ISSN: 2317-0840

Algoritmos genéticos e de aprendizado de máquina na otimização em problemas de classificação

Ricardo Theodoro [†1] , André G. C. Pereira[2], Davi Rogério M. Costa [1], Viviane S. M. Campos[2].

[1] *Universidade São Paulo (USP), São Paulo/SP.*
[2] *Universidade Federal do Rio Grande do Norte (UFRN), Rio Grande do Norte/NO.*

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, a Árvore de Classificação e a Floresta Aleatória são representantes dos algoritmos de classificação. O objetivo desse trabalho é mostrar, através de uma aplicação, como é possível 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 de crédito brasileiras usando o texto de seus estatutos. O banco de palavras utilizado constava de 8.293 palavras que ao longo do processo foi reduzido para 1.037 palavras com proporção de acertos de classificação maior que 81% usando o KNN e de 1.936 e com proporção de acertos maior que 82% usando a Floresta Aleatória.*

Palavras-chave: *Algoritmo Genético Elitista; K-Nearest Neighbors (KNN); Floresta Aleatória.*

Genetic and machine mearning algorithms in optimization in classification problems

Abstract: *There are various types of optimization algorithms as well as classification algorithms. Among such algorithms, the Elitist Genetic Algorithm is one optimization algorithms, while KNN, Decision Tree, and Random Forest are classification algorithms. The objective of this work is to illustrate, through an application, how to use these two classes of algorithms together not only to optimize the number of correct classifications but also to reduce the dimension of the problem. The scenario used is the classification of Brazilian credit cooperatives using the text from their bylaws. The word bank used consisted of 8,293 words, which was reduced to 1,037 words with a classification accurancy higher than 81% when the KNN was used, and it was reduced to 1,936 words with a classification accurancy higher than 82% when the Random Forest was used.*

Keywords: *Elitist Genetic Algorithm; K-Nearest Neighbors (KNN); Random Forest.*

Introdução

As cooperativas foram reconhecidas pela Organização das Nações Unidas (ONU), em 2012, por serem organizações resistentes e viáveis em tempos de dificuldade econômica, cuja atuação contribui para reduzir a desigualdade. Isto é, evitam que seus associados e indivíduos das comunidades onde estão inseridas não estejam na linha de pobreza nem migrem para ela (KI-MOON, 2012). Todavia, esse tipo de empreendimento ainda é pouco presente nas estatísticas dos diferentes países. Além disso, os dados existentes têm baixo potencial de comparabilidade (BOUCHARD; ROUSSELIèRE; GUERNIC, 2017). Esse cenário motivou a Aliança Cooperativa Internacional (ICA) e a Organização Internacional do Trabalho (ILO), em 2017, a propor um guia para nortear a geração de estatísticas sistemáticas e que permitam comparabilidade. O guia propõe que as cooperativas sejam categorizadas em quatro tipos: cooperativas de produção; cooperativa de consumo e serviços; cooperativas de trabalho e; cooperativas de múltiplos interesses (EUM; CARINI; BOUCHARD, 2020).

[†]Autora correspondente: ⟨rtheodoro@usp.br⟩

No Brasil, entretanto, a nomenclatura das cooperativas ainda não foi alterada; persiste aquela proposta pela Organização das Cooperativas Brasileiras (OCB), que as dividem em sete grupos: agropecuárias; consumo; crédito: infraestrutura; saúde; trabalho, produção de bens e serviços (TPBS) e; transporte. Essa divisão segue critérios de representação política e não, necessariamente, uma tipificação a partir das suas características (OCB, 2019). Além disso, não se tem, por exemplo, no Instituto Brasileiro de Geografia e Estatística (IBGE) um conjunto de dados sobre essas organizações. As estatísticas existentes por vezes são viesadas, de difícil acesso e que não permitem replicação.

