Universidade de São Paulo
Escola de Engenharia de São Carlos

**Análise de grandezas cinemáticas e dinâmicas inerentes à
hemiparesia através da descoberta de conhecimento em
bases de dados**

Caio Benatti Moretti

São Carlos
2016

Caio Benatti Moretti

**Análise de grandezas cinemáticas e dinâmicas inerentes à
hemiparesia através da descoberta de conhecimento em
bases de dados**

Dissertação apresentada à Escola de Engenharia de São Carlos
da Universidade de São Paulo, como parte dos requisitos para
obtenção do título de Mestre em Engenharia Mecânica

Área de Concentração: Dinâmica de Máquinas e Sistemas
Orientador: Prof. Dr. Glauco Augusto de Paula Caurin

ESTE EXEMPLAR TRATA-SE DA
VERSÃO CORRIGIDA. A VERSÃO
ORIGINAL ENCONTRA-SE DISPO-
NÍVEL JUNTO AO DEPARTAMENTO
DE ENGENHARIA MECÂNICA DA
EESC-USP.

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

M844a Moretti, Caio Benatti
Análise de grandezas cinemáticas e dinâmicas
inerentes à hemiparesia através da descoberta de
conhecimento em bases de dados / Caio Benatti Moretti;
orientador Glaucio Augusto de Paula Caurin. São Carlos,
2016.

Dissertação (Mestrado) - Programa de Pós-Graduação
em Engenharia Mecânica e Área de Concentração em
Dinâmica de Máquinas e Sistemas -- Escola de Engenharia
de São Carlos da Universidade de São Paulo, 2016.

1. Descoberta de conhecimento. 2. Mineração de
dados. 3. Aprendizado de máquina. 4. Reabilitação
robótica. I. Título.

FOLHA DE JULGAMENTO

Candidato: Bacharel **CAIO BENATTI MORETTI**.

Título da dissertação: "Análise de grandezas cinemáticas e dinâmicas inerentes à hemiparesia através da descoberta de conhecimento em bases de dados".

Data da defesa: 31/03/2016.

Comissão Julgadora:

Resultado:

Prof. Associado **Glaucio Augusto de Paula Caurin**
(Orientador)
(Escola de Engenharia de São Carlos/EESC)

APROVADO

Prof. Associado **Alexandre Claudio Botazzo Delbem**
(Instituto de Ciências Matemáticas e de Computação/ICMC)

APROVADO

Prof. Dr. **Rafael Vidal Aróca**
(Universidade Federal de São Carlos/UFSCar)

APROVADO

Coordenador do Programa de Pós-Graduação em Engenharia Mecânica:
Prof. Associado **Gherhardt Ribatski**

Presidente da Comissão de Pós-Graduação:
Prof. Associado **Paulo César Lima Segantine**

Dedicatória

À querida Osmilda (*in memoriam*), matriarca, fonte incessante de amor e inspiração, ainda
que liberta da matéria que um dia nos amparou.

Agradecimentos

Eis a seguir os meus agradecimentos àqueles que, de alguma forma, contribuíram para o êxito deste trabalho, proporcionando a conclusão de mais uma etapa.

Ao Prof. Dr. Glauco Augusto de Paula Caurin pela confiança e gentil manifestação de interesse na orientação, me proporcionando maior aproximação do labor científico, assim como pela supervisão no desenvolvimento deste trabalho e a disposição para resolver quaisquer problemas que estivessem fora da minha alçada.

Aos Profs. Me. Kleber de Oliveira Andrade e Dr. José Martins Jr. pelo incentivo e recomendação do departamento de Engenharia Mecânica da Escola de Engenharia de São Carlos para o ingresso no programa de pós-graduação. Agradeço também ao Prof. Dr. Pedro Roberto Grosso pela disposição em me auxiliar nos preparativos para o ingresso neste programa.

À minha avó Osmilda Oliveira Benatti (*in memoriam*) a minha inestimável gratidão pelo indubitável e pronto amparo em sua morada na iminência de um infortúnio, sem o qual não seria possível a continuidade dos preparativos para a aprovação neste programa. Ainda a ela, e estendendo também este sentimento à minha mãe Ana Eliza Oliveira Benatti e irmão Ralph Benatti Moretti, agradeço pelo enérgico apoio e preocupação, e também pelas comemorações a cada pequeno passo concluído ao longo desta caminhada. À minha mãe, de forma breve, mas com o ímpeto das mais delongadas epígrafes, ressalto a minha gratidão e admiração pelo vigor nas energias positivas, até mesmo depois do passamento da *vó*.

Aos amigos e funcionários do laboratório (Viviane, Coronel, Romulo, Kleber, Henrique, Gustavo, Thales, Marão, João, Marcello, Leonardo, Rafael e Cris), sempre dispostos a dividir e discutir eventuais problemas, e também ao Ricardo Joaquim, que vivenciou os mesmos anseios e esteve sempre à disposição para somar esforços.

Aos meus queridos amigos da república, Lemuel, Erick, *Tiagão* e Tanaka, pela excelente convivência e inúmeros momentos de lazer que amenizaram a tensão durante grande parte desta caminhada, e também pela paciência e tempo dedicados quando precisei de segundas opiniões em prévias de apresentações e demais trabalhos. O meu muito obrigado a vocês, que demonstraram ser meus amigos ao renunciar de horas a mais sono para o prestígio de etapas importantes, como o exame de qualificação.

Ao Tiago Santana de Nazaré, o *Tiagão*, pelas incansáveis, longas e proveitosas discussões que muitas vezes extrapolaram algumas horas da madrugada, as quais culminaram em ganhos para o desenvolvimento deste trabalho.

Ao Prof. Dr. Stefano Mazzoleni pela sugestão de uma diretriz desafiadora para pesquisa, a qual compõe o objetivo deste trabalho, proporcionando novos aprendizados e oportunidade para trabalhos futuros.

À Thais Terranova e Dra. Linamara Battistella, do Instituto de Medicina Física e Reabilitação (IMREA), da Rede Lucy Montoro, pelo fornecimento dos dados utilizados no experimento deste trabalho, assim como por toda a assistência prestada sob o ponto de vista clínico, aprimorando a perspectiva multidisciplinar abordada.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo financiamento desta pesquisa, sob o processo 830724/1999-0.

“Quanto mais carregada estiver a folha de arroz, mais ela se curva.”

Yoshihide Shinzato

Resumo

MORETTI, C. B. **Análise de grandezas cinemáticas e dinâmicas inerentes à hemiparesia através da descoberta de conhecimento em bases de dados** 2016. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2016.

Em virtude de uma elevada expectativa de vida mundial, faz-se crescente a probabilidade de ocorrer acidentes naturais e traumas físicos no cotidiano, o que ocasiona um aumento na demanda por reabilitação. A terapia física, sob o paradigma da reabilitação robótica com *serious games*, oferece maior motivação e engajamento do paciente ao tratamento, cujo emprego foi recomendado pela *American Heart Association* (AHA), apontando a mais alta avaliação (Level A) para pacientes internados e ambulatoriais. No entanto, o potencial de análise dos dados coletados pelos dispositivos robóticos envolvidos é pouco explorado, deixando de extrair informações que podem ser de grande valia para os tratamentos. O foco deste trabalho consiste na aplicação de técnicas para descoberta de conhecimento, classificando o desempenho de pacientes diagnosticados com hemiparesia crônica. Os pacientes foram inseridos em um ambiente de reabilitação robótica, fazendo uso do InMotion ARM, um dispositivo robótico para reabilitação de membros superiores e coleta dos dados de desempenho. Foi aplicado sobre os dados um roteiro para descoberta de conhecimento em bases de dados, desempenhando pré-processamento, transformação (extração de características) e então a mineração de dados a partir de algoritmos de aprendizado de máquina. A estratégia do presente trabalho culminou em uma classificação de padrões com a capacidade de distinguir lados hemiparéticos sob uma precisão de 94%, havendo oito atributos alimentando a entrada do mecanismo obtido. Interpretando esta coleção de atributos, foi observado que dados de força são mais significativos, os quais abrangem metade da composição de uma amostra.

Palavras-chave: Descoberta de conhecimento em bases de dados. KDD. Mineração de dados. Aprendizado de máquina. Reabilitação. Robótica de reabilitação.

Abstract

MORETTI, C. B. **Analysis of kinematic and dynamic data inherent to hemiparesis through knowledge discovery in databases** 2016. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2016.

As a result of a higher life expectancy, the high probability of natural accidents and traumas occurrences entails an increasing need for rehabilitation. Physical therapy, under the robotic rehabilitation paradigm with serious games, offers the patient better motivation and engagement to the treatment, being a method recommended by American Heart Association (AHA), pointing the highest assessment (Level A) for inpatients and outpatients. However, the rich potential of the data analysis provided by robotic devices is poorly exploited, discarding the opportunity to aggregate valuable information to treatments. The aim of this work consists of applying knowledge discovery techniques by classifying the performance of patients diagnosed with chronic hemiparesis. The patients, inserted into a robotic rehabilitation environment, exercised with the InMotion ARM, a robotic device for upper-limb rehabilitation which also does the collection of performance data. A Knowledge Discovery roadmap was applied over collected data in order to preprocess, transform and perform data mining through machine learning methods. The strategy of this work culminated in a pattern classification with the ability to distinguish hemiparetic sides with an accuracy rate of 94%, having eight attributes feeding the input of the obtained mechanism. The interpretation of these attributes has shown that force-related data are more significant, comprising half of the composition of a sample.

Keywords: Knowledge Discovery in Databases. KDD. Data Mining. Machine learning. Rehabilitation. Rehabilitation Robotics

Publicações

1. MORETTI, C. B.; ANDRADE, K. O.; CAURIN, G. A. P. (2013). Physiotherapy support web-based system for rehabilitation robotics: an initial architecture. In: *22nd. International Congress of Mechanical Engineering (COBEM 2013)*. 2013, p. 1171–1180.
2. MORETTI, C. B.; JOAQUIM, R. C.; CAURIN, G. A.; KREBS, H. I.; MARTINS, J. (2014). Knowledge discovery, rehabilitation robotics, and serious games: Examining training data. In: *Biomedical Robotics and Biomechatronics (2014 5th IEEE RAS EMBS International Conference on)*. 2014, p. 567–572.

Lista de Figuras

2.1	Importância da visualização de dados em duas dimensões (THE HOLZINGER GROUP)	13
2.2	Fluxo de dados do processo de Fayyad et al. Adaptado de FAYYAD et al. (1996). 14	
2.3	Processo de KDD CRISP-DM. Adaptado de CHAPMAN et al. (2000)	17
2.4	Processo de KDD híbrido. Adaptado de CIOU E KURGAN (2005)	18
2.5	Empenho aplicado às etapas de KDD. Adaptado de KURGAN E MUSILEK (2006) 21	
2.6	Técnicas de aprendizado de máquina para mineração de dados.	24
2.7	Situações de <i>underfitting</i> e <i>overfitting</i>	27
3.1	Neurônios biológico e artificial	37
3.2	Estrutura de uma rede PMC. Adaptado de SILVA et al. (2010)	39
4.1	Dispositivo robótico utilizado nas sessões de terapia física	46
4.2	Inspeção visual dos casos de separabilidades mais evidentes	52
4.3	Árvore de decisão utilizada para seleção de atributos	54
4.4	Visualização de atributos de forma isolada	55
5.1	Inspeção visual da fronteira de separabilidade do PMC	60

Lista de Tabelas

2.1	Comparativo das análises de dados em trabalhos de reabilitação robótica	10
2.2	Comparativo de processos de KDD. Adaptado de KURGAN E MUSILEK (2006) e CIOS et al. (2007).	20
4.1	Dados do <i>grid search</i> e resultados dos treinamentos	56

Sumário

Resumo	xi
Abstract	xiii
Publicações	xv
Lista de Figuras	xvii
Lista de Tabelas	xix
1 Introdução	1
1.1 Motivação	3
1.2 Objetivos	4
1.2.1 Objetivos específicos	4
1.3 Estrutura do texto	5
2 Revisão Bibliográfica	7
2.1 Reabilitação robótica	7
2.2 Descoberta de conhecimento em bases de dados	11
2.2.1 Processos de KDD	13
2.2.2 Mineração de dados	23
3 Aprendizado de Máquina e Mineração de Dados	29
3.1 <i>k-Nearest Neighbors</i>	30
3.2 Árvore de decisão	32
3.2.1 Algoritmo <i>Dividir-para-conquistar</i>	32
3.2.2 Atributos contínuos	33
3.2.3 <i>Pruning</i>	34
3.3 <i>Random Forest</i>	35
3.4 <i>Perceptron</i> multicamadas	36

3.4.1	Neurônio artificial	37
3.4.2	Arquitetura	39
3.4.3	Algoritmo de retropropagação	40
4	Análise e extração de características inerentes à hemiparesia	45
4.1	Entendimento do problema e definição de metas	48
4.2	Seleção do conjunto de dados	48
4.3	Pré-processamento	49
4.4	Transformação	50
4.5	Definição de técnicas e algoritmos	53
4.6	Mineração de dados	54
5	Interpretação da saída do KDD	59
6	Considerações finais	63
6.1	Perspectivas futuras	63
	Referências Bibliográficas	65

Capítulo 1

Introdução

A demanda por atividades de cunho reabilitativo se mostrou relevante nos últimos anos, em virtude de uma crescente expectativa de vida mundial. À medida que o contingente populacional se expande, presume-se também o aumento do número de incidentes que ocasionam graves enfermidades, como os acidentes de trânsito. Segundo o Departamento Nacional de Infraestrutura em Transportes (DNIT), em 2010, o número de acidentes envolvendo feridos é quase nove vezes maior do que o número de acidentes envolvendo mortes (DNIT, 2010), o que torna esta parcela de sobreviventes, ou ainda aqueles que foram recuperados física e mentalmente após o trauma, suscetível a apresentar incapacidades motoras, carecendo do exercício da reabilitação.

Além disso, um aumento da população mundial implica em um maior percentual de idosos, os quais, devido aos avanços das tecnologias voltadas à saúde, dispõem de quadros clínicos mais estáveis. Entretanto, ainda que, para estes, nenhuma patologia tenha sido diagnosticada, é cogitável a prática de atividades reabilitativas, de forma a manter um bom potencial motor, ou ainda redesenvolvê-lo nos membros que porventura apresentem enfraquecimento. Agrega-se este coletivo, portanto, àqueles suscetíveis à reabilitação.

As desordens no desempenho motor nem sempre são decorrentes de lesões locais. Fatores extrínsecos também devem ser ponderados, como a integridade de impulsos enviados do sistema nervoso ao membro a desempenhar apropriadamente suas funções motoras (HORAK, 1990). O Acidente Vascular Encefálico (AVE), sucedendo eventualmente de traumatismo crânio-encefálico, sucede também de causas naturais, orgânicas, não originadas de ações externas ao organismo. Circunstâncias estas, nas quais não existem meios para detecções precoces, favorecem ainda mais o número dos que carecem por reabilitação. De acordo com a Organização

Mundial da Saúde, a cada ano, 15 milhões de pessoas sofrem de AVE no mundo, sendo que dois terços deste todo sobrevivem; destes, apenas um terço apresenta condições reabilitáveis (WHO , 2012).

Ao unir estas principais causas, pode-se observar uma demanda por serviços de reabilitação assumindo maiores proporções, o que vem fomentando pesquisas (REGO et al. , 2010; KREBS et al. , 2008, 2003) por novos protocolos e abordagens de reabilitação, como o uso de robôs e jogos digitais.

Adicionar dispositivos robóticos às sessões de terapia física denota uma nova abordagem que vem crescendo consideravelmente (ANDRADE et al. , 2013) em pesquisas, cujo complemento evidencia uma gama de benefícios, proporcionando alcançar resultados significativos em menos tempo. A terapia assistida por robôs, para minimizar dificuldades motoras de membros superiores, causadas por AVE, em 2010, foi recomendada pela American Heart Association (AHA), recebendo a mais alta avaliação para pacientes internados e ambulatoriais (MILLER et al. , 2010). O *Department of Veterans Affairs* e o *Department of Defense*, ambos dos Estados Unidos, também recomendam a terapia assistida por robôs, como um complemento à terapia convencional para pacientes com déficit nas funções motoras dos braços (DEPT. OF VETERANS AFFAIRS E DEPT. OF DEFENSE , 2010).

Em termos práticos, os dispositivos robóticos em geral sustentam o membro a ser exercitado e reproduzem as mesmas movimentações de determinados exercícios fisioterápicos para os quais foram projetados, colocando em atividade um ou mais graus de liberdade simultâneos. Estes dispositivos geralmente oferecem exercícios com movimentos passivos, ativo-assistidos e ativo-resistivos (SANTOS , 2013).

Movimentos passivos são aqueles em que o robô é o único agente causador das movimentações, o que atribui ao paciente um papel meramente passivo, por ser apenas conduzido pelo robô. Nos movimentos ativo-assistidos toda a movimentação envolve os esforços do paciente, porém ainda é necessário o auxílio prestado pelo robô. Os movimentos ativo-resistidos, em contraste, assumem uma ação oposta, apresentando resistência à movimentação realizada pelo paciente. A combinação destes exercícios, no decorrer das sessões de terapia, oferece um grande potencial para revigorar o engajamento do paciente ao tratamento, provendo auxílio, em casos de dificuldade, ou ainda proporcionando desafios, caso o auxílio do robô não seja mais necessário. É notória, portanto, a capacidade de motivar o paciente, evitando eventuais desmotivações ou tédios.

Este benefício, além dos pacientes, se estende aos profissionais que conduzem os tratamentos. O desgaste físico, por exemplo, pode ser uma questão a ser cessada, uma vez que o contato físico com pacientes é realizado pelos dispositivos robóticos, cabendo ao terapeuta instruir e monitorar a atividade dos pacientes. Geralmente também é possível usufruir de estatísticas de desempenho, as quais são computadas pelos robôs, oferecendo diretrizes mais sólidas ao terapeuta, em futuras tomadas de decisão, para melhor conduzir os tratamentos.

A atividade computacional dos robôs, além da aplicação no contexto mecatrônico, se estende, para determinada parcela de dispositivos robóticos, aos jogos digitais, denominados *Serious games* (Jogos sérios). Estes jogos têm o objetivo de motivar o paciente, bem como fornecer estímulos visuais, desviando a atenção dos esforços físicos. ZYDA (2005) define um *serious game* como um jogo destinado a um propósito específico (educação, saúde, ou outros), tendo o entretenimento como atributo secundário. A combinação entre jogos e robôs, voltada à terapia, se baseia no cenário onde os pacientes manejam os dispositivos robóticos como interfaces, cujas movimentações realizadas representam o meio para interagir com o ambiente virtual.

1.1 Motivação

Os softwares dos jogos que são executados junto aos robôs geralmente realizam coletas de dados, registrando todo o desempenho do paciente no decorrer das sessões de terapia. Todos estes dados são então retidos em bases de dados para análises posteriores, oferecendo um grande potencial para a extração de conhecimentos não triviais que, advindos de vastas coleções de dados, dificilmente seriam decorrentes de constatações humanas. A saída pode representar grande valia, quando obtida de maneira apropriada, podendo ainda propor modificações em protocolos e desafios impostos pela terapia convencional (KREBS et al. , 2009, 2014). Apesar da heterogeneidade na coleta de dados suscetíveis a análises (MORETTI et al. , 2013), é preciso certificar que o insumo foi adquirido adequadamente para obter resultados relevantes, sob o ponto de vista clínico, incitando melhor aceitação, bem como confiabilidade na precisão da medição de cada variável envolvida (WINTERS et al. , 2003; THEODOROS E RUSSELL , 2008).

