SELEÇÃO DE CARACTERÍSTICAS VIA ALGORITMO GENÉTICO PARA CLASSIFICAÇÃO COM REDES NEURAIS ARTIFICIAIS

Josias Alexandre Oliveira*
Departmento de Informática
Universidade Federal do Espírito Santo
Vitória, Brasil
josiasalexandre@gmail.com

Vinicius Ferraço Arruda†
Departmento de Informática
Universidade Federal do Espírito Santo
Vitória, Brasil
viniciusferracoarruda@gmail.com

15 de Outubro de 2019

1 Introdução

Métodos de aprendizado de máquina têm seu desempenho fortemente relacionado aos dados apresentados na fase de treino, logo, o pré-processamento dos *datasets* (conjuntos de dados) pode melhorar os resultados. Cada amostra de um determinado *dataset* possui um conjunto de *features* (atributos ou características), podendo variar de dezenas a milhares delas. O grande número de *features* geralmente dificulta o trabalho dos algoritmos de aprendizado de máquina devido o aumento em sua complexidade. É comum que dados possuam *features* redundantes, i.e., podem ser representados por um ou mais *features* através de uma combinação linear, e.g., peso e massa. Além disso, determinadas *features* podem não ser significantes para a tarefa final do modelo, e.g., deseja-se obter um modelo para classificar pessoas saudáveis baseadas em algumas *features* pessoais, onde a *feature* cor de cabelo provavelmente poderia ser descartada. Com isso, surge a necessidade de selecionar as *features* das amostras que são mais relevantes. Neste trabalho, será apresentado um estudo para seleção de *features* utilizando o algorítmo genético para então treinar uma rede neural para classificação.

2 Método

Este trabalho é baseado em [1]. Os autores realizaram a seleção de *features* através da técnica *Improved Salp Swarm Algorithm* (ISSA). Assim como o *Particle Swarm Optimization* (PSO), este algoritmo também é baseado em processos biológicos nos quais populações de indivíduos encontram conjuntamente uma solução adequada para um dado problema. Nesse contexto, a função de *fitness* para cada agente é computada através do classificador *K Nearest Neighbor* (KNN). O procedimento consiste em gerar agentes que representam, cada um, uma seleção de características de um dataset específico. Para avaliar a *fitness* de cada agente, utilizou-se uma métrica com base no resultado da classificação do KNN e na quantidade de *features* usadas. Dessa forma, agentes que utilizem poucas *features* e que permitem classificação com maior acurácia são considerados como sendo os melhores.

Neste trabalho, o algoritmo ISSA foi substituído pelo algoritmo genético (GA) e a função de *fitness* de cada indíviduo se dá por uma rede neural artificial. As métricas de avaliação da *fitness* permaneceram as mesmas: a acurácia e o número de *features* utilizadas são somadas para se obter o resultado final da *fitness* de cada indivíduo. Utilizamos os pesos ρ e φ para multiplicar a acurácia e o número de *features* selecionadas respectivamente, dessa forma pode-se definir a importância de cada componente no cálculo da *fitness*.

Apesar de utilizarmos algoritmos diferentes, procuramos utilizar meta-parâmetros similares. A Tabela 1 relaciona os parâmetros utilizados neste trabalho e os valores equivalentes em [1].

A Subseção 2.1 apresenta os detalhes da configuração do algoritmo genético utilizado em nossos experimentos e a Subseção 2.2 apresenta detalhes das redes neurais empregadas na classificação.

*github.com/josiasalexandre
†viniciusarruda.github.io

Tabela 1: Parâmetros utilizados.

