https://eventos.utfpr.edu.br//sicite/sicite2020

Redução de dimensionalidade em base de dados de microarranjos utilizando autocodificadores

Dimensionality reduction on microarray databases using autoencoders

Rafael Felipe Tasaka de Melo
rafaelftmelo@outlook.com
Universidade Tecnológica Federal do Paraná, Ponta Grossa, Paraná, Brasil

Helyane Bronoski Borges
helyane@utfpr.edu.br
Universidade Tecnológica Federal do Paraná, Ponta Grossa, Paraná, Brasil

RESUMO

Algoritmos de Aprendizagem de Maquina vem sendo cada vez mais utilizados pela sua capacidade de aprender a partir de grandes volumes de dados como, por exemplo, dados de expressão gênica obtidos pela técnica de microarranjo. Uma característica das bases de dados de microarranjos é que, geralmente, ela é formada por grande quantidade de atributos e um pequeno número de amostras. Sabe-se que dados com alta dimensionalidade podem possuir atributos redundantes e muitas vezes irrelevantes, podendo atrapalhar o processo de aprendizagem e o desempenho das predições. Métodos de redução de dimensionalidade são utilizados para reduzir a quantidade de atributos das bases de dados. Redes Neurais Autocodificadoras podem ser adaptadas e utilizadas para a extração de atributos e, consequentemente, a redução da dimensionalidade. Esta pesquisa tem como objetivo utilizar uma rede neural autocodificadora para ser utilizada na extração de atributos em bases de dados de microarranjo. Para isso, serão realizados experimentos em cinco bases de dados. Os resultados foram avaliados por meio da taxa de acerto de classificadores.

PALAVRAS-CHAVE: Mineração de dados. Redução de dimensionalidade. Autocodificadora.

ABSTRACT

Machine Learning Algorithms are being increasingly used for their ability to learn from large volumes of data, such as, for example, gene expression data obtained by the microarray technique. A characteristic of microarray databases is that, generally, it is formed by a large number of attributes and a small number of samples. It is known that data with high dimensionality can have redundant and often irrelevant attributes - which can hinder the learning process and the performance of predictions. Dimensionality reduction methods are used to reduce the number of database attributes. Autocoding Neural Networks can be adapted and used for the extraction of attributes and, consequently, the reduction of dimensionality. This research aims to use an autocoding neural network to be used in the extraction of attributes in microarray databases. For that, experiments will be carried out in five databases. The results were evaluated using the correctness rate of the classifiers.

KEYWORDS: Data Mining. Dimensionality Reduction. Autoencoders.

Recebido: 19 ago. 2020.
Aprovado: 01 out. 2020.

INTRODUÇÃO

Muitos problemas de Aprendizagem de Máquina utilizam milhares de atributos para o treinamento do algoritmo. Utilizar essa quantidade de atributos faz com que, além de deixar o treinamento do algoritmo lento, fique difícil encontrar um padrão entre os dados para realizar uma possível classificação deles (MENG et. al., 2017).

Algumas dessas bases possuem, também, a característica de muitos atributos e poucos exemplares, isso se deve pela dificuldade em coletar amostras em largas quantidades. Esse problema é conhecido como maldição da dimensionalidade (GÉRON, 2017). Por conta dessa característica muitos dos métodos não conseguem criar um modelo de classificação eficiente o suficiente para prever exemplares futuros por conta da dificuldade em generalizar tamanha quantidade de atributos.

Visando diminuir o problema da maldição da dimensionalidade e outros problemas gerados pela alta dimensionalidade, técnicas de redução de dimensionalidade podem ser aplicadas para retirar da base de dados atributos irrelevantes e/ou redundantes (MENG et. al., 2017).

As técnicas de redução de dimensionalidade podem ser divididas em duas abordagens: extração de atributos e seleção de atributos. Métodos de seleção de atributos selecionam os atributos mais relevantes sem alterar a base de dados, enquanto que os métodos de extração de atributos modificam a base de dados para representar os dados, como por exemplo, o método PCA (Principal Component Analysis) e o LDA (GÉRON, 2017).

Contudo, a projeção feita por esses métodos não se importa com as relações entre as bases originais e as bases reduzidas, fazendo com que uma futura utilização não represente bem seus dados originais (MENG et. al., 2017).

