Revista de Estudos Politécnicos
Polytechnical Studies Review
2009, Vol VII, nº 12, 099-118
ISSN: 1645-9911

Data Mining via Redes Neuronais Artificiais e Máquinas de Vectores de Suporte

Armando Cruz; Paulo Cortez
cruz.armando@clix.pt, pcortez@dsi.uminho.pt

(recebido em 7 de Novembro de 2008; aceite em 5 de Setembro de 2009)

Resumo. Este artigo pretende esclarecer quais as vantagens de dois modelos não lineares de Data Mining: as Redes Neuronais Artificiais (RNA) e as Máquinas de Vectores de Suporte (MVS). Em particular, pretende-se medir o desempenho destas técnicas quando aplicadas a tarefas de classificação e regressão, comparando-as com outras técnicas (i.e. Árvores de Decisão/Regressão). Assim, fez-se uma análise de ferramentas de software que implementam os modelos referidos, tendo-se escolhido duas aplicações de utilização livre (i.e. o ambiente R e o Weka) para conduzir as experiências efectuadas. Foram utilizados diversos problemas do mundo real, sendo que os resultados obtidos revelam que as MVS obtêm em geral melhores previsões, sendo seguidas pelas RNA.

Palavras-chave: Descoberta de Conhecimento em Bases de Dados, *Data Mining*, Redes Neuronais Artificiais, Máquinas de Vectores de Suporte, Classificação, Regressão.

Abstract. This paper pretends to infer about the advantages of two nonlinear Data Mining models: Artificial Neural Networks (ANN) and Support Vector Machines (SVM). In particular, the intention is to measure their performance when applied to classification and regression tasks, being compared with other techniques (i.e. Decision/Regression Trees). Thus, an analysis was performed over a wide range of software tools that implement the referred models. From this set, two open-source applications (i.e. R environment and Weka) where selected to conduct the experiments. Several real world problems where used as benchmarks. The results show that in general the SVM achieves better forecasts, followed by the ANN.

Keywords: Knowledge Discovery from Databases, Data Mining, Artificial Neural Networks, Support Vector Machines, Classification, Regression.

1. Introdução

O interesse nas áreas da **Descoberta de Conhecimento em Bases de Dados (DCBD)** e de ***Data Mining* (DM*)*** emergiu devido aos avanços das Tecnologias de Informação e Comunicação (TIC), sendo que hoje em dia é fácil colectar, armazenar, processar e partilhar dados. Assim, tem-se assistido a um crescimento exponencial da quantidade de dados armazenados, sendo que muitos destes dados contêm informação valiosa (padrões ou tendências) que podem auxiliar a tomada de decisão (Turban et al., 2007). Este enorme volume e/ou complexidade de dados condiciona a extracção de conhecimento via técnicas estatísticas clássicas, bem como por seres humanos. Daí que a alternativa seja o uso de ferramentas de descoberta automática, ou de DM, que permitem ultrapassar estas limitações, transformando dados em bruto em conhecimento de alto nível, de forma a auxiliar a tomada de decisão.

A DCBD pode ser definida como o processo de identificar padrões e/ou modelos, a partir de dados em bruto, que sejam novos, potencialmente úteis e compreensíveis (Fayyad et al., 1996). Trata-se de um processo interactivo e iterativo, no qual se distinguem etapas como: definição do domínio de aplicação; criação de um conjunto de dados alvo; limpeza e pré-processamento de dados; redução e projecção de dados; escolha e aplicação do método/algoritmo de DM mais adequado; interpretação dos padrões obtidos; e utilização e/ou documentação do conhecimento obtido. Assim, o DM refere-se à aplicação de algoritmos com vista à descoberta de padrões em dados já pré-processados, sendo que se estime que esta etapa ocupe só uma pequena parte (em termos de esforço) do processo de DCBD (10 a 15%). De referir que o termo DM é mais simples e sonante que DCBD, sendo que na prática ambos termos tendem a ser utilizados como sinónimos.

Existem diversos algoritmos de DM, cada um com as suas vantagens e desvantagens, sendo que a distinção é baseada em dois factores essenciais: a **estrutura de representação** (i.e. a forma do modelo) e o **método de optimização** (ou seja, como se ajustam os parâmetros internos do modelo aos dados). As **Redes Neuronais Artificiais (RNA)**, modelos conexionistas inspirados no funcionamento do cérebro humano, são uma técnica popular de DM, tendo sido utilizadas com ênfase crescente a partir de 1986, quando foi proposto o algoritmo de treino de retropropagação (Rumelhart et al., 1986). Por sua vez, as **Máquinas de Vectores de Suporte (MVS)**[1]**,** são mais recentes (Vapnik, 1995). Quer as RNA quer as MVS permitem efectuar uma modelação não linear de dados. Contudo, é apontada uma vantagem teórica às MVS em relação às RNA, pois existe uma garantia da obtenção de uma solução óptima. De facto, as MVS são consideradas um dos modelos de topo na área do DM (Wu et al., 2008).

[1]Do inglês *Support Vector Machines (SVM)*.

Pretende-se então, neste artigo esclarecer as vantagens e desvantagens das RNA e MVS, quando aplicadas a tarefas de DM, comparando-as com outras técnicas (e.g. Árvores de Decisão/Regressão). Em particular, será dado um maior destaque à capacidade de previsão dos modelos.

2. Análise de Ferramentas de *Data Mining*

2.1 Perspectivas de Caracterização

No desenvolvimento de aplicações de *software* terão necessariamente que ser levados em consideração factores de decisão como o domínio da aplicação ou o âmbito da aplicação, a linguagem de programação a ser utilizada, a plataforma de sistema operativo em que funcionará, etc. Assim, as aplicações podem ser classificadas em diversas perspectivas, consoante as suas características técnicas e não só. Santos e Azevedo (2006) apontam várias possibilidades de caracterização. Começam por apontar a **linguagem de programação** a ser utilizada, e depois a **plataforma de sistema**, realçando que aplicações multi-plataforma são mais vantajosas. Também se pode caracterizar uma ferramenta pela sua: **escalabilidade**, **portabilidade**, **estado de desenvolvimento** e pela possibilidade de **integração** com outras aplicações. Para além destas características, são referidas ainda outras de natureza menos técnica como o **tipo de licenciamento** (*freeware*, *shareware*, *General Public Licence* – ***GNU*** ou licença comercial), e o **tipo de aplicação**, distinguindo as aplicações de carácter académico (desenvolvidas com o intuito de investigação e criação de novas soluções e de protótipos), e as aplicações comerciais (mais orientadas para o suporte empresarial e a prestação de serviços).