Parâmetro	Valor ISSA	Valor atual	Significado
NRuns	20	20	Número de execuções
NIters	50	50	Número máximo de iterações (ISSA e GA)
NAgents	10	10	Tamanho da população
Domínio de Busca			Vetor binário [0,1]
cr	–	0.9	Taxa de cruzamento
mt	–	0.1	Taxa de mutação
N_{in}	–	*dataset*[1]	Quantidade de neurônios na entrada
N_{out}	–	*dataset*[2]	Quantidade de neurônios nas saídas
N_{hid}	–	$\sqrt{N_{in} * N_{out}}$	Quantidade de neurônios ocultos
ρ	0.9	0.9	Peso aplicado no erro da acurácia
φ	0.1	0.1	Peso aplicado na soma da quantidade de *features*
k	10	20	Proporção de dados destinados para teste

2.1 Algoritmo Genético

Neste trabalho foi implementado o algoritmo genético na seguinte configuração:

- Inicialização do indivíduo aleatoriamente com distribuição uniforme.
- Uso da técnica de seleção por torneio com $s = 2$.
- Taxa de cruzamento de 0.9 com recombinação em dois pontos aleatórios.
- Taxa de mutação de 0.1 com a troca de um bit aleatóriamente.
- 10 indivíduos.
- 50 épocas.
- A função de *fitness* é computada através da soma do erro da acurácia com a porcentagem do número *features* selecionadas, ponderadas pelos parâmetros ρ e φ: $\rho * (1 - acc) + \varphi * {}^{|F|}/_{|T|}$, onde acc é a acurácia, $|F|$ é o número de *features* selecionadas e $|T|$ é o número total de *features*.

2.2 Redes Neurais Artificiais

A rede neural artificial utilizada é de arquitetura totalmente conectada com uma única camada oculta. Foram utilizados os métodos de treinamento *Extreme Learning Machine* (ELM) e *Backpropagation*. Em ambos os casos, a quantidade de neurônios na camada de entrada é sempre igual à quantidade de *features* presentes no *dataset* N_{in} e a quantidade de neurônios na camada de saída é sempre igual à quantidade de classes N_{out}. Para a camada oculta, utilizamos a heurística definida em [2] para estabelecer a quantidade de neurônios N_{hid} a serem inseridos. O valor é computado através da fórmula $N_{hid} = \sqrt{N_{in} * N_{out}}$. Como função de transferência, utilizamos a sigmóide para o modelo treinado com o algoritmo ELM e a ReLU para o modelo treinado com o *Backpropagation*.

3 Experimentos

O experimento consistiu em comparar o desempenho das redes neurais diretamente aplicadas aos *datasets* originais, i.e. *baseline*, com aquele obtido pelas redes quando utilizada a abordagem de seleção de *features* através de algoritmo genético e também com os resultados apresentados no trabalho de [2] pelo método ISSA. As estatísticas foram obtidas executando os experimentos por 20 vezes.

3.1 Datasets

Cada *datasets* foi embaralhado e separado em dois subgrupos de dados para treinamento e testes nas proporçoes de 80% e 20%, respectivamente. Além disso, todos os *datasets* foram normalizados para o intervalo $[-1, 1]$. Os conjuntos

[1] Equivalente ao número de *features* do *dataset*.

[2] Equivalente ao número de classes do *dataset*.

de dados utilizados neste trabalho foram *Hepatitis*, *BreastEW* e *Multiple Features* disponíveis no repositório público da UCI[3]. A Tabela 2 descreve os dados..

Tabela 2: *Datasets* utilizados nos experimentos.

Nome	Nº. *Features*	Nº. Amostras	Nº. Classes
BreastEW	19	155	2
Hepatitis	30	569	2
Multiple Features	649	2000	10

4 Resultados

Tabela 3: Melhor valor de *fitness* das 20 execuções dos experimentos.

Dataset	GA+ELM	GA+NN	ELM	NN	ISSA [1]
Hepatitis	0.1062	0.1417	0.1871	0.2742	**0.0684**
BreastEW	0.0351	0.0729	0.1395	0.1711	**0.0160**
Multiple Features	0.0873	**0.0491**	0.1810	0.1067	0.0820

Tabela 4: Média e desvio padrão das *fitness* das 20 execuções dos experimentos.

