Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, v. 10, n. 1, 2023.

Trabalho apresentado no XLII CNMAC, Universidade Federal de Mato Grosso do Sul - Bonito - MS, 2023

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Usando um Algoritmo Genético para diminuir a dimensão de um problema de classificação

André G. C. Pereira[1], Viviane S.M. Campos [2]
DMAT/UFRN, Natal, RN
Davi R.M. Costa[3], Ricardo Theodoro[4]
FEARP/USP, Ribeirão Preto, SP

Resumo. Existem diversos tipos de algoritmos de otimização bem como algoritmos de classificação. Dentre tais algoritmos, o algoritmo genético elitista é um representante dos algoritmos de otimização, enquanto o KNN é um representante dos algoritmos de classificação. O objetivo desse trabalho é mostrar, através de uma aplicação, como podemos usar essas duas classes de algoritmos em conjunto para não apenas otimizar o número de acertos de classificação, mas também para reduzir a dimensão do problema. A situação utilizada é a classificação de cooperativas brasileiras usando o texto de seus estatutos. O banco de palavras utilizado constava de 8293 palavras que ao longo do processo foi reduzido para 1037 palavras e o número de sucessos de classificação foi maior que 81%.

Palavras-chave. Algoritmo Genético, KNN, Otimização, Classificação.

1 Introdução

Algoritmos genéticos são geralmente utilizados para encontrar a solução ótima aproximada de uma uma função $f : A \subset \mathbb{R}^n \to \mathbb{R}$, chamada função objetivo. O termo solução ótima aproximada se deve ao fato de que no processo de busca da solução ótima, uma discretização $D \subset A$ é utilizada como o domínio da função f em questão, ver [9–11].

O conjunto D é obtido de modo a possuir uma quantidade de elementos que seja uma potência de 2, por exemplo 2^l, o que permite identificar cada elemento de D como um vetor binário de comprimento l. Considerar os pontos nesse formato ajuda na execução das etapas de cruzamento e mutação do algoritmo genético.

A apresentação dos pontos no formato binário permite a utilização dos algoritmos genéticos na seleção de variáveis de um modelo de regressão linear como segue abaixo. Suponha que se deseja determinar quais variáveis X_i são estatísticamente significantes no modelo linear

$$Y = \alpha_0 + \alpha_1 X_1 + \alpha_2 X_2 + \alpha_3 X_3 + \varepsilon,$$

onde ε é o erro. Considera-se o conjunto dos vetores binários de três coordenadas $B = \{(x_1, x_2, x_3)/x_i \in \{0, 1\}, i = 1, 2, 3\}$, onde cada $(x_1, x_2, x_3) \in B$ indica quais variáveis estão sendo consideradas no momento, por exemplo, se o ponto escolhido é o $(0, 1, 1)$, então o modelo considerado é o:

$$Y = \alpha_0 + \alpha_2 X_2 + \alpha_3 X_3 + \varepsilon.$$

[1]andre.gustavo.campos@ufrn.br
[2]viviane.simioli@ufrn.br
[3]drmouracosta@usp.br
[4]rtheodoro@usp.br

 © 2023 SBMAC

Em seguida, estima-se o modelo e define-se a função $f : B \to \mathbb{R}$, que a cada elemento de B associa o critério de informação de Akaike (AIC) do modelo estimado. A teoria de seleção de modelos garante que aquele com o menor AIC é o modelo mais ajustado. Assim, o algoritmo genético pode ser usado para encontrar o ponto de mínimo desta função e esse ponto de mínimo determina quais variáveis devem ser utilizadas para a obtenção do modelo mais ajustado aos dados. Essa ideia foi usada em vários problemas, ver [1, 6, 8].

Os algoritmos de aprendizado de máquina são utilizados em problemas de classificação, que podem ser supervisionados ou não supervisionados, dependendo se as variáveis respostas são conhecidas ou não. Tais algoritmos também podem ser usados em problemas de regressão, caso as variáveis respostas sejam numéricas. Dentre os algoritmos de aprendizado de máquina podemos citar o K Nearest Neighbors (KNN), floresta aleatória, Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), etc. Esses algoritmos estão descritos com mais detalhes em [4, 5, 7].

Quando os algoritmos de aprendizado de máquina são utilizados como modelos estatísticos de classificação, um conjunto de informações (conjunto de treinamento) é utilizado para classificar novas observações (conjunto de teste) dentro de um conjunto de categorias previamente conhecidas.

