas de Queratina, 12 das 15 seqüências (80%) foram classificadas corretamente.

**e) Proteína G**

**Tabela 38: Classificação de Proteínas G.**

<b>Seqüência Testada</b>	<b>Classe Predita</b>
>gi 55959535 emb CAI15143.1  regulator of G-protein signaling 7 [Homo sapiens]	<b>Proteína G</b>
>gi 55959534 emb CAI15142.1  regulator of G-protein signaling 7 [Homo sapiens]	<b>Proteína G</b>
>gi 55959533 emb CAI15141.1  regulator of G-protein signaling 7 [Homo sapiens]	<b>Proteína G</b>
>gi 55959532 emb CAI15140.1  regulator of G-protein signaling 7 [Homo sapiens]	<b>Proteína G</b>
>gi 55959388 emb CAI16821.1  regulator of G-protein signaling 7 [Homo sapiens]	<b>Proteína G</b>
>gi 119590495 gb EAW70089.1  regulator of G-protein signalling 7, isoform CRA_d [Homo sapiens]	<b>Proteína G</b>
>gi 119590494 gb EAW70088.1  regulator of G-protein signalling 7, isoform CRA_c [Homo sapiens]	<b>Proteína G</b>
>gi 119590493 gb EAW70087.1  regulator of G-protein signalling 7, isoform CRA_b [Homo sapiens]	<b>Proteína G</b>
>gi 119590492 gb EAW70086.1  regulator of G-protein signalling 7, isoform CRA_a [Homo sapiens]	<b>Proteína G</b>
>gi 55959387 emb CAI16820.1  regulator of G-protein signaling 7 [Homo sapiens]	<b>Proteína G</b>
>gi 55959386 emb CAI16819.1  regulator of G-protein signaling 7 [Homo sapiens]	<b>Proteína G</b>
>gi 55959385 emb CAI16818.1  regulator of G-protein signaling 7 [Homo sapiens]	<b>Proteína G</b>
>gi 134288847 ref NP_722581.4  G-protein coupled receptor 111 [Homo sapiens]	Miosina
>gi 51036603 ref NP_061329.3  G-protein gamma-12 subunit [Homo sapiens]	Miosina
<b>P</b>	
>gi 14336688 gb AAK61221.1 AE006463_1 regulator of G protein signalling 11 [Homo sapiens]	Miosina

A Tabela 38 mostra que 12 das 15 Proteínas G (80%) foram classificadas corretamente.

#### **5.3.2.4. Conclusões**

O classificador gerado a partir de proteínas das classes: Hemoglobina, Ferritina, Miosina, Queratina e Proteína G também obteve um bom desempenho nos testes realizados, com acertos iguais ou superiores a 80% .

Neste caso, como as seqüências selecionadas foram de diferentes espécies, o classificador tornou-se mais geral e eficiente na classificação de proteínas com as quais não foi treinado.

A utilização de diferentes espécies na geração de classificadores pode facilitar a classificação de acordo com os motivos, ou regiões que caracterizam uma classe protéica. Quando a classificação é realizada apenas com uma, ou poucas espécies o classificador poderá considerar regiões similares não importantes para a classificação daquela classe, uma vez que essas seqüências da mesma espécie ou de espécies semelhantes apresentarão alta similaridade de suas seqüências, este é o caso das proteínas das diversas espécies de serpentes.

Devido ao seu bom desempenho, este classificador também será disponibilizado no sistema.

## 6. CONCLUSÕES

O Sistema de Apoio ao Estudo de Proteínas foi construído e testado, conforme determinado na especificação de requisitos do sistema. Os testes realizados evidenciaram a importância da existência dos três módulos e sua relação entre si. Assim, o Módulo de Alinhamento mostrou-se de grande importância para a interpretação dos resultados obtidos no Módulo de Classificação, bem como útil para a busca de estruturas do Módulo de Visualização.

A realização de alinhamentos para a busca de estruturas tridimensionais mostrou que é possível obter uma ideia da conformação de uma proteína, cuja estrutura ainda é desconhecida.

A geração do classificador de venenos de serpentes mostrou que a MLP é capaz de classificar adequadamente seqüências protéicas de outra espécie, a qual não tenha sido utilizada no treinamento. Essa generalização ocorreu para seqüências que possuem similaridade com as seqüências de treinamento. No entanto, para seqüências com baixa similaridade, a MLP não obteve um bom desempenho, o que ocorreu com as seqüências de Fosfolipases de *Apis Mellifera*.

A relação entre similaridade e desempenho na classificação indica que a MLP utiliza informações de regiões conservadas. A utilização dessa informação pode ser melhorada através da adição de seqüências de diferentes espécies, de forma que a MLP utilize apenas as regiões conservadas responsáveis pela identificação de uma classe. As seqüências de venenos de serpente utilizadas possuem alta similaridade entre si, o que indica a presença de várias regiões conservadas, além das responsáveis pela identificação de sua classe protéica. Essas várias regiões, no entanto, não são comuns as proteínas de *Apis Mellifera*. Desse modo, para melhorar o classificador gerado, a adição de seqüências de diferentes espécies será essencial.