Por sua vez, Goebel et al. (1999), apresentam um esquema de caracterização em três grupos: características gerais, conectividade a bases de dados e características de DM. O grupo **características gerais** contém factores tais como: o estado de desenvolvimento do produto; o tipo de licenciamento; a disponibilidade ou não de uma versão de demonstração (*Demo*); as arquitecturas suportadas (*stand alone*, *client/server* ou *parallel processing*); e os sistemas operativos para os quais a aplicação é disponibilizada. Na **conectividade a bases de dados** estão englobados: os formatos de dados reconhecidos pela aplicação; o tipo de conexão (*online, offline*), o número máximo de instâncias suportadas; o tipo de modelo de dados (relacional, orientado a objectos, em tabela); os tipos de atributos suportados (contínuos, discretos ou simbólicos); e os tipos de *queries*, característica esta relacionada com a interface (e.g. *Structered Query Language* - SQL, *Graphical User Interface* – GUI ou linguagem específica da aplicação). Finalmente, nas **características de DM**, incluem-se: as tarefas de descoberta, tais como pré-processamento, previsão, classificação, associação, segmentação, visualização e

análise exploratória; a metodologia de descoberta, referindo-se às técnicas disponibilizadas (RNA, Árvores de Decisão/Regressão, etc.); e a interacção humana. Neste último factor, pretende-se medir qual o grau (maior ou menor) de necessidade de intervenção humana no processo (e.g. autónoma, guiada ou interactiva).

Mais recentemente, King (2004) definiu cinco categorias de características de *software* de DM: **capacidade**, que caracteriza e classifica o que uma ferramenta pode fazer; **facilidade de aprendizagem/utilização**; **interoperabilidade,** que caracteriza a possibilidade de integração com outras aplicações; **flexibilidade** para caracterizar as possibilidades de alteração de parâmetros críticos da ferramenta ao longo do processo; e a **precisão**.

Neste trabalho será utilizado uma caracterização semelhante à proposta por Goebel et al. (1999), mas com algumas alterações. A primeira diz respeito ao primeiro grupo, **características gerais**, ao qual será acrescentado o *site* onde se pode encontrar a ferramenta. A segunda adaptação é no segundo grupo, **conectividade a bases de dados**, que englobará apenas os formatos de dados reconhecidos pela aplicação. A última alteração corresponde à divisão do último grupo, **características de DM**, em dois grupos: **objectivos de DM**, que engloba diversos tipos de tarefas que a ferramenta disponibiliza (e.g. classificação ou regressão), e **técnicas de DM**, englobando os modelos de DM implementados.

2.2 Ferramentas de *DM* mais utilizadas

Santos e Azevedo (2005) fazem a caracterização de uma série de ferramentas consideradas das mais utilizadas, embora neste estudo não seja referido um critério de selecção nem um suporte à sua escolha. Por outro lado, Goebel et al. (1999) não referem critérios de selecção, embora apresente diversos critérios de exclusão de ferramentas, tais como:

- funciona como servidor de informação para outras ferramentas de DM;
- embora possua alguma possibilidade de ser usado para DM não foi desenvolvido especificamente para tal (e.g. *Matlab*);
- designado de DM, embora só sirva como ferramenta de visualização (e.g. *Oracle Discoverer*); e
- disponibilizado por companhias de consultadoria mas não constituindo produtos de aplicação geral.

King (2004) reduz a selecção a catorze ferramentas por serem as que foram disponibilizadas pelos fornecedores para a sua investigação e por usarem uma das seguintes técnicas: Árvores de Decisão/Regressão, Indução de Regras, RNA ou Redes Polinomiais.

No sítio www.kdnuggets.com[2] encontram-se os resultados de um inquérito alargado a utilizadores de DCBD/DM, onde a questão principal era: "qual a ferramenta de *DM*/analítica que utilizou em 2006?"[3]. Embora não tenha o rigor de um estudo estatístico (pode até ter sido adulterado por companhias fornecedoras de *software*, e isso é reconhecido no portal), foi considerado que seria interessante analisar os resultados obtidos (Figura 1), constatando-se que a maioria das ferramentas são referenciadas nos estudos anteriores (e.g. *Clementine*). No entanto, existem diversas novas aplicações (e.g. *Equbits*), não havendo, por isso, um consenso.

2.3 Critérios de Selecção das Ferramentas

Devido à falta de estudos com rigor científico sobre quais as ferramentas mais utilizadas, optou-se por elaborar uma lista de todas as ferramentas conhecidas, sem excepções, utilizando como referências: Santos e Azevedo (2005), Goebel et al. (1999), King (2004), www.kdnuggets.com e www.the-data-mine.com. Ao todo, a lista contém um total de 159 ferramentas. Convém referir que este não é um número inesperado dada a importância atribuída actualmente à área da DCBD/DM.

Como uma análise exaustiva de todas estas ferramentas se encontra fora do âmbito deste trabalho, optou-se por excluir ferramentas que não tivessem ou RNA ou MVS. Com este critério, a lista inicial foi reduzida para 36 ferramentas. Estas ferramentas serão caracterizadas na próxima secção segundo a perspectiva anteriormente referida.

2.4 Caracterização das Ferramentas

Em seguida, é feita a caracterização das ferramentas seleccionadas recorrendo a tabelas e gráficos. Nas tabelas de caracterização de ferramentas (Tabelas 1 e 2), caracterizam-se as ferramentas seleccionadas para estudo segundo as **características gerais, o tipo de Sistema Operativo (Tipo de SO), e a conectividade a bases de dados**. As características gerais são as seguintes:

- **Versão:** final (F) ou beta (B);
- **Licença:** comercial (C), *freeware* e *shareware* (F) ou pública (P);

2Trata-se de um portal que agrega informação a nível mundial relacionada com a DCBD/DM, incluindo software, casos de estudo, notícias, sondagens, etc.

3 Este estudo foi iniciado em 2007, sendo que nessa altura somente estava disponível o inquérito de 2006.

- **Disponibilidade:** se é ou não disponibilizada uma versão de demonstração (*Demo*) ou a ferramenta é totalmente operacional para *download* (*Download*);
- **Aplicação:** académica (A) ou comercial (C); e
- **Arquitectura:** *Stand alone* (S), *Client/Server* (C/S) ou Processamento Paralelo (PP).

