

Comparação de diferentes tamanhos de representação de dígitos para um classificador KNN

Gabriel de Oliveira Pontarolo

1 Introdução

O relatório a seguir faz uma análise do uso de diferentes tamanhos de vetores de características para fazer a representação de dígitos escritos a mão. Esses vetores foram utilizados em um classificador do K-ésimo vizinho mais próximo (KNN).

2 Metodologia

Para a extração de características, foi feito um *thresholding* da imagem, fazendo com que os pixels assumissem o valor 0 ou 1. Assim, o vetor resultante consiste da matriz de pixels “achatada” da imagem após a operação de limiarização.

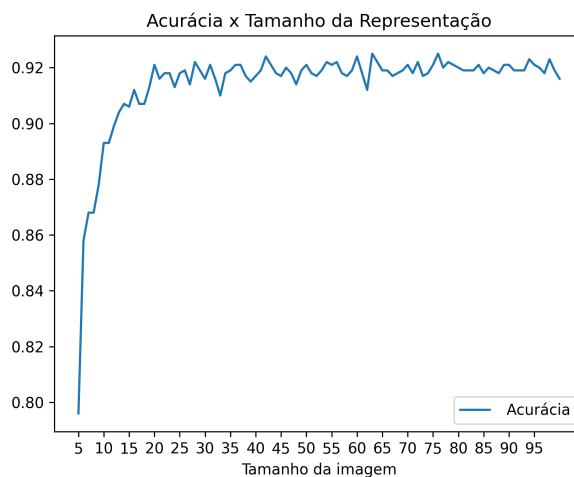


3 Testes

Os testes foram feitos comparando a acurácia do KNN para diferentes tamanhos de imagem variando de 5x5 a 100x100 pixels. Foi utilizada uma divisão 50:50 da base para treinamento e teste com um *random state* igual a 5. A métrica do classificador foi a distância euclidiana com $k=3$ inicialmente.

4 Resultados

O gráfico a seguir mostra a relação entre a acurácia e o tamanho do vetor de características.



Embora o comportamento do gráfico fique consideravelmente instável após 20x20, a melhor acurácia obtida foi 0.924 com uma imagem de 60x60, com a seguinte matriz de confusão:

```
[[ 96  0  0  0  0  1  0  0  0  0]
[  0 95  0  0  0  0  0  0  0  0]
[  0  4 101  1  1  0  1  2  1  0]
[  0  1  0 99  0  1  0  0  2  0]
[  0 10  2  0 82  0  0  0  0  1]
[  1  0  0  4  0 91  1  0  0  0]
[  2  5  0  0  0  0 99  0  0  0]
[  1  8  0  0  0  0  0 85  0  3]
[  1  5  0  3  0  1  0  1 75  1]
[  0  1  0  0  3  0  0  6  1 101]]
```

Ao mesmo tempo, a pior acurácia foi 0.796 para o menor tamanho de imagem testado, 5x5:

```
[[ 90  4  0  2  0  1  0  0  0  0]
[  0 92  0  0  0  0  1  1  1  0]
[  5  7 78  8  0  3  3  2  2  3]
[  1  0  2 94  0  4  0  1  0  1]
[  1 10  0  1 77  1  0  0  0  5]
[  3  3  0 13  0 77  1  0  0  0]
[  1  3  0  1  2  0 99  0  0  0]
[  0  6  0  1  3  1  0 74  0 12]
[  0  5  2  4  4 10  0 11 49  2]
[  0  4  1  0  5  0  0 36  0 66]]
```

Após obter o tamanho de 60x60, foram feitos testes variando a métrica do KNN e o valor de k. As variações na acurácia em função da métrica utilizada foram mínimas, e a melhor se mostrou ser de fato a distância euclidiana para esta aplicação. Quanto ao valor de k, foi obtida uma acurácia ligeiramente melhor de 0.926 com k = 1:

```
[[ 93  2  0  0  0  1  1  0  0  0]
[  0 93  0  0  0  1  0  0  0  1]
[  0  1 103  1  0  0  0  4  1  1]
[  1  0  0 98  0  1  0  0  3  0]
[  0  8  0  0 82  1  1  0  0  3]
[  1  0  0  6  0 89  1  0  0  0]
[  1  5  0  0  0  0 100  0  0  0]
[  0  3  1  0  1  0  0 89  0  3]
[  0  3  0  1  1  2  0  1 79  0]
[  0  1  0  0  1  0  0 10  0 100]]
```

3 Estudos extras

Além da proposta inicial da variação no tamanho da representação para o valor do pixel, foram feitos testes utilizando a técnica HOG (histograma de grafos orientados) para gerar o vetor de características. Para essa técnica, o melhor tamanho de imagem encontrado foi 24x24, $k = 3$ e *random state* = 10. A distribuição de amostras também foi modificada para 80:20. Com essa configuração, foi obtida uma acurácia de 0.9875, consideravelmente maior do que utilizando apenas os valores puros do pixel.

```
[[43  0  0  0  0  0  0  0  0  0]
[  0 47  0  0  0  0  0  0  0  0]
[  0  0 34  0  0  0  0  0  0  0]
[  0  0  0 41  0  0  0  0  0  0]
[  0  0  0  0 48  0  0  0  0  2]
[  0  0  0  1  0 40  0  0  1  0]
[  0  0  0  0  0  0 24  0  0  0]
[  1  0  0  0  0  0  0 35  0  0]
[  0  0  0  0  0  0  0  0 40  0]
[  0  0  0  0  0  0  0  0  0 43]]
```