Neste trabalho, nosso objetivo é descobrir se o conjunto de palavras utilizadas na escrita de um estatuto pode levar a determinação da cooperativa que o emitiu. A técnica aqui empregada pode ser utilizada para realizar vários outros tipos de classificações, como por exemplo classificar através do conjunto de palavras de um texto matemático se ele é um texto da área de probabilidade, álgebra, geometria diferencial, etc.

Utilizamos o algoritmo genético elitista (AGE) juntamente com o KNN (algoritmo de aprendizado de máquina) para classificar os estatutos de cooperativas brasileiras. Foram utilizados 138 estatutos de onde 8293 palavras foram selecionadas e usadas no processo de classificação, tomando o cuidado de retirar o nome e o CNPJ da cooperativa. As categorias previamente estabelecidas foram SICOOB, SICREDI, UNICRED, CRESOL e OUTROS. O algoritmo de aprendizado de máquina foi utilizado para fornecer a função objetivo a ser maximizada pelo AGE. O AGE durante sua execução realiza uma seleção de palavras que ajuda na maximação da função objetivo. Em resumo, o algoritmo de aprendizado de máquina nos fornece a função que calcula o número médio de acertos de classificação enquanto o AGE seleciona conjuntos de palavras diferentes a fim de maximizar esse número de acertos de classificação.

Este trabalho está dividido em cinco seções. Na Seção 2 é apresentada a versão do AGE e a versão do algoritmo de aprendizado de máquina utilizadas, na Seção 3 o problema de classificação dos estatutos é modelado, na Seção 4 apresentamos os resultados numéricos obtidos e a Seção 5 é composta pela conclusão e considerações finais.

2 AGE e KNN

2.1 Algoritmo Genético Elitista

O Algoritmo Genético descrito em [3], é uma ferramenta computacional que tenta emular o processo evolucionário de Darwin, o qual utiliza três estágios: Seleção, Cruzamento e Mutação. Esse tipo de algoritmo é usado para encontrar a solução ótima aproximada de uma dada função $f : A \to \mathbb{R}$.

Para executar os passos do algoritmo o conjunto A deve ser discretizado, ou seja, se constrói um conjunto $D \subset A$ de modo que cada ponto seja representado por vetores binários de comprimento l, onde l depende da precisão desejada. Sem prejuízo para o entendimento, como cada ponto de D é identificado como um vetor binário, assume-se que os pontos de D são esses vetores binários. Uma população de N indivíduos é qualquer N-upla de elementos de D e $Z = \{(u_1, u_2, ..., u_N); u_i \in$

 © 2023 SBMAC

$D, i = 1, 2, ..., N\}$ é o conjunto de todas as populações de N indivíduos, onde cada u_i é um vetor binário de comprimento l.

O resultado esperado depois de executado o algoritmo genético é que ele convergisse para a solução ótima procurada. No entanto, Rudolph, em [12] demonstrou que isso não acontece quase certamente (ou seja, com probabilidade 1) e apresentou o algoritmo genético elitista que resolveu esse problema de convergência. Esse novo algoritmo evolui da seguinte maneira:

a) Escolha aleatoriamente uma população inicial tendo N elementos, cada um sendo um vetor binário de comprimento l, e crie mais uma posição, a $(N+1)$-ésima entrada do vetor população, a qual manterá o "melhor" elemento daqueles N elementos anteriores.

b) Repita

1. Execute a seleção com os N primeiros elementos

2. Execute o cruzamento com os N primeiros elementos

3. Execute a mutação com os N primeiros elementos

4. Se o melhor elemento dessa nova população é melhor que aquele que está na $(N + 1)$-ésima posição, troque a $(N + 1)$-ésima posição por esse melhor elemento, caso contrário preserve a $(N + 1)$-ésima posição inalterada.

c) Até que algum critério de parada seja atingido.

Algumas versões convergentes desse algoritmo em que os parâmetros variam podem ser vistos em [2, 9–11].

2.2 Algoritmos de Aprendizado de Máquina

Os algoritmos de aprendizado de máquina supervisionados precisam de um conjunto de dados cuja variável resposta é conhecida. Esse conjunto é dividido em dois outros conjuntos, a saber: conjunto de treinamento e conjunto de teste. O conjunto de treinamento é usado para ajustar o modelo enquanto o conjunto de teste é usado para avaliar o modelo ajustado. Na sequência, é gerada uma medida que avalia se o modelo ajustado pelo conjunto de treinamento responde bem aos novos dados pertencentes ao conjunto de teste. Essa medida gerada é chamada de validação cruzada (cross-validation).