Como alternativa para a tarefa de redução, utiliza-se a aprendizagem profunda através da rede autocodificadora que pode realizar a extração de atributos em suas camadas ocultas (GÉRON, 2017). Essa rede busca reduzir a dimensionalidade da base através da extração de atributos onde, dado um conjunto de atributos *X*, busca-se a criação de um novo conjunto de atributos *Y*, onde são mais expressivos e que melhor representem a variedade dos atributos originais. A estrutura responsável por essa redução se chama “codificador”.

Logo, essa pesquisa tem como objetivo realizar a redução de dimensionalidade utilizando redes neurais autocodificadoras em bases de dados de microarranjo.

METODOLOGIA

A rede autocodificadora é composta por duas estruturas: o codificador e o decodificador (GÉRON, 2017). O codificador possui um mapeamento de entrada *X* e uma representação reduzida *Y* definida na Eq. (1)(MENG, 2017).

$$Y = f(X) = s_f(WX + b_x) \quad (1)$$

em que s_f é uma função de ativação para a rede, W uma matriz de pesos para parametrização e um vetor bias $b \in R^n$.

O decodificador é responsável por mapear a representação *Y* para fazer a reconstrução *Xi* definida na Eq. (2)(MENG, 2017).

$$\text{Xi} = h(Y) = s_h(W'Y + b_y) \quad (2)$$

em que s_h é uma função de ativação para a rede, W' uma matriz de pesos para parametrização e um vetor bias $b \in R^n$.

A rede é totalmente conectada. Para determinar o número de neurônios em cada camada e a quantidade de camadas ocultas para a estrutura autocodificadora, utilizou-se o uma adaptação do método desenvolvido por Siqueira (2019). Esta consiste em, primeiro, determinar um Fator de Redução (*Fr*) e uma Porcentagem de Redução (*Pr*) definidos pela Eq. (3).

$$Fr = Pr/6 \quad (3)$$

onde *Pr* é a porcentagem de redução desejada para a rede.

A Tabela 1 apresenta o número de neurônios em relação às variáveis *Pr* e *Fr* dada uma quantidade de instâncias *N*.

Tabela 1 - Quantidade de neurônios na rede autocodificadora

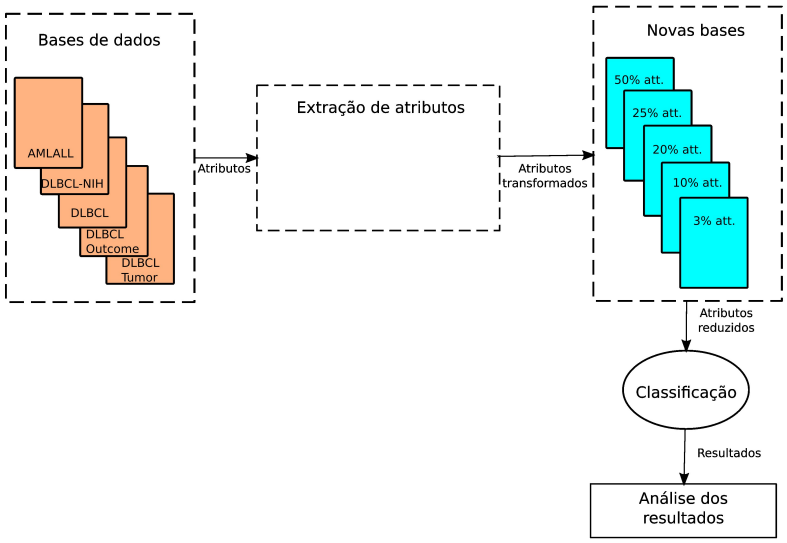
Estrutura da rede	Quantidade de camadas	Quantidade de neurônios
Camada de entrada	1	*N*
Codificadora	4	*N* * (1 - *Fr* * Num. da camada)
Gargalo	1	N * (1 - *Pr*)
Decodificadora	4	*N* * (1 - *Fr* * Num. da camada)
Camada de saída	1	*N*

Fonte: Adaptado de Siqueira (2019)

Conforme proposto por Hinton et. al. (2012), para evitar *overfiting*, utilizou-se *dropout* em diversas camadas ocultas. Além disso se adicionou erro Gaussiano para forçar a rede a aprender atributos relevantes, e também evitar que a rede simplesmente copie os inputs e colocar nos *outputs* (VICENT et. al., 2008), tornando a rede uma *denoising autoencoder*.