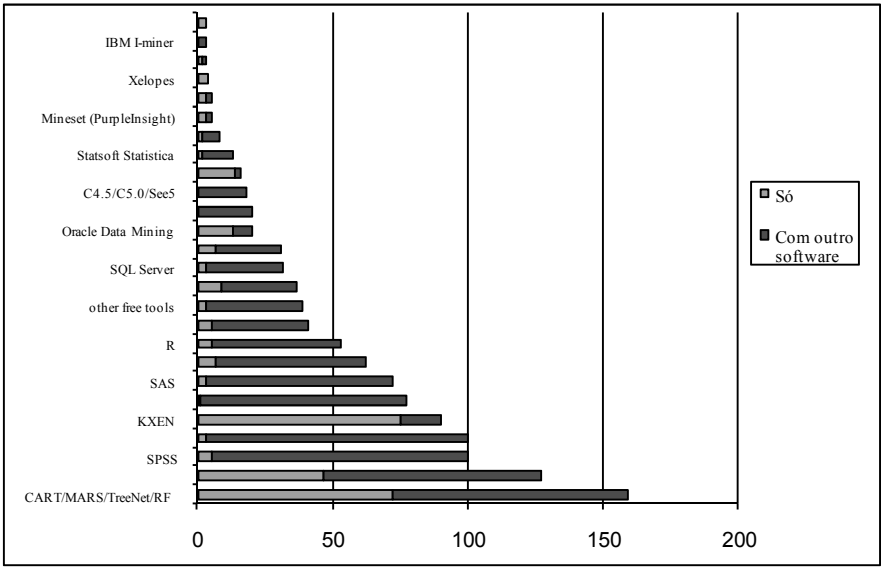
A Tabela 1 classifica, também, as ferramentas segundo o sistema operativo suportado, podendo ser o *Windows*, *Unix* (inclui também o Linux), *Mac OS*, ou *Outro* (qualquer outra plataforma de suporte da ferramenta). Convém referir que não serão distinguidas as versões dos sistemas operativos. Esta tabela mostra ainda, a **conectividade a bases de dados** das ferramentas. O tipo *ASCII* inclui vários formatos próprios das ferramentas e ficheiros de dados separados por tabulações (*tab*) ou por vírgula (também conhecido por formato CSV). Os outros tipos correspondem aos mais representativos, havendo o tipo *Outros* que inclui formatos de bases de dados menos utilizados. Na Tabela 2 caracterizam-se as ferramentas segundo as técnicas de DM disponibilizadas pelas ferramentas. As RNA são classificadas em *Multilayer Perceptrons* (MLP), *Radial Basis Functions* (RBF), *Self Organizating Maps* (SOM) e uma classe (NN) que engloba outros tipos de redes neuronais e ferramentas cujo tipo de rede não é especificado. A caracterização das ferramentas engloba também as técnicas de MVS (SVM) e Árvores de Decisão/Regressão. Ainda na Tabela 2 especificam-se as ferramentas segundo os objectivos de DM implementados: **Classificação**, **Regressão**, **Previsão**, e **Segmentação**.

Por observação directa da Tabela 1 pode verificar-se que a maioria das ferramentas é disponibilizada na versão final e numa versão *demo*, ou então, na versão totalmente funcional para *download*. No que diz respeito ao tipo de licença, verifica-se que a maioria é do tipo comercial (Figura 2).

Por observação da Figura 3 verifica-se que, embora as ferramentas de aplicação exclusiva na área académica se destaquem, as ferramentas de aplicação exclusivamente comercial e as ferramentas de aplicação mista estão em maioria. Também as ferramentas do tipo *Stand alone* se salientam por figurarem em maior número (Figura 4).

A Tabela 1 mostra que o sistema operativo que mais ferramentas suporta é o *Windows*, sendo que cerca de metade das ferramentas suportam somente esta plataforma. Tal não é surpreendente uma vez que o *Windows* domina claramente o mercado mundial dos sistemas operativos de uso pessoal. Verifica-se pelo gráfico da Figura 5 que também o *Unix* e o *Macintosh* são sistemas suportados em exclusivo por algumas ferramentas, embora tenham pouca expressão.

Figura 1 – "Qual a ferramenta de DM/analítica que utilizou em 2006?" (adaptado de www.kdnuggets.com).



Na Tabela 1 mostra-se ainda a conectividade das ferramentas, sendo esta tabela complementada pela Figura 6, onde se apresenta um gráfico com a distribuição das ferramentas pelos vários tipos de conectividade a bases de dados. Claramente o tipo mais comum é o *ASCII*, no entanto, os tipos *ODBC*, *MY SQL* e *MS Excel* também têm expressão significativa. No gráfico da Figura 7 pode-se ver a distribuição das ferramentas pelas diversas técnicas que implementam. As RNA são as mais representadas, mas o número de ferramentas que implementam apenas RNA é pouco superior ao número de ferramentas que implementam apenas MVS. No gráfico da Figura 8 está patente a distribuição dos vários tipos de RNA. Pode-se verificar que só há ferramentas a implementar exclusivamente uma técnica na classe *NN* e *SOM*, mas como para a classe *NN* não foi possível esclarecer o tipo de RNA implementado, fica em aberto a possibilidade de haver algumas que implementem exclusivamente redes do tipo *MLP* ou *RBF*. A Figura 9 apresenta um gráfico com a distribuição das técnicas pelas ferramentas analisadas. Constata-se que as ferramentas que implementam exclusivamente RNA ou MVS são quase metade. Além disso, as aplicações que implementam RNA, MVS e Árvores de Decisão/Regressão (estando assim habilitadas para fazer uma comparação proposta neste trabalho), não chegam a um quarto do número total de ferramentas.

Na Figura 10 apresenta-se a distribuição de ferramentas pelos vários objectivos. Note-se que só a **classificação** tem ferramentas exclusivas.

Tabela 1 – Caracterização das ferramentas segundo as características gerais.

