Capítulo 6

Processo de KDD aplicado à Bioinformática

Ana T. Winck, Karina S. Machado, Duncan D. Ruiz e Osmar Norberto de Souza

Resumo

A computação tem contado com pesquisas interdisciplinares, como a bioinformática, que tem a característica de possuir grandes volumes de dados. O gerenciamento e a análise desse tipo de dados aparecem como importantes campos de investigação. A construção de um ambiente de descoberta de conhecimento em base de dados (KDD – Knowledge Discovery in Databases) pode ser vista como uma abordagem que pode apresentar importantes desafios se direcionado para o tratamento de dados tão peculiares e de característica não convencional, como na bioinformática. Propõe-se apresentar e discutir todas as etapas que compõe o processo de KDD, e como tal processo pode ser aplicado à bioinformática. Acredita-se que as lições aprendidas possam ser úteis no desenvolvimento de futuras pesquisas interdisciplinares.

Abstract

The computer science has relied on interdisciplinary researches, as bioinformatics, which usually posses a large amount of data. The management and analysis of such kind of data appear as an important research field. To build a KDD (Knwoledge Discovery in Databases) environment can be viewed as an approach that present important challenges if directed to the treatment of these particular and non-conventional data, as in bioinformatics. We propose to present and discuss all KDD steps, and how such process can be applied on bioinformatics. We believe that the lessons learned can be useful to the development of future interdisciplinary researches.

6.1. Introdução

A computação, nas suas diferentes áreas de aplicação, vem sendo enriquecida com pesquisas interdisciplinares, isto é, pesquisas que fazem uso da computação para atender a outras áreas de conhecimento. Dentre as diferentes áreas que se beneficiam com essa interação, pode-se citar a bioinformática. Para Wang et al. (2005), bioinformática é a

ciência de gerenciar, minerar, integrar e interpretar informações a partir de dados biológicos, enfatizando o constante crescimento do número de dados que demandam por essas análises. A bioinformática consolidou-se quando da implantação do projeto GENOMA: os seqüenciadores automáticos de DNA produziram uma grande quantidade de sequências, tornando-se necessário o investimento em recursos específicos de armazenamento e análise desses dados biológicos (Luscombe et al. 2001). A partir daí surgiram várias áreas de atuação em bioinformática. Lesk (2002) aponta algumas principais frentes, como genomas, proteomas, alinhamento de árvores filogenéticas, biologia de sistemas, estruturas de proteínas e descoberta de fármacos.

Do ponto de vista da computação, diversas áreas de atuação podem ser empregadas para atender a essa demanda. Dentre elas, destaca-se o processo de descoberta de conhecimento em banco de dados (KDD – *Knowledge Discovery in Databases*), proposto por Fayyad et al. (1996). Este processo, difundido e conhecido pela comunidade de banco de dados, pode apresentar importantes desafios se direcionado para o tratamento de dados tão peculiares e de característica não-convencional, como na bioinformática.

Uma área de pesquisa que tem sido investigada em bioinformática é o desenho racional de fármacos (RDD – *Rational Drug Design*) (Kuntz, 1992). O princípio fundamental do RDD é a interação entre receptores e ligantes (Lybrand, 1995). Ligantes são definidos como moléculas que se ligam a outras moléculas biológicas, chamadas receptores, para realizar ou inibir funções específicas (Balakin, 2009). É na docagem molecular que se investiga e avalia o melhor encaixe do ligante no receptor (Kuntz, 1992). Um dos maiores desafios dessa área de pesquisa é lidar com o grande volume de dados envolvidos: catalogação de ligantes, conformações do receptor obtidas por simulações pela dinâmica molecular (DM) e resultados de experimentos de docagem molecular. Este minicurso propõe-se a apresentar e discutir todas as etapas que compõem o processo de KDD e como esse processo pode ser aplicado à bioinformática no contexto de RDD.

O restante deste trabalho está organizado conforme segue. A seção 6.2 apresenta uma introdução à bionformática, apresentando os principais termos envolvidos. A seção 6.3 relata os passos de construção de um típico processo de KDD. Na seção 6.4 são apresentados os exemplos práticos de construção de um processo de KDD para a bioinformática.

6.2. Introdução à bioinformática

Inicialmente, bioinformática era definida como uma área interdisciplinar envolvendo biologia, ciência da computação, matemática e estatística para analisar dados biológicos (Mount, 2004). Com o advento da era genômica, bioinformática passou a ser definida como biologia em termos de moléculas e a aplicação da computação para entender e organizar a informação associada com esses dados biológicos em larga escala (Luscombe et al., 2001).

Para Lesk (2002) uma das principais características dos dados de bioinformática é o seu grande volume. Como exemplo, tem-se o banco de dados de sequências de nucleotídeos GenBank (Benson et al, 2008), que até agosto de 2009 já continha depositadas 106.533.156.756 bases em 108,431,692 seqüências. O *GenBank* e outros bancos de dados biológicos não são apensas extensos, mas crescem a taxas bastante

elevadas (Lesk, 2002). Esse grande volume de dados biológicos e seu crescimento elevado definem os principais objetivos da bioinformática: organizar os dados biológicos de maneira que os pesquisadores possam acessar informações já cadastradas, assim como submeter novas entradas, na medida em que vão sendo produzidas; desenvolver ferramentas e pesquisas que ajudem na análise dos dados e utilizar as ferramentas desenvolvidas para interpretar os dados de forma que tenham significado biológico (Luscombe et al., 2001).

Hunter (1993) diz que um dos maiores desafios dos cientistas da computação que pretendem trabalhar na área de biologia molecular consiste em familiarizar-se com o complexo conhecimento biológico existente e seu vocabulário técnico extenso. Sendo assim, a seguir serão revisados conceitos necessários para um entendimento básico sobre os dados envolvidos neste trabalho.

6.2.1. O dogma central da biologia molecular

O DNA é responsável por codificar todas as proteínas. Esse processo, definido como o dogma central da biologia molecular, compreende a fase de transcrição, onde o DNA é processado, gerando o mRNA, ou RNA mensageiro, que é enviado para fora do núcleo da célula, onde a mensagem é traduzida em proteínas.

6.2.2. Proteínas

Proteínas são as macromoléculas mais abundantes nas células vivas e se encontram em todas as células e em todas as partes das células. Elas realizam a maioria das funções catalisando as reações químicas nas células vivas. A estrutura das diferentes proteínas é construída com o conjunto dos 20 aminoácidos básicos.

Todos os aminoácidos têm uma estrutura conforme Figura 6.1a, contendo um grupo carboxila (COOH), um grupo amino (NH2), um átomo de hidrogênio, ligados ao mesmo átomo de carbono (C). A diferenciação entre os 20 aminoácidos ocorre devido à presença de um radical R, que confere aos aminoácidos uma série de características como, por exemplo, polaridade e grau de ionização. A Figura 6.1b mostra um exemplo de aminoácido, a alanina, cujo radical é CH3.

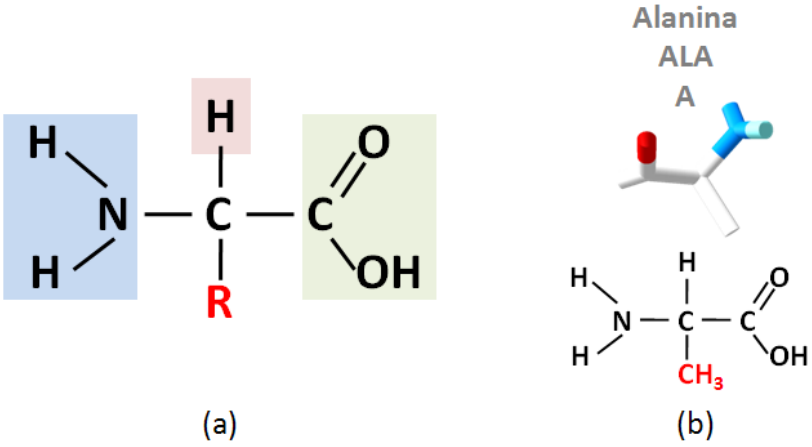


Figura 6.1. (a) Estrutura de um aminoácido. (b) Estrutura do aminoácido Alanina.

De acordo com Hunter (1993), as cadeias de aminoácidos são formadas por uma reação que ocorre entre o átomo de nitrogênio no final do grupo amina de um aminoácido e o átomo de carbono do grupo carboxila de outro, ligando os dois aminoácidos e liberando uma molécula de água. A ligação é chamada de ligação peptídica, e longas cadeias de aminoácidos formam os chamados polipeptídeos. Todas as proteínas são polipeptídeos, apesar deste termo geralmente referir-se a cadeias menores que proteínas inteiras.

A seqüência contínua de aminoácidos de uma proteína é chamada de estrutura primária (Figura 6.2a), e é o nível mais básico de organização de uma proteína. Os aminoácidos da estrutura primária podem adquirir conformações secundárias como α-hélice ou folhas-β (Figura 6.2b). As ligações de hidrogênio entre os aminoácidos que formam a proteína favorecem essas estruturas secundárias. A composição desses elementos estruturais formando um ou mais domínios estabelecem a estrutura terciária de uma proteína (Figura 6.2c). A estrutura final, estrutura quaternária, pode conter várias cadeias polipeptídicas (Figura 6.2d). As formações destas estruturas terciárias e quaternárias permitem o surgimento de regiões funcionais chamadas sítio ativo (que confere atividade biológica à proteína).