Esses algoritmos podem ser utilizados para resolver dois tipos de problemas: de classificação, quando a variável resposta é categórica, ou de regressão, quando a variável resposta é numérica.

A validação cruzada utilizada para medir a eficácia dos algoritmos, no caso de classificação, é o número de acertos obtidos pelo modelo ajustado quando aplicado ao conjunto de teste. No caso de regressão, verifica-se o erro quadrático médio dos dados do conjunto de teste preditos pelo modelo ajustado.

O problema que tratamos neste artigo é um problema de classificação supervisionado, portanto a partir de agora só falaremos das versões dos algoritmos aplicados à classificação.

2.2.1 O algoritmo KNN

O algoritmo KNN conhecido como os K vizinhos mais próximos tem K como parâmetro e no caso de classificação funciona da seguinte maneira:

1. Dado um ponto que se deseja classificar (do conjunto de teste), encontra-se os K pontos do conjunto de treinamento mais próximos desse ponto e atribui-se a esse ponto a mesma classificação da maioria dos K pontos selecionados.

DOI: 10.5540/03.2023.010.01.0115 © 2023 SBMAC

2. Depois das classificações feitas, verifica-se quantas delas realmente coincidiram com a classificação correta.

3. A medida é então a proporção de acertos.

3 Modelagem do problema

O objetivo deste artigo é mostrar que podemos identificar os estatutos das cooperativas brasileiras levando em consideração o conjunto de palavras usado em sua escrita, ou seja, é mostrar que os estatutos das cooperativas brasileiras são escritos de modo que é possível identificar, com uma boa acertabilidade, qual cooperativa o emitiu, caso seu CNPJ e nome não estejam presentes no texto. A modelagem deste problema começa com a escolha de um banco de palavras, dentre as que aparecem no corpo do estatuto. São utilizadas não apenas as palavras mas também a quantidade de vezes que cada uma delas aparece em cada estatuto. As etapas se desenvolvem da seguinte maneira:

a. O algoritmo de aprendizado de máquina é usado para obter a função a ser maximizada, ou seja, a função que calcula o número de acertos levando em conta o conjunto de estatutos utilizados (separando-os em conjuntos de treinamento e teste).

b. O AGE é usado para encontrar o grupo de palavras, dentre as palavras do banco de palavras, que realmente ajuda na maximização da função definida no item anterior.

c. O AGE é usado para mudar o conjunto de palavras utilizadas objetivando conseguir um conjunto "mínimo" .

A organização do banco de palavras no formato utilizado pelo programa R, foi realizada pelo programa Python versão 3.10.6 e o algoritmos AGE e KNN foram implementados no programa RStudio versão 2022.12.0.

O item a. é o ponto essencial dessa modelagem uma vez que fornece a função a ser maximizada pelo AGE. Nesse passo, o algoritmo de aprendizado de máquina utilizado é o KNN com K=1 e a validação cruzada é a Leave-One-Out Cross-Validation (LOOCV), que utiliza o conjunto de teste com um único elemento, o restante dos elementos são usados como o conjunto de treinamento e isso é feito para cada elemento do nosso conjunto de dados. Assim, o número de classificações é igual ao número de estatutos analisados e, por fim, obtém-se o número de acertos, ou a proporção de acertos. Cada subconjunto de palavras utilizado fornece um resultado diferente no número de classificações corretas.

Vemos então que existem dois problemas atuando em conjunto:

- Um problema de classificação: Verificar se a informação disponível (conjunto de palavras escolhidas) classifica de maneira satisfatória (usando a validação cruzada) o estatuto.

- Um problema de seleção de variáveis/otimização: Encontrar qual o conjunto de palavras melhora a identificação, no sentido de que a proporção de acertos é a maior possível.

Os dados dos estatutos são apresentados em uma matriz, conforme a Tabela 1 sendo x_{ij} são as quantidades de vezes que cada palavra P_j aparece no estatuto i. Note que na última coluna temos as classificações corretas conhecidas, uma vez que estamos tratando de um problema supervisionado.

 © 2023 SBMAC

Tabela 1: Organização dos dados.