Ferramenta	Características gerais						Tipo de SO				Conectividade a Base de Dados						
	Final/ Beta	Licença	Demo/ Download	Académi./ Comercial	Arquitect. (S,C/S,PP)		Windows	Unix	Mac	Outro	ASCII	Dbase	ODBC	MY SQL	MS Excel	LOTUS	Outro
Alyuda Neuro Inteligence	F	C	S	C	S		√				√						
BrainMaker	F	C	N	A/ C	S		√		√		√	√			√	√	
BSVM	F	F	S	A	S		√				√						
Clementine	F	C	N	C	S/CS		√	√			√		√		√	√	Informix, Oracle, Sybase
DTREG	F	C	S	A/ C	S		√										
EQUBITS Foresight (tm)	F	C	S	A/ C	S		√				√						
EWA Systems	F	C	N	A/ C	S/CS									√			
GhostMiner	F	C	N	A/ C	S		√				√		√	√	√		
Gist	F	F	S	A	S			√			√						
Gornik	F	C	N	C	S/CS		√	√			√		√	√	√		
Insightful Miner	F	C	S	A/ C	S/CS		√			SOLARIS	√	√	√	√	√		SAS,SPSS, Sybase
Kernel Machines	F	F	S	A	S		√										
Knowledge Miner	F	C	S	A/ C	S		√		√								
KXEN	F	C	N	C	S/CS		√							√			
LIBSVM	F	F	S	A	S		√	√			√						
MATLAB NN Toolbox	F	C	S	A	S		√	√	√	SOLARIS							
MCubiX from Diagnos	F	C	N	C	S		√				√		√	√	√		Access, freetext, imagens
MemBrain	F	F	S	A	S		√										
NeuralWorks Predict	F	C	S	C	S		√	√			√						
NeuroSolutions	F	C	S	A/ C	S/CS		√	√		SOLARIS, IRIX, AIX, Excel	√				√		Access
NeuroXL	F	C	N	C	S										√		
IPNNL Software	B	F	S	A	S		√				√						
Oracle DM	F	C	S	C	S,CS,P P		√	√		JAVA	√	√	√	√	√		Oracle, Sybase
Orange	F	F	S	A	S		√	√	√		√						
pcSVM	B	P	S	A	S			√									
R	F	P	S	A	S		√	√			√						
SAS Enterprise Miner	F	C	S	A/ C	CS		√	√		SOLARIS			√				
StarProbe	F	C	S	A/ C	S/CS		√	√	√	SOLARIS			√				JDBC
STATISTICA NN	F	C	S	A	S/CS												

Ferramenta	Características gerais					Tipo de SO				Conectividade a Base de Dados						
	Final/ Beta	Licença	Demo/ Download	Académi./ Comercial	Arquitect. (S/C/S,PP)	Windows	Unix	Mac	Outro	ASCII	Dbase	ODBC	MY SQL	MS Excel	LOTUS	Outro
SvmFu 3	B	P	S	A	S		√		SOLARIS	√						
SVM-light	F	F	S	A	S	√	√		SOLARIS	√						
TANAGRA	F	F	S	A	S	√				√						
HhinkAnalitics	F	C	N	C	CS						√		√			
Tiberius	F	C	S	A/ C	S/CS	√				√		√	√	√		
Weka	F	P	S	A	S				JAVA	√						
XLMiner	F	C	S	A/ C	S	√										

Figura 2 – Caracterização das ferramentas segundo o tipo de licença.

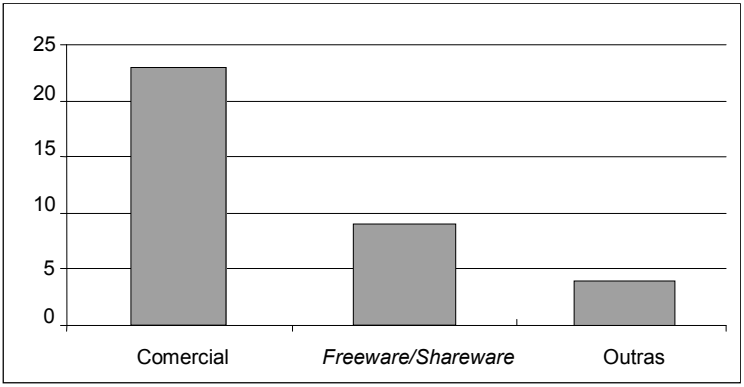


Figura 3 – Caracterização das ferramentas segundo a aplicação.

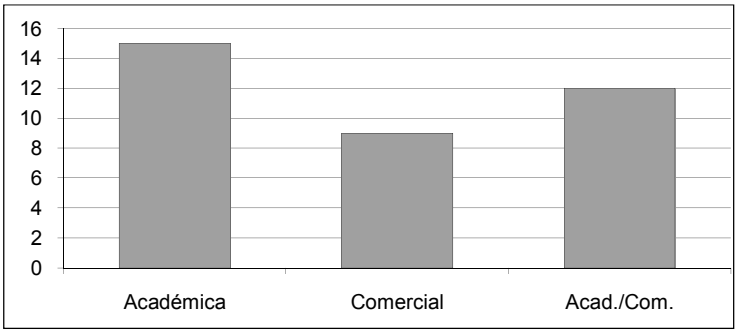


Figura 4 – Caracterização das ferramentas segundo a arquitectura.

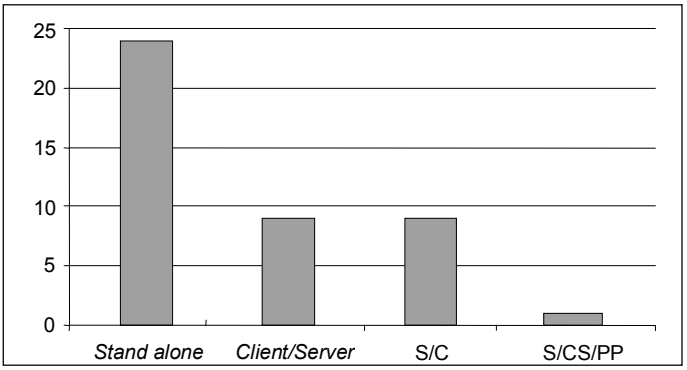


Figura 5 – Caracterização das ferramentas segundo o(s) sistema(s) operativo(s) suportado(s).

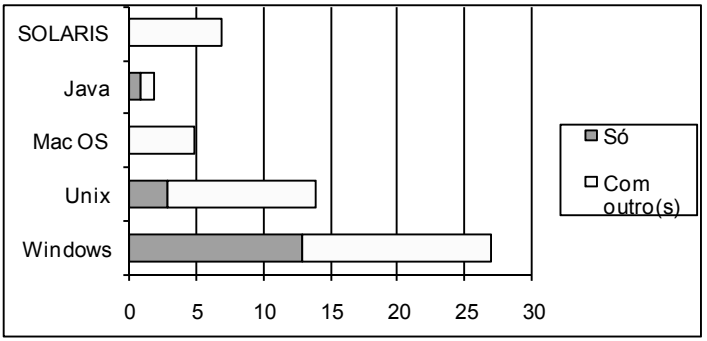


Figura 6 – Conectividade das ferramentas.