O classificador baseado nas funções biológicas desempenhadas pelas proteínas, também obteve bom desempenho nos testes realizados. Neste caso, o classificador gerado também se mostrou eficiente na classificação de seqüências não utilizadas no treinamento.

Os testes também sugeriram mudanças no sistema, como, por exemplo, a inclusão da possibilidade de escolha de bancos de dados específicos de uma dada espécie. Essa opção ainda não implementada no sistema mostrou-se útil durante a realização dos testes. Os alinhamentos realizados com uma espécie específica foram feitos diretamente no NCBI.

De maneira geral, os resultados obtidos mostraram que a utilização do Sistema de Apoio ao Estudo de Proteínas pode ser útil para a investigação de novas proteínas, cuja classe ainda é desconhecida, o sistema possibilita classificá-las de acordo com um classificador pré-existente, alinhá-las a outras seqüências e ainda visualizar estruturas terciárias de seqüências similares. Desse modo, o sistema é capaz de predizer uma classe para a proteína de acordo com as classes utilizadas para a geração do classificador, fornecer proteínas com regiões similares, o que também ajudará na classificação e fornecer uma idéia da estrutura terciária que essa proteína pode possuir.

## 7. REFERÊNCIAS

ALTSCHUL S., GISH W., MILLER W., MYERS E., D. LIPMAN.(1990).”Basic Local Alignment-Search Tool”. *Journal Molecular Biology*, v.215, p.403-410;

AMUI, S.F.(2006). “*Do laboratório ao virtual: Desenvolvimento de um banco de dados de venenos de serpentes brasileiras e análise computacional de estruturas de fosfolipases A2*”. Dissertação (Mestrado em Ciências Farmacêuticas) - Universidade de São Paulo, Fundação de Amparo à Pesquisa do Estado de São Paulo.

BALDI, P. & BRUNNAK, S. (1998). “*Bioinformatics: The Machine Learning Approach*”. Cambridge: MIT Press.

KORF, I.; YANDELL, M; BEDELL, J.. (2003). J.BLAST – “*An essential Guide to the Basic Local Alignment Search Tool*”, 1ed. Californica: O’Reilly & Associates.

CASS M.E., RZEPA H.S., RZEPA D.R., & WILLIAMS C.K.. (2005).“The use of the free, open-source program Jmol to generate an interactive web site to teach molecular symmetry”. *Journal of Chemical Education* 82(11):1736-1740.

COSTA. J.A.F.; BITTENCOURT V. G.; SOUTO C.P. (2005). “Aplicação de Multiclassificadores Heterogêneos no Reconhecimento de Classes Estruturais de Proteína”. *In: Congresso Brasileiro de Redes Neurais, Natal*. Proceedings of CBRN.

HAYKIN, S. (1994).“*Neural Networks: A Comprehensive Foundation*”. New York: Macmillan.

HERRÁEZ A. (2006).“*Biomolecules in the computer: Jmol to the rescue*”. *Biochemistry and Molecular Biology Education* 34(4): 255-261.

WITTEN, I. H. & FRANK.(2000). “*Data mining: practical machine learning tools and techniques with Java implementation*”. USA: Morgan Kaufman Publishers.

LEHNINGER, A. L., NELSON, D. L., COX, M. M. (1998). “ *Principles of Biochemistry with an Extended Discussion of Oxygen – Binding Proteins*”. 2ª ed. New York: Worth Publishers Inc.

LORENA, A.C. & CARVALHO, A. C. P. L. F. de. (2003) “*Utilização de Técnicas Inteligentes em Bioinformática*”. Relatórios Técnicos N° 219 São Carlos.

OLIVEIRA, L. L.; SANTOS, G. F. ; GIULIATTI, S. ; TINÓS, R. (2007). “ Investigation of the classification of snake venom-neutralizing effects of medicinal plants via Artificial Intelligence techniques“. In: *X-Meeting 3rd International Conference of the AB3C*, 2007, São Paulo. Proceedings.

PROSDOCIMI, F. *et tal.*(2003). “*Bioinformática: Manual do Usuário*”. Revista Biotecnologia Ciência &Desenvolvimento - n. 29 - janeiro, 2003.

SANTOS, G. F.; TINÓS, R. & GIULIATTI, S. (2006) “Classification of snake venom-neutralizing effects of medicinal plants via artificial neural networks”. In: *the Proceedings of 14<sup>th</sup> Annual International Conference on Intelligence System for Molecular Biology ISMB*, Fortaleza, Brazil.

SANTOS, E.C. (2004). “*Uma introdução a Bioinformática através da análise de algumas ferramentas de software livre ou de código aberto utilizadas para o estudo de alinhamento de seqüências*”. Monografia apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras e a FAEPE como requisito para obtenção do título de Especialista em Administração em Redes Linux.

