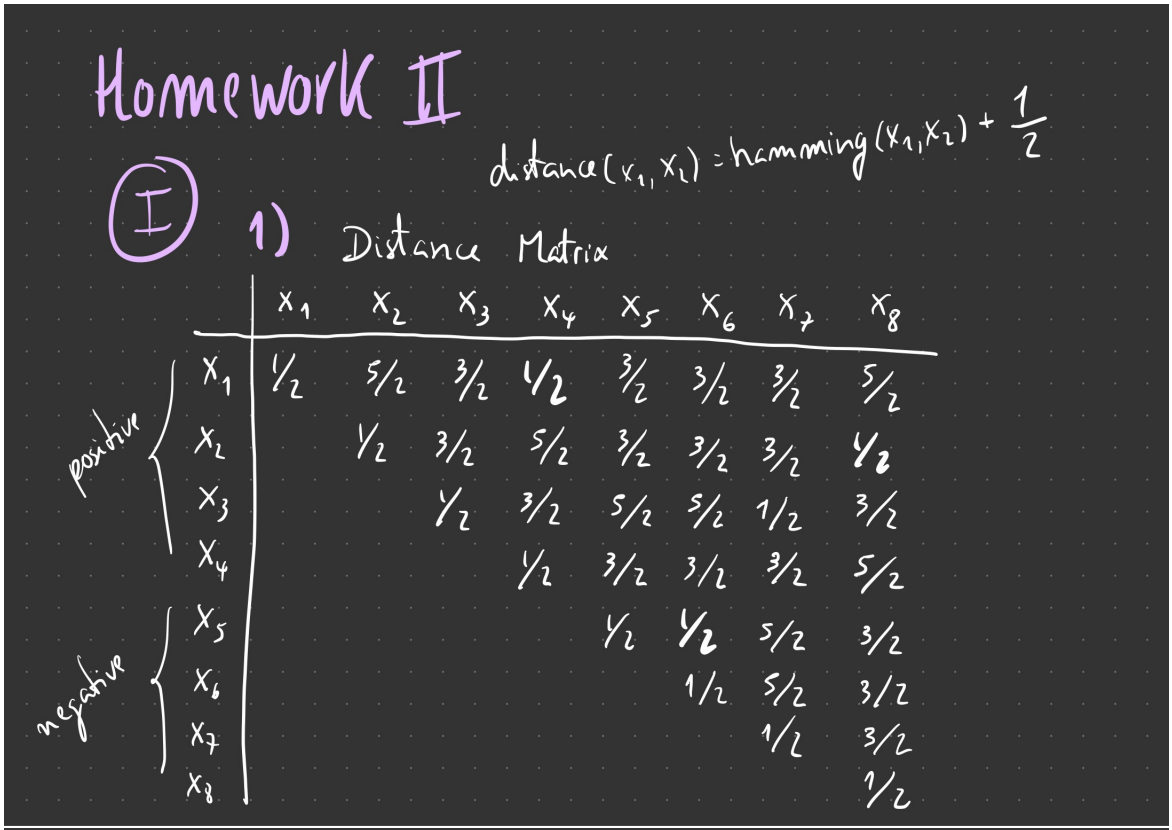
Aprendizagem 2022

Homework II – Group 58

Part I: Pen and paper

1.



	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
x_1	1/2	5/2	3/2	1/2	3/2	3/2	3/2	5/2
x_2		1/2	3/2	5/2	3/2	3/2	3/2	1/2
x_3			1/2	3/2	5/2	5/2	1/2	3/2
x_4				1/2	3/2	3/2	3/2	5/2
x_5					1/2	1/2	5/2	3/2
x_6						1/2	5/2	3/2
x_7							1/2	3/2
x_8								1/2

Weight Matrix

		x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
positive	x_1	2	2/5	2/3	2	2/3	2/3	2/3	2/5
	x_2	2/5	2	2/3	2/5	2/3	2/3	2/3	2
	x_3	2/3	2/3	2	2/3	2/5	2/5	2	2/3
	x_4	2	2/5	2/3	2	2/3	2/3	2/3	2/5
negative	x_5	2/3	2/3	2/5	2/3	2	2	2/5	2/3
	x_6	2/3	2/3	2/5	2/3	2	2	2/5	2/3
	x_7	2/3	2/3	2/3	2/3	2/5	2/5	2	2/3
	x_8	2/5	2	2/3	2/5	2/3	2/3	2/3	2

