Plant Classification Using Weighted k-NN Variants

Larissa F. S. Britto[1], **Luciano D. S. Pacifico**[1]

[1]Departamento de Computação – DC
Universidade Federal Rural de Pernambuco – UFRPE
Recife, Pernambuco, Brasil

{larissa.feliciana, luciano.pacifico}@ufrpe.br

Abstract. *Automatic plant species identification is a difficulty challenge for botanical taxonomy field. Many works have been proposed towards the development of automatic plant species recognition systems through machine learning methods. One of the most popular algorithms for plant classification is the k-Nearest Neighbor (k-NN), given its simplicity and robustness. In this work, we evaluate the performance of two improved weighted k-NN algorithms when dealing with plant classification task. Experimental evaluation includes three real-world data sets obtained from different image processing and feature extraction processes. Also, a statistical hypothesis test is employed to perform an overall evaluation of the selected models.*

Resumo. *A identificão automática de espécies de plantas é um grande desafio na taxonomia botânica. Vários trabalhos têm sido propostos visando o desenvolvimento de sistemas automáticos de reconhecimento de plantas através da aprendizagem de máquina. Um dos algoritmos mais populares na classificação de plantas é o dos k-Vizinhos mais Próximos (k-NN), dada sua simplicidade e robustez. Neste trabalho, a performance de duas variações ponderadas do k-NN é avaliada no cenário de classificação de plantas. A avaliação experimental inclui três bases de dados reais obtidas por diferentes técnicas de processamento de imagens e extração de características. Uma avaliação geral é executada através do uso de testes estatísticos.*

1. Introdução

As plantas realizam um papel importante na natureza, executando muitas funções insubstituíveis na Terra. As plantas são a maior fonte de oxigênio e alimentos para os animais, provendo ainda abrigo, vestimenta, combustível, medicamentos, etc. Tais organimos promovem o balanceamento ecológico em seus ambientes.

Embora em seu dia-a-dia muitas pessoas tenham contato com vários tipos de plantas, a identificação correta de espécies e famílias de plantas ainda é uma tarefa difícil. Tendo em vista a enorme quantidade de aplicações das plantas, a classificação de plantas é uma tarefa que desperta grande interesse tanto para a ciência quanto para a indústria. Mas, mesmo nos dias atuais, a identificação de espécies de plantas é uma tarefa realizada manualmente por especialistas no assunto, tornando essa atividade demorada e suscetível a erros humanos.

Com a evolução da tecnologia, cada vez mais áreas têm se beneficiado do uso de métodos avançados como a Aprendizagem de Máquina e Visão Computacional. A

identificação automática de espécies de planta vem tornando-se um desafio nos últimos anos, sendo uma área ativa de pesquisa tanto na comunidade de botânica quanto em computação. Alguns sistemas de reconhecimento automático de plantas foram propostos recentemente na literatura [Agarwal et al. 2006, Kumar et al. 2012, Mallah et al. 2013, Jin et al. 2015], mas apesar dos esforços, esse ainda é considerado um problema em aberto.

O método dos k-Vizinhos mais Próximos (k-*Nearest Neighbor*, ou k-NN) [Cover and Hart 1967] é um dos algoritmos mais usados em Aprendizagem de Máquina, promovendo uma classificação direta de um padrão cuja classe é desconhecida pela avaliação de sua vizinhança, composta pelos k padrões de treinamentos mais próximos ao mesmo. O voto majoritário na vizinhança é usado para definir qual classe será atribuída ao padrão desconhecido. Vários trabalhos em reconhecimento automático de plantas adotaram o k-NN como classificador [Kumar et al. 2012, Mallah et al. 2013, Mallah and Orwell 2013, Rahmani et al. 2015, Sahay and Chen 2016].

O algoritmo k-NN tradicional apresenta alguns problemas, uma vez que todos os k padrões de treinamentos na vizinhança do padrão cuja classe deseja-se inferir são ponderados igualmente na etapa de determinação da classe majoritária, o que pode levar a vários problemas de classificação, conforme o valor de k aumenta.