SOUTO, M. C. P.; LORENA, A.C.; DELBEM, A.C.B. & CARVALHO, A.C.P.L.F. de. (2003). “Técnicas de Aprendizado de Máquina para Problemas em Biologia Molecular”. In: *II Jornadas de Atualização em Inteligência Artificial*, SBC.

TSUNODA D.F. (2004). “*Abordagens Evolucionárias para descoberta de padrões e classificação de proteínas*”. Tese apresentada ao Centro Federal de Educação Tecnológica do Paraná (CEFET-PR) para obtenção do título de Doutor em Ciências. Curitiba, (2004).

WANG, J. T. L.; MA, Q.; SHASHA, D. & WU, C. H. (2001). “New techniques from extracting features from protein sequences”. *IBM Systems Journal*, 40 (2).

## **Apêndice A**

**UNIVERSIDADE DE SÃO PAULO**  
Faculdade de Medicina de Ribeirão Preto  
Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto

**LARIZA LAURA DE OLIVEIRA**

### **DOCUMENTO DE ESPECIFICAÇÃO DE REQUISITOS**

**SISTEMA DE APOIO AO ESTUDO DE PROTEÍNAS ATRAVÉS DE TÉCNICAS DE  
INTELIGÊNCIA ARTIFICIAL**

**RIBEIRÃO PRETO**

2008

## LISTA DE FIGURAS

<b>FIGURA 1: DIAGRAMA DE CASO DE USO.</b>	<b>71</b>
<b>FIGURA 2: DIAGRAMA DE SEQÜÊNCIA: CLASSIFICAR SEQÜÊNCIAS.</b>	<b>72</b>
<b>FIGURA 3: DIAGRAMA DE SEQÜÊNCIA: VISUALIZAR ESTRUTURA PROTÉICA.</b>	<b>72</b>
<b>FIGURA 4: DIAGRAMA DE SEQÜÊNCIA: REALIZAR ALINHAMENTO.</b>	<b>73</b>
<b>FIGURA 5: DIAGRAMA DE CLASSES DO SISTEMA.</b>	<b>74</b>

# SUMÁRIO

<b>1. DESCRIÇÃO GERAL DO SISTEMA .....</b>	<b>64</b>
<b>2. REQUISITOS DO SISTEMA .....</b>	<b>65</b>
2.1. REQUISITOS FUNCIONAIS .....	65
2.2. REQUISITOS NÃO-FUNCIONAIS.....	66
<b>3. MODELOS DE CASOS DE USO .....</b>	<b>67</b>
3.1. CASO DE USO: CLASSIFICAR SEQÜÊNCIAS PROTÉICAS.....	67
3.2. CASO DE USO: VISUALIZAR ESTRUTURA PROTÉICA. ....	68
3.3. CASO DE USO: REALIZAR ALINHAMENTO .....	69
3.4. DIAGRAMA DE CASO DE USO .....	70
<b>4. DIAGRAMAS DE SEQÜÊNCIA.....</b>	<b>71</b>
<b>5. DIAGRAMA DE CLASSES.....</b>	<b>73</b>

# **1. Descrição Geral do Sistema**

Nesta seção será apresentada uma descrição detalhada do sistema. Para auxiliar na compreensão foram utilizados alguns diagramas UML. Este documento de especificação de requisitos foi baseado em (PRESSMAN, 2005).

## **1.1. Perspectiva do Sistema**

O sistema a ser implementado deverá conter três módulos, sendo um para classificação de proteínas, um para alinhamento e um para visualização. Desse modo, deverá ser possível gerar classificadores, armazená-los, permanentemente, e excluí-los. Também, deve ser possível realizar o alinhamento de determinada seqüência protéica, bem como visualizar sua estrutura terciária, caso ela exista. Os módulos do sistema são independentes.

### **1.1.1. Interfaces de Usuário**

A Interface de Usuário será baseada na apresentação de dados em janelas, sendo desejável uma interface amigável e que privilegie a usabilidade do sistema.

### **1.1.2. Interfaces de Software**

Conforme previsto no projeto inicial, o sistema fará uso de softwares disponíveis, desse modo algumas interfaces de software foram identificadas:

- O sistema proverá visualização de estruturas protéicas através do software JMol.
- Para prover a funcionalidade de alinhamento será utilizado o software BLAST.
- Além disso, o sistema proverá uma interface de software com um banco de dados relacional necessário para o armazenamento de dados dos classificadores gerados.

### **1.1.3. Interfaces de Hardware**

O sistema necessita de um computador *Desktop* para funcionar, com mouse, teclado e monitor para entrada e saída de dados. Não foram identificadas outras interfaces de hardware.

## 2. Requisitos do Sistema

Para facilitar o entendimento os requisitos serão separados de acordo com os módulos do sistema. Os atributos são divididos em evidentes (E) ou ocultos (O) dependendo da maneira como são percebidos pelo usuário. Os evidentes são identificados facilmente pelo usuário, enquanto que os ocultos são realizados pelo sistema e não são visíveis.

