Aprendizado de Máquina para Identificação de Manuscritos indo-arábicos

Marcos Paulo Diniz
Universidade de Brasília
Departamento de Ciência da Computação
Brasília, Brasil
marcosdiniz@aluno.unb.br

***Resumo*—O objetivo do trabalho a ser apresentado é medir a capacidade do algoritmo de aprender o número que está sendo lido, em forma de uma imagem, e classificá-lo entre um dos algarismos arábicos. Os algoritmos que serão implementados para chegar no objetivo serão o *K nearest neighbors* (K-nn) e o *Linear Discriminant Analysis* (LDA).**

***Index Terms*—Algoritmo, aprender, K-nn e LDA.**

I. Introdução

O Projeto consiste em fazer um algoritmo capaz de identificar os números a partir de imagens, isto é, ao ler uma imagem saber qual a informação numérica está presente nela (qual número está sendo representado).

Para a implementação desse algoritmo foi usado a linguagem de programação Python (versão 3.6.3).

Uma das técnicas de classificação foi o K-nn, a ideia principal desse algoritmo é determinar o rótulo de classificação de uma amostra baseado nos vizinhos. Para isso, é preciso separar toda a amostra inicial em duas, uma para treinamento do algoritmo e outra para o teste do algoritmo. No K-nn a variável 'k' representa a quantidade de vizinhos que serão usados para classificar um elemento da amostra de teste.

Assim o algoritmo diz a que classe o elemento de teste pertence, tendo como base os k-vizinhos, olhando sempre qual classe tem mais elemento presente entres os k elementos próximos. Como pode ser visto no exemplo abaixo:

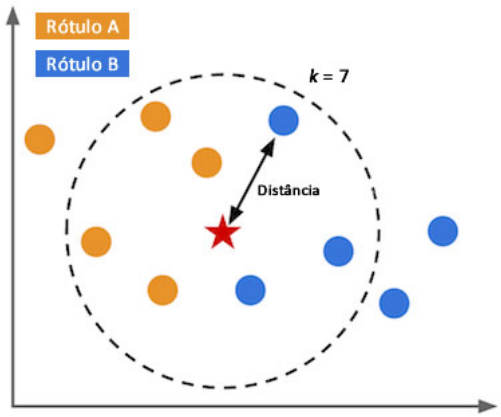

Figura 1. Exemplo K-nn

No experimento, havia um conjunto para treinamento e um conjunto de testes, com 60.000 e 10.000 elementos, respectivamente. E foram usados diferentes valores de K, porém nem todos tiveram um resultado satisfatóriom, com isso alguns desses foram ignorados e não serão apresentados nesse trabalho. Dentre os valores com um resultado aceitável, tem os seguintes valores de K: 1, 10, 100, 245, 490 e 1000.

Além do K-nn foi aplicado também o algoritmo da Análise Discriminante Linear (LDA). A LDA tenta encontrar uma transformação linear através da maximização da distância entre-classes e minimização da distância intra-classe. O método tenta encontrar a melhor direção de maneira que quando os dados são projetados em um plano, as classes possam ser separadas.

II. Analise do Experimento

Nesse experimento, foi usado num primeiro momento somente a LDA. Com a LDA foi possível ter um resultado bastante satisfatório, a taxa média de acerto superou os 87%, e usando a LDA a menor taxa de acerto foi de 79,1% (que foi na identificação do número 2), enquanto a maior foi de 96,6% (que foi na identificação do número 1).

Em seguida, foi usado algoritmo do K-nn com o mesmo conjunto de imagens de treino e teste. Uma parte de fundamental importância foi a escolha dos valores de K, foram escolhidos os seguintes valores: 1, 10, 100 245, 490 e 1000. Sem contar o valor de 245 e 490, os outros valores foram escolhidos de forma puramente experimental, enquanto esses dois foram escolhidos pelo fato de 245 ser a raiz aproximada de 60.000 (número de imagens de treinamento) e 490 por ser o dobro do número anterior.