$Snn(x_1) = [x_4, x_3, x_5, x_6, x_7]$

$wa(x_1) = \left(\frac{2}{3}+2\right) \times (+1) + \left(\frac{2}{3}+\frac{2}{3}+\frac{2}{3}\right) \times (-1) = \frac{8}{3} - \frac{6}{3} = \frac{2}{3} > 0$

$Class(x_1) = P \quad Class\ Predicted(x_1) = P$

$Snn(x_2) = [x_8, x_3, x_5, x_6, x_7]$

$wa(x_2) = \frac{2}{3} \times (+1) + \left(\frac{2}{3}+\frac{2}{3}+\frac{2}{3}+2\right) \times (-1) = \frac{2}{3} - \frac{12}{3} = -\frac{10}{3} < 0$

$Class(x_2) = P \quad class\ Predicted(x_2) = N$

$Snn(X_3) = [X_7, X_1, X_2, X_4, X_8]$

$wa(X_3) = \left(\frac{2}{3} + \frac{2}{3} + \frac{2}{3}\right) \times (+1) + \left(2 + \frac{2}{3}\right) \times (-1) = \frac{6}{3} - \frac{8}{3} = -\frac{2}{3} < 0$

$Class(X_3) = P \quad Class\ Predicted(X_3) = N$

$Snn(X_4) = [X_1, X_3, X_5, X_6, X_7]$

$wa(X_4) = \left(2 + \frac{2}{3}\right) \times (+1) + \left(\frac{2}{3} + \frac{2}{3} + \frac{2}{3}\right) \times (-1) = \frac{8}{3} - \frac{6}{3} = \frac{2}{3} > 0$

$Class(X_4) = P \quad Class\ Predicted(X_4) = P$

$Snn(X_5) = [X_6, X_1, X_2, X_4, X_8]$

$wa(X_5) = \left(\frac{2}{3} + \frac{2}{3} + \frac{2}{3}\right) \times (+1) + \left(2 + \frac{2}{3}\right) \times (-1) = \frac{6}{3} - \frac{8}{3} = -\frac{2}{3} < 0$

$Class(X_5) = N \quad Class\ Predicted(X_5) = N$

$Snn(X_6) = [X_5, X_1, X_2, X_4, X_8]$

$wa(X_6) = \left(\frac{2}{3} + \frac{2}{3} + \frac{2}{3}\right) \times (+1) + \left(2 + \frac{2}{3}\right) \times (-1) = \frac{6}{3} - \frac{8}{3} = -\frac{2}{3} < 0$

$Class(X_6) = N \quad Class\ Predicted(X_6) = N$

$Snn(X_7) = [X_3, X_1, X_2, X_4, X_8]$

$wa(X_7) = \left(\frac{2}{3} + \frac{2}{3} + \frac{2}{3} + \frac{2}{3}\right) \times (+1) + \frac{2}{3} \times (-1) = \frac{8}{3} - \frac{2}{3} = 2 > 0$

$Class(X_7) = N \quad Class\ Predicted(X_7) = P$

$Snn(X_8) = [X_2, X_3, X_5, X_6, X_7]$

$wa(X_8) = \left(2 + \frac{2}{3}\right) \times (+1) + \left(\frac{2}{3} + \frac{2}{3} + \frac{2}{3}\right) \times (-1) = \frac{8}{3} - \frac{6}{3} = \frac{2}{3} > 0$

$Class(X_8) = N \quad Class\ Predicted(X_8) = P$

Leave One Out Execution

$D_{train} = \{x_2, \ldots, x_8\}$ $D_{test} = \{x_1\}$
target prediction = P
target true = P

$D_{train} = \{x_1, x_3, \ldots x_8\}$ $D_{test} = \{x_2\}$
target prediction = N
target true = P