### 2.1.Requisitos Funcionais

Os requisitos funcionais do sistema apresentados abaixo foram divididos entre os três módulos do sistema para facilitar sua compreensão.

- **Modulo de Classificação:**

- 1.1. Gerar um classificador, utilizando um algoritmo de classificação, a partir de um conjunto de seqüências FASTA, que contenham uma classe associada. (O)
- 1.2. Permitir a escolha da classe para uma dada seqüência antes do início do treinamento. (E)
- 1.3. Armazenar persistentemente um classificador gerado.(O)
- 1.4. Classificar um conjunto de seqüências protéicas no formato FASTA a partir de um classificador já existente. (O)
- 1.5. Possibilitar a visualização dos resultados da classificação tais como: erro de classificação, matriz de confusão, entre outros. (E)

- **Modulo de Visualização:**

- 2.1. Permitir a visualização da estrutura de uma proteína, caso esta tenha estrutura identificada. (E)
- 2.2. Buscar arquivo no formato PDB de uma proteína a partir da identificação da proteína contida na seqüência fasta da mesma. A viabilidade desse requisito está sendo avaliada. (O)

- **Modulo de Alinhamento**

- 3.4. Realizar o alinhamento de uma seqüência protéica no formato fasta através do software BLAST. (E)
- 3.5. Permitir a visualização do alinhamento, bem como mostrar parâmetros importantes.(E)

Legenda: (E) Evidente

(O) Oculto

## **2.2.Requisitos Não-Funcionais**

### **2.2.1. Usabilidade**

O sistema possuirá interfaces de usuário baseadas em janelas. Deseja-se que essas interfaces sejam intuitivas e amigáveis. Desse modo, o usuário conseguirá utilizar o sistema sem maiores problemas.

### **2.2.2. Manutenibilidade**

O sistema está sendo desenvolvido segundo padrões de programação orientada a objeto e engenharia de software e o código desenvolvido será documentado. Assim, modificações futuras poderão ser realizadas facilmente.

### **2.2.3. Portabilidade**

O sistema será desenvolvido na linguagem de Programação Java o que garante sua portabilidade.

### **2.2.4. Consistência**

O sistema será monousuário e possíveis inconsistências são improváveis.

### 3. Modelos de Casos de Uso

Nesta seção apresentam-se três Casos de Uso, ou seja, eventos nos quais os Atores (Usuários) interagem com o sistema, estimulando diferentes respostas de acordo com a entrada dos dados.

#### 3.1.Caso de Uso: Classificar Sequências Protéicas.

##### *Caso de Uso Alto Nível*

Caso de Uso:	Classificar Sequências Protéicas
Atores:	Usuário
Tipo:	Primário
Descrição:	O usuário entra com as seqüências no formato FASTA e o sistema as classifica.

##### *Caso de uso expandido*

Caso de Uso:	Classificar Sequências Protéicas
Atores:	Usuário
Finalidade:	Separar um conjunto de seqüências de acordo com suas possíveis classes funcionais.
Visão geral:	O usuário seleciona um arquivo contendo uma ou mais seqüências protéicas no formato FASTA. Seleciona um classificador e em seguida a opção classificar e espera pelos resultados. O usuário visualiza os resultados da Classificação realizada.
Tipo:	Primário e essencial
Referências cruzadas:	Requisitos: 1.1, 1.2, 1.3, 1.4 e 1.5

##### *Caso de uso expandido: Seqüência típica de eventos*

Ação do Ator	Resposta do sistema
1.O Usuário entra no sistema.	
2. O Usuário seleciona um conjunto de seqüências no formato FASTA.	
3. O Usuário seleciona um classificador.	
4. O Usuário solicita a classificação das	4. O sistema oferece um classificador pronto

seqüências (no módulo de classificação).	para que o usuário possa utilizar.
5. O Usuário seleciona a opção iniciar classificação a partir de um classificador pronto.	6. O sistema efetua a Classificação das seqüências a partir do classificador já existente.
	7. Quando a classificação termina o sistema apresenta os resultados obtidos.

### ***Caso de uso expandido: Seqüências alternativas***

Evento 2. O Usuário entra com seqüências em formato desconhecido. O sistema deve informar o erro.

Evento 3. Caso não exista um classificador o usuário deverá fornecer um arquivo de seqüências no formato FASTA contendo a classe de cada proteína para classificação, juntamente com os parâmetros do algoritmo de classificação. Uma vez gerado um classificador, o usuário poderá salva-lo e selecionar um arquivo para testes.

### **3.2.Caso de Uso: Visualizar Estrutura Protéica.**

#### ***Caso de Uso Alto Nível***

Caso de Uso:	Visualizar Estrutura Protéica
Atores:	Usuário
Tipo:	Primário
Descrição:	O Usuário solicita a visualização de uma estrutura protéica. O sistema deverá mostrá-la caso exista.