Para cada K os resultados foram diferentes, para o k = 1 tivemos a melhor taxa de acerto médio superando os 96,8%, onde o melhor resultado do algoritmo foi 99,5% de acerto para o número 1, enquanto o pior resultado foi para o número 8, ficando nos 94,5%.

Ao aumentar o k para k = 10 foi percebido um pequeno decrescismento na taxa de acerto, caindo para 96,6%, mas ainda sim uma taxa bem próxima da melhora taxa (onde k = 1).

Novamente, ao observar a caida do valor de acerto do k = 1 para o k = 10, alteramos o valor de k para 100. Ao realizar tal alteração, novamente foi percebida a redução da taxa de

acerto, dessa vez bem mais significativa do que no a diferença entre o k = 1 e o k = 10. Com o k = 100 a taxa média de acerto caiu para 94,35%, tendo o melhor resultado com o número 1 (com 99,6% de acerto) e o pior resultado com o número 2, alcançando a taxa de acerto de 88,7%, sendo menor registrada entre esses 3 testes do k-nn.

Logo em seguida, testamos o algoritmo com o valor de k = 245 (a raiza aproximada de 60.000, isto é, o numero de elementos usados para treinamento), novamente a taxa de acerto foi diminuindo, conforme o aumento do valor de k, dessa vez a taxa de acerto média foi de 92,31%, onde o maior resultado foi, outra vez, com o número 1 (com 99,6%) e o pior resultado ficou com a taxa de 84,1%, novamente com o número 2. Ao testarmos com o k = 490 a tendencia de diminuição da taxa de acerto foi mantida, conforme pode ser observado nas tabelas abaixas.

Por fim, ao testar com o k = 1000, obtivemos o pior resultado do K-nn, onde a taxa de acerto médio foi inferior à taxa da LDA, ficando em 87,14%, e a menor taxa de acerto foi novamente com o número 2, ficando pouco acima dos 70%, enquanto a melhor taxa foi novamente a do número 1, superando alguns resultados para outros valores de k, ficando em 99,7%.

A seguir tem-se um gráfico que mostra como foi a taxa de acerto para cada número nos diferentes valores de K e na LDA:

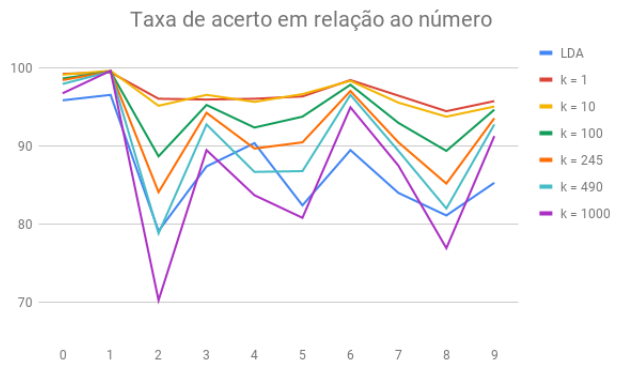

Figura 2. Exemplo K-nn

Ao analisar o gráfico acima, observamos que tanto no K-nn quanto na LDA, foi-se identificado uma dificuldade na identificação no número 2 ocorrendo o mesmo com o número 8 (porém com resultados melhores que o número 2), enquanto foi notável a facilidade em identificar os números 1 e 5.

Em uma segunda análise, é possível observar a diminuição da taxa de acerto com o aumento do valor de k, e quando o valor de k se iguala a 1000, é perceptível que as taxas de acerto caem de forma elevada, em alguns momentos tendo resultados piores do que os observados no algoritmo da LDA.

III. Conclusões

Após os inumeros testes foi percebido que para achar um k "ideal"para o K-nn (que busca maximizar as taxas de acertos), trata-se de uma busca puramente experimental e se torna uma tarefa muito cansativa.