$D_{train} = \{x_1, x_2, x_4 \ldots x_8\}$ $D_{test} = \{x_3\}$
target prediction = N
target true = P

$D_{train} = \{x_1, \ldots x_3, x_5, \ldots x_8\}$ $D_{test} = \{x_4\}$
target prediction = P
target true = P

$D_{train} = \{x_1, \ldots, x_4, x_6, \ldots x_8\}$ $D_{test} = \{x_5\}$
target prediction = N
target true = N

$D_{train} = \{x_1, \ldots, x_5, x_7, x_8\}$ $D_{test} = \{x_6\}$
target prediction = N
target true = N

$D_{train} = \{x_1, \ldots, x_6, x_8\}$ $D_{test} = \{x_7\}$
target prediction = P
target true = N
predictions = [P, N, N, P, N, N, P, P]
truths = [P, P, P, P, N, N, N, N]

$D_{train} = \{x_1, \ldots, x_7\}$ $D_{test} = \{x_8\}$
target prediction = P
target true = N

Confusion Matrix

predicted \ truth	P	N
P	2	2
N	2	2

$TP = 2$ $FP = 2$
$TF = 2$ $FN = 2$

4

$$recall = \frac{TP}{TP + FN} = \frac{2}{2+2} = \frac{1}{2} = 0.5$$

2.

(2) $P(y|x) = \frac{P(x|y)P(y)}{P(x)}$

posterior

all taken from data directly

	y_1	y_2	Class	y_3	Class
x_1	A	0	P	1,2	P
x_2	B	1	P	0,8	P
x_3	A	1	P	0,5	P
x_4	A	0	P	0,9	P
x_5	B	0	N	1	N
x_6	B	0	N	0,9	N
x_7	A	1	N	1,2	N
x_8	B	1	N	0,8	N
x_9	B	0	P	0,8	P

→ y_1 and y_2 are dependent

→ y_3 is independent from $\{y_1, y_2\}$

→ y_3 is normally distributed

P → Positive N → Negative

$$P(P) = \frac{\#(\text{class} = P)}{\#\text{total}} = \frac{5}{9} \qquad P(N) = \frac{\#(\text{class} = N)}{\#\text{total}} = \frac{4}{9}$$

$$P(y_1 = A, y_2 = 0 \mid P) = \frac{\#(y_1 = A, y_2 = 0, \text{Class} = P)}{\#(\text{class} = P)} = \frac{2}{5}$$

$$P(y_1 = A, y_2 = 1 \mid P) = \frac{\#(y_1 = A, y_2 = 1, \text{Class} = P)}{\#(\text{class} = P)} = \frac{1}{5}$$

$$P(y_1 = B, y_2 = 0 \mid P) = \frac{\#(y_1 = B, y_2 = 0, \text{Class} = P)}{\#(\text{class} = P)} = \frac{1}{5}$$

$$P(y_1=B, y_2=1 \mid P) = \frac{\#(y_1=B, y_2=1, Class=P)}{\#(Class=P)} = \frac{1}{5}$$

$$P(y_1=A, y_2=0 \mid N) = \frac{\#(y_1=A, y_2=0, Class=N)}{\#(Class=N)} = 0$$

$$P(y_1=A, y_2=1 \mid N) = \frac{\#(y_1=A, y_2=1, Class=N)}{\#(Class=N)} = \frac{1}{4}$$

$$P(y_1=B, y_2=0 \mid N) = \frac{\#(y_1=B, y_2=0, Class=N)}{\#(Class=N)} = \frac{2}{4} = \frac{1}{2}$$

$$P(y_1=B, y_2=1 \mid N) = \frac{\#(y_1=B, y_2=1, Class=N)}{\#(Class=N)} = \frac{1}{4}$$