Diante desse cenário, consolidar os primeiros passos para gerar as condições necessárias para o desenvolvimento do guia proposto por (BOUCHARD et al., 2020) é fundamental. Em outras palavras, como usar um conjunto de características das cooperativas a fim de classificá-las segundo os tipos sugeridos por (BOUCHARD et al., 2020), a saber: de produção, de consumo e serviços, de trabalho ou de múltiplos interesses? Existe um conjunto mínimo de características que permita a classificação com uma boa acurácia?

Vemos então que a situação descrita acima é um problema de classificação uma vez que vai usar um conjunto de características (dados obtidos das cooperativas) para classificar qual o tipo de cooperativa se adequa melhor aos dados fornecidos. Na busca de uma solução, duas situações devem ser trabalhadas em conjunto. A primeira é que o conjunto de características permita uma boa classificação, ou seja, que o modelo tenha uma boa taxa de acertos nas classificações com os dados considerados. A segunda, é a obtenção de um conjunto de características menor possível e que nos forneça ainda uma boa taxa de acerto nas classificações.

Devido ao tempo exigido na tradução dos documentos que utilizaremos como base de dados, apresentamos nesse trabalho a técnica que será usada para resolver o problema acima aplicada em uma problemática equivalente, a saber: 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 ou geometria diferencial ; pelas palavras utilizadas em um e-mail se o mesmo é spam ou não, etc.

Vemos que a solução do problema envolve a obtenção de uma boa taxa de acerto de classificação juntamente com a busca de um conjunto com menos características possíveis, em outras palavras, temos um problema de classificação e um de otimização rodando em conjunto. Lembramos a seguir de alguns algoritmos apropriados para cada uma dessas etapas e de como eles tem sido utilizados.

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 Pereira e Andrade (2015), Pereira et al. (2018), Pereira et al. (2020).

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:

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

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 Lacerda, Carvalho e Ludermir (2002), Acosta-González e Fernández-Rodríguez (2007), Paterlini e Minerva (2010).

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 é possível citar o *K-Nearest Neighbours* (KNN), Floresta Aleatória (ou *Random Forest*), *Linear Discriminant Analysis* (LDA), *Quadratic Discriminant Analysis* (QDA), entre outros. Esses algoritmos estão descritos com mais detalhes em James et al. (2014), Kasambara (2017), Lantz (2019).

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 foram utilizados o Algoritmo Genético Elitista (AGE) juntamente com o KNN e com a Floresta Aleatória (algoritmos de aprendizado de máquina) para classificar os estatutos de cooperativas de crédito brasileiras. Estes algoritmos foram aplicados em 138 estatutos de onde 8.293 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 (quando não se enquadrava em nenhuma das anteriores). Os algoritmos de aprendizado de máquina foram utilizados 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, os algoritmos de aprendizado de máquina nos fornecem 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 as versões dos algoritmos de aprendizado de máquina utilizadas, na Seção 3 o problema de classificação dos estatutos é modelado, na Seção 4 é apresentado os resultados numéricos obtidos e a Seção 5 é composta pela conclusão e considerações finais.

AGE, KNN, Árvores de Classificação e Floresta Aleatória

Algoritmo Genético Elitista

O Algoritmo Genético descrito em Holland (1975), é uma ferramenta computacional que tenta emular o processo evolucionário de Darwin, o qual utiliza três estágios: Seleção, Cruzamento (o qual possui um parâmetro chamado probabilidade de cruzamento, p_c) e Mutação (o qual possui um parâmetro chamado probabilidade de mutação, p_m). No AGE homogêneo os parâmetros p_c e p_m são mantidos fixos durante a execução do algoritmo. 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 $Z = \{(u_1, u_2, ..., u_N); u_i \in D, i = 1, 2, ..., N\}$ 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 (1994) demonstrou que isso não acontece quase certamente (ou seja, com probabilidade 1) e apresentou o Algoritmo Genético Elitista (AGE) 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 Campos, Pereira e Cruz (2012), Pereira e Andrade (2015), Pereira et al. (2018), Pereira et al. (2020).

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 tratado neste artigo é um problema de classificação supervisionado, portanto a partir de agora só será tratado das versões dos algoritmos aplicados à classificação.

