Métodos de Classificação
Aprendizagem de Máquina
2024

Caio Shimohiro, Gustavo Hanke

1. Introdução

Este documento tem como objetivo descrever os processos de implementação e realizar um estudo e comparação entre os cinco algoritmos classificadores apresentados na matéria de Aprendizagem de Máquina - KNN, SVM, Árvore de Decisão, Naive Bayes e MLP - em uma base de dados escolhida e que será apresentada na sequência.

Vale ressaltar que todo o processo foi realizado na linguagem Python fazendo utilização sobretudo da biblioteca Scikit-Learn.

2. Implementação

2.1. Análise descritiva dos dados

A base de dados escolhida para a aplicação dos métodos, disponibilizada pela Universidade da Califórnia - Irvine, apresenta um conjunto de 3810 instâncias compostas por duas classes que representam dois tipos distintos de grãos de arroz, Cameo e Osmancik.

Para a identificação dos grãos, cada instância é composta por 7 atributos que identificam os grãos:

Atributo	Tipo	Descrição	Valor mínimo	Valor máximo	Valor médio
Area	Inteiro	Número de pixels nos limites do grão	7.551	18.913	12.667,73
Perimeter	Real	Circunferência baseado no número de pixels em volta dos limites do grão	359,10	548,45	454,24
Major_Axis_Length	Real	Linha mais longa que pode ser traçada no grão	145,26	239,01	188,78
Minor_Axis_Length	Real	Linha mais curta que pode ser traçada no grão	59,53	107,54	86,31
Eccentricity	Real	Grau de circunferência da elipse	0,7772	0,948	0,8869
Convex_Area	Inteiro	Número de pixels da menor área convexa do grão	7.723	19.099	12.952,5

Extent	Real	Proporção entre a região formada pelo grão e a caixa delimitadora	0,4974	0,8611	0,6619

O objetivo dessa base de dados é de ser utilizada para treinamento de modelos para reconhecimento e classificação das duas variedades de arroz para poder então ser utilizado para separação automática dos grãos, já que o processo manual é inviável.

2.2. Divisão do conjunto de dados

A divisão dos dados foi realizada como solicitado na definição do trabalho, sendo 50% da base original destinada para o treinamento da base, 25% para validação e os outros 25% para testes, sendo utilizado a função train_test_split() da biblioteca Scikit-Learn. Para garantir que a proporção das classes se manteria, foi utilizada a configuração stratify na função.

Para garantir a aleatoriedade em cada iteração, foi utilizado a função shuffle() também do Sklearn no início de cada novo ciclo.

2.3. Treinamento e calibração dos modelos

Na fase de treinamento dos modelos, foram explorados diversos valores dos hiperparâmetros requisitados para cada método. Serão descritos a seguir cada um deles.

2.3.1. KNN

Para o KNN, o K, ou número de vizinhos, foi testado em uma faixa de 1 até 50. Já o weight (peso) foi variado entre os modos uniform, sem ponderar as distâncias, e distance, ponderando as distâncias.

Abaixo encontram-se os hiperparâmetros selecionados após treinamento e validação em cada iteração:

Iteração	K	Weight
1	5	uniform
2	14	distance
3	45	uniform
4	11	distance
5	23	distance
6	13	uniform
7	7	distance
8	6	distance
9	29	distance
10	6	distance
11	17	distance
12	47	distance

13	42	distance
14	5	distance
15	24	distance
16	11	distance
17	23	uniform
18	11	uniform
19	12	uniform
20	43	distance

Em análise, a média dos valores de K foi de 13,5. Enquanto o peso mais optado foi o de distance, que leva em conta a distância dos vizinhos.

2.3.2. SVM

Para o SVM foram explorados os parâmetros de kernel, selecionados entre linear, polinomial, RBF e sigmoid. Além do valor C de nível de generalização, que variou de 1 até 4. Os resultados se encontram a seguir:

Iteração	**Kernel**	**C**
1	linear	2
2	linear	3
3	linear	2
4	linear	1
5	linear	1
6	linear	2
7	linear	1
8	linear	3
9	linear	2
10	linear	4
11	linear	1
12	linear	1
13	linear	3
14	linear	4
15	linear	4
16	linear	3
17	linear	1
18	linear	3
19	linear	1

20	linear	2

Nota-se que foi selecionado apenas o kernel linear enquanto o valor de C variou bastante tendo um valor médio de 2.

