Aprendizagem 2023
Homework II – Group 60
Maria João Rosa, Mariana Miranda
ist1102506, ist1102904

Part I: Pen and paper

1. Consider x_1–x_7 to be training observations, x_8–x_9 to be testing observations, $y_1 - y_5$ to be input variables and y_6 to be the target variable.

   a. Learn a Bayesian classifier assuming: i) $\{y_1, y_2\}$, $\{y_3, y_4\}$ and $\{y_5\}$ sets of independent variables (e.g., $y_1 \perp\!\!\!\perp y_3$ yet $y_1 \not\perp\!\!\!\perp y_2$), and ii) $y_1 \times y_2 \in \mathbb{R}^2$ is normally distributed. Show all parameters (distributions and priors for subsequent testing).

priors: $P(A) = \frac{3}{7}$ $\quad P(B) = \frac{4}{7}$

distributions:

$\{y_1, y_2\}$ $\quad P(y_1, y_2 | A) \sim N(\mu_A, \Sigma_A)$

$$\mu_A = \begin{bmatrix} \frac{0{,}24 + 0{,}16 + 0{,}32}{3} \\ \frac{0{,}36 + 0{,}48 + 0{,}72}{3} \end{bmatrix} = \begin{bmatrix} 0{,}24 \\ 0{,}52 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} cov(y_1, y_1) & cov(y_1, y_2) \\ cov(y_1, y_2) & cov(y_2, y_2) \end{bmatrix}$$

$$cov(y_1, y_1) = \frac{\sum_{i=1}^{n} (x_{i1} - \overline{y}_1)^2}{n-1} = \frac{(0{,}24-0{,}24)^2 + (0{,}16-0{,}24)^2 + (0{,}32-0{,}24)^2}{2} =$$

porque é uma amostra

$$= 6{,}4 \times 10^{-3} = 0{,}0064$$

$$cov(y_2, y_2) = \frac{\sum_{i=1}^{n} (x_{i2} - \overline{y}_2)^2}{n-1} = \frac{(0{,}36-0{,}52)^2 + (0{,}48-0{,}52)^2 + (0{,}72-0{,}52)^2}{2} =$$

$$= 0{,}0336$$

$$cov(y_1, y_2) = \frac{\sum_{i=1}^{n} (x_{i1} - \overline{y}_1)(x_{i2} - \overline{y}_2)}{n-1} =$$

$$= \frac{(0{,}24-0{,}24)(0{,}36-0{,}52) + (0{,}16-0{,}24)(0{,}48-0{,}52) + (0{,}32-0{,}24)(0{,}72-0{,}52)}{2} =$$

$$= 9{,}6 \times 10^{-3} = 0{,}0096 \qquad \Sigma_A = \begin{bmatrix} 0{,}0064 & 0{,}0096 \\ 0{,}0096 & 0{,}0336 \end{bmatrix}$$

$|\Sigma_A| = 0,0064 \times 0,0336 - 0,0096^2 = 1,2288 \times 10^{-4} = 0,00012288$

$$\Sigma_A^{-1} = \begin{bmatrix} 273,4375 & -78,125 \\ -78,125 & 52,0833 \end{bmatrix}$$

$$p(x \mid \mu, \Sigma) = N(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{m/2}\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right)$$

$$p(x \mid \mu_A, \Sigma_A) = \frac{1}{2\pi \times 0,011085} \exp\left(-\frac{1}{2}\left(x - \begin{bmatrix} 0,24 \\ 0,52 \end{bmatrix}\right)^T \cdot \begin{bmatrix} 273,4375 & -78,125 \\ -78,125 & 52,0833 \end{bmatrix} \cdot \left(x - \begin{bmatrix} 0,24 \\ 0,52 \end{bmatrix}\right)\right)$$

