Universidade Estadual de Campinas

Faculdade de Engenharia Elétrica e de Computação

IA048 – Aprendizado de Máquina 13 de abril de 2024

Docentes: Levy Boccato & Romis Attux

Discente:

– Gabriel Toffanetto França da Rocha – 289320

Atividade 2 – Classificação

Sumário

1 Apresentação dos dados **2**
1.1 Dados tratados 2

2 Classificação via Regressão Logística **3**
2.1 Dados tratados 3
2.1.1 Estrutura de treinamento 3
2.1.2 Passo de evolução 3
2.2 Dados brutos 3

3 Classificação via *k neareast neighbours* **3**
3.1 Dados tratados 3
3.2 Dados brutos 5

Anexos **6**

1 Apresentação dos dados

O problema de classificação com o reconhecimento de atividades humanas utiliza como base de dados amostras tomadas do acelerômetro e do giroscópio do *smartphone* preso à cintura do candidato. Dessa forma, com base na leitura desses sensores, pode-se identificar se a pessoa está caminhando, subindo escadas, descendo escadas, sentada, de pé ou deitada, que representam as seis classes do problema.

Além dos dados brutos, é fornecido também os dados processados, com extração sobre os dados no tempo, na frequência, e também características estatísticas dos mesmos.

1.1 Dados tratados

Os dados tratados são formados por amostras de 561 atributos derivados da análise no tempo e na frequência dos dados provenientes do acelerômetro e do giroscópio do *smartphone*. São um total de 7352 amostras para treinamento e validação, e 2947 amostras para teste.

O balanceamento das classes nos conjuntos de dados foi realizado por meio do cálculo da taxa de ocorrência dos mesmos, dada de acordo com (1). A Figura 1 mostra a distribuição das classes, e pode-se ver que não existe um balanceamento homogêneo, onde a classe 3 é a que menos ocorre, enquanto a classe 6 é a que mais ocorre.

Devido a esse desbalanceamento, a métrica que será utilizada para a avaliação do desempenho de cada classificador será a acurácia balanceada, dada por (2).

$$Rate_i = \frac{N_i}{N} \tag{1}$$

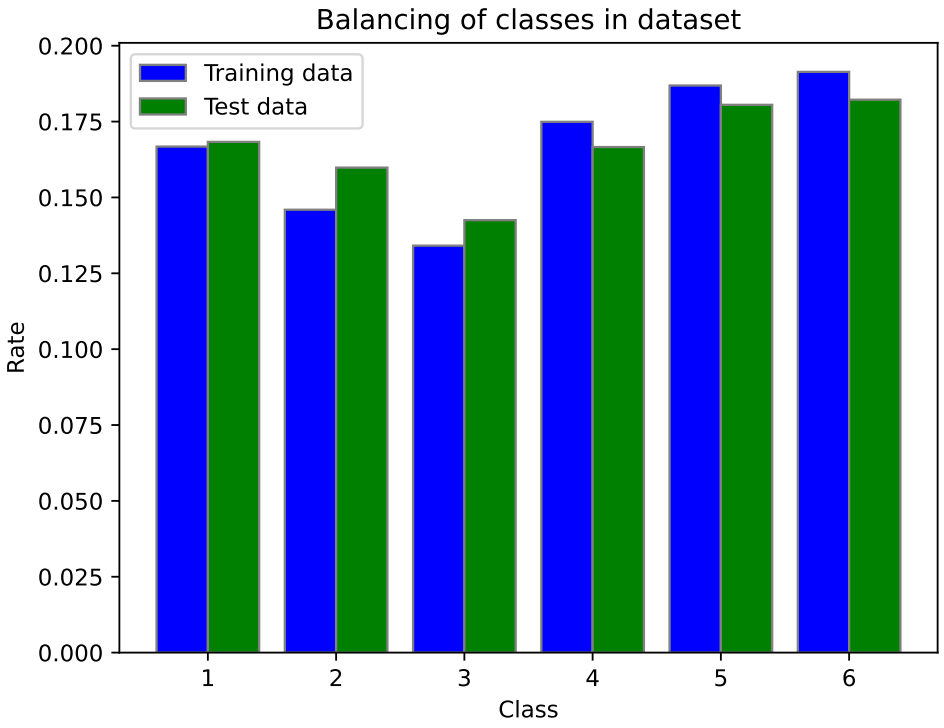

Figura 1: Gráfico da ocorrência das classes nos conjuntos de dados de treinamento e teste.