A Figura 2 ilustra a metodologia para a realização dos experimentos. Para a realização de experimentos foram utilizadas 5 (cinco) base de dados de microarranjo (KLUG, 2010). Umas das características desses tipos de dados é que são formados por uma grande quantidade de atributos, na ordem de milhares, e relativamente poucas amostras. As características dessas bases de dados são mostradas na Tabela 2.

Figura 1 - Representação geral da metodologia implementada



Fonte: Autoria própria (2020)

Tabela 2 - Quantidade de atributos por base

Bases de Dados	Quantidade de atributos	Quantidade de exemplos
AML/ALL	7129	72
DLBCL-NIH	7399	240
DLBCL	4026	47
DLBCL-Outcome	7129	58
DLBCL-Tumor	7129	77

Fonte: Autoria própria (2020)

Para a realização do treinamento da rede (etapa de "extração de atributos") utilizou-se o método *holdout* (RUSSELL, 2013). Como método de imputação dos valores faltantes (NUNES, 2007), foi utilizado a imputação pela média aritmética, ou seja, o valor a ser atribuído é a média aritmética dos valores de todos os exemplares para determinado atributo. Após o treinamento são obtidas as bases reduzidas. Para a redução dos atributos fez-se experimentos reduzindo a quantidade de atributos em 3%, 10%, 20%, 25% e 50%. Esses valores foram selecionados para posterior comparação com os resultados obtidos em Borges (2006).

Para a avaliação das bases reduzidas, as bases originais são divididas em treinamento e teste e ambos os conjuntos são reduzidos no modelo gerado pela rede. Em seguida são aplicados os algoritmos de classificação K-NN (*K-Nearest Neighbor*) (com K igual a 1, 3, 5 e 7), SVM, Naive Bayes e Árvore de Decisão (etapa de classificação). Esse processo se repete vinte vezes para cada base, obtendo-se assim uma média de acurácias bem como seus desvios padrão. Essas médias foram selecionadas para posterior comparação com os resultados obtidos em Borges (2006) (etapa de "análise dos resultados").

RESULTADOS E DISCUSSÕES

Esta seção apresenta os resultados obtidos pelos classificadores após a aplicação do algoritmo de redução de dimensionalidade.

A Tabela 3 apresenta os resultados obtidos na base de dados AML/ALL conforme os algoritmos de classificação utilizado e o percentual de redução atributos. Observa-se que os resultados dos classificadores foram acima de 92%.

Tabela 3 – Resultados obtidos na base de dados AML/ALL

AML/ALL					
Algoritmo	**3%**	**10%**	**20%**	**25%**	**50%**
k-NN (k=1)	0.952 +- 0.043	0.970 +- 0.046	0.958 +- 0.035	0.966 +- 0.038	0.958 +- 0.038
k-NN (k=3)	0.945 +- 0.043	0.960 +- 0.046	0.943 +- 0.035	0.966 +- 0.038	0.960 +- 0.038
k-NN (k=5)	0.935 +- 0.046	0.966 +- 0.038	0.945 +- 0.041	0.958 +- 0.051	0.952 +- 0.035
k-NN (k=7)	0.927 +- 0.045	0.958 +- 0.047	0.935 +- 0.038	0.954 +- 0.057	0.933 +- 0.048
SVM	0.95 +- 0.040	0.977 +- 0.020	0.958 +- 0.034	0.970 +- 0.043	0.931 +- 0.042
Naive Bayes	0.95 +- 0.033	0.964 +- 0.042	0.962 +- 0.045	0.970 +- 0.032	0.962 +- 0.034
Árvore de Decisão	0.920 +- 0.057	0.95 +- 0.048	0.943 +- 0.051	0.958 +- 0.052	0.945 +- 0.069

Fonte: Autoria própria (2020)

A Tabela 4 apresenta os resultados obtidos na base de dados DLBCL-NIH conforme os algoritmos de classificação utilizado e o percentual de redução atributos. Dentre todas as reduções, reduzir a 50% a quantidade de atributos apresentou as piores acurácias quando comparados com os outros percentuais de redução.