$P(x \mid \mu, \sigma^2, P)$

$P(x \mid \mu, \sigma^2, N)$

$$\mu_P = \frac{1,2+0,8+0,5+0,9+0,8}{5} = \frac{4,2}{5} = 0,84$$

$$\mu_N = \frac{1+0,9+1,2+0,8}{4} = \frac{3,9}{4} = 0,975$$

$$\sigma_P^2 = \frac{1}{N-1}\sum_{i=0}^{N}(y_{3i}-\mu_P)^2 = \frac{1}{5-1}\Big[(1.2-0.84)^2+(0.8-0.84)^2+(0.5-0.84)^2+ \\ +(0.9-0.84)^2+(0.8-0.84)^2\Big] =$$

$$= \frac{1}{4}(0.1296+0.0016+0.1156+0.0036+0.0016) = \frac{1}{4}(0.252) = 0.063$$

$$\sigma_N^2 = \frac{1}{N-1}\sum_{i=0}^{N}(y_{3i}-\mu_N)^2 = \frac{1}{4-1}\Big[(1-0,975)^2+(0.9-0.975)^2+(1.2-0.975)^2+(0,8-0.975)^2\Big]$$

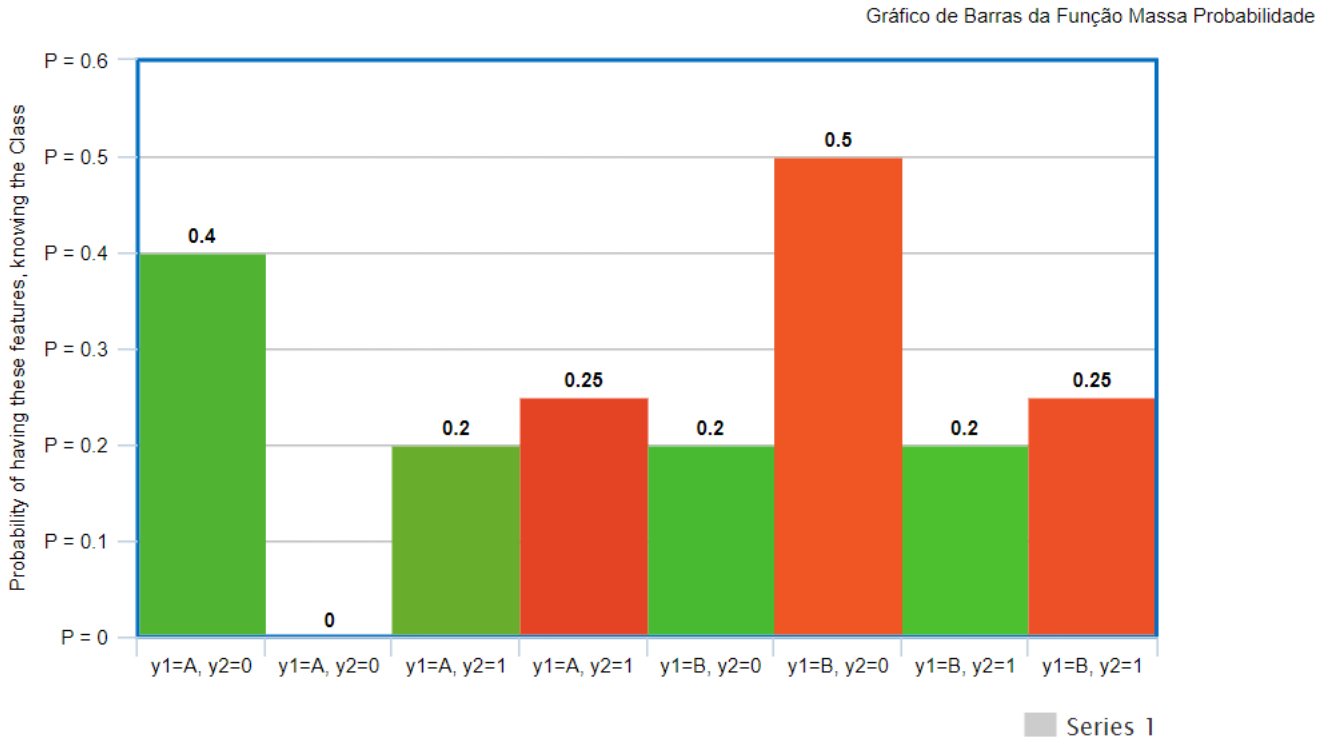
$$= \frac{1}{3}\Big[0.000625+0.005625+0.050625+0.030625\Big] =$$