Apesar do rico potencial para analisar o desempenho dos pacientes, os cenários de reabilitação com robôs e jogos carecem de análises relevantes (MORETTI et al. , 2013), cujos resultados, na iminência de influenciar positivamente os tratamentos, podem, por consequência, impactar variáveis, como o tempo de tratamento restante, o tempo até as primeiras evidências de pro-

gressos, ou ainda o engajamento do paciente. Em vista disso, observa-se uma lacuna, devido ao aproveitamento insuficiente deste benefício, inibindo um aperfeiçoamento ainda maior desta nova abordagem de terapia como um todo.

O que se observa acerca das análises de dados em trabalhos relacionados (BOIAN et al. , 2003; KREBS et al. , 2003; BURKE et al. , 2009) é o seu emprego de forma secundária, complementar ao objetivo principal. Ainda que este feito não denote inferioridade, no que diz respeito à relevância dos resultados obtidos, efetuar análises mais aprofundadas possibilita aperfeiçoar o que se espera como resultado. Esta possibilidade pode ser melhor assimilada ao observar casos (KREBS et al. , 2003) cujas análises são individualmente realizadas, em domínios restritos, referentes às sessões de terapia. Desta forma, parte-se do pressuposto que, ao ampliar a abrangência dos dados para análises, novas estratégias poderão ser elaboradas, considerando também a possibilidade de inserir nestas análises os resultados obtidos individualmente, de forma sequencial, ou iterativa, de acordo com o conhecimento que se deseja obter.

1.2 Objetivos

O objetivo deste trabalho consiste na elaboração de uma estratégia, a partir de um estudo de caso, para a extração de conhecimento em massas de dados concernentes à reabilitação com robôs; dados estes que descrevem o desempenho de pacientes diagnosticados com hemiparesia, na reabilitação de membros superiores. O conteúdo a ser explorado, que geralmente assume grandes proporções, expressa o desempenho de pacientes no decorrer de sessões de reabilitação com robôs. A saída, obtida a partir de uma estratégia de descoberta de conhecimento em bases de dados elaborada neste trabalho, deve culminar em um mecanismo para determinar (classificar) o lado hemiparético de acordo com o desempenho apresentado. Os critérios para tal predição devem salientar aspectos relevantes sob a perspectiva clínica, concretizando as etapas iniciais de um método, o qual pleiteia-se o seu estado final (a continuidade do estudo) em trabalhos futuros, viabilizando melhor suporte às decisões de fisioterapeutas.

1.2.1 Objetivos específicos

Para viabilizar o desenvolvimento deste trabalho, faz-se necessário enumerar requisitos específicos, descritos a seguir, a fim de atender aos objetivos principais.

1. Compreender os processos de descoberta de conhecimento em bases de dados, elegendo a

alternativa mais apropriada para aplicar sobre os dados obtidos;

2. Reunir sintomas e demais indicativos característicos da patologia dos pacientes inseridos no cenário de reabilitação robótica, dos quais os dados coletados, referentes ao desempenho, serão utilizados no presente trabalho;
3. Desenvolver estratégias, ou derivações de uma estratégia principal, destinadas à busca por características comuns entre casos cujo diagnóstico (sob o ponto de vista do experimento, limitando-se aos perfis traçados no objetivo 2) é conhecido;
4. Interpretar os resultados com base nos conhecimentos prévios, referentes ao objetivo 2, envolvendo a perícia de especialistas do campo de aplicação.

1.3 Estrutura do texto

A organização do presente trabalho está definida a seguir:

O **Capítulo 1** apresenta justificativa e motivação para o desenvolvimento deste trabalho, abrangendo o cenário atual da reabilitação com robôs e jogos sérios, bem como os anseios referentes às práticas de análises de dados, elucidando os objetivos.

O **Capítulo 2** apresenta uma revisão bibliográfica, expondo os principais trabalhos voltados à reabilitação com robôs e jogos e o estado da arte dos mecanismos e modelos de estratégias para análise de dados através de extração de conhecimento e inteligência computacional.

O **Capítulo 3**, de forma complementar à revisão bibliográfica, elucida o princípio de funcionamento dos algoritmos de aprendizado de máquina utilizados no experimento do presente trabalho.

O **Capítulo 4** apresenta um estudo de caso de pacientes reais, cujos dados de desempenho foram submetidos aos mecanismos de análise descritos no capítulo 3, elucidando a metodologia do processo de coleta à obtenção de resultados.

O **Capítulo 5** expõe os resultados obtidos na metodologia apresentada, abordando uma interpretação na forma de discussão, mencionando fronteiras com a perspectiva clínica.

O **Capítulo 6** elucida perspectivas futuras e diretrizes para a continuidade deste trabalho.

Capítulo 2

Revisão Bibliográfica

Este capítulo apresenta uma revisão bibliográfica dos campos envolvidos no desenvolvimento deste projeto, expondo os principais trabalhos voltados à reabilitação com robôs e jogos, bem como o estado da arte das pesquisas sobre mecanismos de análise de dados, envolvendo estratégias de extração de conhecimento e inteligência computacional.

2.1 Reabilitação robótica

No que diz respeito às práticas de análise de dados em cenários de reabilitação com robôs, mais especificamente aos meios de aplicação dos mecanismos de inteligência computacional, pode-se observar a existência de diferentes propostas. Assim sendo, os resultados são consumidos por diferentes partes integrantes do cenário reabilitativo, como o terapeuta, ou ainda os softwares em execução nos dispositivos robóticos. Eis a seguir exemplos destes casos, elucidando de forma mais apropriada.

Em trabalhos propostos por ANDRADE et al. (2014), CHEMUTURI et al. (2013) e COLOMBO et al. (2012) as análises dos dados são feitas para o usufruto dos próprios robôs. Estes devem interagir de forma adaptativa no decorrer das sessões, de acordo com o desempenho apresentado pelo paciente em resposta aos desafios propostos, próprios de cada estratégia e/ou algoritmo empregado nestes trabalhos.

ANDRADE et al. (2014) utilizam um dispositivo robótico de um grau de liberdade de reabilitação (passiva e ativa) de punho, o qual processa os dados coletados através do algoritmo Q-Learning, abordando a aprendizagem por reforço. Com o objetivo de combinar a dificuldade do

jogo às condições de cada paciente, a prática da reabilitação neste cenário consiste em estimular o empenho do paciente oferecendo desafios, o que requer maior esforço, bem como incentivando-o, eventualmente moderando a dificuldade do jogo.

CHEMUTURI et al. (2013) utilizaram o dispositivo HapticMaster (LINDE E LAMMERTSE, 2003), componente principal do sistema de reabilitação da GENTLE/A. Projetado para a reabilitação de membros superiores, trata-se de um dispositivo de três graus de liberdade, cuja movimentação exercida é reproduzida em um ambiente virtual. Com o objetivo de realizar movimentos de alcance (ponto-a-ponto), a adaptatividade, através do algoritmo desenvolvido no trabalho, consiste em ajustar o tempo a ser determinado pelo ambiente virtual para que o paciente conclua cada movimentação. Desta forma, em função do número de sessões de terapia, este intervalo de tempo converge para um valor ótimo, proporcional à capacidade motora do paciente.

De maneira similar à descrita por CHEMUTURI et al. (2013), COLOMBO et al. (2012) também propõem o exercício da terapia com movimentos de alcance. Este trabalho utilizou o dispositivo Braccio di Ferro (CASADIO et al., 2006), de dois graus de liberdade, para a reabilitação de ombro e cotovelo, permitindo uma movimentação em duas dimensões, sendo representada, da mesma forma, em um ambiente virtual bidimensional. Após uma fase inicial de treinamento para a avaliação do perfil do paciente, o algoritmo desenvolvido, denominado *Progressive Task Regulation* (PTR), deve propor a dificuldade mais apropriada para motivar e oferecer desafios, os quais possam ser superados pelo paciente. O objetivo do paciente é delinear figuras geométricas através das movimentações de alcance, tal que os níveis de menor e maior dificuldade são respectivamente representados por polígonos de quatro e oito vértices.

Em contraste a estas abordagens de análise de dados, é importante salientar os casos em que a coleta e o processamento dos dados são de grande utilidade aos terapeutas. Apesar da diversidade de dispositivos robóticos envolvidos nas pesquisas no campo de reabilitação, o que naturalmente implica em análises específicas, visto que o insumo (dados) coletado origina-se dos próprios dispositivos, é possível observar práticas comuns ao lidar com o que é passível de processamento. As análises individuais, por exemplo, oferecem, ao final de cada sessão de terapia, estatísticas, predições de avaliações clínicas (ZARIFFA et al., 2012), ou ainda métricas específicas (KREBS et al., 2003), referentes ao desempenho do paciente. Analisar de forma cumulativa o que foi coletado em função do tempo caracteriza outra prática comum, cujos incrementos deste todo representam os dados de desempenho de cada sessão, eventualmente

pré-processados. A fim de identificar características na desenvoltura do paciente no decorrer do tratamento, este processo pode resultar tanto da atividade computacional, sendo também o caso de ZARIFFA et al. (2012) (uma vez que um conjunto inicial de amostras seja necessário), quanto do labor humano (BOIAN et al. , 2003; KREBS et al. , 2003; SCHÖNAUER et al. , 2011), havendo o levantamento de constatações por parte dos terapeutas.

Adotando ambas as práticas, KREBS et al. (2003) desenvolvem métricas para analisar o desempenho dos pacientes submetidos à terapia com o dispositivo MIT-MANUS (HOGAN et al. , 1995; KREBS et al. , 1998), que também oferece exercícios de movimentações de alcance para a reabilitação de ombro e cotovelo. Aspectos como o ímpeto do auxílio prestado pelo robô, a distância entre a posição central de destino e a posição da mão do paciente (o *end-effector*), a magnitude do *jerk* (a terceira derivada da posição), entre outros, foram considerados neste trabalho. Avaliações no âmbito absoluto (inter-sessão) são também factíveis, pela apresentação objetiva dos dados coletados, traçando as trajetórias delineadas pelo paciente a cada sessão, as quais devem se assemelhar a um padrão “estrela”. Desta forma, o terapeuta pode avaliar o desempenho do paciente em função do tempo, uma vez que a visualização destes dados em comparação ao padrão desejado seja facilmente assimilada.

ZARIFFA et al. (2012) expõem outra perspectiva, embora também adotem ambas as práticas para a análise dos dados. Todo o conteúdo extraído do que foi coletado de cada paciente, tanto do exercício individual de cada sessão, quanto do aglomerado que descreve o desempenho em função do tempo, é obtido a partir de rotinas computacionais. Com exceção do que ocorre antes ou depois do processamento dos dados, não há atividade humana de cunho analítico. A partir dos dados de sessões que ocorreram em período próximo às avaliações clínicas, estes, juntamente dos dados clínicos, compõem o conjunto de treinamento utilizado para realizar as predições. O processo de treinamento deste modelo é composto por regressões lineares dos dados coletados, de forma a identificar predições específicas de melhor correlação com as avaliações clínicas. O modelo preditivo, ao final de uma sessão de terapia, deve prever as avaliações clínicas, apresentando resultados aproximados aos valores que seriam estabelecidos por um terapeuta. Isto se torna benéfico nos casos em que a reabilitação remota, utilizando dispositivos de baixo custo em casa, é necessária, devido a inviabilidade de locomoção ao centro de reabilitação mais próximo.

A Tabela 2.1 reúne os trabalhos aqui mencionados, entre outros trabalhos relacionados, apresentando um comparativo entre as principais abordagens para análises de dados. Os trabalhos

Tabela 2.1: Comparativo das análises de dados em trabalhos de reabilitação robótica

	Dados coletados	Análise	Análise humana	Tipo de análise
KREBS et al. (2003)	Cinemáticas, forças	Ambas	Visualização dos dados no âmbito absoluto	Quantitativa
BOIAN et al. (2003)	Cinemáticas, forças	Individual	Visualização dos dados pré-processados	Qualitativa (humana)
BURKE et al. (2009)	Vídeo	Individual	Visualização dos dados de imagens pré-processadas	Qualitativa (humana)
SCHÖNAUER et al. (2011)	Cinemáticas, EMG	Geral	Visualização dos dados pré-processados, disponível para terapeuta e paciente.	Qualitativa (humana)
COLOMBO et al. (2012)*	Cinemáticas	Geral	Não	Adaptativa
ZARIFFA et al. (2012)	Cinemáticas, forças (<i>grip</i>)	Ambas	Confecção do conjunto de treinamento	Preditiva (regressão linear)
CHEMUTURI et al. (2013)*	Cinemáticas, forças	Geral	Não	Adaptativa
ANDRADE et al. (2014)*	Cinemática angular, amplitude	Geral	Não	Adaptativa (aprendizado por reforço)
APPEL et al. (2014)	Imagens térmicas	Individual	Não	Classificação

apontados com asterísco (*) são aqueles cujas análises são destinadas aos softwares dos dispositivos robóticos apenas. Do contrário, as análises são de usufruto dos terapeutas e/ou pacientes. Elucidando a estrutura desta tabela, as colunas, da esquerda para a direita, reúnem as seguintes informações: formato dos dados coletados a cada sessão de terapia; âmbito das análises dos dados, sendo Individual o termo referente às análises relativas a cada sessão, e o termo Geral referente ao todo obtido; o labor humano, se houver, no processo de análise; o tipo de análise realizada, podendo ser um processo humano, um algoritmo próprio desenvolvido, ou uma categoria de aprendizado de máquina.

Ao se tratar de terapia física no contexto das análises de dados, humanas ou computacionais, espera-se, naturalmente, o enfoque nos dados cinemáticos, uma vez que as próprias movimentações designem a essência deste tipo de tratamento. Entretanto, pode-se observar na Tabela 2.1, diferentes perspectivas de análise (APPEL et al. , 2014; BURKE et al. , 2009), sobre dados de diferentes formatos. De caráter complementar, APPEL et al. (2014) propõe a classificação das emoções do paciente no decorrer de cada sessão de terapia, de forma a mensurar o seu engajamento ao exercício. O material coletado consiste em imagens do rosto do paciente, capturadas por câmera térmica, as quais tiveram características (*features*) extraídas, constituindo amostras de entrada de uma rede neural artificial do tipo Perceptron multicamadas. A saída deste classificador deve indicar se o paciente está neutro, sob estresse, ou motivado.

Uma solução de baixo custo para a inspeção dos movimentos de membros superiores, embora tal análise não suceda de dados puramente cinemáticos, é proposta por (BURKE et al. , 2009). Este trabalho se baseia na captura de imagens das mãos do paciente, as quais percorrem por áreas específicas a serem monitoradas. As mãos do paciente devem portar marcadores (vestindo luvas ou segurando objetos) de cores distintas entre si, bem como de qualquer outro elemento visível ao fundo. Coletando as cores referentes aos marcadores, a partir da imagem da câmera, as mãos têm suas trajetórias traçadas num espaço bidimensional, cuja detecção ocorre por meio de algoritmos de processamento de imagem.

2.2 Descoberta de conhecimento em bases de dados

Descoberta de conhecimento em bases de dados, do inglês *Knowledge Discovery in Databases* (KDD), é o processo pelo qual os dados submetidos são analisados, resultando na extração de informações novas, de alto nível, na forma de conhecimento, sob determinado domínio de dados com capacidade informativa de baixo nível. Constituído de diversas etapas, o processo maneja dados de forma sistemática, desde o estado inicialmente bruto, à obtenção de novos conhecimentos.

Historicamente, termos como *Knowledge Extraction*, *Information Discovery*, *Information Harvesting*, *Data Archeology* e *Data Pattern Processing* remetiam à busca por padrões em massas de dados, o que recentemente é conhecido por *Data Mining* (Mineração de dados) (FAYYAD et al. , 1996; CIOS et al. , 2007). Não obstante, faz-se, comumente, por este termo, uma menção equivocada ao KDD, sendo que, de fato, trata-se de somente uma etapa de todo o processo de descoberta de conhecimento. FAYYAD et al. (1996) define KDD por um processo não trivial para a identificação de padrões válidos, novos, potencialmente úteis e compreensíveis a partir dos dados.

Apesar da existência de processos (elucidados posteriormente, na seção 2.2.1) a serem seguidos, não há, do ponto de vista universal, a “melhor” opção, devido a aplicabilidade em diversos contextos, existindo assim, o mais adequado para cada caso (CIOS et al. , 2007). Grande parte do esforço para analisar os dados é direcionada à elaboração uma estratégia, sob as diretrizes de determinado processo, bem como à modelagem adequada do problema, condizente ao que se deseja obter como conhecimento.

Defronte ao emprego de recursos computacionais em contextos de medicina, engenharia, va-

rejo, marketing, e outros, o KDD torna-se aplicável, uma vez que seja possível realizar coleta e armazenamento de dados. Desta forma, de caráter interdisciplinar, a busca por conhecimentos demanda a cooperação de especialistas da área específica, obtendo, complementarmente, da experiência sob um panorama técnico, maior entendimento acerca do domínio dos dados passíveis de análise, o que contribui para a elaboração de uma estratégia precisa e melhor elaborada. No âmbito científico, ou de mercado, este tipo de trabalho em grupo é um desafio para estimular a formação de novas idéias e incentivar o trabalho multidisciplinar (HOLZINGER, 2011). O entendimento prévio dos dados é essencial, possibilitando estabelecer critérios que justifiquem a escolha de cada algoritmo envolvido no processo de descoberta de conhecimento. Do contrário, uma aplicação pouco embasada do processo de análise de dados, ou ainda sem nenhum critério *a priori*, buscando padrões de maneira subjetiva e iterativa nos dados, designa o que é depreciativamente denominado dragagem de dados (*Data dredging*) (ANDERSON E BURNHAM, 1999; FAYYAD et al., 1996). A informação extraída, decorrente de uma dragagem de dados, geralmente traz consigo um conteúdo ilegítimo, mesmo tratando-se de uma busca de cunho investigativo.

A massa de dados na sua forma bruta, evidentemente salientando o ponto comum entre os processos de KDD, a cada contexto, pode assumir diferentes formatos inicialmente. Os dados apresentam-se em aspecto numérico ou textual, geralmente armazenados em bases de dados ou em arquivos não binários; imagens; vídeos; ou ainda na forma semiestruturada, como XML (*Extensible Markup Language*), HTML (*HyperText Markup Language*), JSON (*JavaScript Object Notation*), entre outras estruturas. A saída (conhecimento extraído) geralmente é representada por padrões, regras, modelos de classificação, associações, tendências, análises estatísticas, *clusters*, e outros (CIOS et al., 2007).

O mapeamento dos dados de determinado espaço $\mathbb{R}^n$, em um espaço $\mathbb{R}^m$, tal que $m < n$, denota a definição de visualização de dados (Figura 2.1), de acordo com HOLZINGER et al. (2014). Do ponto de vista pragmático, a visualização de dados representa grande utilidade para a análise, sendo esta utilizável antes, durante, e após a submissão de dados a um processo de KDD.