$p(y_1, y_2 \mid B) \sim N(\mu_B, \Sigma_B)$

$$\mu_B = \begin{bmatrix} 0,5925 \\ 0,3275 \end{bmatrix} \qquad \Sigma = \begin{bmatrix} 0,02289 & -0,009758 \\ -0,009758 & 0,03149 \end{bmatrix} \qquad |\Sigma| \approx 0,00062567$$

$$\Sigma^{-1} = \begin{bmatrix} 50,3325 & 15,59657 \\ 15,59657 & 36,5873 \end{bmatrix}$$

$$p(x \mid \mu_B, \Sigma_B) = \frac{1}{2\pi \times 0,0250134} \exp\left(-\frac{1}{2}\left(x - \begin{bmatrix} 0,5925 \\ 0,3275 \end{bmatrix}\right)^T \begin{bmatrix} 50,3326 & 15,5966 \\ 15,5966 & 36,5873 \end{bmatrix}\left(x - \begin{bmatrix} 0,5925 \\ 0,3275 \end{bmatrix}\right)\right)$$

$\{y_3, y_4\}$

$P(y_3 = 1, y_4 = 1 \mid A) = 1/3 \qquad P(y_3 = 0, y_4 = 0 \mid A) = 0$

$P(y_3 = 1, y_4 = 1 \mid A) = 0 \qquad P(y_3 = 0, y_4 = 0 \mid B) = 2/4 = 1/2$

$P(y_3 = 1, y_4 = 0 \mid A) = 1/3 \qquad P(y_3 = 0, y_4 = 1 \mid A) = 1/3$

$P(y_3 = 1, y_4 = 0 \mid B) = 1/4 \qquad P(y_3 = 0, y_4 = 1 \mid B) = 1/4$

$\{y_5\}$

$P(y_5 = 0 \mid A) = 1/3 \qquad P(y_5 = 1 \mid A) = 1/3 \qquad P(y_5 = 2 \mid A) = 1/3$

$P(y_5 = 0 \mid B) = 1/4 \qquad P(y_5 = 1 \mid B) = 1/2 \qquad P(y_5 = 2 \mid B) = 1/4$

b. Under a MAP assumption, classify each testing observation showing all your calculus.

MAP assumption: $h_{MAP} = \underset{h}{\arg\max}\ P(D|h)\,P(h)$

$\hookrightarrow p(D|h) = \prod p(y_i|h)$ pois os conjuntos são independentes

$$P(x_8|A) = p(y_3=0, y_4=1|A)\,p(y_5=0|A)\,p(y_1=0{,}38, y_2=0{,}52|A) =$$

$$= \frac{1}{3} \times \frac{1}{3} \times \frac{1}{2\pi \times 0{,}011085} \exp\left(-\frac{1}{2}\begin{bmatrix}0{,}14 & 0\end{bmatrix}\begin{bmatrix}273{,}4375 & -78{,}125 \\ -78{,}125 & 52{,}0833\end{bmatrix}\begin{bmatrix}0{,}14 \\ 0\end{bmatrix}\right) =$$

$$= \frac{1}{9} \times \frac{1}{2\pi \times 0{,}011085} e^{-2{,}6796875} \simeq 0{,}10941$$

$$P(x_8|A)\,p(A) = 0{,}10941 \times \frac{3}{7} = 0{,}04689$$

$$P(x_8|B) = p(y_3=0, y_4=1|B)\,p(y_5=0|B)\,p(y_1=0{,}38, y_2=0{,}52|A) =$$

$$= \frac{1}{4} \times \frac{1}{4} \times 1{,}96237 \simeq 0{,}12265$$

$$p(x_8|B)\,p(B) = 0{,}12265 \times \frac{4}{7} \simeq 0{,}07008$$

$$p(A|x_8) = \frac{p(x_8|A)\,p(A)}{p(x_8|A)p(A) + p(x_8|B)p(B)} = \frac{0{,}04689}{0{,}04689 + 0{,}07008} \simeq 0{,}40087$$