$$= \frac{1}{3} \times 0.0875 \simeq 0.0292$$

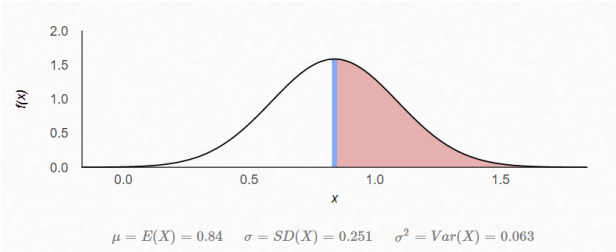
$$P(y_3 = x \mid \mu = 0.84, \sigma^2 = 0.063, P) = \frac{1}{\sqrt{2\pi \times 0.063}} e^{-\frac{(x-0.84)^2}{2 \times 0.063}}$$

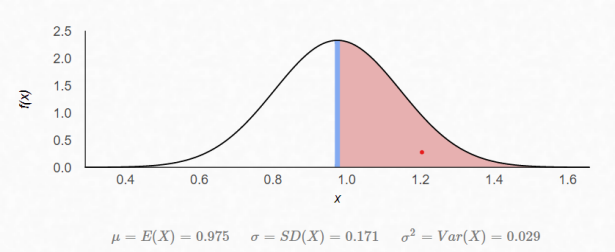
$$P(y_3 = x \mid \mu = 0.975, \sigma^2 = 0.0292, N) = \frac{1}{\sqrt{2\pi \times 0.0292}} e^{-\frac{(x-0.975)^2}{2 \times 0.0292}}$$

The probability mass function for the Bayesian classifier, the red boxes mean $P(y_1, y_2|N)$ and the green mean $P(y_1, y_2|P)$, for y_1 and y_2 are:



The normal distributions of $P(y_3|P)$ and $P(y_3|N)$ are, respectively:





3.

(3) $P(P|y_1,y_2,y_3) = \frac{P(y_1,y_2,y_3|P) \times P(P)}{P(y_1,y_2,y_3)} = \frac{P(y_1,y_2|P) \times P(y_3|P) \times P(P)}{P(y_1,y_2,y_3)}$

observação 1: (A, 1, 0.8)

$P(\text{Positive}|y_1=A, y_2=1, y_3=0.8) = P(\text{Positive}) \times \frac{P(y_1=A, y_2=1, y_3=0.8|P)}{P(y_1=A, y_2=1, y_3=0.8)}$

$P(\text{Positive}) = \frac{5}{9}$

$P(y_1=A, y_2=1|P) = 0.2 \quad P(y_1=A, y_2=1|N) = 0.25$

$P(y_1,y_2,y_3) = P(y_1,y_2|P) \times P(y_3|P) \times P(P) + P(y_1,y_2|N) \times P(y_3|N) \times P(N)$

$= 0.2 \times 1.5694 \times \frac{5}{9} + 0.25 \times 1.3819 \times \frac{4}{9} = 0.17438 + 0.15354$

$= 0.32792$

$P(P|y_1,y_2,y_3) = \frac{0.2 \times 0.8719}{0.32792} = 0.53178$

observação 2: (B, 1, 1)

$P(\text{Positive}|y_1=B, y_2=1, y_3=1) = \frac{P(y_1,y_2,y_3|P) \times P(P)}{P(y_1,y_2,y_3)}$

$P(\text{Positive}) = \frac{5}{9}$

$P(y_1=B, y_2=1|P) = 0.2 \quad P(y_1=B, y_2=1|N) = 0.25$

$P(y_1,y_2,y_3) = P(y_1,y_2|P) \times P(y_3|P) \times P(P) + P(y_1,y_2|N) \times P(y_3|N) \times P(N)$

$= 0.2 \times 1.2972 \times \frac{5}{9} + 0.25 \times 2.3095 \times \frac{4}{9} =$