O algoritmo KNN

Em Lantz (2019), vemos que o KNN tem como objetivo ser um método para estimar a função de classificação para uma dada configuração das covariáveis X, com base nas respostas Y dos K-vizinhos mais próximos ao vetor X.

Assim, o algoritmo conhecido como o 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 (usando a distância euclidiana) e atribui-se a esse ponto a mesma classificação da maioria das classificações dos K pontos selecionados.

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

3. A medida usada para verificar a eficiência é a proporção de acertos (validação cruzada).

Dessa forma, se a classificação do ponto $x = (x_1, x_2, ..x_n)$ do conjunto de teste (T) dado pelos K pontos do conjunto de treinamento mais próximos a x é y_x e $\hat{y}_x$ é a classificação correta de x, então a validação cruzada é dada por:

$$CV(T) = \sum_{x \in T} I_{(y_x = \hat{y}_x)},$$

onde $I_{(x=y)}$ é a função indicadora.

Árvore de Classificação

Em Lopes (2018), vemos que a ideia do método não paramétrico das árvores de regressão (classificação) é a criação de uma partição do espaço das covariáveis em regiões distintas e disjuntas, ou seja, $R_1, R_2, ..., R_j$. Essas partições são construídas de modo que nenhuma delas seja vazia, i.e., cada uma delas possua pelo menos um ponto do conjunto de amostras (das covariáveis). Na verdade essas partições são escolhidas de modo a ser a mais homogênea possível, i.e., possuir uma quantidade de pontos aproximadamente iguais. Como nosso problema é de classificação, diferentemente daquele apresentado em Lopes (2018) que trabalhou com um problema de regressão, a predição para a resposta Y de uma observação com as covariáveis $X = (X_1, ..., X_n)$ que estão em R_k é dada pela classificação dominante das respostas que estão no conjunto de treinamento e que tem suas covariáveis X em R_k. Ilustrativamente, se o conjunto de treinamento é T e queremos classificar a resposta associada a amostra $x = (x_1, ..., x_n)$ onde x_i é a amostra da covariável X_i, observamos a que região R_i esse ponto x pertence. Uma vez detectado que $x \in R_p$ observamos quais amostras do conjunto de treinamento estão em R_p, ou seja, $B_p = \{x_{p_i} / x_{p_i} \in T \cap R_p\}$. Como estamos tratando de algoritmos supervisionados, sabemos a resposta associada a cada x_{p_i}, sejam, y_{p_i} tais respostas (classificações). Dessa forma, a classificação de x será dada por aquela classificação que mais aparece em R_p.

Floresta Aleatória

Ainda segundo Lopes (2018), o método das Florestas Aleatórias consiste em criar N árvores aleatórias. A criação dessas árvores se dá pelo processo de *bootstrap* onde N conjuntos de treinamento são obtidos da amostra original e para cada conjunto de treinamento uma árvore de classificação é criada. O processo de classificação da resposta associada à amostra $x = (x_1, ..., x_n)$ se dá da seguinte forma: a amostra $x = (x_1, ..., x_n)$ resultará em uma classificação dada por cada árvore, ou seja, teremos N classificações para cada amostra. Novamente, a classificação da amostra $x = (x_1, ..., x_n)$ será aquela classificação que mais aparece nas N classificações obtidas pelas N árvores de classificação.

A principal ideia é modificar o método de criação das árvores para que as mesmas se tornem diferentes umas das outras, com objetivo de criar árvores não correlacionadas.