Neste trabalho, duas versões ponderadas do k-NN são empregadas à tarefa de classificação automática de plantas em uma tentativa de solucionar o problema da ponderação equalitária da vizinhança: O algoritmo dos k-Vizinhos mais Próximos Ponderado (*Weighted* k-*Nearest Neighbor*, ou W-k-NN) [Dudani 1976] e o algoritmo dos k-Vizinhos mais Próximos de Igualdade (k-*Nearest Neighbor Equality*, ou k-NNE) [Sierra et al. 2011]. A metodologia de avaliação empregada segue o modelo apresentado por [Rahmani et al. 2015], estendendo o trabalho desses autores pelo acréscimo de duas versões melhoradas por ponderação do k-NN, assim como pela inclusão da análise de duas outras bases de dados reais de plantas. Os principais objetivos deste trabalho são:

1. Apresentar as principais características do k-NN e de suas variações ponderadas;
2. Analisar o comportamento dos algoritmos de classificação em bases de dados de plantas, sendo tais bases obtidas por diferentes técnicas de extração de características;
3. Comparar, através do uso de testes de hipóteses estatísticos, o desempenho dos classificadores selecionados.

O trabalho está dividido da seguinte forma: Os algoritmos k-NN, W-k-NN e k-NNE serão brevemente apresentados (Seção 2); As bases de dados adotadas serão descritas na Seção 3; Os resultados experimentais serão avaliados em seguida (Seçaõ 4); Finalmente, as conclusões e possíveis linhas para pesquisas futuras serão apresentadas (Seção 5).

2. k-NN e k-NN Ponderados

A regra do k-NN padrão é bastante simples. Dado um conjunto θ_S contendo S padrões rotulados distribuídos em M classes $\theta_S = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_S, y_S)\}$, onde um padrão é representado por um vetor de características $\mathbf{x}_i \in \Re^n$ e o rótulo da classe de cada padrão é representado por $y_i \in \{\beta_1, \beta_2, \ldots, \beta_M\}$, a regra do k-NN para classificar um padrão

não rotulado **x** é pela afetação de **x** à classe com maior frequência (ou seja, classe majoritária) entre seus k vizinhos mais próximos do conjunto de treinamento θ_S. O processo de treinamento nesse algoritmo ocorre apenas pelo armazenamento dos dados rotulados de treinamento para comparação futura. O algoritmo k-NN é apresentado no Algorithm 1.

Algorithm 1 k-Vizinhos mais Próximos

para cada padrão não rotulado **x faça**
Calcule a distância entre **x** e cada dado de treinamento $\mathbf{x}_i$.
Ordene os padrões de treinamento de acordo com a distância em relação à **x**, em ordem crescente.
Pegue os primeiros k padrões do conjunto ordenado de treinamento para compor a vizinhaça D de **x**.
Associe x à classe majoritária em D.
fim_para

Para a formação da vizinhança D de **x**, uma função de distância precisa ser definida, sendo a Distância Euclidiana (eq. (1)) uma das medidas de dissimilariades mais adotadas para a comparação entre vetores de dados com valoração real.

$$d_2(\mathbf{x}, \mathbf{x}_i) = \sqrt{\sum_{j=1}^{n} (x_j - x_{ij})^2} \tag{1}$$

Uma das primeiras tentativas de resolver o problema da ponderação igualitária dos vizinhos mais próximos no k-NN tradicional foi proposta por Dunadi [Dudani 1976]. O algoritmo dos k-Vizinhos mais Próximos Ponderado (W-k-NN) atribui à vizinhança do padrão **x** pesos de acordo com suas distâncias ao mesmo. Os pesos atribuídos aos vizinhos de **x** são inversamente proporcionais ao quadrado da distância desses padrões a **x**, de acordo com a eq. (2).

$$w_i = \frac{1}{(d_2(\mathbf{x}, \mathbf{x}_i))^2} \tag{2}$$

No W-k-NN, o rótulo de classe y de **x** é determinado pela regra do voto majoritário (eq. (3)).

$$y = \arg\max_{\beta_l} \sum_{(\mathbf{x}_i, y_i) \in D} w_i \delta(y_i = \beta_l) \tag{3}$$

onde β_l é um rótulo de classe e $\delta(\cdot)$ é a função delta de Dirac que recebe valor 1 se o seu argumento for verdadeiro, ou 0, caso contrário. O algoritmo W-k-NN é apresentado no Algorithm 2.

Uma outra abordagem interessante de k-NN ponderado foi proposta por Sierra et al. [Sierra et al. 2011], chamado k-Vizinhos mais Próximos de Igualdade (*k-Nearest Neighbor Equality*, ou k-NNE). O k-NNE estende o k-NN tradicional por buscar para cada uma das classes do problema tratado os k padrões de treinamento mais próximos ao

Algorithm 2 k-Vizinhos mais Próximos Ponderado

para cada padrão não classificado **x faça**
Calcule a distância entre **x** e cada padrão de treinamento $\mathbf{x}_i$.
Ordene os padrões de treinamento de acordo com suas distâncias a **x**, em ordem crescente.
Pegue os primeiros k padrões do conjunto ordenado de treinamento para compor a vizinhaça D de **x**.
Calcule o peso de cada vizinho de **x** (eq. (2)).
Associe x à classe majoritária de acordo com a eq. (3).
fim_para

padrão não categorizado **x**. **x** é então atribuído à classe cuja distância média entre seus k padrões em relação a **x** seja a menor. O k-NNE é apresentado no Algorithm 3.