$$P(B|x_8) = 1 - p(A|x_8) = 0{,}59913$$

$h_{MAP} = B$

$$p(x_9|A) = p(y_3=0, y_4=1|A)\,p(y_5=1|A)\,p(y_1=0{,}42, y_2=0{,}59|A) =$$

$$= \frac{1}{3} \times \frac{1}{3} \times 0{,}40307 \simeq 0{,}04478$$

$$P(x_9|B) = p(y_3=0, y_4=1|B)\,p(y_5=1|B)\,p(y_1=0{,}42, y_2=0{,}59|B) =$$

$$= \frac{1}{4} \times \frac{1}{2} \times 1{,}72857 = 0{,}21607$$

$p(x_9|A)\,p(A) = 0{,}04478 \times \frac{3}{7} \approx 0{,}01919$

$P(x_9|B)\,p(B) = 0{,}21607 \times \frac{4}{7} \simeq 0{,}12347$

$$P(A|x_9) = \frac{p(x_9|A)p(A)}{p(x_9|A)p(A) + p(x_9|B)p(B)} = \frac{0{,}01919}{0{,}01919 + 0{,}12347} \approx 0{,}13451$$

$P(B|x_9) = 1 - 0{,}13451 = 0{,}86549 \qquad h_{MAP} = B$

c. Consider that the default decision threshold of $\theta = 0.5$ can be adjusted according to

$$f(x|\theta) = \begin{cases} A & P(A|x) > \theta \\ B & otherwise \end{cases}$$

Under a maximum likelihood assumption, what thresholds optimize testing accuracy?

ML assumption: $h_{ML} = \underset{h}{\operatorname{argmax}}\ P(D|h) \Rightarrow$ assume que $P(A) = P(B)$
logo $P(D|h) \propto P(h|D)$

$P(x_8|A) = 0{,}10941 \qquad p(x_9|A) = 0{,}04478$

$P(x_8|B) = 0{,}12265 \qquad p(x_9|B) = 0{,}21607$

$$P(A|x_8) = \frac{P(x_8|A)}{P(x_8|A) + P(x_8|B)} = \frac{0{,}10941}{0{,}10941 + 0{,}12265} \simeq 0{,}47147$$

$P(B|x_8) = 1 - P(A|x_8) = 1 - 0{,}47147 = 0{,}52853 \qquad h_{ML} = B$

$$P(A|x_9) = \frac{0{,}04478}{0{,}04478 + 0{,}21607} \simeq 0{,}17167$$

$h_{ML} = B$

$P(B|x_9) = 1 - 0{,}17167 = 0{,}82833$

$f(x_8|\theta = 0{,}5) = B \qquad f(x_9|\theta = 0{,}5) = B \qquad P(\text{certo}) = 50\%$

$f(x_8|\theta \in [0{,}17167, 0{,}47147[) = A$

$f(x_9|\theta \in [0{,}17167, 0{,}47147[) = B$

$P(\text{certo}) = 100\%$

Logo θ no intervalo $0{,}17167 \leq \theta < 0{,}47147$ otimiza a testing accuracy para as observações teste.

2. Let y_1 be the target numeric variable, y_2-y_6 be the input variables where y_2 is binarized under an equal-width (equal-range) discretization. For the evaluation of regressors, consider a 3-fold cross-validation over the full dataset (x_1- x_9) without shuffling the observations.

   a. Identify the observations and features per data fold after the binarization procedure.

a. Binarização

$D_{y_2} = [0,1]$ $\quad \frac{1-0}{2} = 0,5$

$\underset{[0;0,5[}{\text{bin}_0} = \{0,11; 0,28; 0,36; 0,39; 0,489$

$\underset{[0,5;1]}{\text{bin}_1} = \{0,52; 0,53; 0,59; 0,724$

novo $y_2 = [0, 0, 1, 0, 0, 0, 1, 1, 1]$

3-fold cross-validation
- fold 1 : $x_1 - x_3$
- fold 2 : $x_4 - x_6$
- fold 3 : $x_7 - x_9$

Iteração 1

D	y_1	y_2	y_3	y_4	y_5	y_6	
x_1	0,24	0	1	1	0	A	
x_2	0,16	0	1	0	1	A	treino
x_3	0,32	1	0	1	2	A	
x_4	0,54	0	0	0	1	B	
x_5	0,66	0	0	0	0	B	treino
x_6	0,76	0	1	0	2	B	
x_7	0,41	1	0	1	1	B	
x_8	0,38	1	0	1	0	A	teste
x_9	0,42	1	0	1	1	B	

fold 2 Iteração 2

D	y_1	y_2	y_3	y_4	y_5	y_6	
x_1	0,24	0	1	1	0	A	
x_2	0,16	0	1	0	1	A	treino
x_3	0,32	1	0	1	2	A	
x_4	0,54	0	0	0	1	B	
x_5	0,66	0	0	0	0	B	teste
x_6	0,76	0	1	0	2	B	
x_7	0,41	1	0	1	1	B	
x_8	0,38	1	0	1	0	A	treino
x_9	0,42	1	0	1	1	B	