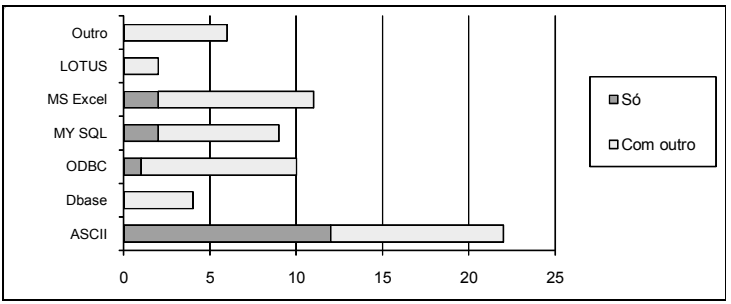


Figura 7 – Distribuição das ferramentas pelas várias técnicas que implementam

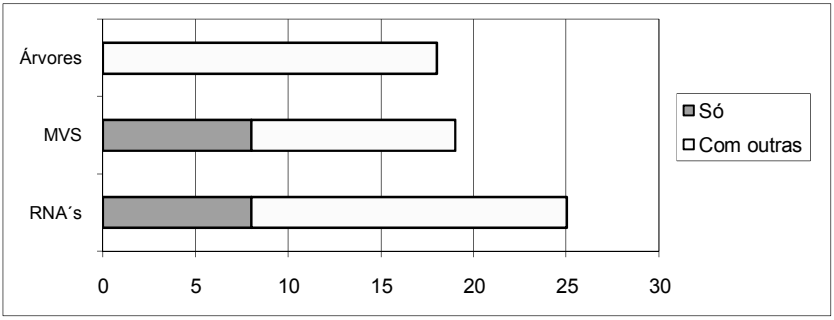


Figura 8 – Distribuição das ferramentas pelos tipos de RNA que implementam.

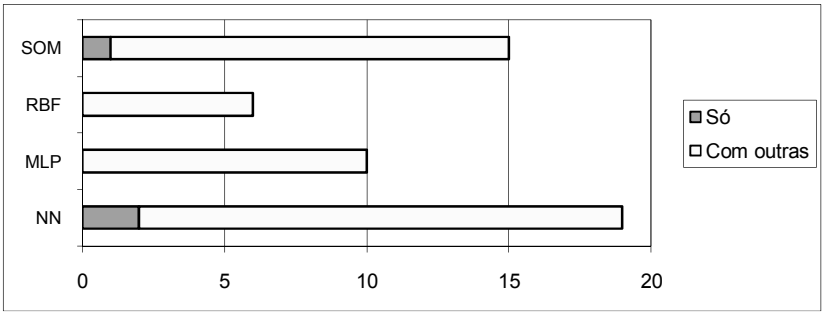


Figura 9 – Distribuição das técnicas pelas ferramentas analisadas.

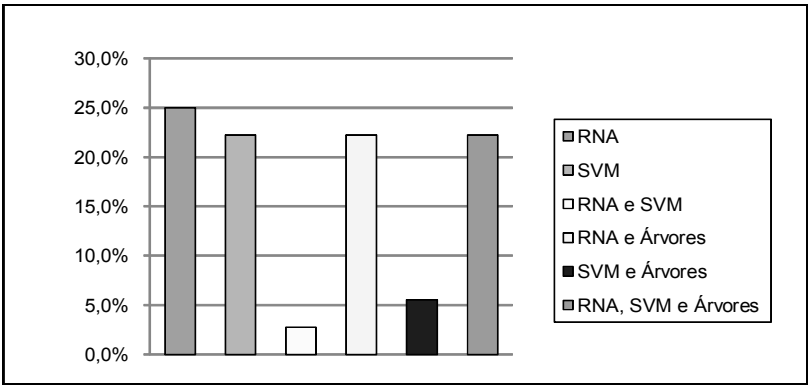


Figura 10 – Distribuição das ferramentas pelos objectivos de DM que implementam.

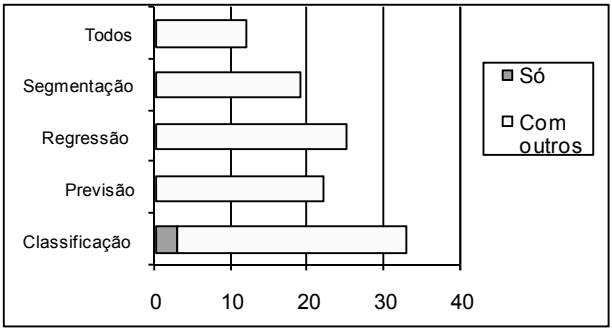


Tabela 2 – Caracterização das ferramentas segundo as técnicas de DM.

Ferramenta	Tipos de RNA			Outras Técnicas			Objectivos implementados			
	NN	MLP	RBF	SOM	SVM	Árvores	CL	PR	RG	SG
Alyuda NeuroIntelligence	√						√	√		
BrainMaker	√						√	√		
BSVM					√		√		√	
Clementine		√	√	√	√	√	√	√	√	√
DTREG					√	√	√	√	√	
EQUBITS Foresight(tm)					√		√	√	√	
EWA Systems	√			√	√	√	√		√	√
GhostMiner	√				√	√	√			√
Gist					√		√			
Gornik		√		√	√	√	√			√
Insightful Miner		√				√	√	√	√	√
Kernel Machines	√		√		√		√		√	
Knowledge Miner				√			√	√		
KXEN					√		√	√	√	√
LIBSVM					√		√		√	
MATLAB NN Toolbox		√	√	√			√	√	√	
MCubiX from Diagnos	√					√	√	√	√	
MemBrain	√	√		√						
NeuralWorks Predict	√			√				√		√
NeuroSolutions	√			√			√	√	√	√
NeuroXL	√			√			√	√	√	√
IPNNL Software		√		√			√		√	
Oracle DM					√	√	√	√	√	√
Orange	√			√	√	√	√	√	√	√
pcSVM					√		√			
R	√	√	√	√	√	√	√	√	√	√
SAS Enterprise Miner	√			√		√	√	√	√	√
StarProbe	√					√	√	√	√	√
STATISTICA NN		√				√	√	√		√
SvmFu 3					√		√			
SVM-light					√		√		√	
TANAGRA	√	√	√	√		√			√	√
HhinkAnalitics	√					√	√		√	√
Tiberius	√				√	√	√	√	√	
Weka	√	√	√	√	√	√	√	√	√	√
XLMiner	√					√	√	√	√	√