Algorithm 3 k-Vizinhos mais Próximos de Igualdade

Dado um padrão não classificado **x**:
para cada classe β_i **faça**
Selecione os k vizinhos mais próximos a **x** que pertençam à classe β_i.
Calcule a distância média d_i entre os padrões selecionados e **x**.
fim_para
Associe x à classe β_l tal que $d_l = \arg\min_{d_l}\{d_1, d_2, \ldots, d_M\}$

3. Bases de Dados

Nesta seção, as bases de dados adotadas nos experimentos serão apresentadas: Planta Iris (*Fisher's Iris Plant*), Núcleos das Sementes de Trigo (*Wheat Seed Kernels*) e 100 Folhas de Plantas (100 *Plant Leaves*). Todas as bases de dados são problemas reais obtidos através do UCI Machine Learning Repository [Asuncion and Newman 2007].

A base de dados 100 Folhas de Plantas foi dividida de modo que cada um de seus vetores de características (margem da folha, forma da folha e textura da folha) pudesse ser utilizado de forma independente para a classificação das espécies de plantas, assim como para a avaliação do potencial discriminatório de cada vetor de características e de suas combinações, resultando em um total de sete bases de dados formadas pelos dados desse problema. Essa abordagem também foi utilizada na avaliação do desempenho de classificadores na tarefa de reconhecimento automático de espécies de plantas realizada em [Rahmani et al. 2015].

3.1. Planta Iris

A base de dados da Planta Iris [Fisher 1936] é um dos problemas mais conhecidas em Reconhecimento de Padrões e Aprendizagem de Máquina. Essa base de dados é composta por 150 padrões igualmente distribuídos em 3 classes (50 instâncias por classe), onde cada classe se refere a um tipo de planta Iris (*Iris Setosa*, *Iris Versicolour* e *Iris Virginica*).

Cada padrão da base é descrito por um conjunto de quatro características: comprimento da sépala (em cm), largura da sépala (em cm), comprimento da pétala (em cm) e largura da pétala (em cm). O processo de extração de características adotado na

elaboração desta base foi o manual, sendo esse processo realizado por Edgar Anderson [Anderson 1935] em um mesmo dia, fazendo uso dos mesmos instrumentos. A primeira classe (*Iris Setosa*) é linearmente separável das duas outras, porém as classes *Iris Versicolour* e *Iris Virginica* não são linearmente separáveis entre si.

3.2. Núcleos das Sementes de Trigo

A base de dados Núcleos das Sementes de Trigo (*Wheat Seed Kernels*, ou *Seeds*) [Charytanowicz et al. 2010] contempla 210 amostras selecionadas aleatoriamente e igualmente distribuídas em 3 classes de trigo (*Kama*, *Rosa* e *Canadian*). Os autores usaram uma técnica de raio-X para a visualização da estrutura interna dos núcleos das sementes de trigo em seu experimento.

A base de dados é composta por 7 parâmetros geométricos dos grãos: área A (em mm^2), perímetro P (em m), compacidade C ($C = 4\pi A/P^2$), comprimento do núcleo (em mm), largura do núcleo (in mm), coeficiente de assimetria e comprimento do sulco do núcleo (in mm).

3.3. 100 Folhas de Plantas

A base de dados 100 Folhas de Plantas (*100 Plant Leaves*) [Mallah et al. 2013] apresenta 100 espécies de folhas de plantas (classes do problema). A base é composta por 1600 padrões, e para cada espécie existem 16 amostras (padrões). As amostras foram obtidas através de fotografias coloridas das folhas em um plano de fundo branco.

Para cada amostra, 3 vetores de características distintos foram extraídos: uma assinatura da Curva de Contorno do Centróide da forma da folha (Sha), um histograma interior da textura (Tex), e um histograma de escala fina da margem da folha (Mar). Detalhes do processo de extração adotado para cada característica da folha são apresentados em [Mallah et al. 2013].

A base 100 Folhas de Plantas apresenta um grande número de classes e uma baixa quantidade de amostras por classe, o que a caracteriza como um problema desafiador. Outros problemas apresentados por essa base estão relacionados ao fato de que muitas subespécies de folhas são bastante semelhantes a outras espécies, assim como espécies próximas de folhas podem apresentar aparências bastante distintas entre si [Mallah and Orwell 2013].