	P_1	P_2	P_3	...	P_N	Classificação
estatuto_1	x_{11}	x_{12}	x_{13}	...	x_{1N}	c_1
estatuto_2	x_{21}	x_{22}	x_{23}	...	x_{2N}	c_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	...	$\vdots$	$\vdots$
estatuto_n	x_{n1}	x_{n2}	x_{n3}	...	x_{nN}	c_n

3.1 Modelando o conjunto discreto e a função objetivo

Como colocado na Seção 3, o conjunto de dados consiste de uma matriz cujas colunas representam o banco de palavras e a classificação correta, as linhas representam os estatutos, e as entradas da matriz representam o número de vezes que cada uma das palavras aparece em cada estatuto. Para montar a função objetivo o procedimento é o seguinte:

1. É selecionado um conjunto qualquer de palavras.

2. Para cada um dos n estatutos, verifica-se a distância dele em relação a todos os outros, considerando apenas o conjunto de palavras selecionadas.

3. Cada um desses n estatutos é classificado com a mesma classificação do estatuto mais próximo dele, usando o algoritmo KNN (ou qualquer outro algoritmo de aprendizado de máquina que possa ser utilizado para classificação).

4. Depois de estabelecida as classificações verifica-se quantas delas eram corretas, obtendo assim a validação cruzada desse conjunto de estatutos, considerando o conjunto de palavras selecionadas no item 1.

Seja D_P o conjunto de vetores binários de comprimento N, onde N é o número de palavras utilizadas. Dado $v \in D_P$, as coordenadas nulas desse vetor significam que as palavras relativas àquelas colunas não serão consideradas, já as coordenadas 1 indicam as palavras que serão consideradas.

Considerando o conjunto D_P e os passos 1,2,3 e 4 acima, é construída a função objetivo $f : D_p \to \mathbb{R}$, que a cada vetor binário (o qual informa que palavras estão sendo utilizadas naquele momento) associa o número de sucessos obtido pelo classificador KNN (ou qualquer outro algoritmo de classificação). O AGE utiliza essa função f como a função objetivo a fim de obter a solução ótima.

Uma vez determinada essa solução ótima, ou seja, o conjunto de palavras que gera o maior número de acertos com o KNN, outros algoritmos de aprendizado de máquina podem ser utilizados com esse conjunto de palavras fixas, a fim de aumentar o número de acertos de classificação.

Na Seção 4 são apresentados os resultados obtidos na análise de 138 estatutos e um conjunto inicial com 8293 palavras. Foi utilizado o KNN com K=1 tanto para a classificação do estatuto (passo 3) como também para melhorar o número de acertos de classificação.

4 Resultados Numéricos

A fim de reduzir o número de palavras, o AGE foi utilizado em quatro etapas. Na primeira etapa, todas as palavras foram utilizadas e uma quantidade de passos que o algoritmo deveria executar foi pré-estabelecida. Na segunda etapa, verificou-se quais palavras estavam sendo utilizadas para obtenção da melhor solução. Na terceira etapa, o AGE foi novamente utilizado com o conjunto de palavras sendo aquele obtido ao final da etapa anterior, ou seja, apenas as palavras presentes na melhor solução foram consideradas na etapa seguinte. Novamente depois de uma certa quantidade

 © 2023 SBMAC

de passos pré-determinada, verificou-se, na quarta etapa, qual subconjunto de palavras estava sendo utilizado para uma melhor acertabilidade. Essa melhor solução foi colocada como melhor solução na etapa inicial e o processo recomeçou com a população inicial usando todas as palavras novamente. A terceira etapa é considerada uma busca local em torno da melhor solução obtida até aquele momento.

Note que o número de acertos de uma fase para outra não diminui. Isso acontece devido ao fato do AGE estar maximizando a função que conta a quantidade de acertos. Uma vez que estamos sempre retornando com a população inicial, sempre haverá a possibilidade do conjunto de palavras ótimo estar presente em algum momento na população.

Neste trabalho foi utilizado um conjunto inicial de 8293 palavras (palavras que constam nos estatutos). O algoritmo de aprendizado de máquina usado foi o KNN com parâmetro K=1, o AGE usado teve como probabilidade de cruzamento $p_c = 0.5$, probabilidade de mutação $p_m = 0.4$ e na primeira tentativa usamos a quantidade de passos de cada etapa igual a 30. Depois de executadas essas quatro etapas foi obtida uma quantidade de palavras igual a 1037 e a taxa de acerto foi de 81.88%. Uma observação que se faz necessário nesse momento devido ao caráter estocástico do algoritmo, é que cada realização do programa fornece uma resposta diferente, seja do conjunto de palavras (a maioria iguais as encontradas anteriormente) seja da acertabilidade (variando entre 75% a 90% na maioria das simulações).