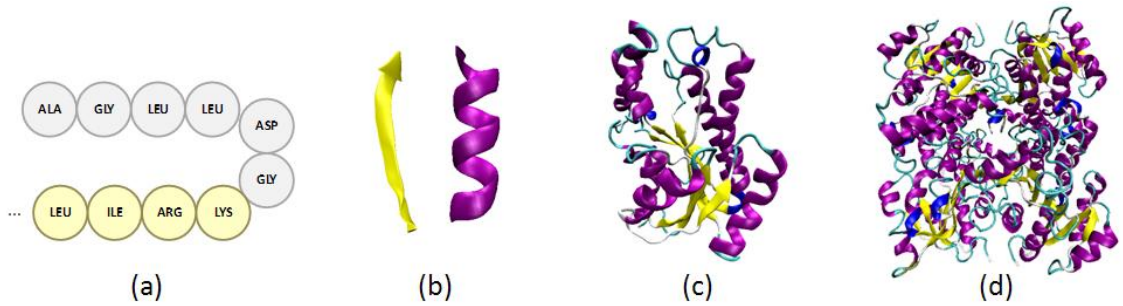


Figura 6.2. (a) Estrutura primária de uma proteína. (b) Estruturas secundárias de uma proteína: folha-β e α-hélice. (c) Exemplo de estrutura terciária de uma proteína (PDBCode:1ENY). (d) Exemplo de estrutura quaternária de uma proteína (PDBCode:1BVR).

6.2.3. Principais áreas de atuação em bioinformática

Os dados de DNA, RNA e proteínas podem ser utilizados em diferentes áreas de bioinformática, definindo algumas das principais atividades na área: análise de seqüências de proteínas, alinhamento múltiplo de seqüências, visualização de estrutura de proteínas, predição de estrutura de proteínas, ferramentas para genômica e proteômica, bancos de dados biológicos, desenho racional de fármacos, entre outros. Entre as atividades descritas, este trabalho está inserido no contexto de desenho racional de fármacos, ou *Rational Drug Design* (RDD) (Kuntz, 1992). Esse contexto e os processos envolvidos no mesmo serão descritos a seguir.

6.2.4. Desenho racional de fármacos

Segundo Drews (2003), com o avanço da biologia molecular e das técnicas de simulação por computador, o planejamento de medicamentos passou a ser feito de

maneira mais lógica, o que é chamado de desenho racional de drogas. O processo de RDD é composto de quatro etapas (Kuntz, 1992):

1. A primeira etapa consiste em isolar um alvo específico, chamado receptor, (proteínas, receptores de membrana, DNA, RNA, etc.). A partir da análise computacional da estrutura tridimensional (3D) dessa proteína armazenada em um banco de dados estrutural como o *Protein Data Bank* (PDB) (Berman et al., 2000), é possível apontar prováveis regiões de ligação, por exemplo, regiões onde uma pequena molécula, chamada ligante, pode se ligar a esse receptor;
2. Baseado nas prováveis regiões de ligação identificadas na etapa anterior é selecionado um conjunto de prováveis candidatos a ligantes que podem se ligar a essa região no receptor. As diferentes conformações que um dado ligante pode assumir dentro do sítio de ligação de uma determinada proteína podem ser simuladas por software de docagem molecular como AutoDock3.0.5 (Morris et al. 1998);
3. Os ligantes que teoricamente obtiveram melhores resultados nas simulações são experimentalmente sintetizados e testados;
4. Baseado nos resultados experimentais, o medicamento é gerado ou o processo retorna à etapa 1 com pequenas modificações no ligante.

Sendo assim, a interação entre moléculas é o princípio do desenho de drogas, sendo um dos mais interessantes desafios nessa área (Broughton, 2000). É na docagem molecular que é procurado o melhor encaixe do ligante na estrutura alvo ou receptor.

6.2.5. Docagem molecular e simulações pela dinâmica molecular

De acordo com Lybrand (1995), um medicamento deve interagir com um receptor para exercer uma função fisiológica vinculada à ligação dessa com outras moléculas, e essas ligações determinam se as funções serão estimuladas ou inibidas. Essas ligações ocorrem em locais específicos, chamados sítios ativos ou de ligação. A associação entre duas moléculas no sítio de ligação não depende somente do encaixe. Existe a necessidade de haver uma energia favorável para que essa interação ocorra. Essa energia é determinada pela carga e tamanho dos átomos ali contidos. Estas ligações que ocorrem entre os átomos são medidas pela quantidade de energia despendida, quanto mais negativa, melhor a interação entre as moléculas.

Esse processo de ligar uma pequena molécula a uma proteína-alvo não é um processo simples: muitos fatores entrópicos e entálpicos influenciam as interações entre eles. A mobilidade do ligante e do receptor, o efeito do ambiente no receptor (proteína), a distribuição de carga no ligante, e outras interações dos mesmos com a água, complicam muito a descrição desse processo. A maioria dos algoritmos que executam docagem molecular somente considera a flexibilidade do ligante, considerando o receptor rígido ou as possíveis orientações das cadeias laterais são selecionadas baseadas em uma visão subjetiva. Entretanto, sabe-se que as proteínas não permanecem rígidas em seu ambiente celular, sendo de fundamental importância a consideração dessa flexibilidade do receptor na execução dos experimentos de docagem molecular.

Há muitos trabalhos sendo desenvolvidos para a incorporação da flexibilidade de receptores na docagem molecular, como revisado por Totrov e Abagyan (2008) e Chandrika et al. (2009). Entre as várias abordagens disponíveis, neste trabalho estamos considerando a execução de uma série de experimentos de docagem molecular, considerando em cada experimento uma conformação do receptor gerada por uma simulação pela dinâmica molecular (DM) (Lin et al. 2002; Machado et al., 2007, Amaro et al., 2008). A simulação pela DM é uma das técnicas computacionais mais versáteis e amplamente utilizadas para o estudo de macromoléculas biológicas (van Gunsteren 1990). Com simulações pela DM é possível estudar o efeito explícito de ligantes na estrutura e estabilidade das proteínas, os diferentes parâmetros termodinâmicos envolvidos, incluindo energias de interação e entropias.

O principal problema com a utilização da trajetória da DM em docagem molecular é o tempo necessário para a execução de experimentos e a grande quantidade de dados gerados. Visando reduzir esse tempo de execução e melhor entender como ocorre a interação receptor-ligante considerando a flexibilidade do receptor, nesse trabalho está sendo aplicado um processo de KDD. Para alcançar esse objetivo é necessário um conjunto de etapas que serão descritas nas próximas seções.

6.3. Processo de KDD

O processo KDD (Fayyad et. al, 1996; Han e Kamber, 2006) é uma sequência de etapas interativas e iterativas direcionadas à tomada de decisão, especialmente quando da existência de um grande volume dados. A Figura 6.3 ilustra uma adaptação da visão de Han e Kamber (2006) quanto às diferentes etapas do processo de KDD. Por essa figura nota-se que um processo de KDD inicia-se a partir da existência de um grande volume de dados (a) que merecem ser analisados. Como esse volume de dados, na maioria das vezes, é proveniente de fontes heterogêneas, os mesmos passam por uma etapa de transformação (b) para que sejam representados em um único padrão de referência. Uma vez transformados, os dados são devidamente armazenados em uma base de dados alvo (c), muitas vezes projetada na forma de um *data warehouse* (DW). Os dados devidamente organizados na base de dados alvo fornecem suporte para diferentes tipos de análises, onde a principal técnica de análise presente no processo de KDD é a mineração de dados (e). Para que os dados sejam minerados, é importante que passem por um processo de pré-processamento (d), etapa importante para que algoritmos de mineração produzam melhores resultados. Os modelos induzidos a partir da mineração apresentam padrões (f) e, após analisados, podem atingir o conhecimento (g) esperado.

Embora cada etapa que compõe o processo de KDD seja suficientemente abrangente para ser tratada de forma independente, existe uma forte relação de dependência entre elas. Para melhor falar a respeito delas, iremos dividi-las em duas principais etapas:

- A construção de uma base de dados alvo – abrangendo as etapas (a), (b) e (c) da Figura 6.3; e
- Etapas de mineração de dados – a qual é composta pelas etapas (d), (e), (f) e (g) da Figura 6.3.