Algumas imagens binárias das silhuetas de amostras da base de dados 100 Folhas de Plantas são apresentadas na Fig.1.

4. Resultados Experimentais

Nesta seção, os resultados experimentais serão apresentados. Cinco classificadores provenientes da literatura de Aprendizagem de Máquina são comparados: Árvore de Decisão (*Decision Tree*, ou DT), Naïve Bayes (NB), k-Vizinhos mais Próximos (k-NN), k-Vizinhos mais Próximos Ponderado (W-k-NN), k-Vizinhos mais Próximos de Igualdade (k-NNE). Todas as variações do k-NN foram testadas com $k = 3, 4$ e 5 usando a distância euclidiana. Os testes com valores de $k = 6$ e 7 não foram incluídos, tendo em vista que os resultados apresentados por [Rahmani et al. 2015] apontaram baixas performances para esses valores de k. Todos os algoritmos foram implementados na linguagem

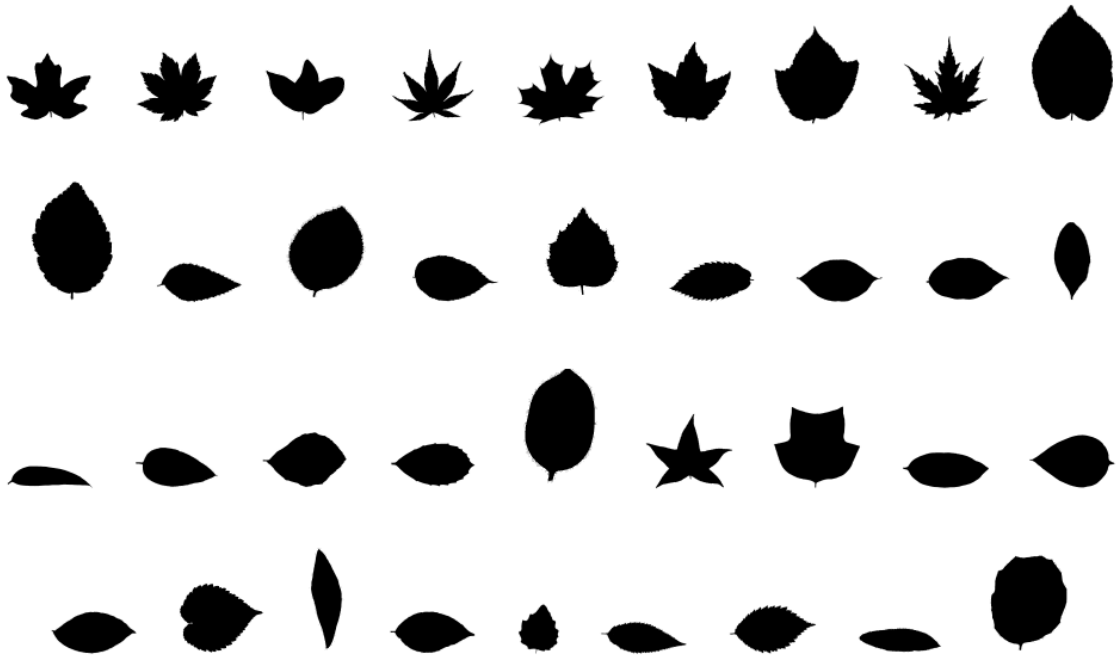

Figura 1. Exemplos de amostras de algumas espécies de plantas contidas na base de dados 100 Folhas de Plantas [Asuncion and Newman 2007, Mallah et al. 2013]

de programação Python, e todos os testes foram executados em um computador com uma CPU i5-5250U e 8 GB de RAM. Os algoritmos DT e NB foram implementados usando a biblioteca *scikit-learn* [Pedregosa et al. 2011, Buitinck et al. 2013] e seus valores de parâmetro *default*. Todas as variações do k-NN foram implementadas usando a biblioteca *TensorFlow* [Abadi et al. 2016].

Os experimentos foram conduzidos de acordo com um esquema do tipo validação cruzada com 10-*folds*, sendo cada *fold* determinada aleatoriamente. As bases de dados selecionadas são apresentadas resumidamente na Tabela 1.

A avaliação inclui uma análise empírica relativa à acurácia média no conjunto de testes e o tempo médio de execução em cada base de dados. A avaliação também inclui um sistema de *ranks* obtido através do uso do teste de Friedman [Friedman 1937] em

Tabela 1. Descrição das Bases de Dados de Plantas. Atributos: número de características; Classes: número de classes; Total: número total de padrões.