2.3.3. Árvore de decisão

Para a AD, foram testados os valores de criterion variando entre gini e entropy, a profundidade máxima variando de 3 até 19, o mínimo de amostras por folha de 1 até 6 e o mínimo de amostras para divisão de nó de 2 x mínimo de amostras por folha até 4 x mínimo de amostras por folha.

Os resultados a seguir:

Iteração	Criterion	max_depth	min_samples_leaf	min_samples_split
1	gini	4	2	7
2	entropy	5	4	8
3	gini	4	3	10
4	entropy	3	1	2
5	entropy	3	5	10
6	gini	4	1	2
7	gini	3	1	2
8	gini	4	1	3
9	entropy	5	2	4
10	gini	3	1	2
11	entropy	3	1	2
12	gini	4	5	14
13	entropy	4	3	6
14	entropy	3	2	4
15	gini	3	1	2
16	entropy	7	5	13
17	gini	4	5	10
18	gini	5	4	9
19	entropy	4	1	4
20	entropy	4	2	4

O parâmetro de criterion teve uma distribuição igual entre gini e entropy. Já a profundidade, apesar de ser testada com valores muito maiores, manteve-se baixa em uma média de 4 apenas. O número mínimo de amostras para divisão de nó também teve uma média baixa de apenas 2

amostras enquanto o mínimo de amostras por folha obteve uma média de 4 variando bastante nos valores optados.

2.3.5. MLP

O número de iterações do método variou entre 150, 300, 500 e 1000. Para o número de neurônios nas camadas ocultas no MLP, foram variados valores entre 6, 12, 25, 50 e 100, sendo sempre 3 camadas ocultas apenas. Em relação à função de ativação, foram entre identity (sem transformação), logistic (sigmoide), tanh (tangente hiperbólica) e relu (rectified linear unit). Por fim, a taxa de aprendizado variou entre constant (constante), invscaling (inversamente proporcional às iterações) e adaptative (mantém constante enquanto desempenho não cai). Os resultados abaixo:

Iteração	n de iterações	Neurons	Ativação	Taxa Aprendizagem
1	500	12	relu	invscaling
2	1000	6	relu	constant
3	500	12	relu	invscaling
4	1000	6	identity	invscaling
5	1000	12	relu	adaptative
6	1000	6	relu	invscaling
7	300	6	relu	invscaling
8	150	12	relu	constant
9	500	6	relu	constant
10	500	6	relu	constant
11	150	6	relu	invscaling
12	1000	6	identity	invscaling
13	150	6	relu	adaptative
14	500	6	identity	constant
15	1000	6	relu	adaptative
16	150	12	relu	constant
17	300	6	relu	adaptative
18	150	6	identity	constant
19	300	6	relu	adaptative
20	150	6	identity	adaptative

O número de iterações variou bastante e manteve uma média de 515 no final. O número de neurônios variou pouco e manteve-se baixo, explicando a média de 7,5. Para a função de

ativação, apenas relu e identity foram optados por, sendo relu a mais constante. Já para a taxa de aprendizagem, ambos constant e invscaling tiveram a mesma frequência, 7 cada.

2.3.5. Naive Bayes

Esse método não teve hiperparâmetros a serem explorados, portanto essa fase é ignorada.

2.4. Avaliação dos modelos

Levando em consideração os hiperparâmetros encontrados durante o treinamento e validação, foram realizados os testes de acordo com a divisão previamente realizada da base de dados:

Iteração	Naive Bayes	KNN	SVM	AD	MLP
1	0,9035	0,8709	0,9318	0,9286	0,8541
2	0,9181	0,8867	0,9265	0,9202	0,8594
3	0,9108	0,8814	0,915	0,915	0,8751
4	0,915	0,8951	0,9297	0,9276	0,9098
5	0,916	0,8867	0,9297	0,9108	0,9077
6	0,9057	0,8783	0,9265	0,9181	0,8856
7	0,9066	0,8804	0,9297	0,9255	0,9087
8	0,9066	0,8657	0,9129	0,9129	0,8006
9	0,8993	0,8814	0,9181	0,9098	0,8415
10	0,9297	0,8909	0,936	0,9381	0,8384
11	0,9066	0,8646	0,9202	0,9192	0,8132
12	0,9119	0,8888	0,936	0,9339	0,8604
13	0,9171	0,8793	0,9265	0,9265	0,8594
14	0,9087	0,8751	0,9297	0,9244	0,9003
15	0,8982	0,8667	0,9223	0,9181	0,8258
16	0,916	0,8898	0,936	0,9213	0,808
17	0,9098	0,8762	0,9339	0,9318	0,9024
18	0,9119	0,8846	0,9328	0,9244	0,8678
19	0,914	0,8835	0,937	0,9223	0,8793
20	0,916	0,8993	0,9297	0,9276	0,873
Média (DP)	**0,91135 (0,007)**	**0,8814 (0,009)**	**0,9297 (0,007)**	**0,9233 (0,007)**	**0,8641 (0,033)**

Após 20 iterações, pode-se observar que todos os classificadores conseguiram um resultado bom e não tão discrepante, com exceção do MLP, que ficou distante dos outros.

2.5. Análise comparativa

Para a análise, primeiro é realizado o teste de Krukal-Wallis com 5% de significância como instruído pela descrição da atividade. Foram analisados então os valores apresentados acima e

obteve-se como resultado um p-value de 0,29305. Levando em conta os 0.05 de significância, conclui-se que não há diferença significativa entre os classificadores, logo H_0 é verdadeira.

Por consequência, não foram realizados os testes de Mann-Whitney.

2.6. Sistemas de múltiplos classificadores

Após os testes regulares de todos os classificadores, seus resultados foram utilizados para o cálculo dos métodos de múltiplos classificadores, sendo eles os métodos: Soma, Voto Majoritário e Borda Count. Os resultados seguem:

Iteração	Soma	Voto Majoritário	Borda Count
1	0,9171	0,9192	0,9202
2	0,9202	0,9129	0,9139
3	0,9108	0,9108	0,9108
4	0,9276	0,9297	0,9297
5	0,9181	0,9234	0,9234
6	0,9129	0,915	0,915
7	0,915	0,9213	0,9213
8	0,9045	0,9003	0,8992
9	0,9077	0,916	0,915
10	0,9286	0,9328	0,9318
11	0,9108	0,9098	0,9108
12	0,9181	0,9213	0,9213
13	0,9213	0,915	0,915
14	0,914	0,9202	0,9202
15	0,9066	0,914	0,914
16	0,9255	0,9223	0,9213
17	0,9234	0,9213	0,9202
18	0,9202	0,9171	0,9171
19	0,9181	0,914	0,914
20	0,9171	0,9181	0,9181
Média	**0,9176 (0,007)**	**0,9176 (0,007)**	**0,9176 (0,007)**

Analisando os resultados dos três métodos, observa-se que, com 4 casas de precisão, suas médias ficaram iguais, apesar de apresentarem valores distintos em cada iteração. Vale ressaltar que, apesar de estar invisível com 4 casas decimais, foi observada uma divergência pequena tanto nas médias quanto nos seus respectivos desvios padrões, porém foram pequenas demais para se encaixarem nesse nível de precisão.

Realizando o teste de Kruskal-Wallis sobre os três métodos, o resultado previsivelmente indica que não há discrepância significativa, apresentando um p-value de 0,99709. Logo dispensa-se Mann-Whitney assim como ocorreu na abordagem monolítica.

2.7. Comparação entre abordagem monolítica e SMC

Levando em consideração que não foi possível inferir um método factualmente melhor para nenhuma das abordagens, foi optado por selecionar os métodos para comparação baseados puramente em suas acurácias médias. Sendo assim, foi selecionado o classificador SVM para representar a abordagem monolítica e o classificador de Voto Majoritário.

Como já era esperado, utilizando o teste de Mann-Whitney bicaudal com 5% de significância, obteve-se um p-value de 0,4654, indicando que não há discrepância significativa.