Dataset	GA+ELM	GA+NN	ELM	NN	ISSA [1][4]
Hepatitis	0.1290 ± 0.0116	0.2755 ± 0.1191	0.2727 ± 0.0523	0.5515 ± 0.1940	$\mathbf{0.1100} \pm -$
BreastEW	0.0495 ± 0.0063	0.1460 ± 0.0804	0.2737 ± 0.0999	0.4051 ± 0.1566	$\mathbf{0.0240} \pm -$
Multiple Features	0.0932 ± 0.0035	$\mathbf{0.0542 \pm 0.0024}$	0.2002 ± 0.0116	0.1207 ± 0.0081	$0.1220 \pm -$

Tabela 5: Pior valor de *fitness* das 20 execuções dos experimentos.

Dataset	GA+ELM	GA+NN	ELM	NN	ISSA [1]
Hepatitis	0.1444	0.5574	0.3613	0.8839	**0.1282**
BreastEW	0.0624	0.3429	0.5579	0.7237	**0.0320**
Multiple Features	0.1018	**0.0590**	0.2305	0.1427	0.1430

As Tabelas 3, 4 e 5 mostram os melhores, as médias e os piores valores para as *fitness*, respectivamente, das 20 execuções dos experimentos. Conclui-se que para os *datasets Hepatitis* e *BreasEW* o método proposto por [1] se mostrou mais eficaz, no entanto, para o *dataset Multiple Features* o método GA+NN se mostrou superior. Conjecturamos que este resultado é diretamente relacionado ao número de amostras presente nos *datasets* devido a natureza das redes neurais desempenharem melhor com maior volume de dados.

Como mostra o *boxplot* da *fitness* na Figura 1, para os *datasets Hepatitis* e *BreastEW* o padrão dos resutlados se manteve, no entanto, com o *dataset* com maior número de amostras, os métodos baseados em redes neurais treinados com *backpropagation* se mostraram melhores. A Figura 5 (ao final do documento) ilustra a matriz de confusão dos métodos apresentados neste trabalho com melhor *fitness* dentre as 20 execuções de cada *dataset*.

A Tabela 6 apresenta a média e o desvio padrão do número de *features* encontradas no experimento, sendo também apresentados na Figura 2 em forma de *boxplot*. O método GA+ELM mostrou maior eficácia para o *dataset BreastEW*, porém, o ISSA [1] selecionou as *features* com maior eficácia nos outros dois *datasets*.

A Tabela 7 apresenta a média e o desvio padrão da acurácia de classificação do experimento. O método ISSA se mostrou superior para os *datasets Hepatitis* e *BreastEW*. Para o *Multiple Features*, o uso da rede neural convencional, com todas as *features* se mostrou melhor. Nota-se que a perda de desempenho ao utilizar o método GA+NN é pequena e tem a vantagem de utilizar menos da metade do número de *features*.

[3] `https://archive.ics.uci.edu/ml/`

[4] Desvio padrão não disponibilizado no artigo.