Base de Dados	Atributos	Classes	Total
Iris	4	3	150
Seeds	7	3	210
Mar (M)	64	100	1600
Sha (S)	64	100	1600
Tex (T)	64	100	1600
Mar e Sha (MS)	128	100	1600
Mar e Tex (MT)	128	100	1600
Sha e Tex (ST)	128	100	1600
Mar, Sha e Tex (MST)	192	100	1600

relação à avaliação global da acurácia média de teste. O teste de Friedman é um teste de hipóteses não-paramétrico que calcula valores de *ranks* para os algoritmos para cada base de dados separadamente. Se a hipótese nula de que os *ranks* não são significativamente diferentes for rejeitada, o teste de Nemenyi [Nemenyi 1962] é adotado com um teste *post hoc* para o teste de Friedman. De acordo com o teste de Nemenyi, a performance de dois algoritmos é considerada significativamente diferente se a diferença entre seus valores médios de *rank* for ao menos maior que uma *diferença crítica* dada por:

$$CD = q_a \sqrt{\frac{n_{alg}(n_{alg}+1)}{6n_{bases}}} \tag{4}$$

onde n_{bases} representa o número de bases de dados, n_{alg} representa o número de algoritmos comparados e q_a são valores críticos baseados em estatísticas nos limites do modelo t de *Student* divididas por $\sqrt{2}$ [Demšar 2006]. Como os experimentos foram realizados com $n_{data} = 9$ e $n_{alg} = 11$, temos que $CD = 5.0323$. Nos testes realizados, o grau de significância foi fixado em $\alpha = 0.05$ para o teste de Friedman. Como o objetivo é maximizar a acurácia no conjunto de testes, os melhores algoritmos irão obter os valores mais altos para o *rank* médio.

Os resultados experimentais são apresentados na Tabela 2.

Pelos resultados apresentados na Tabela 2, de acordo com uma análise empírica, podemos observar que o W-k-NN e o k-NNE foram os responsáveis pelos melhores desempenhos médios para as bases de dados selecionadas, no que diz respeito à acurácia média de teste.

Considerando a base 100 Folhas de Plantas, quando características individuais são analisadas, observamos que a forma das folhas (Sha) é a característica com menor poder discriminatório para os sistemas automáticos de reconhecimento. Uma vez que muitas espécies de plantas (por exemplo, espécies de plantas da mesma família) apresentam folhas com formato semelhante, todos os algoritmos avaliados apresentaram baixa acurácia média quando apenas a forma das folhas é levada em consideração. A melhor característica individual (isto é, a característica com maior poder discriminatório) para a base 100 Folhas de Plantas é a textura (exceto para NB, onde a margem foliar foi responsável pelas melhores precisões), onde os melhores métodos obtiveram desempenhos médios de 79.81%

Quase todos os classificadores selecionados apresentaram melhores precisões médias quando as características das folhas são combinadas duas a duas. A melhor combinação entre duas características para a maioria dos algoritmos avaliados foi obtida quando a textura e a margem foram combinadas (exceto para o algoritmo DT, onde a melhor acurácia média de teste foi obtida pela combinação das características de margem e forma, e NB, que alcançou os melhores desempenhos quando as características de margem e forma foram combinadas). Quando as características de margem e textura são combinados, alguns algoritmos alcançaram uma acurácia média no conjunto de testes de até 97.25%.

Para todos os algoritmos baseados no k-NN e para a DT, os melhores resultados para a base de dados 100 Folhas de Plantas foram alcançados quando todas as três características foram combinadas, onde alguns algoritmos conseguiram alcançar uma acurácia

Tabela 2. Resultados experimentais. M: margem da folha; S: forma da folha; T: textura da folha; Média: acurácia média no conjunto de testes; Std: desvio padrão; Tempo: tempo médio de execução em segundos.