#### ***Caso de uso expandido***

Caso de Uso:	Visualizar Estrutura Protéica
Atores:	Usuário
Finalidade:	Visualizar a estrutura de uma protéica caso exista.
Visão geral:	O Usuário solicita a visualização da estrutura de uma dada proteína informando seu identificador. O sistema realiza busca do arquivo no formato PDB dessa proteína no Banco de Dados Público do PDB. Se o arquivo for encontrado o sistema proverá a visualização da estrutura.
Tipo:	Primário e essencial

---

Referências      Requisitos: 2.1 e 2.2  
cruzadas:

*Caso de uso expandido: Seqüência típica de eventos*

<b>Ação do Ator</b>	<b>Resposta do sistema</b>
1. O Usuário entra no sistema.	
2.O Usuário solicita a visualização da estrutura de uma dada seqüência.	3. O sistema verifica a existência da estrutura da proteína fornecida. Caso exista o sistema mostrará sua visualização. Caso contrário, informará o usuário.

*Caso de uso expandido: Seqüências alternativas*

Evento 2. O Usuário poderá fornecer ele próprio um arquivo no formato PDB para a visualização da estrutura da proteína.

### **3.3.Caso de Uso: Realizar Alinhamento**

*Caso de Uso de Alto Nível*

Caso de Uso:	Realizar Alinhamento
Atores:	Usuário
Tipo:	Primário
Descrição:	O Usuário entra com seqüências para realizar o alinhamento. O sistema permite a visualização do alinhamento e de seus parâmetros.

*Caso de uso expandido*

Caso de Uso:	Realizar Alinhamento
Atores:	Usuário
Finalidade:	Verificar o alinhamento de uma seqüência com outras.
Visão geral:	O Usuário entra com as seqüências as quais deseja alinhar. O usuário seleciona a opção alinhamento. O sistema realizará o alinhamento utilizando o BLAST e proverá a visualização dos resultados.
Tipo:	Primário e essencial

---

Referências      Requisitos: 3.1 e 3.2  
cruzadas:

*Caso de uso expandido: Seqüência típica de eventos*

<b>Ação do Ator</b>	<b>Resposta do sistema</b>
1. Usuário entra no sistema.	
2. Usuário seleciona arquivo contendo seqüências no formato FASTA.	
3. Usuário entra no Módulo de Alinhamento e solicita a opção alinhamento.	4. O sistema deverá realizar o Alinhamento e retornar os resultados.

*Caso de uso expandido: Seqüências alternativas*

Evento 2: O Usuário não precisa entrar com um novo arquivo fasta caso já tenha feito isso durante a utilização de outro módulo.

Evento 4: Caso haja algum problema com a execução do Blast, que será realizada remotamente, o sistema deverá informar ao usuário.

### **3.4.Diagrama de Caso de Uso**

Para ilustrar outros Casos de Usos não abordados neste Documento foi elaborado um Diagrama de Caso de Uso envolvendo todos os eventos e os respectivos Atores, dando uma visão geral do sistema e os papéis de cada Usuário.