Iteração 3

D	y_1	y_2	y_3	y_4	y_5	y_6	
x_1	0,24	0	1	1	0	A	
x_2	0,16	0	1	0	1	A	teste
x_3	0,32	1	0	1	2	A	
x_4	0,54	0	0	0	1	B	
x_5	0,66	0	0	0	0	B	treino
x_6	0,76	0	1	0	2	B	
x_7	0,41	1	0	1	1	B	
x_8	0,38	1	0	1	0	A	treino
x_9	0,42	1	0	1	1	B	

b. Consider a distance-weighted kNN with $k = 3$, Hamming distance (d), and $1/d$ weighting. Compute the MAE of this kNN regressor for the 1^{st} iteration of the cross-validation (i.e. train observations have the lower indices).

b. $x_1 - x_6$ – treino

$x_7 - x_9 \rightarrow$ test

Hamming distance = número de atributos diferentes entre dois valores

$d(x_7, x_1) = 4$
$d(x_7, x_2) = 4$
$\rightarrow d(x_7, x_3) = 2$
$\rightarrow d(x_7, x_4) = 2$
$\rightarrow d(x_7, x_5) = 3$
$d(x_7, x_6) = 4$

$\rightarrow d(x_8, x_1) = 2$
$d(x_8, x_2) = 4$
$\rightarrow d(x_8, x_3) = 1$
$d(x_8, x_4) = 4$
$\rightarrow d(x_8, x_5) = 3$
$d(x_8, x_6) = 5$

$d(x_9, x_1) = 4$
$d(x_9, x_2) = 4$
$\rightarrow d(x_9, x_3) = 2$
$\rightarrow d(x_9, x_4) = 2$
$\rightarrow d(x_9, x_5) = 3$
$d(x_9, x_6) = 4$

3 vizinhos mais próximos de x_7: x_3, x_4, x_5

3 vizinhos mais próximos de x_8: x_1, x_3, x_5

3 vizinhos mais próximos de x_9: x_3, x_4, x_5

Distancias pesadas:

$$w_i = \frac{1}{d(x, x_i)}$$

x_7: $w_3 = \frac{1}{2} = 0{,}5$, $w_4 = \frac{1}{2} = 0{,}5$, $w_5 = \frac{1}{3}$

x_8: $w_1 = \frac{1}{2} = 0{,}5$, $w_3 = \frac{1}{1} = 1$, $w_5 = \frac{1}{3}$

x_9: $w_3 = \frac{1}{2} = 0{,}5$, $w_4 = \frac{1}{2} = 0{,}5$, $w_5 = \frac{1}{3}$

$$\hat{z}_i = \frac{\sum w_i \times \text{output}(x_i)}{\sum w_i}$$

$$\hat{z}_7 = \frac{0.5 \times 0{,}32 + 0{,}5 \times 0{,}54 + \frac{1}{3} \times 0{,}66}{0{,}5 + 0{,}5 + \frac{1}{3}} = 0{,}4875$$

$$\hat{z}_8 = \frac{0{,}5 \times 0{,}24 + 1 \times 0{,}32 + \frac{1}{3} \times 0{,}66}{0{,}5 + 1 + \frac{1}{3}} = 0{,}36$$

$$\hat{z}_9 = \frac{0{,}5 \times 0{,}32 + 0{,}5 \times 0{,}54 + \frac{1}{3} + 0{,}66}{0{,}5 + 0{,}5 + \frac{1}{3}} = 0{,}4875$$

$$MAE = \frac{\sum |z_i - \hat{z}_i|}{n}$$

$$MAE = \frac{|0{,}41 - 0{,}4875| + |0{,}38 - 0{,}36| + |0{,}42 - 0{,}4875|}{3} = 0{,}055$$

