24 - APLICAÇÃO DE APRENDIZADO DE MÁQUINA EM DADOS MENSURADOS NUMA SEÇÃO DO RIO ATIBAIA/SP

Maria Rejane Lourençoni Siviero[(1)]
Aluna Especial Curso Aprendizado de Máquina e Mineração de Dados do Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de São Carlos. Mestrado e Doutorado em Recursos Hídricos pela FEC/UNICAMP. Pós- Doutorado pela FEAGRI/UNICAMP.
Estevam Rafael Hruschka Junior[(2)]
Prof. Dr. da Pós- Graduação da Faculdade de Ciência da Computação da UFSCar.

Endereço[(1)]**:** Rua Sebastião Vieira de Andrade, 137 - Est. Recr. San Fernando - Valinhos - SP - CEP: 13278-110 - Brasil - Tel: +55 (19) 3881-1753 - e-mail: rsiviero@hotmail.com.

RESUMO

Este artigo trata-se da aplicação de aprendizado de máquina e mineração de dados, os quais foram mensurados, realizado com algoritmos supervisionados e não supervisionados para treinamento. O banco de coleta dos dados utilizados, compõe-se de quarenta medições, realizadas no rio Atibaia/SP: vazão (Q), declividade da linha d'água (S), raio hidráulico (RH), largura do espelho d'água (B), descarga sólida transportada no leito (Gsb) e descarga sólida transportada em suspensão (Gss), nos anos de 1993 e 1994. Os algoritmos supervisionados utilizados foram: Árvore de Decisão - C4.5, Naive-Bayes, Regressão Logística e não supervisionado foi Expectation Maximization (EM). OS resultados foram obtidos mediante uso do software WEKA 3 (Waikato Environment for Knowledge Analysis) à título de comparação. Nas comparações entre os algoritmos, não foram satisfatórias, devido os anos adotados para as amostras ou não refletirem a distribuição dos dados e ou conterem 'outliers'. Por outro lado, propõem-se se for possível a ampliação do período estudado, bem como nova análise dos resultados para os mesmos algoritmos e aplicações de outros, onde a dependência condicional entre os parâmetros seja levada em consideração.

PALAVRAS-CHAVE: Aprendizado de máquina, algoritmo supervisionado, algoritmo não supervisionado.

1. INTRODUÇÃO

Aprendizado de máquina é um sub-campo da inteligência artificial, dedicado ao desenvolvimento de algoritmos e técnicas que permitam o computador aprender, isto é, que permitam o computador aperfeiçoar seu desempenho em alguma tarefa.
Segundo Russell e Norving (2004), as técnicas de inteligência artificial possuem três características principais: busca (para explorar as distintas possibilidades em problemas onde os passos não são claramente definidos), emprego do conhecimento (permite explorar a estrutura, relações do mundo ou domínio à que pertence o problema e a redução do número de possibilidades a considerar, tal como os humanos fazem) e abstração (proporciona a maneira de generalizar nos passos intrinsecamente similares).
Assim, pode-se utilizar aprendizado de máquina em bancos de dados, reconhecimento de objetos, tais como: face, fala e escrita, diagnósticos médicos, entre outros, porém não se obtém êxito de utilização em sistemas estáticos, como: armazenamento e recuperação de dados.
Embora o aprendizado de máquina seja uma ferramenta poderosa para a aquisição automática de conhecimento, deve ser observado que não existe um único algoritmo que apresente o melhor desempenho para todos os problemas (Monard e Baranauskas, 2005).
Denomina-se algoritmo de aprendizado a um conjunto de regras bem definidas para solução de um problema de aprendizado. Segundo Monard e Baranauskas (2005), é importante compreender o poder e a limitação dos diversos algoritmos de aprendizado de máquina, utilizando alguma metodologia, que permita avaliar os conceitos induzidos por esses algoritmos em determinados problemas.

2. REVISÃO BIBLIOGRÁFICA

2.1. Paradigmas de Aprendizado

De acordo com os conceitos/padrões a serem aprendidos e a disponibilidade de dados para treinamento, pode-se separar em dois tipos de aprendizado, os quais são conhecidos como paradigmas do aprendizado de máquina: aprendizado supervisionado e não supervisionado.