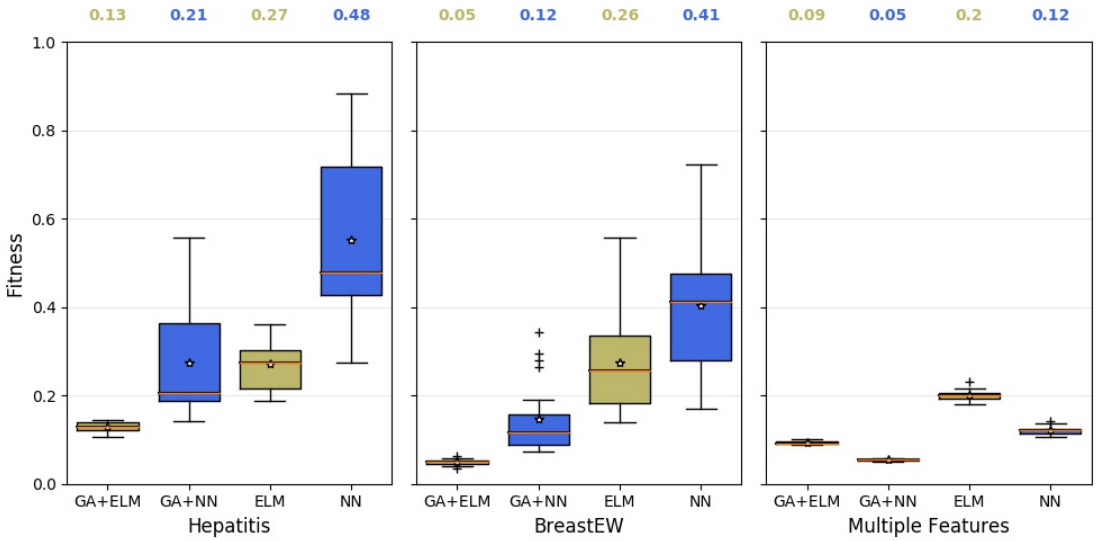


Figura 1: *Boxplot* da *fitness* dos experimentos.

Tabela 6: Média e desvio padrão do número de *features* selecionadas das 20 execuções dos experimentos.

Dataset	GA+ELM	GA+NN	ELM	NN	ISSA [1][5]
Hepatitis	4.10 ± 1.1790	5.60 ± 2.6721	19 ± 0	19 ± 0	$\mathbf{3.40} \pm -$
BreastEW	$\mathbf{7.20 \pm 1.7776}$	12.00 ± 3.3015	30 ± 0	30 ± 0	$12.91 \pm -$
Multiple Features	310.80 ± 15.8417	297.40 ± 9.8864	649 ± 0	649 ± 0	$\mathbf{125.47} \pm -$

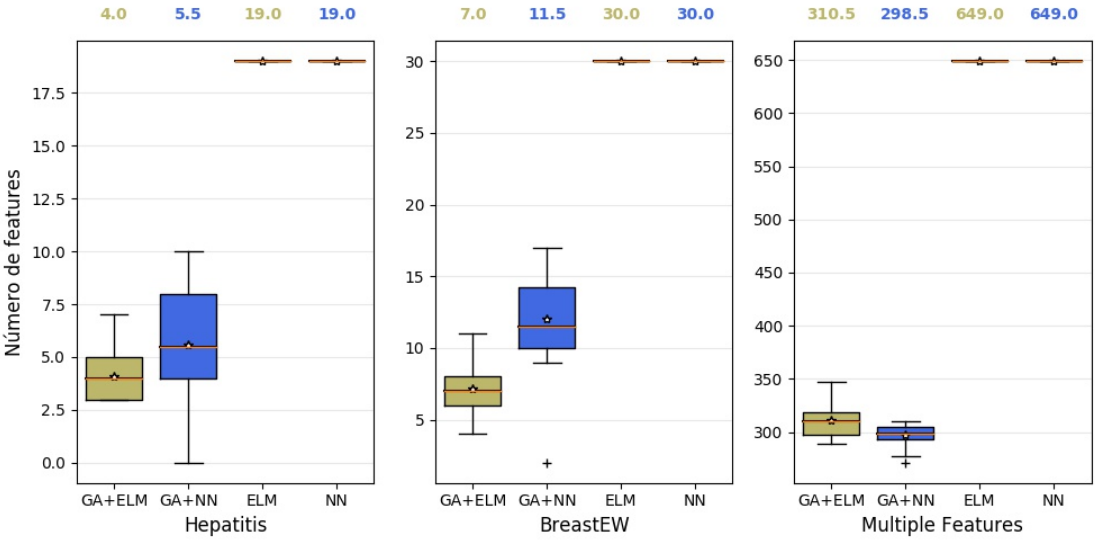


Figura 2: Boxplot do número de *features* dos experimentos.