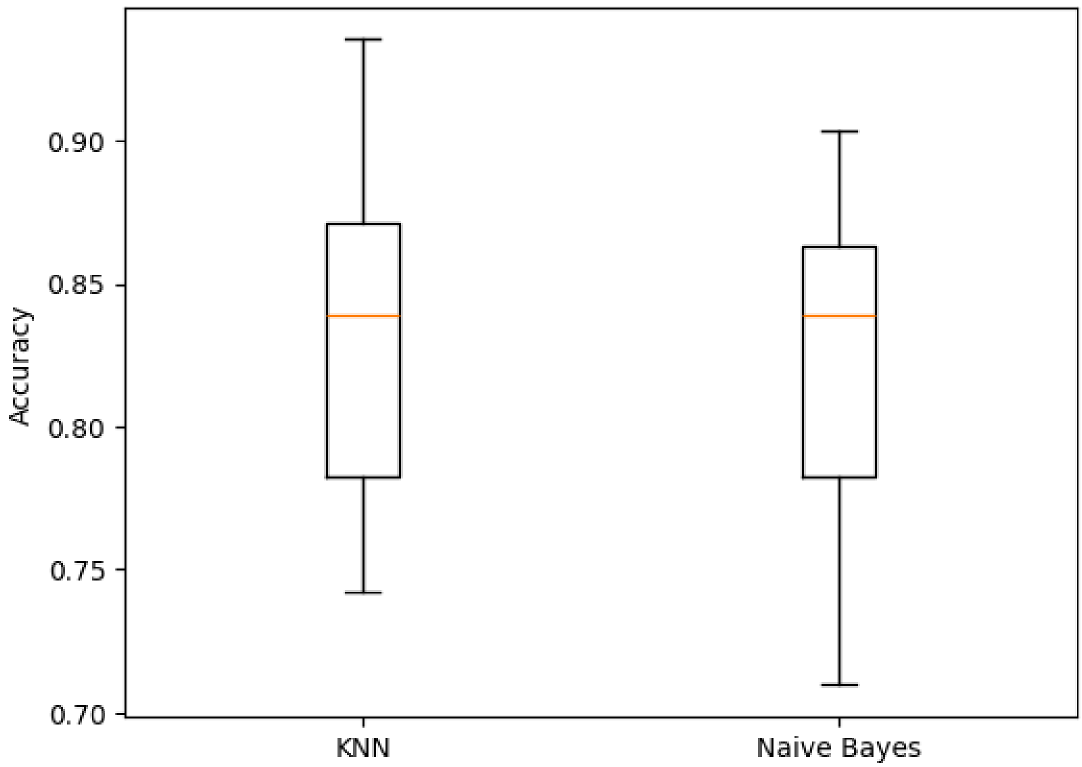
Part II: Programming

Considering the `column_diagnosis.arff` dataset available at the course webpage's homework tab. Using `sklearn`, apply a 10-fold stratified cross-validation with shuffling ($random\_state = 0$) for the assessment of predictive models along this section.

1. Compare the performance of kNN with $k = 5$ and naïve Bayes with Gaussian assumption (consider all remaining parameters for each classifier as `sklearn`'s default):

   a. Plot two boxplots with the fold accuracies for each classifier.

```python
from sklearn.model_selection import StratifiedKFold
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
import pandas as pd
from scipy.io.arff import loadarff
import matplotlib.pyplot as plt
import numpy as np

data = loadarff('column_diagnosis.arff')
df = pd.DataFrame(data[0])
y = df['class'].astype(str)
x = df.drop('class', axis=1).astype(np.float64)

skf = StratifiedKFold(n_splits=10, shuffle=True, random_state=0)

knn = KNeighborsClassifier(n_neighbors=5)
naive_bayes = GaussianNB()

knn_scores = []
naive_bayes_scores = []
for train_index, test_index in skf.split(x, y):
        x_train, x_test = df.iloc[train_index, :-1], df.iloc[test_index, :-1]
        y_train, y_test = df.iloc[train_index, -1], df.iloc[test_index, -1]
        x_train = x_train.astype(np.float64)
        x_test = x_test.astype(np.float64)
        y_train = y_train.astype(str)
        y_test = y_test.astype(str)
        knn.fit(x_train, y_train)
        knn_scores.append(knn.score(x_test, y_test))
        naive_bayes.fit(x_train, y_train)
        naive_bayes_scores.append(naive_bayes.score(x_test, y_test))

plt.boxplot([knn_scores, naive_bayes_scores])
plt.xticks([1, 2], ['KNN', 'Naive Bayes'])
plt.ylabel('Accuracy')
plt.show()
```