3. Experiências

Visto haver uma enorme oferta de ferramentas de DM, naturalmente tem que se proceder a uma selecção de modo a tornar viável uma comparação de técnicas. Os critérios de escolha da ferramenta passam pelo facto da mesma ter que implementar RNA, MVS e Árvores aplicáveis à classificação e à regressão[4]. A escolha recai sobre as ferramentas R e WEKA. A ferramenta R tem uma interface via linha de comandos (RCDT, 2008), enquanto que a ferramenta Weka (Witten e Frank, 2005) utiliza um ***Graphic User Interface*** **(GUI)**. Implementa as técnicas das RNA, MVS e Árvores de Decisão/Regressão, podendo estas ser aplicadas tanto à classificação como à regressão. Ambas têm uma licença não comercial (*freeware*).

3.1 Dados Utilizados

Os conjuntos de dados foram retirados do repositório UCI (Asuncion e Newman, 2007). Neste repositório público, encontram-se cerca de uma centena de dados disponibilizados por fontes governamentais e universidades (entre outras), para serem utilizados livremente em investigação. De notar que este repositório tem sido deveras utilizado, a nível mundial, pela comunidade da Aprendizagem Automática/DM para testar algoritmos (Soares, 2003). Além disso, estes dados encontram-se praticamente prontos para o processo de DM, isto é, necessitam de pouco trabalho de pré-processamento. Na escolha dos 14 conjuntos de dados (*datasets*) procurou-se escolher somente problemas que se adequassem à classificação ou à regressão. Também existiu o cuidado de escolher dados com características variadas, ou seja, com diferentes dimensões (entre menos de 200 instâncias até mais de 2000 exemplos), passando por valores em falta e atributos variados. Nos parágrafos seguintes estão descritos em pormenor os problemas que foram utilizados nas experiências. Os primeiros sete destinam-se à classificação, sendo que os restantes sete à regressão. Os problemas utilizados para classificação estão resumidos na Tabela 3. Os atributos de entrada são descritos como numéricos (num), binários (bin) e nominais (nom). Na tabela consta ainda, o número de instâncias (Nº Inst.), e o método de validação utilizado (10-*fold Cross-validation* – CV[3], ou *holdout Percentage Split* – PS[4]). A Tabela 4 resume os problemas de

4 Aqui a preocupação era escolher um algoritmo de DM que fosse popular para ter uma base de comparação com as RNA e MVS. Embora pudessem ter sido escolhido outros algoritmos (e.g. k-nearest neighbor), neste estudo escolheram-se as árvores de decisão/regressão, dado que são actualmente muito utilizadas (Rexer 2008).

5 Validação cruzada com 10 desdobramentos. Este método é mais lento do que a divisão em casos de treino/teste, pelo que será utilizado somente nos conjuntos de dados de menor dimensão.

6 Separação simples em casos de treino (66%) e teste (33%), com amostragem aleatória.

regressão, separando-os também segundo o método de validação utilizado para a estimação do desempenho do algoritmo.

Tabela 3 – Sumário dos conjuntos de dados utilizados em classificação.

Conjunto de Dados	Descrição	Atributos			Nº Inst.	Valores em Falta	Método
		Num	Bin	Nom			
Agaricus-lepiota	Toxicidade de cogumelos.	0	0	23	8124	Sim	PS
Balance-scale	Posição do ponteiro de uma balança.	4	0	1	625	Não	PS
Bupa	Perturbações do fígado.	6	0	1	345	Não	CV
House-votes-84	Votos nos congressistas.	0	6	1	435	Sim	CV
Ionosphere	Reflexões de radar na ionosfera.	4	0	1	351	Não	CV
Pima-indians	Mulheres que poderão ter diabetes.	8	0	1	768	Não	PS
Post-operartive	Decisão para onde enviar pacientes após uma operação.	0	0	9	90	Sim	CV

Tabela 4 – Sumário dos conjuntos de dados utilizados em regressão.

Conjunto de Dados	Descrição	Atributos			Nº Inst.	Valores em Falta	Método
		Num	Bin	Nom			
Abalone	Idade de moluscos.	7	0	1	4177	Não	PS
Auto-mpg	Consumo por automóveis.	8	0	1	398	Sim	CV
Autos	Preço de automóveis.	6	0	5	205	Sim	CV
Brest	Recorrência de cancro do peito.	3	0	1	198	Sim	CV
CPU	Desempenho de computadores.	7	0	2	209	Não	CV
Housing	Preço de casas em Boston.	2	1	0	506	Não	PS
Servo	Resposta de servomecanismos.	2	0	2	167	Não	CV

3.2 Classificação/Regressão com RNA e MVS no Weka

A interface utilizada para as experiências foi o *Experimenter* já que é possível escolher várias tarefas e técnicas a serem testadas numa única experiência. As experiências foram efectuadas em quatro partes, duas para classificação e duas para regressão. Em cada caso fez-se a separação segundo o método de validação (i.e.

uma parte com o 10-*fold Cross-validation* , e outra com o *holdout Percentage Split*). Cada parte, ainda, foi repetida 20 vezes (*runs*). A parametrização das técnicas utilizou os valores por omissão.

3.3 Classificação/Regressão com RNA e MVS no ambiente R

Para a realização das experiências com o R, utilizaram-se os *packages nnet*, *e1071* e *tree*. As experiências foram divididas em quatro partes, tal como no WEKA. Também foram repetidas 20 vezes. E ainda, optou-se pela utilização da parametrização por omissão. No entanto, o *package* de RNA exige que o parâmetro *size* (número de neurónios intermédios do MLP) seja definido pelo utilizador. Idealmente, este parâmetro deveria ser escolhido através de uma selecção de modelos, onde se testariam diversos valores, escolhendo-se a rede que apresentasse o melhor valor num conjunto de validação. Tal procedimento resultaria em melhores resultados de previsão, contudo exige recursos computacionais mais elevados. Dado que o Weka utiliza por omissão um valor fixo de neurónios intermédios, e para ter uma comparação mais justa, aqui optou-se também pelo uso de valor fixo: 4 neurónios intermédios. Trata-se de um valor que permite criar uma RNA de reduzida dimensão e que, por isso, é menos propícia ao fenómeno de sobre ajustamento: i.e., perda de capacidade de generalização. Após algumas experiências preliminares, nos parâmetros *range* e *decay* foram utilizados os valores de 0,2 e 5E-4.