A Figura 3 mostra os resultados da acurácia em *boxplot*, indicando um mesmo padrão para os *datasets Hepatitis* e *BreastEW*. Já para o *Multiple Features*, os métodos baseados em *backpropagation* se mostraram melhores.

A Tabela 8 apresenta a média e o desvio padrão da execução dos experimentos. Devido à sua natureza, o algoritmo de treinamento ELM se mostrou mais eficiente em relação aos demais em todos os *datasets* utilizados. Note que é

[5]Desvio padrão não disponibilizado no artigo.

Tabela 7: Média e desvio padrão da acurácia de classificação das 20 execuções dos experimentos.

Dataset	GA+ELM	GA+NN	ELM	NN	ISSA [1][6]
Hepatitis	0.8258 ± 0.0472	0.6855 ± 0.1536	0.8081 ± 0.0581	0.4984 ± 0.2155	$\mathbf{0.9123} \pm -$
BreastEW	0.8478 ± 0.1079	0.7596 ± 0.1141	0.8070 ± 0.1110	0.6610 ± 0.1740	$\mathbf{0.9610} \pm -$
Multiple Features	0.8971 ± 0.0163	0.9716 ± 0.0069	0.8886 ± 0.0129	$\mathbf{0.9770 \pm 0.0090}$	$0.9611 \pm -$

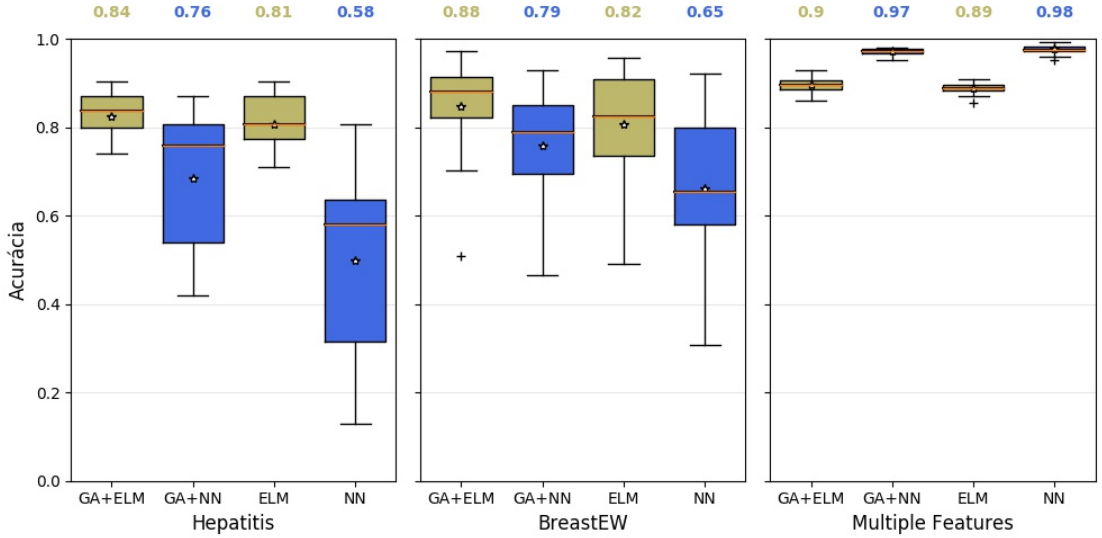


Figura 3: Boxplot da acurácia dos experimentos.

Tabela 8: Média e desvio padrão do tempo de execução das 20 execuções dos experimentos.