Inspecionar humanamente o conjunto de dados no seu estado inicial e/ou após pré-processado, possibilita a percepção de eventuais características que estejam explicitamente dispostas no espaço, auxiliando na detecção de *outliers* (vide seção 2.2.2) para descarte, bem como na escolha dos mecanismos mais apropriados, sendo necessário ponderar o que se espera como saída (de

Figura 2.1: Importância da visualização de dados em duas dimensões (THE HOLZINGER GROUP)

cada etapa e de todo o processo, tendo em vista o formato e o conteúdo da informação). A visualização durante o processo de KDD ocorre após a conclusão de uma etapa, como a mineração de dados, por exemplo, permitindo verificar posições de fronteiras de separabilidade, a abrangência de *clusters*, inclinação de retas de regressões lineares, e assim por diante. Após o KDD, a visualização ocorre de maneira similar, porém com o objetivo de simplificar e tornar o conteúdo compreensível ao usuário final. Uma ressalva importante posta em HOLZINGER et al. (2014) e em BARTKE (2005) destaca a dificuldade de humanos em compreender a distribuição de dados num espaço com mais de duas dimensões, sendo a disposição bidimensional ($\mathbb{R}^2$) a mais adequada. Técnicas de visualização de dados podem ser vistas nos trabalhos de BARTKE (2005); WARE (2004); FAYYAD et al. (2002).

2.2.1 Processos de KDD

Antes de prosseguir para os processos de KDD, a fim elucidativo, é importante salientar o significado de terminologias como Processo (ou Modelo de Processo) e Metodologia, as quais tendem a ser erroneamente atribuídas a um mesmo significado (MARBÁN et al. , 2009). A descrição destes termos, a seguir, foram baseadas em PRESSMAN (2005) e MARBÁN et al. (2009).

Modelo de Processo, ou simplesmente Processo, refere-se ao conjunto de etapas a serem seguidas, visando o desenvolvimento de uma saída, um produto, decorrente de uma sequência de tarefas no âmbito conceitual. Proveniente de um modelo de processo, metodologia faz menção a determinada instância de um processo, ao que há de concreto e factível, com base no que era inicialmente abstrato. A execução sucessiva de técnicas e algoritmos, encadeada na forma de fluxo, descreve uma metodologia.

Voltada à indústria, ou ao meio acadêmico, a evolução do KDD e da mineração de dados nos últimos anos fomentou o surgimento de outros modelos de processo (MARBÁN et al. , 2009). Sob o prisma de GOLDSCHMIDT E PASSOS (2005), pode-se observar que a descoberta de conhecimento, de maneira mais abrangente, consiste em três etapas: pré-processamento, mineração de dados e pós-processamento. O pré-processamento deve circundar o que é preliminar em relação à saída desejada, desde o acesso à coleção de dados (ou a coleta, quando possível), até a organização dos dados, por meio de etapas como seleções de subconjuntos, reduções de dimensionalidade ou ajustes finos. A mineração de dados, sucessivamente, refere-se à atividade dos algoritmos de aprendizado de máquina, obtendo classificações de padrões, regras, modelos, entre outros formatos de saída. Por fim, o pós-processamento é composto das etapas de cunho avaliativo, havendo inspeções humanas para verificar a qualidade do conhecimento extraído em relação à pergunta inicialmente feita (a modelagem do problema), para que o resultado final possa ser de usufruto dos usuários finais. Há também a possibilidade de, em alguns casos, reiterar todo o processo, visando aperfeiçoar a acurácia da informação.

O primeiro processo a ser apresentado, de FAYYAD et al. (1996), é interativo e iterativo, envolvendo nove etapas que exigem tomadas de decisão na escolha de técnicas e algoritmos, para a construção de uma metodologia. Dentre os diversos modelos de processo disponíveis na literatura acadêmica, este foi escolhido para ser explanado, por ser visto como o principal modelo (CIOS et al. , 2007). No decorrer das nove etapas, os dados seguem um fluxo, ilustrado na Figura 2.2, indicando o estado dos dados ao longo do processo.

Figura 2.2: Fluxo de dados do processo de Fayyad et al. Adaptado de FAYYAD et al. (1996).

A primeira etapa consiste no entendimento do domínio dos dados, bem como na obtenção de conhecimento prévio para as etapas seguintes. Ainda nesta etapa, deve-se identificar e definir

as metas do processo com base no panorama do usuário final.

Em seguida, na segunda etapa, define-se um conjunto-alvo de dados. A partir do insumo integralmente disponível em uma base de dados, ou em outra estrutura, é preciso definir um subconjunto que contenha apenas os dados concernentes às metas definidas anteriormente, desprezando assim o restante. Outra opção é formar uma coleção de variáveis ou amostras. O conjunto-alvo formado é então submetido à etapa seguinte.

A limpeza e o pré-processamento dos dados ocorrem na terceira etapa. O conjunto-alvo obtido deve ser refinado, eliminando ruídos, bem como ser melhor estruturado, lidando com lacunas eventualmente presentes em determinada tupla de uma tabela, por exemplo. A preservação do comportamento dos dados, quando estão em função do tempo, é imprescindível. Ajustes finos também são realizados nesta etapa, como a conversão dos dados e/ou da própria estrutura, de acordo com o formato de entrada de algoritmos posteriores.

Na quarta etapa, caso necessário, os dados são submetidos a reduções de dimensionalidade e projeções. Caso o conjunto de dados assuma grandes proporções, estes métodos devem diminuir o número de variáveis, eliminando também o teor invariante dos dados. A aplicação desta etapa deve resultar em um conjunto mais reduzido e sólido.

A quinta etapa é baseada na combinação das metas de KDD definidas na primeira etapa com uma técnica de mineração de dados, como sumarização, classificação, *clustering*, regressão, e outros. É considerável adotar como um dos critérios a verificação do formato de entradas e saídas destes mecanismos, tendo estas de serem equivalentes ao formato do que se deseja obter como conhecimento.

Após definir a técnica para a mineração de dados, um algoritmo, na sexta etapa, deve ser selecionado. Nesta etapa é feita uma análise exploratória dos algoritmos aplicáveis a este contexto, visando testar parâmetros, arquiteturas e outras configurações de cada algoritmo, possibilitando então optar pela alternativa mais adequada.

Na sétima etapa a mineração de dados é realizada. Os algoritmos selecionados devem processar os dados e resultar em uma saída representativa na forma de *clusters*, modelos de predição, árvores ou regras de classificação. Nestas saídas faz-se uma busca efetiva por padrões e características. É importante ressaltar que o elo entre as etapas de todo o processo designa uma cadeia. Portanto, as etapas anteriores devem ser executadas de maneira cautelosa, pois falhas humanas em determinada etapa podem afetar o resultado dos passos subsequentes, ou ainda comprometer a qualidade da saída do próprio processo.

A interpretação do que foi adquirido com a mineração de dados ocorre na oitava etapa. Geralmente são utilizados métodos de visualização de dados, oferecendo uma maneira mais objetiva para inspecionar o conhecimento obtido e ponderar a viabilidade de realizar iterações do processo, retornando a uma das etapas anteriores.

Por último, a nona etapa consiste em consolidar o conhecimento, usufruindo-o diretamente, ou incorporando-o em determinada aplicação, a critério do usuário final, para ações futuras. O conhecimento obtido pode também ser documentado e utilizado em processos maiores. Também é feita nesta etapa uma verificação e resolução, se houver, de conflitos entre conhecimentos anteriormente extraídos ou definidos hipoteticamente.

Apesar da essência da extração de conhecimento ser característica da mineração de dados, as demais etapas são, da mesma forma, indispensáveis, em virtude da ordem adequada em que as quais se dispõem. Isto, na prática, permite refinar o insumo que posteriormente é apresentado aos mecanismos de aprendizado de máquina na etapa de mineração de dados.

O CRISP-DM (*CRoss-Industry Standard Process for Data Mining*), outro processo aqui descrito, se tornou o principal modelo na indústria (CIOS et al. , 2007), por ser aplicado com veemência neste contexto. Este modelo de processo foi inicialmente elaborado por quatro empresas: Integral Solutions Ltd., provedora de soluções em mineração de dados; NCR, provedora de banco de dados; DaimlerChrysler, fabricante de automóveis; OHRA, companhia de seguros. As duas últimas companhias participaram como provedoras de dados (CIOS et al. , 2007; CHAPMAN et al. , 2000).

Este modelo de processo é composto de seis etapas, ilustradas na Figura 2.3, não havendo uma sequência estrita de passos. As etapas deste processo foram descritas com base em CHAPMAN et al. (2000).

O entendimento do negócio (*Business Understanding*), na primeira etapa, tem como objetivo compreender os requisitos e objetivos do projeto, sob uma perspectiva de negócios. A partir deste embasamento, deve-se definir o problema e as metas da mineração de dados, bem como elaborar um plano preliminar para alcançar os objetivos.

A segunda etapa envolve o entendimento dos dados (*Data Understanding*). Faz-se uma coleta inicial do que será processado, possibilitando então a verificação de problemas na qualidade dos dados. Ainda nesta etapa, a familiarização deste subconjunto, de maneira exploratória, estimula a formação de idéias para a elaboração de possíveis hipóteses acerca de informações implícitas e passíveis de descoberta.

Figura 2.3: Processo de KDD CRISP-DM. Adaptado de CHAPMAN et al. (2000)

Na terceira etapa, a preparação dos dados (*Data Preparation*) abrange as atividades referentes ao pré-processamento (seleção, filtragem, etc.) e as demais necessárias para a confecção do conjunto de dados final, devendo este estar na forma adequada para alimentar a entrada de ferramentas de modelagem na próxima etapa. A iteratividade entre técnicas para a preparação dos dados é plausível, não havendo uma ordem estática de passos a serem seguidos.

A modelagem (*Modeling*) ocorre na quarta etapa. Diversas técnicas devem ser selecionadas e aplicadas sobre os dados, repetidamente, de forma a ajustar os parâmetros específicos de cada algoritmo em direção às configurações ótimas. Tipicamente, para o mesmo tipo de problema de mineração de dados, existem várias técnicas diferentes. Entretanto, as técnicas podem exigir diferentes formatos de dados, o que pode influenciar na escolha de um algoritmo, ou ainda sendo necessário retornar à etapa anterior.

Após a obtenção do(s) modelo(s), sua avaliação (*Evaluation*) é realizada na quinta etapa. Primeiramente, é importante verificar a execução dos passos anteriores, pois, assim como no processo de FAYYAD et al. (1996), as etapas de extração e inspeção humana estão encadeadas, o que implica na propagabilidade de eventuais falhas. Deve-se então certificar que o modelo atende às metas e objetivos, determinando se as questões inicialmente estabelecidas foram ou não sanadas. Ao final desta etapa, uma decisão a respeito do uso dos dados extraídos deve ser tomada, havendo a possibilidade de iterar o processo.

A implantação (*Deployment*) é feita na etapa final, atribuindo a devida utilidade ao conhecimento extraído. Apesar do propósito de processos de KDD ser a extração do conhecimento, é importante que haja um aproveitamento efetivo da saída obtida, de modo a aplicá-la no contexto de uma pesquisa ou projeto. Neste caso, sendo voltado à indústria, os usuários finais (clientes) usufruem destes dados na forma explícita, de relatórios ou planos de implementação para tomadas de decisões em negócios. Em muitos casos, os próprios clientes realizam a implantação, ao invés do analista de dados, do qual houve o labor nas etapas anteriores.

Analisando novamente a Figura 2.3, podemos observar a ausência do termo Mineração de Dados. Isto se deve à diferente conotação atribuída a este termo, não sendo referente a uma etapa, mas sim ao processo todo, conforme descrito no próprio nome do modelo.

Na forma híbrida, um compêndio dos processos de ambas as esferas acadêmica e industrial constitui o modelo proposto por CIOS et al. (2000). Organizado em seis etapas (Figura 2.4), este modelo foi baseado no CRISP-DM, porém preservando a noção de descoberta de conhecimento, a qual não é vigorosamente característica de tal processo. Em termos práticos, a mineração de dados é uma etapa do processo, uma parte, e não uma forma de designar o todo. As seis etapas expostas a seguir foram explanadas com base em CIOS et al. (2007), CIOS E KURGAN (2005) e CIOS et al. (2000).

Figura 2.4: Processo de KDD híbrido. Adaptado de CIOS E KURGAN (2005)

O entendimento do domínio do problema (*Understanding the domain problem*), na primeira

etapa, envolve o auxílio de especialistas da área específica para a definição do problema, bem como das metas específicas e as partes interessadas. Acerca das possíveis soluções, faz-se um estudo do domínio dos dados disponíveis, envolvendo inclusive o aprendizado de terminologias específicas por parte do analista de dados. A partir de uma descrição elaborada do problema, as metas definidas devem ser convertidas para metas de mineração de dados, havendo orientações iniciais a respeito da escolha dos métodos a serem utilizados.

Em seguida, o entendimento dos dados (*Understanding the data*), envolve atividades exploratórias, visando conhecer os dados disponíveis, bem como decidir quais destes serão necessários, em termos de conteúdo, formato e tamanho, para suprir as metas inicialmente definidas; geralmente neste ponto o conhecimento prévio obtido na primeira etapa é relevante. Após buscas por características como redundâncias, lacunas e integridade nos dados, verifica-se o potencial de uso destes dados em relação às metas de mineração de dados.

Os dados, na terceira etapa, são então preparados (*Preparation of the data*), decidindo o que será efetivamente submetido aos métodos de mineração de dados na etapa seguinte. A adequação dos dados para o formato de entrada desejado envolve atividades de pré-processamento, tais como correção de ruídos e lacunas. Correlações, testes de significância, entre outras operações também são efetuadas sobre os dados. Quando necessário, aplica-se técnicas como extração de características (*feature extraction*), redução de dimensionalidade, discretização ou granularização de dados. Ao fim desta etapa, os dados devem apresentar o mesmo formato da entrada do algoritmo utilizado para a mineração de dados.

A mineração de dados (*Data mining*), na quarta etapa, refere-se à aplicação de métodos de aprendizado de máquina, resultando na extração de conhecimento para a interpretação na etapa seguinte.

Apoiadas nas saídas extraídas anteriormente, as avaliações (*Evaluation of the discovered knowledge*) são aplicadas na quinta etapa. O entendimento dos resultados envolve a verificação de vigência, valia e impacto do que foi extraído, por parte de especialistas, bem como de quem conduz tecnicamente o processo. As práticas das etapas anteriores são revisadas, em busca de ações alternativas com o potencial de coadjuvar um resultado mais aprimorado.

Na sexta etapa, decide-se a aplicação do conhecimento obtido (*Using the discovered knowledge*). Um plano para a implementação do conhecimento deve ser elaborado e documentado no projeto envolvido.

Conforme ilustrado na Figura 2.4 por linhas tracejadas, as iterações representam *feedbacks*

Tabela 2.2: Comparativo de processos de KDD. Adaptado de KURGAN E MUSILEK (2006) e CIOS et al. (2007).

Modelo	Fayyad et al.	Cabena et al.	Anand & Buchner	CRISP-DM	Cios et al.	Modelo Genérico
Área	Acadêmica	Industrial	Acadêmica	Industrial	Híbrido (Acadêmica/Industrial)	–
No. de etapas	9	5	8	6	6	6
Refs.	(FAYYAD et al. , 1996b)	(CABENA et al. , 1998)	(ANAND E BUCHNER , 1998)	(SHEARER , 2000)	(CIOS et al. , 2000)	(KURGAN E MUSILEK , 2006)
Etapas	1. Desenvolvimento e entendimento do domínio de aplicação	1. Determinar objetivos de negócio	1. Identificação de recursos humanos 2. Especificação do problema	1. Entendimento do negócio	1. Entender o domínio do problema	1. Entendimento do domínio de aplicação
	2. Criação do conjunto-alvo de dados	2. Preparação de dados	3. Prospeção de dados 4. Levantamento de conhecimento do domínio	2. Entendimento dos dados	2. Entendendo os dados	2. Entendimento dos dados
	3. Filtragem e pré-processamento dos dados		5. Identificação da metodologia	3. Preparação de dados	3. Preparação dos dados	3. Preparação dos dados e identificação de tecnologias de mineração de dados
	4. Redução e projeção de dados		6. Pré-processamento dos dados			
	5. Escolha da técnica de mineração de dados					
	6. Escolha do algoritmo de mineração de dados					
	7. Mineração de dados	3. Mineração de dados	7. Descoberta de padrões	4. Modelagem	4. Mineração de dados	4. Mineração de dados
	8. Interpretação dos padrões minerados	4. Levantamento de conhecimentos do domínio	8. Pós-processamento do conhecimento	5. Avaliação	5. Avaliação do conhecimento descoberto	5. Avaliação
	9. Consolidação do conhecimento descoberto	5. Assimilação do conhecimento		6. Implantação	6. Uso do conhecimento descoberto	6. Consolidação e implantação do conhecimento

para etapas específicas durante o processo e, para cada caso, existem questões importantes a serem ponderadas, as quais são descritas em CIOS E KURGAN (2005).

Embora os três processos aqui descritos sejam voltados a diferentes esferas de aplicação, estes tornam-se semelhantes entre si, no que diz respeito à essência do processamento dos dados (ainda na forma conceitual), quando ofuscadas discrepâncias, como o número de etapas devido a generalização/isolamento de atividades e a possibilidade de iteração entre etapas. O comparativo da Tabela 2.2, de KURGAN E MUSILEK (2006), alinha os principais modelos de processo de KDD, incluindo aqueles aqui mencionados, dispostos de forma a evidenciar semelhanças e discrepâncias, características de cada escopo para o qual foram projetados. Analisando este comparativo obtém-se uma percepção mais clara do grau de semelhança entre os diferentes modelos de processo, o qual indica homogenia nos casos de CABENA et al. (1998), CIOS et al. (2000), SHEARER (2000) e KURGAN E MUSILEK (2006), e em menor ímpeto nos casos de FAYYAD et al. (1996) e ANAND E BUCHNER (1998).

Em termos de tempo e esforço, é necessário o emprego de determinado empenho nas etapas de um processo de KDD, tendo em vista que a substancialização de atividades abstratas não se trata de uma prática trivial. Assimilando os modelos de processo da Tabela 2.2, sob o prisma do modelo genérico, KURGAN E MUSILEK (2006) reuniu estimativas de diferentes trabalhos (CABENA et al. , 1998; SHEARER , 2000; CIOS E KURGAN , 2005), demonstradas na Figura 2.5, viabilizando a comparação dos esforços atribuídos a cada etapa dos processos envolvidos.

Figura 2.5: Empenho aplicado às etapas de KDD. Adaptado de KURGAN E MUSILEK (2006)

Pode-se observar na Figura 2.5 que a soma dos valores indicados pelas barras não totaliza 100%. Isto ocorre, visto que os dados exibidos representam estimativas, as quais, na íntegra, com exceção de CABENA et al. (1998), correspondem às faixas de valores indicadas pelas

linhas verticais sobre as barras. Mesmo havendo subjetividade por parte dos diferentes autores na estipulação destas estimativas, é evidente o apontamento unânime da etapa de preparação dos dados como a atividade de maior labor em relação às demais. Assim sendo, a conclusão desta etapa deve demandar do analista aspectos como perícia e cautela, uma vez que a qualidade dos dados influencia na qualidade da informação.