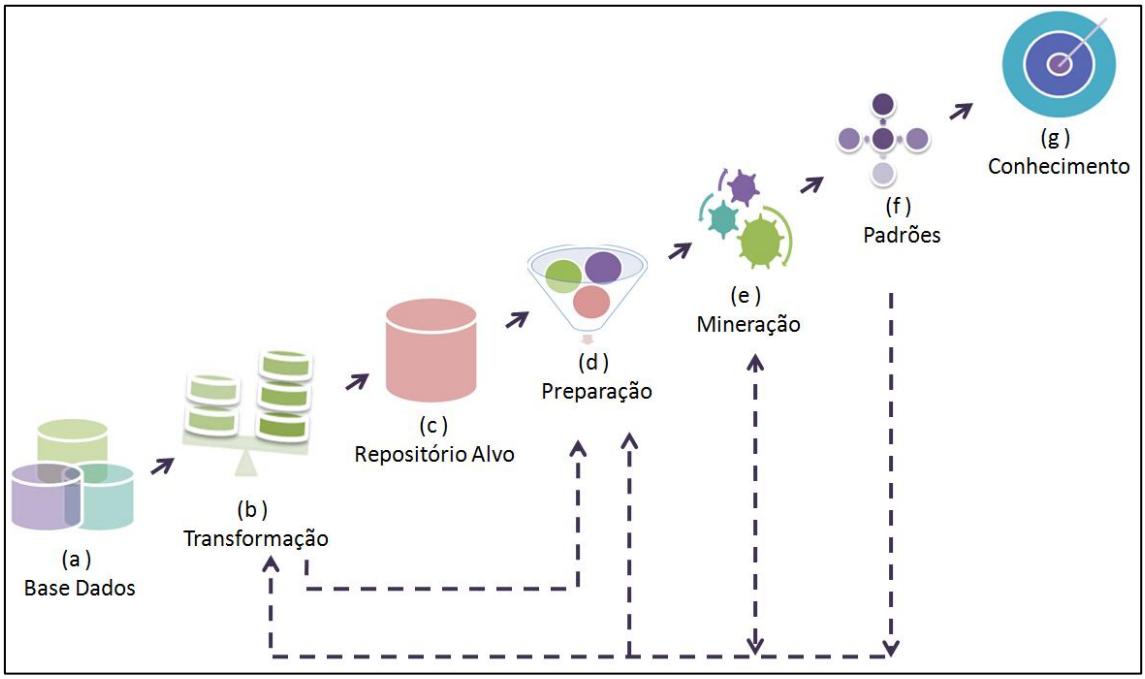


Figura 6.3. Processo de KDD adaptado de Han e Kamber (2006).

6.3.1. Construção de uma base de dados alvo

Um processo de KDD é comumente construído quando há interesse analítico sobre distintas fontes de dados, sendo que essas diferentes fontes muitas vezes têm características heterogêneas. A idéia é que esses dados sejam armazenados em um repositório alvo, semanticamente consistente. Nesse sentido, a heterogeneidade dos dados de origem é tratada de forma a garantir a integração dos dados em um único formato.

Han e Kamber (2006) sugerem que esse repositório alvo seja construído na forma de um DW, buscando manter um histórico organizado dos registros, de modo a auxiliar a tomada de decisão. Inmon (1997) apresenta a clássica definição de um DW como sendo um conjunto de dados baseado em assuntos, integrado, não-volátil, e variável em relação ao tempo, de apoio às decisões estratégicas. DW são construídos de forma a satisfazer sua estrutura multidimensional (Kimball e Ross, 2002). Um modelo analítico, que representa essa estrutura multidimensional, possui dois tipos de tabelas: fato e dimensão. Entende-se por dimensão as perspectivas de uma base de dados que possam gerar registros referentes às características do problema modelado. Essas são, tipicamente, organizadas em torno de um termo central, ou tabela fato, a qual contém os atributos chave das dimensões e atributos que representem valores relevantes ao problema.

Um dos processos mais importantes para a construção de uma base de dados alvo é a etapa de ETC – extração, transformação e carga. A extração é a primeira fase, a qual visa percorrer as distintas bases de dados e capturar apenas os dados significantes ao domínio. Na fase de transformação, os dados capturados podem sofrer algum tipo de tratamento, ajustando possíveis conflitos de tipificação, e eliminando possíveis ruídos ou conteúdos irrelevantes ao repositório alvo. Por fim, é feita a devida carga no

repositório (Kimball e Ross, 2002). No processo de KDD representado pela Figura 6.3, a fase de ETC está presente na etapa de transformação, Figura 6.3(b).

Tendo o repositório alvo modelado, desenvolvido e os dados devidamente armazenados, já é possível efetuar consultas analíticas sobre esses dados. Um exemplo, para o caso de um DW construído, é a manipulação de seus dados por ferramentas OLAP (*On-Line Analytical Processing*). Operações como pivoteamento de dados, *roll-up*, *drill-down* e *slice&dice*, são operações típicas de ambientes de processamento analítico de dados. Além dessa abordagem, entretanto, e seguindo as demais etapas do processo de KDD, os dados estão aptos a serem trabalhados pela etapa de mineração de dados.

6.3.2. Etapas de mineração de dados

A mineração de dados é a etapa do processo de KDD que converte dados brutos em informação. A mineração de dados divide-se em diferentes tarefas: descritivas e preditivas. As técnicas descritivas sumarizam relações entre dados, tendo como objetivo aumentar o entendimento a respeito deles. As técnicas preditivas têm o objetivo de apontar conclusões quanto aos dados analisados, prevendo valores para um dado atributo de interesse. Dentre as tarefas descritivas pode-se citar como exemplo regras de associação e agrupamento. Já algoritmos de classificação e regressão podem ser vistos como exemplos de tarefas preditivas (Tan et al., 2006). Este trabalho concentra-se em regras de associação, para tarefas descritivas, e árvores de decisão para classificação e regressão, para tarefas preditivas.

Regras de associação são implicações na forma $x \rightarrow y$. Em aprendizagem de máquina, regras de associação possuem duas principais medidas de qualidade: suporte e confiança. A primeira diz respeito à significância de uma regra e a segunda se refere ao percentual de ocorrências em que os itens de x ocorrem simultaneamente (Alpaydim, 2004).

Tarefas preditivas são compostas, basicamente, de uma entrada x e um resultado de saída y, onde a tarefa é aprender como mapear a entrada para a o resultado de saída. Esse mapeamento pode ser definido como uma função $y = g(x|\theta)$ onde $g(.)$ é o modelo e θ são seus respectivos parâmetros (Alpaydim, 2004). Embora existam vários algoritmos que fazem uso de tarefas preditivas, muitos deles apenas constroem uma função que indica um dado atributo alvo a que os objetos pertencem. Entretanto, em muitos problemas de mineração é necessário compreender o modelo induzido. Segundo Freitas et al. (2010), apesar da falta de consenso na literatura de mineração de dados sobre as tarefas que apresentam resultados mais compreensíveis, existe um acordo razoável de que representações como árvores de decisão são melhor compreendidas pelos usuários do que algoritmos de representação caixa-preta. Ainda segundo Freitas et al. (2010), árvores de decisão têm a vantagem de graficamente representar o conhecimento descoberto e sua estrutura hierárquica pode apontar a importância dos atributos utilizados para a predição.

Independentemente da tarefa e algoritmo de mineração sendo aplicados, é importante que os dados a serem minerados passem por uma criteriosa etapa de pré-processamento. Essa etapa, que não deve ser confundida com a etapa de ETC, é fundamental para que os modelos de mineração induzidos apresentem resultados mais satisfatórios. O pré-processamento dos dados faz uso dos dados que já estão inseridos na

base de dados alvo e adequá-los para o algoritmo de mineração o qual esses dados serão submetidos. Diversos autores (Tan et al., 2006; Han e Kamber, 2001; Fayyad et al., 1996) estão de acordo que um pré-processamento conveniente é um importante ponto a ser considerado.

6.4. Exemplos práticos

Neste trabalho propõe-se apresentar como desenvolver um processo de KDD em bioinformática, sobre bases de dados de docagem molecular. Esta seção é dividida em duas subseções. Em 6.4.1 são descritos os dados utilizados nesse trabalho, apresentando a coleta e tratamento dos mesmos, bem como a construção de uma base de dados alvo para seu armazenamento. Em 6.4.2 apresenta-se desafios e oportunidades de análise sobre esses dados, relatando o processo de pré-processamento dos mesmos até a utilização de algoritmos de mineração. Para esse trabalho é utilizado o ambiente WEKA (Hall et al., 2009) para a mineração de dados.

6.4.1. Dados biológicos utilizados neste trabalho

Para a execução de docagem molecular é necessário um receptor, um ligante e um software para executar as simulações. Nesse trabalho estamos considerando como receptor a enzyma InhA do *Mycobacterium tuberculosis* (Mtb) (Dessen et al, 1995); como ligante, a nicotinamida adenina dinucleotídeo (NADH) (Dessen et al, 1995); e como software de docagem o AutoDock3.0.5 (Morris et al., 1998).

6.4.1.1. Receptor e ligantes

A proteína InhA representa um importante alvo para o controle da tuberculose (Oliveira et al. 2007), pois ela inibe a biossíntese de ácidos micólicos, um importante componente da parede celular da micobactéria, e conseqüentemente uma das estruturas essenciais para a sobrevivência da mesma. A estrutura tridimensional dessa proteína está descrita na Figura 6.4, na forma de *ribbons*. Ao lado da estrutura, na mesma figura, é apresentado parte do arquivo PDB desta proteína. O arquivo PDB da InhA utilizado neste trabalho (PDBCode:1ENY) foi obtido do *Protein Data Bank* (Berman et al 2000). Um arquivo PDB é composto por um cabeçalho, que não está descrito na Figura 6.4, que descreve como a estrutura da proteína foi obtida (o método, a resolução, o artigo que descreve a estrutura, os autores, etc) e pela descrição dos átomos e resíduos que compõem essa estrutura juntamente com suas coordenadas cartesianas.