Algoritmo	Métrica	Base de Dados								
		Iris	Seeds	M	S	T	MS	MT	ST	MST
DT	Média	0.9533	**0.9000**	0.4587	0.4369	0.5313	0.6081	0.5981	0.6625	0.7031
	Std.	0.0450	**0.0690**	0.0365	0.0443	0.0287	0.0315	0.0320	0.0473	0.0219
	Tempo	0.0008	0.0008	0.0834	0.2669	0.1330	0.3844	0.2178	0.4376	0.5815
NB	Média	0.9533	**0.9000**	0.7519	0.5275	0.6650	0.8187	0.7419	0.7919	0.7525
	Std.	0.0450	**0.0653**	0.0262	0.0289	0.0416	0.0228	0.0487	0.0458	0.0440
	Tempo	0.0008	0.0008	0.0182	0.0181	0.0197	0.0246	0.0247	0.0268	0.0313
k-NN$_3$	Média	0.9600	0.8762	0.7500	0.5894	0.7681	0.8844	0.9688	0.9044	**0.9750**
	Std.	0.0344	0.0904	0.0357	0.0441	0.0266	0.0291	0.0088	0.0153	**0.0118**
	Tempo	0.0422	0.0492	0.2238	0.2110	0.2883	0.2807	0.2802	0.2787	0.4283
k-NN$_4$	Média	0.9600	0.8857	0.7481	0.5906	0.7669	0.8856	0.9631	0.9056	**0.9756**
	Std.	0.0344	0.0784	0.0320	0.0434	0.0248	0.0217	0.0119	0.0119	**0.0112**
	Tempo	0.0493	0.0562	0.2208	0.2336	0.2169	0.2938	0.3131	0.2909	0.4356
k-NN$_5$	Média	**0.9667**	0.8857	0.7544	0.5806	0.7725	0.8925	0.9613	0.9012	0.9719
	Std.	**0.0351**	0.0846	0.0301	0.0350	0.0261	0.0284	0.0121	0.0179	0.0099
	Tempo	0.0624	0.0632	0.2200	0.2226	0.2320	0.3045	0.3028	0.3043	0.4525
W-k-NN$_3$	Média	0.9267	0.8810	0.7637	**0.6175**	**0.7925**	0.8906	**0.9700**	**0.9137**	**0.9769**
	Std.	0.0492	0.0786	0.0453	**0.0382**	**0.0288**	0.0272	**0.0117**	**0.0150**	**0.0089**
	Tempo	0.0520	0.0528	0.2109	0.2164	0.2086	0.3045	0.2660	0.2667	0.4332
W-k-NN$_4$	Média	0.9267	0.8762	0.7737	**0.6156**	**0.7981**	0.8994	**0.9719**	**0.9181**	**0.9800**
	Std.	0.0492	0.0717	0.0426	**0.0345**	**0.0302**	0.0201	**0.0107**	**0.0091**	**0.0087**
	Tempo	0.0518	0.0557	0.2327	0.2353	0.2241	0.2978	0.3043	0.3030	0.4477
W-k-NN$_5$	Média	0.9333	0.8810	0.7712	0.6069	**0.7950**	0.8950	0.9625	**0.9150**	**0.9762**
	Std.	0.0544	0.0817	0.0369	0.0376	**0.0271**	0.0257	0.0121	**0.0145**	**0.0087**
	Tempo	0.0613	0.0649	0.2431	0.2402	0.2400	0.3177	0.3053	0.2887	0.4625
k-NNE$_3$	Média	0.9600	0.8762	**0.7838**	**0.6144**	**0.7981**	**0.9063**	**0.9719**	**0.9181**	**0.9806**
	Std.	0.0344	0.0816	**0.0366**	**0.0345**	**0.0313**	**0.0253**	**0.0126**	**0.0123**	**0.0062**
	Tempo	0.0218	0.0313	0.9949	1.0087	0.9746	1.1539	1.1529	1.1766	1.2071
k-NNE$_4$	Média	0.9600	0.8762	**0.7813**	0.6019	0.7837	**0.9063**	**0.9725**	**0.9125**	**0.9794**
	Std.	0.0344	0.0816	**0.0358**	0.0286	0.0236	**0.0232**	**0.0111**	**0.0102**	**0.0072**
	Tempo	0.0222	0.0306	1.0516	1.0503	1.0311	1.1992	1.2508	1.1103	1.2511
k-NNE$_5$	Média	0.9600	0.8810	**0.7806**	0.5813	0.7738	**0.9044**	0.9675	0.9063	**0.9769**
	Std.	0.0344	0.0753	**0.0352**	0.0347	0.0195	**0.0208**	0.0121	0.0132	**0.0098**
	Tempo	0.0221	0.0330	1.0917	1.1122	1.1239	1.3244	1.3532	1.1320	1.3109

média no conjunto de teste de 98.06% (k-NNE$_3$).

Os experimentos realizados no problema de 100 Folhas de Plantas mostraram que é muito importante encontrar o melhor conjunto de características quando se trata da classificação automática de plantas, uma vez que algumas características apresentam melhor poder discriminatório do que outras. Mas, como apontado pelos tempos médios de execução, o custo computacional para os classificadores pode aumentar consideravelmente quando estamos lidando com problemas com maior dimensionalidade (o problema da *maldição da dimensionalidade* [Bellman 1957]).