Dataset	GA+ELM	GA+NN	ELM	NN	ISSA [1][6]
Hepatitis	0.7011 ± 0.0083	5.7907 ± 0.0526	$\mathbf{0.0189 \pm 0.0015}$	0.0529 ± 0.0020	$23.98 \pm -$
BreastEW	1.0245 ± 0.0147	12.7208 ± 0.1114	$\mathbf{0.0214 \pm 0.0013}$	0.0944 ± 0.0016	$45.39 \pm -$
Multiple Features	244.67 ± 14.703	2263.81 ± 187.88	$\mathbf{0.5168 \pm 0.0782}$	4.2208 ± 0.3542	$475.84 \pm -$

inapropriado comparar o tempo de execução com o método ISSA, uma vez que foi executado em uma máquina distinta da utilizada nos experimentos deste trabalho.

A Figura 4 mostra o *boxplot* do tempo de execução dos métodos. Em todos os *datasets* o método GA+NN se mostrou menos eficiente. A diferença do tempo de execução entre os *datasets* está diretamente relacionado ao tamanho do *dataset*.O algoritmo genético foi o responsável pelo aumento significativo do tempo de execução quando comparado ao *baseline*. Já os métodos baseados no *backpropagation* foram os superiores em tempo de execução devido à sua natureza de treinamento iterativa.

5 Conclusão

Os resultados indicaram que o uso do algoritmo genético para seleção de *features* é promissor. No entanto, um estudo mais aprofundado com relação aos hiper-parâmetros e configurações de implementação pode melhorar os resultados. De modo semelhante, os métodos de classificação utilizados também devem ter seus hiper-parâmetros e estratégias de implementação explorados, como por exemplo, outro método de calcular o número de neurônios na camada oculta.

Dentre os métodos de treinamento de redes neurais, o ELM mostrou ser o mais adequado para uso combinado ao algoritmo genético, pois o tempo de execução é inferior em relação ao *backpropagation*.

[6]Desvio padrão não disponibilizado no artigo.