Estrutura - InhA

PDB - InhA

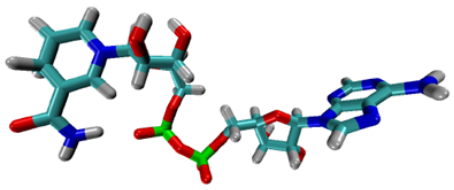
```
ATOM      1  N   ALA A   2       9.036  61.230  21.932  1.00 66.94
ATOM      2  CA  ALA A   2       8.304  60.436  22.930  1.00 65.47
ATOM      3  C   ALA A   2       8.723  58.949  23.009  1.00 63.24
ATOM      4  O   ALA A   2       9.847  58.587  23.415  1.00 63.92
ATOM      5  CB  ALA A   2       6.778  60.550  22.702  1.00 65.37
ATOM      6  N   GLY A   3       7.794  58.097  22.595  1.00 59.43
ATOM      7  CA  GLY A   3       7.959  56.651  22.672  1.00 54.68
ATOM      8  C   GLY A   3       6.751  56.322  23.547  1.00 49.10
ATOM      9  O   GLY A   3       6.798  56.505  24.763  1.00 49.91
ATOM     10  N   LEU A   4       5.621  56.069  22.913  1.00 42.86
ATOM     11  CA  LEU A   4       4.409  55.790  23.644  1.00 37.47
ATOM     12  C   LEU A   4       4.703  54.675  24.622  1.00 33.16
ATOM     13  O   LEU A   4       4.123  54.641  25.700  1.00 31.92
ATOM     14  CB  LEU A   4       3.333  55.321  22.649  1.00 35.56
ATOM     15  CG  LEU A   4       1.921  55.094  23.156  1.00 33.46
...
```

Figura 6.4. Estrutura tridimensional do receptor InhA e parte do arquivo PDB desta estrutura (PDBCode: 1ENY).

No caso da InhA, essa proteína é composta por 4.008 átomos (os números e nomes de alguns desses átomos estão descritos nas colunas 2 e 3 do PDB da Figura 6.4) distribuídos em 268 resíduos (o nome e o número de alguns resíduos estão descritos nas colunas 4 e 5 do PDB descrito na Figura 6.4).

O ligante NADH - Nicotinamida Adenina Dinucleotídeo, forma reduzida (Dessen et al, 1995) é o ligante natural da enzima InhA. Essa molécula tem um total de 71 átomos (após a preparação para a docagem molecular, permanece com um total de 52 átomos, pois perde os hidrogênios polares) e sua estrutura tridimensional está descrita na Figura 6.5, assim como parte do arquivo mol2 deste ligante utilizando nos experimentos de docagem molecular com o mesmo. O arquivo mol2 descreve as coordenadas dos átomos do ligante, a carga de cada átomo (na última coluna onde estão descritos os átomos) e como os átomos são conectados na estrutura do ligante (parte final do arquivo mol2).

Estrutura - NADH



Mol2 - NADH

```
@<TRIPOS>MOLECULE
UNTITLED
71 75 1 0 0
SMALL
USER_CHARGES


@<TRIPOS>ATOM
      1 C6N         -5.5959   -4.3594    1.0216 C.2       1     ****   -0.3550
      2 H6N         -5.0019   -4.4531    0.1243 H         1     ****    0.2220
      3 C5N         -6.1594   -5.4464    1.7507 C.2       1     ****   -0.1730
      4 H5N         -6.0147   -6.4642    1.4173 H         1     ****    0.1260
      5 C4N         -6.9471   -5.1565    2.8972 C.3       1     ****    0.1350
      6 H41         -7.9253   -5.5901    2.6881 H         1     ****    0.0200
...
    64   62   61  1
    65   63   62 ar
    66   64   63  1
    67   65   64  1
    68   66   64  1
    69   67   63 ar
    70   68   67 ar
    71   69   68  1
    72   70   68 ar
    73   71   58  1
    74   71   62 ar
    75   71   70 ar
@<TRIPOS>SUBSTRUCTURE
```

Figura 6.5. Estrutura tridimensional do ligante NADH e parte do arquivo mol2 desta estrutura (PDBCode: 1ENY).

6.4.1.2. Simulações pela DM do receptor InhA

Os estudos de simulação por DM desse receptor, que originaram as conformações utilizadas nesse trabalho, foram realizados com a InhA complexada com o NADH utilizando o software AMBER6.0 (Pearlman et al, 1995) por um período de 3.100 ps (picosegundos = 10^{-12} s) e estão descritos no trabalho de Schroeder et al. (2005). Cada conformação do receptor é um arquivo PDB, conforme descrito na Figura 6.4.

6.4.1.3. Docagem molecular com o receptor InhA e o ligante NADH

Para considerar o receptor flexível nas simulações de docagem molecular, utilizando o ligante NADH no seu formato mol2 (Figura 6.5), foram submetidos 3.100 experimentos de docagem molecular (onde, em cada experimento uma das 3.100 conformações do receptor foi considerada) utilizando o workflow científico descrito em Machado et al. (2007). Esse workflow foi desenvolvido para executar automaticamente este tipo de experimento de docagem molecular com o receptor flexível. O software de docagem molecular utilizado foi o AutoDock3.0.5 (Morris et al. 1998) e como protocolo de execução o algoritmo *Simulated Annealing* (SA) com 10 runs onde o ligante foi mantido rígido. Cada *run* corresponde a uma diferente tentativa de encontrar a melhor energia de

interação entre o receptor-ligante (*Free Energy of Binding* - FEB). Como resultado da execução de um experimento de docagem utilizando o Autodock3.0.5 tem-se um arquivo de saída descrito em parte na Figura 6.6, que lista os resultados da docagem (destacados na figura): as coordenadas finais do ligante (sua posição final após a docagem), a FEB, o valor do *Root Mean Squared Deviation* – RMSD que corresponde a distância do centro de massa do ligante em sua posição inicial e final após a docagem e outros valores para cada uma das 10 tentativas executadas, ordenadas ascendentemente por FEB.

```
Residue number will be set to the conformation's cluster rank.

MODEL        8
USER    Run = 8
USER    Cluster Rank = 1
USER    Number of conformations in this cluster = 1
USER
USER    RMSD from reference structure       = 5.197 A
USER
USER    Estimated Free Energy of Binding    =  -15.18 kcal/mol  [=(1)+(3)]
USER    Estimated Inhibition Constant, Ki   =   +7.52e-12       [Temperature = 298.15 K]
USER
USER    Final Docked Energy                 =  -17.10 kcal/mol  [=(1)+(2)]
USER
USER    (1) Final Intermolecular Energy     =  -17.04 kcal/mol
USER    (2) Final Internal Energy of Ligand =   -0.06 kcal/mol
USER    (3) Torsional Free Energy           =   +1.87 kcal/mol
USER
USER
USER    DPF = LIGmoved.MACRO_pdbqs.dpf
USER    NEWDPF move     LIGmoved.pdbq
USER    NEWDPF about    -4.837000 6.875000 0.114000
USER    NEWDPF tran0    -0.354736 7.653123 4.259508
USER    NEWDPF quat0    -0.510294 -0.778544 0.365334 -38.520748
USER    NEWDPF ndihe    6
USER    NEWDPF dihe0    104.50 157.90 61.05 22.51 -174.54 103.78
USER
USER                  Rank         x       y       z       vdW    Elec       q      RMS
ATOM      1  C5'A***     1      -0.355   7.653   4.260    -0.34  +0.04  +0.192   5.197
ATOM      2  C4'A***     1       0.951   8.331   3.948    -0.45  +0.06  +0.224   5.197
ATOM      3  O4'A***     1       0.876   9.415   2.950    -0.13  -0.01  -0.355   5.197
ATOM      4  C3'A***     1       1.606   8.966   5.195    -0.36  +0.05  +0.264   5.197
ATOM      5  O3'A***     1       2.759   8.146   5.512    +0.02  -0.16  -0.654   5.197
ATOM      6  HO3A***     1       3.344   8.137   4.752    +0.09  +0.11  +0.438   5.197
```

Figura 6.6. Parte do arquivo de saída do software AutoDock3.0.5.

6.4.1.4. Armazenamento dos dados

Como mencionado, uma das etapas do processo de KDD é o desenvolvimento de uma base de dados alvo. Sendo assim, todos os dados sobre o receptor e suas conformações geradas na simulação pela DM, dados sobre o ligante e dados sobre os resultados dos experimentos de docagem molecular, passaram por um processo de ETC que homogeneizou e integrou os dados das diferentes fontes onde os mesmos foram obtidos, e foram armazenados em uma base de dados chamada FReDD (Winck et al., 2009). O modelo desse banco de dados está na Figura 6.7. Atualmente esse banco de dados é composto por 17 tabelas contendo um total de 15.814.183 registros. Esses registros estão relacionados aos dados sobre o receptor InhA e suas 3.100 conformações, sobre o ligante NADH e mais 3 ligantes: Isoniazida Pentacianoferrato-IPF (Oliveira et al, 2004), Triclosano-TCL (Kuo et al, 2003) e Etionamida ou ETH (Wang et al., 2007) . Além disso, estão também armazenados todos os resultados de docagem molecular da InhA flexível e os 4 ligantes. Neste trabalho serão utilizados na etapa de mineração de dados somente os dados do complexo InhA-NADH.