$$BA = \frac{\sum_{i=1}^{Q} Recall_i}{Q} = \frac{\sum_{i=1}^{Q} \frac{\mathrm{TP}_i}{N_i}}{Q} = \frac{\sum_{i=1}^{Q} \frac{\mathrm{TP}_i}{N \cdot Rate_i}}{Q} \tag{2}$$

2 Classificação via Regressão Logística

2.1 Dados tratados

2.1.1 Estrutura de treinamento

2.1.2 Passo de evolução

2.2 Dados brutos

3 Classificação via *k neareast neighbours*

A classificação pelo método *k neareast neighbours* é baseada em inferir a classe do dado a ser classificado com base nos k dados mais próximos à ele. Como hiper-parâmetros para esse problema, têm-se principalmente o valor de k, a ordem p da distância de Minkowski entre os dados e o critério de classificação.

O critério de classificação pode se basear puramente na classe majoritária entre os k vizinhos, ou levar em consideração a distância como um peso, que normalmente é inversamente proporcional a distância, evidenciando o rótulo dos pontos mais próximos do dado teste.

3.1 Dados tratados

Para implementação do algoritmo de kNN, foi escolhido a utilização da distância euclidiana no espaço dos atributos, e a decisão do rótulo vencedor por meio do voto majoritário dos k vizinhos mais próximos.

Para obtenção do valor de k, foi executada uma busca em *grid* do hiper-parâmetro, variando seu valor entre 1 e 29. Utilizando da técnica de validação cruzada *k-fold*, com quatro pastas, foi realizada a inferência das classes dos dados da pasta de validação com base nos vizinhos mais próximos encontrados nas pastas de treinamento, para cada valor de k testado. A Figura 2 exibe à evolução da acurácia balanceada para os valores de k, obtendo um conjunto de valores ótimos em (3).

$$k = [17\ 28\ 12\ 18] \tag{3}$$

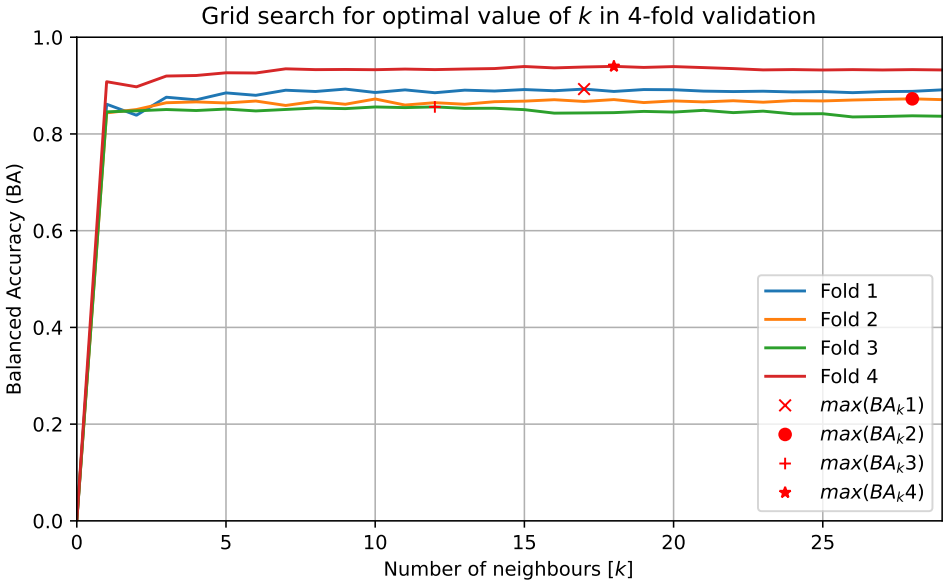

Figura 2: Busca em *grid* do valor de k ótimo utilizando *4-fold validation*.