No R, a utilização das técnicas é feita à custa de linha de comandos, tal como já foi dito. Assim, torna-se necessário criar código que faça a leitura dos dados, implemente a forma de validação (*k-fold cross-validation* ou *percentage split*), crie o modelo utilizando uma das técnicas, e faça o teste do modelo e cálculo de uma métrica sobre os resultados do teste. Assim, foi desenvolvido um código R, que implementa a validação *10-fold ou percentage split* de 66%, para as diversas técnicas utilizadas.

4. Análise dos Resultados

As Tabelas 5 e 6 resumem os resultados obtidos (média e respectivo desvio padrão) da aplicação das técnicas de DM em classificação e regressão. No primeiro caso, foi utilizado como métrica de erro a **percentagem de instâncias correctamente classificadas (PCC)**. Portanto, quanto maior for o valor do avaliador melhor é a avaliação do modelo. Quanto à regressão, foi utilizada a medida ***Root Relative Squared Error*** **(RRSE)**, em percentagem (Witten e Frank, 2005). Esta métrica compara o erro quadrático obtido pelo algoritmo com o erro obtido pela previsão simples da média dos valores. Assim, quanto menor for o seu

valor, melhor será o algoritmo. Acrescenta-se também a Tabela 7, que apresenta por técnica, o número de vezes que obteve o melhor resultado, o número de vezes que obteve o segundo melhor resultado e o número de vezes que ficou em último lugar. Para além disso, será utilizada uma análise comparativa adicional, onde se traduz em pontuação os valores obtidos na Tabela 7, ou seja, admite-se que o 1º lugar corresponde a três pontos, o 2º a dois e o 3º a um ponto. A Tabela 8 apresenta o correspondente *ranking*, onde se pode observar que as MVS obtêm o primeiro lugar, seguidas das RNA e finalmente pelas Árvores de Decisão/Regressão.

De realçar que para o mesmo algoritmo (e.g. MVS) existem por vezes diferenças entre o desempenho obtido com as ferramentas Weka e R (Tabelas 5 e 6). Tais diferenças são explicadas por duas razões. A primeira tem a ver com o facto que embora se tratem dos mesmos modelos (i.e. RNA, MVS, Árvores), cada ferramenta tem a sua própria implementação ao nível dos algoritmos de aprendizagem. Assim, é possível por exemplo que o algoritmo de árvores utilizado pelo R é baseado no CART, enquanto que o Weka utiliza o algoritmo C4.5. A segunda razão, tem a ver com a parametrização dos algoritmos. Cada modelo tem um conjunto de parâmetros (e.g. neurónios intermédios da RNA) que afectam o desempenho final, sendo que cada ferramenta tem heurísticas/procedimentos distintos para atribuir valores a esses parâmetros.

Tabela 5 – Resumo dos resultados obtidos em classificação (valores de PCC).

Conjuntos de Dados	RNA		MVS		Árvores	
	R	Weka	R	Weka	R	Weka
Agaricus	99,9	58,7	**99,9**	**63,9**	99,9	60,9
Balance-scale	**90,4**	**90,8**	90,3	87,5	68,7	77,8
Bupa	65,6	**69,1**	**66,5**	58,0	59,8	68,8
House-votes-84	95,1	94,6	**95,6**	96,0	95,1	**96,6**
Ionosphere	85,9	**90,9**	**93,1**	88,2	84,4	89,9
Pima-indians	67,6	75,6	**76,3**	**76,8**	73,4	74,3
Post-operative	60,4	54,0	**72,5**	67,7	56,8	**68,7**
Média	80,7	76,2	**84,9**	**76,9**	76,9	76,7

Tabela 6 – Resumo dos resultados obtidos em regressão (valores de RRSE).

Conjuntos de Dados	RNA		MVS		Árvores	
	R	Weka	R	Weka	R	Weka
Abalone	**66,9**	71,0	71,1	**69,4**	76,2	72,5
Auto-mpg	44,4	43,4	**41,5**	44,2	56,3	**43,0**
Autos	100,4	33,6	**100,0**	**32,9**	106,7	48,9
Brest	102,0	175,6	**95,4**	**90,0**	115,0	101,7
CPU	87,8	141,7	**47,4**	**28,1**	102,4	100,0
Housing	86,9	**50,3**	**49,9**	55,9	56,1	54,4
Servo	**46,0**	**39,9**	66,5	80,8	50,0	51,5
Média	76,3	79,4	**67,4**	**57,3**	80,4	67,4

Para cada algoritmo, também foi medido o esforço computacional, em termos de tempo de execução exigido pelo processador. Em termos médios e para ambas as ferramentas analisadas, as árvores apresentam-se como a técnica de aprendizagem mais rápida (0.1 segundos). De seguida surgem as MVS, exigindo um maior esforço computacional (8.5 segundos), sendo que em último lugar surgem as RNA (54 segundos). De facto, estas últimas revelaram uma elevada exigência computacional, especialmente para conjuntos de dados de elevada dimensão.

Tabela 7 – Ranking das técnicas.

Técnica	Classificação			Regressão		
	1º	2º	3º	1º	2º	3º
RNA	4	5	5	4	7	3
MVS	8	4	2	9	1	4
Árvores	2	5	7	1	6	7

Tabela 8 – Ranking por pontuação.

Técnica	Classificação	Regressão
RNA	27	29
MVS	**34**	**33**
Árvores	23	22

5. Conclusão

Pretendeu-se avaliar qual a qualidade, em termos de previsão, obtida por duas técnicas não lineares, as RNA e MVS, comparando-as com Árvores de Decisão/Regressão. Estas técnicas contêm diversos parâmetros, bem como variações do algoritmo de procura do modelo óptimo, que afectam o seu desempenho final, sendo que existem diversas implementações conforme o tipo de ferramenta que se utiliza. Por exemplo, a aplicação Weka contém vários algoritmos que implementam RNA ou Árvores de Decisão/Regressão. Ora, o utilizador comum (não especializado), terá dificuldades em tomar escolhas, tendendo a aceitar os parâmetros/algoritmos sugeridos por estas ferramentas. Contudo, existem largas dezenas de ferramentas de DM, sendo que cada uma apresenta um conjunto distinto de técnicas. Optou-se somente por, numa primeira fase, efectuar uma análise geral às ferramentas que disponibilizam RNA ou MVS, constatando-se que actualmente existem pelo menos 36 ferramentas com estas características. Este elevado número é um bom indicador de que há um elevado interesse no uso de RNA e MVS em aplicações de DM.