Trazendo consigo uma série de técnicas distintas, desde rudimentares filtragens de ruídos, a projeções e reduções de dimensionalidade, a preparação dos dados, na sua completa abordagem, atrai potenciais desafios intrínsecos de cada uma destas práticas. A correção apropriada de erros no domínio dos dados, filtragem de dados em ambientes integrados, e a manutenção de dados filtrados são exemplos de empecilhos eventualmente aparentes (MÜLLER E FREYTAG, 2003) que devem ser superados, evitando o exercício de práticas incorretas. Falhas estas ocasionam danos aos dados e, por consequência, distorcem as configurações dos padrões passíveis de extração.

Além da importância dos resultados obtidos em si, a documentação e conclusão adequada da preparação dos dados, e das demais etapas de KDD, faz da estratégia utilizada um trabalho potencialmente reproduzível. De grande utilidade no meio acadêmico, a reprodutibilidade é a supressão de quaisquer ambiguidades e omissões de informações das atividades realizadas, permitindo a verificação e reprodução dos mesmos resultados por terceiros, o que raramente ocorre no cenário das comunidades de pesquisa modernas (HOLZINGER et al., 2014). Em contraste, a prática negligente das etapas de KDD pode comprometer além do que se espera como resultado, havendo a possibilidade da replicação exponencial deste tipo de trabalho, gerando uma cadeia de resultados de baixa qualidade (PASTRELLO et al., 2014; HOLZINGER et al., 2014).

No entanto, vale salientar que reprodutibilidade refere-se ao potencial do que é reproduzível, não havendo nenhum tipo de critério absoluto para definir o que é característico ou não deste atributo. Pondera-se, portanto, os fatores envolvidos em cada caso, como o acesso ao mesmo conjunto de dados utilizado no experimento original; acesso a materiais equivalentes para a reprodução do experimento; uso de algoritmos que envolvem aleatoriedade em seu estado inicial, como as Redes Neurais Artificiais, por exemplo, provendo resultados de mesma qualidade, apesar de não idênticos; entre outras peculiaridades. Logo, ao analisar uma estratégia de KDD, o grau de reprodutibilidade deve ser estabelecido subjetivamente.

Uma questão essencial, concernente à reprodutibilidade, mas principalmente voltada à qualidade do conhecimento extraído, é a avaliação e a interpretação dos resultados. Desde o início de

uma estratégia de KDD, deve-se assegurar que os dados utilizados envolvam o cerne daquilo que se deseja obter como conhecimento, constituindo a ideia de extração. De maneira análoga, a relação dados/conhecimento pode ser expressa em termos de matéria-prima/produto. Em grande parte das aplicações, os resultados da mineração de dados devem ser expostos da forma mais clara e objetiva possível para humanos. Para este fim, destacam-se meios como estruturação de regras, geração de linguagem natural, representações gráficas e técnicas de visualização de dados (FAYYAD et al. , 1996).

O apoio de especialistas da área nesta fase também se torna indispensável, visto que o estado dos dados processados se aproxima do que será efetivamente de usufruto do usuário final (podendo este ser o próprio especialista). Após definir o formato final do conhecimento extraído, tanto o analista, quanto o especialista da área, devem ponderar a qualidade da informação adquirida, cogitando a possibilidade de readequar a estratégia, caso o conhecimento não seja compatível com os objetivos inicialmente estabelecidos.

2.2.2 Mineração de dados

Reiterando as descrições anteriormente sintetizadas desta etapa, trata-se da aplicação de técnicas para extrair características dos dados através de mecanismos computacionais inteligentes. A saída destes mecanismos, devidamente interpretada, deve oferecer grande potencial para descrever informações inicialmente implícitas no conjunto de dados da entrada. Além da elucidação acerca de técnicas e mecanismos comumente utilizados, esta seção envolve questões que circundam estas práticas, recorrentes no exercício da mineração de dados.

No que diz respeito aos dados, o presente trabalho versa, até este ponto, a respeito de ajustes preliminares, visando alimentar adequadamente a entrada de mecanismos computacionais para a extração de informações. Complementarmente a isto, a estrutura e o teor informativo do conjunto de dados disponível, aparentes desde o seu estado inicial, apresentam características satisfatórias para decidir a abordagem que será utilizada. Algumas das técnicas de mineração de dados mais utilizadas, ilustradas pela Figura 2.6 e sucintamente retratadas a seguir, estão subdivididas em dois principais paradigmas: aprendizado supervisionado e não-supervisionado. A mescla de técnicas de ambos os paradigmas implica ainda na origem de um terceiro, chamado de aprendizado semi-supervisionado (ZHU E GOLDBERG , 2009). Mesmo que não concernente à forma cíclica e finita do KDD, pela sua execução contínua, vale mencionar a existência do aprendizado por reforço (KAELBLING et al. , 1996), apesar da presente revisão ater-se aos

paradigmas principais.

Quando o conjunto de entrada é composto por porções de dados (amostras), geralmente de mesma dimensão e formato, bem como retém consigo uma coleção de saídas desejadas em relação às respectivas amostras, aplica-se técnicas de aprendizado supervisionado. Quando não há junto das amostras o conjunto das saídas desejadas, técnicas de aprendizado não-supervisionado devem ser aplicadas.

(a) Classificação

(b) Regras e árvore de decisão

(c) Regressão

(d) Agrupamento (*Clustering*)

Figura 2.6: Técnicas de aprendizado de máquina para mineração de dados.

Classificação (Figura 2.6(a)) consiste no mapeamento do espaço referente ao domínio do problema, seccionando a disposição das amostras em subespaços, os quais são relacionados a diferentes categorias (classes). Sendo uma técnica de aprendizado supervisionado, o número de classes envolvidas é definido de acordo com o conjunto das saídas desejadas. As demarcações dos subespaços referentes às classes, chamadas de fronteiras de separabilidade (*decision boundaries*), apresentam, de acordo com o algoritmo utilizado, diferentes formas visuais, como retas ou elipses, assumindo estas o mesmo número de dimensões do espaço em que se situam. As retas, por exemplo, próprias do espaço bidimensional, passam a ser planos no espaço tridimensional, e hiperplanos em espaços de maiores dimensões. Os algoritmos *Perceptron* (ROSENBLATT, 1958), *Perceptron* Multicamadas (PMC) (RUMELHART et al., 1986), *k-Nearest Neighbors* (kNN) (FIX E HODGES, 1951) e *Support Vector Machines* (SVM) (VAPNIK, 1995) são exemplos destinados à classificação.

Similar à classificação, e também concernente ao aprendizado supervisionado, a definição de regras com árvores de decisão (Figura 2.6(b)) possibilita também particionar o espaço amostral. Esta técnica, no entanto, oferece resultados facilmente compreensíveis por humanos, não sendo necessariamente a visualização de dados o meio para avaliar a saída. De acordo com as regras obtidas, dispostas em árvore, um dado padrão percorre esta estrutura conforme suas características, em direção às folhas, representando as classes. ID3 (QUINLAN, 1986) e C4.5 (QUINLAN, 1993) são exemplos de algoritmos desta técnica.

Ilustrada na Figura 2.6(c), a regressão refere-se à representação contínua do comportamento de uma variável dependente, discretizada, que está em função de uma variável independente. O modelo resultante da análise de uma série temporal, por exemplo, possibilita a aquisição de valores em qualquer ponto do intervalo que, a princípio, era parcialmente conhecido; assim como, de caráter preditivo, é possível conhecer à frente do que é descrito no domínio apresentado à regressão. Uma das maneiras para descrever dados discretos é através da regressão linear, resultando em uma equação de reta, da qual pode-se usufruir não somente da continuidade, mas também do coeficiente angular, indicando a ascendência/descendência do comportamento dos dados, quando esta não é visualmente evidente. Regressões não-lineares são utilizadas na representação de variáveis não lineares, realizando aproximações funcionais por meio de mecanismos como *Multilayer Perceptron* e o método dos mínimos quadrados.

O agrupamento (*Clustering*), concernente ao aprendizado não supervisionado e representado na Figura 2.6(d), consiste na concentração de padrões com características semelhantes, cujo

número de grupos (*Clusters*), quando não conhecido previamente, é arbitrário. Para este fim, faz-se uso de algoritmos como *k-means*, mapa auto-organizável de Kohonen (KOHONEN , 1982, 1984), DAMICORE (SANCHES et al. , 2011), entre outros. Uma separação adequada de padrões pode ainda ser reproduzida em mecanismos de aprendizado supervisionado, fazendo dos *clusters* as classes envolvidas, possibilitando complementar um processo de análise de dados, ou simplesmente obter resultados de diferentes algoritmos para comparações. O emprego do aprendizado supervisionado sobre o resultado de técnicas de aprendizado não supervisionado, como o agrupamento, remete à noção do aprendizado semi-supervisionado.

As escolhas de técnica e algoritmo para realizar a mineração de dados, atividades parcialmente concernentes a uma etapa, ou ainda próprias de etapas inteiras, evidenciam a importância desta decisão. Aspectos como o formato da fronteira de separabilidade de determinada técnica, eficácia de um algoritmo ao lidar com elevadas quantidades de dados na entrada e/ou na saída, potencial para processar dados qualitativos e/ou quantitativos, entre outros, devem ser confrontados com a disposição dos dados a serem analisados, optando pela alternativa mais condizente ao problema em questão.

Apesar de concernir às etapas anteriores de KDD, uma questão estritamente vinculada à mineração de dados, que influencia nas escolhas de técnica e algoritmo apropriados, é o manejo de conjuntos de dados que apresentam alto número de dimensões, aspecto que é conhecido por *curse of dimensionality* (CATCHPOOLE et al. , 2010). Ainda que haja mecanismos de aprendizado de máquina eficazes no processamento de muitas entradas, deve-se aplicar técnicas de projeção e redução de dimensionalidade, tais como *Principal Component Analysis* (PCA) (SMITH , 2002) e *Linear Discriminant Analysis* (LDA) (FISHER , 1938). Próprias da etapa de preparação/transformação, estas técnicas destinam-se à redução/remoção do teor invariante dos dados, possibilitando submeter uma entrada adequada ao mecanismo escolhido para mineração de dados.

Feitas as escolhas, é necessário enumerar alguns pontos que requerem a atenção do analista de dados, pois quando negligentemente desprezados, tornam-se empecilhos para o processo de análise como um todo. Assim sendo, expõe-se a seguir os seguintes temas: *overfitting/underfitting*, validação cruzada (*cross-validation*) e *outliers*.

A imperícia no ajuste excessivo de parâmetros de um algoritmo, com o objetivo de potencializar os resultados, implica na memorização excessiva dos padrões, podendo comprometer a capacidade de generalização (Figura 2.7(b)) e ocasionar o que é chamado de *overfitting* (Figura

2.7(c)). SILVA et al. (2010) ressaltam um exemplo deste caso em uma arquitetura de rede neural artificial, cuja configuração topológica é composta por uma quantidade elevada de neurônios em uma ou mais camadas: durante a etapa de treinamento, o erro a ser minimizado tende a ser baixo; em contrapartida, na etapa de testes, utilizando um conjunto diferente de amostras, o erro passa a ser alto, em virtude do superajuste em relação às amostras de treinamento. Vale mencionar o outro caso extremo, no sentido oposto do *overfitting*, quando o ajuste dos parâmetros envolvidos é moderado, ocasionando o *underfitting* (Figura 2.7(a)). Neste caso, de acordo com SILVA et al. (2010), o erro é alto em ambas as etapas de treinamento e testes.

Figura 2.7: Situações de *underfitting* e *overfitting*

Geralmente a involuntária indução do *overfitting* pode propiciar a interferência de eventuais ruídos dos dados, memorizando o que não é pertinente ao problema, prejudicando ainda mais o que já é tido como resultado inexpressivo (FAYYAD et al. , 1996). Entretanto, a indução voluntária deste tipo de circunstância mostrou-se útil em (MORETTI et al. , 2014), proporcionando analisar, nas amostras, o impacto da ausência de cada um dos atributos que as constituem, sendo estes ocultados seguindo uma abordagem inspirada na técnica *jackknife* (MILLER , 2006).

Possíveis soluções a respeito da mitigação do risco de *overfitting*, baseiam-se no emprego de técnicas como a validação cruzada (*cross-validation*) (KOHAVI , 1995), regularização, ou outras estratégias estatísticas mais sofisticadas (FAYYAD et al. , 1996). Dividindo o conjunto total de dados em duas partes (treinamento e testes), existem três técnicas de validação cruzada para este fim: amostragem aleatória, k-partições e validação cruzada por unidade, sendo que as duas primeiras são mais usuais (SILVA et al. , 2010).

Random subsampling cross-validation, ou validação cruzada por amostragem aleatória, consiste na alocação aleatória de amostras para o conjunto de teste, cabendo ao analista decidir uma porcentagem do todo, referente ao tamanho deste subconjunto, sendo que as demais amostras são utilizadas para a etapa de treinamento. Faz-se, então, diversos ensaios, utilizando conjuntos aleatoriamente definidos a cada iteração deste procedimento. Havendo diferentes topologias

do algoritmo envolvido, elege-se aquela de maior média dos resultados dos ensaios individuais realizados.

A validação cruzada por k -amostras (*k-fold cross-validation*) representa a segmentação do conjunto de amostras em k partições, utilizando uma destas para testes, e as $k - 1$ restantes para treinamento. Composto por k ensaios para cada topologia, este procedimento faz uso de todas as partições para testes, cuja média destes resultados é também própria do critério de seleção da melhor configuração topológica. Seja n o total de amostras, um caso específico desta técnica ($k = n$) fundamenta a validação cruzada por unidade (*leave-one-out cross-validation*), utilizando apenas uma amostra a cada ensaio para testes. SILVA et al. (2010) ressalta a inassiduidade desta terceira técnica, em virtude do alto custo computacional das n iterações do processo de aprendizado para cada topologia candidata.

Ao aplicar a validação cruzada, ou outro método que lide com amostras, deve-se partir do pressuposto de que estas constituem um conjunto adequado para a obtenção de um modelo suficientemente eficaz, consoante aos objetivos inicialmente estabelecidos. Em termos práticos, uma amostra deve carregar consigo somente a essência do padrão, assim como as demais integrantes do conjunto, sejam estas inerentes ao mesmo *cluster*/classe ou não; aquela que não condiz com este quesito pode ser considerada um *outlier*. HAWKINS (1980) define *outlier* como uma observação impetuosamente divergente das demais, despertando suspeitas de que esta tenha sido gerada por um mecanismo diferente. O surgimento destas amostras atípicas é geralmente decorrente de disfunções no processo de coleta, seja por um comportamento inesperado de uma rotina computacional; avarias em determinado aparato, afetando a capacidade sensorial e a aquisição de dados; negligências operacionais, quando se envolve a ação de humanos; ou qualquer atividade destoante do curso do processo de coleta estabelecido.

Capítulo 3

Aprendizado de Máquina e Mineração de Dados

Complementarmente à revisão bibliográfica, este capítulo apresenta de modo sucinto uma revisão dos mecanismos de aprendizado de máquina envolvidos na metodologia apresentada no capítulo 4, sendo esta precedida de uma breve discussão, a fim de elucidar terminologias que foram utilizadas anteriormente e que concernem ao conteúdo a seguir.

Sem prévias definições, o termo *aprendizado* foi mencionado ao longo deste trabalho, assumindo acepções distintas em determinadas ocorrências. Estas ocorrências comumente remetem ao potencial de inteligência, divergindo, no entanto, no agente (humano ou computador) que retém o ganho do aprendizado em si. Não se destina aqui discutir de forma filosófica o que essencialmente define aprendizado; é sensato, no entanto, salientar a conotação mais compatível com o contexto de aprendizado de máquina e mineração de dados.

Com base em dicionário, WITTEN et al. (2011) reuniram cinco definições de aprendizado, as que a seguir se enumeram:

1. Adquirir conhecimento através de estudo, experiência, ou ser ensinado;
2. Tomar consciência através de informações ou observação;
3. Consolidar à memória;
4. Ser informado de, averiguar;
5. Receber instrução.

Pode-se afirmar, sob o contexto deste capítulo, que as definições 3 e 5 são as mais coerentes, uma vez que computadores sejam considerados agentes passivos. Estes, assim sendo, carecem de um estado de consciência, o que os faz inaptos a assimilar (não confundir com reter ou armazenar) informações, ainda que ações externas a um agente façam do aprendizado uma atividade constante; desta forma, as definições 1, 2 e 4 tornam-se insustentáveis.

Proveniente de mecanismos computacionais e metodologias bioinspiradas, esta interpretação peculiar de aprendizado remete à capacidade de receber instruções lógicas e armazená-las em memória, em prol de otimizações em seus comportamentos em relação a estados anteriores, a qual pode ser considerada como um atributo de inteligência computacional.

Considerada como uma fronteira entre campos da estatística e da computação (MITCHELL, 2006), o aprendizado de máquina pode ser definido como um ramo da inteligência artificial que aborda o aprendizado sob o prisma da inteligência computacional, para o estudo de padrões e comportamento em dados, bem como meios para extração e reconhecimento. A mineração de dados consiste na aplicação de ferramentas de aprendizado de máquina no processo de extração de padrões ou conhecimento, a partir de um conjunto de dados.

Dependendo do mecanismo envolvido no processo de mineração de dados, é importante ressaltar que a saída obtida pode ser compreensível por humanos, possibilitando analisar o que foi aprendido (o conhecimento) por determinado mecanismo na forma de regras encadeadas (*e.g.*: árvore de decisão), por exemplo. Este tipo de análise pode representar grande valia ao integrar um processo de KDD, além da abordagem trivial de predição de classes ou aproximações funcionais em dados não envolvidos em etapas de treinamento.

3.1 *k-Nearest Neighbors*

Desenvolvido por FIX E HODGES (1951) e posteriormente estudado por COVER E HART (1967), o *k-Nearest Neighbors* (*k*NN) utiliza a abordagem *lazy learning* para a classificação de padrões. Esta abordagem, diferentemente dos demais mecanismos aqui descritos, não dispõe de uma etapa prévia para treinamento, sendo o respectivo conjunto utilizado na mesma etapa em que os dados de teste são apresentados; esta abordagem recebe este nome por adiar o uso dos dados de treinamento.

O *k*NN consiste na regra de atribuir a uma amostra utilizada como entrada, da qual não se conhece a respectiva classe, aquela mais frequente entre os *k* pontos mais próximos (os *vi-*

zinhos mais próximos) no espaço, visto que é razoável afirmar que pontos próximos são mais suscetíveis a apresentarem características semelhantes em relação àqueles menos próximos. Esta regra, portanto, visa ponderar as similaridades das instâncias que se encontram no conjunto de treinamento T , para então desempenhar a classificação.