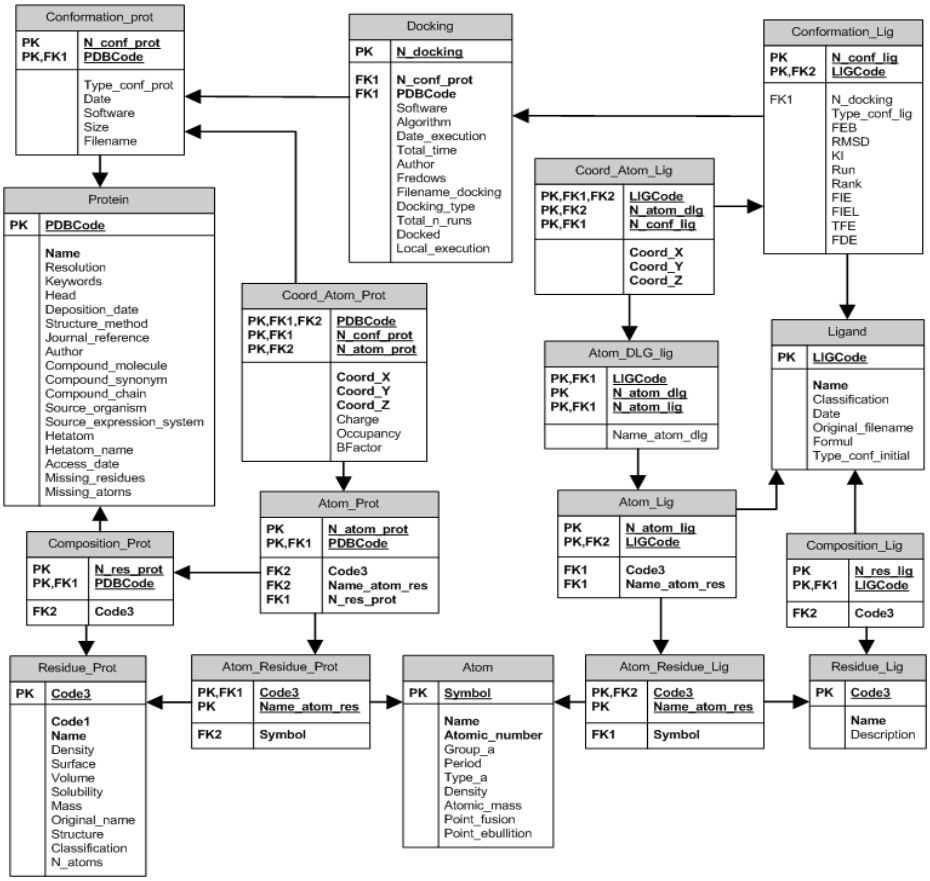


Figura 6.7. Modelo do banco de dados FReDD.

6.4.2. Experimentos com mineração de dados

A construção do repositório FReDD foi motivada para que os dados nele contidos pudessem ser facilmente acessados e preparados para serem utilizados por algoritmos de mineração de dados. O que se espera com a mineração é aumentar o entendimento a respeito do comportamento da flexibilidade do receptor sendo estudado e, com isso, contribuir para o aceleramento das execuções dos experimentos de docagem molecular. Para tanto, busca-se responder algumas perguntas, como:

- Como selecionar um subconjunto de conformações do receptor que seja o mais relevante para predizer se um dado ligante é promissor?
- Como, a partir de ligantes considerados promissores, selecionar outros que também tenham chance de serem promissores?

Para obter informações que direcionem para responder às perguntas acima, nesse trabalho são utilizadas regras de associação, árvores de decisão para classificação e árvore de decisão para regressão.

6.4.2.1. Pré-processamento de dados de docagem molecular

Os dados armazenados no FReDD precisam ser criteriosamente pré-processados para que os diferentes algoritmos de mineração aplicados possam induzir os melhores

modelos possíveis, de modo com que as perguntas enunciadas anteriormente tenham maior chance de serem respondidas.

O primeiro passo do pré-processamento é a escolha do atributo alvo, para as tarefas preditivas. Como sabe-se que uma das maneiras de avaliar os resultados de docagem é através dos valores de FEB, optou-se por utilizar esse atributo como alvo. Durante os 3.100 experimentos de docagem molecular do complexo InhA-NADH, considerando somente a melhor execução (*run*) de cada experimento, o valor médio da FEB foi de -12,9 ± 4,2 Kcal/mol, onde todos os experimentos de docagem convergiram para valores negativos de FEB, sendo o valor máximo de FEB (o mais negativo) de -20,6 Kcal/mol e ovalor mínimo de 0 Kcal/mol. Para árvores de regressão, considera-se o valor real do FEB (Machado et al, 2010b, Winck et al., 2010). Para árvores de decisão para classificação, esse atributo é discretizado em 5 classes, considerando moda e desvio padrão (Machado et al., 2010a) da distribuição da FEB. Para regras de associação o valor de FEB não é considerado.

A preparação dos atributos preditivos busca identificar as distâncias mínimas (representadas em angstroms, Å) entre os átomos de cada resíduo do receptor com os átomos do ligante (Machado et al, 2010b, Winck et al., 2010). Para a determinação dessa distância mínima são calculadas todas as distâncias entre todos os átomos do NADH e todos os átomos de cada resíduo, selecionando a menor entre elas. A Figura 6.8 ilustra esse conceito, apresentando as distâncias mínimas entre o ligante NADH e os resíduos Alanina 259 (ALA259) e Isoleucina 20 (ILE20) do receptor. Neste caso de exemplo, a menor distânicia entre o resíduo ILE20 e o NADH é de 2,35 Å e entre o resíduo ALA259 e o NADH é de 3,98 Å.

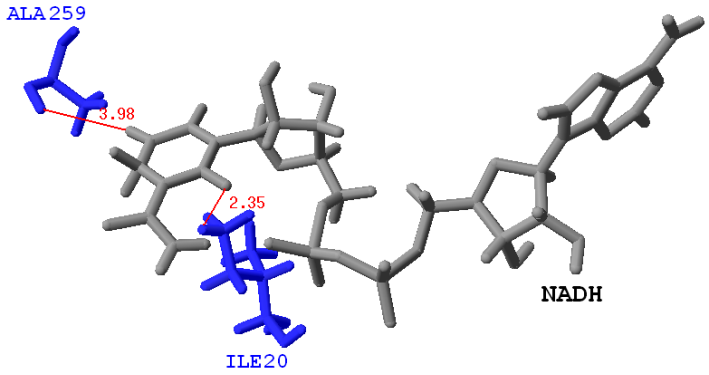


Figura 6.8. Distâncias atômicas entre o ligante NADH e o resíduo ILE20 e ALA259 do receptor InhA.

Para obter essas distâncias é necessário combinar todas as coordenadas do receptor com todas as coordenadas do ligante, a partir dos dados armazenados no repositório FReDD, conforme a consulta SQL ilustrada pela Figura 6.9. Por exemplo, se essa consulta é executada para o receptor InhA (PDBCode=1ENY) e o ligante NADH (LIGCode=NADH), tem-se uma combinação dos 12.424.800 registros de coordenadas do receptor (os 4.008 átomos multiplicados pelas 3.100 conformações), com os 161.200 registros de coordenadas do NADH (os 52 átomos do NADH multiplicados pelos 3100 resultados de docagem molecular para esse ligante). Isso significa que, ao realizar a

consulta ilustrada na Figura 6.9, será produzido mais de 2 trilhões de registros apenas para esse ligante. Os resultados obtidos precisam ser tratados de modo com que sejam transpostos e organizados em um total de 268 atributos (mais o atributo alvo), onde cada atributo corresponde a um resíduo do receptor, sendo que seus valores são atribuídos com as distâncias mínimas com relação ao ligante calculada para cada um desses resíduos. Cada instância corresponde a um experimento de docagem molecular.

```
select N_conf_prot, code3, N_res_prot, feb,
  min((car.Coord_x - cal.Coord_x)**2 +
      (car.Coord_y - cal.Coord_y)**2 +
      (car.Coord_z - cal.Coord_z)**2)
from Atom_Prot inner join Coord_Atom_Prot car
      using(PDBCode, N_atom_prot)
    inner join Conformation_prot
      using(PDBCode, N_conf_prot)
    inner join Docking
      using(PDBCode, N_conf_prot)
    inner join Conformation_lig using(N_docking)
    inner join Coord_Atom_Lig cal
      using(LIGCode, N_conf_lig)
where LIGCode = 'NADH'
  and PDBCode = '1ENY'
group by N_conf_prot, code3, N_res_prot, FEB;
```

Figura 6.9. Consulta SQL para obter as distâncias mínimas entre os átomos do ligante com os átimos do receptor.