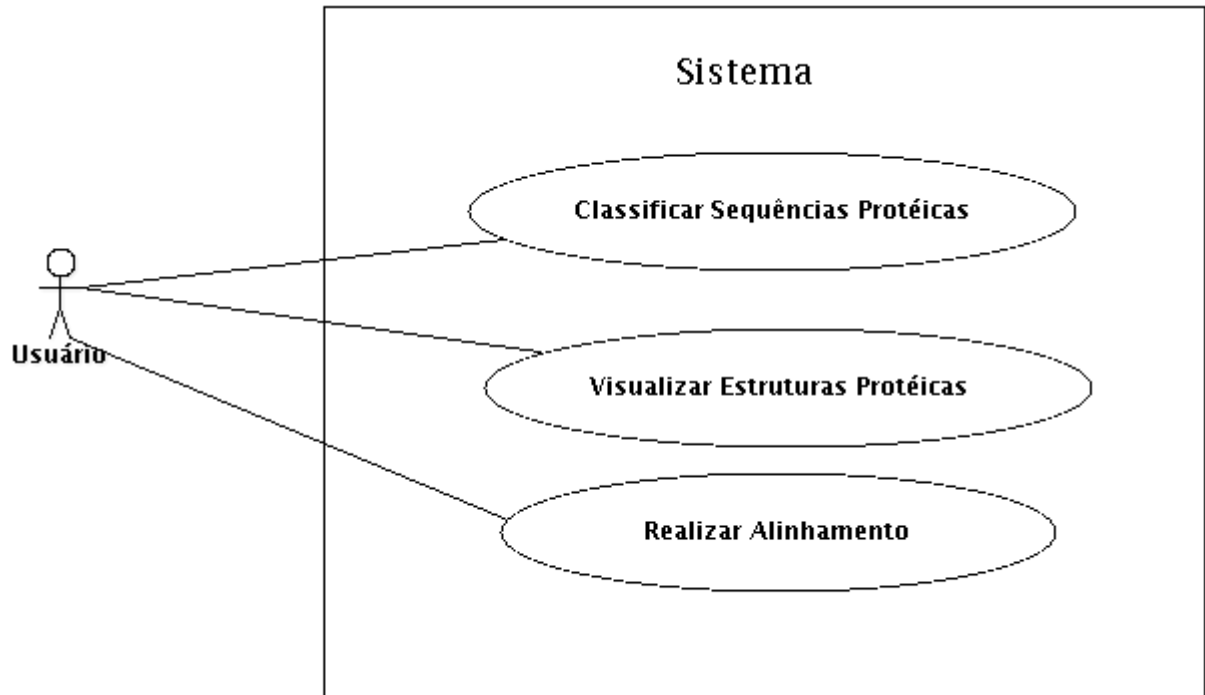
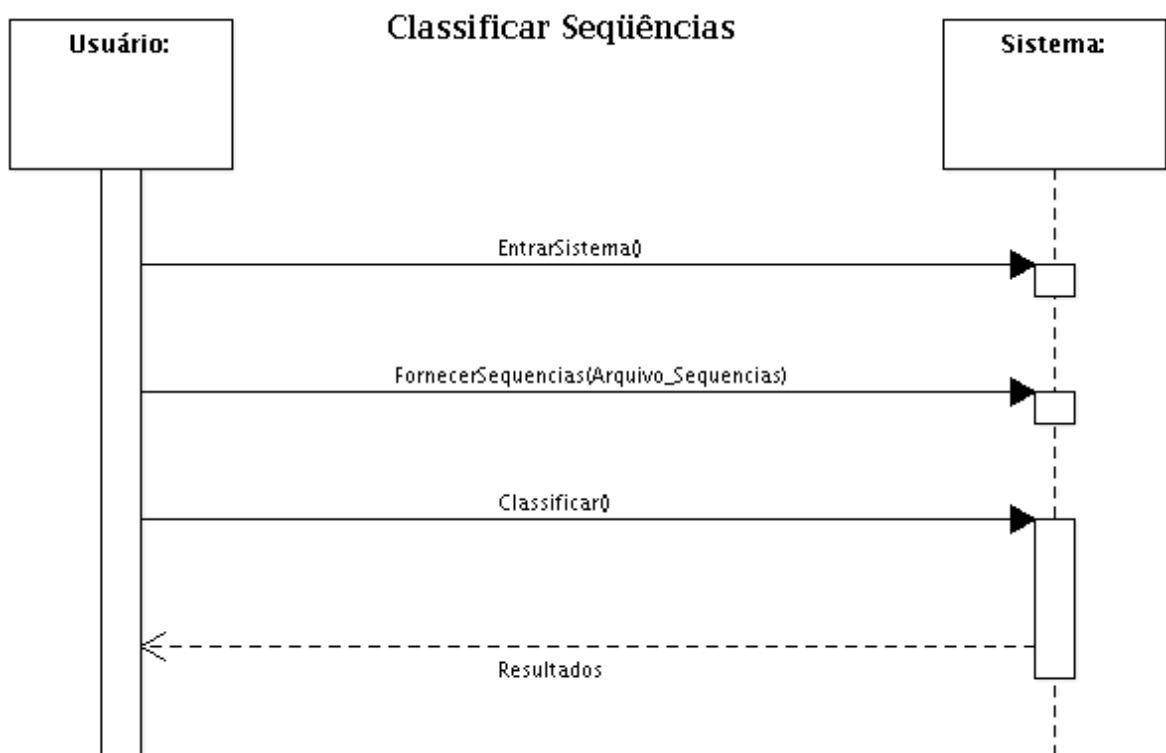