No que diz respeito à similaridade entre pontos em um espaço m -dimensional, calcula-se

$$\text{sim}(a, b) = \sqrt{\sum_{i=1}^m s(a_i, b_i)} \quad (3.1)$$

onde $s(a_i, b_i)$ torna a equação uma distância euclidiana quando $s(a_i, b_i)$ assume o primeiro caso (Equação 3.2); há também a possibilidade de envolver dados simbólicos (o segundo caso da Equação 3.2), não contínuos, no caso de algoritmos semelhantes, como o IBk (AHA E KIBLER, 1991).

$$s(a_i, b_i) = \begin{cases} (a_i - b_i)^2, & \text{se } a_i \text{ e } b_i \text{ são contínuos} \\ a_i \neq b_i, & \text{caso contrário} \end{cases} \quad (3.2)$$

Dentre os possíveis valores de k , COVER E HART (1967) ressaltam que a regra do *vizinho* mais próximo ($k = 1$) é a mais aceitável, não existindo outra regra ($k > 1$) que assuma menor probabilidade de erro. Todavia, em prol de maiores taxas de acerto, é coerente atribuir um valor grande para k , e ao mesmo tempo proporcionalmente pequeno (em relação ao número de instâncias de T), de forma a aumentar a precisão do classificador.

Considerando o princípio de funcionamento baseado na maioria dos votos entre os k *vizinhos* mais próximos, apesar de não haver nenhuma restrição em relação aos possíveis valores de k , adota-se valores ímpares para evitar a possibilidade de empates. Caso contrário, é necessário que, de maneira secundária, critérios de desempate sejam aplicados sobre a entrada a ser classificada, como a ponderação de cada voto baseada na distância do k -ésimo *vizinho*.

Em contrapartida da ausência de uma etapa de treinamento nos métodos de *lazy learning*, é importante considerar aspectos como a qualidade de generalização do classificador, diretamente associada à disposição das instâncias de T no espaço; o desempenho do algoritmo, visto que sua execução pode ser custosa sob o ponto de vista computacional caso T contenha muitas instâncias; a capacidade e viabilidade de um sistema para armazenar T , possibilitando então a ação do classificador; a intolerância a dados ruidosos ou irrelevantes, uma vez que a acurácia

dependa essencialmente da qualidade dos exemplos descritos em T (AHA E KIBLER , 1991).

3.2 Árvore de decisão

A árvore de decisão é um método destinado à classificação, sendo um dos mecanismos de aprendizado de máquina mais conhecidos pela facilidade de interpretação dos resultados, seja na forma de regras estruturadas em árvore, ou por suas fronteiras de separabilidade quando o espaço é visualizável em duas ou três dimensões.

Desenvolvido por QUINLAN (1986), o primeiro algoritmo de árvores de decisão é conhecido por ID3 (Iterative Dichotomiser 3), sendo este aprimorado, culminando na versão mais recente, denominada C4.5. Atribui-se a esta versão, no entanto, restrições legais no que diz respeito ao uso de programas. Assim sendo, a metodologia do presente trabalho faz uso de uma implementação *opensource* do algoritmo C4.5, denominada J48, disponível na ferramenta WEKA¹.

3.2.1 Algoritmo *Dividir-para-conquistar*

A definição dos nós de uma árvore de decisão é realizada a partir do algoritmo *dividir-para-conquistar*, desempenhando análises isoladas de atributos presentes em um conjunto de dados. O princípio de funcionamento deste algoritmo consiste em seccionar o conjunto de dados original, gerando subconjuntos cujas instâncias sejam pertencentes a uma única classe.

Antes de abordar as etapas deste algoritmo, faz-se necessário estabelecer uma notação a ser utilizada em seguida. Seja T um conjunto de dados utilizado como entrada, este possui n atributos $\{T_1, T_2, \dots, T_n\}$. Caso T_i seja discreto, assumindo m valores distintos, os subconjuntos de instâncias de cada valor são dados por $\{T_{i,1}, T_{i,2}, \dots, T_{i,m}\}$; a notação concernente a atributos contínuos é elucidada em 3.2.2. A quantidade de instâncias de um conjunto S qualquer pertencentes a uma classe C_j é dada por $\text{freq}(C_j, S)$.

Define-se inicialmente o cálculo da quantia média de informações necessárias para identificar uma classe, o que é chamado de entropia. Medida em *bits*, a entropia é dada por

$$\text{ent}(S) = - \sum_{j=1}^k \frac{\text{freq}(C_j, S)}{|S|} \log_2 \left(\frac{\text{freq}(C_j, S)}{|S|} \right) \quad (3.3)$$

¹Waikato Environment for Knowledge Analysis - Universidade de Waikato. Disponível em <http://www.cs.waikato.ac.nz/~ml/weka/>.

onde $|S|$ é o tamanho total de S .

Seccionando T , uma medida similar é calculada para cada T_i na equação 3.4, e então o ganho de informação, na equação 3.5.

$$\text{info}_x(T_i) = \sum_{j=1}^m \frac{|T_{i,j}|}{|T_i|} \text{ent}(T_{i,j}) \quad (3.4)$$

$$\text{gain}(T_i) = \text{ent}(T) - \text{info}_x(T_i) \quad (3.5)$$

O critério de seleção do atributo a integrar a estrutura de árvore se baseia na seleção do atributo T_i que apresenta maior ganho de informação. Ao percorrer a árvore a ser construída, descarta-se determinadas instâncias de T , por não satisfazerem as condições atribuídas aos nós percorridos. Desta forma, este critério de seleção, transformando atributos em regras condicionais, deve ser aplicado recursivamente para a definição dos nós seguintes, até que uma folha seja alcançada, representando instâncias isoladas pertencentes a uma única classe.

3.2.2 Atributos contínuos

Seja T_i um atributo composto por valores contínuos a cada instância e isolado de T , este deve ser ordenado, mantendo apenas os valores distintos deste conjunto, de forma a obter $\{v_1, v_2, \dots, v_m\}$. Deve-se então determinar limiares (*thresholds*) entre v_i e v_{i+1} , resultando em outro conjunto, de $m - 1$ elementos, tal que o i -ésimo termo deste segundo conjunto é definido na equação 3.6.

$$t_i = \frac{v_i + v_{i+1}}{2} \quad (3.6)$$

Para cada um destes limiares, divide-se T_i em dois subconjuntos S_1 e S_2 , os quais abrangem, respectivamente, os valores que não excedem o atual limiar (ou simplesmente $\{x \in T_i \mid x \leq t_i\}$) e os valores restantes ($T_i - S_1$). Cada limiar é então avaliado por

$$\frac{|S_1|}{|T_i|} \text{ent}(S_1) + \frac{|S_2|}{|T_i|} \text{ent}(S_2) \quad (3.7)$$

e eventualmente selecionado como o ponto de corte mais indicado, caso a equação 3.7 resulte no menor valor.

3.2.3 *Pruning*

O processo recursivo para a definição de nós de uma árvore de decisão, até que se isole todas as instâncias de uma única classe nas folhas desta estrutura, pode influenciar negativamente em determinadas aplicações salientando um potencial *overfitting*. Isto é observável, visto que esta separação de instâncias do conjunto de treinamento implicaria em uma acurácia de 100% na classificação, caso o mesmo conjunto fosse utilizado na etapa de testes.

O conceito de *pruning* visa atenuar a memorização dos casos apresentados na etapa de treinamento, eliminando os nós mais próximos às folhas da árvore de decisão correspondentes às fronteiras de separabilidade que seccionam minuciosamente o espaço. Esta estratégia resulta em subespaços maiores, intensificando a capacidade de generalização da estrutura do classificador. Considera-se generalização, neste caso, o equilíbrio entre *overfitting* e *underfitting* ao remover nós de uma árvore de decisão. Para esta tarefa, diferentes métodos são aplicáveis de forma a se obter tal equilíbrio, como *Reduced Error Pruning* (REP), *Pessimistic Error Pruning* (PEP) e *Critical Value Pruning* (CVP), cujos princípios de funcionamento são discutidos a seguir, e também *Cost-Complexity Pruning* (CCP/CART) e *Error-Based Pruning* (EBP) (KUNCHEVA, 2004).

A estratégia de redução de erros de *pruning* (REP) é considerada como a mais simples entre as formas de se reduzir uma árvore de decisão. É necessário um conjunto de dados extra, denominado *pruning set* (T_p), cujas amostras contidas não fazem parte daquelas vistas na etapa de treinamento.

Tem-se como objetivo do REP substituir por folhas os nós imediatamente acima das efetivas folhas da estrutura original, elegendo a classe C_j com maior valor de $\text{freq}(C_j, T_p)$ nas instâncias restantes naquele nó. Aplica-se então uma etapa de testes sobre as árvores alterada e original utilizando T_p ; se a substituição de determinado nó culminar em maior taxa de acerto da árvore alterada, em relação à original, o nó se torna efetivamente uma folha; caso contrário, o nó é mantido. Contudo, tem-se como desvantagem do REP a propensão ao *overpruning*, afetando a capacidade de generalização.

No caso do PEP, não se faz necessário o uso de T_p , e os nós da árvore são percorridos da raiz às folhas. O PEP se baseia em uma análise de complexidade dada pelo número de folhas por nó. Utilizando a notação de KUNCHEVA (2004), n atribui-se ao número de instâncias isoladas por um nó t e $e(t)$ ao número de erros caso t fosse substituído por uma folha (cujo critério para

seleção da classe se assemelha ao REP). Considerando T_t como a subárvore com raiz em t e L_t como o conjunto de folhas de T_t , define-se $e'(T_t)$ na equação 3.8.

$$e'(T_t) = \sum_{l \in L_t} e(l) + \frac{|L|}{2} \quad (3.8)$$

$$e(t) \leq e'(T_t) + \sqrt{\frac{e'(T_t)[n - e'(T_t)]}{n}} - \frac{1}{2} \quad (3.9)$$

Um nó t deve ser substituído por uma folha caso a condição da equação 3.9 seja verdadeira. KUNCHEVA (2004) ressalta que há a possibilidade de *underpruning* ou *overpruning*.

Dos nós imediatamente acima das folhas, em direção à raiz, a abordagem de *pruning* por valor crítico (CVP) faz deste valor um limiar relacionado à taxa de acerto. Temporariamente tidos como folhas, os nós passam a ser efetivamente substituídos quando a taxa de acerto da estrutura temporária não destoar além do limiar definido em relação a taxa original. É importante mencionar que quando o valor crítico de um nó está abaixo do limiar, mas a mesma condição não se aplica a uma ramificação descendente, este nó é mantido. Nesta abordagem, uso de T_p é aplicável, mas não se garante que a melhor configuração da estrutura seja obtida.

3.3 *Random Forest*

Inspirado nas árvores de decisão, o mecanismo consiste na combinação de diversas estruturas deste tipo de classificador, coletivo que metaforicamente remete à idéia de floresta. Proposto por BREIMAN (2001), o princípio de funcionamento de uma *random forest* (RF) é descrito a seguir.

Cada árvore de decisão deste mecanismo se forma de maneira distinta a partir de uma amostragem aleatória do conjunto de dados original. Não sendo somente a coleção de dados específica apresentada em treinamento o quesito terminante para se obter estruturas distintas de classificadores, a aleatoriedade característica a este método ocorre na seleção parcial (e aleatória) de atributos que serão avaliados para então definir um nó de uma árvore.

Seja $|F|$ o número de árvores de uma *random forest*, é necessário confeccionar $|F|$ conjuntos de treinamentos a partir do conjunto T original. Denota-se aqui, por $T^{(k)}$, o conjunto de dados utilizado para o treinamento da k -ésima árvore de decisão. Cada $T^{(k)}$ consiste em uma amos-

tragem aleatória simples de T com reposição (tal que $|T^{(k)}| = |T|$), isto é, uma mesma instância pode ser sorteada mais de uma vez, assim como também pode nunca ser sorteada. Este processo de amostragens para o treinamento de diversos classificadores em prol de uma única tarefa de classificação é chamado de *bootstrapping*.

A etapa de treinamento da árvore F_k , utilizando $T^{(k)}$, assim como elucidado em 3.2.1, se inicia pelo processo recursivo de análise dos atributos, selecionando os nós que compõem a estrutura do classificador; destoa-se, no entanto, no que diz respeito ao número de atributos a serem analisados. Do total de M atributos, m são selecionados de forma aleatória, tal que $m < M$, os quais são então analisados, elegendo aquele mais apropriado em termos de ganho de informação (equação 3.5) para separar as instâncias de $T^{(k)}$ e compor a estrutura de F_k . Vale ressaltar que não se aplica o *pruning* nestas estruturas.

Em modo de operação, uma amostra é submetida às árvores de decisão da *random forest*, as quais eventualmente apresentam diferentes saídas. Cada saída é então considerada como um voto para determinada classe e a classificação efetiva deste mecanismo aponta aquela mais votada pelas $|F|$ árvores como a classe inerente à amostra apresentada como entrada. BREIMAN (2001) ressalta que um número elevado de árvores não provoca *overfitting*, mas produz um limite na capacidade de generalização.

3.4 *Perceptron* multicamadas

O *Perceptron* multicamadas (PMC), do inglês, *Multi-Layer Perceptron* (MLP), é uma das arquiteturas mais utilizadas entre as Redes Neurais Artificiais (RNA), conhecida pela versatilidade e aplicabilidade em diversos contextos. Além do PMC, as RNAs abrangem arquiteturas como a *Radial Basis Function*, desempenhando aprendizado semi-supervisionado; Rede e mapa auto organizável de Kohonen, utilizados para agrupamento (não supervisionado); *Learning Vector Quantization*, versão supervisionada da rede de Kohonen; Memórias associativas de Hopfield, sob o aprendizado supervisionado; *Adaptive Resonance Theory*, utilizando o aprendizado *online*.

O PMC é uma arquitetura de alimentação adiante (*feedforward*) comumente utilizada para reconhecimento de padrões, aproximação de função, previsão de séries temporais, regressão, entre outras aplicações. Este mecanismo passou a ter notoriedade após o trabalho de RUMELHART et al. (1986), no qual elucida-se de forma consistente o princípio de funcionamento do algoritmo de retropropagação (*backpropagation*), próprio da arquitetura PMC.

Nas subseções seguintes, maiores detalhes da arquitetura, bem como das rotinas computacionais envolvidas serão elucidados de forma conceitual, com o propósito de revisar a metodologia. Um acompanhamento aprofundado, do ponto de vista algorítmico, é abordado em HAYKIN (2001) e SILVA et al. (2010).

3.4.1 Neurônio artificial

O PMC consiste em uma arquitetura e, sob uma configuração topológica, as rotinas computacionais envolvidas são executadas. A etapa de treinamento é iniciada, realizando ajustes em toda a estrutura, até que tal processo seja finalizado, quando a rede alcançou um estado tido como satisfatório. Esta estrutura é composta por camadas de neurônios paralelamente dispostos, cujas saídas se conectam a camadas posteriores, ou à saída do mecanismo; de forma semelhante, as entradas recebem saídas de camadas anteriores, ou da camada de entrada do mecanismo.

As partes que compõem uma rede PMC são denominadas neurônios artificiais, cuja concepção, proposta por MCCULLOCH E PITTS (1943), foi inspirada no neurônio biológico e então modelada matematicamente; este modelo ainda é o mais simplificado e utilizado como unidades de processamento nas redes PMC (SILVA et al. , 2010).

Figura 3.1: Neurônios biológico e artificial. (a) Estrutura de um neurônio biológico (Adaptado de http://mns.k.u-tokyo.ac.jp/~mashio/neuron_BK.svg); (b) Modelo de neurônio artificial, inspirado na estrutura biológica, proposto por MCCULLOCH E PITTS (1943).

Conforme ilustra a Figura 3.1, é possível examinar os elementos do neurônio artificial e observar as semelhanças com a estrutura biológica. Percorrendo a via dendrito-axônio, as entradas x_i do neurônio artificial são processadas, sendo inicialmente ponderadas pelos pesos sinápticos w_i ; juntamente com o limiar de ativação θ , estas são agregadas pelo combinador linear (Σ), resultando em um potencial u de ativação; uma função de ativação $g(u)$ satura o valor de u em um intervalo específico, resultando em um sinal de saída a ser consolidado na aplicação em

questão, ou utilizado como entrada para neurônios da camada neural posterior. De maneira específica, y pode ser definido por

$$y = g \left(\sum_{i=0}^n w_i \cdot x_i \right) \quad (3.10)$$

considerando θ como w_0 e -1 como x_0 .

Uma função de ativação $g(u)$ deve ser utilizada de acordo com o domínio apresentado ao neurônio, considerando adequações como a normalização dos dados. Podem ser utilizadas funções parcialmente diferenciáveis (cujas derivadas de primeira ordem não são conhecidas em todo o domínio), ou totalmente diferenciáveis. As funções degrau, degrau bipolar e rampa simétrica (Equações 3.11, 3.12 e 3.13) são exemplos de funções de ativação parcialmente diferenciáveis. As funções logística e tangente hiperbólica, conhecidas como sigmoidais, bem com a função linear (Equações 3.14, 3.15 e 3.16) são totalmente diferenciáveis (SILVA et al. , 2010).

$$g(u) = \begin{cases} 1 & , \text{se } u \geq 0 \\ 0 & , \text{se } u < 0 \end{cases} \quad (3.11)$$

$$g(u) = \begin{cases} 1 & , \text{se } u \geq 0 \\ -1 & , \text{se } u < 0 \end{cases} \quad (3.12)$$

$$g(u) = \begin{cases} a & , \text{se } u \geq a \\ u & , \text{se } -a \leq u \leq a \\ -a & , \text{se } u < -a \end{cases} \quad (3.13)$$

$$g(u) = \frac{1}{1 + e^{-\beta \cdot u}} \quad (3.14)$$

$$g(u) = \frac{1 - e^{-\beta \cdot u}}{1 + e^{-\beta \cdot u}} \quad (3.15)$$

$$g(u) = u \quad (3.16)$$

Antes de abordar o algoritmo de retropropagação utilizado para treinamento, elucidase na subseção seguinte a estrutura do PMC. O funcionamento do algoritmo sobre tal estrutura é então visto na subseção 3.4.3.

3.4.2 Arquitetura

A arquitetura do PMC consiste na propagação dos sinais de entrada (os dados de determinada aplicação $\{x_1, x_2, \dots, x_n\}$) adiante às camadas compostas por neurônios paralelamente dispostos, incumbidos de processar estas entradas. Uma rede é composta pela camada de entrada (não neural), camadas neurais intermediárias, denominadas ocultas, e uma camada de saída, cujo tamanho desta última deve ser igual ao número de saídas. A Figura 3.2 ilustra a disposição dos neurônios em camadas de uma rede PMC.

Figura 3.2: Estrutura de uma rede PMC. Adaptado de SILVA et al. (2010)

As n entradas enviadas à rede são recebidas por todos os neurônios da primeira camada neural, mantendo o fluxo até que toda a estrutura seja percorrida, transformando os n sinais de entrada em n_3 sinais de saída. Nesta arquitetura não há realimentações, isto é, em nenhum momento os sinais da rede são enviados para uma camada já percorrida.