Tabela 4 - Resultados obtidos na base de dados DLBCL-NIH

DLBCL-NIH					
Algoritmo	**3%**	**10%**	**20%**	**25%**	**50%**
k-NN (k=1)	0.838 +- 0.030	0.825 +- 0.042	0.819 +- 0.037	0.801 +- 0.032	0.729 +- 0.040
k-NN (k=3)	0.866 +- 0.030	0.845 +- 0.042	0.849 +- 0.037	0.826 +- 0.032	0.744 +- 0.040
k-NN (k=5)	0.876 +- 0.028	0.863 +- 0.044	0.848 +- 0.041	0.824 +- 0.037	0.710 +- 0.041
k-NN (k=7)	0.873 +- 0.029	0.870 +- 0.040	0.846 +- 0.032	0.823 +- 0.030	0.686 +- 0.033
SVM	0.861 +- 0.029	0.869 +- 0.044	0.863 +- 0.025	0.842 +- 0.031	0.666 +- 0.052
Naive Bayes	0.865 +- 0.027	0.825 +- 0.047	0.822 +- 0.047	0.805 +- 0.041	0.658 +- 0.055
Árvore de Decisão	0.84 +- 0.036	0.817 +- 0.039	0.833 +- 0.043	0.818 +- 0.040	0.733 +- 0.043

Fonte: Autoria própria (2020)

A tabela 5 apresenta os resultados obtidos na base de dados DLBCL conforme os algoritmos de classificação utilizado e o percentual de redução atributos. A base se apresentou com boas acurácias - sendo muito idêntica à base AML/ALL, porém apresentou desvios padrões maiores.

Tabela 5 – Resultados obtidos na base de dados DLBCL

DLBCL					
Algoritmo	**3%**	**10%**	**20%**	**25%**	**50%**
k-NN (k=1)	0.937 +- 0.061	0.943 +- 0.045	0.909 +- 0.063	0.925 +- 0.068	0.931 +- 0.055
k-NN (k=3)	0.95 +- 0.061	0.946 +- 0.045	0.896 +- 0.063	0.921 +- 0.068	0.937 +- 0.055
k-NN (k=5)	0.943 +- 0.065	0.931 +- 0.058	0.881 +- 0.058	0.931 +- 0.062	0.937 +- 0.044
k-NN (k=7)	0.943 +- 0.065	0.921 +- 0.065	0.9 +- 0.05	0.943 +- 0.062	0.940 +- 0.041
SVM	0.940 +- 0.066	0.940 +- 0.054	0.909 +- 0.046	0.934 +- 0.063	0.956 +- 0.028
Naive Bayes	0.962 +- 0.041	0.940 +- 0.060	0.903 +- 0.050	0.943 +- 0.065	0.928 +- 0.045
Árvore de Decisão	0.95 +- 0.061	0.915 +- 0.066	0.871 +- 0.082	0.906 +- 0.064	0.928 +- 0.049

Fonte: Autoria própria (2020)

A Tabela 6 apresenta os resultados obtidos na base de dados DLBCL-Outcome conforme os algoritmos de classificação utilizado e o percentual de redução atributos. Esta base de dados foi a que apresentou os piores resultados se comparada com as demais, ela também apresenta a pior acurácia obtida dentre todos os resultados do estudo (somente 42% de acurácia para 50% dos atributos para o Algoritmo 7NN).

Tabela 6 - Resultado obtidos na base de dados DLBCL-Outcome

DLBCL-Outcome					
Algoritmo	**3%**	**10%**	**20%**	**25%**	**50%**
k-NN (k=1)	0.817 +- 0.063	0.807 +- 0.074	0.815 +- 0.065	0.777 +- 0.073	0.532 +- 0.114
k-NN (k=3)	0.815 +- 0.063	0.847 +- 0.074	0.842 +- 0.065	0.837 +- 0.073	0.475 +- 0.114
k-NN (k=5)	0.837 +- 0.056	0.845 +- 0.052	0.830 +- 0.064	0.855 +- 0.058	0.447 +- 0.108
k-NN (k=7)	0.84 +- 0.062	0.845 +- 0.056	0.815 +- 0.054	0.855 +- 0.063	0.427 +- 0.108
SVM	0.83 +- 0.059	0.85 +- 0.063	0.825 +- 0.062	0.847 +- 0.062	0.455 +- 0.110
Naive Bayes	0.807 +- 0.067	0.857 +- 0.063	0.822 +- 0.055	0.837 +- 0.066	0.527 +- 0.116
Árvore de Decisão	0.785 +- 0.098	0.84 +- 0.069	0.8 +- 0.083	0.815 +- 0.080	0.5 +- 0.088