A validação cruzada é dada pelo número de acertos de classificação quando utilizamos os dados do conjunto de teste T. Se $\hat{y}_x$ é a classificação correta de x e y_x é a classificação dada pela Floresta Aleatória, então a validação cruzada será

$$CV(T) = \sum_{x \in T} I_{(y_x = \hat{y}_x)}.$$

Modelagem do problema

O objetivo deste artigo é mostrar que é possível 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. Os algoritmos de aprendizado de máquina são usados 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 foi realizada pelo programa Python versão 3.10.6 e os algoritmos AGE, KNN e Floresta Aleatória foram implementados no programa R na versão 4.2.2, utilizando a *interface* 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 (no nosso caso 138 estatutos) e, por fim, obtém-se o número de acertos, ou a proporção de acertos. Só para deixar claro o que significa esse número de acertos, vemos que para cada ponto do conjunto teste é gerado com ajuda do conjunto de treinamento uma classificação. Se a classificação dada ao ponto coincide com a classificação verdadeira (estamos trabalhando com um problema supervisionado) contamos um acerto de classificação. Somando o número de acertos referentes a todos os pontos do conjunto de teste temos o número total de acertos. Esse número de acertos é dado pela validação cruzada. Cada subconjunto de palavras utilizado fornece um resultado diferente para o número de classificações corretas. Quando nos referirmos a Floresta Aleatória tradicional, estaremos nos referindo de como o algoritmo disponibilizado no R funciona em sua forma predefinida, ou seja, o número de características utilizadas na construção das árvores é igual a raiz quadrada do número total de características (no nosso caso as caractetísticas refere-se ao banco de palavras, ou seja, 8.293 palavras), são usadas 500 árvores em cada floresta e a validação cruzada utilizada é a descrita na Seção .

Assim é possível observar 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 classificação, 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 onde x_{ij} são as quantidades de vezes que cada palavra P_j aparece no estatuto i. Note que na última coluna aparece as classificações corretas conhecidas, uma vez se trata de um problema supervisionado.

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

Fonte: dos autores

Modelando o conjunto discreto e a função objetivo

Como colocado na Seção , 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 e na Floresta Aleatória pela classificação dominante gerada pelas árvores de classificação.

4. Depois de estabelecida as classificações verifica-se quantas delas eram corretas, obtendo assim a validação cruzada desse conjunto de estatutos (conjunto teste), 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 e pela Floresta Aleatória (validação cruzada do conjunto de teste). O AGE utiliza essa função f como a função objetivo a fim de obter a solução ótima, ou seja, o conjunto de palavras que gera o maior número de acertos de classificação com o KNN e com a Floresta Aleatória.

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 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 classificações corretas de uma fase para outra não diminui. Isso acontece devido ao fato do AGE estar maximizando a função que conta a quantidade de acertos (validação cruzada). Uma vez que se está sempre retornando com a população inicial, sempre haverá a possibilidade do conjunto de palavras ótimo estar presente em algum momento na população e esse não será perdido pelo uso do AGE.

Neste trabalho foram utilizados 138 estatutos de cooperativas brasileiras de onde foram retirados um conjunto inicial de 8.293 palavras (palavras que constam nos estatutos). O fato do AGE trabalhar com o conjunto de palavras no sentido de buscar um conjunto mínimo que maximize o número de classificações corretas, faz com que nenhum tratamento preliminar no conjunto de palavras seja necessário. Os algoritmos de aprendizado de máquina usados foram o KNN com parâmetro $K = 1$ com LOOCV como validação cruzada e a Floresta Aleatória tradicional com validação cruzada dada na Subseção . O AGE usado teve como probabilidade de cruzamento $p_c = 0.5$, probabilidade de mutação $p_m = 0.4$ e na primeira tentativa foi utilizada a quantidade de passos de cada etapa igual a 30. Na Tabela 2 resumimos os resultados obtidos depois de executadas essas quatro etapas.

Tabela 2: Resultados obtidos no final da Simulação

	Proporção de acertos	Número de palavras usadas
KNN	81,88%	1037
Floresta Aleatória	82,25%	1936

Fonte: dos autores

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).

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 (maximizar a validação cruzada do conjunto de teste). 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 utilizando o KNN 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 utilizada usando o Floresta Aleatória foi o descrito na Seção . A classificação do elemento do conjunto de teste usando o KNN foi determinado pelo estado do ponto do conjunto de treinamento que está mais próximo do elemento em análise com taxa de acerto muito boa, mais de 81% para essa primeira abordagem . Quando a Floresta Aleatória foi usada a taxa de acerto foi maior que 82%. Houve uma maior redução no número de palavras usando o KNN (1.037) se comparado ao Floresta Aleatória (1.936), contudo houve uma maior acertabilidade no uso da Floresta Aleatória 82,25% em relação ao KNN, 81,88%.