2.1.1.Aprendizado Supervisionado

Nesse tipo de aprendizado a função a ser aprendida, seja de classificação ou de regressão, está claramente definida, além de o algoritmo ter em sua estrutura uma espécie de instrutor indicando quando a solução está aceitável no exemplo em treinamento. O algoritmo na tarefa de classificação pode ser com saída binária (0, 1; sim, não) ou com saída multi-classes (1, 2, 3 ou 4) e na tarefa de regressão a saída é contínua (pessoa com peso variando de 50 a 80 kg).

2.1.2.Aprendizado Não Supervisionado

Nesse tipo de aprendizado a função a ser aprendida não está explícita e o algoritmo deve aprender os conceitos/padrões, os quais se referem os dados, baseados em agrupamentos e vizinhança.

2.2. Algoritmos de Aprendizado

A seguir, estão explanados os algoritmos de aprendizado de máquina utilizados para treinamento dos dados: supervisionado e não supervisionado, os quais foram empregados neste artigo.

2.2.1.Árvore de Decisão – C4.5

É um algoritmo utilizado na tarefa de classificação em aprendizado supervisionado. Trata-se de um algoritmo guloso, o qual faz uma busca ′top-down′ no espaço para todas as possíveis árvores, tendo a entropia (medida da pureza do conjunto de instâncias), utilizada no cálculo da razão de ganho (GR), o qual penaliza os atributos com muitos valores possíveis. Porém, se a amostra for pequena pode ocorrer ′overfitting′ (classificação tendenciosa), uma maneira de evitar o overfitting na árvore de decisão é a poda -′ Occam′s razor′ (Mitechell, 1997).
O maior problema da utilização de algoritmos gulosos é a possibilidade do processo de otimização ficar preso em um ótimo local (Jambeiro F., 2007).
De qualquer forma, a árvore pode ser facilmente convertida em regras do tipo: “Se ..... então ....”, o que facilita a compreensão humana. O número de folhas é igual ao número de regras e a profundidade da árvore define a quantidade de antecedentes nas regras.
A árvore de decisão não possui uma estrutura pronta, a mesma vai sendo construída ao longo do processo de aprendizado e não há tratamento da incerteza.
O aprendizado de árvore de decisão é um dos algoritmos mais utilizados devido a aplicações práticas (Bifet *et al*, 2017; Mitchell, 1997). É um método de aproximação de funções discretas, robusta a ruídos, capaz de aprender expressões disjuntivas (Mitchell, 1997).

2.2.2.Naive Bayes

É um algoritmo utilizado na tarefa de classificação em aprendizado supervisionado, freqüentemente chamado de classificador Bayes (Mitchell, 1997).
Naive Bayes é um modelo probabilístico, sendo um caso particular de rede bayesiana com inferência, onde o número de pais na rede corresponde a um nó; utiliza mecanismo de suavização de Laplace e a abordagem da probabilidade, pode dar-se: por máxima verossimilhança (ML) ou máxima posteriori (MAP). A estrutura do modelo é fixa (Figura 1), dispensa mecanismo de busca, e assume que todas as variáveis (X_1, X_2, X_3, ..., X_n) são independentes dada a classe, concepção que simplifica bastante os cálculos e reduz drasticamente o número de parâmetros a serem estimados na classificação.

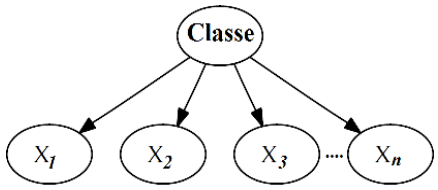

Figura 1: Estrutura do Naive Bayes

Por outro lado, essa simplificação leva à perda na estimativa da probabilidade, o qual fica tendenciosa por encontrar-se nos extremos (0 ou 1), o qual é determinada por normalização entretanto, considera-se que naive Bayes possui bom desempenho para determinar o ranking da classificação.
Segundo Jambeiro F. (2008), a hipótese de independência das variáveis dada a classe é falsa em praticamente todos os problemas de interesse prático. Apesar disso, segundo Mitchell (2010) é um algoritmo bastante utilizado, seja para variáveis discretas ou contínuas, devido ser de fácil aplicação para treinar com um conjunto de amostras.
Segundo Mitchell (1997), a performance do naive Bayes em alguns domínios tem mostrado ser comparável com aprendizado de rede neural e árvore de decisão.