Para além desta análise geral, também se efectuaram um conjunto de experiências, tendo-se utilizado duas ferramentas: o Weka e o R. Os resultados obtidos em diversos problemas do mundo real, revelam as MVS como a melhor técnica de DM em previsão, sendo que as Árvores de Classificação/Regressão obtêm os piores resultados. No entanto, a melhoria das MVS é conseguida à custa de um maior esforço computacional, quando comparadas com as Árvores de Decisão/Regressão. Tal facto é deveras relevante, principalmente quando o domínio de aplicação der origem quantidades de dados de elevada dimensão.

Há que referir que os resultados foram obtidos com as técnicas configuradas com os valores de omissão e por isso não requerendo conhecimentos especializados por parte do utilizador. Tal facto contradiz de certo modo o argumento de que as RNA e/ou MVS são de difícil utilização. De modo algo surpreendente, os resultados obtidos também contradizem a necessidade de selecção de modelos e noção que o desempenho das RNA e MVS é mais sensível a uma correcta escolha dos seus hiper-parâmetros (e.g. número de nós internos da RNA ou parâmetros do kernel da MVS), do que no caso das Árvores de Decisão/Regressão. Contudo, há que realçar que uma cuidada selecção de modelos (não efectuada neste estudo) iria, de modo muito provável, produzir ainda melhores desempenhos para as RNA e MVS.

Este estudo foi iniciado em 2007, sendo que por isso não inclui ferramentas recentes que na altura não estavam disponíveis, como o ambiente de *open source* RapidMiner (RME, 2009). Contudo, há que referir que ambas as ferramentas testadas (Weka e R) continuam a ser populares hoje em dia, conforme pode ser verificado pelos inquéritos recentes do portal KDnuggets (2008) e da Rexer Analytics (2008). Importa também referir que existem outras limitações, nomeadamente:

- Testaram-se apenas 7 problemas de classificação e 7 tarefas de regressão, não existindo garantias que estes problemas, embora variados, correspondam ao que se espere encontrar no mundo real;
- Foram utilizados os conjuntos de dados originais conforme disponibilizados no repositório UCI, ou seja, já pré-processados, não existindo preocupações com as fases de pré-processamento (e.g. selecção de dados, transformação de variáveis, substituição de valores omissos);
- Foram somente analisadas duas métricas de avaliação das técnicas: a capacidade de previsão e o tempo. Não foram analisados outras dimensões como: facilidade de compreensão dos modelos, ou a novidade e utilidade do conhecimento adquirido.

Em termos de trabalho futuro, tencionamos abordar diversas destas limitações, tais como a análise da ferramenta RapidMiner ou a elaboração de um inquérito com rigor estatístico sobre a opinião de utilizadores comuns acerca das ferramentas de DM mais utilizadas.

Agradecimentos

Este trabalho foi suportado pelo projecto FCT PTDC/EIA/64541/2006.

Referências

Asuncion, A. and Newman, D. (2007). UCI Machine Learning Repository, University of California, Irvine, School of Information and Computer Sciences, http://archive.ics.uci.edu/ml/.

Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, S., and Uthurusamy, R. (1996). Advances in Knowledge Discovery and Data Mining, MIT Press.

Goebel M. and Gruenwald L. (1999). A Survey of Data Mining and Knowledge Discovery Software Tools, ACM SIGKDD Explorations Newsletter, 1(1):20-33, ACM.

KDnuggets (2008). KDnuggets: Polls: Data Mining Software, http://www.kdnuggets.com/polls/2008/data-mining-software-tools-used.htm, May.

King, D. (2004). Numerical Machine Learning, Technical Report CS 4803B, Georgia Technology College of Computing.

R Development Core Team (2008), R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, http://www.R-project.org, ISBN: 3-900051-07-0.

Rexer, K. (2008). Second Annual Data Miner Survey, Rexer Analytics.

Rumelhart, D., Hinton, G., Williams, R. (1986). Learning Internal Representations by Error Propagation, In Rulmelhart and McClelland (Eds.), Parallel Distributed Processing: Explorations in the Microstructures of Cognition, MIT Press, Cambridge MA, pp. 318-362.

Santos, M. F. e Azevedo, C. (2005). Data Mining, Descoberta de Conhecimento em Bases de Dados, FCA – Editora de Informática.

Soares, C. (2003), Is the UCI Repository Useful for Data Mining?, In F. Pires and S. Abreu (Eds.), Progress in Artificial Intelligence, 11th Portuguese Conference on Artificial Intelligence, Lecture Notes in Artificial Intelligence 2902, pp 209-223.

Turban, E., Sharda, R., Aronson, J., King, D (2007). Business Intelligence – A managerial approach, Pearson Prentice-Hall, NY, USA.

Vapnik V. (1995). The Nature of Statistical Learning Theory, Springer-Verlag, New York.

Witten, I. and Frank, E. (2005), Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, 2nd Ed., San Francisco, CA, Morgan Kaufmann.

Wu, X., Kumar, V., Ross, Quinlan, J., Gosh, J., Yang, Q., Motoda, H., MacLachlan, G., Ng, A., Liu, B., Yu, P., Zhou, Z., Steinbach, M., Hand, D., Steinberg, D. (2008). Top 10 algorithms in data mining, Knowledge Information Systems, 14:1-37.

RapidMiner Enterprise (2009). RapidMiner Community Edition, http://rapid-i.com/

Notas Curriculares

Armando Cruz, Licenciado em Engenharia Electrotécnica pela Universidade de Trás-os-Montes e Alto Douro, Mestre em Sistemas de Informação pela Universidade do Minho. É assistente no Instituto Politécnico de Viseu, Escola Superior de Tecnologia e Gestão de Lamego, desde 2001.

Paulo Cortez (PhD) é Professor Auxiliar do Departamento de Sistemas de Informação e investigador integrado do centro Algoritmi, Universidade do Minho, onde desenvolve actividades de investigação nas áreas do Business Intelligence, Data Mining e Inteligência Artificial. Actualmente é editor associado da revista científica Neural Processing Letters, tendo também participado em 7 projectos de I&D (dos quais foi líder em 2). Tem dezenas de publicações, incluindo artigos nas revistas Journal of Heuristics, Neurocomputing e Decision Support Systems (ver http://www3.dsi.uminho.pt/pcortez).