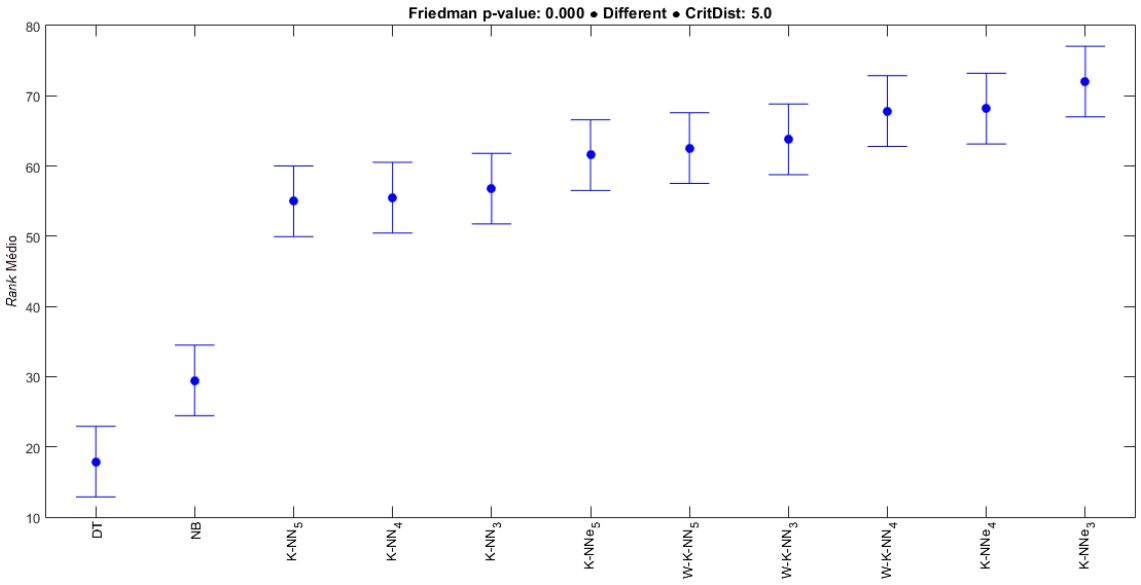

Figura 2. *Ranks* do teste de Friedman/Nemenyi (do pior resultado à esquerda, ao melhor resultado, à direita).

A Tabela 3 apresenta os *ranks* médios obtidas pelo teste de hipóteses de Friedman/Nemenyi. O teste de Friedman/Nemenyi mostra que k-NNE$_3$ obteve os melhores desempenhos médios de acordo com uma avaliação geral, em comparação com todos os outros algoritmos selecionados. O segundo e o terceiro melhor *rank* foram obtidos pelo k-NNE$_4$ e W-k-NN$_4$, respectivamente (sem diferenças estatísticas significativas em relação ao k-NNE$_3$ e entre si). Os piores desempenhos gerais foram alcançados pelos algoritmos DT e NB, que foram gravemente comprometidos quando aplicados à base de dados 100 Folhas de Plantas, que apresenta um grande número de classes. A avaliação geral também mostrou que, em geral, todos os métodos baseados no k-NN obtiveram melhores acurácias de teste quando adotaram $k = 3$ ou $k = 4$, já que ao lidar com um problema como 100 Folhas de Plantas que apresenta um alto número de classes e poucas instâncias por classe, o aumento no valor de k pode ocasionar a escolha de padrões de diferentes classes na composição da vizinhança de um padrão de teste, o que pode vir a prejudicar o processo decisório do algoritmo. A Fig. (2) apresenta os *ranks* médios do teste de Friedman/Nemenyi, do pior algoritmo (à esquerda) ao melhor (à direita).

5. Conclusões

Neste trabalho, avaliamos o desempenho de cinco classificadores bem estabelecidos da literatura de Aprendizagem de Máquina quando aplicados ao problema de reconhecimento

Tabela 3. Avaliação geral: *Ranks* médios do teste de Friedman/Nemenyi.

Algoritmo	*Rank* médio
DT	17.8389
NB	29.4667
k-NN$_3$	56.7778
k-NN$_4$	55.4944
k-NN$_5$	54.9778
W-k-NN$_3$	63.7944
W-k-NN$_4$	**67.8333**
W-k-NN$_5$	62.5611
k-NNE$_3$	**72.0278**
k-NNE$_4$	**68.1778**
k-NNE$_5$	61.5500

de plantas: Árvore de Decisão, Naïve Bayes, k-Vizinhos mais Próximos, k-Vizinhos mais Próximos Ponderado e k-Vizinhos mais Próximos de Igualdade. O trabalho tem por objetivos complementar a pesquisa realizada por [Rahmani et al. 2015] pelo acréscimo de duas abordagens melhoradas por ponderação do algoritmo dos k-Vizinhos mais Próximos e pela avaliação dos modelos através de duas outras bases de dados de plantas.