Foi possível então 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 os algoritmos de classificação sejam o KNN ou a Floresta Aleatória, abrindo assim várias possibilidades para outros trabalhos.

Referências

ACOSTA-GONZáLEZ, E.; FERNáNDEZ-RODRíGUEZ, F. Model selection via genetic algorithms illustrated with cross-country growth data. *Empirical Economics*, n. 33, p. 313–337, 2007.

BOUCHARD, M.J.; ROUSSELIèRE, D.; GUERNIC, M. Le. *Conceptual Framework for the purpose of Measurement of Cooperatives and its Operationalization*. [S.l.], 2017. Disponível em: ⟨https://shorturl.at/xLQZ8⟩.

BOUCHARD, M. J. et al. Statistics on cooperatives: concepts, classification, work and economic contribution measurement. *ILO, CIRIEC, COPAC: Geneva, Switzerland*, 2020.

CAMPOS, V.S.M.; PEREIRA, A.G.C.; CRUZ, J.A. Rojas. Modeling the genetic algorithm by a non-homogeneous markov chain: Weak and strong ergodicity. *Theory of Probability and its Applications*, v. 57, p. 185–192, 2012.

EUM, H.; CARINI, C.; BOUCHARD, M. J. Classification of cooperatives. a proposed typology. *Statistics on cooperatives: Concepts, classification, work and economic contribution measurement*, p. 13–22, 2020.

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

JAMES, G. et al. *An introduction to statistical learning with R applications*. [S.l.]: Springer, 2014.

KASAMBARA, A. *Machine Learning Essentials : Practical Guide in R*. [S.l.]: Published by STHDA, 2017, 2017.

KI-MOON, B. *Organização das Nações Unidas (ONU), Secretário Geral (2007-2017: Ban Ki-Moon). Mensagem do Secretário Geral por ocasião da celebração do Dia Mundial da Alimentação. 16 out 2012*. 2012. Online. Acessado em 19/02/2024, ⟨https://shorturl.at/qruI7⟩.

LACERDA, E. G.; CARVALHO, A. C.; LUDERMIR, T. B. Model selection via genetic algorithms for rbf networks. *Journal of Intelligent & Fuzzy Systems, IOS Press*, v. 13, p. 111–122, 2002.

LANTZ, B. *Machine Learning with R : Expert techniques for predictive modeling*. [S.l.]: Packt, Birmingham, 2019.

LOPES, L. P. Predição do preço do café naturais brasileiro por meio de modelos de statistical machine learning. *Sigmae, Alfenas*, v. 7, p. 1–16, 2018.

OCB. *Ramos do Cooperativismo - conheça nossa nova organização. Brasília,*. [S.l.], 2019. Acessado em 19/02/2024, ⟨https://shorturl.at/cDZ19⟩.

PATERLINI, S.; MINERVA, T. Regression model selection using genetic algorithms. In: *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)*. [S.l.: s.n.], 2010. p. 19–27.

PEREIRA, André GC et al. On the convergence rate of the elitist genetic algorithm based on mutation probability. *Communications in Statistics-Theory and Methods*, Taylor & Francis, v. 49, n. 4, p. 769–780, 2020.

PEREIRA, A. G. C.; ANDRADE, B.B. On the genetic algorithm with adaptive mutation rate and selected statistical applications. *Computational Statistics (Zeitschrift)*, v. 30, p. 131–150, 2015.

PEREIRA, A. G. C. et al. Convergence analysis of an elitist non-homogeneous genetic algorithm with crossover/mutation probabilities adjusted by a fuzzy controller. *Chilean Journal of Statistics*, v. 9, p. 19–32, 2018.

RUDOLPH, G. Convergence analysis of canonical genetic algorithms. *IEEE Transactions on Neural Networks*, v. 5, p. 96–101, 1994.