2.2.3.Regressão Logistica

É um algoritmo utilizado na tarefa de classificação em aprendizado supervisionado, geralmente é referido como classificador discriminativo das variáveis.
A regressão logística assume forma paramétrica da distribuição da probabilidade direta, estimando os pesos (W_0, W_1, ... W_n) dos parâmetros dos dados de treinamento; assim, com esse procedimento vai ajustando a função que define o comportamento desses dados. A solução do espaço de busca para a regressão pode não ser necessariamente a ótima e linear, mas possui um único ponto de máximo ou mínimo e não constrói o modelo.
Segundo Mitchell (2010) overfitting nos dados de treinamento é um problema que atinge a regressão logística. Assim, uma medida para redução do overfitting é a regularização, cuja função é penalizar grandes valores dos pesos (W_i) pelo log da máxima verossimilhança.
A regressão logística assume a mesma forma do classificador Naive Bayes Gaussiano quando houver independência condicional das variáveis (Mitchell, 2010).
Por outro, segundo Mitchell (2010), quando os dados de treinamento forem em grande quantidade a Regressão Logística supera o Naive Bayes Gaussiano e neste caso pode detectar a dependência entre as variáveis, porém quando os dados forem escassos ocorre o inverso.

2.2.4.Expectation Maximization (EM)

É um algoritmo utilizado nas tarefas de agupamento em aprendizado não supervisionado.
Expectation Maximization (EM) é um método iterativo que alterna entre realizar numa etapa a expectativa (E), que calcula a expectativa pelo log da verossimilhança utilizando estimativa atual para as variáveis, e noutra etapa a maximização (M), que calcula os parâmetros de maximização da probabilidade encontrada na etapa (E). Estas estimativas de parâmetros são, então, utilizados para determinar a distribuição das variáveis na próxima etapa (E), e, assim, sucessivamente até a convergência do agrupamento, deste modo há garantia de encontrar um máximo local, porém EM não consegue identificar claramente 'outliers' (dados atípicos).
Geralmente, ocorre implementação do EM com modelos probabilísticos mais simples, devido às iterações envolvidas no processo até a convergência do agrupamento, dado que agrupamento por definição são processos iterativos, pois não possuem classe definida e necessitam de priori para a inicialização.
No entanto, pode-se utilizar EM com árvore de decisão, porém não há garantia de convergência do agrupamento e, ainda mais, que o agrupamento é o correto. Outro fato, o agrupamento em si pode ser o correto, mas não necessariamente há acerto das classes.

3. METODOLOGIA

3.1. Banco de Dados

O banco de coleta de dados utilizado compõe-se de quarenta medições fluviossedimentométricas realizadas numa seção do rio Atibaia/SP: Vazão (Q), Declividade da linha d'água (S), Raio hidráulico (RH), Largura do espelho d'água (B), Descarga sólida transportada no leito (Gsb) e Descarga sólida transportada em suspensão (Gss), nos anos de 1993 e 1994 (Siviero, 2003).

3.2. Software WEKA 3 - Waikato Environment for Knowledge Analysis

O software WEKA 3(2012) é uma coleção de algoritmos de aprendizado de máquina concebido para realizar tarefas de mineração de dados. Os algoritmos podem ser aplicados diretamente a um conjunto de dados ou inseridos a partir do seu próprio código Java. Assim, WEKA 3 (2012), possui ferramentas para os dados de pré-processamento, classificação, regressão, agrupamento, regras de associação e visualização.

3.3. Procedimento

O banco de dados original incluía dados de área molhada (A) e perímetro molhado (P), os quais foram retirados, por conter informação redundante, deste modo foi utilizado somente o raio hidráulico (RH=A/P) e não houve tratamento de valores ausentes, devido os mesmos não existirem.
O banco de dados foi convertido em arquivo.arff, para inserção no banco de dados do WEKA 3. Foi realizado pré-processamento dos dados para discretização dos dados numéricos utilizando o comando PKI-Discretize.
Para evitar overfitting, a validação cruzada (cross-validation) foi escolhida e, após, a seleção dos algoritmos, sendo três algoritmos de aprendizado supervisionado - C4.5, Naive Bayes e Regressão Logística e um algoritmo não supervisionado – EM.