Sob esta arquitetura, a definição (configuração) do tamanho das camadas neurais (n_1 , n_2 e n_3) designa uma topologia. Apesar do número de dimensões do espaço, bem como as entradas e saídas da rede influenciarem na configuração de uma topologia, não existe método específico para a escolha de tamanho e número de camadas neurais ocultas para que uma boa aproximação seja realizada; isto remete a uma questão empírica. Uma solução alternativa admissível é realizar repetidamente o mesmo treinamento, atribuindo faixas de valores (e seus incrementos) a serem adotados, a cada iteração, para cada um destes parâmetros, incluindo taxa de aprendizado, termo de *momentum*, entre outros envolvidos, mencionados na subseção 3.4.3. Este processo iterativo que comuta valores de parâmetros, visando encontrar a melhor combinação, é chamado

de *grid search*.

Como pode ser visto na Figura 3.2, o número de camadas ocultas é intencionalmente representado de maneira estática, inibindo a intuição de utilizar mais do que duas camadas ocultas, embora tal prática seja factível. Isto se deve à capacidade da rede PMC de realizar aproximação de uma função contínua arbitrária utilizando apenas uma camada oculta, ou ainda mapeamentos em $\mathbb{R}^n$ com no máximo duas camadas ocultas, utilizando funções de ativação sigmoidais (CYBENKO, 1989).

3.4.3 Algoritmo de retropropagação

O algoritmo de retropropagação é composto de duas principais etapas: a propagação adiante (*forward*) do sinal, assim como ocorre quando a estrutura está em modo de operação, e a retropropagação (*backward*), destinada a mensurar os erros obtidos como saída na etapa anterior, realizando, camada a camada, os devidos ajustes nos pesos sinápticos dos neurônios da rede. A explanação a seguir emprega a notação utilizada por SILVA et al. (2010).

Assimilando de forma mais detalhada a estrutura da Figura 3.2, w é a matriz de pesos sinápticos referentes aos neurônios de uma camada neural, onde $w_{ji}^{(L)}$ representa o valor do i -ésimo peso, do j -ésimo neurônio, da L -ésima camada neural. Vale ressaltar que o i -ésimo peso pondera a i -ésima saída da $(L - 1)$ -ésima camada, ou seja, i faz referência a um dos pesos do neurônio j de determinada camada, assim como também referencia a saída do neurônio da camada anterior.

Denota-se aqui, por $I_j^{(L)}$, o potencial de ativação u do neurônio j , da camada L , e por $Y_j^{(L)}$ a efetiva saída deste mesmo neurônio. Sendo assim,

$$I_j^{(L)} = \sum_{i=0}^{n_{(L-1)}} w_{ji}^{(L)} \cdot Y_i^{(L-1)}, \quad L \geq 1 \quad (3.17)$$

$$Y_j^{(L)} = \begin{cases} x_j & , \text{ se } L = 0 \\ g(I_j^{(L)}) & , \text{ caso contrário} \end{cases} \quad (3.18)$$

ressaltando ainda que $Y_0^{(L)} = x_0 = -1$, sendo que o limiar de ativação é dado por $w_{j0}^{(L)}$. A etapa *forward* em uma rede de l camadas neurais é então concluída ao obter $Y^{(l)}$.

Para a etapa *backward*, são utilizadas as funções erro quadrático e erro quadrático médio, a

fim de mensurar os desvios da saída da rede em relação aos valores desejados:

$$E(k) = \frac{1}{2} \sum_{j=1}^{n_l} (d_j(k) - Y_j^{(l)}(k))^2 \quad (3.19)$$

$$E_M = \frac{1}{p} \sum_{k=1}^p E(k) \quad (3.20)$$

onde $d_j(k)$ representa o j -ésimo valor da saída desejada, e $Y_j^{(l)}(k)$ o respectivo valor de saída da rede a ser comparado. O erro quadrático médio (E_M) consiste na média dos valores de $E(k)$ de cada uma das p amostras.

A etapa *backward* é então iniciada ajustando os pesos sinápticos da camada neural de saída, aquela mais próxima aos resultados esperados (d). Este ajuste consiste em calcular o gradiente descendente local ($-\delta^{(l)}$), incrementando, de forma ponderada, aos pesos atuais, a derivada da função erro quadrático (Equação 3.19) em relação aos pesos sinápticos da última camada:

$$\nabla E^{(l)} = \frac{\partial E}{\partial w_{ji}^{(l)}} = \frac{\partial E}{\partial Y_j^{(l)}} \cdot \frac{\partial Y_j^{(l)}}{\partial I_j^{(l)}} \cdot \frac{\partial I_j^{(l)}}{\partial w_{ji}^{(l)}} \quad (3.21)$$

$$= -(d_j - Y_j^{(l)}) \cdot g'(I_j^{(l)}) \cdot Y_i^{(l-1)} \quad (3.22)$$

$$\delta_j^{(l)} = (d_j - Y_j^{(l)}) \cdot g'(I_j^{(l)}) \quad (3.23)$$

$$\Delta w_{ji}^{(l)} = -\eta \cdot \frac{\partial E}{\partial w_{ji}^{(l)}} = \eta \cdot \delta_j^{(l)} \cdot Y_i^{(l-1)} \quad (3.24)$$

onde η é a taxa de aprendizado, que indica a velocidade de ajuste dos pesos durante a etapa de treinamento; $\delta_j^{(l)}$ é o gradiente local em relação neurônio j ; g' é a derivada da função de ativação.

O ajuste de pesos para as camadas ocultas (quando $1 \leq L < l$) se difere do processo acima exposto por não existir uma forma direta de mensurar desvios entre as saídas de determinada camada e respostas desejadas, assim como ocorre na camada l . Estes erros são retropropagados, estando em função da camada imediatamente posterior. Tem-se, portanto, para a camada $l-1$:

$$\nabla E^{(l-1)} = \frac{\partial E}{\partial w_{ji}^{(l-1)}} = \frac{\partial E}{\partial Y_k^{(l)}} \cdot \frac{\partial Y_k^{(l)}}{\partial I_k^{(l)}} \cdot \frac{\partial I_k^{(l)}}{\partial Y_j^{(l-1)}} \cdot \frac{\partial Y_j^{(l-1)}}{\partial I_j^{(l-1)}} \cdot \frac{\partial I_j^{(l-1)}}{\partial w_{ji}^{(l-1)}} \quad (3.25)$$

Desenvolvendo a Equação 3.25 considerando os termos em comum com as Equações 3.21 e 3.23, resulta-se em:

$$\frac{\partial E}{\partial w_{ji}^{(l-1)}} = - \left(\sum_{k=0}^{n_l} \delta_k^{(l)} \cdot w_{kj}^{(l)} \right) \cdot g'(I_j^{(2)}) \cdot Y_i^{(l-2)} \quad (3.26)$$

De maneira semelhante à camada l , define-se $\delta_j^{(l-1)}$, bem como a ponderação do ajuste da camada $l - 1$:

$$\delta_j^{(l-1)} = \left(\sum_{k=0}^{n_l} \delta_k^{(l)} \cdot w_{kj}^{(l)} \right) \cdot g'(I_j^{(l-1)}) \quad (3.27)$$

$$\Delta w_{ji}^{(l-1)} = -\eta \cdot \frac{\partial E}{\partial w_{ji}^{(l-1)}} = \eta \cdot \delta_j^{(l-1)} \cdot Y_i^{(l-2)} \quad (3.28)$$

Este mesmo processo é aplicável às camadas anteriores a $l - 1$, até que a camada 1 seja ajustada. De forma generalizada, substituindo a Equação 3.27 em 3.28, o ajuste de pesos para a L -ésima camada escondida é dado por:

$$\Delta w_{ji}^{(L)} = -\eta \cdot \frac{\partial E}{\partial w_{ji}^{(L)}} = \eta \cdot \left(\sum_{k=0}^{n_{L+1}} \delta_k^{(L+1)} \cdot w_{kj}^{(L+1)} \right) \cdot g'(I_j^{(L)}) \cdot Y_i^{(L-1)} \quad (3.29)$$

Vale recordar o caso $L = 1$, onde o último termo da Equação 3.29 passa a ser uma entrada da rede PMC, conforme definido na Equação 3.18.

O ajuste de pesos das l camadas deve ser realizado para cada par $\{x, d\}$ disponível no conjunto de treinamento, procedimento este que é chamado de *época*; ao final de cada época, calcula-se o erro quadrático médio (Equação 3.20). O algoritmo de retropropagação executa épocas repeditamente, até que um critério de parada seja atendido, cessando as iterações. A diferença entre os erros quadráticos médios das épocas atual e anterior sendo menor do que uma precisão ε definida, ou um limite de épocas a serem executadas são exemplos de critérios de parada para o algoritmo de treinamento da rede PMC.

Um aprimoramento simples ao algoritmo de retropropagação, contribuindo, entretanto, de forma significativa no processo de convergência, culminando em um menor número de épocas, é a inclusão do termo de *momentum* no ajuste dos pesos das camadas da rede:

$$w_{ji}^{(L)}(t+1) = w_{ji}^{(L)}(t) + \alpha \cdot (w_{ji}^{(L)}(t) - w_{ji}^{(L)}(t-1)) + \eta \cdot \delta_j^{(L)} \cdot Y_i^{(L-1)} \quad (3.30)$$

onde α é a taxa de *momentum*, tal que $0 \leq \alpha \leq 1$. Caso $\alpha = 0$, a convergência da rede passa a

ter o mesmo comportamento do algoritmo sem a contribuição do termo de *momentum*. O termo de *momentum* tem a finalidade de realizar ajustes sobre os pesos de maneira mais vigorosa quando o estado atual estiver longe do mínimo global (da função de erro), assim como ajustes minuciosos são realizados quando os pesos se aproximam do ótimo (w^*).

Reitera-se aqui a importância de procedimentos como o *grid search* para a obtenção da melhor configuração topológica, bem como dos parâmetros envolvidos (η , α e ε), uma vez que valores inadequados adotados a estes atributos culminem em desempenho insatisfatório.

Além da inclusão do termo de *momentum*, existem outros aprimoramentos para o algoritmo de retropropagação que oferecem um processo de convergência mais rápido, isto é, os treinamentos são realizados em números significativamente menores de épocas, como o caso do *Resilient Propagation* (RProp) ou o algoritmo Levenberg-Marquardt (LMA). Maiores detalhes para estes aprimoramentos podem ser vistos em HAYKIN (2001) e SILVA et al. (2010).

Capítulo 4

Análise e extração de características inerentes à hemiparesia

Este capítulo apresenta um estudo de caso para o desenvolvimento de uma estratégia de extração de conhecimento aplicada no contexto da reabilitação robótica. Elucida-se nas seções seguintes detalhes do curso de todo o processo de KDD, desde a aquisição de dados, à elaboração e aplicação da estratégia para obtenção de resultados.

Para todo processo de KDD, deve-se obter os dados de entrada, para então aplicar as técnicas concernentes às etapas seguintes. No contexto de aplicação deste trabalho, a obtenção/coleta dos dados sucede no decorrer das sessões de terapia física, sendo desempenhada pelo dispositivo robótico, o que consiste no armazenamento de informações oriundas dos esforços do paciente que, por sua vez, é orientado pelo ambiente virtual do *software* em execução.

Utilizado para reabilitação em clínicas e hospitais, o dispositivo robótico utilizado neste trabalho é o InMotion ArmTM (Figura 4.1(a), versão clínica do MIT-MANUS), da Interactive Motion Technologies¹ (Watertown, MA, EUA), dispondo de todo *hardware* envolvido na interação humano-computador com o paciente. Neste dispositivo, o ambiente virtual (um *serious game*, por exemplo), exibido em uma tela, deve atrair a atenção do paciente, sendo o *end-effector*, de um braço robótico de três graus de liberdade, o meio para interagir com o jogo em execução. O uso deste dispositivo, no entanto, não dispensa o ofício dos terapeutas no monitoramento e condução de tratamentos.

A coleta de dados do InMotion ArmTM compõe, a cada instância adquirida, um arranjo de

¹Vide <http://interactive-motion.com/healthcarereform/upper-extremity-rehabilitation/inmotion2-arm/>

Figura 4.1: Dispositivo robótico utilizado nas sessões de terapia física. (a) Dispositivo robótico InMotionArmTM utilizado na terapia física com *serious games* (adaptado de <http://imt.lynxdesign.com/wp-content/uploads/2012/11/InMotionARM240.jpg>); (b) *Serious game* utilizado nos exercícios de reabilitação neste trabalho, ilustrando o fluxo das movimentações que o paciente deve realizar; (c) Visualização dos dados (posição nos eixos x e y) de desempenho do paciente coletados durante uma sessão de reabilitação.

dados em relação ao *end-effector*: posição em metros nos eixos x e y , velocidade em metros por segundo nos eixos x e y , e força em newtons nos eixos x , y , e z . A aquisição de instâncias ocorre a cada cinco milissegundos (200 Hz), sendo o tempo (*offset*) também pertencente a este arranjo de dados. Ao final de cada sessão, os dados de desempenho são armazenados em arquivos, que representam movimentações individuais. Uma forma de visualizar os dados coletados referentes às posições é representada na Figura 4.1(c).

O *serious game* utilizado nas sessões de terapia é denominado *Clock*. Conforme ilustrado na Figura 4.1(b), este jogo apresenta uma disposição visual semelhante a de um relógio, com oito alvos periféricos, e um central; o objetivo se baseia em realizar movimentos de alcance, movendo o círculo amarelo, o qual indica a posição da mão do paciente, em direção ao alvo sinalizado

em vermelho. A cada movimentação em direção aos alvos periféricos, faz-se, em seguida, a movimentação em direção ao alvo central, delineando um padrão “estrela” após percorrer todos os alvos.

Ao final de cinco repetições deste padrão, medidas de desempenho (KREBS et al. , 2003) são calculadas e armazenadas junto aos demais dados coletados. A primeira medida, denominada *distance from target* indica, em milímetros, a distância entre a mão do paciente e o ponto central do alvo, no momento em que o dispositivo robótico deixa de prestar auxílio, sendo zero o valor ótimo para esta medida. A segunda medida, *robot power*, indica em miliwatts a assistência média prestada pelo robô; variando até a força máxima necessária para auxiliar o paciente, o valor ótimo para esta medida também é zero. A terceira medida é chamada de *motion jerk*, medindo, em metros por segundo ao cubo, as irregularidades das movimentações de alcance, naturais, por exemplo, de espasmos e tremores; o valor ótimo para esta medida é baseado na equação do mínimo *jerk*.

Um ensaio piloto (MORETTI et al. , 2014) foi realizado com o objetivo de analisar a relevância das variáveis envolvidas na coleta em relação às medidas *distance from target*, *robot power* e *motion jerk*. Os dados envolvidos são oriundos do desempenho de um único paciente, cuja identidade não foi informada, ao longo de 38 sessões de terapia física realizadas no Instituto Lucy Montoro; os dados foram fornecidos pelo Núcleo de Estudos Avançados em Reabilitação (NEAR).

Foi utilizado neste trabalho um conjunto de dados fornecido pelo Instituto Lucy Montoro (São Paulo, SP). Estes dados foram coletados em sessões de terapia física a partir do exercício de pacientes diagnosticados com hemiparesia, patologia esta que concerne ao domínio do problema de análise dos dados. Vale ressaltar que a identidade dos pacientes foi preservada, não constando nos dados utilizados em experimento quaisquer dados que designem eles com informações pessoais, ou ainda com o indivíduo.

O modelo de FAYYAD et al. (1996), dentre as alternativas mencionadas na seção 2.2, será utilizado para fundamentar a elaboração de estratégias de KDD, por ser destinado à área acadêmica, bem como por apresentar uma organização minuciosa e sistemática de etapas, as quais são abordadas nas seções seguintes.

4.1 Entendimento do problema e definição de metas

O domínio dos dados utilizados neste experimento retrata o desempenho de pacientes hemiparéticos no estágio crônico. Hemiparesia refere-se ao enfraquecimento (paresia) unilateral (hemi) do corpo, decorrente de um acidente vascular cerebral, cujo lado afetado no cérebro corresponde ao lado oposto do corpo. Diz-se dos lados hemiparéticos em relação ao lado do corpo que apresenta déficits motores (NATIONAL STROKE ASSOCIATION, 2006), ou seja, hemiparesia à esquerda está relacionada à fraqueza no lado esquerdo do corpo, havendo lesão no lado direito do cérebro, e vice-versa para a hemiparesia à direita. Quanto ao desenvolvimento temporal desta patologia, diz-se que o estado é agudo até a primeira semana, subagudo no período entre uma semana e um mês, e crônico quando o período é superior a um mês (THE INTERNET STROKE CENTER, 2011).

Apesar dos hemisférios cerebrais serem destinados a funções distintas, e que a paresia em cada hemisfério possa acarretar em sintomas característicos (NATIONAL STROKE ASSOCIATION, 2006), o presente trabalho atém-se à análise e extração de características do comportamento motor dos pacientes, de forma condizente à semântica dos dados passíveis de processamento e análise.

Com base nos objetivos estabelecidos na Seção 1.2, define-se como meta deste estudo de caso o desenvolvimento de uma estratégia de KDD para a obtenção de um modelo para determinar (classificar) o lado hemiparético de acordo com o desempenho do paciente apresentado. Isto implica em uma prévia extração de características (*features*) inerentes às saídas desejadas, entre outras atividades de pré-processamento, as quais serão elucidadas nas seções seguintes.

Os dados envolvidos neste trabalho descrevem o desempenho de dezesseis pacientes (oito homens e oito mulheres) hemiparéticos no exercício da reabilitação para membros superiores (ombro e cotovelo) durante sessões de reabilitação com o dispositivo InMotion ArmTM. Registradas no início dos tratamentos, variando numa faixa entre 33 e 74 anos, a média das idades dos pacientes é de 57 anos.

4.2 Seleção do conjunto de dados

Na iminência de se tornarem amostras, os dados de movimentos, quando inspecionados utilizando técnicas de visualização, como na Figura 4.1(c), passam por seleções devido a identificação

de eventuais partes inadequadas para análise, como os *outliers*. Neste cenário, a presença de dados desprezíveis é geralmente decorrente de usos indevidos do dispositivo robótico, ou ainda de indisposições apresentadas pelo paciente, as quais, sob o panorama do exercício, influenciam no desempenho motor de maneira significativa.

Na sua forma ríspida, o conjunto de dados consiste em uma coleção de arquivos no formato texto, representando separadamente movimentações ponto-a-ponto de alcance. A respeito da estrutura destes arquivos, dispostas em colunas, medidas como posição (p_x e p_y), velocidade (v_x e v_y) e força (f_x e f_y) nos eixos x e y são armazenadas. Os dados selecionados, submetidos às etapas seguintes para processamento e análise, ocupam um total de 135 *megabytes*.

A mescla de conjuntos de dados obtidos a partir de diferentes dispositivos robóticos foi desconsiderada para este experimento, devido a possibilidade de potenciais desgastes mecânicos, os quais eventualmente influenciariam no sensoramento fino do desempenho de um paciente em termos das medidas acima mencionadas; isto, por consequência, implicaria em dados espacialmente discrepantes entre diferentes dispositivos, assim como em amostras ruidosas, inadequadas para as etapas seguintes.

4.3 Pré-processamento

A partir do conjunto-alvo selecionado, os dados das movimentações dos pacientes serão pré-processados, de forma a confeccionar instâncias que posteriormente se tornarão amostras para as próximas etapas desta estratégia de KDD.