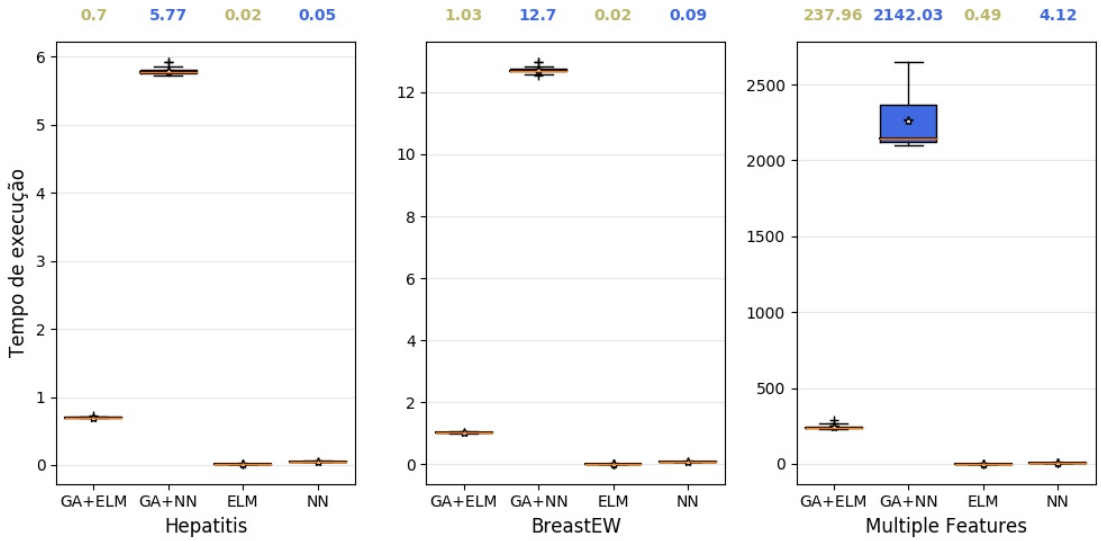


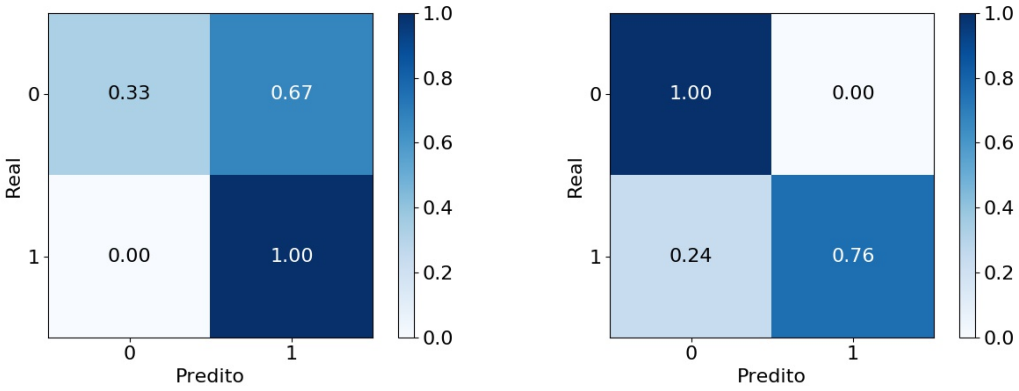
Figura 4: Boxplot do tempo de execução dos experimentos.

Por fim, o trabalho proposto em [1] foi superior para *datasets* com pouco número de amostras, porém, com tempo de execução maior do que todos os outros métodos quando aplicados aos mesmos *datasets*.

Referências

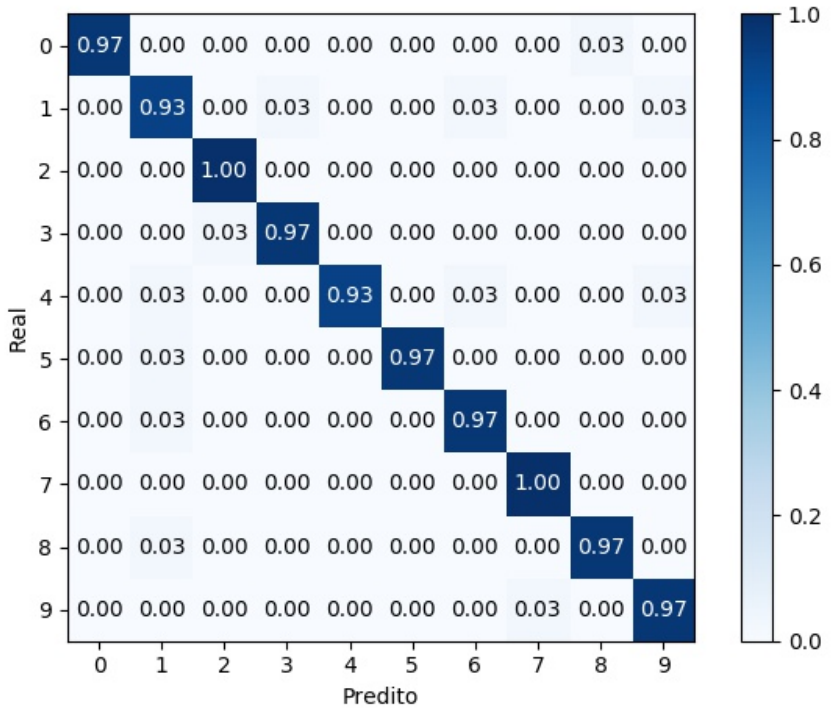
[1] Ah E Hegazy, MA Makhlouf, and Gh S El-Tawel. Improved salp swarm algorithm for feature selection. *Journal of King Saud University-Computer and Information Sciences*, 2018.

[2] Timothy Masters. *Practical neural network recipes in C++*. Morgan Kaufmann, 1993.



(a) *Hepatitis* - GA+ELM

(b) *BreastEW* - GA+ELM



(c) *Multiple Features* - GA+NN

Figura 5: Matriz de confusão dos melhores resultados apresentados pelos métodos deste trabalho em cada *dataset*.