$= 0.14413 + 0.25661 = 0.40074$

$P(P|y_1,y_2,y_3) = \frac{0.2 \times 1.2972 \times \frac{5}{9}}{0.40074} = 0.3597$

observação 3 : (B, 0, 0.9)

$$P(P|y_1,y_2,y_3) = \frac{P(y_1,y_2,y_3|P)\times P(P)}{P(y_1,y_2,y_3)}$$

$P(y_1=B, y_2=0|P) = 0.2 \quad P(y_1=B, y_2=0|N) = 0.5$

$$P(y_1,y_2,y_3) = P(y_1,y_2|P)\times P(y_3|P)\times P(P) + P(y_1,y_2|N)\times P(y_3|N)\times P(N)$$

$$= 0.2\times 1.5446\times\frac{5}{9} + 0.5\times 2.1201\times\frac{4}{9} =$$

$$= 0.17162 + 0.47113 = 0.64275$$

$$P(P|y_1,y_2,y_3) = \frac{P(y_1,y_2|P)\times P(y_3|P)\times P(P)}{P(y_1,y_2,y_3)} = \frac{0.2\times 1.5446\times\frac{5}{9}}{0.64275} = 0.2670$$

4.

(4) values of P(Positive|x) for observations: 1 - 0.53178
2 - 0.3597
3 - 0.2670

Classification of observations w/ different thresholds:

	$\theta = 0.3$	$\theta = 0.5$	$\theta = 0.7$	actual Class
obs. 1	P	P	N	P
obs. 2	P	N	N	P
obs. 3	N	N	N	N

$$\text{accuracy} = \frac{TP+TN}{All}$$

$\theta = 0.3$

$TP = 2 \quad TN = 1 \quad All = 3 \quad accuracy = \frac{2+1}{3} = 100\%$

$\theta = 0.5$

$TP = 1 \quad TN = 1 \quad FN = 1 \quad All = 3 \quad accuracy = \frac{1+1}{3} = 66,7\%$

$\theta = 0.7$

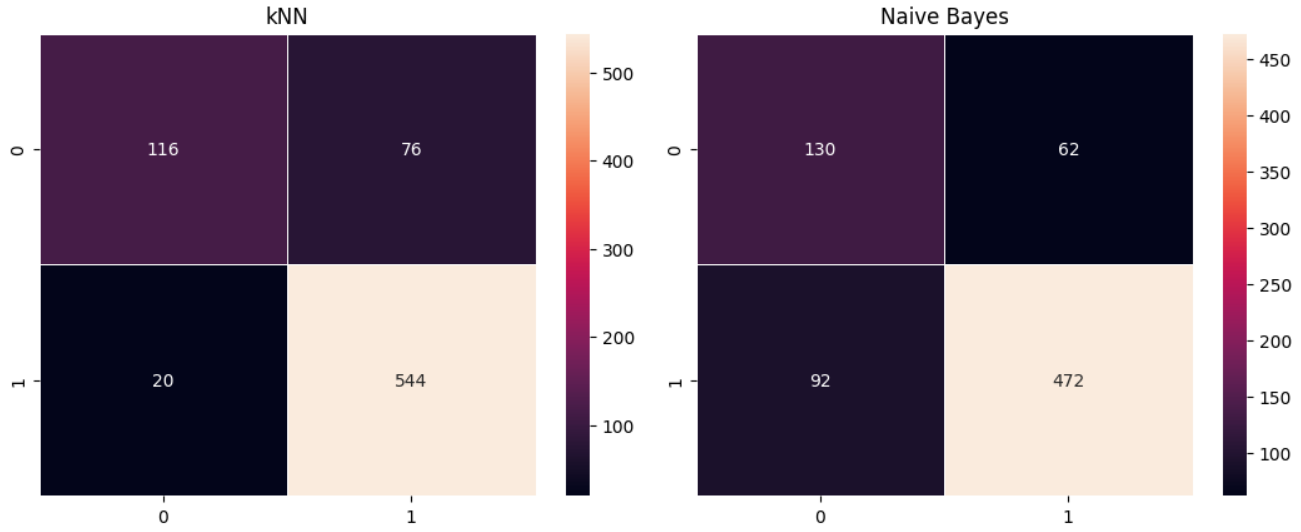
$FN = 2 \quad TN = 1 \quad accuracy = \frac{1}{3} = 33,3\%$

$$accuracy_{\theta=0.3} > accuracy_{\theta=0.5} > accuracy_{\theta=0.7}$$

The accuracy for $\theta = 0.3$ is the biggest, 100%, for our previous observations, the threshold that optimizes accuracy is $\theta = 0.3$

Part II: Programming

5.