b. Using `scipy`, test the hypothesis "*kNN* is statistically superior to naïve Bayes regarding accuracy", asserting whether is true.

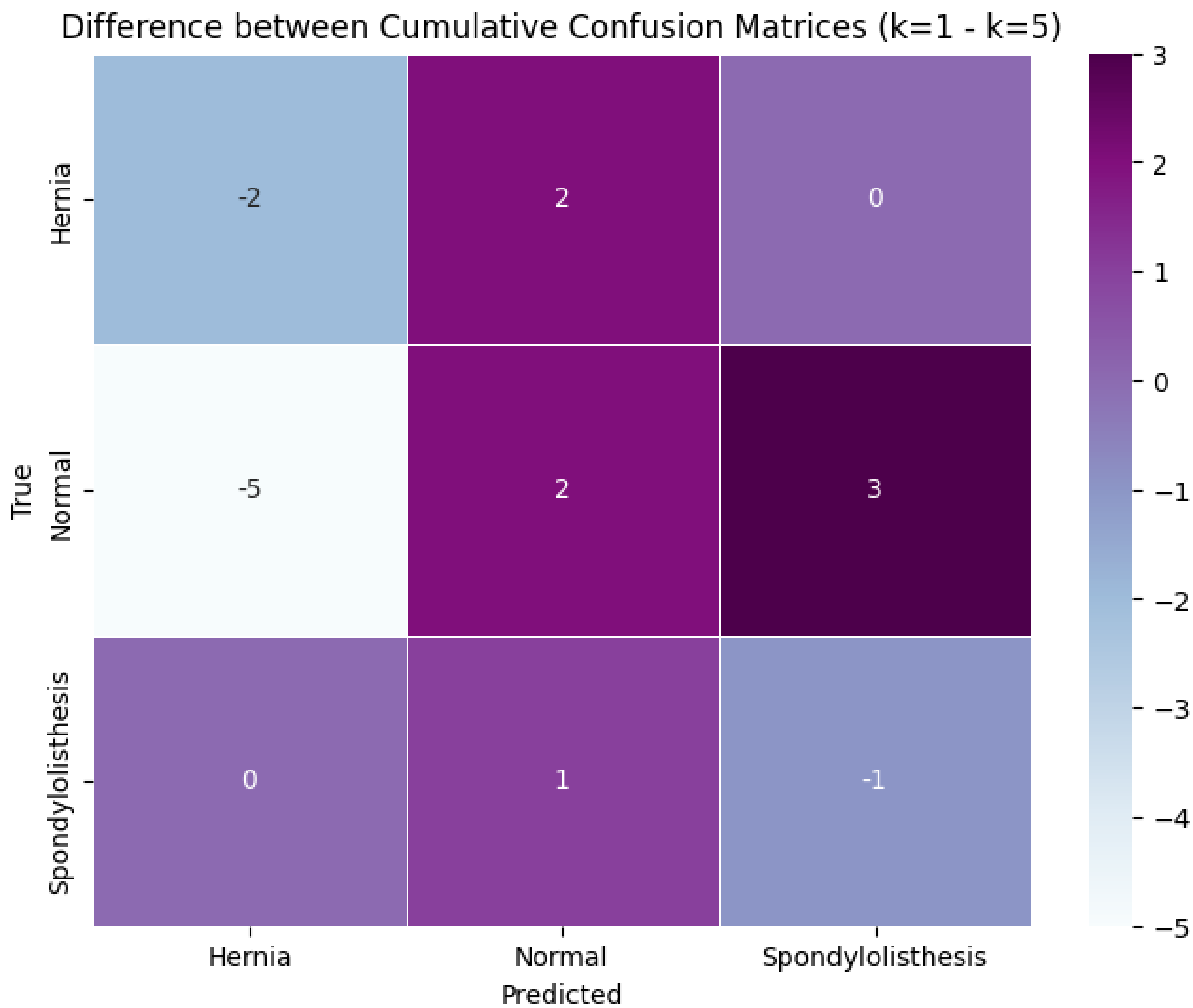
```python
from scipy.stats import ttest_rel

stat, p = ttest_rel(knn_scores, naive_bayes_scores)
print('Statistics=%.3f, p=%.3f' % (stat, p))
```

```
Statistics=0.921, p=0.381
```