A heurística escolhida para avaliar o melhor valor de k com base no conjunto obtido por meio da busca em *grid* com validação cruzada se dá em obter a acurácia balanceada média das pastas e obter o número de vizinhos que maximiza essa combinação das pastas. A Figura 3 mostra a progressão da acurácia balanceada média de acordo com k, e assim se obtém o valor de k ótimo em $k = 15$.

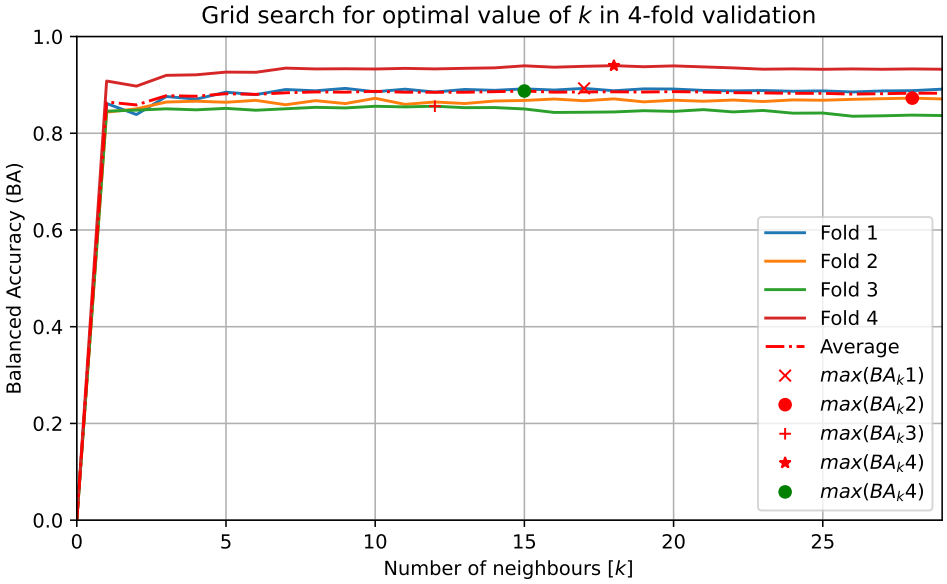

Figura 3: Busca em *grid* do valor de k ótimo utilizando *4-fold validation*.

Uma vez definido o classificador ótimo, obtém-se os indicadores de performance do classificador com base nos dados de teste. A acurácia balanceada encontrada foi de 0,8991 e a matriz de confusão do classificador por ser vista na Tabela 1.

$$BA = 0,8991 \tag{4}$$

	1	**2**	**3**	**4**	**5**	**6**
1	488	0	8	0	0	0
2	39	427	5	0	0	0
3	51	44	325	0	0	0
4	0	4	0	389	98	0
5	0	0	0	31	501	0
6	0	0	0	1	1	535

Tabela 1: Matriz de confusão do classificador k-NN com k = 8.

Extraindo da matriz de confusão as métricas de precisão e *recall*, obtém-se a Tabela 2. Pode-se observar que a classe 3 foi a que apresentou menor precisão, sendo muito confundida com a classe 1 e 2. Já a classe 1 possuí o pior *recall*, uma vez que as classes 2 e 3 se confundem com a 1. A classe 6 foi a que apresentou o melhor desempenho, apresentando *recall* unitário, logo, nenhuma classe se confunde com ela, e a maior precisão, muito próxima de 1.

Classe	**Precisão**	***Recall***
1	0.9839	0.8443
2	0.9066	0.8989
3	0.7738	0.9615
4	0.7923	0.9240
5	0.9417	0.8350
6	0.9963	1.0000

Tabela 2: Precisão e *Recall* do classificador por classe.

3.2 Dados brutos

Anexos

Códigos fonte

Todos os códigos fonte e arquivos de dados utilizados para a elaboração deste documento podem ser encontrados no repositório do GitHub no link: github.com/toffanetto/ia048.