Recapitulando o funcionamento do aparato robótico envolvido, as movimentações de alcance realizadas pelo paciente completam um *ciclo* quando todos os alvos são alcançados. A partir de então, o processo de coleta reinicia, descrevendo a movimentação em direção ao primeiro alvo, e assim sucessivamente.

Apesar de selecionados, os dados do conjunto-alvo, até este momento, ainda apresentam aspecto ríspido, sendo dados diretamente oriundos do dispositivo robótico. Com o objetivo de preparar estes dados para as etapas seguintes, um pré-processamento foi aplicado, concatenando as movimentações (os 16 arquivos texto) em direção aos alvos (e em direção ao centro) do jogo *Clock* (Figura 4.1(b)) que sejam pertencentes a um mesmo ciclo. Cada uma destas concatenações deve descrever o desempenho de um paciente ao delinear o padrão “estrela”, de forma que a visualização das variáveis de posição destas concatenações apresentem aspecto semelhante àquele

da Figura 4.1(c).

Observa-se, no entanto, que cada paciente se encontra em uma condição específica para realizar a mesma tarefa, isto é, os movimentos ponto-a-ponto possuem duração variável, implicando em porções de dados de tamanho variável. Isto acaba sendo um empecilho para o processo de análise, não sendo conveniente considerar estas concatenações como amostras, assim sendo necessárias transformações (elucidadas a seguir), até que as instâncias destes ciclos sejam consideradas amostras.

4.4 Transformação

Devido à precisão no sensoramento do desempenho dos pacientes em termos de taxa de amostragem (aquisição de dados), a representação em dados das movimentações assumem proporções muito grandes para serem utilizadas como entradas de mecanismos inteligentes, o que remete à necessidade de aplicar transformações como a extração de *features*.

A etapa de transformação tem como objetivo extrair uma coleção de *features* de cada uma das grandes porções de dados obtidas na etapa anterior. Estas características devem carregar informações essenciais que são teoricamente inerentes à hemiparesia; o tamanho destas coleções de características deve ser muito menor do que as porções de dados anteriormente obtidas, representando-as de forma sucinta. Uma vez que seja possível a extração destas coleções, estas passam a se aproximar do formato de amostras, já sendo adequadas para alimentar as entradas de mecanismos para mineração de dados.

A fim de evitar ambiguidades ou interpretações equivocadas, faz-se necessário definir uma terminologia que será utilizada a partir de então nas seções seguintes em relação aos formatos de dados obtidos nas entradas/saídas das etapas de KDD. *Feature* refere-se à coleção de características extraída de cada porção de dados obtida no pré-processamento, tendo os seus elementos constituintes chamados de atributos. Amostra remete à instância, a um caso e ser apresentado aos mecanismos de mineração de dados, estando este no seu formato final ou não.

Entende-se por inerência, no contexto de um processo de KDD, a propensão das *features* extraídas a serem separáveis espacialmente, isto é, ao visualizar seus atributos em um gráfico de dispersão (*scatter plot*) bidimensional, por exemplo, é preciso haver *nuvens* (agregações) de pontos distintas, visualmente perceptíveis, constituindo padrões. Caso contrário, se a disposição espacial dos pontos apresentar aspecto emaranhado, cujas agregações não sejam detectáveis

visualmente ou programaticamente, diz-se que a *feature* não é inerente às metas de análise, às saídas (classes) desejadas, e uma nova estratégia de extração de *features* se faz necessária.

A estratégia utilizada neste caso, dadas as porções de dados obtidas no pré-processamento, consiste em calcular os momentos estatísticos da distribuição das colunas: média ($\bar{X}$), desvio padrão (σ), obliquidade (*skewness*, γ) e curtose (*kurtosis*, κ), colunas estas que são referentes às variáveis especificadas na subseção 4.2, produzindo *features* na estrutura representada a seguir:

$$\begin{aligned} &\{\bar{p}_x, \quad \bar{p}_y, \quad \bar{v}_x, \quad \bar{v}_y, \quad \bar{f}_x, \quad \bar{f}_y, \\ &\sigma_{p_x}, \quad \sigma_{p_y}, \quad \sigma_{v_x}, \quad \sigma_{v_y}, \quad \sigma_{f_x}, \quad \sigma_{f_y}, \\ &\gamma_{p_x}, \quad \gamma_{p_y}, \quad \gamma_{v_x}, \quad \gamma_{v_y}, \quad \gamma_{f_x}, \quad \gamma_{f_y}, \\ &\kappa_{p_x}, \quad \kappa_{p_y}, \quad \kappa_{v_x}, \quad \kappa_{v_y}, \quad \kappa_{f_x}, \quad \kappa_{f_y}\} \end{aligned} \tag{4.1}$$

Embora a *feature* extraída seja sucinta, como demonstra a estrutura da Equação 4.1, e descrevendo uma grande porção de dados, esta ainda não deve ser considerada adequada para a mineração de dados. Ainda é necessário avaliar sua inerência, e então desempenhar alguma atividade de redução sobre a estrutura das *features*, descartando atributos com contribuições irrelevantes.

Neste caso, com uma estrutura de tamanho 24, cada combinação de dois atributos foi inspecionada visualmente em um gráfico de dispersão em busca de duas concentrações diferentes de pontos, referentes aos lados hemiparéticos que se deseja detectar. Um total de 276 espaços bidimensionais distintos foram inspecionados, buscando os casos em que há separações mais evidentes.

Os atributos de força aparentam ser os mais inerentes aos lados hemiparéticos, assim como é ilustrado na Figura 4.2, havendo duas concentrações homogêneas e distintas de pontos, as quais são compostas por um único símbolo. Pontos amarelos e azuis representam, respectivamente, instâncias de ciclos cujas movimentações foram desempenhadas por pacientes com hemiparesia à esquerda e à direita.

Ainda acerca da Figura 4.2, observa-se uma intersecção entre ambas as concentrações de pontos nos quatro casos, o que pode ser interpretado como zonas de incertezas, nas quais não há demarcações de fronteiras capazes de realizar separações de forma satisfatória; logo, é necessário considerar espaços com mais dimensões de modo a viabilizar a separação. A coleção de atributos que compuser o espaço ideal (ou equivalente a este) para separações é então considerada uma *feature* no seu estado final na etapa de transformação.

Figura 4.2: Inspeção visual dos casos de separabilidades mais evidentes

Embora concernente à etapa de transformação, a seleção final de atributos neste trabalho ocorre essencialmente de forma conjunta à mineração de dados; isto se justifica pela capacidade dos algoritmos de aprendizado de máquina de mensurar a inerência dos atributos através de taxas de acerto.

A partir do conjunto de dados estruturados no formato definido em 4.1, inicialmente com 24 atributos, um único atributo é retirado temporariamente desta coleção, e um algoritmo de aprendizado de máquina é executado para a avaliação da inerência dos atributos restantes; o atributo cuja ausência causa menor impacto na taxa de acerto é então desconsiderado, resultando em um conjunto de dados com 23 atributos. Este critério foi aplicado repetidamente sobre o conjunto de dados até que indícios de perda de inerência dos atributos remanescentes sejam observáveis, isto é, as taxas de acerto se aproximam de 90%, tendendo a zero. Maiores detalhes sobre critérios de seleção de atributos são elucidados na subseção 4.6.

Após o processo da transformação efetiva dos dados ser satisfatório (o que resultou na estrutura definida em 4.1), é importante ressaltar ainda a relevância desta etapa, especificamente no que diz respeito às inspeções visuais para a seleção de atributos. Mesmo não deliberando-se acerca de irrelevâncias de atributos, o que os tornariam passíveis de exclusão, faz-se uma ponderação daqueles mais inerentes, que oferecem melhor separabilidade, tornando-os menos propensos a serem desconsiderados no processo de seleção que ocorre juntamente da mineração de dados.

4.5 Definição de técnicas e algoritmos

De forma precedente à mineração de dados utilizando algoritmos de aprendizado de máquina, conforme já elucidado na seção 2.2.1, faz-se a escolha de uma técnica específica em conformidade com as metas estabelecidas anteriormente, bem como o formato dos dados que serão utilizados neste experimento, estratégia esta que abrange uma gama específica de algoritmos; elege-se então as opções mais aplicáveis, possibilitando a comparação de resultados na mineração de dados.

Visto que é possível discernir as diferentes aglomerações de pontos nos espaços da Figura 4.2, tem-se as saídas desejadas conhecidas, também chamadas de classes, respectivas às instâncias. Em relação às metas de KDD estabelecidas, pleiteando a determinação do lado hemiparético do paciente a partir de características de seu desempenho, a classificação é a técnica mais adequada para abordar este tipo de problema.

Para a seleção de algoritmos de classificação, foram considerados aspectos como facilidade de interpretação dos resultados, capacidade de seccionar regiões espaciais de forma não linear e boa aceitação no campo.

Árvore de decisão (J48) e *Random Forest* foram algoritmos selecionados para este experimento devido à facilidade de interpretação através de regras, compreensíveis por humanos, encadeadas em estrutura de árvore; o *k-Nearest Neighbors* também se enquadra em boa interpretabilidade, também sendo um algoritmo candidato para as análises. Em contraste, foi considerado o *Perceptron* Multicamadas que, mesmo apresentando uma estrutura complexa, de difícil compreensão, oferece a capacidade de obter uma configuração eventualmente próxima da configuração ótima, no que diz respeito à disposição espacial de suas fronteiras de separabilidade, através da atividade de minimização do algoritmo *backpropagation*.

Os algoritmos de aprendizado de máquina acima mencionados, selecionados para desem-

Em busca dos atributos mais relevantes, foi utilizado o J48 com o objetivo de analisar a estrutura em árvore produzida (Figura 4.3(a)), na qual, em termos de entropia e ganho de informação, o arranjo dos atributos mais inerentes está na raiz da estrutura, enquanto aqueles menos relevantes estão próximos às folhas.

Adaptando, neste caso, a estrutura de árvore obtida, tem-se uma forma reduzida, na qual nós repetidos e sequencialmente encadeados são mesclados; isto resulta na estrutura da Figura 4.3(b) que evidencia a hierarquia dos atributos, desde o mais relevante, na raiz, até os menos relevantes, nas folhas. Os nós dos três primeiros níveis desta estrutura reduzida foram tidos como mais inerentes, posto que os níveis seguintes tendem a abranger atributos para separações minuciosas das instâncias apresentadas ao algoritmo, o que denota graus ínfimos de inerência a tais atributos à proporção que a profundidade da estrutura aumenta.

Sendo mais uma forma de ponderar a relevância dos atributos, inspecionando-os visualmente e de forma individual, aqueles cujas concentrações de pontos estão ligeiramente separadas foram os primeiros a se desconsiderar do total de 24 atributos, para então avaliar o impacto da ausência de cada um destes no espaço através de algoritmos de aprendizado de máquina. Vale ressaltar, no que diz respeito a tal avaliação, que os algoritmos de aprendizado de máquina utilizados foram os mesmos adotados para desempenhar a classificação do formato final das *features*; neste ponto, onde a classificação não é essencialmente o fim, foram utilizados valores arbitrários aos parâmetros dos algoritmos.

Figura 4.4: Visualização de atributos de forma isolada

A Figura 4.4 ilustra a disposição das instâncias sob os atributos que melhor distinguem as diferentes concentrações; o resultado desta inspeção reforça ainda os resultados obtidos pela análise do algoritmo J48, dado que os quatro atributos da Figura 4.4 se encontram nos níveis tidos como aceitáveis na estrutura da Figura 4.3(b). Para evitar a dificuldade na visualização,

dada pela sobreposição de pontos, foi aplicado nestes pontos um ruído na direção vertical em faixas distintas, referentes às aglomerações de cada classe.

As estratégias de seleção e exclusão de atributos resultaram na seguinte *feature*:

$$\{\bar{f}_x, \sigma_{v_y}, \sigma_{f_y}, \gamma_{p_x}, \gamma_{v_x}, \gamma_{v_y}, \gamma_{f_x}, \gamma_{f_y}\} \quad (4.2)$$

a fim de tornar este experimento reproduzível, os conjuntos de dados contendo as *features* descritas em termos das Equações 4.1 e 4.2 estão disponíveis² no formato ARFF, próprio da ferramenta WEKA.

Dado o formato final das instâncias, a efetiva mineração de dados ocorre a partir de diversas execuções dos algoritmos selecionados, adotando valores diferentes aos devidos parâmetros a cada uma destas execuções com o objetivo de obter uma configuração satisfatória que alcance altas taxas de acerto. Para um ajuste fino dos parâmetros dos algoritmos envolvidos, foi realizado um *grid search* havendo validação cruzada (10-*fold*) para cada configuração testada.

Tabela 4.1: Dados do *grid search* e resultados dos treinamentos

Algoritmo	Parâmetros	Intervalos	Passos	Melhor config.	Acerto
<i>k</i> NN	<i>k</i>	[1, 50]	50	<i>k</i> = 5	93%
J48	<i>C</i>	[0.1, 0.9]	100	<i>C</i> = 0.21	81.7%
RF	<i>T</i>	[1, 1000]	1000	<i>T</i> = 5	89.5%
PMC	<i>H</i>	[1, 10]	10	<i>H</i> = 1	94%
	<i>L</i>	[0.01, 0.5]	15	<i>L</i> = 0.3	
	<i>M</i>	[0.1, 0.5]	5	<i>M</i> = 0.2	
	<i>E</i>	[500, 2000]	3	<i>E</i> = 500	
	<i>F</i>	-	-	<i>F</i> = equação 3.14	

A Tabela 4.1 abrange o resultado dos treinamentos dos algoritmos sob a melhor configuração dentre as faixas de valores estabelecidas aos devidos parâmetros. Elucidando a estrutura da tabela, a coluna *Parâmetros* indica quais parâmetros sofreram ajustes através do *grid search*, onde *k* é o número de vizinhos mais próximos do algoritmo *k*NN; *C* diz respeito ao fator de confiança do algoritmo J48, utilizado para o *pruning*; *T* é o número de árvores de uma *random forest*; *H*, *L*, *M*, *E* e *F* representam, respectivamente, o tamanho da primeira camada oculta, taxa de aprendizado, termo de *momentum*, número de épocas de uma rede neural *Perceptron* multicamadas e função de ativação. A coluna *Intervalos* descreve as faixas de valores (igualmente espaçados) atribuídos aos parâmetros dos algoritmos; a coluna *Passos* indica o número de valores

²Os conjuntos de dados *Hemiparesis1* e *Hemiparesis2* estão disponíveis em <https://github.com/moretticb/Research>.

nos intervalos estabelecidos na coluna anterior. A melhor configuração dentre as possíveis opções dadas pelas colunas *Intervalos* e *Passos* é mostrada na coluna *Melhor config.*, configurações estas que resultam nas taxas de acerto mostradas na coluna *Acerto*.

Pertinente às etapas seguintes do processo de KDD, a interpretação e consolidação dos resultados obtidos nesta etapa são discutidas nos capítulos seguintes.

Capítulo 5

Interpretação da saída do KDD

Este capítulo expõe os resultados do processo de KDD de FAYYAD et al. (1996) aplicado sobre a coleção de dados inicialmente ríspida, abordando também uma interpretação na forma de discussão. Sob o ponto de vista clínico, esta discussão salienta a semântica por trás das descobertas decorrentes dos mecanismos de aprendizado de máquina, mencionando também questões que tramam diretrizes para a continuidade deste estudo em trabalhos futuros.

Analisando o funcionamento dos algoritmos de aprendizado de máquina envolvidos, suas fronteiras de separabilidade (exceto o caso do k NN) são compostas por hiperplanos, os quais seccionam regiões do espaço linearmente. Isto posto, analisando os dados da Tabela 4.1, observa-se que a separação de maior precisão foi realizada pelo PMC. Apesar da dificuldade na interpretação de uma rede neural PMC, sua configuração topológica induziu uma interpretação interessante e simples: há apenas um neurônio na camada oculta, o que apresenta indícios dos dados serem linearmente separáveis, uma vez que a separação das classes por um único hiperplano culminou na maior taxa de acerto entre as demais topologias e algoritmos candidatos.

Através de uma estimativa da probabilidade de classes em determinados pontos de um espaço específico (FRANK E HALL, 2003), a Figura 5.1 ilustra a disposição da fronteira de separabilidade obtida a partir da topologia do PMC descrita na Tabela 4.1, possibilitando a visualização de subespaços onde as separabilidades de classes são mais evidentes. Ressalta-se aqui que a fronteira de separabilidade visualizada separa classes no espaço composto pelos atributos descritos na Equação 4.2.

A respeito dos atributos remanescentes, submetidos à mineração de dados, observa-se que dados de força compõem uma grande porção da *feature* como um todo. Até mesmo em estágios

Figura 5.1: Inspeção visual da fronteira de separabilidade do PMC

preliminares da seleção de atributos, na etapa de transformação, é perceptível através da Figura 4.2 a presença da força nos subespaços onde há boa separabilidade entre classes; é observável ainda que os atributos de força anteriormente evidentes são os mesmos utilizados no formato final da *feature*.

Sob este prisma, levando em consideração a precisão de 94% na classificação, pode-se dizer que a força possui forte inerência à hemiparesia quando se trata do desempenho em exercícios de reabilitação de membros superiores, ou mais precisamente exercícios de ombro e cotovelo. Este trabalho proporcionou uma perspectiva diferente de lidar com os dados, sobretudo aqueles de posição, os quais são intuitivamente concatenados à idéia de referências espaciais, como os

lados esquerdo e direito. No entanto, a posição pouco apresentou significância, não constituindo padrões de forma independente (em relação às demais grandezas). Em um ensaio piloto (MORETTI et al. , 2014) também foi possível constatar a relevância da força, cujos indícios foram decorrentes do impacto causado em medidas de desempenho (KREBS et al. , 2003), quando estas foram previstas (através de classificação) a partir de padrões onde os dados de força foram omitidos.

Completando esta discussão, é importante salientar também os detalhes obtidos no decorrer da estratégia KDD como um todo, os quais seriam propensos a compor diretrizes de trabalhos futuros. Pormenores estes, de cunho hipotético, são apresentados no próximo capítulo.

Capítulo 6

Considerações finais

O presente trabalho apresentou um estudo de caso de pacientes diagnosticados com hemiparesia no exercício da reabilitação. A fim de extrair características inerentes à patologia em questão, técnicas de descoberta de conhecimento (KDD) foram aplicadas sobre o conjunto de dados que inicialmente apresentava aspecto ríspido, descrevendo o desempenho de pacientes em termos de grandezas cinemáticas e dinâmicas diretamente coletadas das informações sensoriais do dispositivo robótico.

No decorrer do processo de KDD foi possível obter coleções reduzidas de atributos que descrevem grandes porções de dados de desempenho em termos de média, desvio padrão, *skewness* e *kurtosis*; estas coleções foram submetidas a algoritmos de aprendizado de máquina para realizar a mineração de dados. Uma rede neural artificial *Perceptron* multicamadas possibilitou o discernimento dos lados hemiparéticos com uma precisão de 94%, o que demonstra boa separabilidade entre classes. Os dados de força se destacaram, compondo metade dos dados de entrada utilizados para a classificação, sendo força a grandeza mais inerente à hemiparesia neste estudo de caso.