Ao ir realizando vários testes resolvemos adicionar marcador de tempo no nosso programa. A média do tempo de execução para encontrar todos os resultados apresentados nesse trabalho foi de 47 minutos, contando o trainamento e as cassificações, esse tempo compreeende os tempos somados da LDA e do K-nn. Após isso, realizamos a analisa do tempo de forma individual e também comparando com a matriz de confusão de cada caso de teste.

Foi observado que o algarismo da LDA realizava todo o treinamento e teste em cerca de 18 segundos, enquanto o algarismo do K-nn levava em torno de 7 a 8 minutos para cada execução, variando de acordo com k (ambos testatos usando a mesma máquina nas mesmas condições).

Com isso ao fazer a análise, foi percebido que o valor de k, não interferia tanto no tempo de execução, mas foi percebido que o quanto o k crescia, maior ficava a taxa de erro, e os melhores resultados foram obtidos com os menores valores de k (o melhor foi com o valor de k = 1).

Tal fato se deve, para um valor de k muito grande, considerar muitos elementos para a classificação, e com isso acabar agrupando muitos elementos não relevantes, com isso diminuindo a taxa de acerto. Enquanto que quando foi olhado somente o vizinho mais próximo, isto é, k =1, o resultado foi melhor do que o esperado. Enquanto um k muito grande pode considerar elementos de mais e acabar prejudicando o desempenho, o mesmo poderia acontecer com um k muito pequeno, já que só consideraria o elemento mais próximo, no entando os menores valores de k foram os que nos trouxeram os melhores resultados.

Agradecimentos

Ao professor Alexandre Zaghetto que me apresentou, em um primeiro momento, matérias relativas à aprendizado de máquina e, com isso, criei interesse pela área.

Referências

[1] Bishop, C. Pattern Recognition and Machine Learning. Springer, 2006
[2] Mitchell, T. Machine Learning. McGraw Hill, 1997.
[3] Lichman, M. (2013). UCI Machine Learning Repository. Irvine, CA: the University of California, School of Information and Computer Science.
[4] Bird, S., Klein, E., and Loper, E. (2009). Natural language processing with Python: Analyzing text with the natural language toolkit. Sebastopol, CA: O'Reilly Media.

Tabela I
MATRIZ DE CONFUSÃO - LDA

	0	1	2	3	4	5	6	7	8	9
0	940	0	15	5	0	8	12	2	7	9
1	0	1096	32	5	12	8	8	30	27	7
2	1	4	816	25	6	4	11	15	8	1
3	4	3	34	883	0	44	0	9	27	13
4	2	2	21	4	888	12	25	22	20	63
5	13	2	5	25	4	735	29	2	53	6
6	9	3	37	3	7	15	857	0	10	0
7	1	0	9	16	2	10	0	864	6	37
8	9	25	57	29	10	38	16	4	790	12
9	1	0	6	15	53	18	0	80	26	861
Méd.	95,9	96,6	79,1	87,4	90,4	82,4	89,5	84	81,1	85,3

Tabela II
MATRIZ DE CONFUSÃO - K-NN: K = 1

	0	1	2	3	4	5	6	7	8	9
0	973	0	7	0	0	1	4	0	6	2
1	1	1129	6	1	7	1	2	14	1	5
2	1	3	992	2	0	0	0	6	3	1
3	0	0	5	970	0	12	0	2	14	6
4	0	1	1	1	944	2	3	4	5	10
5	1	1	0	19	0	860	5	0	13	5
6	3	1	2	0	3	5	944	0	3	1
7	1	0	16	7	5	1	0	992	4	11
8	0	0	3	7	1	6	0	0	920	1
9	0	0	0	3	22	4	0	10	5	967
Méd.	99,3	99,5	96,1	96	96,1	96,4	98,5	96,5	94,5	95,8