6. $knn_{acc} > gnb_{acc}$? $pval = 0.0013168178284908 26$
   $knn_{acc} < gnb_{acc}$? $pval = 0.9986831821715092$
   $knn_{acc} \neq gnb_{acc}$? $pval = 0.002633635656981652$

   Through the $pval$ we can observe that the accuracy of $k$nn is indeed statistically superior to Naive Bayes' accuracy, with a $pval$ of less than 0.05 ($0.001 < 0.05$), which indicates strong statistical significance, so, our hypothesis is true.

7. Since, Naive Bayes is a parametric algorithm, it assumes a functional form, Gaussian distribution in this case, our data set might be non-Gaussian, hence, the poorer performance compared to $k$nn. On a problem that focuses on similarity between observations, $k$nn has an overall better performance due to its nature to locally optimize, although, outliers can worsen its performance significantly, hence, the importance of adjusting k to maximise performance.

Appendix

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from scipy import stats
from scipy.io.arff import loadarff
from sklearn.model_selection import StratifiedKFold
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, confusion_matrix
from sklearn.preprocessing import StandardScaler
from sklearn.naive_bayes import GaussianNB

cm_sum_knn = [[0, 0],[0, 0]]
cm_sum_gnb = [[0, 0],[0, 0]]
knn_acc = []
gnb_acc = []
data = loadarff('pd_speech.arff')
df = pd.DataFrame(data[0])
df['class'] = df['class'].str.decode('utf-8')
X = df.drop('class', axis=1)
y = df['class']

skf = StratifiedKFold(n_splits = 10, shuffle = True, random_state = 0)
knn = KNeighborsClassifier()
gnb = GaussianNB()

for train_index, test_index in skf.split(X, y):
     X_train, X_test, y_train, y_test = X.iloc[train_index], X.iloc[test_index],\
                                        y.iloc[train_index], y.iloc[test_index]
     scaler = StandardScaler().fit(X_train)
     X_train, X_test = scaler.transform(X_train), scaler.transform(X_test)


     knn.fit(X_train, y_train)
     y_pred_knn = knn.predict(X_test)
     cm_knn = confusion_matrix(y_test, y_pred_knn)
     knn_acc.append(accuracy_score(y_test, y_pred_knn))

     for i in range(2):
          for j in range(2):
               cm_sum_knn[i][j] += cm_knn[i][j]

     gnb.fit(X_train, y_train)
     y_pred_gnb = gnb.predict(X_test)
     cm_gnb = confusion_matrix(y_test, y_pred_gnb)
     gnb_acc.append(accuracy_score(y_test, y_pred_gnb))

     for i in range(2):
          for j in range(2):
               cm_sum_gnb[i][j] += cm_gnb[i][j]

f, ax = plt.subplots()
sns.heatmap(cm_sum_knn,annot = True, linewidths= 0.5, fmt="g")
plt.title('kNN')
plt.show()
```

```
55
56 sns.heatmap(cm_sum_gnb,annot = True, linewidths= 0.5, fmt="g")
57 plt.title('Naive Bayes')
58 plt.show()
59
60 res = stats.ttest_rel(knn_acc, gnb_acc, alternative='greater')
61 print("knn>gnb? pval=",res.pvalue)
62
63 res = stats.ttest_rel(knn_acc, gnb_acc, alternative='less')
64 print("knn<gnb? pval=",res.pvalue)
65
66 res = stats.ttest_rel(knn_acc, gnb_acc, alternative='two-sided')
67 print("knn!=gnb? pval=",res.pvalue)
```