Fonte: Autoria própria (2020)

A Tabela 7 apresenta os resultados obtidos na base de dados DLBCL-Tumor conforme os algoritmos de classificação utilizado e o percentual de redução atributos. A base se apresentou satisfatória para as acurácias, porém em diversas situações apresentou desvios padrões maiores que as outras bases.

Tabela 7 - Resultados obtidos na base de dados DLBCL-Tumor

DLBCL-Tumor					
Algoritmo	**3%**	**10%**	**20%**	**25%**	**50%**
k-NN (k=1)	0.909 +- 0.044	0.911 +- 0.040	0.875 +- 0.063	0.932 +- 0.051	0.917 +- 0.054
k-NN (k=3)	0.907 +- 0.044	0.923 +- 0.040	0.876 +- 0.063	0.913 +- 0.051	0.890 +- 0.054
k-NN (k=5)	0.896 +- 0.045	0.882 +- 0.057	0.869 +- 0.068	0.9 +- 0.042	0.855 +- 0.056
k-NN (k=7)	0.890 +- 0.044	0.863 +- 0.069	0.834 +- 0.085	0.898 +- 0.057	0.836 +- 0.072
SVM	0.905 +- 0.046	0.907 +- 0.063	0.878 +- 0.071	0.930 +- 0.051	0.825 +- 0.093
Naive Bayes	0.934 +- 0.047	0.913 +- 0.054	0.896 +- 0.045	0.946 +- 0.039	0.938 +- 0.037
Árvore de Decisão	0.907 +- 0.071	0.905 +- 0.072	0.882 +- 0.047	0.959 +- 0.046	0.923 +- 0.053

Fonte: Autoria própria (2020)

A Tabela 8 apresenta as médias aritméticas obtidas nos experimentos. Nota-se que as médias não se distanciam das observações já apresentadas. As piores médias se situam na base de dados DLBCL-Outcome e as melhores na base de dados AML/ALL.

Tabela 8 - Média aritmética dos resultados obtidos por base de dados do trabalho proposto.

Média geral por Algoritmo de Classificação - trabalho proposto					
Algoritmo	**AML/ALL**	**DLBCL-NIH**	**DLBCL**	**DLBCL-OUTCOME**	**DLBCL-Tumor**
k-NN (k=1)	0.960 +- 0.04	0.802 +- 0.036	0.929 +- 0.058	0.749 +- 0.077	0.908 +- 0.050
k-NN (k=3)	0.954 +- 0.04	0.826 +- 0.036	0.93 +- 0.058	0.770 +- 0.077	0.901 +- 0.050
k-NN (k=5)	0.951 +- 0.042	0.824 +- 0.038	0.924 +- 0.057	0.762 +- 0.067	0.880 +- 0.053
k-NN (k=7)	0.941 +- 0.047	0.812 +- 0.032	0.929 +- 0.056	0.756 +- 0.068	0.864 +- 0.065
SVM	0.957 +- 0.035	0.820 +- 0.036	0.935 +- 0.051	0.761 +- 0.071	0.889 +- 0.064
Naive Bayes	0.961 +- 0.037	0.795 +- 0.043	0.935 +- 0.052	0.77 +- 0.073	0.925 +- 0.044

Árvore de Decisão	0.943 +- 0.055	0.808 +- 0.040	0.914 +- 0.064	0.748 +- 0.083	0.915 +- 0.057

Fonte: Autoria própria (2020)

Borges (2006) realizou experimentos com as mesmas bases de dados utilizando o método de extração de atributos por meio da Projeção Aleatória. A Tabela 9 mostra as médias dos resultados obtidos pelo autor.

Tabela 9 - Média aritmética dos resultados obtidos por base de dados de Borges (2006).