Tabela III
MATRIZ DE CONFUSÃO - K-NN: K = 10

	0	1	2	3	4	5	6	7	8	9
0	972	0	13	0	2	4	6	0	6	7
1	1	1132	12	3	11	0	4	27	4	6
2	1	2	982	3	0	0	0	4	5	3
3	0	0	2	976	0	12	0	0	11	7
4	0	0	1	1	940	1	3	2	7	10
5	2	0	0	10	0	863	2	0	9	3
6	3	1	2	1	4	6	943	0	4	1
7	1	0	17	7	1	1	0	983	7	10
8	0	0	3	6	1	1	0	0	914	2
9	0	0	0	3	23	4	0	12	7	960
Méd.	99,2	99,7	95,2	96,6	95,7	96,7	98,4	95,6	93,8	95,1

Tabela IV
MATRIZ DE CONFUSÃO - K-NN: K = 100

	0	1	2	3	4	5	6	7	8	9
0	967	0	22	0	0	5	9	0	12	9
1	1	1130	43	8	19	9	8	43	11	9
2	1	2	915	3	0	0	0	2	3	3
3	0	1	7	963	0	13	0	0	20	7
4	0	0	2	1	907	1	2	2	12	7
5	3	0	1	11	0	837	1	0	17	3
6	7	2	5	1	11	14	938	0	5	1
7	1	0	28	12	3	2	0	956	9	14
8	0	0	9	7	2	0	0	0	871	0
9	0	0	0	4	40	11	0	25	14	956
Méd.	98,7	99,6	88,7	95,3	92,4	93,8	97,9	93	89,4	94,7

Tabela V
MATRIZ DE CONFUSÃO - K-NN: K = 245

	0	1	2	3	4	5	6	7	8	9
0	965	0	23	0	1	5	11	0	16	11
1	1	1131	78	15	29	20	9	63	19	13
2	0	2	868	4	0	0	0	2	2	2
3	0	1	8	952	0	23	0	0	29	8
4	0	0	5	1	881	2	5	3	11	9
5	4	0	1	9	1	807	3	0	24	2
6	9	1	7	1	11	15	930	0	6	2
7	1	0	30	12	2	3	0	930	10	18
8	0	0	12	8	1	1	0	0	830	0
9	0	0	0	8	56	16	0	30	27	944
Méd.	98,5	99,6	84,1	94,3	89,7	90,5	97,1	90,5	85,2	93,6

Tabela VI
MATRIZ DE CONFUSÃO - K-NN: K = 490

	0	1	2	3	4	5	6	7	8	9
0	960	0	26	0	1	7	12	0	18	9
1	1	1131	108	23	38	32	11	70	33	18
2	0	2	813	4	0	0	0	2	2	2
3	0	1	12	937	0	31	0	0	38	9
4	0	0	12	1	851	3	7	5	9	10
5	6	0	2	10	1	774	2	0	23	1
6	12	1	8	3	13	17	925	0	8	3
7	1	0	30	15	3	5	1	919	11	21
8	0	0	20	9	1	2	0	0	799	0
9	0	0	1	8	74	21	0	32	33	936
Méd.	98	99,6	78,8	92,8	86,7	86,8	96,6	89,4	82	92,8

Tabela VII
MATRIZ DE CONFUSÃO - K-NN: K = 1000

	0	1	2	3	4	5	6	7	8	9
0	949	0	27	1	0	7	15	1	22	10
1	1	1132	177	46	46	51	18	85	63	24
2	0	1	724	5	0	0	0	2	3	2
3	0	1	18	904	0	48	0	0	45	8
4	0	0	13	1	822	5	11	4	10	13
5	8	0	3	10	0	721	4	0	18	1
6	19	1	8	3	14	23	910	0	10	3
7	1	0	32	13	3	7	0	899	12	25
8	2	0	27	14	1	2	0	0	749	2
9	0	0	3	13	96	28	0	37	42	921
Méd.	96,8	99,7	70,2	89,5	83,7	80,8	95	87,5	76,9	91,3