A Tabela 6.1 ilustra parte do arquivo de entrada produzido. Esta tabela apresenta a estrutura básica, e inicial, do arquivo de entrada. Para cada experimento de mineração realizado, entretanto, o arquivo de entrada precisa ser ajustado conforme necessidades do algoritmo sendo utilizado.

Tabela 6.1. Exemplo do arquivo de entrada para o complexo InhA-NADH.

Exp. Docagem	...	ILE20	...	ALA259	...	FEB
1	...	2,35	...	3,98	...	-13,81
2	...	3,73	...	9,02	...	-9,75
...	...	...	...	...	...	...
3099	...	6.75	...	19,40	...	-7,59
3100	...	4,21	...	11,00	...	-14,56

6.4.2.2. Experimentos com regressão

Ao utilizar árvores de decisão sobre os dados de docagem molecular, busca-se descobrir quais são os resíduos do receptor e suas distâncias com relação ao ligante que mais contribuem para valores de FEB mais promissores. Considerando que os dados pré-processados contêm, essencialmente, valores numéricos, sobre os mesmos pode-se aplicar técnicas de regressão. Regressão é a tarefa de mineração mais utilizada para

predição de dados numéricos (Han e Kamber, 2006), sendo que existem diferentes algoritmos. Ávores modelo são algoritmos que utilizam-se da abordagem de árvores de decisão, sendo que nos nodos folha são apresentados modelos de regressão que correspondem a uma equação linear para o valor predito (Han e Kamber, 2006). Os modelos lineares (LM – *Linear Models*) podem ser utilizados para quantificar a contribuição de cada atributo para predizer o atributo alvo, no caso o FEB.

Neste trabalho é utilizado o algoritmo M5P (Quinlan, 1992), implementação de árvores modelo no WEKA. Dentre os parâmetros disponíveis para este algoritmo, concentramos na calibragem dos parâmetros relacionados à legibilidade e precisão das árvores modelo induzidas. Portanto, definimos o número mínimo de instâncias para 100. Este tamanho está relacionado com o tamanho da árvore modelo resultante e o número de LM produzidos.

Para reduzir os atributos preditivos, aplicou-se uma seleção de atributos baseada no contexto (Winck et al, 2010), considerando as distâncias mínimas dos resíduos. O maior valor de distância que permite um contato biologicamente significante entre os átomos do receptor e do ligante é 4Å (da Silveira et al., 2009). Assim, os dados de entrada gerados no pré-processamento inicial (Tabela 6.1) são refinados, eliminando todos aqueles atributos cuja distância mínima seja maior do que 4Å.

Executando o M5P sobre esses dados, o modelo induzido produziu uma árvore com 10 nodos e 11 LM, e com uma correlação de 92%. Para melhor entender a saída do M5P, a Figura 6.10 ilustra a árvore modelo induzida para o ligante NADH e a equação 1 ilustra o LM1 gerado. Cada nó interno da árvore (Figura 6.10) corresponde ao código de três letras do aminoácido e seu número na seqüência receptor. Cada ramo representa o limite de distância do ligante para o receptor.

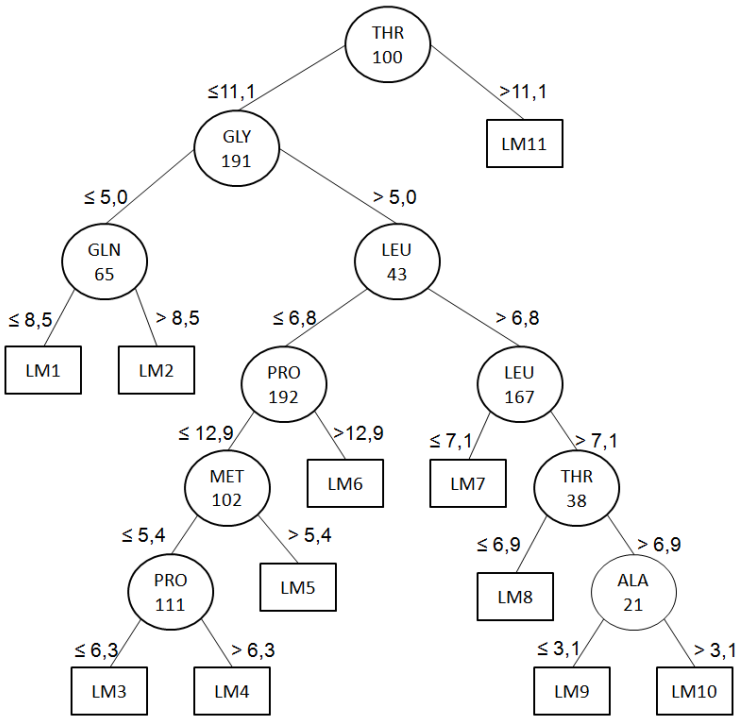


Figura 6.10. Árvore de regressão do complexo InhA-NADH.

```
LM num: 1
FEB = 0.0019 * ILE14 + 0.0141 * SER19 - 0.0129 * ALA21 +
      0.0034 * PHE22 + 0.0015 * HIE23 + 0.0085 * ILE24 +
      0.0157 * THR38 + 0.0165 * LEU43 + 0.0081 * LEU45 +
      0.0027 * LEU62 + 0.0844 * GLN65 - 0.0039 * PHE96 -
      0.0035 * MET97 0.0233 * PRO98 + 0.0179 * THR100 -
      0.0051 * GLY101 - 0.0138 * MET102 + 0.0155 * PRO111 -
      0.0016 * ASP114 - 0.0014 * LYS117 + 0.0125 * GLY118 -
      0.0012 * MET146 - 0.0019 * ALA163 + 0.0855 * LYS164 +
      0.0386 * LEU167 + 0.0338 * ALA190 - 0.0116 * GLY191 +
      0.0358 * PRO192 + 0.0014 * ET198 - 0.0025 * ALA200 +
      0.0142 * ILE201 + 0.0037 * ALA205 - 0.0012 * VAL237 - 13.6659
```
(1)

Pra analisar as árvores e identificar quais LM que indicam valores de FEB mais promissores, é efetuado um pós-processamento sobre os modelos induzidos (Machado et al., 2010; Machado et al., 2010b), com base no valor médio de FEB. Para o caso da Figura 6.10, o modelo linear mais representativo é o LM 11. Nesse sentido, a partir da Figura 6.10 pode-se observar, por exemplo, que sempre que o resíduo Treonina 100 (THR100) está a uma distância superior a 11Å do ligante NADH o modelo linear correspondente é o LM11. Isso significa que um resíduo do receptor que, aparentemente, não é importante (já que está distante do ligante), passa a ser crucial para apontar quais são as conformações que levam a FEB mais promissoras.

6.4.2.3. Experimentos com classificação

Ao submeter o conjunto de dados a um algoritmo de árvore de decisão, temos o mesmo objetivo da regressão: identificar quais conformações podem ter melhores resultados de FEB em experimentos de docagem moleculares. Para efetuar esses experimentos utilizamos o algoritmo J48, implementação do C4.5 (Quinlan, 1986) no WEKA.

Como classificação utiliza-se de classes categóricas, o conjunto de dados pré-processados precisa ser ajustado à classificação. Nesse sentido, o valor de FEB é discretizado em 5 categorias, considerando-se moda e desvio padrão (Machado et al, 2010a). Essas categorias dividem-se em:

- excelente (E);
- bom (B);
- regular (Re);
- ruim (R); e
- muito ruim (MR).

Para gerar árvores mais legíveis, configuramos o parâmetro relacionado ao número mínimo de objetos para 50, mantendo os demais parâmetros em sua configuração original. O algoritmo foi executado considerando a opção de validação cruzada com 10 partes. Como resultado para o NADH obteve-se árvores com acurácia de 73%. A Figura 6.11 ilustra a árvore de decisão induzida para o complexo InhA-NADH. Por essa árvore observa-se que, assim como na árvore modelo produzida (Figura 6.10), o nodo raiz também é formado pelo resíduo THR100 e, embora ele seja um resíduo distante do ligante, é importante para predizer bons resultados de FEB.