5 Conclusão

Neste trabalho as cooperativas foram classificadas através das palavras presentes em seus estatutos. Nesse intento, dois problemas apareceram, o primeiro foi o de detectar quais palavras estavam sendo importantes na classificação e o segundo foi o de otimizar o número de acertos no processo de classificação. A ferramenta utilizada foi o AGE, cuja função objetivo foi construída a partir da teoria de aprendizado de máquina. A validação cruzada utilizada para medir a eficácia do processo foi o Leave-One-Out Cross-Validation (LOOCV) que utiliza o conjunto de teste com um elemento e o restante dos elementos são usados como conjunto de treinamento, e isso foi feito para cada elemento do conjunto de dados. A classificação do elemento do conjunto de teste foi determinado pelo estado do ponto do conjunto de treinamento que está mais próximo do elemento em análise. A taxa de acerto se mostrou muito boa, mais de 80% para essa primeira abordagem mais simples.

Então foi possível utilizar um algoritmo de otimização, tanto para diminuir a dimensão de um problema de classificação quanto para otimizar o número de acertos de classificação. No entanto, não existe nenhum motivo particular para que o algoritmo de otimização seja o AGE e que o algoritmo de classificação seja o KNN, abrindo assim várias possibilidades para outros trabalhos.

Referências

[1] E. Acosta-González e F. Fernández-Rodríguez. "Model selection via genetic algorithms illustrated with cross-country growth data". Em: **Empirical Economics** 33 (2007), pp. 313–337. DOI: 10.1007/s00181-006-0104-3.

[2] V.S.M. Campos, A.G.C. Pereira e J.A. Rojas Cruz. "Modeling the Genetic Algorithm by a Non-Homogeneous Markov Chain: Weak and Strong Ergodicity". Em: **Theory of Probability and its Applications** 57 (2012), pp. 185–192. DOI: 10.4213/tvp4440.

[3] J.H. Holland. **Adaptation in natural and artificial systems**. Ann Arbor: The University of Michigan Press, 1975.

 © 2023 SBMAC

[4] G. James et al. **An introduction to statistical learning with R applications**. Springer, 2014.

[5] A. Kasambara. **Machine Learning Essentials : Practical Guide in R**. Published by STHDA, 2017, 2017.

[6] E. G. Lacerda, A. C. Carvalho e T. B. Ludermir. "Model selection via genetic algorithms for rbf networks". Em: **Journal of Intelligent & Fuzzy Systems, IOS Press** 13 (2002), pp. 111–122.

[7] B. Lantz. **Machine Learning with R : Expert techniques for predictive modeling**. Packt, Birmingham, 2019.

[8] S. Paterlini e T. Minerva. "Regression model selection using genetic algorithms". Em: **Proceedings of the 11th WSEAS international conference on nural networks and 11th WSEAS international conference on evolutionary computing and 11th WSEAS international conference on Fuzzy systems. World Scientific and Engineering Academy and Society (WSEAS)**. 2010, pp. 19–27.

[9] A. G. C. Pereira e B.B. Andrade. "On the Genetic Algorithm with Adaptive Mutation Rate and Selected Statistical Applications". Em: **Computational Statistics (Zeitschrift)** 30 (2015), pp. 131–150. DOI: 10.1007/s00180-014-0526-x.

[10] A. G. C. Pereira et al. "Convergence Analysis of an Elitist non-Homogeneous Genetic Algorithm with Crossover/Mutation Probabilities Adjusted by a Fuzzy Controller". Em: **Chilean Journal of Statistics** 9 (2018), pp. 19–32.

[11] A. G. C. Pereira et al. "On the convergence rate of the elitist genetic algorithm based on mutation probability." Em: **Communications in Statistics - Theory and Methods** 49 (2019), pp. 769–780. DOI: 10.1080/03610926.2018.1528361.

[12] G. Rudolph. "Convergence Analysis of Canonical Genetic Algorithms". Em: **IEEE Transactions on Neural Networks** 5 (1994), pp. 96–101. DOI: 10.1109/72.265964.

 © 2023 SBMAC