6.1 Perspectivas futuras

A zona de incerteza, visível em todos os subespaços aqui apresentados, é um potencial ponto a ser investigado, ocasionando o levantamento de hipóteses concernentes a um eventual estado de reabilitação, por exemplo, no qual as funções motoras do paciente não apresentariam indicativos de desordens inerentes a um lado hemiparético específico. De forma minuciosa, agrupando

instâncias por pacientes e fazendo de tais grupos séries temporais (em função das datas de cada evento), onde cada elemento consiste na distância entre a própria instância e o centro da zona de incerteza, surge a possibilidade de investigar se estas distâncias convergem para zero. Caso positivo, isto remete à idéia de que os indicativos da patologia atenuem com o exercício da reabilitação até que o desempenho do paciente, representado em termos da Equação 4.2, adentrando a zona de incerteza, seja uma evidência de recuperação.

Outra questão a ser explorada é a possibilidade de reduzir a quantidade de atributos inerentes à patologia em questão, pleiteando um espaço bidimensional onde a visualização possa abranger completamente o domínio dos dados a serem analisados. Tal arranjo pode eventualmente contribuir para melhor uma compreensão dos padrões encontrados, ou ainda requerer estratégias secundárias para a separação dos padrões.

Uma inspeção minuciosa nos resultados discutidos, bem como na estrutura obtida que constitui o padrão referente aos lados hemiparéticos, seja no estado atual, ou ainda em estado decorrente de melhorias futuras, configura a consolidação do conhecimento extraído. Incorporando a perícia clínica no labor de transladar o conhecimento técnico deste trabalho para o contexto clínico, de forma que as interpretações sejam sustentadas pela literatura médica, a saída do KDD torna-se passível de avaliação por especialistas do campo médico, deliberando-se acerca de aplicabilidades das novas informações a protocolos de reabilitação ou outras abordagens de tratamento já postas em prática.

Referências Bibliográficas

- AHA, D.; KIBLER, D. (1991). Instance-based learning algorithms. *Machine Learning*, v.6p.37–66.
- ANAND, S.; BUCHNER, A. (1998). *Decision support using data mining*. Financial Times Management, London.
- ANDERSON, D. R.; BURNHAM, K. P. (1999). General strategies for the analysis of ringing data. *Bird Study*, v.46, n.sup001, p.S261–S270.
- ANDRADE, K.; FERNANDES, G.; CAURIN, G.; SIQUEIRA, A.; ROMERO, R.; DE PEREIRA, R. (2014). Dynamic Player Modelling in Serious Games Applied to Rehabilitation Robotics. In: *Robotics: SBR-LARS Robotics Symposium and Robocontrol (SBR LARS Robocontrol)*, 2014 Joint Conference on. 2014, p. 211–216.
- ANDRADE, K. O.; FERNANDES, G.; MARTINS JR, J.; ROMA, V. C.; JOAQUIM, R. C.; CAURIN, G. A. P. (2013). Rehabilitation robotics and serious games: An initial architecture for simultaneous players. In: *Biosignals and Biorobotics Conference (BRC)*, 2013 ISSNIP. 2013, p. 1–6.
- APPEL, V. C. R.; BELINI, V. L.; JONG, D. H.; MAGALHAES, D. V.; CAURIN, G. A. P. (2014). Classifying emotions in rehabilitation robotics based on facial skin temperature. In: *Biomedical Robotics and Biomechatronics (2014 5th IEEE RAS EMBS International Conference on)*. 2014, p. 276–280.
- BARTKE, K. 2D, 3D and High-Dimensional Data and Information Visualization. http://archiv.iwi.uni-hannover.de/lv/seminar_ss05/bartke/Assets/Paper.pdf (2005). Acesso em 25 mar. 2015.
- BOIAN, R.; DEUTSCH, J.; LEE, C. S.; BURDEA, G.; LEWIS, J. (2003). Haptic effects for virtual reality-based post-stroke rehabilitation. In: *Haptic Interfaces for Virtual Environment*

- and Teleoperator Systems, 2003. HAPTICS 2003. Proceedings. 11th Symposium on.* 2003, p. 247–253.
- BREIMAN, L. (2001). Random Forests. *Mach. Learn.*, v.45, n.1, p.5–32, October.
- BURKE, J.; MCNEILL, M.; CHARLES, D.; MORROW, P.; CROSBIE, J.; MCDONOUGH, S. (2009). Optimising engagement for stroke rehabilitation using serious games. *The Visual Computer*, v.25, n.12, p.1085–1099.
- CABENA, P.; HADJINIAN, P.; STADLER, R.; VERHEES, J.; ZANASI, A. (1998). *Discovering Data Mining: From Concepts to Implementation*. Prentice Hall.
- CASADIO, M.; SANGUINETI, V.; MORASSO, P. G.; ARRICHIELLO, V. (2006). Braccio di Ferro: a new haptic workstation for neuromotor rehabilitation. *Technology And Health Care: Official Journal Of The European Society For Engineering And Medicine*, v.14, n.3, p.123 – 142.
- CATCHPOOLE, D. R.; KENNEDY, P.; SKILLICORN, D. B.; SIMOFF, S. (2010). The curse of dimensionality: a blessing to personalized medicine. *J. Clin. Oncol.*, v.28, n.34, p.e723–724, Dec.
- CHAPMAN, P.; CLINTON, J.; KERBER, R.; KHABAZA, T.; REINARTZ, T.; SHEARER, C.; WIRTH, R. CRISP-DM 1.0: Step-by-step data mining guide. <http://the-modeling-agency.com/crisp-dm.pdf> (2000). Acesso em 4 abr. 2015.
- CHEMUTURI, R.; AMIRABDOLLAHIAN, F.; DAUTENHAHN, K. (2013). Adaptive training algorithm for robot-assisted upper-arm rehabilitation, applicable to individualised and therapeutic human-robot interaction. *Journal of NeuroEngineering and Rehabilitation*, v.10, n.1, p.102.
- CIOŚ, K.; TERESINSKA, A.; KONIECZNA, S.; POTOCKA, J.; SHARMA, S. (2000). A knowledge discovery approach to diagnosing myocardial perfusion. *Engineering in Medicine and Biology Magazine, IEEE*, v.19, n.4, p.17–25, July.
- CIOŚ, K. J.; KURGAN, L. A. *Advances Techniques in Knowledge Discovery and Data Mining*, chapter Trends in Data Mining and Knowledge Discovery, p. 1–26. Springer (2005).
- CIOŚ, K. J.; PEDRYCZ, W.; SWINIARSKI, R. W.; KURGAN, L. A. (2007). *Data Mining: A Knowledge Discovery Approach*. Springer.

-
- COLOMBO, R.; STERPI, I.; MAZZONE, A.; DELCONTE, C.; PISANO, F. (2012). Taking a Lesson From Patients' Recovery Strategies to Optimize Training During Robot-Aided Rehabilitation. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, v.20, n.3, p.276–285, May.
- COVER, T.; HART, P. (1967). Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, v.13, n.1, p.21–27, January.
- CYBENKO, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, v.2, n.4, p.303–314.
- DEPT. OF VETERANS AFFAIRS; DEPT. OF DEFENSE. VA/DoD Clinical Practice Guideline for the Management of Stroke Rehabilitation, Guideline Summary. <http://www.rehab.research.va.gov/jour/10/479/pdf/VADODclinicalGuidelines479.pdf> (2010). Acesso em 12 fev. 2015.
- DNIT. Anuário Estatístico das Rodovias Federais. <http://www.dnit.gov.br/rodovias/operacoes-rodoviarias/estatisticas-de-acidentes/anuario-2010.pdf> (2010). Acesso em 9 fev. 2015.
- FAYYAD, U.; GRINSTEIN, G. G.; WIERSE, A. (2002). *Information Visualization in Data Mining and Knowledge Discovery*. Morgan Kaufmann, San Francisco.
- FAYYAD, U.; PIATETSKY-SHAPIO, G.; SMYTH, P. (1996). From Data Mining to Knowledge Discovery in Databases. *AI Magazine*, v.17p.37–54.
- FAYYAD, U. M.; PIATETSKY-SHAPIO, G.; SMYTH, P. (1996). Knowledge discovery and data mining: Towards a unifying framework. In: *Proceedings of the 2nd international conference on Knowledge Discovery and Data mining (KDD'96)*. 1996, p. 82–88. AAAI Press.
- FISHER, R. A. (1938). The Statistical Utilization of Multiple Measurements. *Annals of Eugenics*, , n.8, p.376–386.
- FIX, E.; HODGES, J. L. (1951). Discriminatory analysis, nonparametric discrimination: Consistency properties. *US Air Force School of Aviation Medicine*, v.Technical Report 4, n.3, p.477+, January.
- FRANK, E.; HALL, M. A. (2003). Visualizing Class Probability Estimators. In: *Knowledge Discovery in Databases: PKDD 2003, 7th European Conference on Principles and Practice of Knowledge Discovery in Databases, Cavtat-Dubrovnik, Croatia, September 22-26, 2003, Proceedings*. 2003, p. 168–179.

- GOLDSCHMIDT, R.; PASSOS, E. (2005). *Data Mining: Um Guia Prático*. Campus, Rio de Janeiro.
- HAWKINS, D. M. (1980). *Identification of outliers*. Monographs on applied probability and statistics. Chapman and Hall, London.
- HAYKIN, S. (2001). *Redes neurais: Princípios e prática*. Bookman, 2 edition.
- HOGAN, N.; KREBS, H.; SHARON, A.; CHARNNARONG, J. Interactive robotic therapist (1995).
- HOLZINGER, A. (2011). *Successful Management of Research and Development*. BoD, Norderstedt.
- HOLZINGER, A.; DEHMER, M.; JURISICA, I. (2014). Knowledge Discovery and interactive Data Mining in Bioinformatics–State-of-the-Art, future challenges and research directions. *BMC Bioinformatics*, v.15 Suppl 6p.I1.
- HORAK, F. B. (1990). Assumptions Underlying Motor Control for Neurologic Rehabilitation. In: *Contemporary Management of Motor Problems*. 1990, p. 11–27. Chapter 4.
- KAEHLING, L. P.; LITTMAN, M. L.; MOORE, A. W. (1996). Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, v.4, n.1, p.237–285, May.
- KOHAVER, R. (1995). A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection. In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*. 1995, IJCAI'95, San Francisco, CA, USA, p. 1137–1143. Morgan Kaufmann Publishers Inc.
- KOHOEN, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, v.43, n.1, p.59–69.
- KOHOEN, T. (1984). *Self-Organization and Associative Memory*. Springer, Berlin; New York, first edition.
- KREBS, H. I.; DIPIETRO, L.; LEVY-TZEDEK, S.; FASOLI, S.; RYKMAN-BERLAND, A.; ZIPSE, J.; FAWCETT, J.; STEIN, J.; POIZNER, H.; LO, A.; VOLPE, B.; HOGAN, N. (2008). A paradigm shift for rehabilitation robotics. *Engineering in Medicine and Biology Magazine, IEEE*, v.27, n.4, p.61–70, July.

-
- KREBS, H. I.; HOGAN, N.; AISEN, M. L.; VOLPE, B. T. (1998). Robot-Aided Neurorehabilitation. *IEEE transactions on rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, v.6, n.1, p.75–87, March.
- KREBS, H. I.; KRAMS, M.; AGRAFIOTIS, D. K.; DIBERNARDO, A.; CHAVEZ, J. C.; LITTMAN, G. S.; YANG, E.; BYTTEBIER, G.; DIPIETRO, L.; RYKMAN, A.; MCARTHUR, K.; HAJJAR, K.; LEES, K. R.; VOLPE, B. T. (2014). Robotic measurement of arm movements after stroke establishes biomarkers of motor recovery. *Stroke*, v.45, n.1, p.200–204, Jan.
- KREBS, H. I.; PALAZZOLO, J. J.; DIPIETRO, L.; FERRARO, M.; KROL, J.; RANNEKLEIV, K.; VOLPE, B. T.; HOGAN, N. (2003). Rehabilitation Robotics: Performance-Based Progressive Robot-Assisted Therapy. *Autonomous Robots*, v.15, n.1, p.7–20.
- KREBS, H. I.; VOLPE, B.; HOGAN, N. (2009). A working model of stroke recovery from rehabilitation robotics practitioners. *J Neuroeng Rehabil*, v.6p.6.
- KUNCHEVA, L. I. (2004). *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience.
- KURGAN, L. A.; MUSILEK, P. (2006). A Survey of Knowledge Discovery and Data Mining Process Models. *Knowl. Eng. Rev.*, v.21, n.1, p.1–24, March.
- MARBÁN, O.; MARISCAL, G.; SEGOVIA, J. *Data Mining and Knowledge Discovery in Real Life Applications*, chapter A Data Mining & Knowledge Discovery Process Model, p. 1–16. I-Tech Education and Publishing (2009).
- MCCULLOCH, W. S.; PITTS, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, v.5p.115–133.
- MILLER, E. L.; MURRAY, L.; RICHARDS, L.; ZOROWITZ, R. D.; BAKAS, T.; CLARK, P.; BILLINGER, S. A. (2010). Comprehensive Overview of Nursing and Interdisciplinary Rehabilitation Care of the Stroke Patient: A Scientific Statement From the American Heart Association. *Stroke*, v.41, n.10, p.2402–2448.
- MILLER, R. G. Jackknife - A Review. <http://www.stat.cmu.edu/~fienberg/Statistics36-756/jackknife.pdf> (2006). Acesso em 29 abr. 2015.
- MITCHELL, T. M. The Discipline of Machine Learning. <http://www.cs.cmu.edu/~tom/pubs/MachineLearning.pdf> (2006). Acesso em 11 jan. 2016.

- MORETTI, C. B.; ANDRADE, K. O.; CAURIN, G. A. P. (2013). Physiotherapy support web-based system for rehabilitation robotics: an initial architecture. In: *22nd. International Congress of Mechanical Engineering (COBEM 2013)*. 2013, p. 1171–1180.
- MORETTI, C. B.; JOAQUIM, R. C.; CAURIN, G. A.; KREBS, H. I.; MARTINS, J. (2014). Knowledge discovery, rehabilitation robotics, and serious games: Examining training data. In: *Biomedical Robotics and Biomechatronics (2014 5th IEEE RAS EMBS International Conference on)*. 2014, p. 567–572.
- MÜLLER, H.; FREYTAG, J. Problems, Methods and Challenges in Comprehensive Data Cleansing. Technical Report HUB-IB-164, Humboldt-Universität zu Berlin, Institut für Informatik (2003).
- NATIONAL STROKE ASSOCIATION. Hemiparesis: Muscle Weakness after Stroke. http://www.stroke.org/sites/default/files/resources/NSA_Hemiparesis_brochure.pdf (2006). Acesso em 20 mai. 2015.
- NEAR. Núcleo de Estudos Avançados em Reabilitação. http://www2.fm.usp.br/nap_near/index.php. Acesso em 5 mai. 2015.
- PASTRELLO, C.; PASINI, E.; KOTLYAR, M.; OTASEK, D.; WONG, S.; SANGRAR, W.; RAHMATI, S.; JURISICA, I. (2014). Integration, visualization and analysis of human interactome. *Biochemical and Biophysical Research Communication*, v.445, n.4, p.757–773, Mar.
- PRESSMAN, R. (2005). *Software Engineering: A Practitioner's Approach*. McGraw Hill, New York.
- QUINLAN, J. R. (1986). Induction of Decision Trees. *Machine Learning*, v.1p.81–106.
- QUINLAN, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- REGO, P.; MOREIRA, P. M.; REIS, L. P. (2010). Serious games for rehabilitation: A survey and a classification towards a taxonomy. In: *Information Systems and Technologies (CISTI), 2010 5th Iberian Conference on*. 2010, p. 1–6.
- ROSENBLATT, F. (1958). The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review*, v.65, n.6, p.386–408.

-
- RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1. chapter Learning Internal Representations by Error Propagation, p. 318–362. MIT Press, Cambridge, MA, USA (1986).
- SANCHES, A.; CARDOSO, J.; DELBEM, A. (2011). Identifying Merge-Beneficial Software Kernels for Hardware Implementation. In: *Reconfigurable Computing and FPGAs (ReConFig)*, 2011 International Conference on. 2011, p. 74–79.
- SANTOS, W. M. (2013). Desenvolvimento de um dispositivo robótico interativo para reabilitação de lesões da articulação do joelho. Dissertação (Mestrado), Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos.
- SCHÖNAUER, C.; PINTARIC, T.; KAUFMANN, H. (2011). Full Body Interaction for Serious Games in Motor Rehabilitation. In: *Proceedings of the 2Nd Augmented Human International Conference*. 2011, AH '11, New York, NY, USA, p. 4:1–4:8. ACM.
- SHEARER, C. (2000). The CRISP-DM Model: The New Blueprint for Data Mining. *Journal of Data Warehousing*, v.5, n.4, p.13–19.
- SILVA, I. N.; SPATTI, D. H.; FLAUZINO, R. A. (2010). *Redes neurais artificiais: para engenharia e ciências aplicadas*. São Paulo.
- SMITH, L. A tutorial on Principal Component Analysis. http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf (2002). Acesso em 03 mai. 2015.
- THE HOLZINGER GROUP. HCI-KDD - The Holzinger Group. <http://hci-kdd.org/>. Acesso em 1 abr. 2015.
- THE INTERNET STROKE CENTER. Acute Infarction. <http://www.strokecenter.org/professionals/stroke-diagnosis/neuropathology-image-library/acute-infarction/> (2011). Acesso em 20 mai. 2015.
- THEODOROS, D.; RUSSELL, T. (2008). Telerehabilitation: current perspectives. *Stud Health Technol Inform*, v.131p.191–209.
- LINDE, R. V. D.; LAMMERTSE, P. (2003). HapticMaster - a generic force controlled robot for human interaction. *Industrial Robot: An International Journal*, v.30, n.6, p.515–524.
- VAPNIK, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA.

- WARE, C. (2004). *Information Visualization: Perception for Design*. Morgan Kaufmann, San Francisco, 2nd edition.
- WHO. World Health Statistics. http://www.who.int/gho/publications/world_health_statistics/EN_WHS2012_Full.pdf (2012). Acesso em 9 fev. 2015.
- WINTERS, J. M.; WANG, Y.; WINTERS, J. M. (2003). Wearable sensors and telerehabilitation. *IEEE Eng Med Biol Mag*, v.22, n.3, p.56–65.
- WITTEN, I. H.; FRANK, E.; HALL, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition.
- ZARIFFA, J.; KAPADIA, N.; KRAMER, J.; TAYLOR, P.; ALIZADEH-MEGHRAZI, M.; ZIVANOVIC, V.; ALBISSER, U.; WILLMS, R.; TOWNSON, A.; CURT, A.; POPOVIC, M.; STEEVES, J. (2012). Relationship Between Clinical Assessments of Function and Measurements From an Upper-Limb Robotic Rehabilitation Device in Cervical Spinal Cord Injury. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, v.20, n.3, p.341–350, May.
- ZHU, X.; GOLDBERG, A. (2009). *Introduction to semi-supervised learning*. San Rafael, Calif. Morgan & Claypool.
- ZYDA, M. (2005). From visual simulation to virtual reality to games. *Computer*, v.38, n.9, p.25–32, Sept.