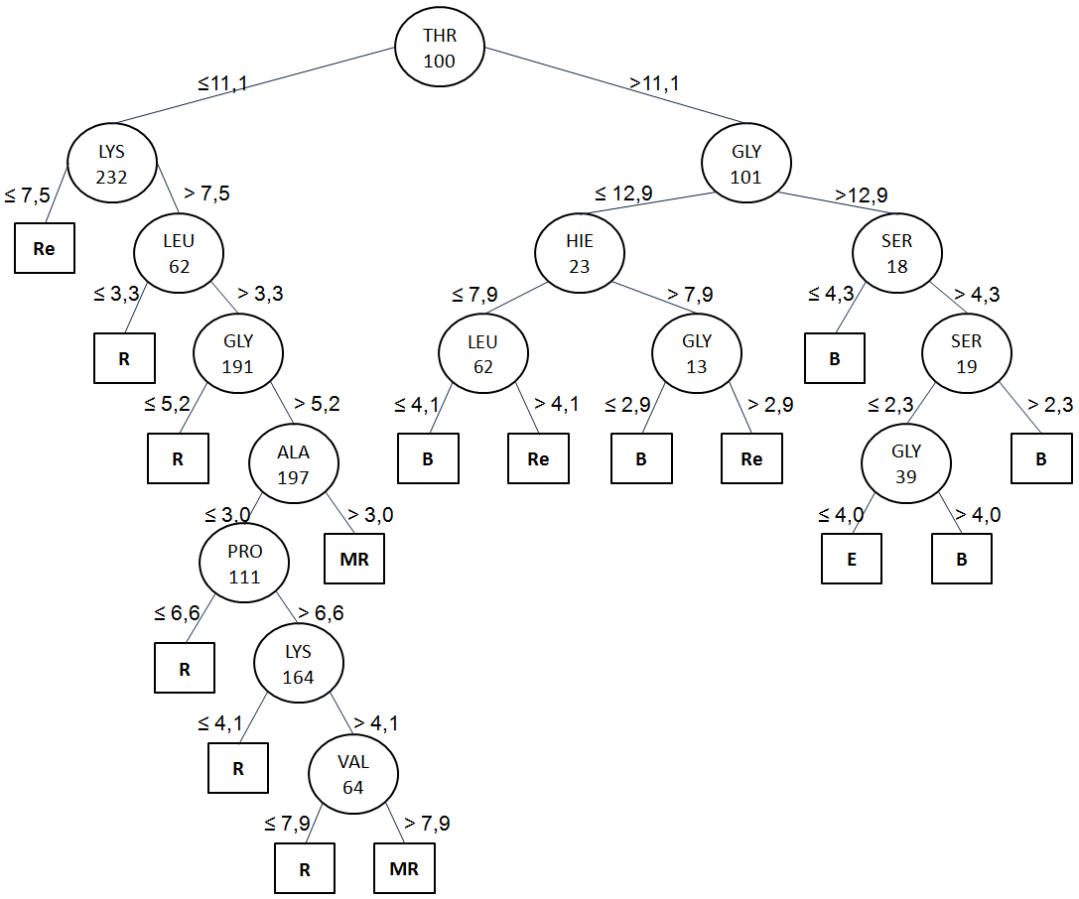


Figura 6.11. Árvore de regressão para o complexo InhA-NADH.

6.4.2.4. Experimentos com regras de associação

Ao utilizar regras de associação sobre o conjunto de dados apresentado nesse trabalho, busca-se identificar relações de interação entre resíduos. Para essas regras, o conjunto de dados utilizado foi preparado em duas etapas. Primeiramente o atributo alvo (FEB) foi removido. Em seguida os valores dos atributos passaram a conter valores binários, indicando a existência de interação dos resíduos do receptor com o ligante: 0 para distâncias maiores que 4Å; e 1 para distâncias menores ou igual a 4Å (Machado et al., 2008).

Os arquivos preparados foram submetidos ao algoritmo *Apriori* (Agrawal, 1993) do WEKA, ajustando o valor de suporte para 0,005 e confiança para 0,9 e um número máximo de 1000 regras. As regras produzidas foram pós-processadas, seguindo a abordagem proposta em Winck et al. (2010a), onde as regras mais especializadas são removidas, atingindo modelos mais enxutos e eficazes. Pra exemplificar algumas regras geradas para o ligante NADH, ilustramos três regras representativas:

- THR100=0 → ILE94=1;
- THR100=0 → SER19=1;
- THR100=0→THR195=1;

Por essas regras nota-se que, quando o resíduo THR100 não interage com o NADH, os resíduos ILE94, SER19 e THR195 interagem. Isso ratifica os padrões identificados através das árvores de que o resíduo THR100, mesmo não interagindo com o ligante, é representativo para indicar aqueles resíduos que interagem.

Muitas outras regras podem ser extraídas. Embora essas regras não estabeleçam relações entre os resíduos do receptor e os valores de FEB, elas podem ser úteis para indicar quais são os resíduos que mais interagem com o ligante sendo estudado. Essa informação pode ser útil na busca de novos ligantes para este receptor, como no trabalho de Quevedo et al. (2010). Em um trabalho anterior (Machado et al, 2008), ao utilizar apenas 40 resultados de experimentos de docagem e 40 conformações do receptor, foi possível extrair conhecimento sobre os resíduos de receptor que mais interagem com os ligantes estudados.

6.5. Considerações

Neste trabalho foram descritas todas as etapas envolvidas em um processo de KDD aplicado à bioinformática no contexto de desenho racional de fármacos, utilizando especificamente resultados de docagem molecular. Os experimentos de docagem molecular com um receptor flexível com 3.100 conformações e um ligante, o complexo InhA-NADH, foram executados previamente e os resultados todos armazenados um repositório apropriado. Os dados armazenados no repositório FReDD foram então preprocessados e assim preparados para serem utilizados nos algoritmos de mineração. Os algoritmos de mineração utilizados foram: M5P de árvores de regressão, J48 para árvores de decisão e Apriori para regras de associação.

Os resultados obtidos com as diferentes técnicas de mineração aplicadas mostram alguns exemplos de informações que podem ser obtidas sobre os experimentos de docagem molecular, que não seria possível de serem extraídas sem um processo de KDD. Um exemplo são os resíduos que aparecem tanto na árvore de regressão quanto na árvore de decisão do NADH, que são resíduos que em uma inspeção visual com uma conformação desse receptor e o NADH não parecem estar em contato com o mesmo (não estão a uma distância menor do que 4Å do ligante). Com as informações obtidas durante esse processo de KDD espera-se que, no futuro, seja possível acelerar os experimentos de docagem molecular, utilizando com novos e diferentes ligantes as conformações indicadas como mais promissoras nos experimentos já executados.

6.6. Biografias

Ana Trindade Winck possui graduação em Ciência da Computação pela Universidade Feevale (2005) e mestrado em Ciência da Computação pela PUCRS (2007). Atualmente é aluna de doutorado em Ciência da Computação pela PUCRS, com início em 2008 e previsão de conclusão em 2011. É integrante do GPIN, e desenvolve sua tese com ênfase no pré-processamento de dados biológicos. Tem desenvolvido pesquisas em catalogação e mineração de dados de docagem molecular. Possui interesse em bioinformática e mineração de dados. Tem proferiu palestras e ministrou minicurso nestas áreas.

Karina dos Santos Machado possui graduação em Engenharia de Computação pela FURG (2004) e mestrado em Ciência da Computação pela PUCRS (2007). Atualmente é aluna do último ano de doutorado do curso de Ciência da Computação da PUCRS,

integrante do LABIO. Desenvolve sua tese na aplicação de diferentes técnicas de mineração de dados à seleção de conformações do receptor. Tem desenvolvido pesquisas em catalogação e mineração de dados de docagem molecular. Possui interesse em bioinformática e mineração de dados, e experiência em ministrar minicurso na área.

Duncan Dubugras Alcoba Ruiz possui graduação em Engenharia Elétrica pela Universidade Federal do Rio Grande do Sul (1983), mestrado em Ciências da Computação pela Universidade Federal do Rio Grande do Sul (1987) e doutorado em Ciências da Computação pela UFRGS (1995). Atualmente é professor adjunto da PUCRS, onde coordena o GPIN. Tem experiência na área de Ciência da Computação, com ênfase em Banco de Dados, atuando principalmente nos seguintes temas: banco de dados, *workflow modeling*, bancos de dados temporais, engenharia de software e KDD.

Osmar Norberto de Souza possui graduação em Física pela Universidade Federal de Minas Gerais (1987), especialização em Biotecnologia Moderna pelo Centro de Biotecnologia da UFRGS (1988), mestrado e doutorado em Biofísica Molecular Computacional - University of London (1994). Trabalhou de 1995 a 1998, no Environmental Molecular Sciences Laboratory do Pacific Northwest National Laboratory do Departamento de Energia (DOE) dos EUA e de 1999 a 2001, no Laboratório Nacional de Luz Síncrotron (LNLS), Campinas, SP. Desde 2002 é professor adjunto da PUCRS, onde coordena o LABIO. Atualmente é bolsista de produtividade em pesquisa CNpQ - Nível 2. Tem experiência nas áreas de Bioinformática Estrutural e Biofísica, com ênfase em Biofísica Molecular Computacional, atuando principalmente nos seguintes temas: simulação computacional por dinâmica molecular, estrutura e dinâmica de proteínas e complexos proteína-ligante, protocolos e algoritmos computacionais para a predição da estrutura tridimensional de proteínas, planejamento de fármacos assistido por computador, tuberculose e malária.

Referências

Agrawal, R., Imielinski, T., Swami, A. (1993). "Mining association rules between sets of items in large databases". In Conference on Management of Data, Washington DC, 207-216.

Alpaydin, E. Introduction to machine learning. The Mit Press, Cambridge, 2004.

Amaro, R.E., Baron, R., McCammon J.A. (2008). "An Improved Relaxed Complex Scheme for Receptor Flexibility in Computer-aided Drug Design." Journal of Computer Aided Molecular Design, 22:693-705.

Balakin, K. V. Pharmaceutical data mining: approaches and applications for drug discovery. John Wiley & Sons, New York, 2009.