O *p-value* obtido é bastante superior a 0.1 e a estatística bastante próxima de 1. Assim sendo, não podemos rejeitar a possibilidade de as amostras serem estatisticamente equivalentes, ou seja, a afirmação não é verdadeira.

2. Consider two kNN predictors with $k = 1$ and $k = 5$ (uniform weights, Euclidean distance, all remaining parameters as default). Plot the differences between the two cumulative confusion matrices of the predictors. Comment.

```python
from sklearn.metrics import confusion_matrix
import seaborn as sns

knn1 = KNeighborsClassifier(n_neighbors=1, weights='uniform', metric='euclidean')
knn5 = KNeighborsClassifier(n_neighbors=5, weights='uniform', metric='euclidean')

skf = StratifiedKFold(n_splits=10, shuffle=True, random_state=0)

cumulative_cm1 = np.zeros((3, 3))
cumulative_cm5 = np.zeros((3, 3))


for train_index, test_index in skf.split(x, y):

        x_train, x_test = df.iloc[train_index, :-1], df.iloc[test_index, :-1]
        y_train, y_test = df.iloc[train_index, -1], df.iloc[test_index, -1]
        x_train = x_train.astype(np.float64)
        x_test = x_test.astype(np.float64)
        y_train = y_train.astype(str)
        y_test = y_test.astype(str)

        knn1.fit(x_train, y_train)
        knn5.fit(x_train, y_train)

        y_pred1 = knn1.predict(x_test)
        y_pred5 = knn5.predict(x_test)

        cm1 = confusion_matrix(y_test, y_pred1)
        cm5 = confusion_matrix(y_test, y_pred5)

        cumulative_cm1 += cm1
        cumulative_cm5 += cm5


diff_cm = cumulative_cm1 - cumulative_cm5

diff_cm = pd.DataFrame(diff_cm, index=['Hernia', 'Normal', 'Spondylolisthesis'], columns=['Hernia', 'Normal', 'Spondylolisthesis'])
plt.figure(figsize=(8, 6))
sns.heatmap(diff_cm, annot=True, cmap="BuPu", linewidths=0.5)
plt.title("Difference between Cumulative Confusion Matrices (k=1 - k=5)")
plt.ylabel('True')
plt.xlabel('Predicted')
plt.show()
```



Ao observar a matriz resultante da diferença entre as matrizes de confusão dos dois classificadores, é posivel concluir que não há uma diferença significativa de um classificador para o outro, logo, o classificador com $k = 1$ é tão bom quanto o classificador com $k = 5$.

3. Considering the unique properties of `column_diagnosis`, identify three possible difficulties of naïve Bayes when learning from the given dataset.

   1- Devido à suposição de independência condicional feita pelo Naïve Bayes entre as variáveis, pode ocorrer uma redução na exatidão do modelo nos casos em que, de fato, existe uma dependência. Isso é evidenciado pelo fato de que as características biomédicas no dataset fornecido estão inter-relacionadas de alguma forma, o que viola essa suposição.

   2- O dataset apresenta uma reduzida dimensionalidade, podendo levar a uma redução da exatidão.

   3- Observa-se um desiquilíbrio no número de observações para cada classe, nomeadamente, existem 150 observações classificadas como ”spondylolisthesis” e apenas 60 como ”hernia”. Isso pode dificultar a aprendizgem e classificação correta das classes minoritárias, resultando em um viés em direção às classes maioritárias.