4. RESULTADOS OBTIDOS

A Tabela 1, a seguir, contém os resultados dos algoritmos: supervisionado e não supervisionado utilizados neste artigo.

Tabela 1: Resultados obtidos pelos algoritmos

ALGORITMO	CLASSIFICAÇÃO	
	CORRETA	INCORRETA
C4.5	16 – 40,0%	24 – 60,0%
Naive Bayes	19 – 47,5%	21 – 52,5%
Regressão Logística	12 – 30,0%	28 – 70,0%
EM	Agrupamento	
	0 07 (18%)	
	1 07 (18%)	
	2 07 (18%)	
	3 08 (20%)	
	4 11 (28%)	

5. ANÁLISE E DISCUSSÃO DOS RESULTADOS

A amostra de dados para o treinamento mostrou-se pequena, não refletiu a distribuição dos dados, dado que se traduziu nas classificações obtidas no aprendizado. O algoritmo de aprendizado de supervisão Naive Bayes foi o que apresentou desempenho melhor, em comparação com Árvore de Decisão C4.5 e Regressão Logística.
O banco de dados foi insuficiente para teste no algoritmo Regressão Logística, pois o mesmo classificou incorretamente 70%, condição levantada durante a revisão bibliográfica, algoritmo requer amostras suficientemente grandes.

O algoritmo Expectation Maximization (EM) realizou agrupamento de 5 grupos, nota-se que nos grupos 0, 1 e 2, contém sete elementos em cada com 18%, em um estudo posterior, é preciso verificar quais parâmetros esse algoritmo está agrupando nesses casos.
Supõe-se que, as variáveis do banco de dados coletados, provavelmente, são não lineares e os algoritmos utilizados possuem interação linear, uma possível causa da não obtenção de êxito nas tarefas realizadas pelos algoritmos classificadores.

6. CONCLUSÃO E RECOMENDAÇÕES

Para ter um desempenho melhor nos algoritmos utilizados, sugere-se a utilização do banco de dados maior se for disponível, dado que não se conseguiu fazer deduções mais contundentes nos anos utilizados para o estudo. Por outro lado, tentar algoritmos com concepções mais elaboradas, no sentido de fazer comparação com outras abordagens, por exemplo, relação de interdependência entre os parâmetros conforme sugere Barddal *et al.* (2017).

REFERÊNCIAS BIBLIOGRÁFICAS

1. BARDDAL, J.P. *et al. A survey on feature drift adaptation: Definition, benchmark, challenges and future directions*. *Journal of Systems and Software*, 127, p. 278-294, 2017.
2. BIFET, A. *et al*. *Extremely fast decision tree mining for evolving data streams*, *Part EN: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, Aug.13 - 17*, p. 1733-1742, 2017.
3. JAMBEIRO F., J.E.S. Tratamento Bayesiano de Interações entre Atributos de Alta Cardinalidade. Tese, Doutorado em Ciência da Computação - Instituto de Computação – UNICAMP, 2008.
4. MITCHELL,T. M. *Machine Learning*. McGraw Hill, New York, 414p., 1997.
5. MITCHELL, T.M. *Generative and Discriminative classifiers: Naïve Bayes and Logistic Regression. Machine Learning*. 2º ed., 1: p.1 – 17, 2010. www.cs.cmu.edu/~tom/mlbook.html.
6. MONARD, M.C., BARANAUSKAS, J.A. *Conceitos sobre Aprendizado de Máquina,Part EM Sistemas Inteligêntes: Fundamentos e Aplicações.* Ed. Manole Ltda, Baueri, 4: p.89-114, 2005.
7. RUSSELL, S.R., NORVING, P. *Inteligência Artificial*. Editora Campus, 2º ed., 1: p.1 -31, 2004.
8. SIVIERO, M.R.L. Estudo da Ocupação do Solo a Montante de uma Seção do Rio Atibaia Associada à Descarga Sólida Transportada. Tese, Doutorado em Recursos Hídricos - Faculdade de Engenharia Civil, Arquitetura e Urbanismo – UNICAMP, 116p., 2003.
9. WEKA 3. *Machine Learning Software in Java.* Universidade de Waikato, Nova Zelândia, 2012. http://www.cs.waikato.ac.nz/ml/weka.