Benson, D.A., Karsch-Mizrachi, I., Lipman D.J., Ostell J., Wheeler D.L. (2008). "GenBank". Nucleic Acids Res., 36, Database issue D25-D30.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., Bourne, P. E. (2000). "PDB - Protein Data Bank". Nucleic Acids Research, 28:235-242.

Broughton, H. B. (2000). "A Method for Including Protein Flexibility in Protein-ligand Docking: Improving Tools for Database Mining and Virtual Screening". Journal of Molecular Graphics and Modelling, 18: 247-257.

Chandrika, B., Subramanian, J., Sharma, S.D. (2009). “Managing Protein Flexibility in Docking and its Applications” Drug Discovery Today, 14:394-400.

da Silveira, C.H., Pires, D.E.V., Minardi, R.C., Ribeiro, C., Veloso, C.J.M., Lopes, J.C.D., Meira Jr, W., Neshich, G., Ramos, C.H.I., Habesch, R., Santoro, M.M. (2009). “Protein Cutoff Scanning: A comparative Analysis of Cutoff Dependent and Cutoff Free Methods for Prospecting Contacts in Proteins”. Proteins, 74:727-743.

Dessen, A., Quemard, A., Blanchar, J.S. Jacobs Jr, W.R., Sacchenttini, J.C. (1995). “Crystal structure and function of the isoniazid target of Mycobacterium tuberculosis”. Science. 267(5204):1638-41

Drews, J. (2003). “Strategic trends in the drug industry”. Drug Discovery Today, 8:411-420.

Fayyad, U., Piatestsky-S, G., Smyth, P. (1996). “The KDD process for extracting useful knowledge from volumes of data”. Communications of the ACM, 39:27-34.

Freitas, A., Wieser, D., Apweiler, R. (2010). “On the importance of comprehensible classification models for protein function prediction”. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 7:172-182.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutmann, P., Witten, I. (2009). “The WEKA data mining software: an update”. SIGKDD Explorations, 11:10-18.

Han, J., Kamber, M. Data Mining: concepts and techniques. Morgan & Kaufmann, San Francisco, 2006.

Hunter, L. Artificial intelligence and molecular biology. American Association for Artificial Intelligence. Menlo Park, CA, USA, 1993.

Inmon, W.H. Como construir um data warehouse. Campus, Rio de Janeiro, 1997.

Kimball, R., Ross, M. The datawarehouse toolkit: the complete guide to dimensional modeling. John Wiley & Sons, New York, 2002.

Kuntz, I. D. (1992). “Structure-based strategies for drug design and discovery”. Science, 257: 1078-1082.

Kuo, M.R., Morbidoni, H. R., Alland, D., Sneddon,S. F., Gourlie, B. B., Staveski, M. M., Leonard, M., Gregory,J. S., Janjigian,A. D., Yee, C., Musser, J.M., Kreiswirth, B., Iwamoto, B., Perozzo, R., Jacobs Jr, W.R., Sacchettini, J.C., Fodock, D.A. (2003). “Targeting Tuberculosis and Malaria through Inhibition of Enoyl Reductase: Compound Activity and Structural Data”. Journal of Biological Chemistry, 278: 20851-20859.

Lesk, A. Introduction to bioinformatics. Oxford University Press, 2002.

Lin, J-H., Perryman, A.L., Schames, J.R., McCammon, J.A. (2002) “Computational drug design accommodating receptor flexibility: the relaxed complex scheme” Journal of American Chemical Society. 124:5632-5633.

Luscombe, N.M., Greenbaum, D., Gerstein, M. (2001). “What is bioinformatics? A proposed definition and overview of the field” Methods Information in Medicine, 40:346-358.

Lybrand, T.P. (1995). “Ligand-protein docking and rational drug design”. Current Opinion in Structural Biology, 5: 224-228.

Machado, K.S., Schroeder, E.K., Ruiz, D.D., Norberto de Souza, O. (2007). “Automating molecular docking with explicit receptor flexibility using scientific workflows”. In: Second Brazilian Symposium on Bioinformatics, LNCS, 1-11, Rio de Janeiro.

Machado, K.S., Schroeder, E.K., Ruiz, D.D., Winck, A.T., Norberto de Souza, O. (2008). “Extracting information from flexible ligand-flexible receptor docking experiments”. In: Third Brazilian Symposium on Bioinformatics, LNCS, 104-112, Santo André.

Machado, K.S., Winck, A.T., Ruiz, D.D., Norberto de Souza, O. (2010a). “Discretization of Flexible-Receptor Docking Data”. In: Fifth Brazilian Symposium on Bioinformatics, LNCS, 6268:75-79.

Machado, K.S., Winck, A.T., Ruiz, D.D., Norberto de Souza, O. (2010b). “Mining flexible-receptor docking experiments to select promising protein receptor snapshots”. BMC Genomics. To be published.

Machado, K.S., Winck, A.T., Ruiz, D.D., Norberto de Souza, O. (2010). “Applying model trees on flexible-receptor docking experiments to select promising protein receptor snapshots” In: ISCB Latin America, 66-66, Montevideo.

Morris, G.M., Goodsell, D.S., Halliday, R.S., Huey, R., Hart, W.E., Belew, R.K., Olson, A.J. (1998). “Automated Docking Using a Lamarckian Genetic Algorithm and Empirical Binding Free Energy Function”. Journal of Computational Chemistry, 19: 1639-1662.

Mount, DW. Bioinformatics: Sequence and Genome Analysis. Cold Spring Harbor Laboratory Press. New York. 2004.

Oliveira, J.S., Moreira, I.S., Santos, D.S., Basso, L.A. (2007). “Enoyl reductases as targets for the development of anti-tubercular and anti-malarial agents”. Current Drug Targets, 8:399-411.

Oliveira, J.S., Souza, E.H.S., Basso, L.A., Palaci, M., Dietze, R., Santos, D.S., Moreira, I. (2004). “An Inorganic Iron Complex that Inhibits Wild-type and an Isoniazid-resistant Mutant 2-transenoyl-ACP (CoA) Reductase from Mycobacterium tuberculosis”. Chemical Communication, 15:312-313.

Pearlman, D.A., Case, D.A, Caldwell, J.W., Ross, W.R., Cheatham, T.E.., DeBolt, S., Fergunson, D., Seibel, G., Kollman, P. (1995). “AMBER, a computer program for applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to elucidate the structures and energies of molecules”. Computer Physics Communications., 91:1-41.

Quevedo C.V., Winck, A.T., Machado, K.S., Norberto de Souza, O., Ruiz D.D. (2010). “A study of molecular descriptor to rank candidate ligands to inhibit the InhA Receptor”. In: ISCB Latin America, 79-79, Montevideo.

Quinlan, J.R. (1986). “Introduction to decision trees”. Machine Learning, 1:81-106.

Quinlan, J.R. (1992). “Learning with continuous classes” In Fifth Australian Joint Conference on Aritifical Intelligence, Singapure, 343-348.

Schroeder, E.K., Basso, L.A., Santos, D.S., Norberto de Souza, O. (2005). "Molecular Dynamics Simulation Studies of the Wild-Type, I21V, and I16T Mutants of Isoniazid-Resistant Mycobacterium tuberculosis Enoyl Reductase (InhA) in Complex with NADH: Toward the Understanding of NADH-InhA Different Affinities". Biophysics Journal, 89:876-884.

Tan, P-N., Steinbach, M., Kumar, V. Introduction to data mining. Addison-wesley, Boston, 2006.

Totrov, M., Abagyan, R. (2008). "Flexible Ligand Docking to Multiple Receptor Conformations: a Practical Alternative". Current Opinion on Structural Biology, 18:178-184.

van Gunsteren, W.F., Berendsen, H.J.C. (1990). "Computer Simulation of Molecular Dynamics: Methodology, Applications and Perspectives in Chemistry". Angewandt Chemie International Edition, 29:992-1023.

Wang, F., Langley, R., Gulten, G., Dover, L.G., Besra, G.S., Jacobs Jr, W.R., Sacchettini, J.C. (2007). "Mechanism of thioamide drug action against tuberculosis and leprosy". Journal of Experimental Medicine, 204:73-78.

Wang, J. T. L, Zaki, M. J. and Toivonen, H. T. T. Data mining in bioinformatics: advances information and knowledge processing, Springer, 2005.

Winck, A.T., Machado, K.S., Norberto de Souza, O., Ruiz, D.D. (2010). "A context-based preprocessing on flexible-receptor docking data". In: ISCB Latin America, 68-68, Montevideo.

Winck, A.T., Machado, K.S., Norberto de Souza, O., Ruiz, D.D. (2009). "FReDD: supporting mining strategies through a flexible-receptor docking database". In: Fourth Brazilian Symposium on Bioinformatics, LNCS, 6098:143-146, Porto Alegre.

Winck, A.T., Machado, K.S., Ruiz, D.D., Strube de Lima, V.L. (2010a). "Association rules to identify receptor and ligand structures through named entities recognition" In: The Twenty Third International Conference on Industrial Engineering & Other Applications of Applied Intelligent Systems, LNCS, 119-128, Córdoba.