Para fins de comparação, três bases de dados de plantas do mundo real, obtidos do UCI Machine Learning Repository são empregados: Iris, Núcleos de Sementes de Trigo e 100 Folhas de Plantas. A base de dados de 100 Folhas de Plantas foi dividida em sete conjuntos, para que pudéssemos testar cada vetor de característica da folha individualmente, assim como todas as combinações possíveis dos três vetores de características (*margem da folha*, *forma da folha* e *textura da folha*).

O critério de avaliação é baseado em uma análise empírica complementada por um teste de hipótese do tipo teste de Friedman/Nemenyi em relação à acurácia média de teste obtida por cada classificador para cada um dos nove conjuntos de dados adotados.

Os resultados experimentais mostraram o potencial das abordagens melhoradas do k-NN para lidar com problemas de reconhecimento automático de espécies de plantas. Para os experimentos selecionados, os melhores desempenhos médios foram obtidos pelo k-NNE e pelo W-k-NN com valores baixos de k. Os resultados experimentais também mostraram a importância da seleção de um bom conjunto de características da planta para os sistemas de reconhecimento, e como algumas características podem comprometer a qualidade das soluções finais fornecidas pelos algoritmos de classificação.

Como trabalhos futuros, pretendemos estender nosso estudo pela inclusão de outras características de plantas extraídas automaticamente usando técnicas de processamento de imagens e adotando bases de dados maiores. Também pretendemos avaliar a influência do novo conjunto de características no comportamento de classificadores aplicados ao reconhecimento automático de plantas e o uso de técnicas para a seleção automática do conjunto de características mais relevantes (como os Algoritmos Evolutivos), de modo a possibilitar o desenvolvimento de sistemas mais precisos e com baixo custo computacional.

Referências

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283.

Agarwal, G., Belhumeur, P., Feiner, S., Jacobs, D., Kress, W. J., Ramamoorthi, R., Bourg, N. A., Dixit, N., Ling, H., Mahajan, D., et al. (2006). First steps toward an electronic field guide for plants. *Taxon*, 55(3):597–610.

Anderson, E. (1935). The irises of the gaspe peninsula. *Bulletin of American Iris Society*, 59:2–5.

Asuncion, A. and Newman, D. (2007). Uci machine learning repository.

Bellman, R. E. (1957). *Dynamic programming*. Princeton University Press.

Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., and Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.

Charytanowicz, M., Niewczas, J., Kulczycki, P., Kowalski, P. A., Łukasik, S., and Żak, S. (2010). Complete gradient clustering algorithm for features analysis of x-ray images. In *Information technologies in biomedicine*, pages 15–24. Springer.

Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30.

Dudani, S. A. (1976). The distance-weighted k-nearest-neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics*, 1(4):325–327.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of human genetics*, 7(2):179–188.

Friedman, M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the american statistical association*, 32(200):675–701.

Jin, T., Hou, X., Li, P., and Zhou, F. (2015). A novel method of automatic plant species identification using sparse representation of leaf tooth features. *PloS one*, 10(10):e0139482.

Kumar, N., Belhumeur, P. N., Biswas, A., Jacobs, D. W., Kress, W. J., Lopez, I. C., and Soares, J. V. (2012). Leafsnap: A computer vision system for automatic plant species identification. In *Computer Vision–ECCV 2012*, pages 502–516. Springer.

Mallah, C., Cope, J., and Orwell, J. (2013). Plant leaf classification using probabilistic integration of shape, texture and margin features. *Signal Processing, Pattern Recognition and Applications*, 5(1).

Mallah, C. D. and Orwell, J. (2013). Probabilistic classification from a k-nearest-neighbour classifier. *Computational Research*, 1(1):1–9.

Nemenyi, P. (1962). Distribution-free multiple comparisons. *Biometrics*, 18(2):263.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Rahmani, M. E., Amine, A., and Hamou, M. R. (2015). Plant leaves classification. *ALLDATA 2015*, 82.

Sahay, A. and Chen, M. (2016). Leaf analysis for plant recognition. In *Software Engineering and Service Science (ICSESS), 2016 7th IEEE International Conference on*, pages 914–917. IEEE.

Sierra, B., Lazkano, E., Irigoien, I., Jauregi, E., and Mendialdua, I. (2011). K nearest neighbor equality: giving equal chance to all existing classes. *Information Sciences*, 181(23):5158–5168.