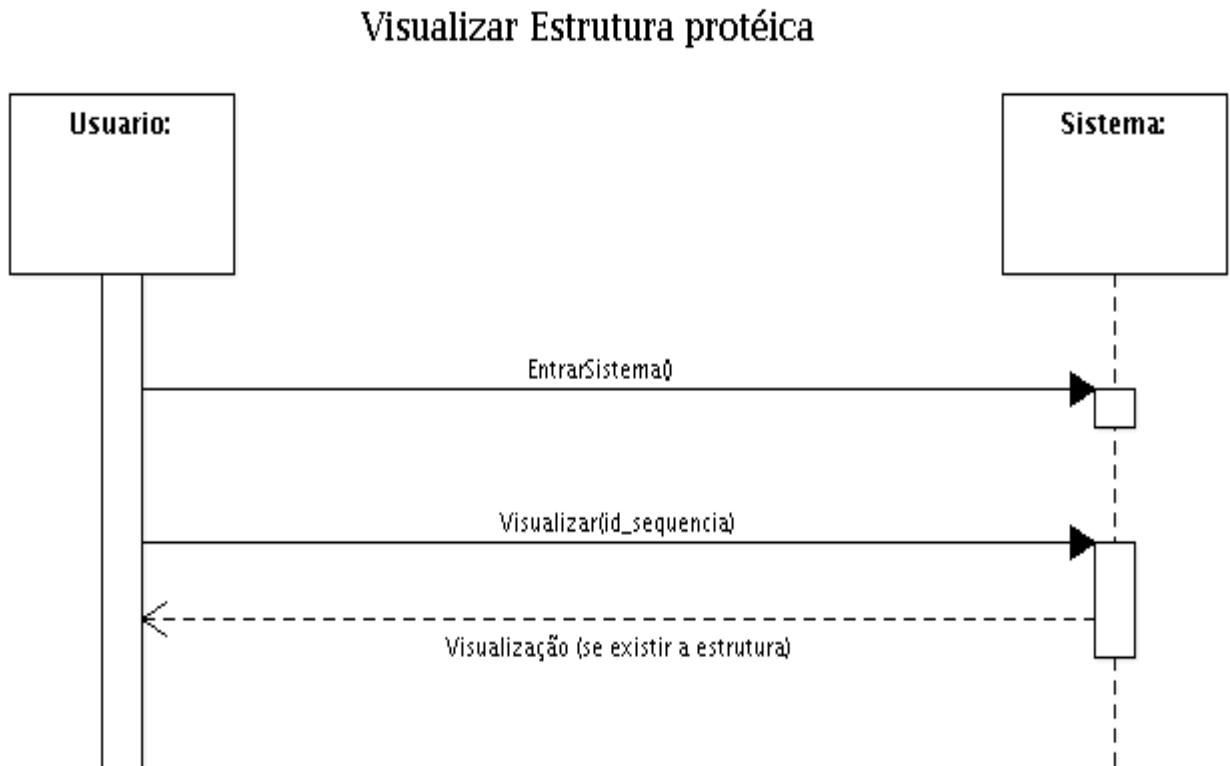
Figura 27: Diagrama de Caso de Uso.

#### 4. Diagramas de Seqüência

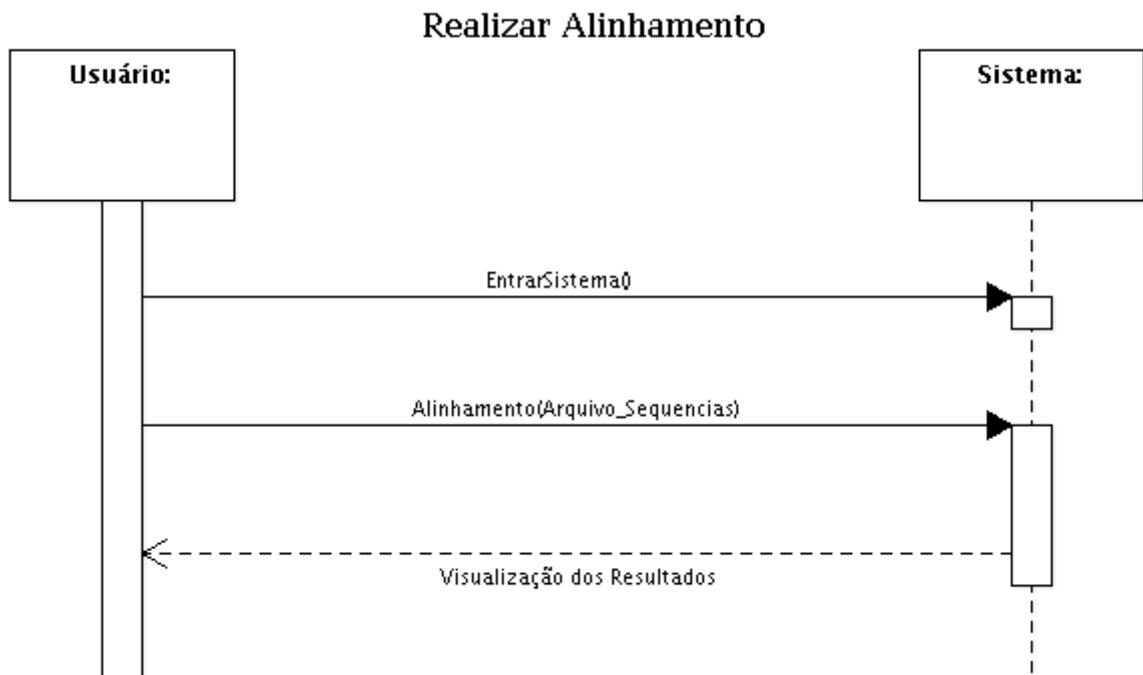
Para auxiliar a compreensão da ordem dos eventos e as respectivas respostas do sistema foram criados Diagramas de Seqüência, exibindo os principais acontecimentos de cada caso de uso e como o sistema reage, através do sentido das setas que os representam.