Média geral por Algoritmo de Classificação - Borges (2006)					
Algoritmo	**AML/ALL**	**DLBCL-NIH**	**DLBCL**	**DLBCL-OUTCOME**	**DLBCL-Tumor**
k-NN (k=1)	0.915 +- 0.013	0.569 +- 0.004	0.8 +- 0.023	0.540 +- 0.020	0.879 +- 0.002
k-NN (k=3)	0.921 +- 0.016	0.600 +- 0.009	0.846 +- 0.031	0.568 +- 0.009	0.868 +- 0.004
k-NN (k=5)	0.908 +- 0.013	0.584 +- 0.008	0.856 +- 0.027	0.539 +- 0.012	0.901 +- 0.009
k-NN (k=7)	0.899 +- 0.010	0.581 +- 0.005	0.852 +- 0.029	0.509 +- 0.010	0.910 +- 0.008
SVM	0.974 +- 0.007	0.624 +- 0.016	0.930 +- 0.014	0.520 +- 0.007	0.964 +- 0.004
Naive Bayes	0.927 +- 0.008	0.600 +- 0.003	0.935 +- 0.018	0.384 +- 0.013	0.885 +- 0.004
Árvore de Decisão	0.816 +- 0.011	0.545 +- 0.014	0.753 +- 0.037	0.475 +- 0.013	0.848 +- 0.016

Fonte: Adaptado de Borges (2006)

Ao comparar o trabalho proposto com o de Borges (2006), é possível notar que, em todos os casos, o desvio padrão é maior neste trabalho, contudo a acurácia é perceptivelmente maior nas bases DLBCL-NIH e DLBCL-Outcome, nas demais bases as acurácias são próximas.

CONCLUSÕES

A redução de dimensionalidade pode melhorar o desempenho na tarefa de classificação, visualização e armazenamento de grandes bases de dados. Este estudo teve como objetivo propor um método para a redução treinando, primeiro, a estrutura codificadora com classificação e, em seguida, utilizar a estrutura codificadora numa rede autocodificadora. A rede reduz a quantidade de atributos das bases de dados de microarranjo, contudo seu custo computacional se mostrou elevado.

O método apresentou resultados, em sua grande maioria, acima de 75%, contudo a base de dados DLBCL-Outcome obteve resultados inferiores quando comparados com as outras bases de dados.

Como trabalhos futuros, deve ser realizado uma análise estatística para comprovar a eficiência do método.

REFERÊNCIAS

BORGES, H. B. **Redução de dimensionalidade em bases de dados de expressão gênica.** 2006. Dissertação (Mestrado em Informática) - Pontifícia Universidade Católica do Paraná, Curitiba, 2006.

GÉRON, A. **Handos-On Machine Learning with Scikit-Learn & TensorFlow.**1. ed. Sebastopol: O'Reilly Media Inc. 2017.

HINTON, G. E. et. al. **Improving neural networks by preventing co-adaptation of feature detectors**. 2012. Disponível em: https://arxiv.org/abs/1207.0580. Acesso em: 03 set. 2020.

KLUG, W.S et. al. **Conceitos de Genética**. 9. ed. Ponto Alegre: Artmed, 2010.

MENG, Q. et al. Relational autoencoder for feature extraction. **2017 International Joint Conference On Neural Networks (IJCNN)**, Anchorage, p. 364-371, maio 2017. IEEE.

NUNES, Luciana Neves. **Métodos de imputação de dados aplicados na área da saúde**. 2007. Tese (Doutorado em Epidemiologia) - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2007.

RUSSELL, Stuart; NORVIG, Peter. **Inteligência Artificial.** 3. ed. Rio de Janeiro: Elsevier. 2013

SIQUEIRA, Rafael Fernandes Siqueira. **Redução de dimensionalidade em bases de dados de classificação hierárquica multirrótulo usando autoencoders**. 2019. Dissertação (Mestrado em Ciência da Computação) - Universidade Tecnológica Federal do Paraná, Ponta Grossa, 2019.

VINCENT, Pascal et al. Extracting and composing robust features with denoising autoencoders. **Proceedings Of The 25Th International Conference On Machine Learning - ICML '08**, Helsinque, p. 1096-1103, ago. 2008. ACM Press.