**Figura 28: Diagrama de Seqüência: Classificar Seqüências.**



**Figura 29: Diagrama de Seqüência: Visualizar Estrutura Protéica.**



**Figura 30: Diagrama de Sequência: Realizar Alinhamento.**

## 5. Diagrama de Classes

Para entendimento geral do funcionamento do sistema e do relacionamento das classes que o compõe será apresentado seu Diagrama de Classes. Para simplificar os métodos de encapsulamento de dados (*get* e *set*) e as classes de processamento de arquivos foram omitidas. As classes principais são:

- Sistema: Classe principal que compreende os módulos de classificação, alinhamento e visualização.
- Classificador: Interface do módulo de classificação, que deverá ser utilizada pelos classificadores existentes. Neste caso, utilizaremos apenas uma Rede Neural, porém caso deseje-se posteriormente adicionar um novo classificador essa operação se tornará mais fácil.
- MLP: Implementação da Rede Neural Artificial.
- Alinhamento: Classe responsável pela interface com o software BLAST.
- VisualizacaoEstrutura: Classe responsável pela interface com o Software Jmol.

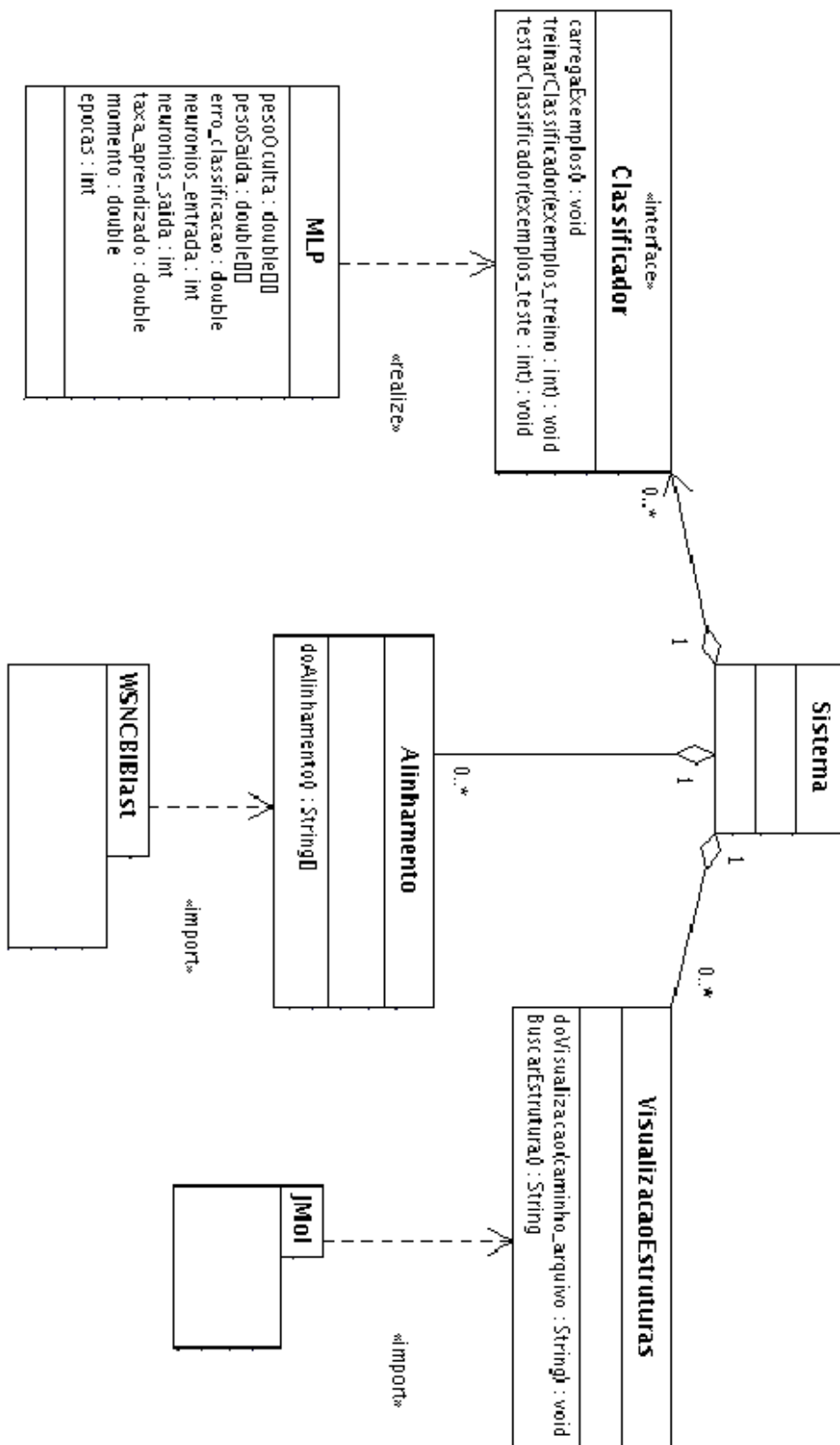


Figura 31: Diagrama de Classes do Sistema.

## **6. Referências**

PRESSMAN, Roger S. "Engenharia de Software". Editora Makron. 1995, São Paulo, Brasil.